

USB 対応ツール QE for USB

R20AN0413JJ0110

Rev.1.10

2025.06.10

使用ガイド

はじめに

ルネサス ソリューション・ツールキット→アプリケーション特化ツール QE (Quick and Effective Tool Solution) 製品のひとつ、QE for USB V2.0.0 [テクニカルプレビュー版]を使用することで、USB システムのデバッグが簡単に行え、開発期間の短縮およびコスト低減を実現できます。

本書では、このツールの適用方法について、実例をあげながら図解して説明します。各々の機能の詳細な説明は QE for USB ヘルプも参照してください。

動作確認デバイス

RX ファミリ : RX111、RX231、RX65N、RX651、RX64M、RX71M

RL78 ファミリ : RL78/G1C、RL78/L1C

RA ファミリ : RA2L2 (#)

※下記目次の 7 章と 8 章は(#)が付いたデバイスのみ対応です。

目次

1. システム構成.....	3
2. QE for USB のインストール.....	4
3. サンプル・プロジェクトをインポート.....	6
4. QE for USB で USB 接続をチェック.....	8
4.1 USB ステート・チャート (QE) ビューで USB ステートを表示.....	8
5. QE for USB で USB の設定レジスタをチェック.....	11
5.1 設定レジスタを表示.....	11
5.2 設定レジスタをデバッグ.....	12
6. QE for USB で USB ディスクリプタの値をチェック.....	14
6.1 ディスクリプタの値を表示.....	14
6.2 ディスクリプタをデバッグ.....	15
7. QE for USB で USB Type-C の情報をチェック.....	17
7.1 USB Type-C 情報確認 (QE) ビューで供給電力を表示.....	17
7.2 USB Type-C 情報確認 (QE) ビューで利用している信号線を表示.....	18
8. QE for USB で USB VBUS モニタの情報をチェック.....	20
8.1 USB VBUS モニタ (QE) ビューで電圧と電流を表示.....	20
9. QE for USB V2.0.0 のサポート USB ファームウェア一覧.....	21

改訂記録22

1. システム構成

QE for USB 使用時のシステム構成は、以下の通りです。

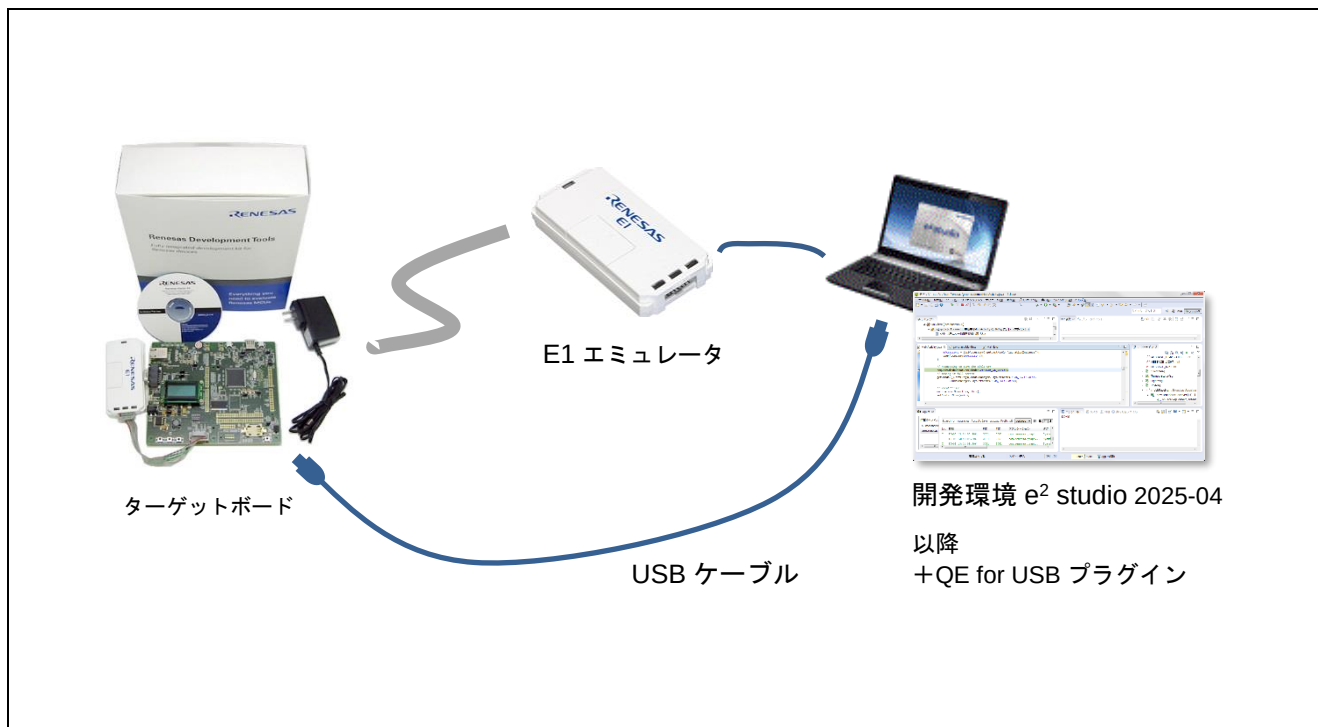


図 1-1 システムの構成

本書では、ターゲットボード EK-RA2L2 および USB ファームウェアを、システムの一例として使用しています。

対応環境

- ・ ホスト OS
Windows 10, 11(日・英)
- ・ エミュレータ
E2 エミュレータ、E2 エミュレータ Lite
- ・ 開発環境
e² studio 2025-04 以降
- ・ ターゲットボード
動作確認デバイス(マイコン)の各種 RSK、HMI ソリューションキット
および動作確認デバイスを搭載した任意のターゲットボード

