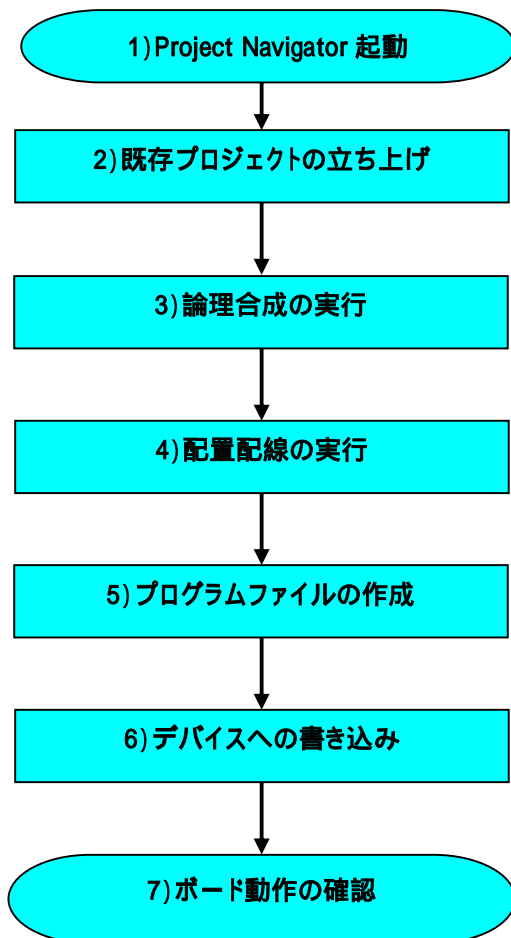

ATLYS ボード操作マニュアル (VHDL)

TOKYO ELECTRON DEVICE

本操作マニュアルは、ATLYS ボードと一緒に Digilent 社オプションモジュールである「VmodCAM - Stereo Camera Module」のリファレンスデザインをそのまま流用した操作手順のご紹介になります。リファレンスデザインは他にも WEB に掲載ございますので、活用頂き、本操作は一例としてご参照下さい。

リファレンスデザイン内の ISE プロジェクトを立ち上げ、完成済みのソースファイル、テストベンチファイル、制約ファイル(UCF ファイル)を読み込み、ファンクションシミュレーション、論理合成、配置配線、コンフィグレーションファイルの作成、デバイスへの書き込み、ボード上での動作確認を行います。下記のフローに従って下さい。



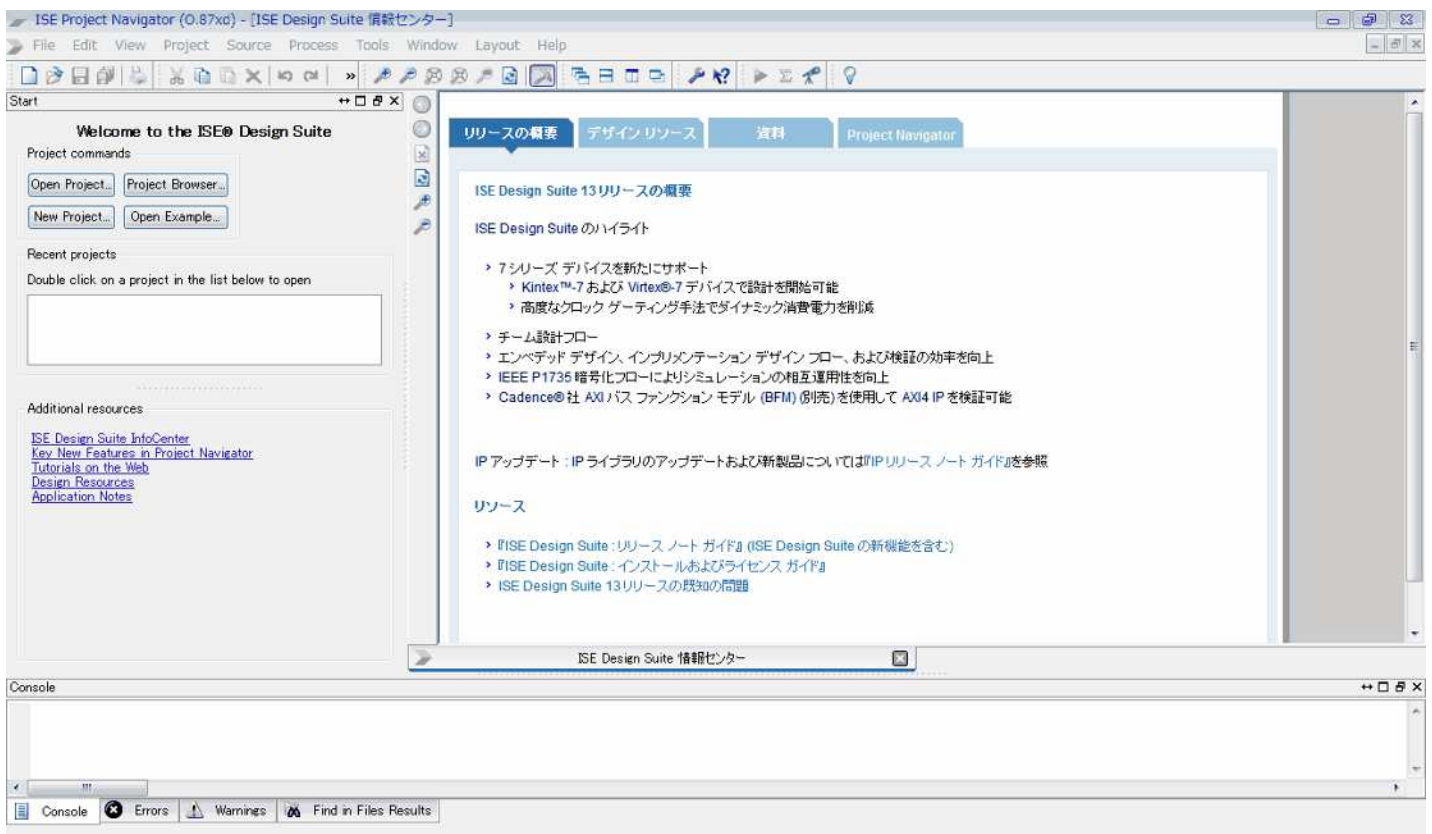
1) Project Navigator の起動

1. 「VmodCAM - Stereo Camera Module」のリファレンスデザインを任意のローカルフォルダに格納し、ISE プロジェクトを立ち上げます。リファレンスデザインは、以下の WEB サイトよりダウンロード可能です。

URL : <http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,648,931&Prod=VMOD-CAM>

(Doc# DSD_0000309)

2. Windows のスタートメニューから、[スタート] - [プログラム] - [Xilinx ISE Design Suite 13.4] - [ISE デザインツール] - [Project Navigator] を選択します。もしくは、デスクトップ上のショートカットから起動します。



