μT-Kernel 3.0 BSP2 スタートガイド

e² studio & EK_RA8M1編

スタートガイドについて

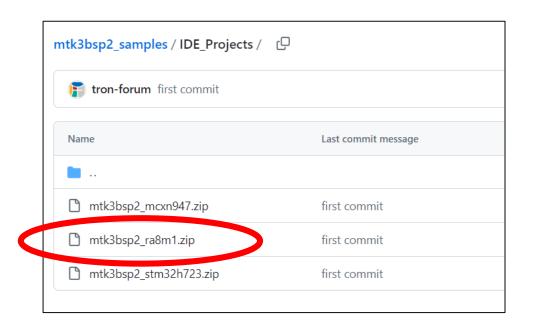


- ■本スタートガイドは、µT-Kernel 3.0 BSP2とマイコンメーカの提供するIDE(統合開発環境)を使用して、マイコンボードで実行するプログラムの作成、デバッグの基本的な方法を説明します。
- ■µT-Kernel 3.0 BSP2やIDEなどの詳細な情報は、他のドキュメントを参照してください。

μT-Kernel 3.0 BSP2のダウンロード



- µT-Kernel 3.0 BSP2のプロジェクト mtk3bsp2_ra8m1.zipをダウンロードします。
 - https://github.com/tron-forum/mtk3bsp2_samples/tree/main/IDE_Projects
- Zipファイルを任意のディレクトリに展開します。
 - Zipファイルを展開するディレクトリのパス名に日本語が入らないように注意してください。



IDEのインストール



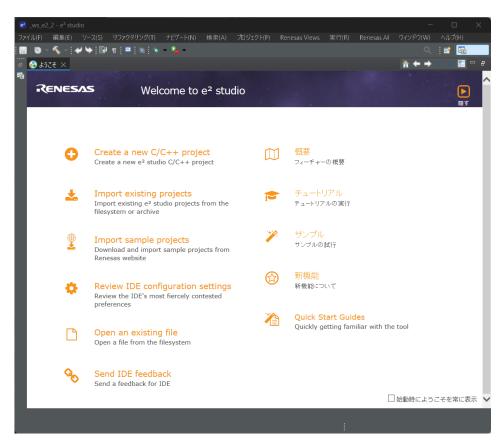
- Flexible Software Package (FSP)のインストーラを以下よりダウンロードしインストールします。
 - https://www.renesas.com/jp/ja/software-tool/flexible-software-packagefsp
 - 統合開発環境(e² studio)も一緒にインストールされます。
 - e² studio、FSPについて詳細は上記のWebサイトをご覧ください。



e² studioの実行



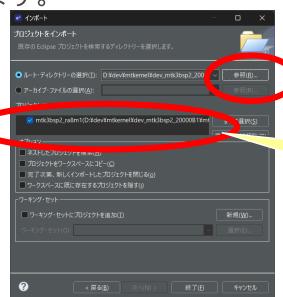
- インストールしたe² studioを実行します。
 - 起動時にワークスペースを聞かれます。任意のディレクトリを指定してください。ここにIDEの各種情報が保存されます。



プロジェクトのインポート



- ① メニュー[ファイル]→[インポート]を選択します。
- ② 開いたダイアログから[一般]→[既存プロジェクトをワークスペースへ]を選択し[次へ]を押下します。
- ③ [ルート・ディレクトリの選択]の[参照]ボタンを押し、BSP2のプロジェクトのディレクトリを指定します。
- ④ BSP2のプロジェクトが表示されていることを確認のうえ[終了]を 押下します。



