



# μT-Kernel 3.0 BSP2 スタートガイド

MCUXpresso IDE & FRDM-MCXXN947編

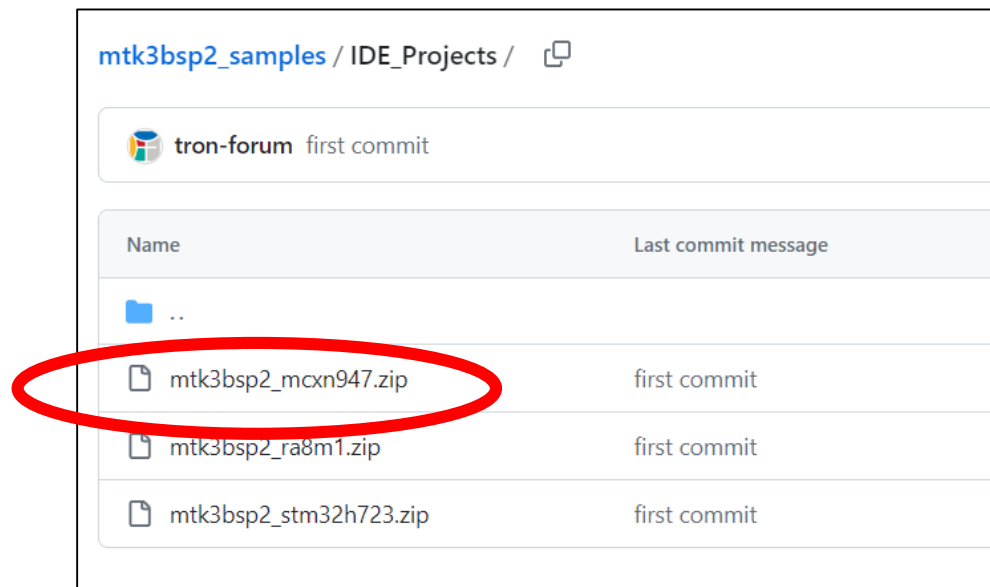


- 本スタートガイドは、 $\mu$ T-Kernel 3.0 BSP2とマイコンメーカーの提供するIDE(統合開発環境)を使用して、マイコンボードで実行するプログラムの作成、デバッグの基本的な方法を説明します。
- $\mu$ T-Kernel 3.0 BSP2やIDEなどの詳細な情報は、他のドキュメントを参照してください。

# 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\_mcxn947.zipをダウンロードします。
  - [https://github.com/tron-forum/mtk3bsp2\\_samples/tree/main/IDE\\_Projects](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



- 以下のMCUXpresso IDE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.nxp.jp/design/design-center/software/development-software/mcuxpresso-software-and-tools-/mcuxpresso-integrated-development-environment-ide:MCUXpresso-IDE>
  - MCUXpresso IDEについて詳細は上記のWebサイトをご覧ください。



My NXP アカウント 日本語 

ホーム / デザイン・センター / ソフトウェア / 開発ソフトウェア / MCUXpressoソフトウェアおよびツール / MCUXpresso統合開発環境 (IDE)

## MCUXpresso統合開発環境 (IDE)

MCUXpresso-IDE [フォローする](#)

概要 [ソフトウェア詳細](#) [ドキュメント](#) [設計・リソース](#) [トレーニング](#) [サポート](#) [ダウンロード](#)



画像をロールオーバーしてズームイン

MCUXpresso IDEは、汎用クロスオーバーMCUやワイヤレス対応MCUを含む、Arm® Cortex®-Mコアを搭載したNXP® MCUに対応するEclipseベースの使いやすい開発環境を実現します。MCUXpresso IDEは高度な編集機能、コンパイル機能、デバッグ機能を備え、さらにMCU固有のデバッグ・ビュー、コードのトレースとプロファイリング、マルチコア・デバッグ、統合設定の各ツールが追加されています。MCUXpresso IDEのデバッグ接続機能は、Freedom、Tower®システム、LPCXpresso、i.MX RTベースの評価ボード、カスタムの開発ボードに加え、オープンソースの最適化されたデバッグ・プローブと、NXP製、P&E Micro®製、SEGGER®製の最適化された製品版デバッグ・プローブをサポートしています。