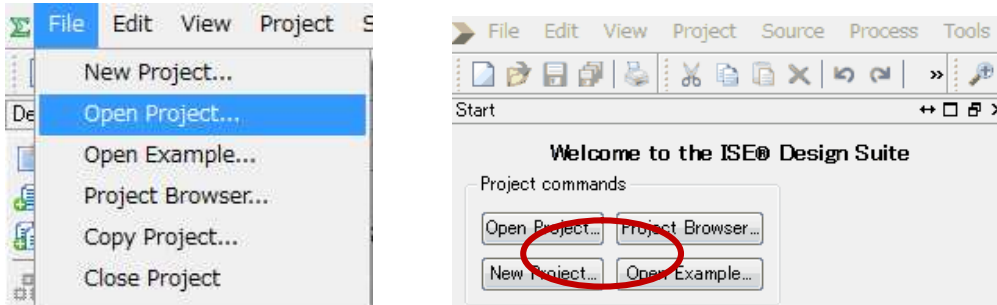
【One-Point】

上記の画面は初めてプロジェクトを立ち上げた場合の画面です。
初回以降は、前回使用したプロジェクトファイルがそのまま開きます。

2) 既存プロジェクトの格納と立ち上げ

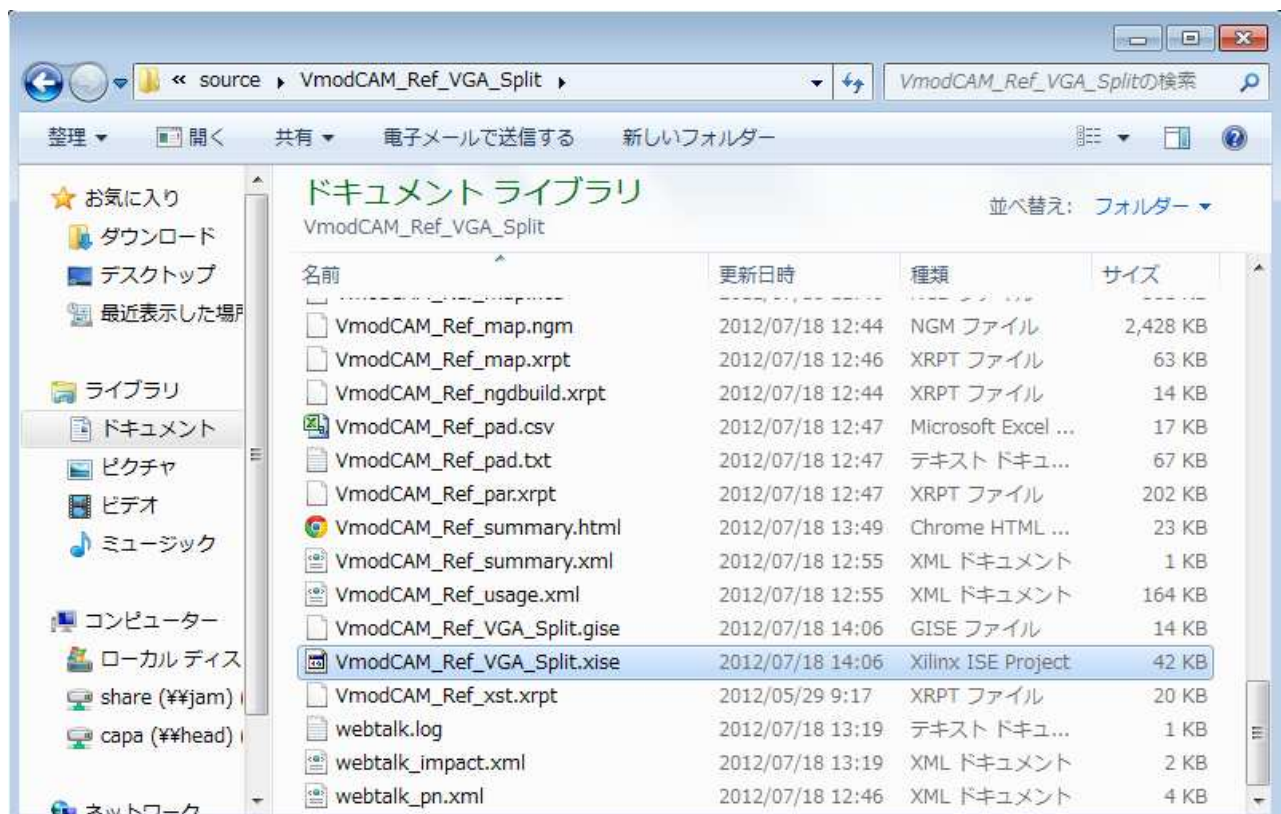
1. [File] - [Open Project] を選択します。

もしくは、画面上の [Open Project] のボタンをクリックします。



例: C:\MyDocs\VmodCAM_Ref_VGA Demo_13\source\VmodCAM_Ref_VGA_Split にある.xise が ISE のプロジェクトファイルになります。この.xise ファイルを選択します。

この.xise ファイルが、既存の ISE Project file です。





The screenshot displays the ISE Project Navigator interface for a project named 'VmodCAM_Ref_VGA_Split'. The main window is divided into several panes:

- Design Overview:** Shows a tree view of the project hierarchy, including components like 'VmodCAM_Ref - Behavioral', 'Inst_SysCon - SysCon - Behavioral', and 'MODE_SYNC - InputSyncV - Behavi...'.
- Design Properties:** A list of checkboxes for configuring the design summary, such as 'Enable Message Filtering', 'Show Clock Report', and 'Show Errors'.
- VmodCAM_Ref Project Status:** A summary table providing key project metrics.
- Device Utilization Summary:** A table detailing the usage of various resources on the target device.
- Console:** Displays warning messages from the Project Manager regarding missing files.

VmodCAM_Ref Project Status

Project File:	VmodCAM_Ref_VGA_Split.xise	Parser Errors:	No Errors
Module Name:	VmodCAM_Ref	Implementation State:	Programming File Generated
Target Device:	xc6s1x45-8csg324	Errors:	No Errors
Product Version:	ISE 13.4	Warnings:	338 Warnings (338 new)
Design Goal:	Balanced	Routing Results:	All Signals Completely Routed
Design Strategy:	Xilinx Default + Multithreading	Timing Constraints:	All Constraints Met
Environment:	System Settings	Final Timing Score:	0 (Timing Report)

Device Utilization Summary

Slice Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Registers	910	54,576	1%	
Number used as Flip Flops	910			
Number used as Latches	0			
Number used as Latch-thrus	0			
Number used as AND/OR logics	0			
Number of Slice LUTs	1,454	27,288	5%	
Number used as logic	1,381	27,288	5%	
Number using O6 output only	898			
Number using O5 output only	192			
Number using O5 and O6	291			
Number used as ROM	0			
Number used as Memory	42	6,408	1%	

Console

```

WARNING:ProjectMgmt - File C:/Users/006523/Documents/MyDocs/VmodCAM_Ref_VGA_Demo_13/source/VmodCAM_Ref_VGA_Split/ipcore_dir/dcm_recfg/example_design/dcm_re
WARNING:ProjectMgmt - File C:/Users/006523/Documents/MyDocs/VmodCAM_Ref_VGA_Demo_13/source/VmodCAM_Ref_VGA_Split/ipcore_dir/dcm_recfg/dcm_recfg.ucf is miss
WARNING:ProjectMgmt - File C:/Users/006523/Documents/MyDocs/VmodCAM_Ref_VGA_Demo_13/source/VmodCAM_Ref_VGA_Split/ipcore_dir/dcm_recfg/example_design/dcm_re
    
```

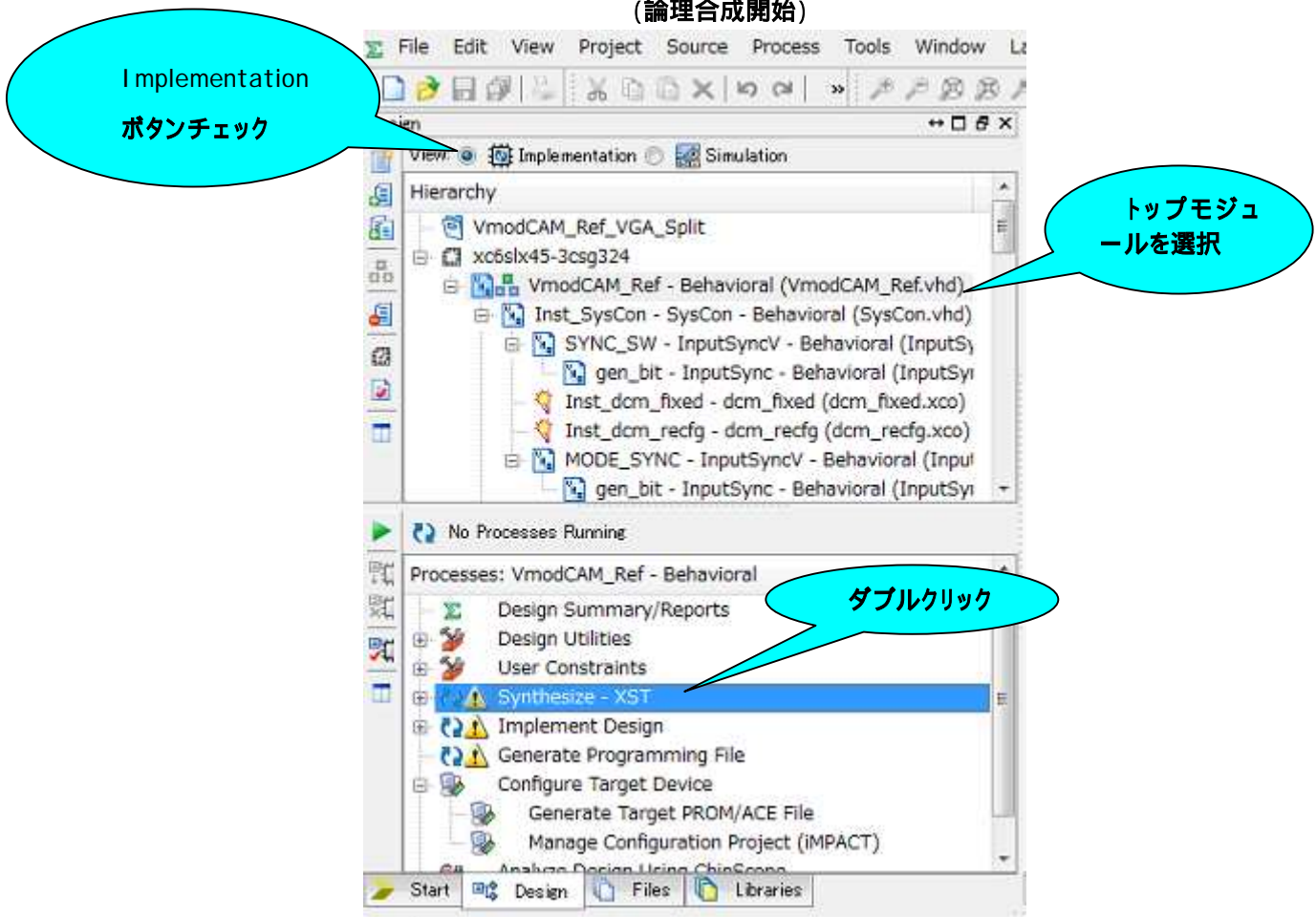
3) 論理合成の実行

1. まずは論理合成を実行します。

Implementation ボタンにチェックが入っていることを確認して下さい。トップモジュールを指定してから、プロセスウィンドウの [Synthesize-XST]をダブルクリックすると論理合成が開始されます。

⚠ や ✅ が表示されたら、論理合成の完了です。⚠ が表示された場合は必要に応じて修正して下さい。エラーがある場合、❌ が表示されますのでメッセージに従って修正して下さい。

(論理合成開始)



⚠ ワーニング や ❌ エラー詳細は、GUI 下部のコンソール(Console)ウィンドウに表示されます。

4) 配置配線の実行

1. ピン固定ファイルを読み込んだ後(今回は既に読み込み済み)、プロセスウィンドウの[Implement Design]をダブルクリックし、配置配線を実行します。

⚠ や ✅ が表示されたら、配置配線の完了です。⚠ が表示された場合必要に応じて修正して下さい。

エラーがある場合、❌ が表示されますのでメッセージに従って修正して下さい。

The screenshot displays the Xilinx ISE Design Suite interface. In the 'Processes' window, the 'Implement Design' button is highlighted, and a callout bubble with the text 'ダブルクリック' (Double-click) points to it. The 'Design Overview' window shows the 'Errors and Warnings' section. On the right, the 'VmodCAM_Ref Project Status' window displays the following information:

