# 現象

新規にコンポーネントを作成する際にSimulation StudioからExportしたプロジェクト（.dsp）がIntel Parallel Studio XE 2017では変換に失敗する。

# 検証条件

Intel Parallel Studio XE 2017、VS2015

TRNSYS17.02.0005

Type名：MyType201

C:\Trnsys17\Studio\Proformas\My Components\MyType201.tmf

# 対策検討

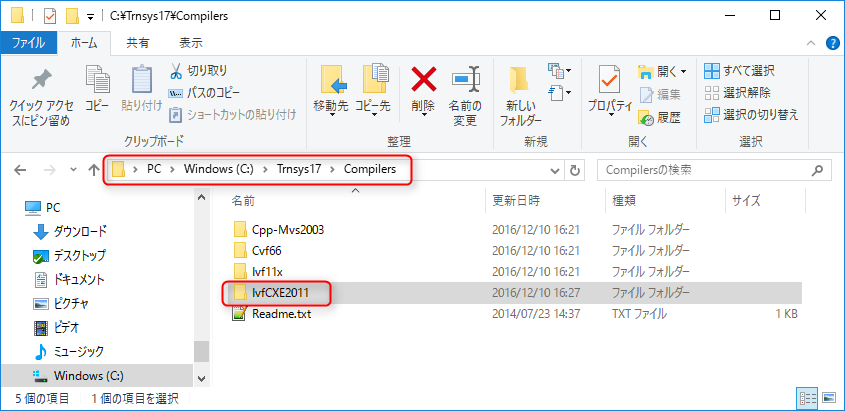
なぜかIntel Parallel Studio XE 2017(名前が長いので、以下XE2017)でコンポーネントを作ろうとするとうまく行かないようです。Simulation StudioからExportしたプロジェクト(.dsp)を変換する段階でエラーになってしまいます。

あれこれ試したのですが、どうもダメ。プロジェクト(.dsp)の変換がダメなら、ドキュメントを参考にスクラッチから新しく作ろうかと思いましたが、それもいろいろ設定が細かくて面倒です。もしかして既存のプロジェクトを元に修正すれば簡単になるかと思いついて試してみました。

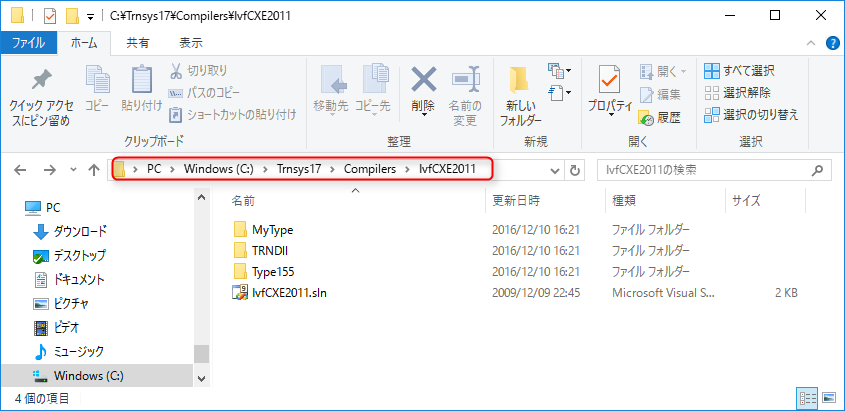
TRNSYSにはXE2011のソリューション(IvfCXE2011.sln)が添付されています。これに含まれているプロジェクトを元に新しいコンポーネント用のプロジェクトを作成してみます。

