

Renesas S5D9 用サンプル(e2studio WiFi_DHCP_TCP)の説明

(e2studio Version:2022-7 / SSP Version 2.3.0)

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ください。
- **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。

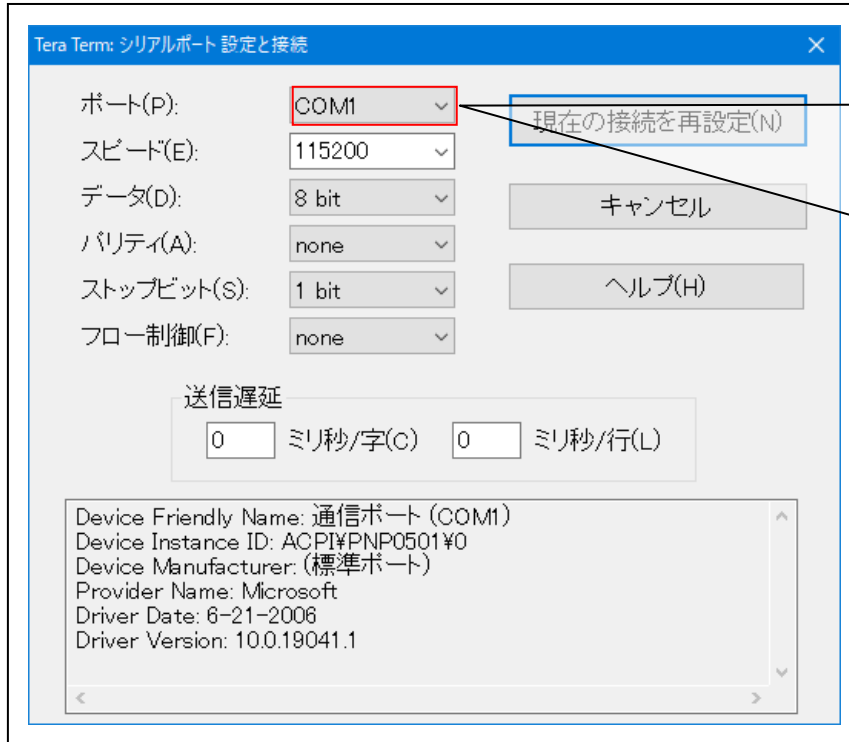
2. サンプルのプロジェクト名

ワークスペース名	概要	プロジェクト名
S5D9_e2std_WiFi_NetX_1	WiFi モジュール(GT201)を各 PMODA/B/C に接続して DHCP と TCP 通信のサンプル	WiFi_DHCP_TCP_GT202_PMODA Azure RTOS で動作 NetXDHCP Client (g_dhcp_client0) TCP 通信 (nx_tcp_socket_.....)
		WiFi_DHCP_TCP_GT202_PMODB Azure RTOS で動作 NetXDHCP Client (g_dhcp_client0) TCP 通信 (nx_tcp_socket_.....)
		WiFi_DHCP_TCP_GT202_PMODC Azure RTOS で動作 NetXDHCP Client (g_dhcp_client0) TCP 通信 (nx_tcp_socket_.....)

統合開発環境
Renesas e2studio(Version 2022-7)
SSP(Version2.3.0)

3. Tera Term Pro のインストール

- ① 「teraterm-4.106.exe」 を検索してダウンロードする。
- ② PC 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