※対応する e² studio、エミュレータ、ターゲットボードは予めご用意下さい

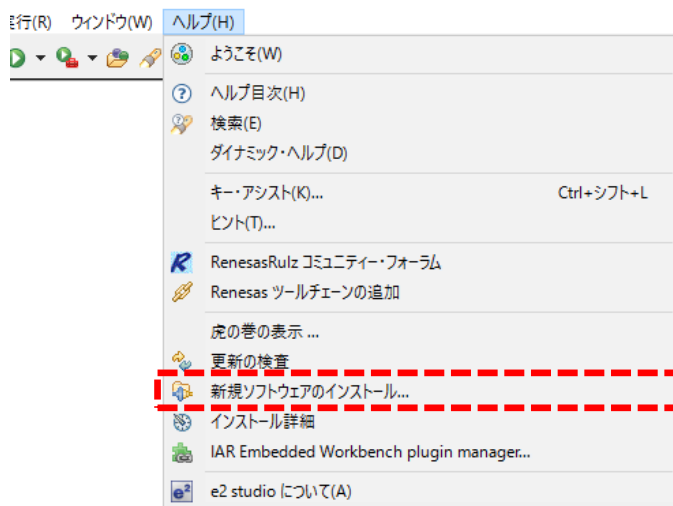
2. QE for USB のインストール

QE for USB の入手先 URL : <https://www.renesas.com/qe-usb>

または、e² studio のスマート・ブラウザ上で本書を選択し、右クリックメニュー[サンプル・コード(ダウンロード)]よりダウンロードできます。

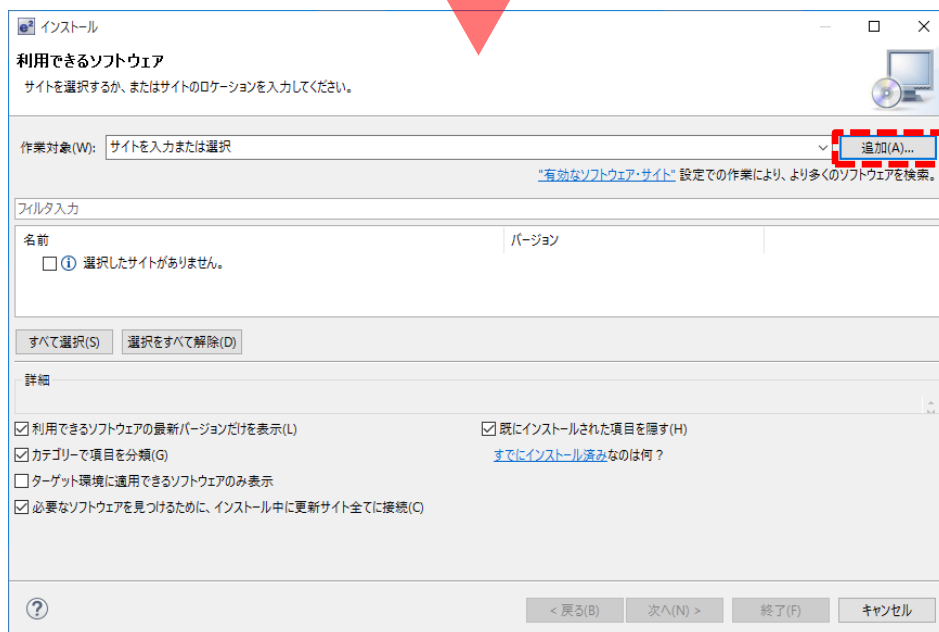
以下の手順で QE for USB をインストールします。

Step①



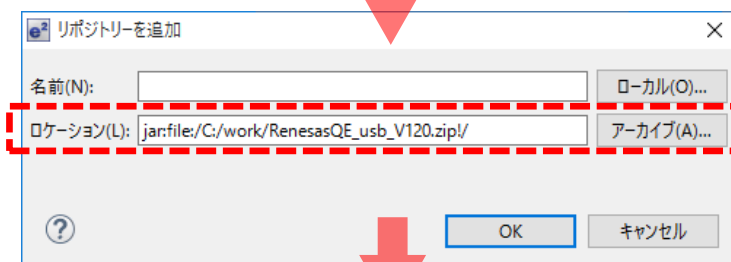
e² studio [ヘルプ]メニュー
[新規ソフトウェアのインストール...]

Step②



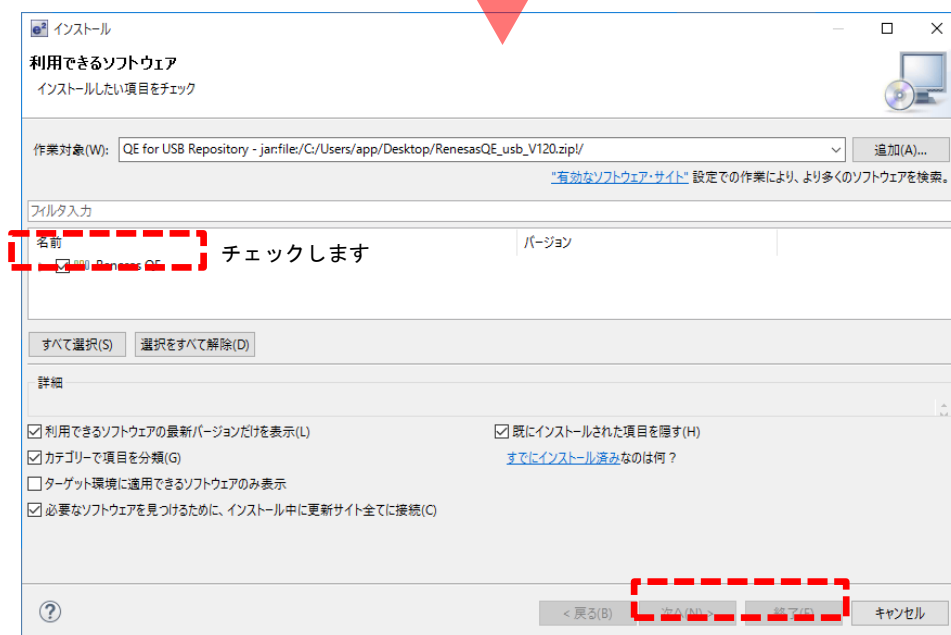
[追加]ボタン

Step③



ダウンロードした
QE for USB の zip を指定

Step④



[次へ]ボタンを押します。
この後、セキュリティ警告、証明書の選択、
e² studio の再起動と進め、インストールを完了
させます。

図 2-1 インストール方法 (概略)

<インストール方法 (詳細)>

1. e² studio を起動する。
2. [ヘルプ]→[新規ソフトウェアのインストール...]メニューを選択し、[インストール]ダイアログを開く。
3. [追加(A)...]ボタンを押下し、[リポジトリを追加]ダイアログを開く。
4. [アーカイブ(A)...]ボタンを押下し、開いたファイル選択ダイアログで、インストール用ファイル(zip ファイル)を選択し、[開く(O)]ボタンを押下。
5. [リポジトリを追加]ダイアログで、[OK]ボタンを押下。
6. [インストール]ダイアログに、表示された[Renesas QE for USB] および[Renesas QE common]チェックボックスをチェックし、[次へ(N)>]ボタンを押下。
7. インストール対象が [Renesas QE for USB] および[Renesas QE common]となっていることを確認し、[次へ(N)>]ボタンを押下。
8. ライセンスを確認した後、[使用条件の条項に同意します(A)]ラジオ・ボタンを選択し、[終了(F)]ボタンを押下。
9. セキュリティ警告が表示されるが[OK]ボタンを押下してインストールを継続する。
10. 信頼する証明書の選択ダイアログが表示された場合、表示された証明書をチェックした後、[OK]ボタンを押下してインストールを継続する。
11. e² studio の再起動を促されるので再起動を行う。

3. サンプル・プロジェクトをインポート

RX 用サンプル・プロジェクトの入手先 URL :

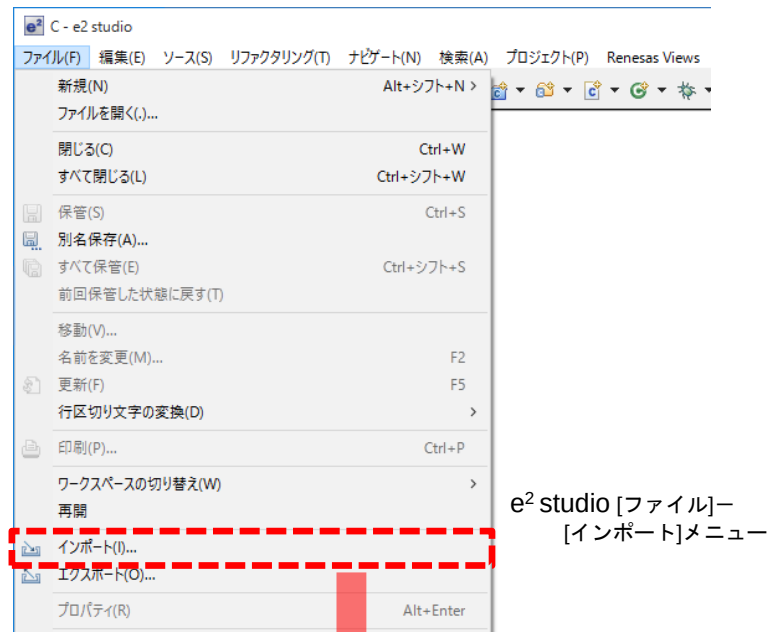
<https://www.renesas.com/software-tool/usb-drivers>

RA2L2 用サンプル・プロジェクトの入手先 URL :