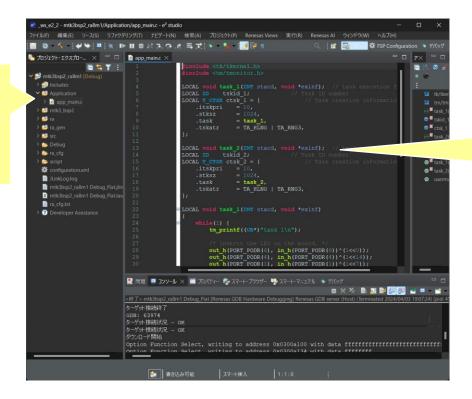
- ③ [ルート・ディレクトリの選択]の[参照] ボタンを押し、BSP2のプロジェクトの ディレクトリを指定します。
- ④ BSP2のプロジェクトが表示されている ことを確認のうえ[終了]を押下します。

プロジェクトの表示



- インポートが正常に終了すると、プロジェクトマネージャーにµT-Kernel 3.0 BSP2のプロジェクトが表示されます。
- 表示されているファイルをダブルクリックすると、その内容が表示され、編集ができます。

BSP2のプロジェクトが表示されます。



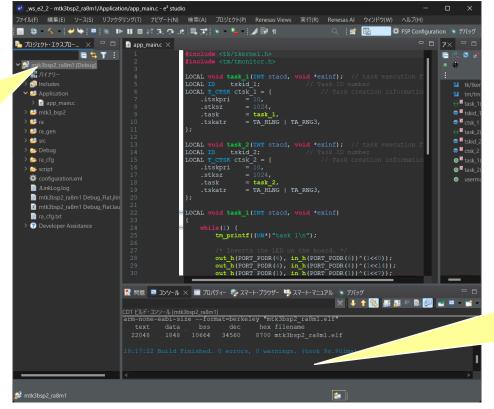
BSP2のプロジェクト の選択したファイルが 表示され、編集ができ ます。

プロジェクトのビルド



- ■プロジェクトマネージャーのプロジェクト名を右クリックし、[プロジェクトのビルド]を選択します。
- プロジェクトのビルドが開始され、正常に終了すると「Build Finished.」が表示されます。

プロジェクト名を 右クリックし、[プロジェクトのビルド]を選択します。



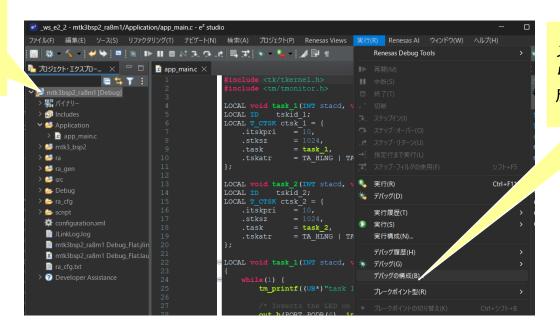
ビルドが正常に終 了すると「Build Fi nished.」が表示さ れます。

プログラムの実行とデバッグ(1)



- ボード(EK-RA8M1)とPCをUSBで接続します。
- シリアル通信を行う場合は、ボードのUART(TXD3, RXD3)をUSBシリアル変換器を介して、PCのUSBに接続します。
- プロジェクトマネージャーのプロジェクト名を選択した状態でメニュー[実行]から[デバッグの構成]を選びます。

プロジェクト名を選択します。



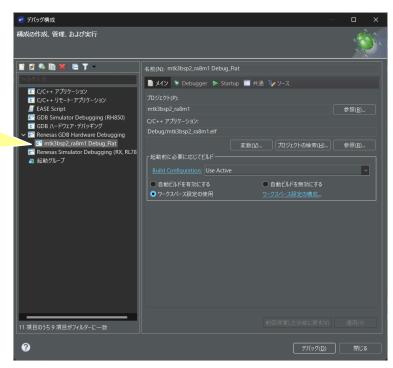
メニュー[実行]から[デバッグの構成]を選びます。

プログラムの実行とデバッグ(2)



- 表示されたダイアログから[Renesas GDB Hardware Debugging] の[mtk3bsp2_ra8m1 Debug_Flat]を選択します。
 - [mtk3bsp2_ra8m1 Debug_Flat]が表示されていない場合は、[Renesas GDB Hardware Debugging]をダブルクリックしてください。
 直前にビルドしたプロジェクトが対象となります。ビルドの直後に操作してください。