[ユーザーガイド](#)

# MCUXpresso IDEの実行



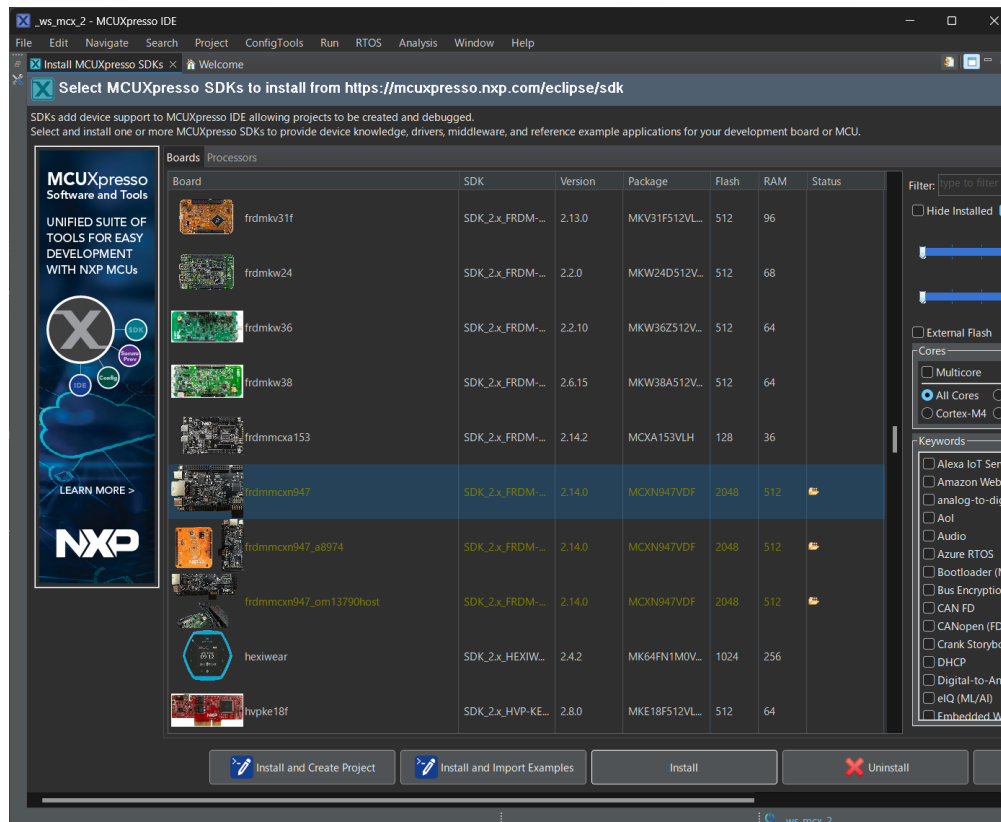
- インストールしたMCUXpresso IDEを実行します。
  - 起動時にワークスペースを聞かれます。任意のディレクトリを指定してください。ここにIDEの各種情報が保存されます。



