

Renesas S7G2 用サンプル(e2studio Demo)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
S7G2_e2std_demo	MP-S7G2-01 基板に実装した全デバイスの総合サンプルデモソフト	S7G2_demo_STD Azure RTOS で動作 SCI Driver(r_sci_uart) SCI(Interrupt Priority2) RTC Driver(g_rtc0) RTC(Interrupt Priority3) E2P Driver(g_riic) RIIC(Interrupt Priority2) FRAM Driver(r_sci_spi) SPI(Interrupt Priority2) sFROM Driver(r_sci_spi) SPI(Interrupt Priority2) TCP 通信 (nx_tcp_socket.....) USBX Device (g_fx_media0 FileX) (g_ux_device_class_cdc_acm) NMI Interrupt Azure RTOS 使用例

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。
- ②PCにインストールし実行する
- ③シリアルポートの設定

Tera Term: シリアルポート 設定と接続

ポート(P): COM1

スピード(E): 115200

データ(D): 8 bit

パリティ(A): none

ストップビット(S): 1 bit

フロー制御(F): none

送信遅延
0 ミリ秒/字(C) 0 ミリ秒/行(L)

Device Friendly Name: 通信ポート (COM1)
Device Instance ID: ACPI\PNP0501\0
Device Manufacturer: (標準ポート)
Provider Name: Microsoft
Driver Date: 6-21-2006
Driver Version: 10.0.19041.1

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

Tera Term: 端末の設定

端末サイズ(T): 90 x 54

☒ ウィンドウサイズ(S):
☐ 自動的に調整(W):

端末ID(I): VT100

応答(A):