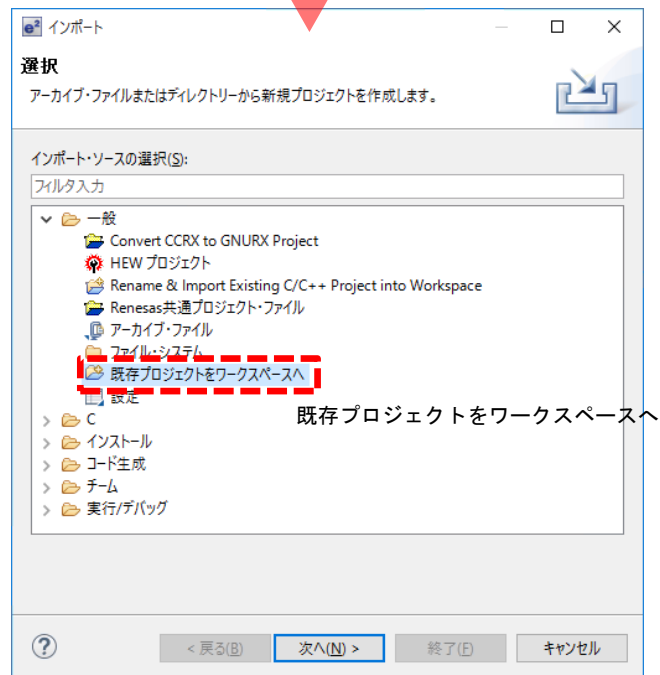
<https://github.com/renesas/ra-fsp-examples>

プロジェクトをダウンロードした場合、以下の手順で e² studio にインポートします。

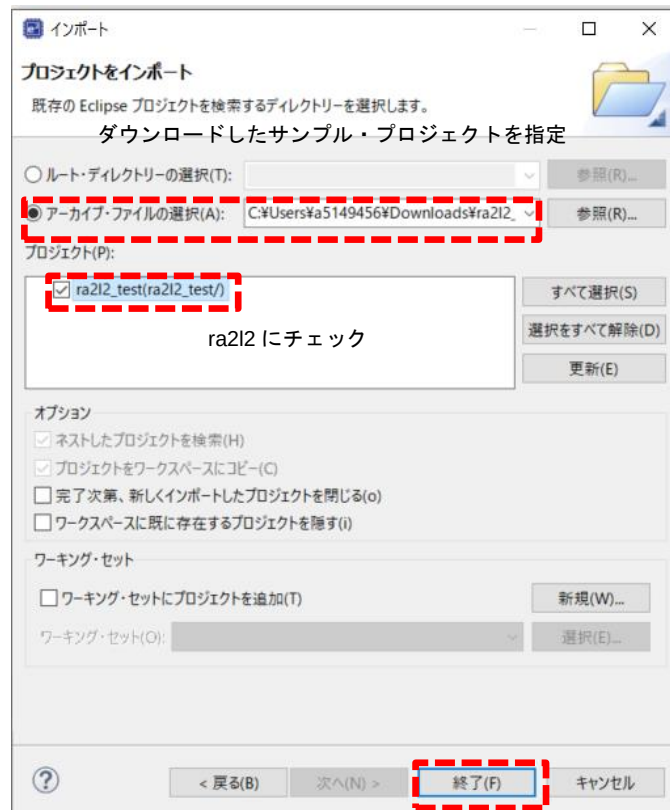
Step①



Step②



Step③



[終了]をクリックし、プロジェクトにインポート完了

図 3-1 サンプル・プロジェクトのインポート方法

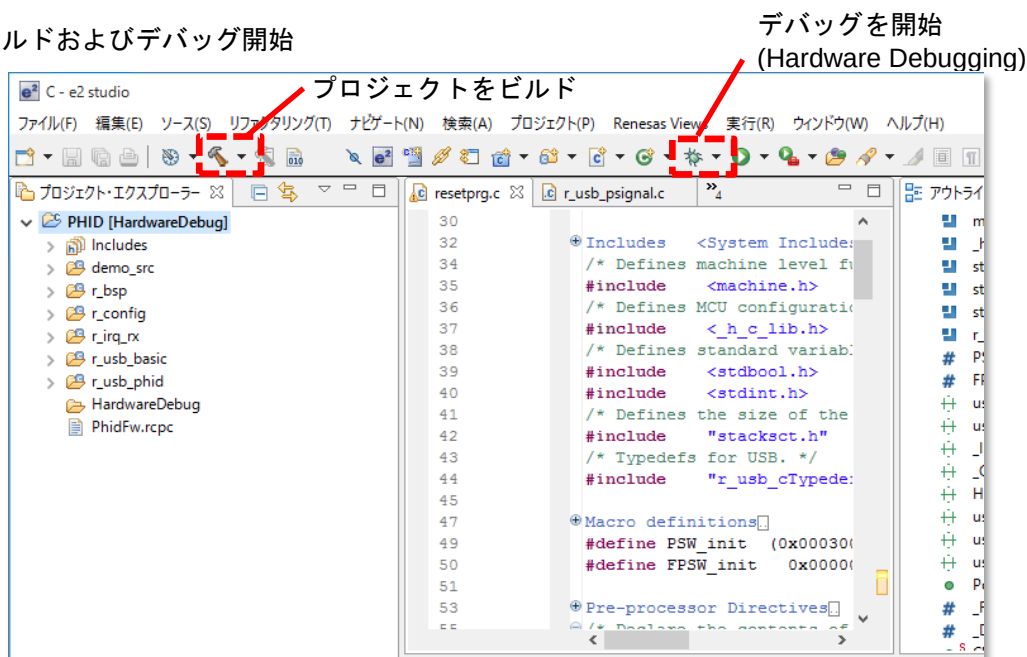
4. QE for USB で USB 接続をチェック

サンプル・プロジェクトをビルド・実行し、QE for USB ツールを用いて、USB 接続のステータスを確認します。

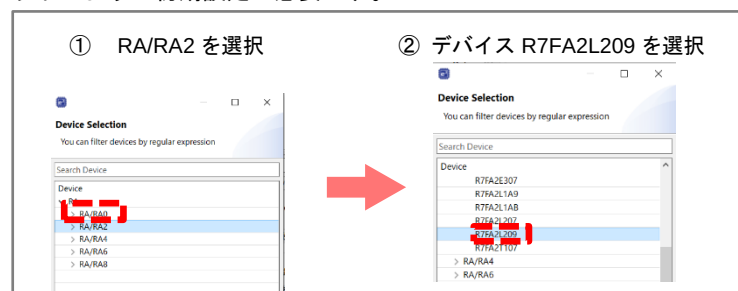
ターゲットボードと USB ホスト(PC)を接続する、USB ケーブルを用意してください。