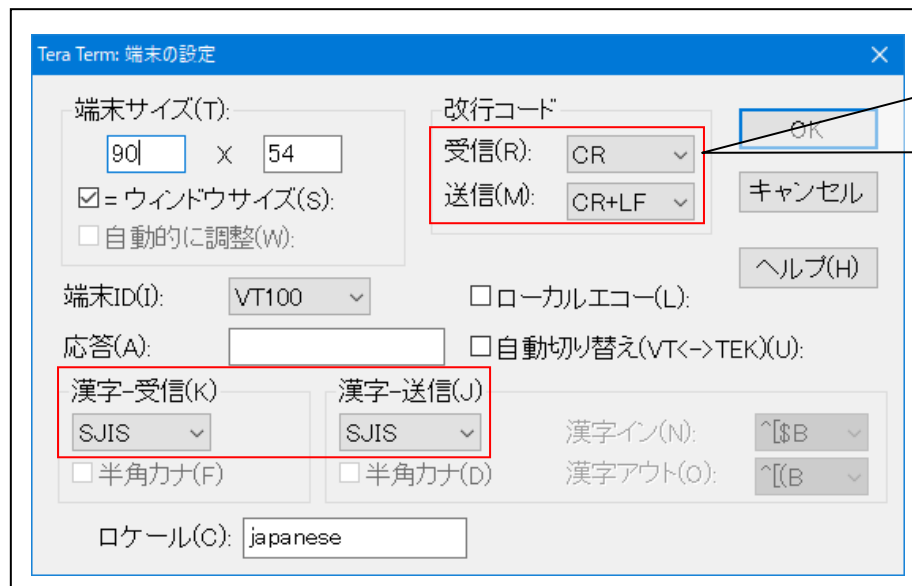
none

1bit

none

の仕様にする。

④ 端末の設定

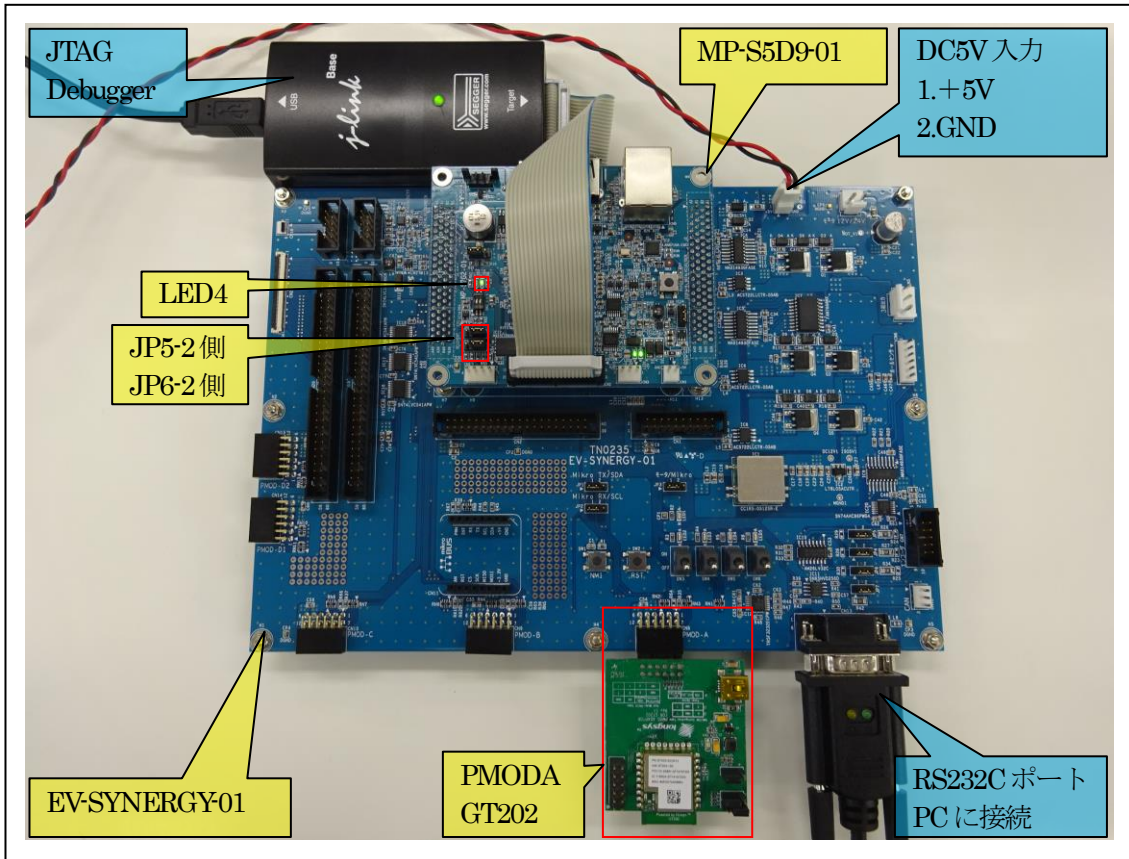


USB シリアルコンバー
タ使用時に CR コ
ードがカットされる
設定の場合は、**受
信：LF** にして下さ
い。

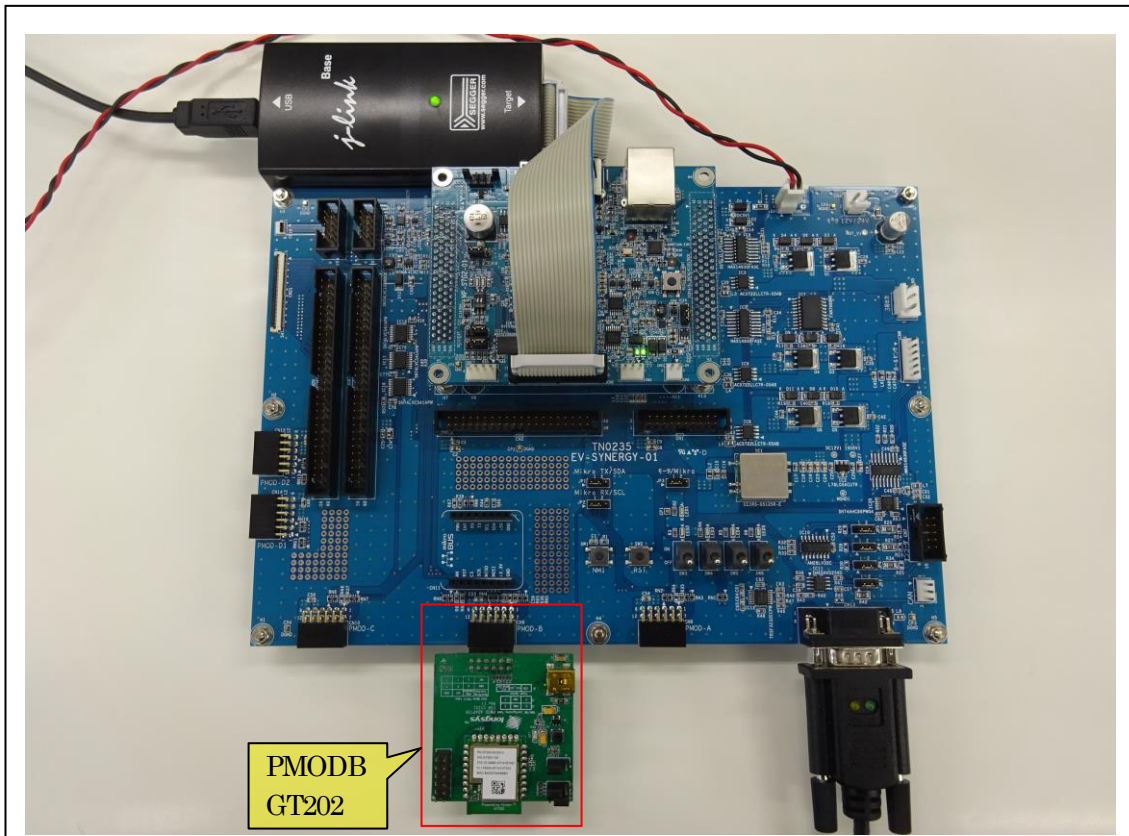
赤枠の設定にする。

4. 動作構成

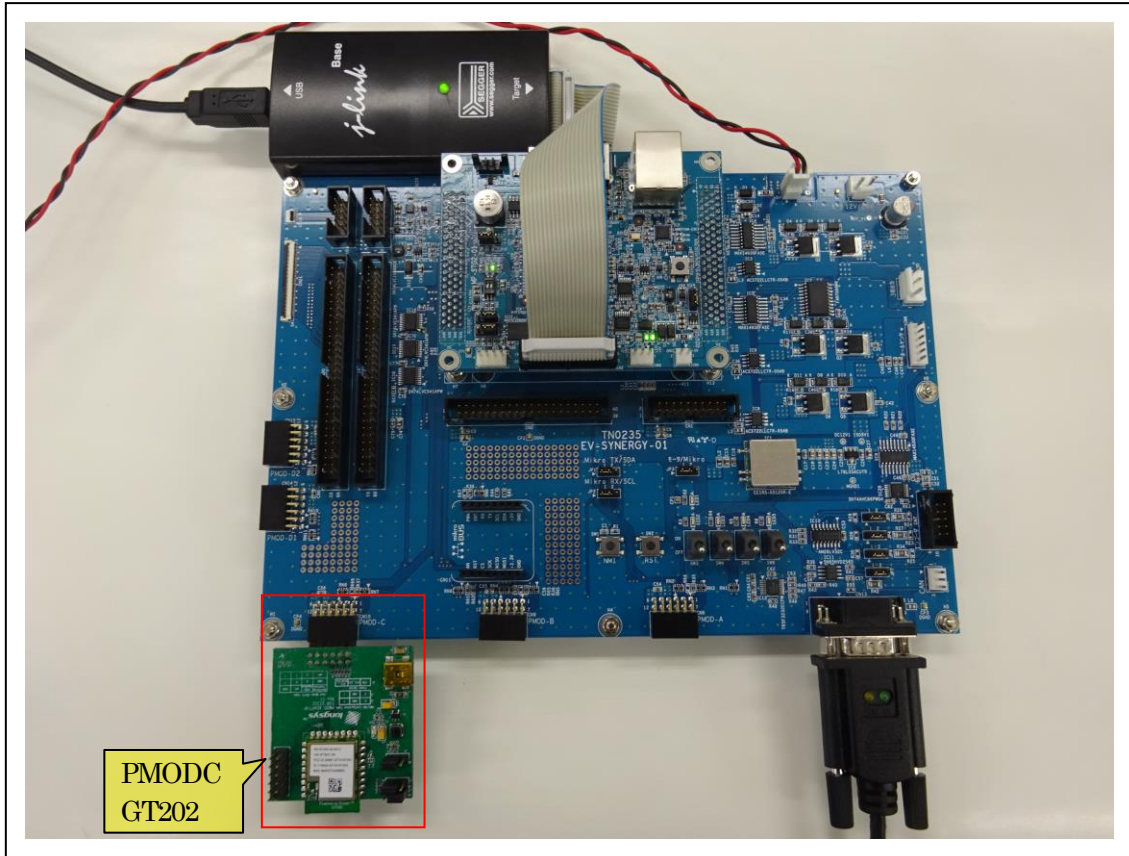
【WiFi_DHCP_TCP_GT202_PMODA】



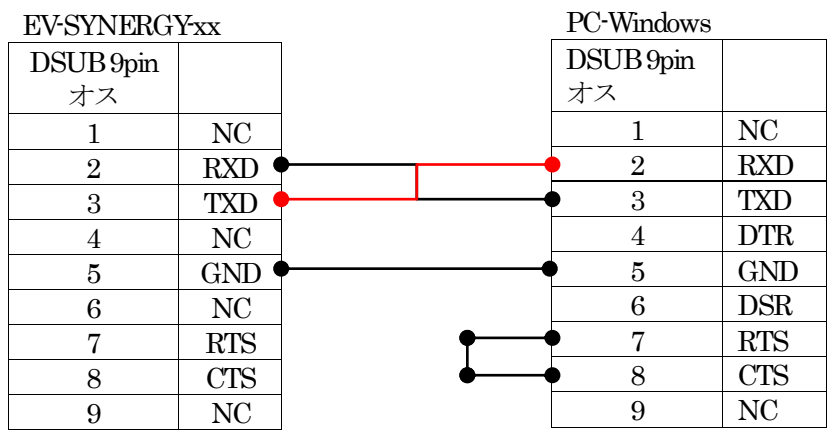
【WiFi_DHCP_TCP_GT202_PMODB】



【WiFi_DHCP_TCP_GT202_PMODC】



- 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。



5. 「S5D9_e2std_WiFi_NetX_1」 サンプルの説明

5-1. 「WiFi_DHCP_TCP_GT202_PMODA」 フォルダ構成とファイル名

S5D9_e2std_WiFi_NetX_1\WiFi_DHCP_TCP_GT202_PMODA		
Debug	WiFi_DHCP_TCP_GT202_PMODA.elf	ELF ファイル、JTAG で使用
	WiFi_DHCP_TCP_GT202_PMODA.map	MAP ファイル、アドレス情報管理
	WiFi_DHCP_TCP_GT202_PMODA.srec	モトローラーHEX ファイル
	その他	自動生成ファイル
Script	S5D9.ld	ロケーション定義ファイル
Src	tcp_thread_entry.c	TCP thread サンプルファイル
	wifi_dhcp_thread_entry.c	WiFi DHCP thread サンプルファイル
MP-S5D9-01 (リンク指定)	led.c	LED 処理モジュール
	led.h	led.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
	stcharc	文字系処理モジュール
	stcahr.h	stcharc 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-S5D9.pincfg	PIN configuration 用ファイル	
WiFi_DHCP_TCP_GT202_PMODA.jlink	Jlink デバッガー用ファイル	
その他	自動生成ファイル	

