

PSIMチュートリアル DLLのデバッグ方法

Tutorial – Debugging DLL October 2016

Mywayプラス株式会社

【はじめに】

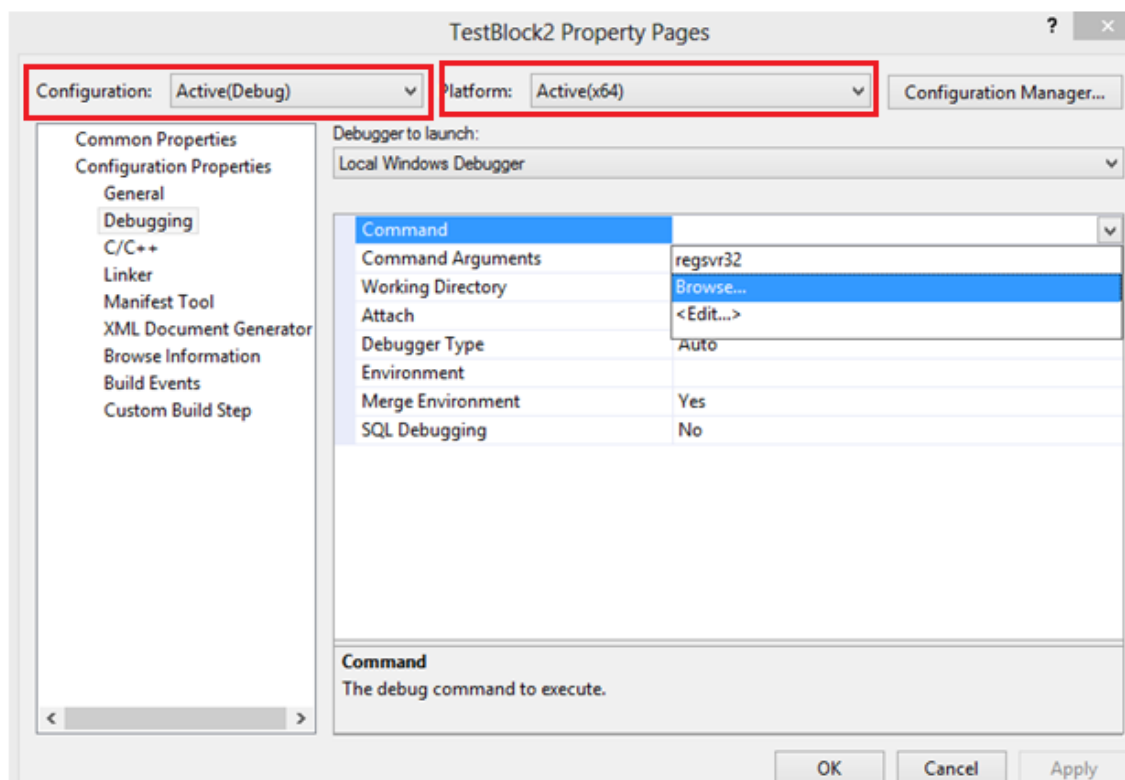
DLL ブロックを扱うときは、32 ビット DLL は 32 ビット PSIM でのみ動作し、64 ビット DLL は 64 ビット PSIM でのみ動作することに注意することが重要です。

Microsoft Visual C ++ (Visual Studio) で 64 ビット DLL をコンパイルするには、MSDN ヘルプ : 「How to Configure Visual C++ Projects to Target 64-Bit Platforms.」を参照してください。

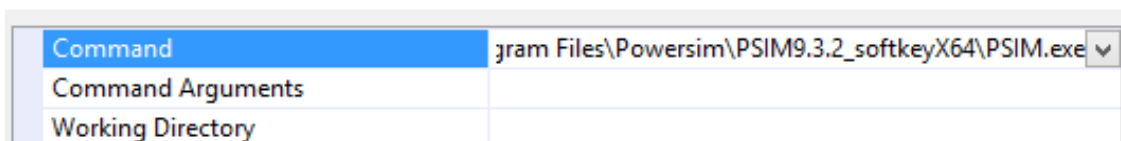
1. Microsoft Visual C ++ (DLL プロジェクト) の設定