VmodCAM_Ref Project Status (05/29/2012 - 09:17:14)			
Project File:	VmodCAM_Ref_VGA_Split.xise	Parser Errors:	No Errors
Module Name:	VmodCAM_Ref	Implementation State:	Synthesized
Target Device:	xc6s1x45-3csg324	Errors:	No Errors
Product Version:	ISE 13.4	Warnings:	286 Warnings (0 new)
Design Goal:	Balanced	Routing Results:	
Design Strategy:	Xilinx Default + Multithreading	Timing Constraints:	
Environment:	System Settings	Final Timing Score:	

Below the project status, the 'Device Utilization Summary (estimated values)' is shown:

Logic Utilization	Used	Available	Utilization
Number of Slice Registers	928	54676	1%
Number of Slice LUTs	1679	27288	6%
Number of fully used LUT-FF pairs	779	1828	42%
Number of bonded IOBs	106	218	48%
Number of BUFG/BUFGCTRLs	9	16	56%
Number of PLL_ADVs	2	4	50%

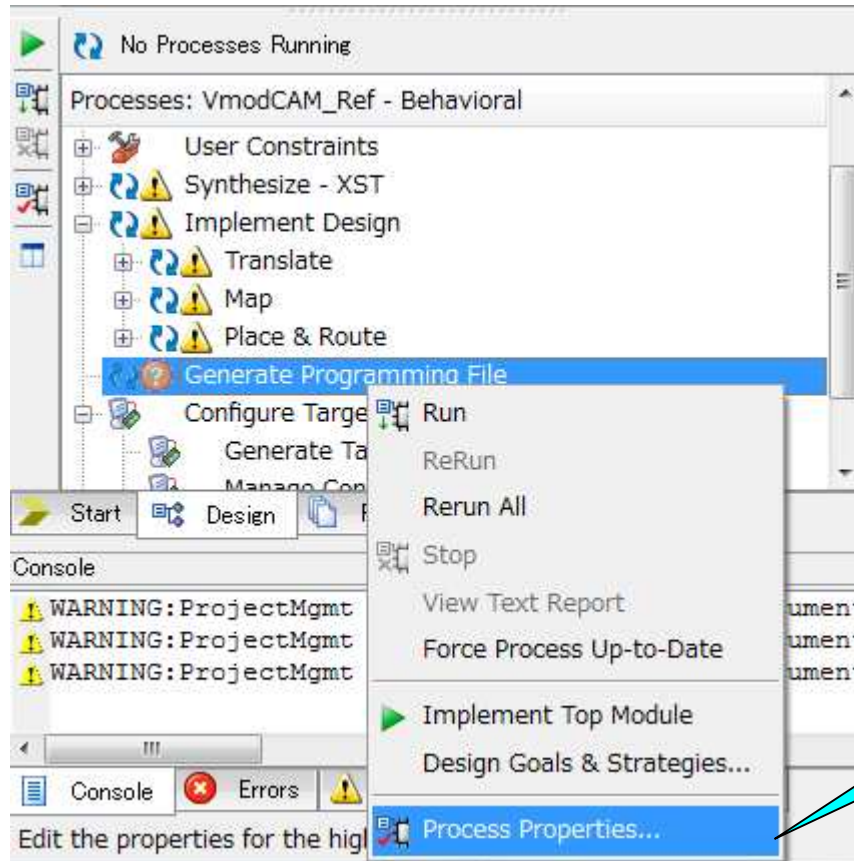
The 'Detailed Reports' section at the bottom right shows:

Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	火 5 29 09:17:12	0	286 Warnings (0 new)	160 Infos (07)

The console window at the bottom shows several warning messages related to file paths.

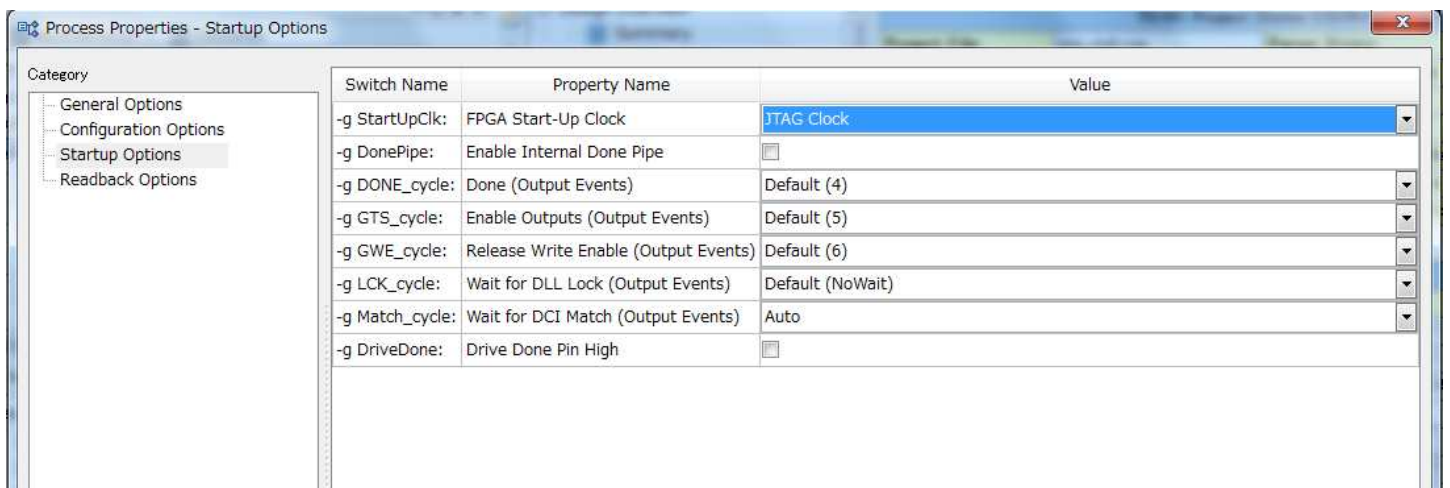
5) プログラムファイルの生成 (FPGAにデータを直接書き込む場合)

1. 配置配線が終了したら、プロセスウィンドウの[Generate Programming File] を選択し、マウスの右クリックで表示されるメニューから[Properties]を選択します。

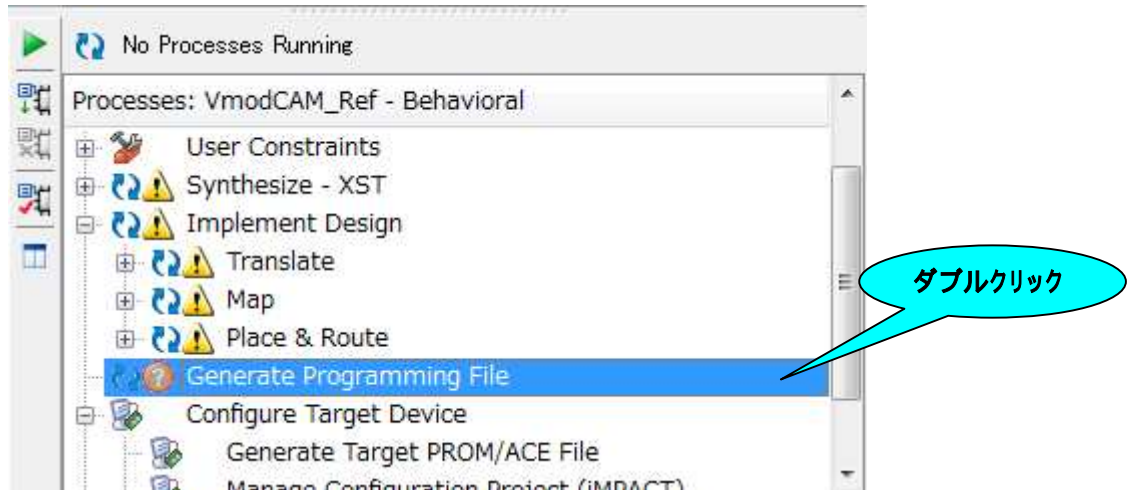


2. Process Properties ウィンドウが表示されます。

そこでカテゴリー(Category)の[Startup options]を選び、[FPGA Start-Up Clock]の項目を[JTAG Clock]にプルダウンから変更し、[OK]をクリックします。



3. プロパティの設定後、プロセスウィンドウの[Generate Programming File]をダブルクリックします。
プログラムファイル作成が実行されます。

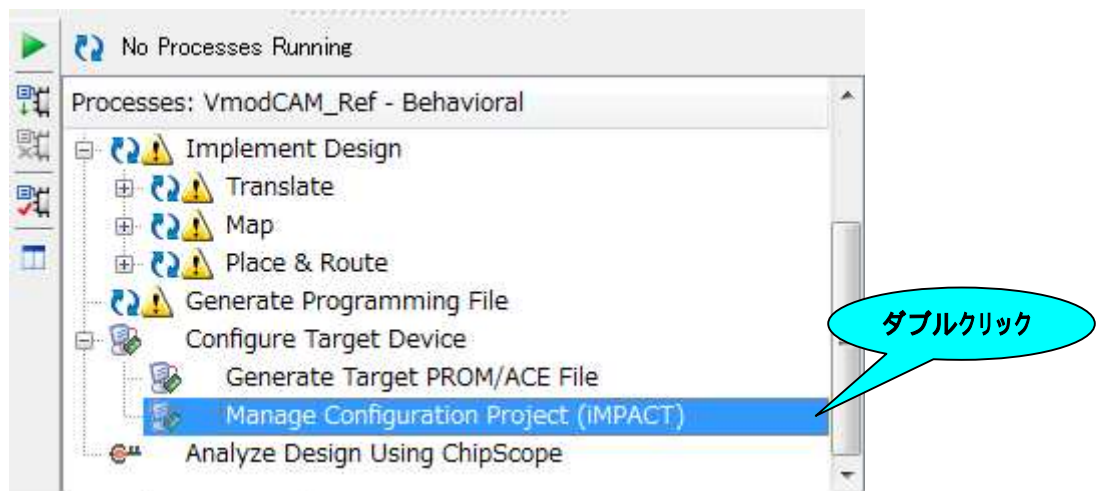


4. コンソールウィンドウに「Processes "Generate Programming File" complete successfully」というメッセージが出ると、プログラムファイル(bit ファイル)がプロジェクト内に自動作成されます。