5-2. 「WiFi_DHCP_TCP_GT202_PMODB」フォルダ構成とファイル名

S5D9_e2std_NetX_WiFi_1¥WiFi_DHCP_TCP_GT202_PMODB		
Debug	WiFi_DHCP_TCP_GT202_PMODB.elf	ELF ファイル、JTAG で使用
	WiFi_DHCP_TCP_GT202_PMODB.map	MAP ファイル、アドレス情報管理
	WiFi_DHCP_TCP_GT202_PMODB.srec	モトローラーHEX ファイル
	その他	自動生成ファイル
Script	S5D9.ld	ロケーション定義ファイル
Src	tcp_thread_entry.c	TCP thread サンプルファイル
	wifi_dhcp_thread_entry.c	WiFi DHCP thread サンプルファイル
MP-S5D9-01 (リンク指定)	led.c	LED 処理モジュール
	led.h	led.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
	stcharc	文字系処理モジュール
	stcahr.h	stcharc 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-S5D9.pincfg	PIN configuration 用ファイル	
WiFi_DHCP_TCP_GT202_PMODB.jlink	Jlink デバッガー用ファイル	
その他	自動生成ファイル	

5-3. 「WiFi_DHCP_TCP_GT202_PMODC」フォルダ構成とファイル名

S5D9_e2std_WiFi_NetX_1¥ WiFi_DHCP_TCP_GT202_PMODC		
Debug	WiFi_DHCP_TCP_GT202_PMODC.elf	ELF ファイル、JTAG で使用
	WiFi_DHCP_TCP_GT202_PMODC.map	MAP ファイル、アドレス情報管理
	WiFi_DHCP_TCP_GT202_PMODC.srec	モトローラーHEX ファイル
	その他	自動生成ファイル
Script	S5D9.ld	ロケーション定義ファイル
Src	tcp_thread_entry.c	TCP Thread サンプルファイル
	wifi_dhcp_thread_entry.c	DHCP client Thread サンプルファイル
MP-S5D9-01 (リンク指定)	led.c	LED 処理モジュール
	led.h	led.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.c	文字系処理モジュール
	stcahr.h	stchar.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-S5D9.pincfg	PIN configuration 用ファイル	
WiFi_DHCP_TCP_GT202_PMODC.jlink	Jlink デバッガー用ファイル	
その他	自動生成ファイル	

5-4. サンプルの動作説明【PMODA / PNODB / PMODC 共通】

<DHCP client Thread>

- ① DHCPによるIPアドレスの取得を待つ

Term 画面

< 1 > 「<tcp_thread waiting to get IP address>」

< 2 > 「<Wait fragment_enable>」

< 3 > 「<interface status check>」

< 4 > 「<Start WiFi NetX DHCP>」

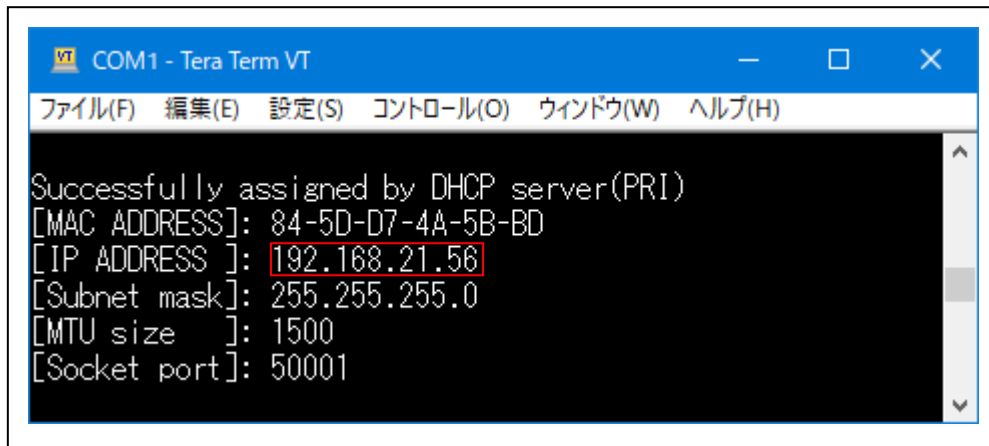
< 5 > 「<Wait Joint>」

< 6 > 「<Wait DHCP Start>」

< 7 > 「<Wait IP Resolved>」の順次処理して表示する。

・IPアドレス取得成功により、MP 基板上的 LED3 を 100msec 毎に点滅

- ② IPアドレス取得情報を Term 画面に表示する。



<TCP 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 「<Recv>」と受信データのダンプ表示をする。

- ③ ②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 「<Send>」と送信データのダンプ表示をする。