# MCUXpresso SDKのインストール



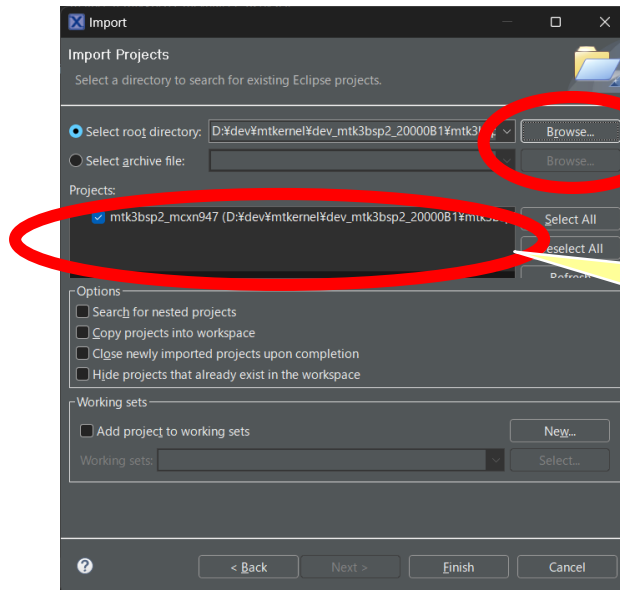
- MCUXpresso IDEのWelcome画面からDownload and Install SDKsを選び、frdmmcxn947のSDKをインストールします。



# プロジェクトのインポート



- ① メニュー[File]→[Import]を選択します。
- ② 開いたダイアログから[General]→[Existing Projects into Workspace]を選択し[Next]を押下します。
- ③ [Select root directory]の[Browse]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[Finish]を押下します。



③ [Select root directory]の[Browse]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。

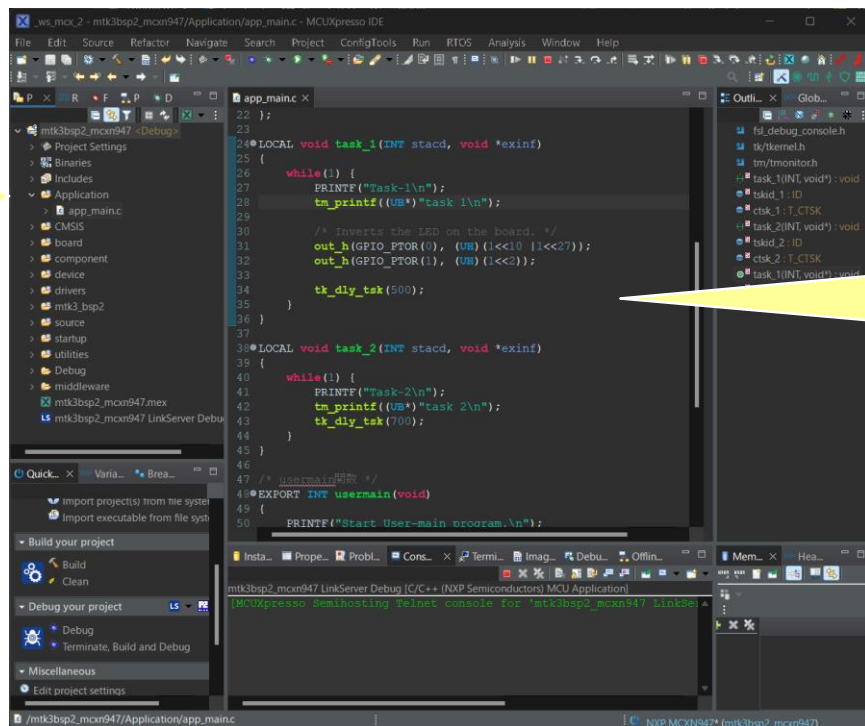
④ BSP2のプロジェクトが表示されていることを確認のうえ[Finish]を押下します。