改行コード
受信(R): CR
送信(M): CR+LF

ローカルエコー(L): ☐
自動切り替え(VT<->TEK)(U): ☐

漢字-受信(K): SJIS
漢字-送信(J): SJIS

半角カナ(F): ☐ 半角カナ(D): ☐

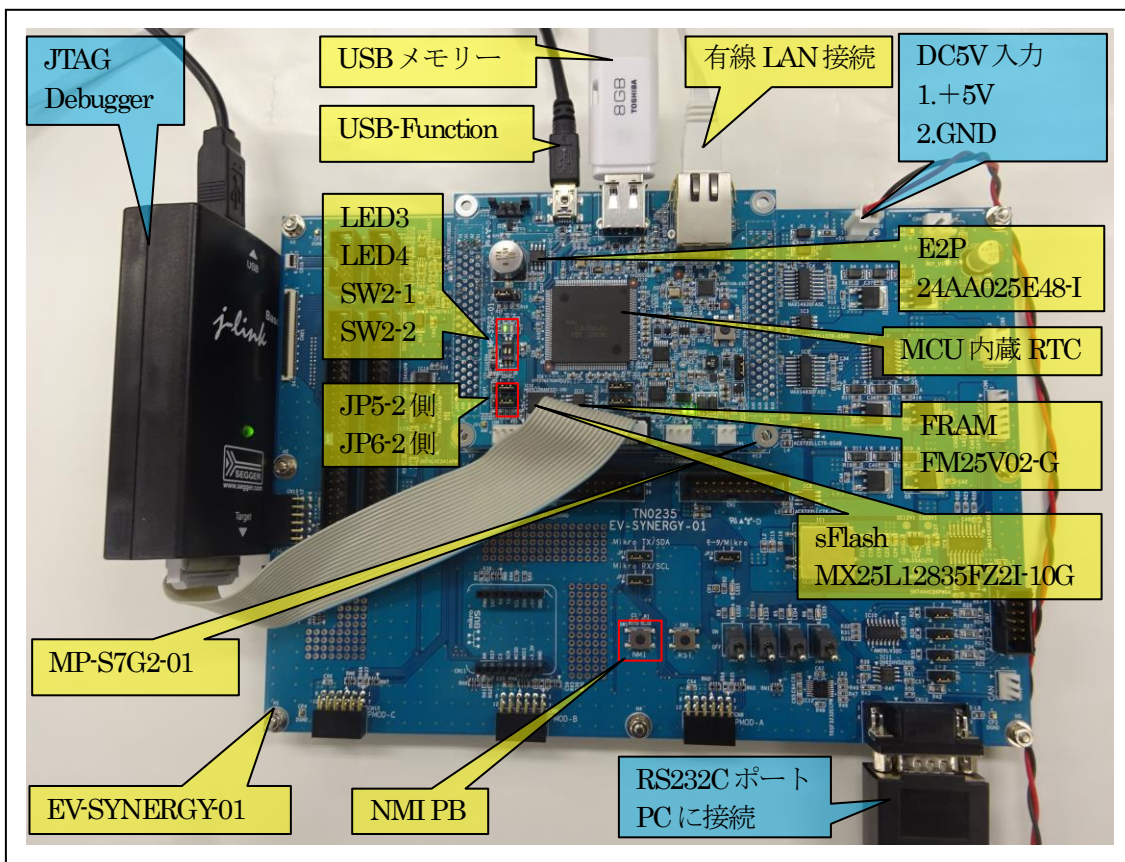
ローカル(C): japanese

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。

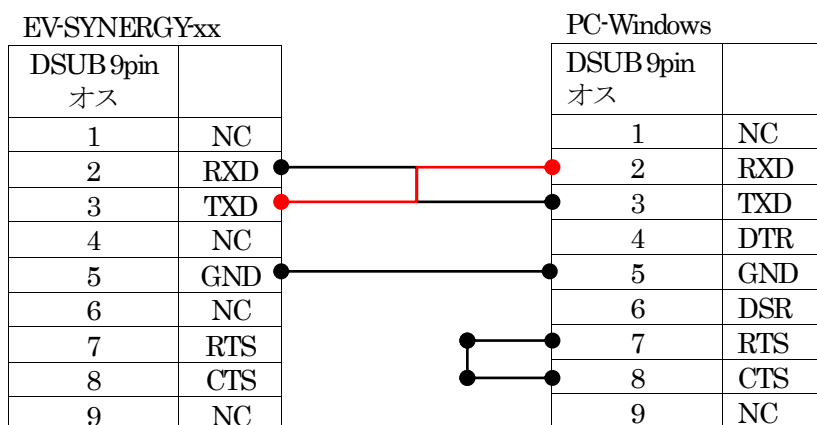
赤枠の設定にする。

4. 動作構成

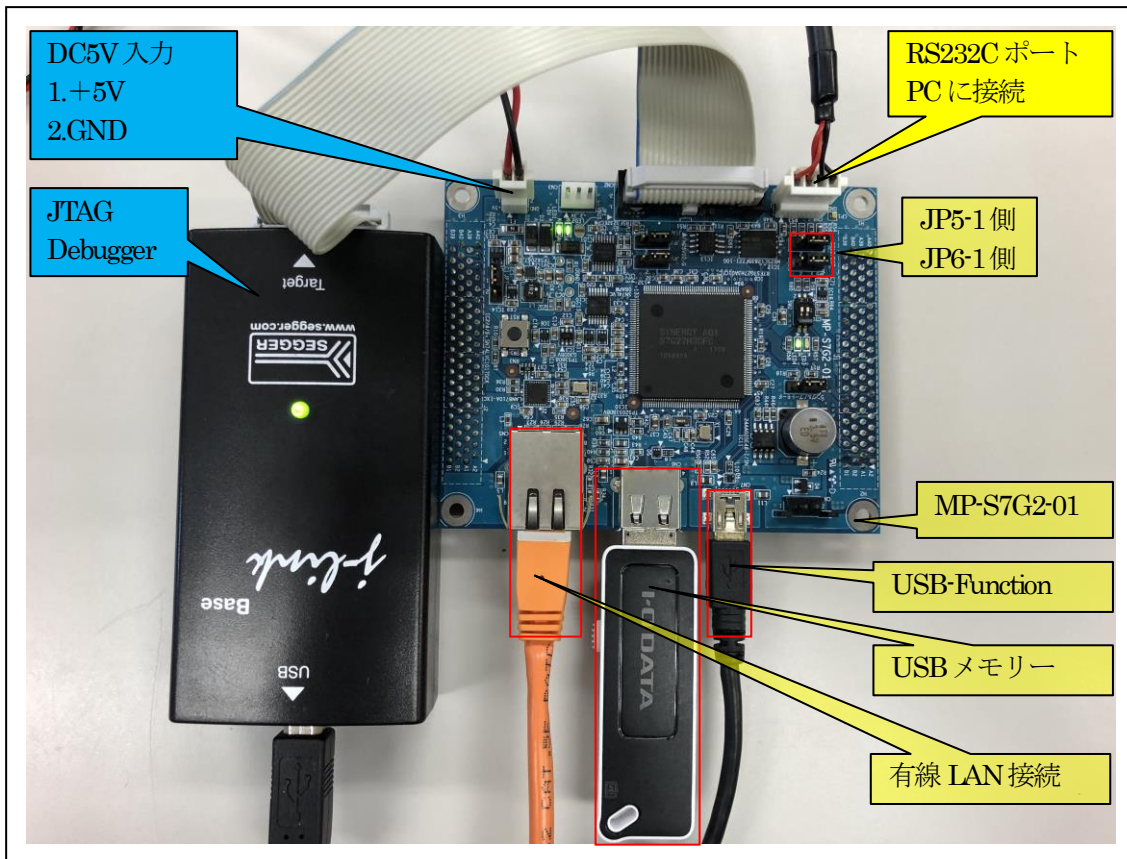