以下、すでにプロフォルマからExportしたソースコード（Type201.for）がある前提で話を進めます。

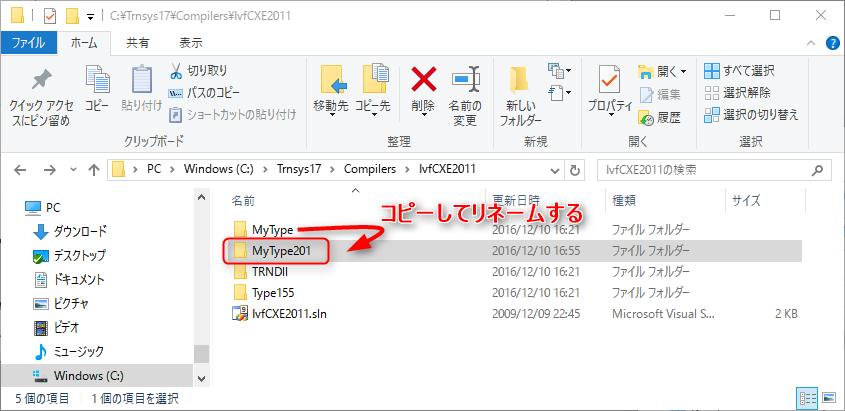
まずは、IvfCXE2011.slnの所在を確認します。「C:\TRNSYS17\Compilers」フォルダにある「IvfCXE2011」フォルダがそれです。



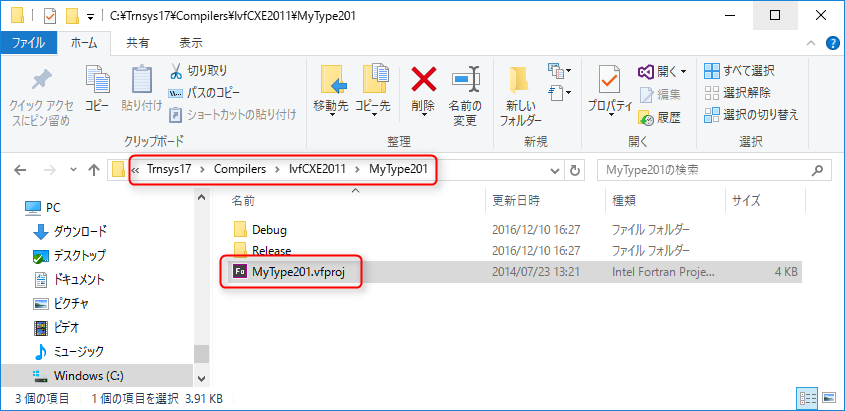
このソリューションはTRNDLL、その他をビルドするものなので、さらに3個のフォルダ（プロジェクト）が含まれています。



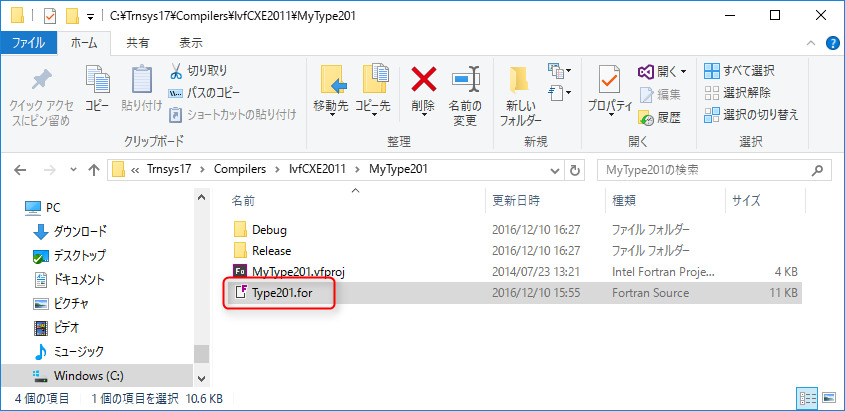
ここから「MyType」フォルダを丸ごとコピーして雛形として流用します。コピーしたら分かり易いように「MyType201」といういう名前にリネームしておきます。