4.1 USB ステータス・チャート (QE) ビューで USB ステータスを表示

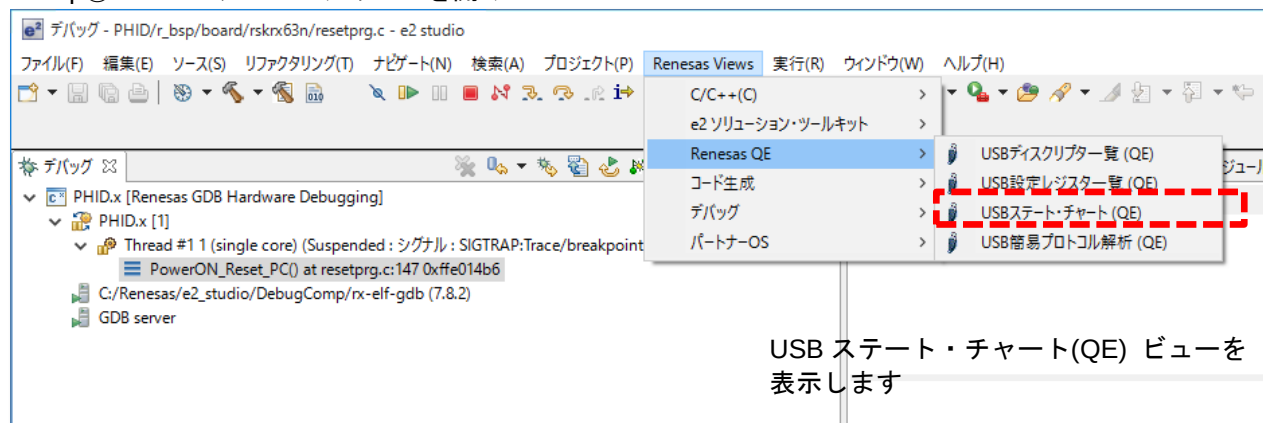
Step① ビルドおよびデバッグ開始



初めてデバッグを開始する場合、表示されるダイアログに沿って以下のように初期設定が必要です。



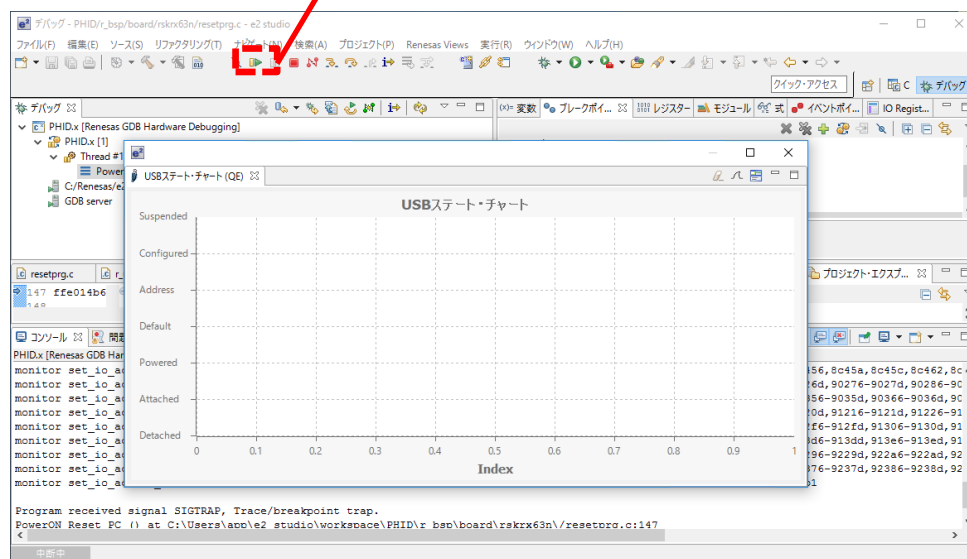
Step② USB ステータス・チャートを開く



USB ステータス・チャート(QE) ビューを表示します

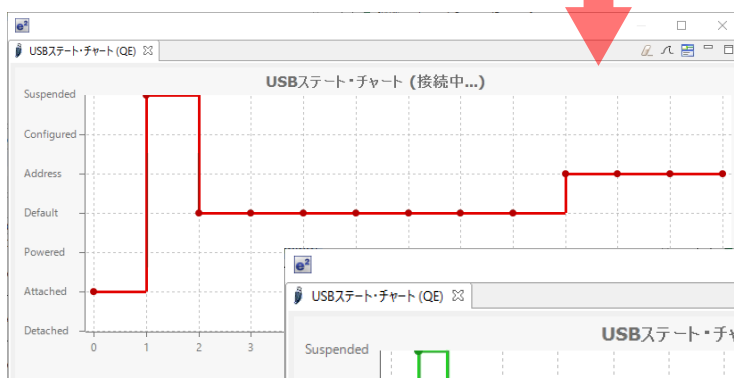
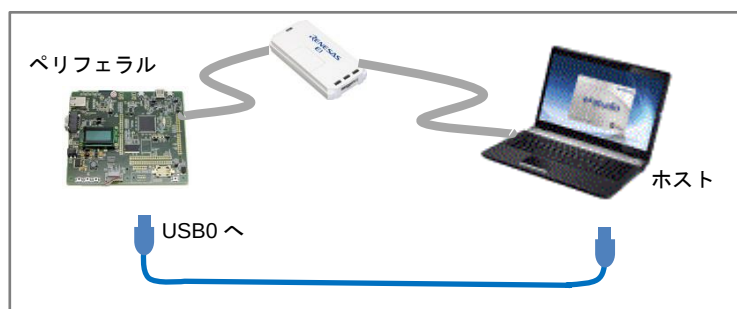
Step③ 実行

実行



この状態で、システムを実行させ、
ターゲットボードと、PC(ホスト)を USB ケーブルで
接続します。

Step④ USB ケーブル接続



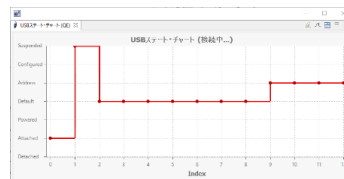
USB 接続(エニユメレーション)処理
が、どう進んでいるかをウォッチで
できます。左図では、[Address]ス
テートであり、「接続中」。



接続が成功すると、[Configured]ス
テートになり、「接続成功」となり
ます。

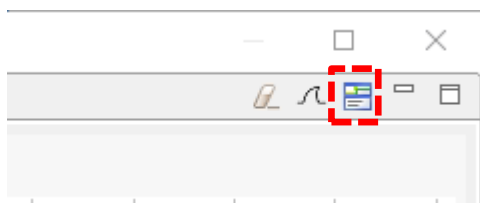
注意 1 :

USB ステートが、[configured]ステートにならない時は、USB のドライバがインストールされていない場合があります。ご使用のシステムにあった USB ドライバをインストールしてください。

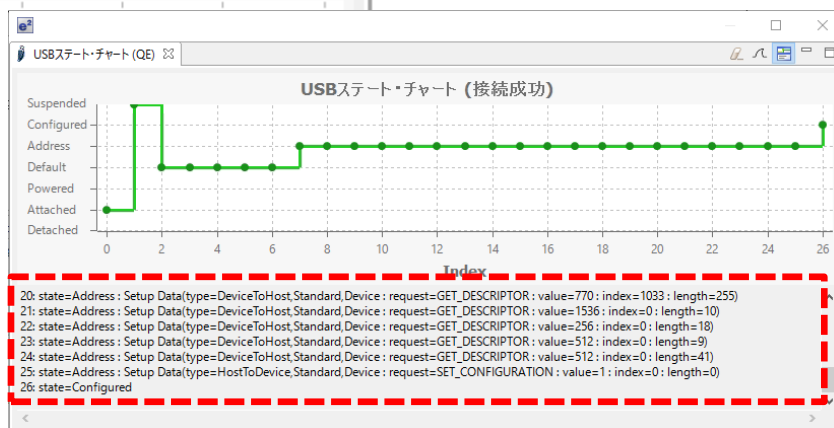
**注意 2 :**

「このビューが使用する関数が見つかりません。このビューのヘルプを参照してください。」というメッセージが表示されチャートが描画されない場合は、コンパイラオプションの最適化によってチャートの描画に必要な関数が見つからなくなっている可能性があります。

USB ステート・チャートビューのヘルプ、[トラブル シューティング]を参照してください。

セットアップデータの確認方法 :

ビュー右上の、[セットアップデータを表示する]をクリックすることで、チャートの各プロットに対応したセットアップデータを確認できます。



チャートの横軸は、ステートの遷移のインデックスを示しています。時間ではありません。(最大 50 まで表示) セットアップデータ行頭のインデックスと対応しています。

セットアップデータ :

USB 接続処理時に、ホスト PC からターゲットデバイス(ペリフェラル)へ、情報取得や設定を行うために送られるデータです。USB 接続処理中に問題がある場合は、このデータを確認する必要があります。