【EV-SYNERGY01】を使用の場合



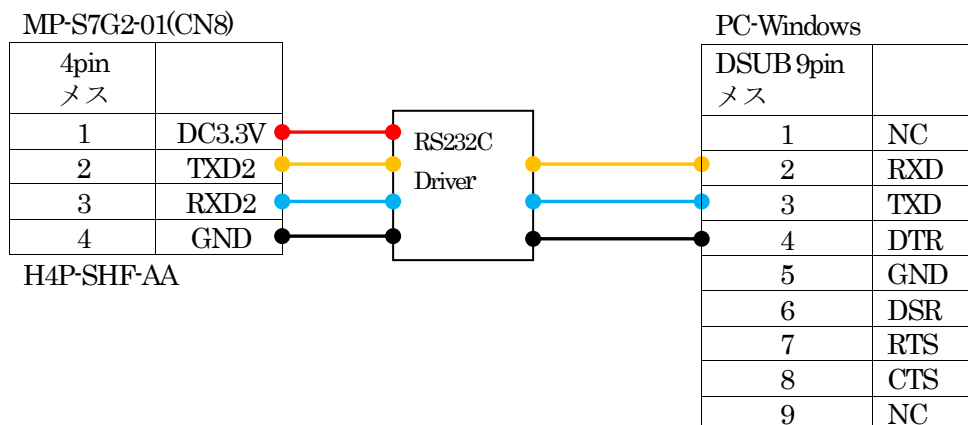
- ①PC機と接続する RS232C ケーブルは、市販「クロスケーブル」でも可能です。
- ②USB-シリアル変換ケーブルを使用される場合は、「StarTech.com 社 ICUSB232FIN」推奨
- ③自作する場合は、下記の配線になります。



【MP-S7G2-01】のみ使用の場合



- ①PC機と接続する RS232C ケーブルは、製作が必要です。
 ②「RS232C-Driver」は、下記 URL の「RS232CAB4」を推奨します。
http://tool-kobo.ddo.jp/Files/Product/RS232_422/RS232CAB.htm