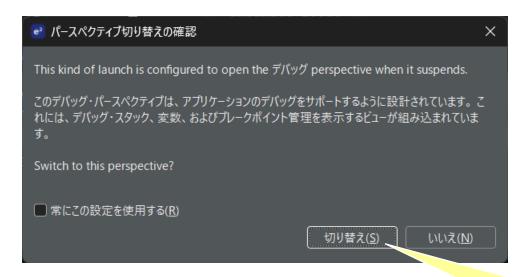
[mtk3bsp2_ra8m1 Debug_Flat]を選択 します。



プログラムの実行とデバッグ(3)



- ダイアログの[デバッグ]ボタンを押すと、実行プログラムがボードに 転送されて実行されデバッグが始まります。
- [デバッグ・パースペクティブ]への切り替えが表示されますので[切り替え]ボタンを押下します。デバッグ画面に切り替わります。

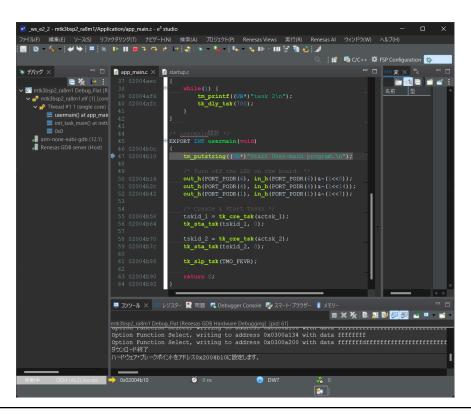


[切り替え]を押下します。

プログラムの実行とデバッグ(4)



- デバッグが開始すると、app_main.cのusermain関数でブレークします。
 - 一番最初に最初にリセットハンドラでブレークします。
- メニューバーのボタンから以下の基本的なデバッグ操作が可能です。
 - e² studioの使用方法は、メーカのWebサイトなどをご覧ください。



プログラムの実行とデバッグ(5)



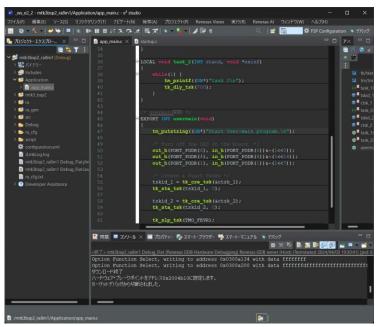
- ■ボードのプログラムからのtm_printf関数によるデバッグ用シリアル 出力は、PCのUSBの仮想シリアルポートに入力されます。
- PCでターミナルソフトを実行すると、デバッグ用シリアル出力を表示することができます。
 - PCのターミナルソフトにはTera Termなどが使用できます。
 - シリアル通信の設定は以下にしてください。



ユーザプログラムの作成



- µT-Kernel 3.0 BSP2のApplicationディレクトリにユーザプログラムを記述します。
 - ユーザプログラムのディレクトリは任意の場所に作成可能です。
 - 他のディレクトリから独立に作成しておくと、BSP2のバージョンアップの際に移 行が楽になります。
- 初期状態では、タスクを2つ実行し、それぞれのタスクがボード上のLEDの 点滅とデバッグ用シリアル出力を行うプログラムがapp_main.cファイルに 記述されています。



ペリフェラルの制御



- µT-Kernel 3.0 BSP2は、A/DコンバータとI²C通信のサンプルデバイスドライバが組み込まれています
 - サンプルデバイスドライバからはEK_RA8M1ボードのArduino互換コネクタの以下の信号が使用可能です
 - 他の信号もプロジェクトのコンフィギュレーション等の変更により使用できます

信号名	デバイス名	機能
Arduino A0	hadca	アナログ信号入力
Arduino A2	hadca	アナログ信号入力
Arduino I ² C	htiica	I ² C通信(マスター)



Arduino 互換インタフェース