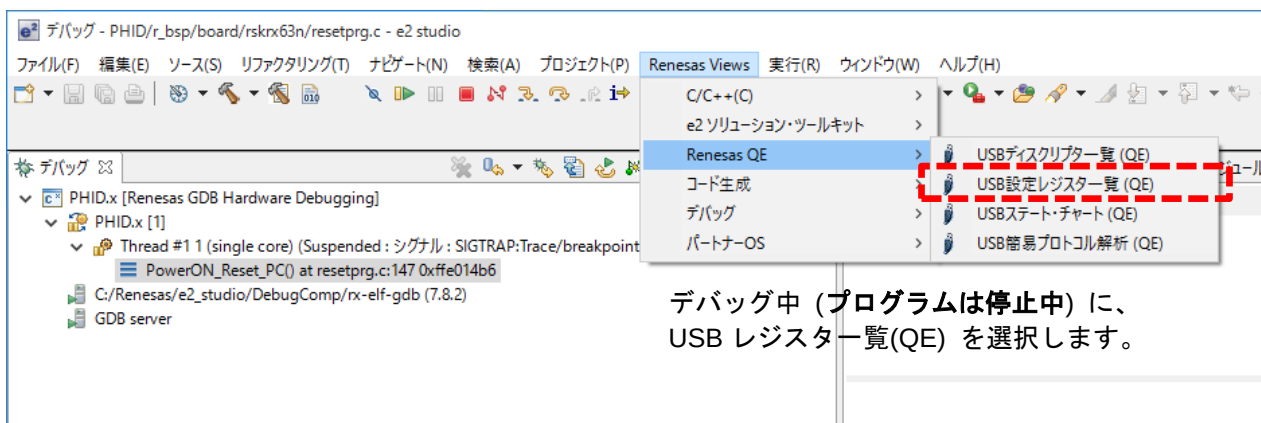
図 4-1 USB 接続のチェック方法

5. QE for USB で USB の設定レジスタをチェック

QE for USB で USB コントローラのレジスタ設定をチェックします。このビューでは、USB コントローラの使用に必要な設定レジスタの値や意味のチェックが可能です。また、設定値に問題がある場合には、NG マークを表示します。

5.1 設定レジスタを表示

Step① レジスタ一覧ビューを表示します。

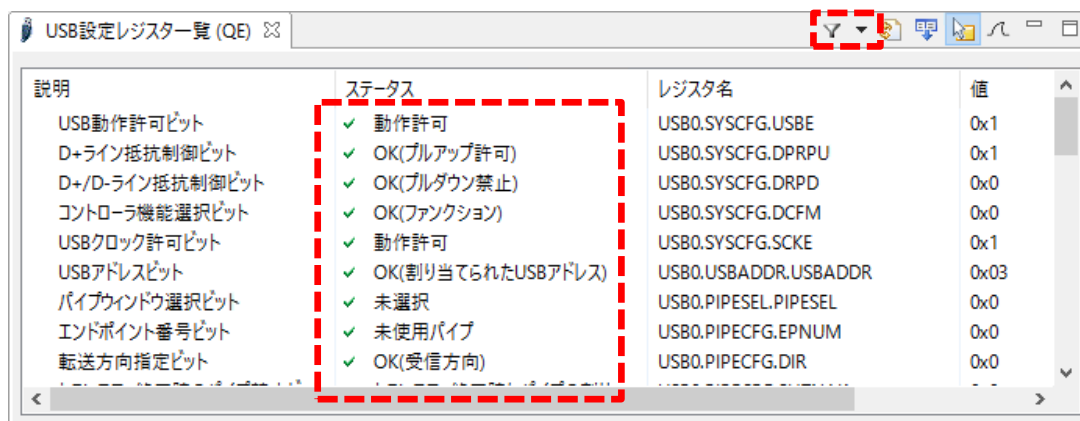


Step②

設定レジスタの値を確認します

Memo:

USB が 2 チャンネルあるデバイスの場合、フィルタ機能が便利です。



緑のチェック・マークは、OK 値であることを表しています。

NG 値は、下記の表示になります。

✗ NG

図 5-1 レジスタのチェック方法

5.2 設定レジスタをデバッグ

レジスタの設定不備で USB の接続がうまくいかない場合、レジスタ一覧ビューをチェックすることにより、問題解決できる可能性があります。[転送タイプ・ビット]レジスタで NG があった場合を例に紹介します。

Step① デバッグしプログラム停止中に、「レジスタ一覧ビュー」を開きます。

説明	ステータス	レジスタ名	値
パイプウィンドウ選択ビット	✓ OK(パイプ1)	USB0.PIPESEL.PIPESEL	0x1
エンドポイント番号ビット	✓ 未使用パイプ	USB0.PIPECFG.EPNUM	0x0
転送方向指定ビット	✓ OK(受信方向)	USB0.PIPECFG.DIR	0x0
転送終了時のパイプ禁止ビット	✓ トランスファ終了時にパイプの割り...	USB0.PIPECFG.SHTNAK	0x0
転送タイプビット	✗ NG(設定できない値です)	USB0.PIPECFG.TYPE	0x2

設定が間違っています。
QE for USB を使用して、問題を解決できます。

Step② ポップアップヘルプにより、レジスタの意味と正しい値を確認します。

説明	ステータス	レジスタ名	値
パイプウィンドウ選択ビット	✓ OK(パイプ1)	USB0.PIPESEL.PIPESEL	0x1
エンドポイント番号ビット	✓ 未使用パイプ	USB0.PIPECFG.EPNUM	0x0
転送方向指定ビット	✓ OK(受信方向)	USB0.PIPECFG.DIR	0x0
転送終了時のパイプ禁止ビット	✓ トランスファ終了時にパイプの割り...	USB0.PIPECFG.SHTNAK	0x0
転送タイプビット	✗ NG(設定できない値です)	USB0.PIPECFG.TYPE	0x2

ビット	シンボル	ビット名	機能	R/W
b15-b14	TYPE[1:0]	転送タイプビット (注1)	・パイプ1、2の場合 b15 b14 00: パイプを不使用 01: バルク転送 10: 設定しないでください 11: アイスクロナス転送 ・パイプ3～5の場合 b15 b14 00: パイプを不使用 01: バルク転送 10: 設定しないでください 11: 設定しないでください ・パイプ6～9の場合 b15 b14 00: パイプ不使用 01: 設定しないでください 10: インターラプト転送 11: 設定しないでください	R/W

ポップアップヘルプを確認します。
その結果、[転送タイプ・ビット]は、パイプ 1 に対して「10=0x2」は設定不可であり、その他の値に設定する必要があることがわかります。

Step③ 問題のレジスタへブレークポイントを設定します。

説明	ステータス	レジスタ名	値
パイプウィンドウ選択ビット	✓ OK(パイプ1)	USB0.PIPESEL.PIPESEL	0x1
エンドポイント番号ビット	✓ 未使用パイプ	USB0.PIPECFG.EPNUM	0x0
転送方向指定ビット	✓ OK(受信方向)	USB0.PIPECFG.DIR	0x0
転送終了時のパイプ禁止ビット	✓ トランスファ終了時にパイプの割り...	USB0.PIPECFG.SHTNAK	0x0
転送タイプビット	✗ NG(設定できない値です)	USB0.PIPECFG.TYPE	0x2

まず、問題のレジスタの行で右クリックし、[書き込みブレーク]メニューを選択します。これにより本レジスタへの書き込みブレークが設定され、設定したレジスタへの書き込みアクセスがあった時に、プログラムがブレークします。

Step③ ソース・コードを特定し、値を修正します。



CPU リセット後、[実行]を繰り返します。ブレークした箇所のコードをチェックし、問題の値を書き込んでいるソース・コードを探します。

```
in.c | [C] r_usb_pstare... | [C] r_usb_csched... | [C] r_usb_creg... 25 | [C] r_usb...  
usb_creg_write_pipecfg(ptr, 0x0000 + 0) 37  
usb_creg_write_pipecfg(ptr, 0x0000 + 4) 37  
  
usb_creg_data  
  
/* Write PIPECFG.TYPE */  
usb_creg_write_pipecfg(ptr, data);|  
  
/* FIFO buffer DMA-FID initialization */  
usb_creg_write_pipecfg(ptr, USB_FID0);
```

問題となる値を書き込んでいるソース・コードを特定できたら、設定不可値“0x02”以外の値へ、修正を検討ください。

なお、設定した書き込みブレークは、e²studio の[ウインドウ]–[ビューの表示]–[ブレークポイント]メニューから開く[ブレークポイント]ビューから削除できます。