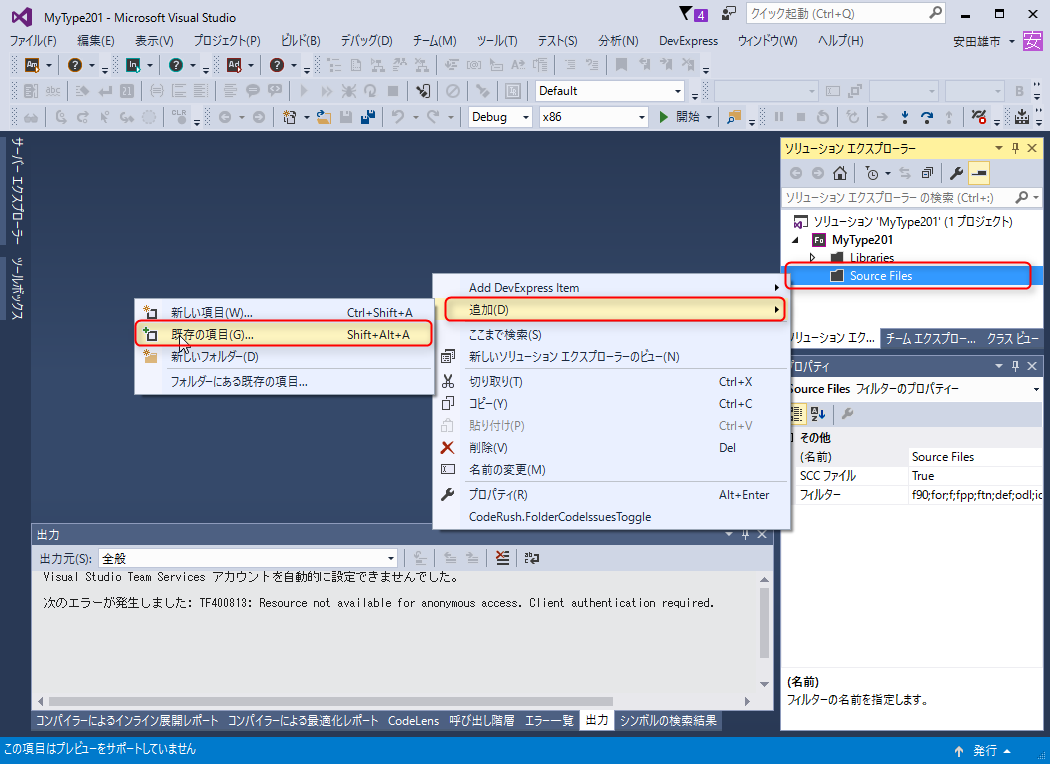
さらにフォルダの中身を覗くとプロジェクトファイル(.vfproj)があるので、これも分かり易いように「MyType201.vfproj」にリネームします。



さらにSimulation StudioからExportしたソースコードを放り込んだら準備完了です。



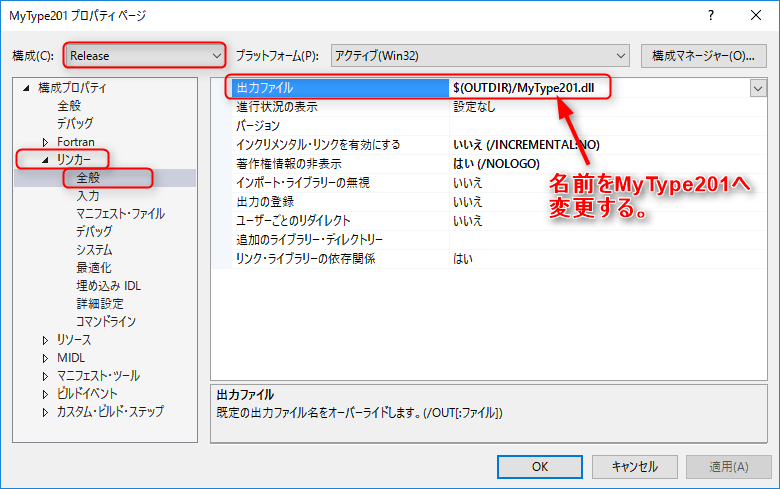
XE2017（画面はVisual Studio2015で起動）を起動。先ほど用意したプロジェクト「MyType201.vfproj」を開いて、「Sourace Files」フォルダへソースコード「Type201.for」を追加します。