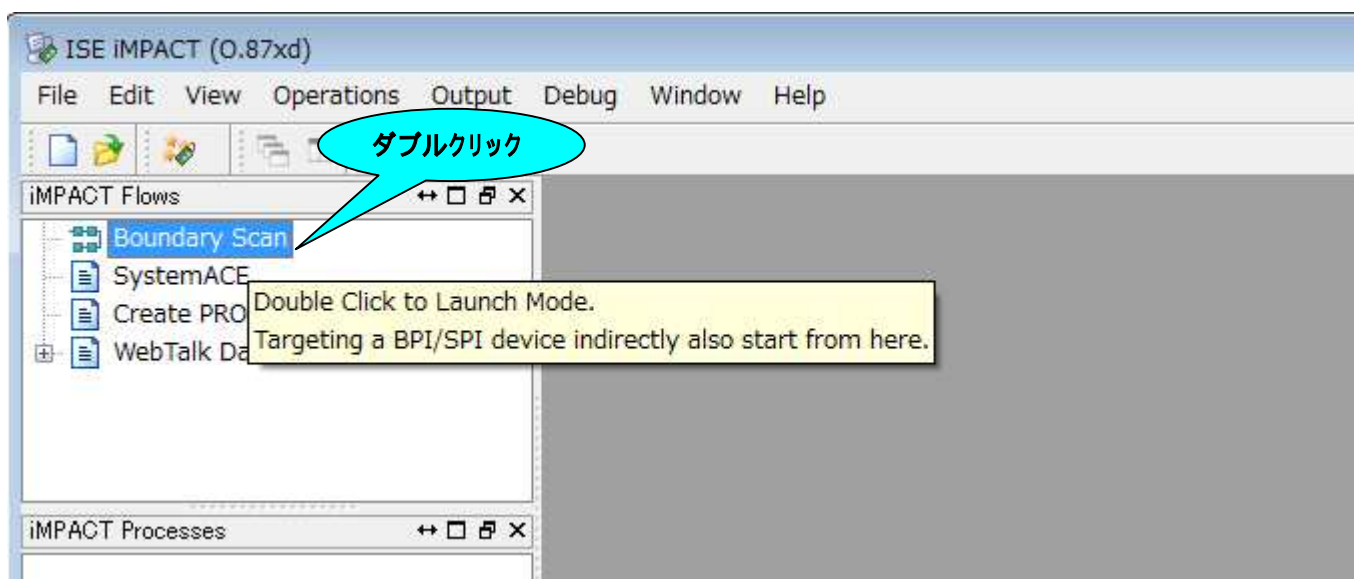
6) デバイスへの書き込み (FPGAにデータを直接書き込む場合)

ボード上の SPI-Flash にデータを書き込んで FPGA を動作させる場合は、12ページからを参照して下さい。
デバイスに書き込みを行なう前に、USB ケーブルを接続して、本体機器の電源を入れて下さい。

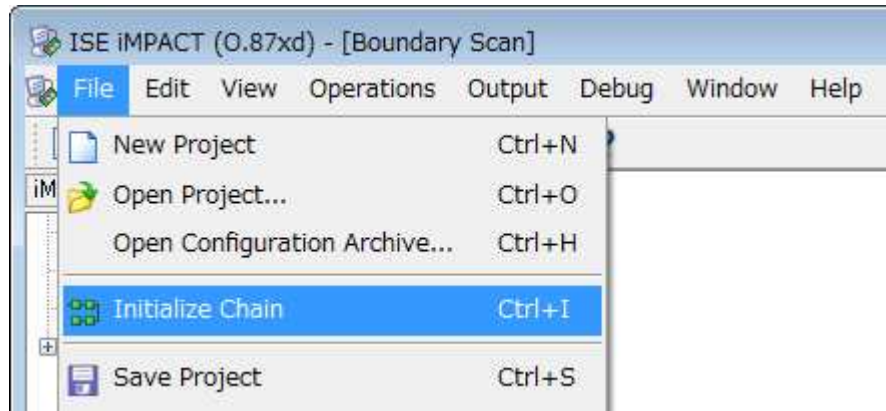
1. プロセスウィンドウの[Configure Target Device]を展開し、[Manage Configuration Project(iMPACT)] をダブルクリックするとダウンロードツールの iMPACT が起動します。



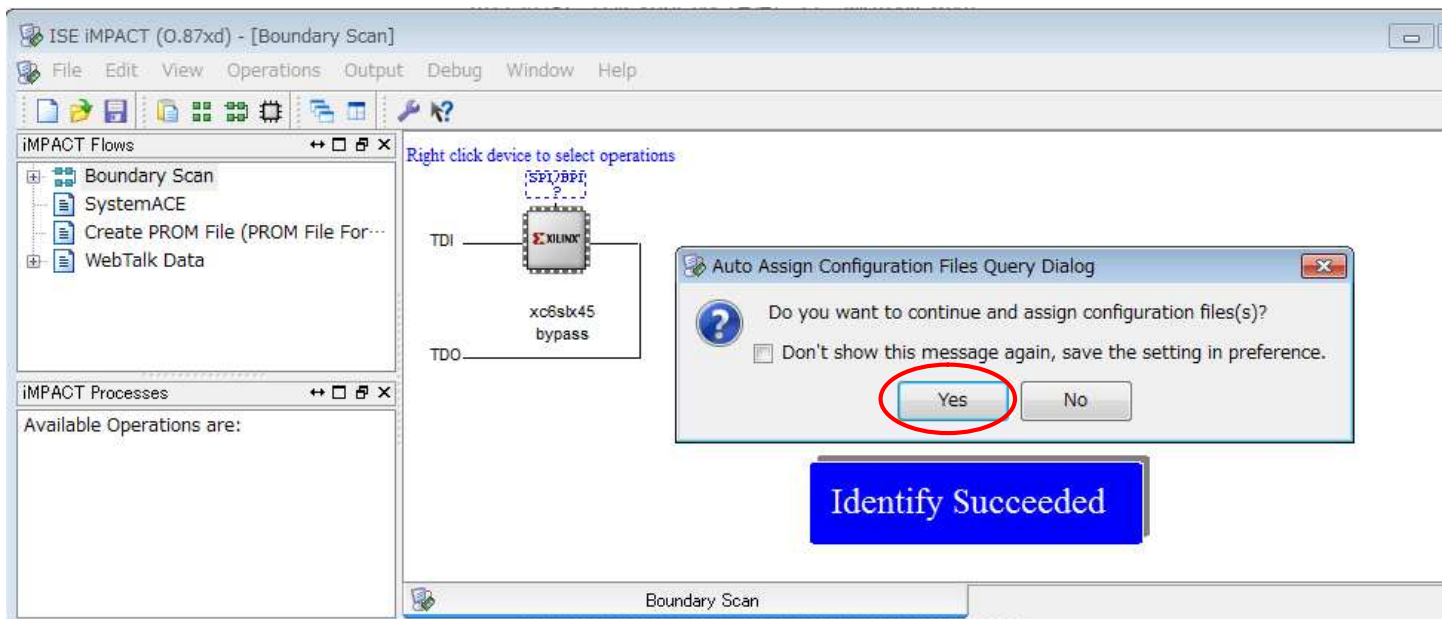
2. iMPACT 起動後に、[iMPACT Flows]内の[Boundary Scan]をダブルクリックします。