5. 「S7G2_demo_STD」 サンプルの説明

5-1. 「S7G2_demo_STD」 フォルダ構成とファイル名

S7G2_e2std_demo¥S7G2_demo_STD		
Debug	S7G2_demo_STD.elf	ELF ファイル、JTAG で使用
	S7G2_demo_STD.map	MAP ファイル、アドレス情報管理
	S7G2_demo_STD.srec	モトローラーHEX ファイル
	その他	自動生成ファイル
Script	r7fs7g27h3a01cfc.ld	ロケーション定義ファイル
Src	main_thread_entry.c	main thread サンプルソース
	main.h	#include を纏めたヘッダーファイル
	MP_SwLed_thread_entry.c	MP 基板の SW/LED サンプルソース
	nmi_thread_entry.c	NMI 制御サンプルソース
	Socket_fix_thread_entry.c	Socket server Thread サンプルソース
	usb_thread_entry.c	usb thread サンプルソース(Host)
	usbf_thread_entry.c	usbf thread サンプルソース(Function)
<div>app</div> <div>MP-S7G2-01 (リンク指定)</div> <div>synergy_gen</div>	command.c	Command shell サンプルソース
	command.h	command.c 用ヘッダーファイル
	help.c	Command Help サンプルソース
	help.h	help.c 用ヘッダーファイル
	dipsw.c	DIP-SW 処理モジュール
	dipsw.h	dipsw.c 用ヘッダーファイル
	e2p.c	E2PROM 処理モジュール
	e2p.h	e2p.c 用ヘッダーファイル
	led.c	LED 処理モジュール
	led.h	led.c 用ヘッダーファイル
	rtc.c	RTC 処理モジュール
	rtc.h	rtc.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
	sFlash.c	sFlash ROM 処理モジュール
	sFlash.h	sFlash.c 用ヘッダーファイル
	sFram.c	sFRAM 処理モジュール
	sFram.h	sfram.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-MP-S7G2-01.pincfg	PIN configuration 用ファイル
	S7G2_demo_STD.jlink	Jlink デバッガー用ファイル
	その他	自動生成ファイル

6. 「S7G2_e2std_demo」をインポートする。

6-1. ワークスペース名の指定

e² studio ランチャー

ディレクトリーをワークスペースとして選択

e² studio は、ワークスペースディレクトリーを使用して、環境設定と開発成果物を保存します。

ワークスペース(W): 参照(B)...

☐ この選択をデフォルトとして使用し、今後この質問を表示しない(U)

▶ 最近のワークスペース(R)

起動(L) キャンセル

参照をクリック

ワークスペース・ディレクトリーの選択

① ワークスペース名「S7G2_e2std_demo」を指定

② 「フォルダの選択」をクリック

フォルダ: S7G2_e2std_demo

フォルダの選択 キャンセル

e² studio ランチャー

ディレクトリーをワークスペースとして選択

e² studio は、ワークスペースディレクトリーを使用して、環境設定と開発成果物を保存します。

ワークスペース(W): 参照(B)...

☐ この選択をデフォルトとして使用し、今後この質問を表示しない(U)

▶ 最近のワークスペース(R)

起動(L) キャンセル

起動をクリック

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

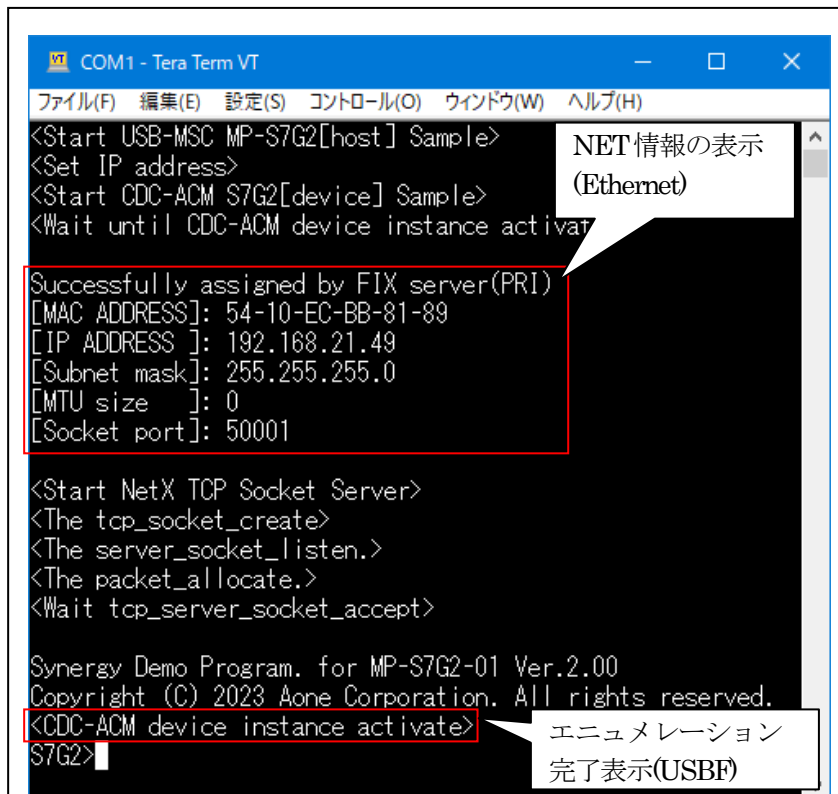
☆詳細操作は「[e2studio_synergy_Import.pdf](#)」の3-1項を参照して下さい。