送信不可の場合、Term 画面 「<nx_tcp_socket_send() Error!!>」と表示する。

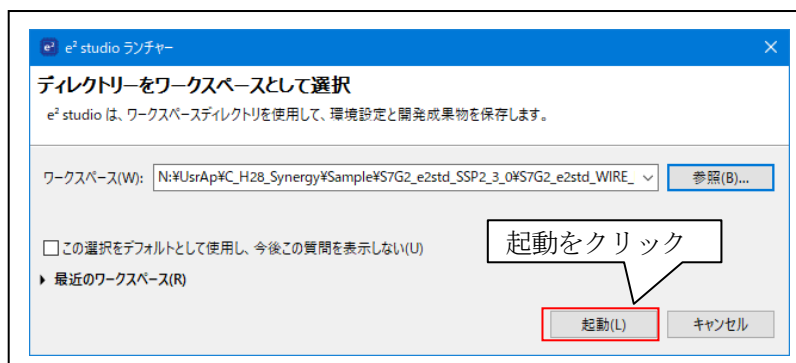
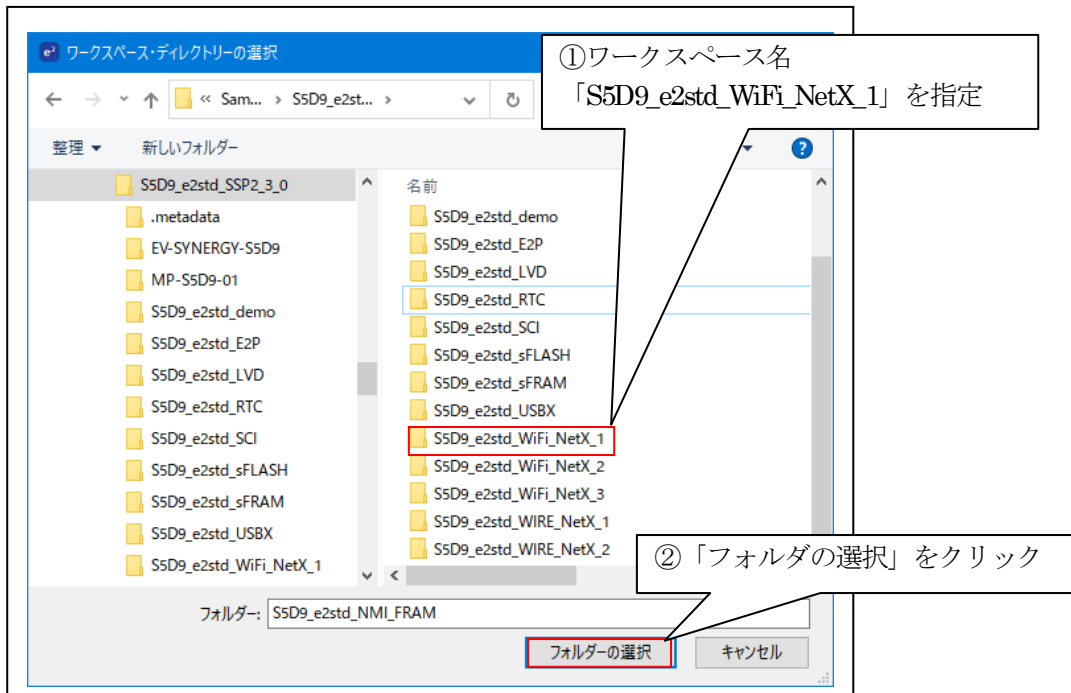
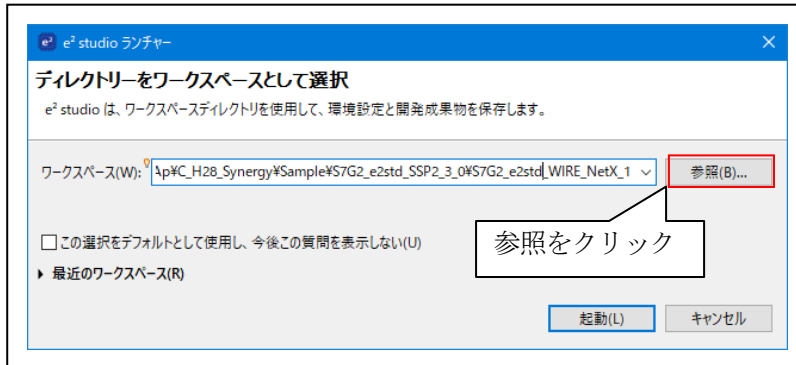
- ④ Windows PC 側でディスコネクション「切断」するまで、②からの処理を繰り返す。