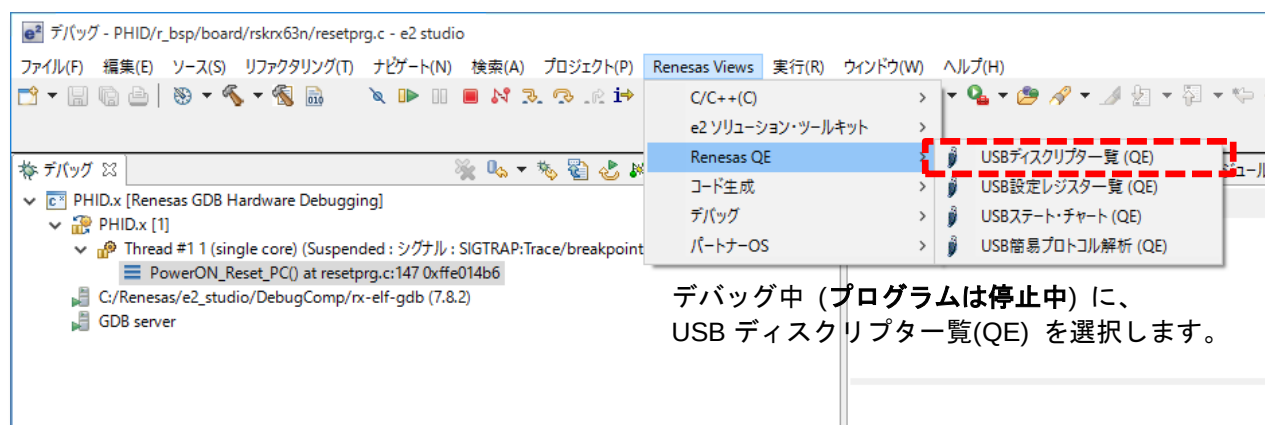
図 5-2 レジスタのデバッグ方法

6. QE for USB で USB ディスクリプタの値をチェック

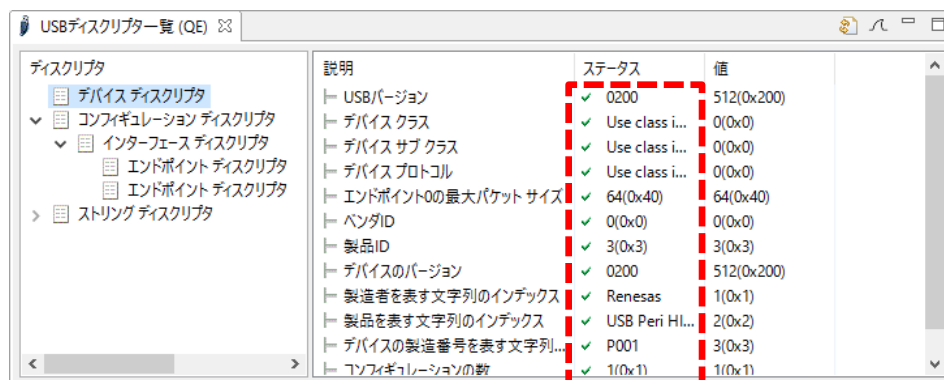
ここでは、QE for USB で USB のディスクリプタ設定をチェックします。このビューでは、USB 機能の動作に必要なディスクリプタ値やその意味のチェック、NG 値を知ることができます。

6.1 ディスクリプタの値を表示

Step① ディスクリプター一覧ビューを表示します。



Step② ディスクリプタをチェックします。



緑のチェック・マークは、OK 値であることを表しています。

NG 値は、下記の表示になります。

✖ NG

注意：

「このビューが使用する変数が見つかりません。このビューのヘルプを参照してください。」というメッセージが表示されデータが表示されない場合は、コンパイラオプションの最適化によって必要な変数が見つからなくなっている可能性があります。

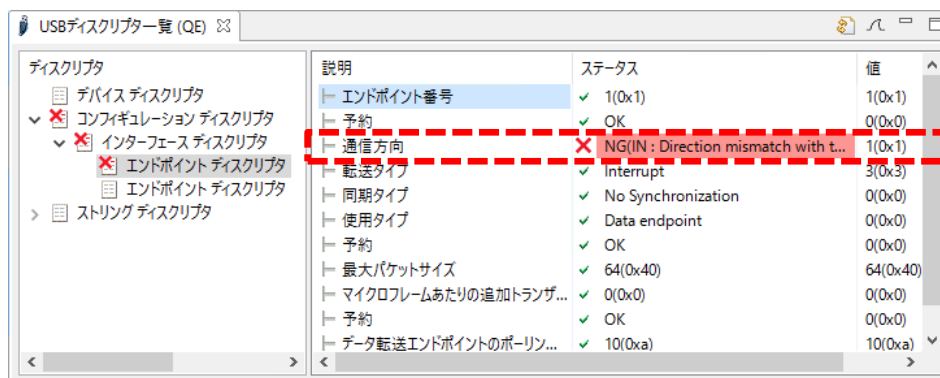
USB ディスクリプター一覧ビューのヘルプ、[トラブル シューティング]を参照してください。

図 6-1 ディスクリプタのチェック方法

6.2 ディスクリプタをデバッグ

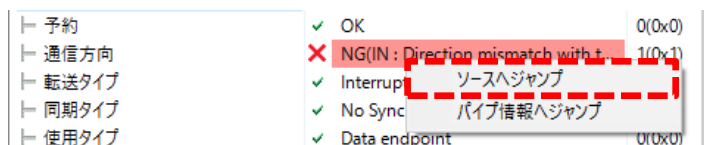
USB の接続や、接続後の通信がうまくいかない場合、ディスクリプタの設定が誤っている可能性があります。本機能により、誤りのチェック、および問題の修正ができる可能性があります。エンドポイント・ディスクリプタの[通信方向]で NG があった場合を例に紹介します。

Step① ディスクリプター一覧ビューを表示します。



設定が間違っており、デバッグを開始します。
この場合、NG メッセージに「IN: Direction mismatch with the pipe information table」とあり、設定値である"IN"がパイプ情報テーブルの値と食い違っていることが考えられます。

Step② 問題のディスクリプタを設定しているソースをチェックします。



右クリック[ソースヘジャンプ]を選択

```

USB_SMPLEPRLEN, /* 7: wItemLength */
0x00, /* 8: wItemLength */
/* Endpoint Descriptor 0 */
7, /* 0: bLength */
USB_DT_ENDPOINT, /* 1: bDescriptorType */
(uint8_t)(USB_EP_IN | USB_EP1), /* 2: bEndpointAddress */
USB_EP_INT, /* 3: bmAttributes */
0, /* 4: wMaxPacketSize */
0, /* 5: wMaxPacketSize */
0x0A, /* 6: bInterval */
/* Endpoint Descriptor 1 */
7, /* 0: bLength */
USB_DT_ENDPOINT, /* 1: bDescriptorType */
(uint8_t)(USB_EP_OUT | USB_EP2), /* 2: bEndpointAddress */
USB_EP_INT, /* 3: bmAttributes */
0, /* 4: wMaxPacketSize */
0, /* 5: wMaxPacketSize */
0x0A, /* 6: bInterval */

```

該当のディスクリプタを設定しているソース箇所が自動で選択されます。

設定したい値は"IN"方向であり、ディスクリプタの値は"USB_EP_IN"となり、正しいことがわかります。

Step③ パイプ情報のテーブルを設定しているソースをチェックします。

予約	OK	0(0x0)
通信方向	NG(IN: Direction mismatch with t...	1(0x1)
転送タイプ	Interrupt	3(0x3)
同期タイプ	No Synchron...	0(0x0)
使用タイプ	Data endpoint	0(0x0)