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。

7-3. 「S7G2_e2std_demo」の動作説明

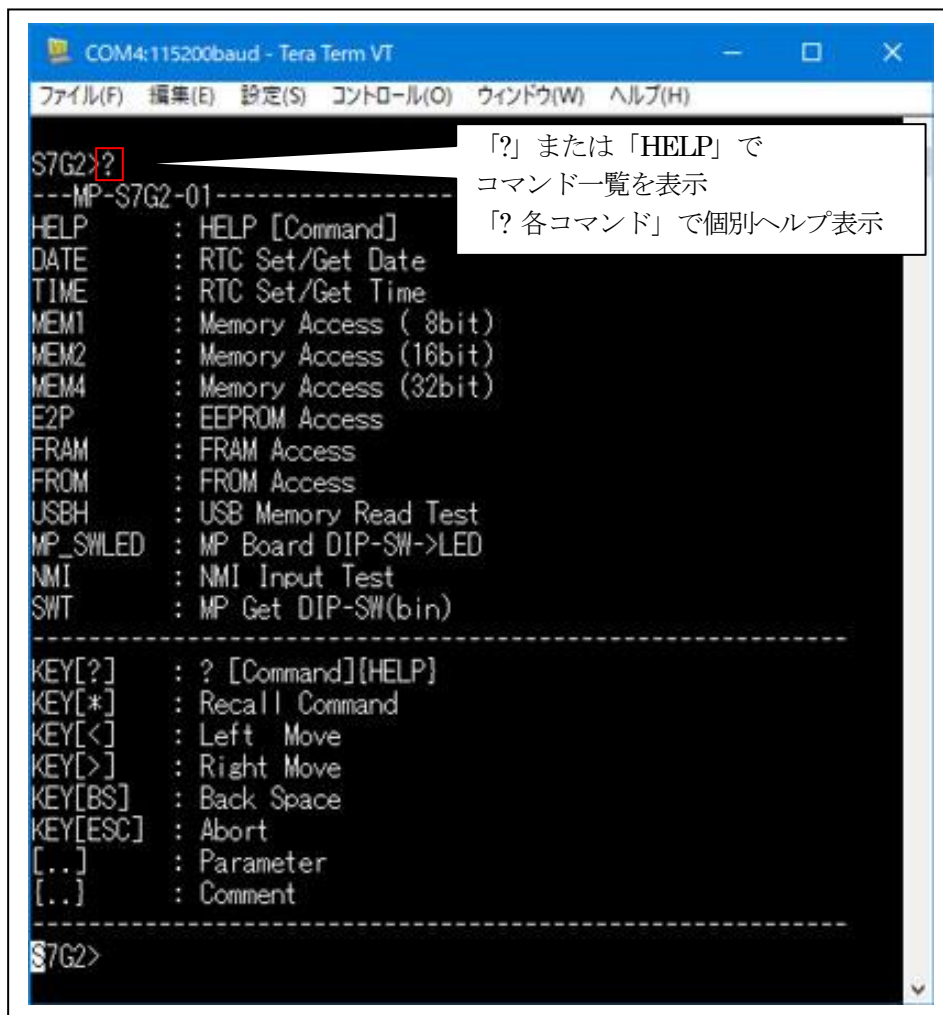
7-3-1. 「S7G2_demo__STD」プロジェクトを実行する。



TeraTerm 画面にオープニングメッセージが表示される

<補足>

- ・[USB-CDC-ACM]の WindowsPC 機へのドライバーインストール方法
アプリケーションノート「[S7G2_e2std_USBX_CDC_ACM.pdf](#)」7-3項を参照して下さい。