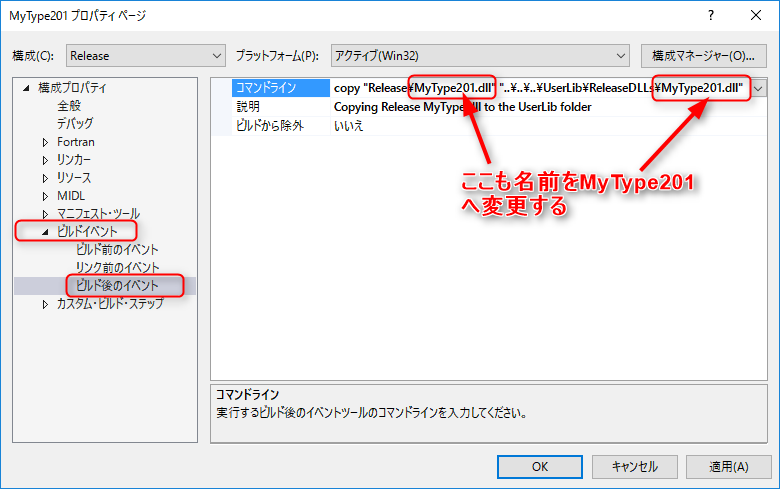
あとは、追加したソースコードに合せてプロジェクトの設定を何カ所か変更します。

[プロジェクト]-[プロパティ]で設定画面を表示して、以下の赤枠の項目を変更します。

構成「Release」で、「リンカー」の出力ファイルの項目をプロジェクト名に合せて「MyType201.dll」へ変更します。

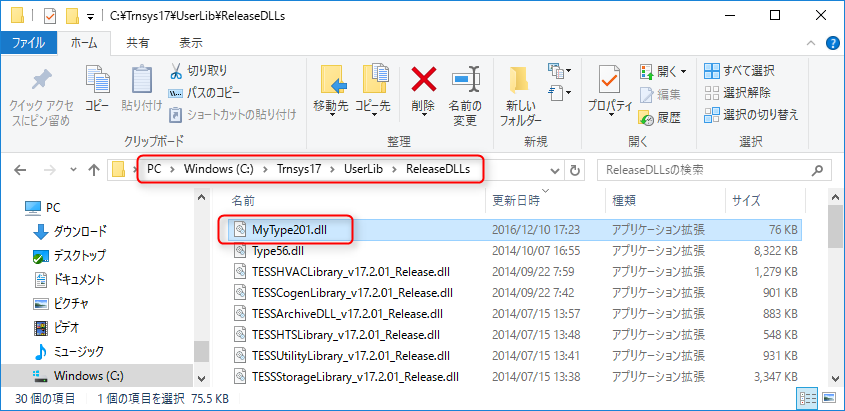


つづいて「ビルドイベント」のコマンドラインのファイル名を同じように変更します。



【重要】構成「Debug」も忘れずに同じように変更を行います。

あとはビルドして、「C:\Trnsys17\UserLib\ReleaseDLLs」フォルダにMyType201.dllができあがっていれば無事成功です。



これでType201が利用できる状態になっているので、プロジェクトを作成して計算に利用できます。

もっと面倒かと思ったけど、なんとかなった。

さらにソリューション（IvfCXE2011.sln）へプロジェクトを追加しておくと、TRNDLL.DLLのビルドも同時にできてDebug、Releaseモードの切り替えが楽になります。複数のコンポーネントを開発する場合には同じ手順でプロジェクトを作成、まとめてソリューションへ登録しておくと、ソースコードのメンテナンスがやりやすくなります。