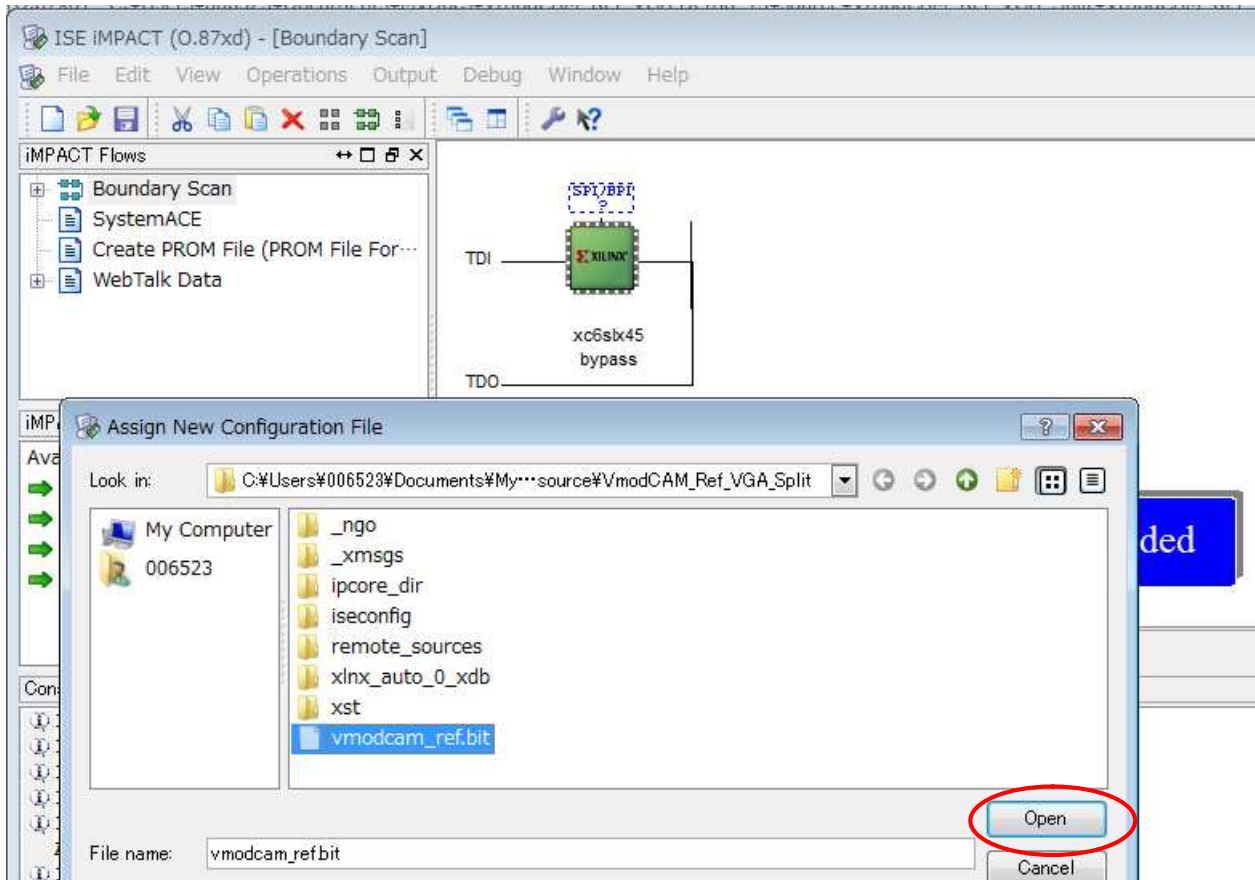
- 次に[Initialize Chain]を実行し、JTAG チェーン上のデバイス(PROM と FPGA)を認識させます。
デバイスは自動で認識されます。



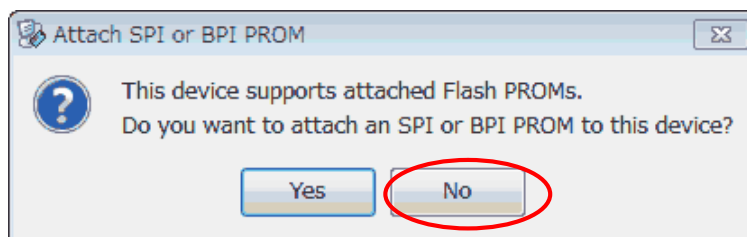
- [YES]を押してから、コンフィグレーションファイルの選択を開始します。



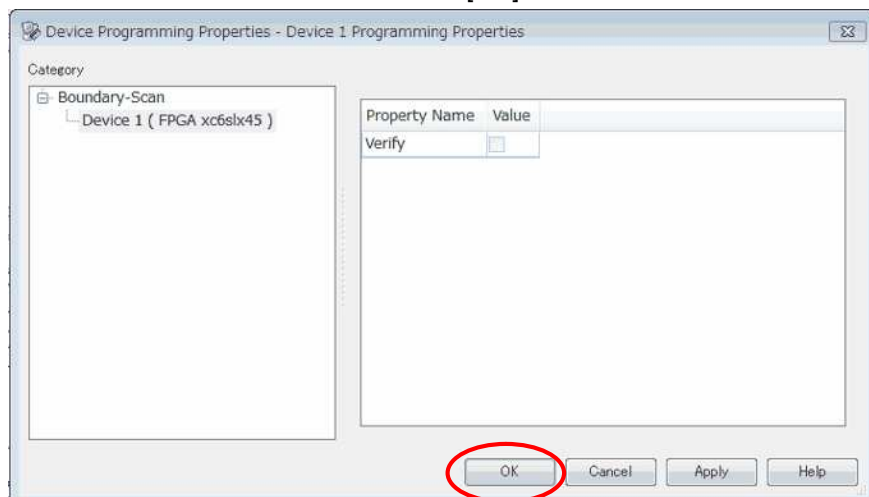
FPGA が選択されておりますので、7 ページで生成した vmodcam_ref.bit ファイルを指定して、[Open]ボタンを押します。



以下の画面は自動で出ますが、今回は PROM は現時点では追加しないため、NO ボタンを押します。

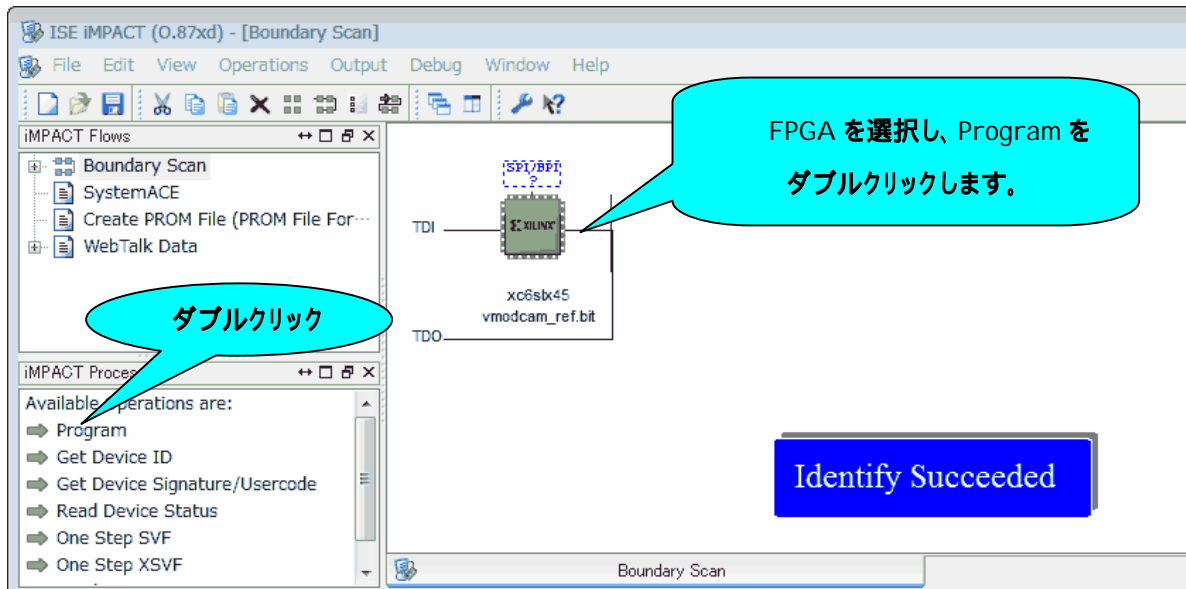


プロパティの設定画面は、OK ボタンを押します。次の画面も[OK]ボタンを押します。

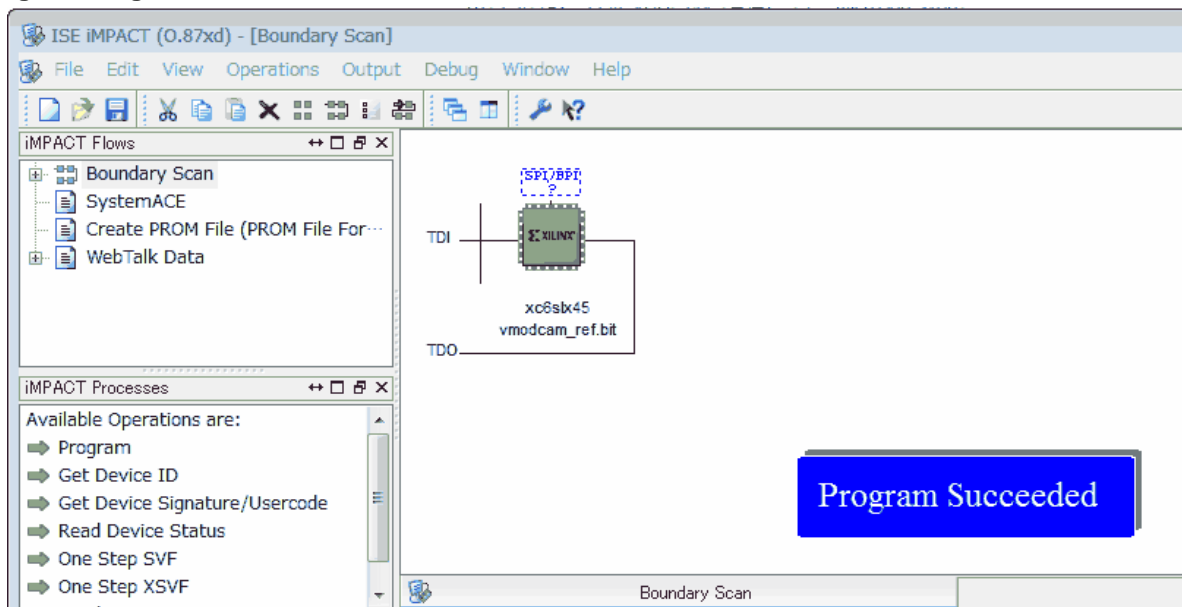


5. デバイスのダウンロードの実行

デバイスの絵の部分をクリックし、グレーからグリーンに変わると、デバイスへのダウンロードが可能となります。



6. ダウンロードが開始され下図のように、**Programming Succeeded** のメッセージが表示されたらダウンロード完了ですが、**Programming Failed** の場合はエラーの為、エラーメッセージを確認し、再実行します。



6-1) デバイスへの書き込み (SPI-Flashにデータを書き込む場合)