DLL ファイルをデバッグするには、DLL プロジェクトを Microsoft Visual C ++ にロードし、Debug 構成と適切なビットプラットフォーム (32-bit または 64-bit) で再構築 (リビルド) します。「**Project >> Properties**」に移動して、プロジェクトのプロパティページを開きます。設定は "Active(Debug)" で、プラットフォームを正しく設定します。左側のパネルで、「**Configuration properties >> Debugging**」を選択し、右側のパネルで「**Command**」をクリックして、PSIM フォルダー内の "PSIM.exe" を参照します。

プロパティ ダイアログ ウィンドウを以下に示します。

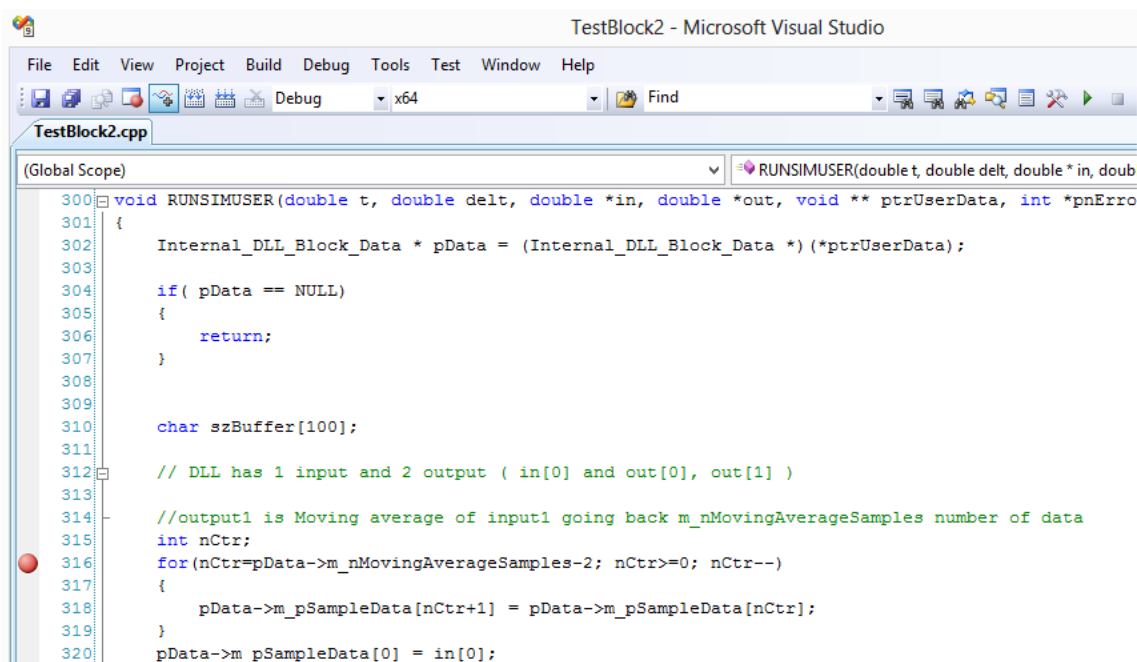


コマンドを選択した後、ダイアログは以下のようになります。



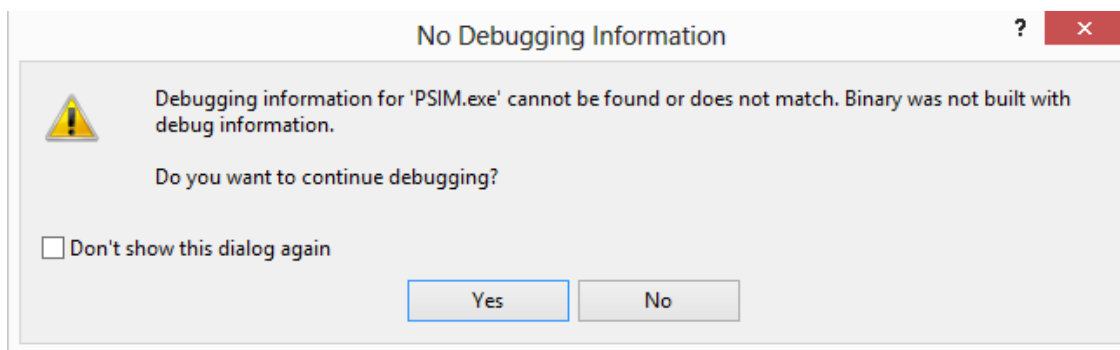
2. デバッグの準備、Breakpoint の設定

次に、下図のように、コード内で、1つ以上のブレークポイントを目的の位置に配置します。



```
TestBlock2 - Microsoft Visual Studio
File Edit View Project Build Debug Tools Test Window Help
Debug x64 Find
TestBlock2.cpp
(Global Scope) RUNSIMUSER(double t, double delt, double *in, doub