- ⑤ Windows PC 側でディスコネクション「切断」した場合は、①の処理を待つ。

・TCP Thread 起動後、MP 基板上的 LED4 を 100msec 毎に点滅

6. 「S5D9_e2std_WiFi_NetX_1」をインポートする。

6-1. ワークスペース名の指定



6-2. プロジェクトのインポート

☆詳細操作は「[e2studio_synergy_Import.pdf](#)」の2項を参照して下さい。

7. デバッグ操作

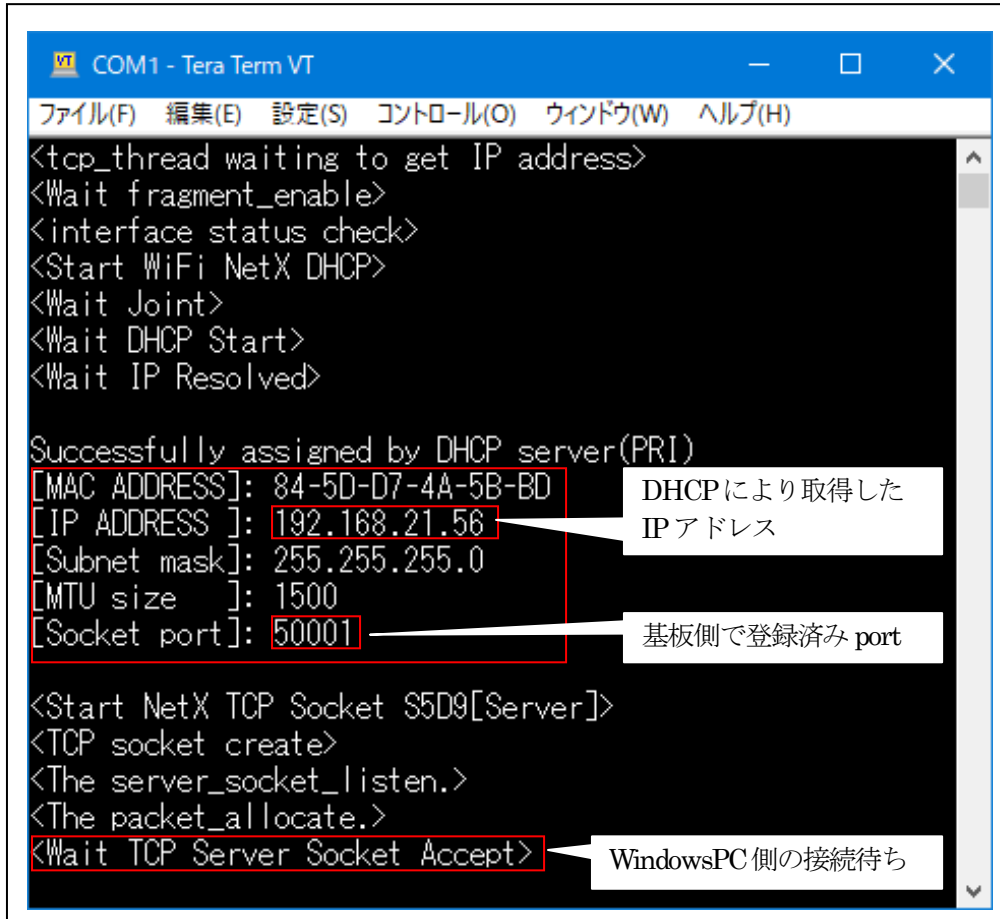
7-1. デバッグ構成の設定

☆詳細操作は「[e2studio_synergy_Import.pdf](#)」の3-1項を参照して下さい。

7-2. デバッグの開始

☆詳細操作は「[e2studio_synergy_Import.pdf](#)」の3-2項を参照して下さい。

<WiFi_DHCP_TCP_GT202_PMODx 実行画面>



```

COM1 - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
<tcp_thread waiting to get IP address>
<Wait fragment_enable>
<interface status check>
<Start WiFi NetX DHCP>
<Wait Joint>
<Wait DHCP Start>
<Wait IP Resolved>

Successfully assigned by DHCP server(PRI)
[MAC ADDRESS]: 84-5D-D7-4A-5B-BD
[IP ADDRESS ]: 192.168.21.56
[Subnet mask]: 255.255.255.0
[MTU size  ]: 1500
[Socket port]: 50001

<Start NetX TCP Socket S5D9[Server]>
<TCP socket create>
<The server_socket_listen.>
<The packet_allocate.>
<Wait TCP Server Socket Accept>
  
```

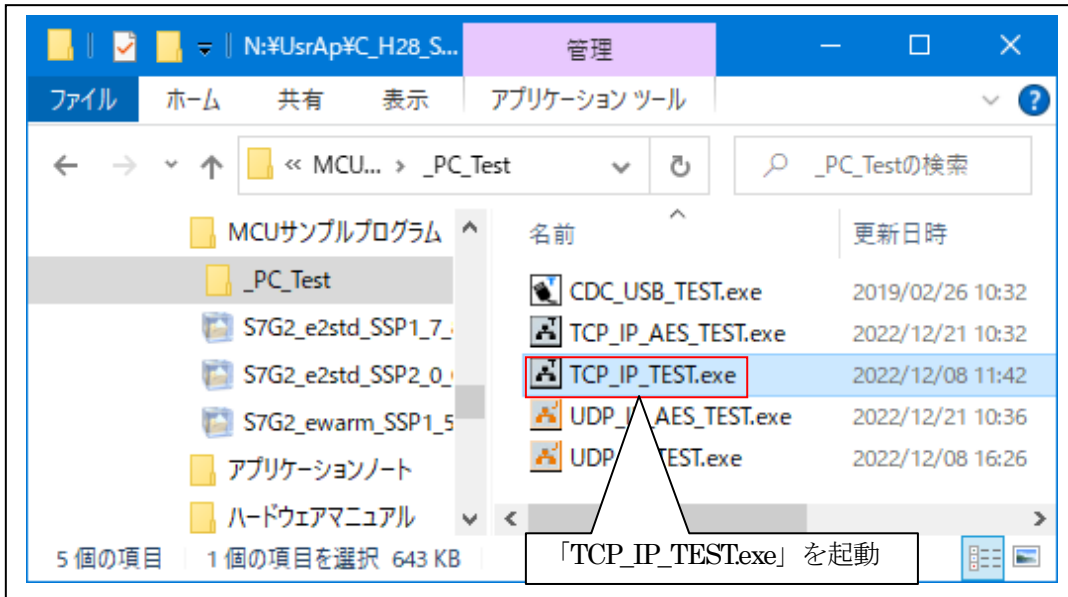
Callout boxes in the image:

- DHCPにより取得した IP アドレス (points to 192.168.21.56)
- 基板側で登録済み port (points to 50001)
- WindowsPC 側の接続待ち (points to <Wait TCP Server Socket Accept>)