※パイプ情報ヘジャンプ機能に対応していないファームウェアでは、メニュー選択時に何も起きません。

右クリック→[パイプ情報ヘジャンプ]を選択

ソースが自動で選択されます。
エンドポイント 1(EP1)側をチェックします。

```

/* End point table for Demonstration */
uint16_t usb_gphid_EpTbl[] =
{
    USB_PHID_USE_PIPE_IN, /* Pipe No. 0 */
    /* TYPE / DIR / EPNUM */
    USB_INT | USB_DIR_P_OUT, /* PIPECFG */
    USB_NONE, /* PIPEBUF */
    USB_INTEPMAXP, /* PIPEMAX */
    USB_NONE, /* PIPEPER */
    USB_CUSE, /* FIFO Access mode */
};

#if USB_PHID_MODE != USB_PHID_MOUSE_MODE
    USB_PHID_USE_PIPE_OUT, /* Pipe No. 1 */
    /* TYPE / DIR / EPNUM */
    USB_INT | USB_DIR_P_OUT, /* PIPECFG */
    USB_NONE, /* PIPEBUF */
    USB_INT_OUT_EPMAXP, /* PIPEMAX */
    USB_NONE, /* PIPEPER */
    USB_CUSE, /* FIFO Access mode */
#endif /* USB_PHID_MODE != USB_PHID_MOUSE_MODE */
};
        
```

EP1 OUT となっています

EP2

Step④ パイプ情報テーブルのソースを修正します。

上記、Step③で発見したコードを"USB_DIR_P_IN"に修正し、再ビルドおよび実行します。

その後、更新

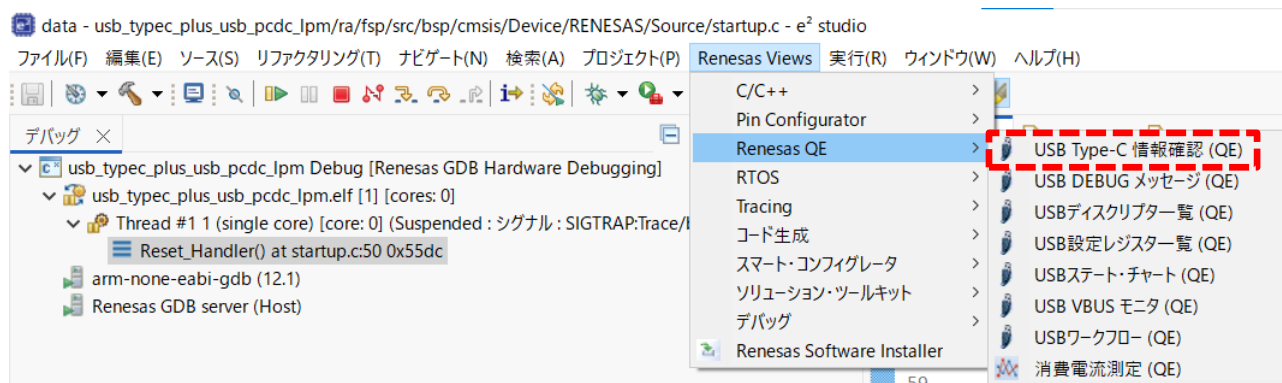
設定の食い違い問題が修正され、NG が解消されていることが確認できます。

図 6-2 ディスクリプタのデバッグ方法

7. QE for USB で USB Type-C の情報をチェック

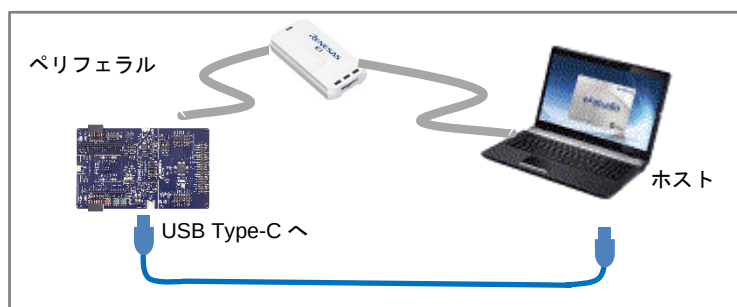
7.1 USB Type-C 情報確認 (QE) ビューで供給電力を表示

Step① USB Type-C 情報確認(QE)ビューを表示します。

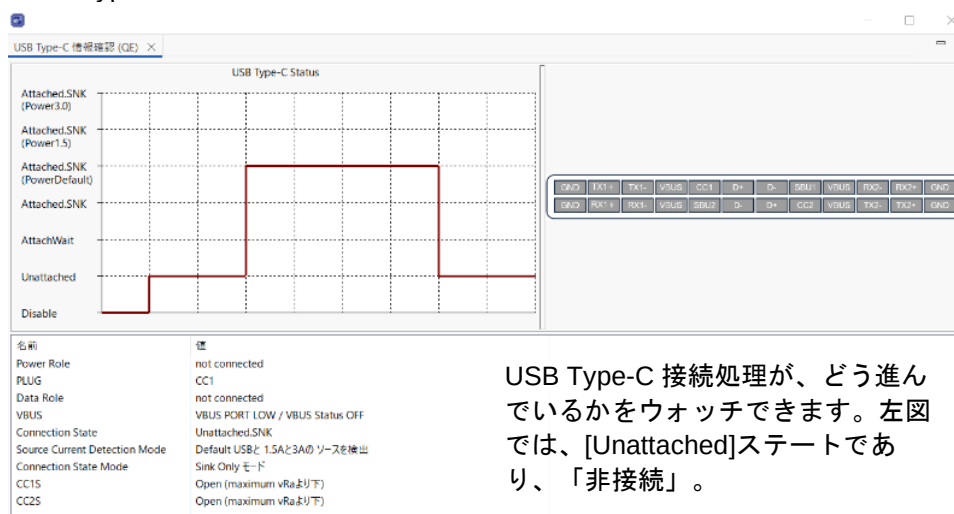


この状態で、システムを実行させ、ターゲットボードと、PC(ホスト)を USB ケーブルで接続します。

Step② USB Type-C ケーブル接続



Step③ USB Type-C ステータス表示



USB Type-C 接続処理が、どう進んでいるかをウォッチできます。左図では、[Unattached]ステータスであり、「非接続」。

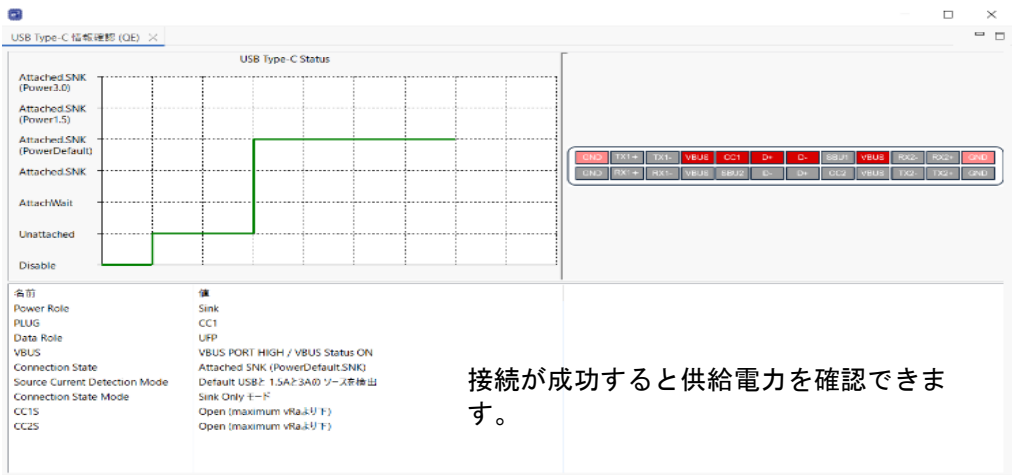
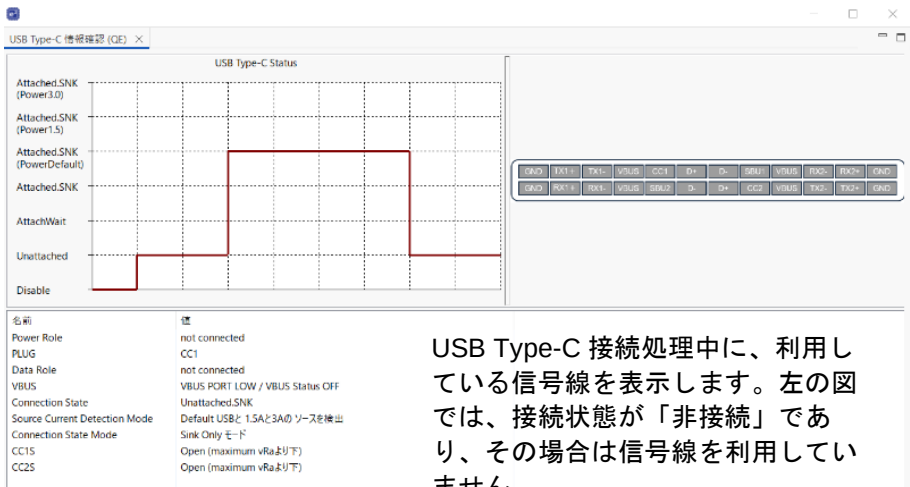