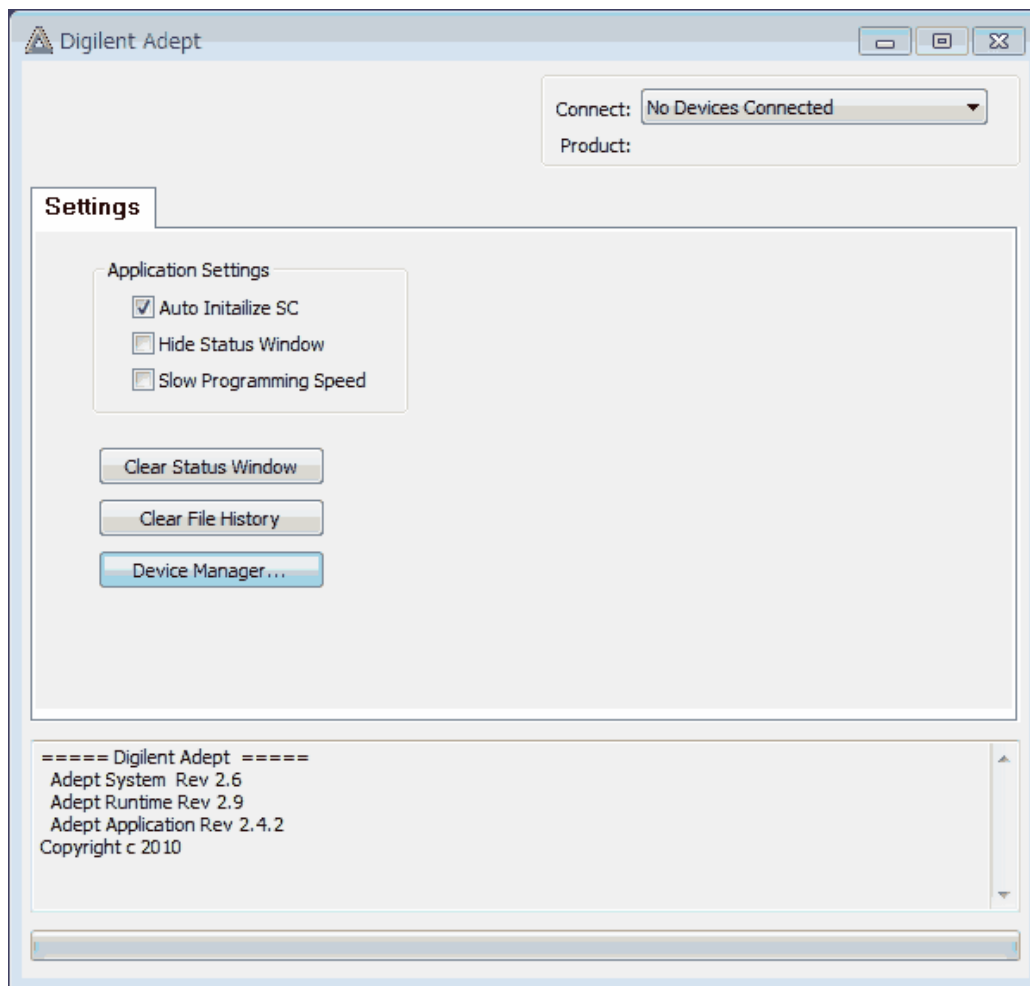
SPI-Flash データを書き込む場合は、ISE ツールではなく Digilent 社専用の書き込みツール[Adept]を使用します。Adept ツールは、以下の WEB サイトよりダウンロード可能です。

URL : <http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,66,828&Prod=ADEPT2>

Adept ツールを使用するのは、ボード上の SPI-Flash のデバイス特有の ID が ISE の iMPACT で読み込めないからです。

デバイスに書き込みを行なう前に、USB ケーブルを接続して、本体機器の電源を入れて下さい。

1. Windows のスタートメニューから、[スタート] - [プログラム] - [Digilent] - [Adept] を選択します。

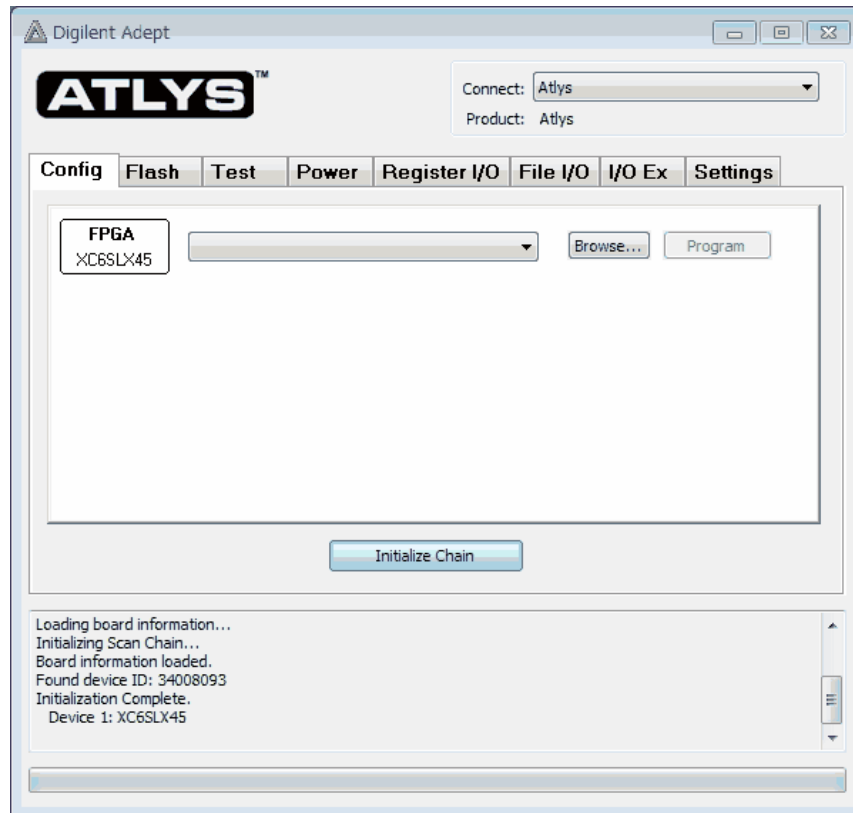


【One-Point】

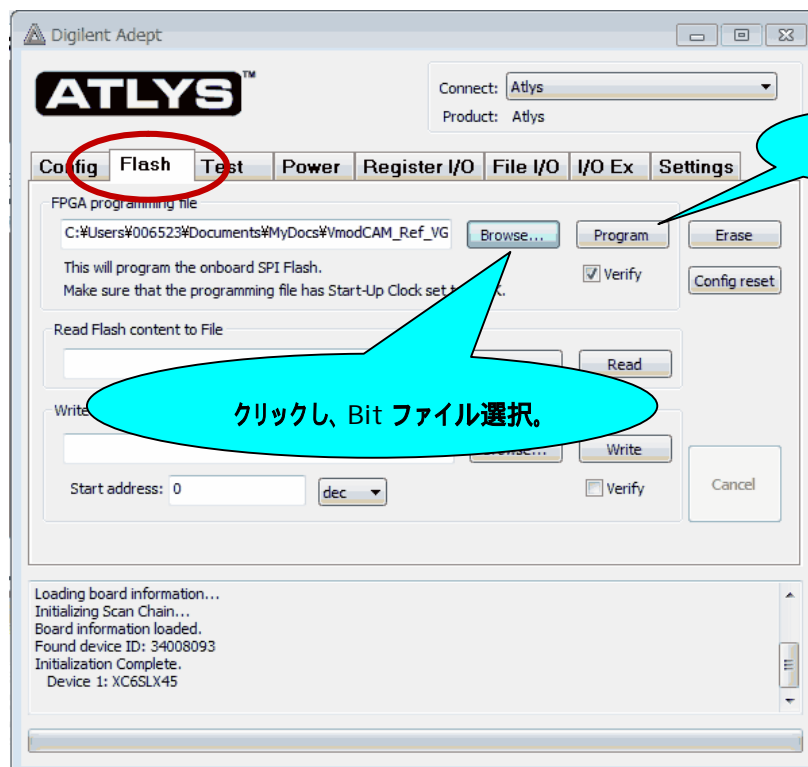
上記の画面は初めて Adept を立ち上げた場合の画面です。
ボードを接続すると自動的に認識しますが、接続しない状態ですと上記のように Connect に「No Device Connected」と表示されます。

2. ATLYS ボードを接続します。

Adept ツールは、ATLYS ボードを自動認識し、ATLYS 上の FPGA(XC6SLX45)を表示します。



Flash タブをクリックし、FPGA programming file の Brows... ボタンをクリックし、7 ページ生成した bit ファイルを選択します。Verify のチェックボックスをチェックし、Program ボタンをクリックするとコンフィグレーションが完了します。



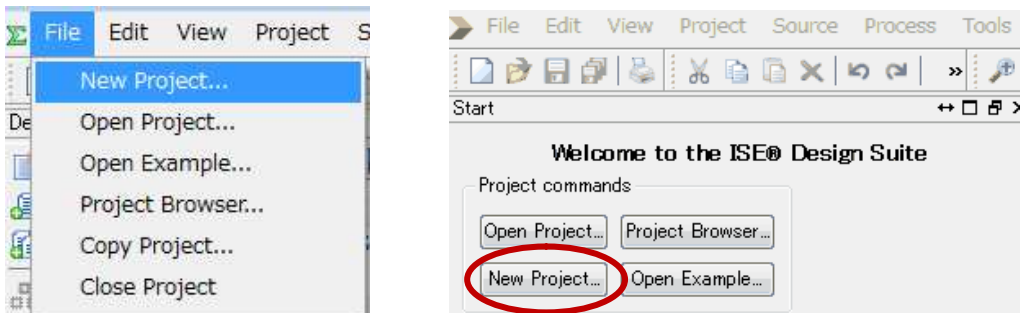
本ページからは、リファレンスデザインを使用せずに新規にプロジェクトを作成するなど、他の ISE ツールの操作手順のご紹介になります。