TeraTerm 画面に Help が表示される

7-3-2. 各コマンドの説明

DATE コマンド

MCU 内蔵の RTC に年月日曜を設定する。

```
DATE 年 月 日 曜日 ↵ //DATE 2015 4 5 0 2015/4/5 日曜日
                        //曜日 0:日 1:月 2:火 3:水 4:木 5:金 6:土
DATE ↵                //現設定データを表示
```

TIME コマンド

MCU 内蔵の RTC に時間を設定する。

```
TIME 時 分 秒 ↵      //TIME 9 0 0 9時 0分 0秒
TIME ↵               //現設定データを表示
```

MEM1 コマンド

メモリーを 8bit アクセスで Read/Write/FILL/インクリメント FILL する。

MEM2 コマンド

メモリーを 16nit アクセスで Read/Write/FILL/インクリメント FILL する。

MEM4 コマンド

メモリーを 32bit アクセスで Read/Write/FILL/インクリメント FILL する。

```
MEM{1/2/4} 先頭アドレス サイズ {パターン} ↵
```

{READ}

```
MEM1 R 0x2002_8000 0x100 ↵ //0x2002_8000 から 0x100 要素分 8bit ダンプ
MEM2 R 0x2002_8000 0x100 ↵ //0x2002_8000 から 0x100 要素分 16bit ダンプ
MEM4 R 0x2002_8000 0x100 ↵ //0x2002_8000 から 0x100 要素分 32bit ダンプ
```

{FILL}

```
MEM1 F 0x2002_8000 0x100 0 ↵ //0x2002_8000 から 0x100 要素分(0)8bitFILL
MEM2 F 0x2002_8000 0x100 0 ↵ //0x2002_8000 から 0x100 要素分(0)16bitFILL
MEM4 F 0x2002_8000 0x100 0 ↵ //0x2002_8000 から 0x100 要素分(0)32bitFILL
```

{Increment FILL} 指定初期データからアクセスごとに + 1 しながら FILL 動作

```
MEM1 I 0x2002_8000 0x100 0 ↵ //0x2002_8000 から 0x100 要素分 8bitFILL
MEM2 I 0x2002_8000 0x100 0 ↵ //0x2002_8000 から 0x100 要素分 16bitFILL
MEM4 I 0x2002_8000 0x100 0 ↵ //0x2002_8000 から 0x100 要素分 32bitFILL
```

{WRITE}

```
MEM1 W 0x2002_8000 0x12 ↵ //0x2002_8000 に 0x12 を Write
MEM2 W 0x2002_8000 0x1234 ↵ //0x2002_8000 に 0x1234 を Write
MEM4 W 0x2002_8000 012345678 ↵ //0x2002_8000 に 0x12345678 を Write
```

E2P コマンド

EEPROM の Read/Write 処理をします。

E2P_ {R/W}_ EEPROM アドレス_ {メモリアドレス}_ サイズ

{READ}

E2P_ R_ EEPROM アドレス_ サイズ

ex)

E2P_ 0x0_ 0x100 // EEPROM の 0x0 番地から 0x100 サイズ分ダンプ表示

{WRITE}

E2P_ W_ EEPROM アドレス_ メモリアドレス_ サイズ

ex)

E2P_ W_ 0x0_ 0x2000_0000_ 0x80 // EEPROM の 0x0 番地に 0x2000_0000 番地の内
// 容を 0x80 サイズ分 Write