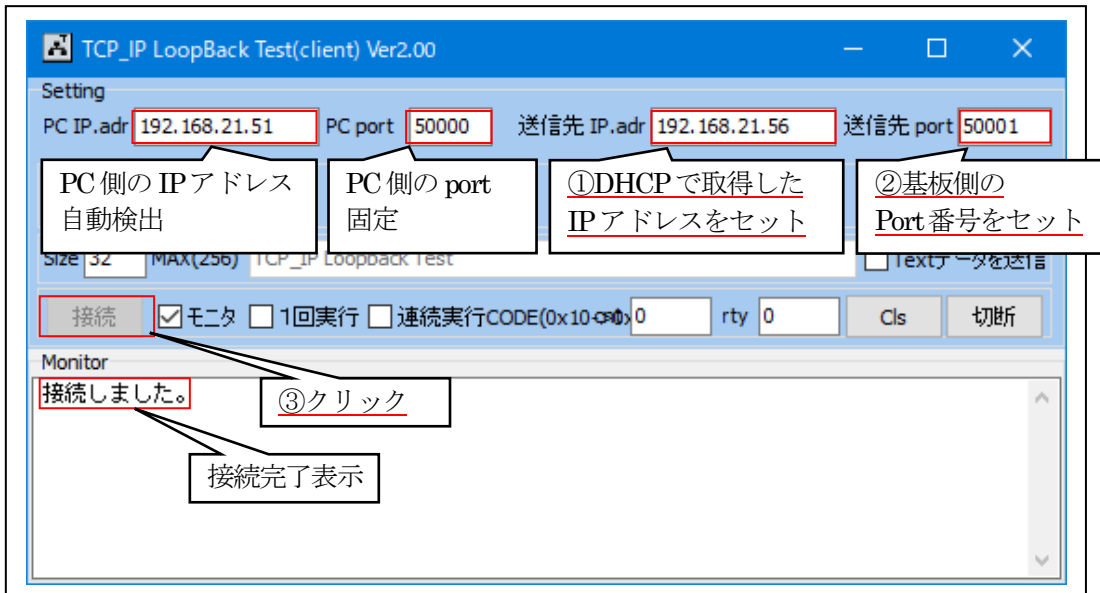
7-3. Windows PC側のテスト用プログラムを使用して動作確認

1) 「TCP_IP_TEST」を起動する。

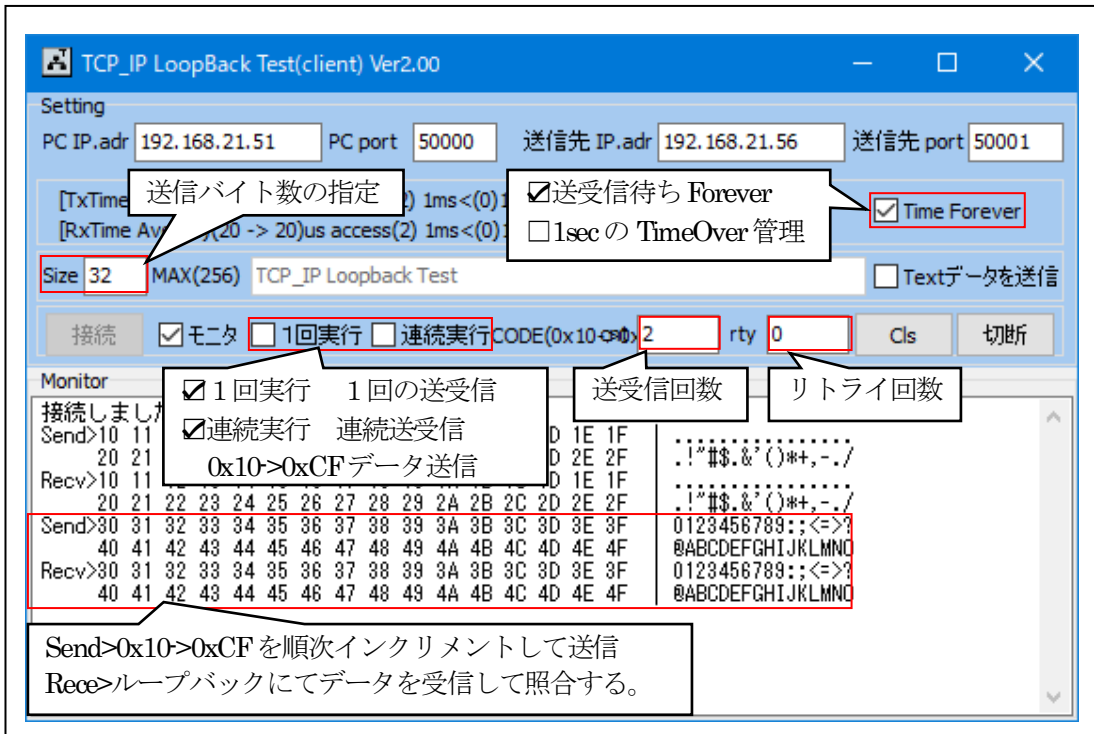
プログラム場所【ご購入 CD¥MCU サンプルプログラム¥_PC_Test】



2) TCP/IP Portを「接続」する。



3) ループバック送受信を確認する。



The screenshot shows the 'TCP_IP LoopBack Test(client) Ver2.00' application window. The 'Setting' section includes fields for PC IP (192.168.21.51), PC port (50000), destination IP (192.168.21.56), and destination port (50001). It features checkboxes for '送信待ち Forever' (checked), '1secのTimeOver管理' (unchecked), and 'Time Forever' (checked). The 'Size' is set to 32. The 'Monitor' section shows execution mode options: '1回実行 1回の送受信' (checked) and '連続実行 連続送受信' (checked). The 'CODE' field is set to 0x10 and 'rty' to 0. The monitor log displays a sequence of hexadecimal data being sent and received, with annotations explaining the data format and the loopback process.

Annotations in the image:

- 送信バイト数の指定 (Transmission byte count specification) - points to the Size field.
- 送受信待ち Forever (Waiting for transmission/reception Forever) - points to the checked checkbox.
- 1secのTimeOver管理 (1sec TimeOver management) - points to the unchecked checkbox.
- Time Forever - points to the checked checkbox.
- 1回実行 1回の送受信 (Execute once, one-time transmission/reception) - points to the checked checkbox.
- 連続実行 連続送受信 (Continuous execution, continuous transmission/reception) - points to the checked checkbox.
- 0x10>0xCFデータ送信 (0x10>0xCF data transmission) - points to the CODE field.
- 送受信回数 (Transmission/reception count) - points to the CODE field.
- リトライ回数 (Retrial count) - points to the rty field.
- Send>0x10>0xCFを順次インクリメントして送信 (Increment Send data sequentially and transmit) - points to the monitor log.
- Rece>ループバックにてデータを受信して照合する。 (Receive data via loopback and verify) - points to the monitor log.