# 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のプロジェクトは表示されます。



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。

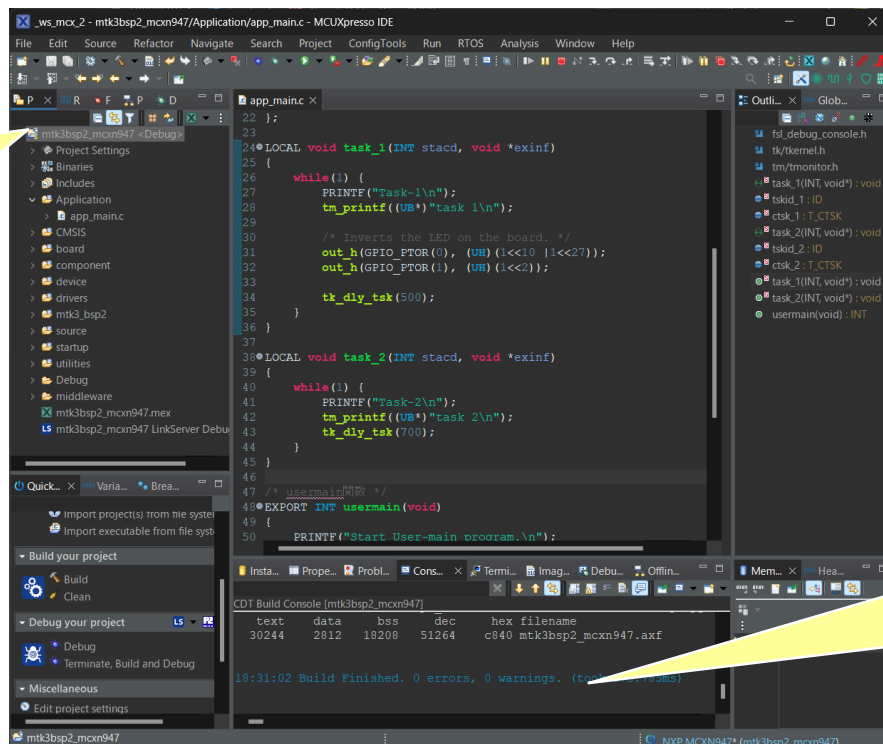


# プロジェクトのビルド



- 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、[Build Project]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、[B  
uild Project]を選  
択します。

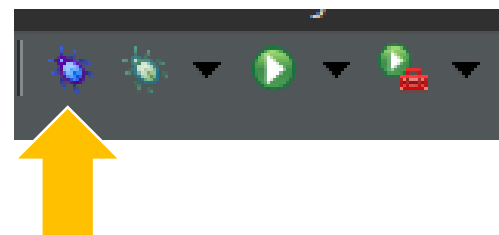
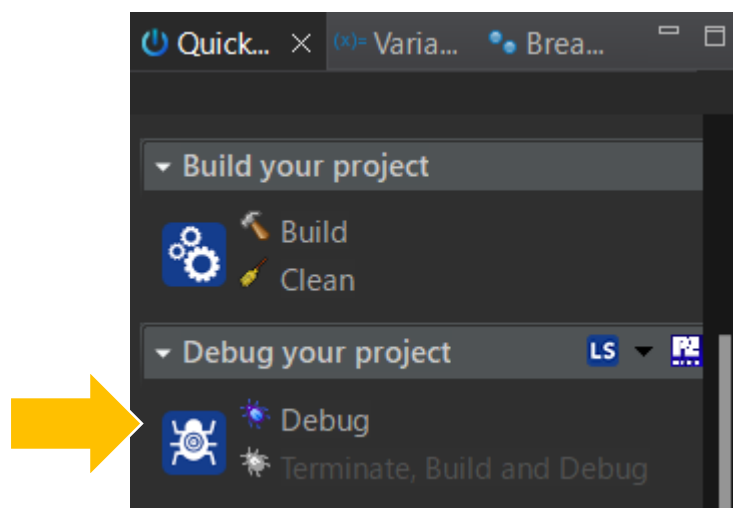