300 void RUNSIMUSER(double t, double delt, double *in, double *out, void ** ptrUserData, int *pnErro
301 {
302     Internal_DLL_Block_Data * pData = (Internal_DLL_Block_Data *) (*ptrUserData);
303
304     if( pData == NULL)
305     {
306         return;
307     }
308
309
310     char szBuffer[100];
311
312     // DLL has 1 input and 2 output ( in[0] and out[0], out[1] )
313
314     //output1 is Moving average of input1 going back m_nMovingAverageSamples number of data
315     int nCtr;
316     for(nCtr=pData->m_nMovingAverageSamples-2; nCtr>=0; nCtr--)
317     {
318         pData->m_pSampleData[nCtr+1] = pData->m_pSampleData[nCtr];
319     }
320     pData->m_pSampleData[0] = in[0];
```

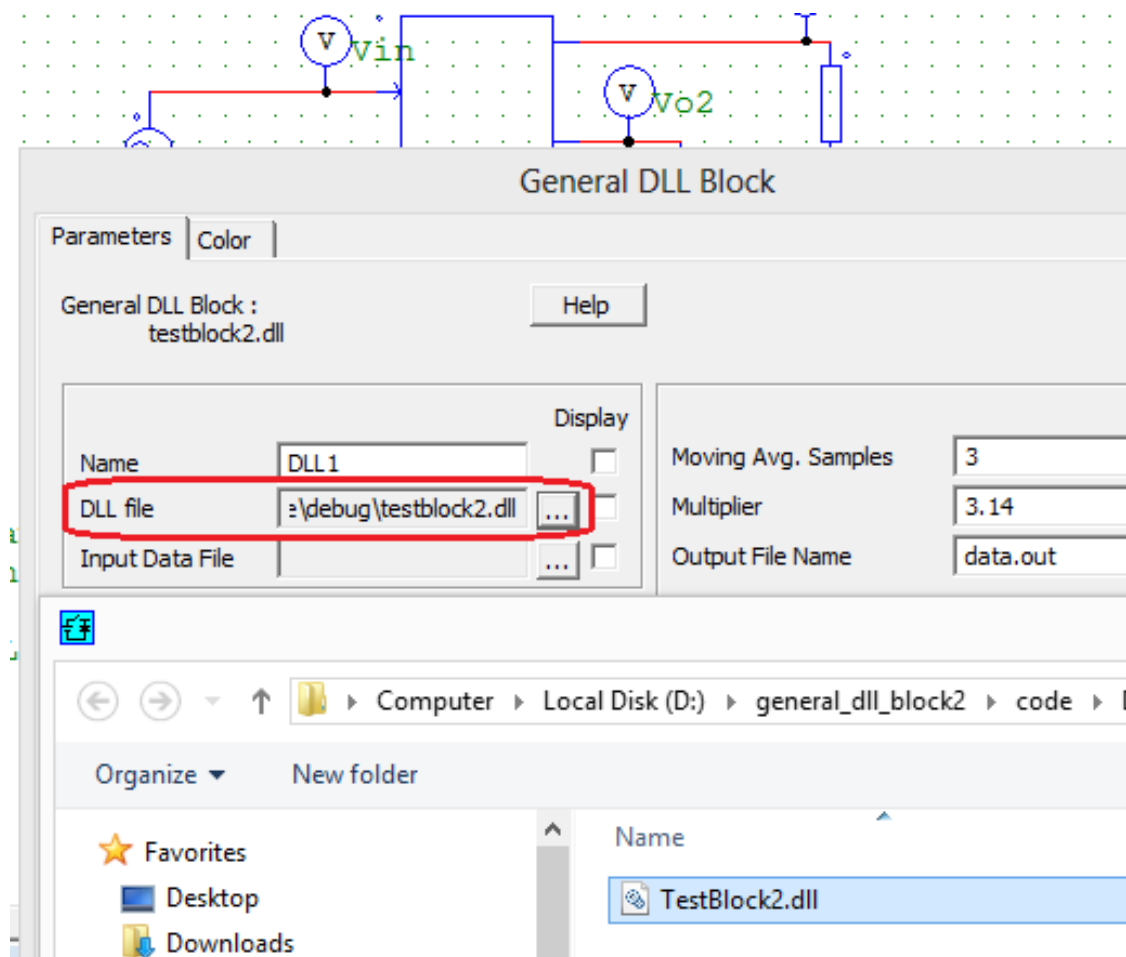
メニューから「Debug >> Start debugging」に移動します。 以下のような警告メッセージが表示されます。