【Error表示】

- ・送信待ち 1secTime Over 表示 「"Send Time Over Error!!"」 「Forever ノーチェックの場合」
- ・受信待ち 1secTime Over 表示 「"Receive Time Over Error!!"」 「Forever ノーチェックの場合」
- ・送信と受信データの照合エラー表示 「"Send <=>Recv 照合 Error!!"」

<Term 画面 MP-S5D9-01 側の表示>

```

COMPT-tera-term v1
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
<tcp_thread waiting to get IP address>
<Wait fragment_enable>
<interface status check>
<Start WiFi NetX DHCP>
<Wait Joint>
<Wait DHCP Start>
<Wait IP Resolved>

Successfully assigned by DHCP server(PRI)
[MAC ADDRESS]: 84-5D-D7-4A-5B-BD
[IP ADDRESS ]: 192.168.21.56
[Subnet mask]: 255.255.255.0
[MTU size   ]: 1500
[Socket port]: 50001

<Start NetX TCP Socket S5D9[Server]>
<TCP socket create>
<The server_socket_listen.>
<The packet_allocate.>
<Wait TCP Server Socket Accept>
<The connection established it.>
Recv>10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F | .....
      20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F | #$.&'()*+,-./
Send>10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F | .....
      20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F | .!"#$%&'()*+,-./
Recv>30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F | 0123456789:;<=>?
      40 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F | @ABCDEFGHIJKLMNO
Send>30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F | 0123456789:;<=>?
      40 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F | @ABCDEFGHIJKLMNO
<Wait TCP Server Socket Accept>

```

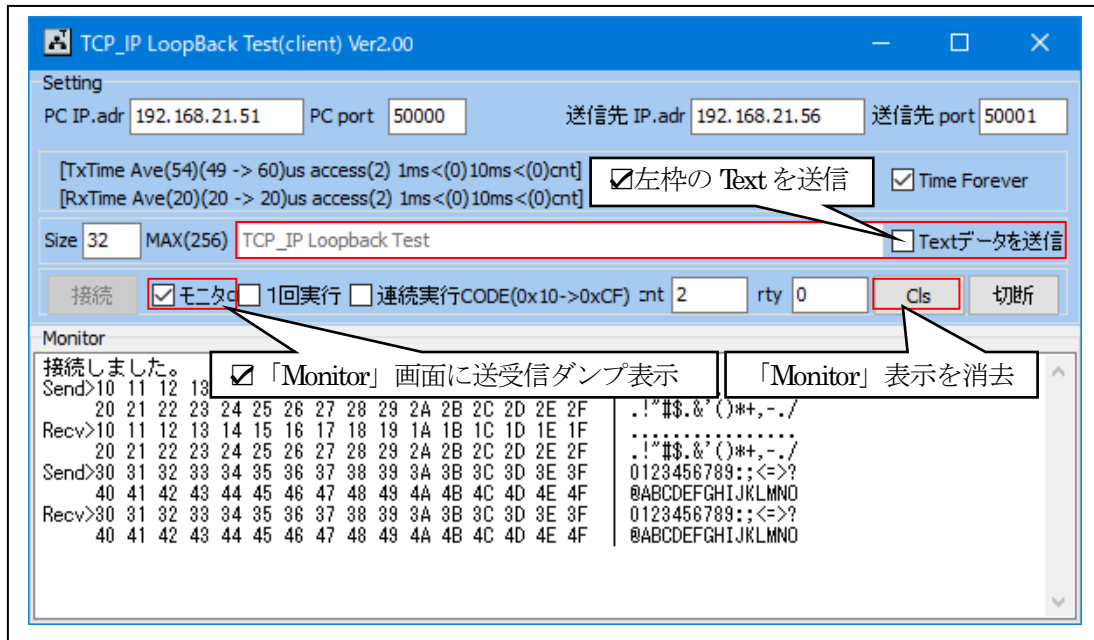
接続完了

受信データ

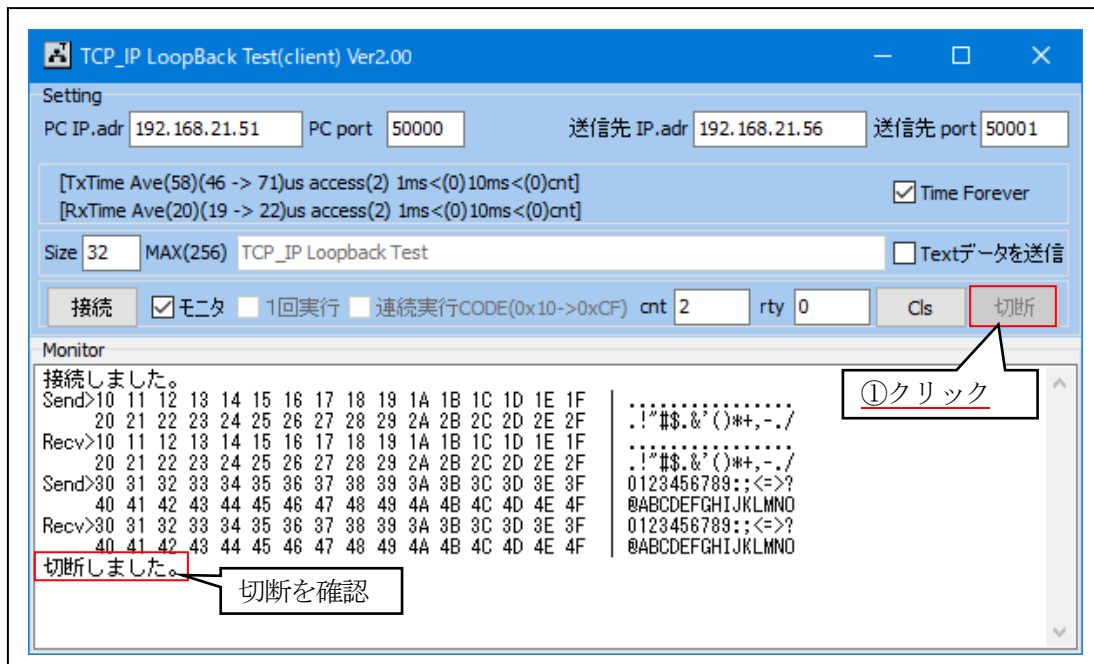
送信データ

接続待ち

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「[e2studio_synergy_Import.pdf](#)」の3-3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を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

- e2studio 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
- 「S5D9 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
- 「e2studio ユーザーズマニュアル 入門ガイド」 ルネサス エレクトロニクス株式会社
- 「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>