ビルドが正常に終  
了すると「Build F  
inished.」が表示さ  
れます。

# プログラムの実行とデバッグ(1)



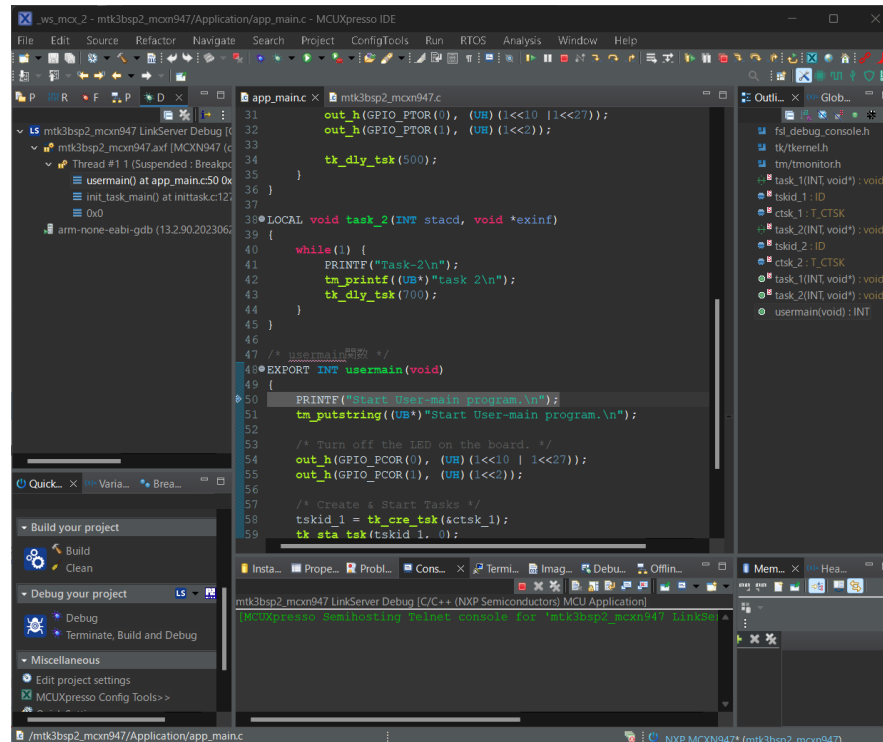
- ボード(FRDM-MCXM947)とPCをUSBで接続します。
  - USBはデバッガI/Fとシリアル通信I/Fを兼ねています。
- [Quickstart Panel]の[Debug your project]の[Debug]、またはツールバーのボタンから、デバッグを開始できます。



# プログラムの実行とデバッグ(2)



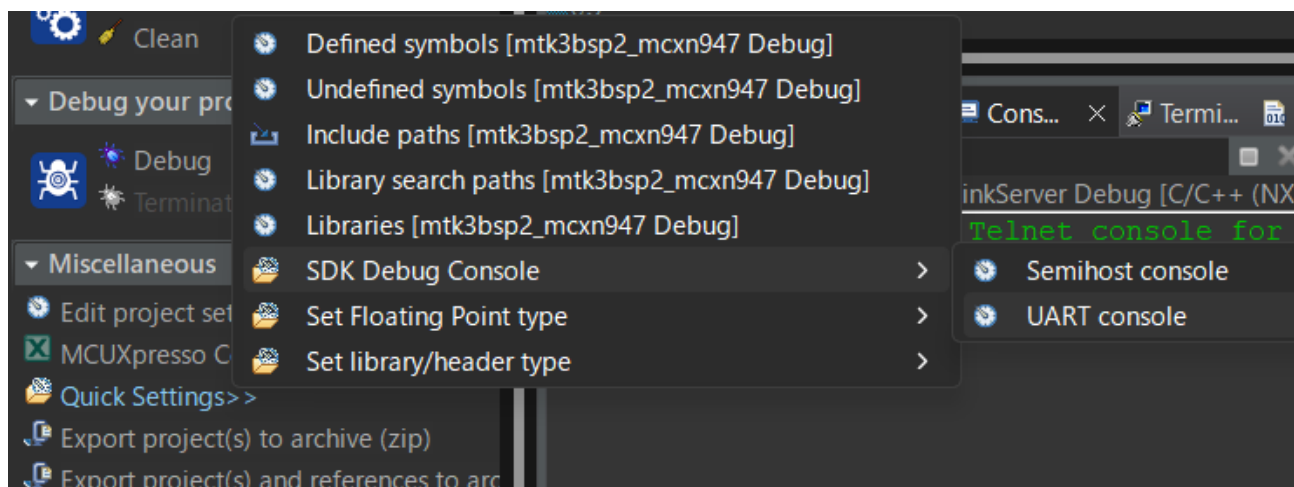
- デバッグが開始すると、app\_main.cのusermain関数でブレークします。
- メニューバーのボタンから以下の基本的なデバッグ操作が可能です。
  - MCUXpresso IDEの使用方法は、メーカーのWebサイトなどをご覧ください。