◆ 新規プロジェクトの作成

まず、Project Navigator を起動します。(3ページ参照)

1. [File] - [New Project] を選択します。

もしくは、画面上の [New Project] のボタンをクリックします。

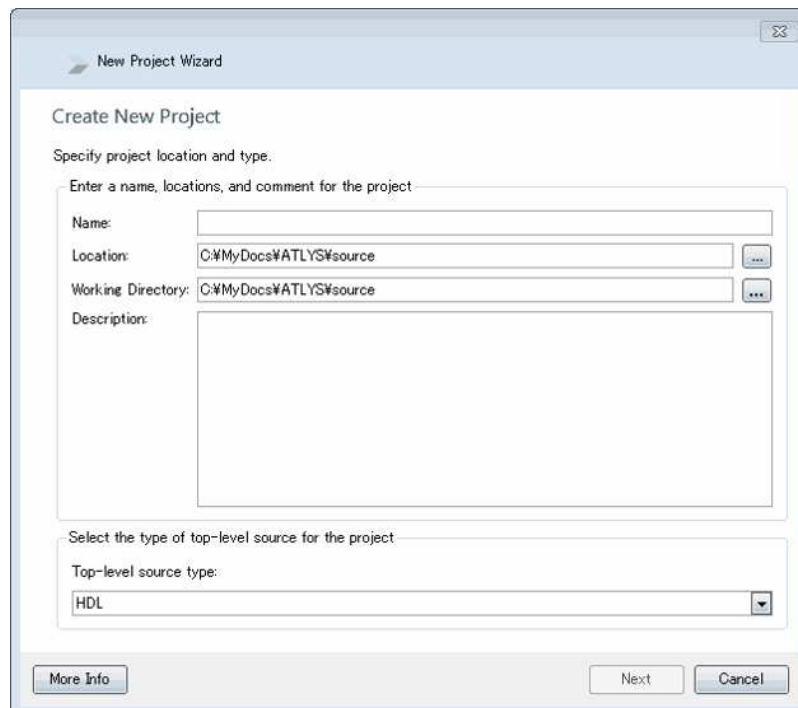


2. Name / Location / Working Directory / Top-level source type を設定し[Next]をクリックします。

Name : [任意に指定]

Location / Working Directory : [任意に指定]

Top-level source type : [HDL]



[One-Point]

通常は、Location / Working Directory は同じ場所を指定します。

また、日本語の入った階層は指定しないで下さい。エラーの原因となります。

3. ターゲットデバイスの情報とデザインフロー等を設定します。

Family	Spartan6
Device	XC6SLX45
Package	CSG324
Speed	-3
Top-Level Module Type	HDL
Synthesis Tool	XST(VHDL/Verilog)
Simulator	ISim(VHDL/Verilog)
Preferred Language	VHDL

Project Settings

Property Name	Value
Top-Level Source Type	HDL
Evaluation Development Board	None Specified
Product Category	All
Family	Spartan6
Device	XC6SLX45
Package	CSG324
Speed	-3
Synthesis Tool	XST (VHDL/Verilog)
Simulator	ISim (VHDL/Verilog)
Preferred Language	VHDL
Property Specification in Project File	Store all values
Manual Compile Order	<input type="checkbox"/>
VHDL Source Analysis Standard	VHDL-93
Enable Message Filtering	<input type="checkbox"/>

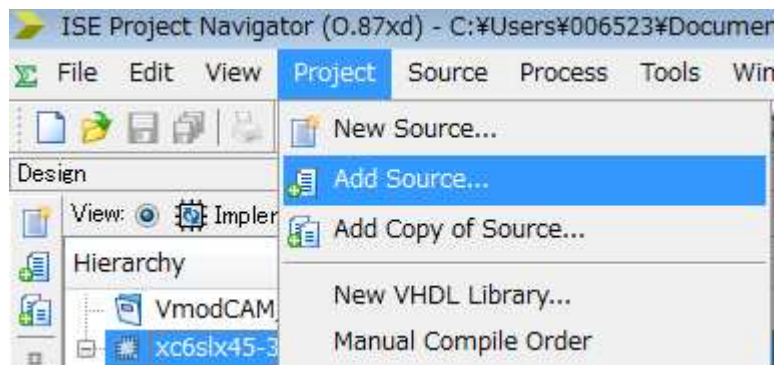
4. 設定後、順次[Next]ボタンをクリックし画面を先へ進め、最終画面で[Finish]をクリックします。これでプロジェクトは完成です。

◆ ソースファイルの読み込み

既存の HDL ソースファイルをお持ちであれば、各ファイルをツールに読み込ませます。

(ソースファイル・テストベンチファイル・制約ファイル)

1. [Project] - [Add Source] 又は [Project] - [Add Copy of Source] を選択し、ファイルを読み込みます。



[One-Point]

ここでもファイルは、日本語のパスがないフォルダに保存して下さい。

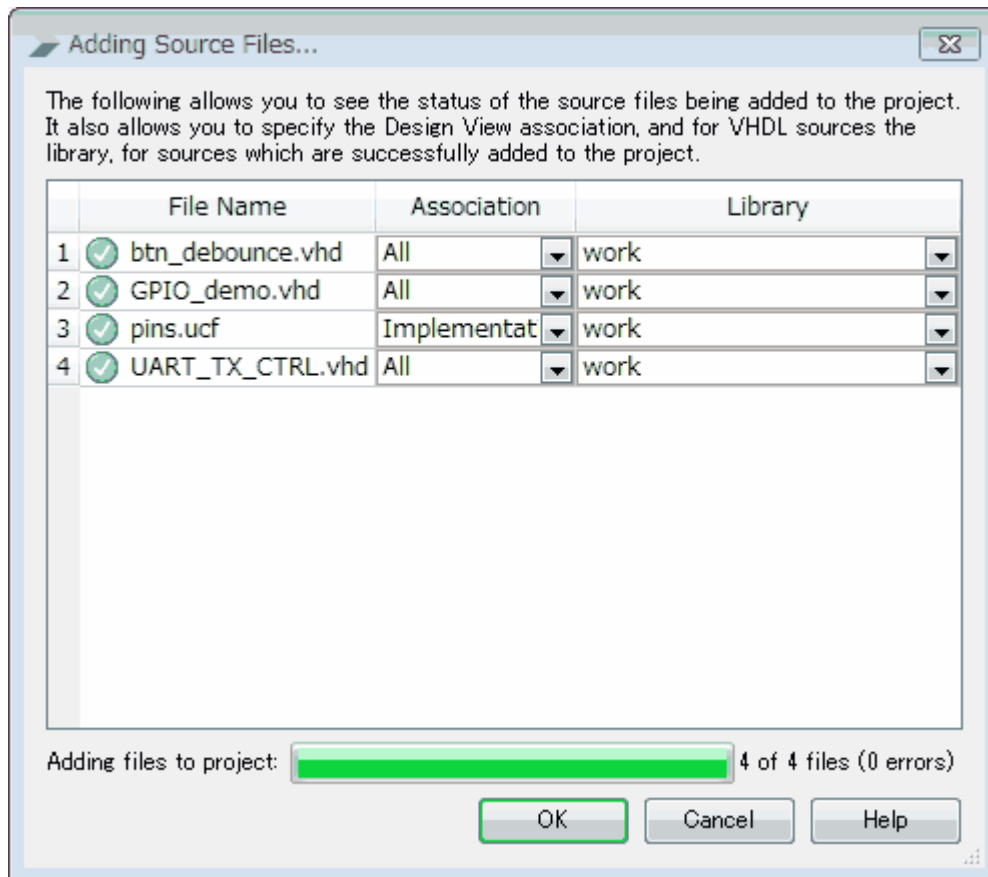
[Add Source]の場合は、読み込むファイルにリンクした状態でウィンドウに追加されます。

変更内容が元のファイルに反映されます。

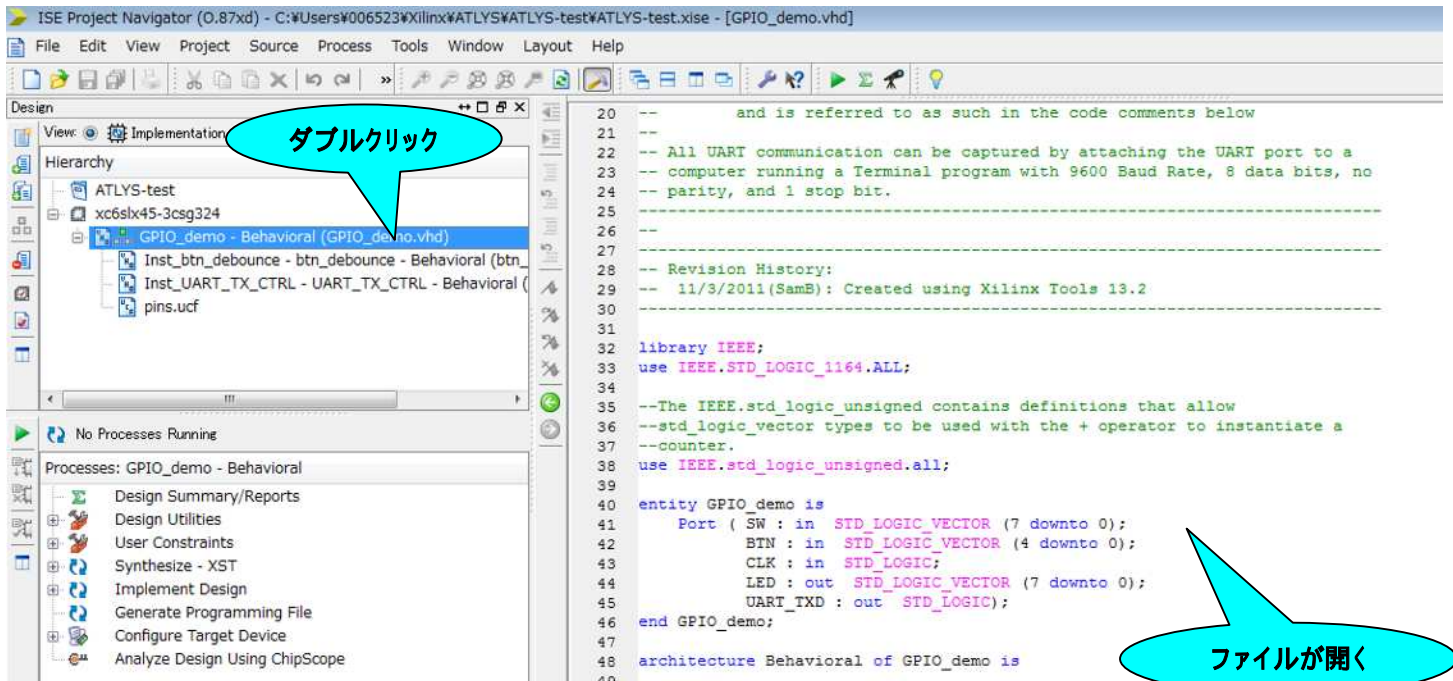
[Add Copy of Source]の場合は、読み込むファイルをプロジェクトのフォルダにコピーされますので、元のファイルとは別扱いになります。