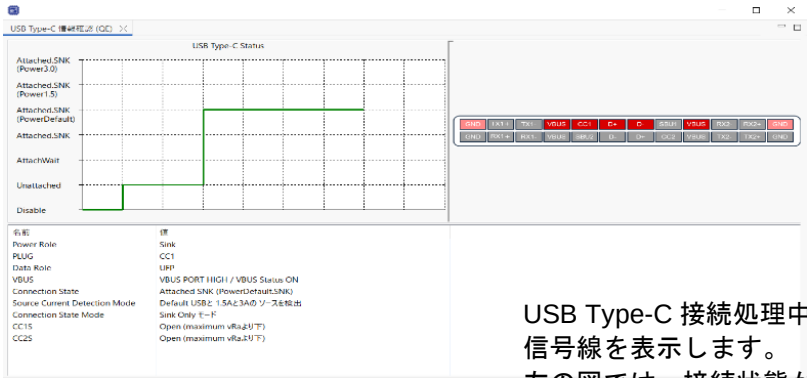
図 7.1 USB Type-C 供給電力のチェック方法

7.2 USB Type-C 情報確認 (QE) ビューで利用している信号線を表示

Step① } 7.1USB Type-C 情報確認 (QE) ビューで供給電力を表示 と同様の操作を行ってください。
Step② }

Step③ USB Type-C ステート表示





USB Type-C 接続処理中に、利用している信号線を表示します。
左の図では、接続状態が「接続(power default)」であり、その場合に利用している信号線を表示しています。

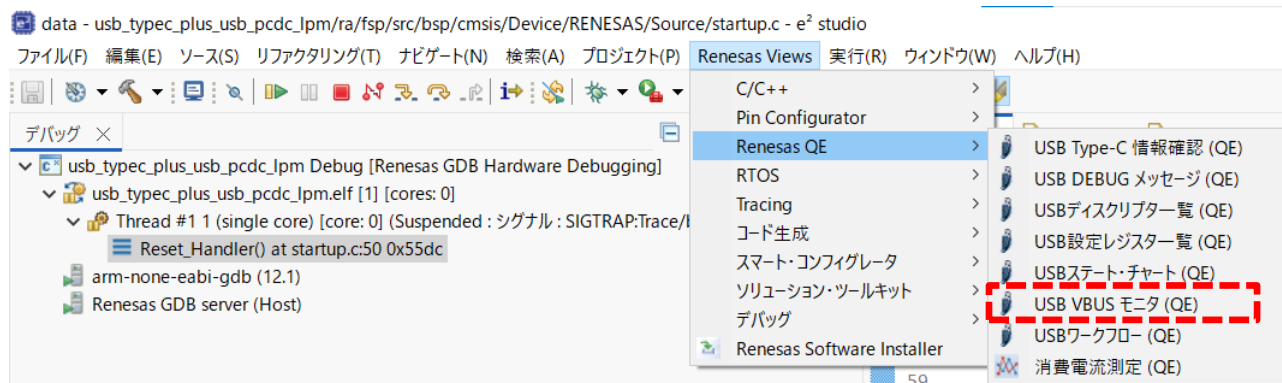
図 7.2 USB Type-C 信号線のチェック方法

8. QE for USB で USB VBUS モニタの情報をチェック

8.1 USB VBUS モニタ (QE) ビューで電圧と電流を表示

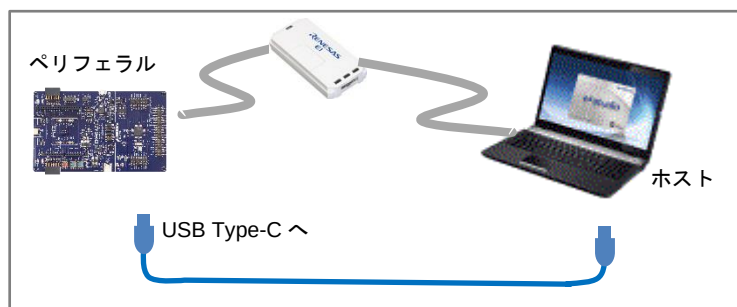
ここでは、USB VBUS モニタの電圧と電流を確認することができます。
 ※この機能を使うには、VBUS の電圧と電流を測定するサンプルプログラムが必要です。

Step① USB VBUS モニタ(QE)ビューを表示します。



この状態で、システムを実行させ、
 ターゲットボードと、PC(ホスト)を USB ケーブルで
 接続します。

Step② USB Type-C ケーブル接続



Step③ USB Type-C ステート表示



USB Type-C 接続時の電流と電圧を
 表示します。

図 8.1 USB Type-C 電流と電圧のチェック方法

9. QE for USB V2.0.0 のサポート USB ファームウェア一覧

以下の USB ファームウェアの、ペリフェラル機能に対応しています。

デバイス	ファームウェア	Rev.
RX231, RX111	USB Basic Mini Host and Peripheral Driver (USB Mini Firmware) Firmware Integration Technology	1.30
	USB Peripheral Mass Storage Class Driver for USB Mini Firmware Firmware Integration Technology	1.30
	USB Peripheral Communications Device Class Driver for USB Mini Firmware Firmware Integration Technology	1.30
	USB Peripheral Human Interface Device Class Driver for USB Mini Firmware Firmware Integration Technology	1.30
	USB Peripheral Mass Storage Class Driver for USB Mini Firmware Using Firmware Integration Technology Modules	1.30
	USB Peripheral Communications Devices Class Driver for USB Mini Firmware Using Firmware Integration Technology Modules	1.30
	USB Peripheral Human Interface Devices Class Driver for USB Mini Firmware Using Firmware Integration Technology Modules	1.30
RX65N, RX651 RX64M, RX71M	USB Basic Host and Peripheral Driver Firmware Integration Technology	1.40
	USB Peripheral Mass Storage Class Driver (PMSC) Firmware Integration Technology	1.40
	USB Peripheral Communications Device Class Driver (PCDC) Firmware Integration Technology	1.40
	USB Peripheral Human Interface Device Class Driver Firmware Integration Technology	1.40
	USB Peripheral Mass Storage Class Driver(PMSC) Using Firmware Integration Technology Modules	1.40
	USB Peripheral Communications Device Class Driver(PCDC) Using Firmware Integration Technology Modules	1.40
	USB Peripheral Human Interface Devices Class Driver Using Firmware Integration Technology Modules	1.40
RL78/G1C, RL78/L1C	USB Host and Peripheral Basic Mini Firmware	2.15
	USB Peripheral Mass Storage Class Driver (PMSC) using Basic Mini Firmware	2.15
	USB Peripheral Communications Device Class Driver (PCDC) using USB Basic Mini Firmware	2.15
	USB Peripheral Human Interface Devices Class Driver (PHID) using Basic Mini Firmware	2.15
RA2L2	RA Flexible Software Package (FSP)	5.9.0

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2016/5/20	全	初版
1.10	2025/5/28	17-21	QE for USB V2.0.0 に合わせて記述を更新

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。

標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等

高品質水準： 輸送機器（自動車、電車、船舶等）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。

7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア／ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア／ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/