# プログラムの実行とデバッグ(4)



- SDKのPRINTF関数によりデバッグ出力を行うことができます。
- デバッグ出力の出力先は[Quickstart Panel]の[Miscellaneous]の[Quick Setting]から[SDK Debug Console]を選び、以下が選択できます。
  - Semihost console
  - UART console



# プログラムの実行とデバッグ(4)



- PRINTF関数の出力先をUART consoleとした場合、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



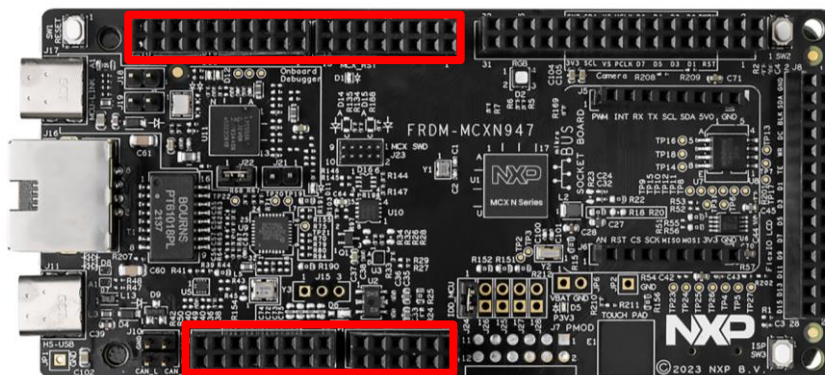
- $\mu$ T-Kernel 3.0 BSP2のApplicationディレクトリにユーザプログラムを記述します。
  - ユーザプログラムのディレクトリは任意の場所に作成可能です。
  - 他のディレクトリから独立に作成しておく、BSP2のバージョンアップの際に移行が楽になります。
- 初期状態では、タスクを2つ実行し、それぞれのタスクがボード上のLEDの点滅とデバッグ用シリアル出力を行うプログラムがapp\_main.cファイルに記述されています。

```
22 };
23
24 LOCAL void task_1(INT stacd, void *exinf)
25 {
26     while(1) {
27         PRINTF("Task-1\n");
28         tm_printf((UB*) "task 1\n");
29
30         /* Inverts the LED on the board. */
31         out_h(GPIO_FTOR(), (UB) (1<<10 | 1<<27));
32         out_h(GPIO_FTOR(), (UB) (1<<1));
33
34         tk_dly_tsk(500);
35     }
36 }
37
38 LOCAL void task_2(INT stacd, void *exinf)
39 {
40     while(1) {
41         PRINTF("Task-2\n");
42         tm_printf((UB*) "task 2\n");
43         tk_dly_tsk(700);
44     }
45 }
46
47 /* usermain関数 */
48 EXPORT INT usermain(void)
49 {
50     PRINTF("Start User-main program.\n");
```



- $\mu$ T-Kernel 3.0 BSP2は、A/DコンバータとI<sup>2</sup>C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FRDM-MCXN947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 A1	hadca	アナログ信号入力
Arduino I <sup>2</sup> C	hiica	I <sup>2</sup> C通信(マスター)



Arduino  
互換インタフェース