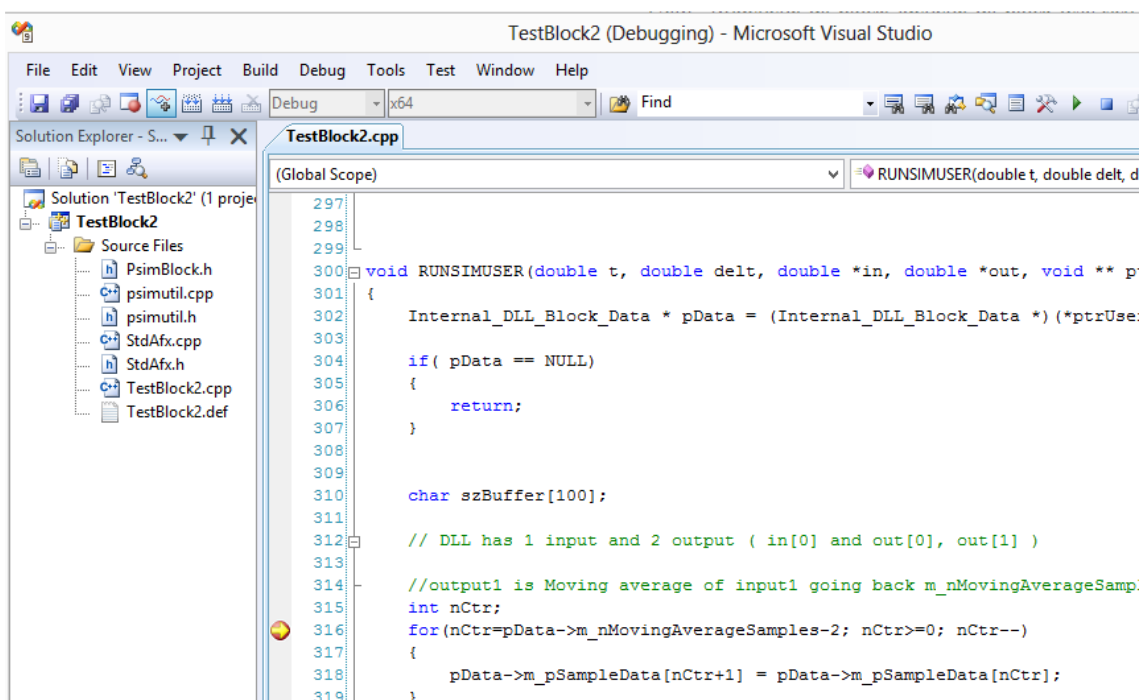
これは想定されることですので“Yes”をクリックして続行すると、PSIM が起動します。