この EEPROM は、MAC アドレス内蔵の EEPROM です。

EEPROM の (0x80~0xFF) は、ライトプロテクトになっていますので Write できません。

MAC アドレスは、【0xFA~0xFF】の 8 バイトに格納してあります。

{Read Only Memory アドレス}

- EEPROM エリア {0x80 ~ 0xFF}
- 内蔵 RAM エリア {0x1FFE_0000 ~ 0x2007_FFFF}

{Write Memory アドレス}

- EEPROM エリア {0x0 ~ 0x7F}
- 内蔵 RAM エリア {0x2007_0000 ~ 0x2007_FFFF}

E2P コマンドには LAN 用データ設定コマンドが用意されています。

① E2P_ MAC {Read Only}

MAC の表示

② E2P_ PORT {Read/Write}

PORT 番号の表示と設定

{Read}

E2P_ PORT

{Write}

E2P_ PORT_ {xxxxx} // E2P_ PORT_ 50000

③ E2P_ IP {Read/Write}

デフォルト IP アドレスの表示と設定

{Read}

E2P_ IP

{Write}

E2P_ IP_ {xx.xx.xx.xx} // E2P_ IP_ 192.168.21.12

④ E2P_SUB {Read/Write}

サブネットマスクの表示と設定

{Read}

E2P_SUB↵

{Write}

E2P_SUB_{xx.xx.xx.xx}↵ // E2P_SUB_255.255.255.0 ↵

FRAM コマンド

FRAM の内容を内蔵メモリーに Read します。また、内蔵 RAM の内容を FRAM に Write する。

{READ}

FRAM_R_FRAM アドレス Store アドレス サイズ

ex)

FRAM_R_0x0_0x2007_0000_0x8000

FRAM アドレス(0x0)からサイズ(0x8000)分 Store アドレス(0x2007_0000)に Read する。

{WRITE}

FRAM_W_FRAM アドレス Memory アドレス サイズ

ex)

FRAM_W_0x0_0x1FFE_0000_0x8000

FRAM アドレス(0x0)に Memory アドレス(0x1FEE_0000)からサイズ(0x8000)分 Write する。

- FRAM アドレス {0x0 ~ 0x7FFF}
- Store アドレス {0x2007_0000 ~ 0x2007_FFFF}
- Memory アドレス {0x1FFE_0000 ~ 0x2007_FFFF}

FROM コマンド

FROM の内容を内蔵メモリーに Read します。また、内蔵 RAM の内容を FROM に Write する。

{READ}

FROM_R_FROM アドレス Store アドレス サイズ

ex)

FROM_R_0x0_0x2007_0000_0x8000

FROM アドレス(0x0)からサイズ(0x8000)分 Store アドレス(0x2007_0000)に Read する。

{WRITE}

FROM_W_FROM アドレス Memory アドレス サイズ

ex)

FROM_W_0x0_0x1FFE_0000_0x8000

FROM アドレス(0x0)に Memory アドレス(0x1FFE_0000)からサイズ(0x8000)分 Write する。

- FROM アドレス {0x0 ~ 0xFF_FFFF}
- Store アドレス {0x2007_0000 ~ 0x2007_FFFF}
- Memory アドレス {0x1FFE_0000 ~ 0x2007_FFFF}

MP_SWLED コマンド

MP 基板上の DIP-SW2 と LED3/LED4 の動作確認をする。

SW2-1[ON] --->LED3[点灯] : SW2-1 [OFF] --->LED3[消灯]
SW2-2[ON] --->LED4[点灯] : SW2-2 [OFF] --->LED4[消灯]

NMI コマンド

NMI 割り込みの有効・無効を指定する。

NMI_1 ↵ // NMI 割り込み有効
NMI_0 ↵ // NMI 割り込み無効

【EV-SYNERGY01】を未使用の場合

- ・MP-SG2-01 基板の J1-A4 を Low にすると NMI 割り込みが発生します。

SWT コマンド

MP 基板上の DIP-SW の状態を数値で知らせる。

SWT ↵ // DIP-SW の状態を調べる。

Answer)

#SWT[0] // SW2ALLOFF
#SWT[1] // SW2-1 ON
#SWT[2] // SW2-2 ON
#SWT[3] // SW2ALL ON

USBH コマンド

USBメモリーの登録ファイルを Read して画面表示します。

USBH R {Filename.ext}↵

ex)

USBH R Sample0.mot↵

7-4. WindowsPC機との USB 通信確認方法

☆アプリケーションノート「S7G2_e2std_USBX_CDC_ACM.pdf」7項を参照して下さい。

7-5. WindowsPC機との TCP 通信確認方法

☆アプリケーションノート「S7G2_e2std_WIRE_NetX_2_TCP.pdf」の7項を参照して下さい。

7-6. 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

- ・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>