3. PSIM の General DLL ブロック設定およびデバッグ

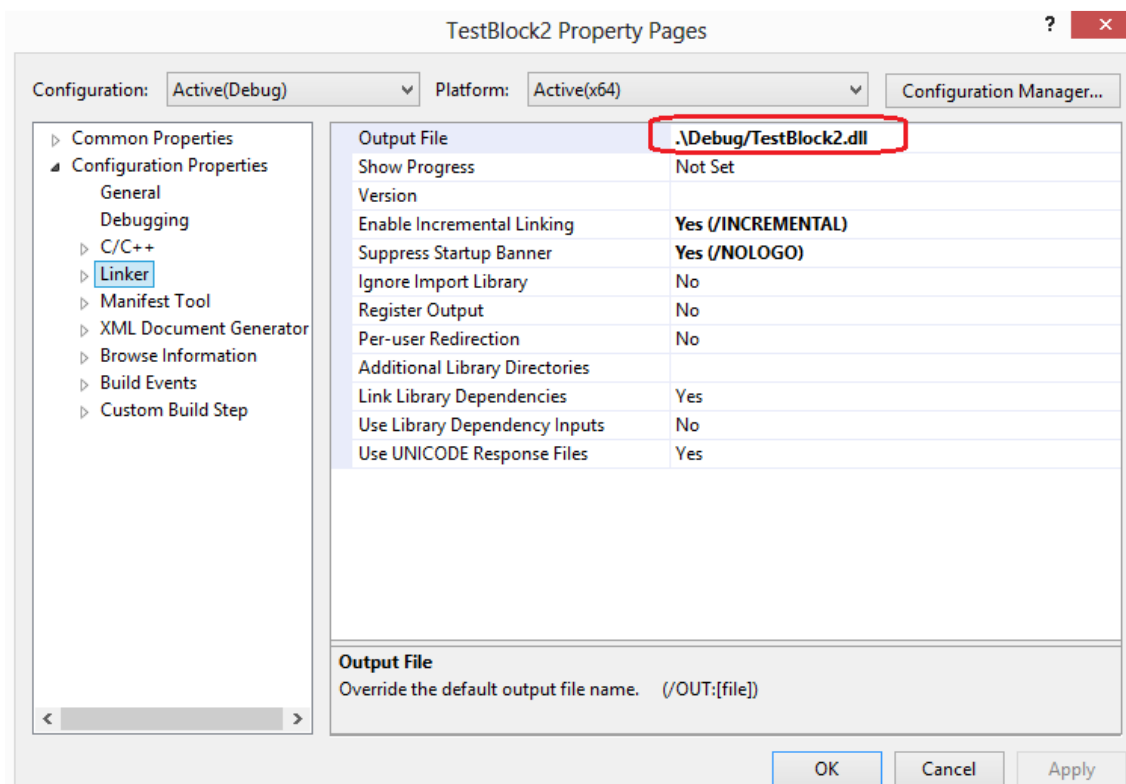
新しい回路図ファイルを作成し、General DLL ブロックを回路図に配置します。作成したばかりの DLL のデバッグバージョンを General DLL ブロックの DLL ファイルとして選択します。DLL ブロックが既に含まれている既存の回路図ファイルを開くこともできます。下の画像は、DLL ブロックの DLL ファイル定義を示しています。



必要に応じて回路を修正してください。回路図ファイルを保存してシミュレーションを実行します。以下に示すように、実行はブレークポイントで停止します。



実行が停止しない場合は、ブレークポイントの位置を確認してください。よくあるエラーは、General DLL ブロックに誤った DLL ファイルや誤ったリリースバージョンの DLL ファイルが選択される場合です。生成された DLL ファイルのパスは、次に示すように Visual C++ で表示されます。



ご注意

1. 本資料に記載された製品の仕様は、予告なく変更することがあります。
2. 本資料の内容については、万全を期しておりますが、万一ご不明な点などがありましたら、弊社までお申しつけください。
3. 本資料に記載された情報に起因する損害または特許権その他権利の侵害に関しては、弊社は一切の責任を負いません。
4. 本資料によって第三者または弊社の特許権その他権利の実施権を許諾するものではありません。
5. 弊社の書面許諾なく、本資料の一部または全部を無断で複製することを固くお断りします。
6. (本資料に記載された製品をユーザ装置に組み込む際には、バックアップやフェイルセーフ機能を系統的に設置してください。)
7. (弊社は、人命に関わる装置として特別に開発したものは用意しておりません。)
8. 本資料に記載されている会社名、商品名は、各社の商標または登録商標です。

Copyright 2019 by Myway Corporation

All rights reserved. No part of this manual may be photocopied or reproduced in any form or by any means without the written permission of Myway Corporation.

発行: Myway プラス株式会社

〒220-0022

横浜市西区新花咲町 6-145

横浜花咲ビル

TEL: 045-548-8836

FAX: 045-548-8832

ホームページ : <https://www.myway.co.jp>

Eメール : sales@myway.co.jp