2. 任意の保存先からファイルを読み込みます。(ソース/テストベンチ/制約の3種類のファイルがあります。)
ファイル読み込み時には、ソースタイプを指定します。

下記でソースタイプの指定を行います。



3. 読み込んだファイルを開く時は、ソースウィンドウで開きたいファイルをダブルクリックするとエディタウィンドウが開きます。この状態で、RTL ファイルの変更も可能です。



[One-Point]

この状態でプロジェクトファイルの作成は終了です。

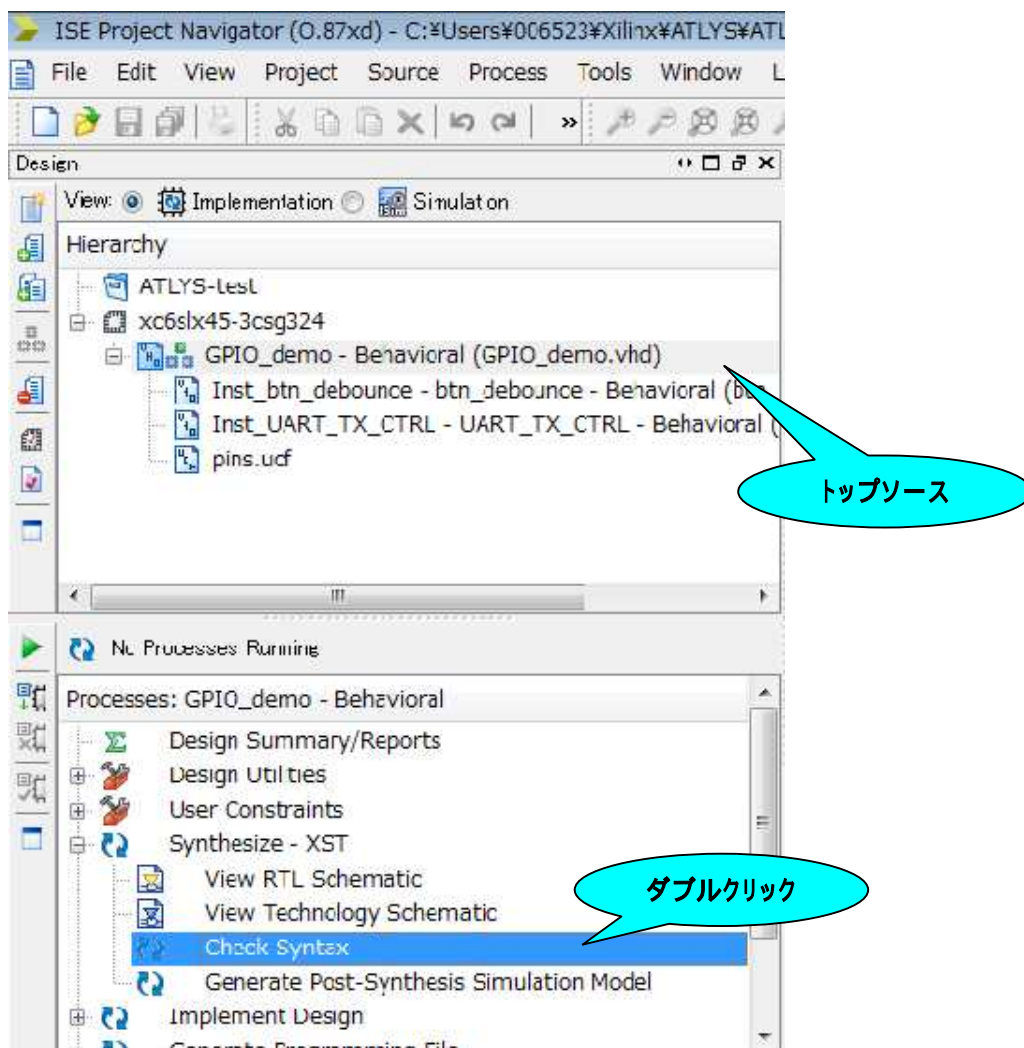
RTL 記述の変更、文法チェック等を行い、ファンクションシミュレーションを行い、論理検証を行います。

また、画面左上の Window には、本デザインの階層構造が示されています。

本デザインは、GPIO デモモジュールを TOP にして、下層に 3 個のモジュールを持つ構造になっています。階層最後に表示されている pins.ucf ファイルは、FPGA に様々な制約を加える制約ファイルになっています。

◆ ソースファイルの文法チェック

1. 文法チェック(Check Syntax)を行う前に、Simulation から Implementation ボタンにチェックがあることを確認して下さい。
チェックを行いたいソースファイルがトップソースの場合、プロセスウィンドウの [Synthesize-XST] - [Check Syntax] をダブルクリックします。
チェックを行いたいソースファイルが下位階層の場合、ソースを選択し、[Check Syntax]をダブルクリックします。
⚠ や ✅ が表示されたら、文法チェック完了です。⚠ が表示された場合必要に応じてソースファイルを修正して下さい。エラーがある場合、❌ が表示されますのでメッセージに従ってソースファイルを修正して下さい。



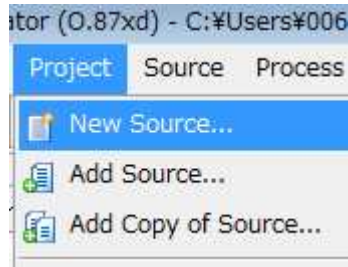
【One-Point】

エラーがある場合は、次のステップに進むことが出来ません。

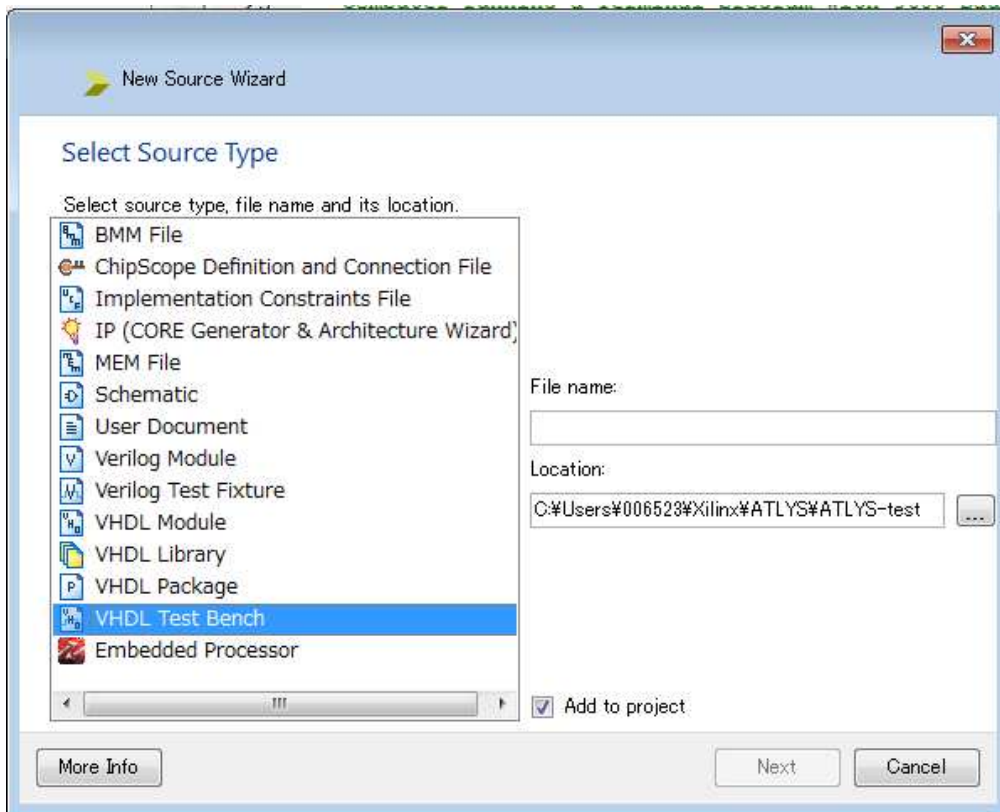
ツールの画面下部の Console に表示されるメッセージを参照して、文法違反部分を修正して下さい。

◆ 新規ファイル(テストベンチファイル:VHDL Test Bench)の作成

1. [Project] - [New Source] を選択します。



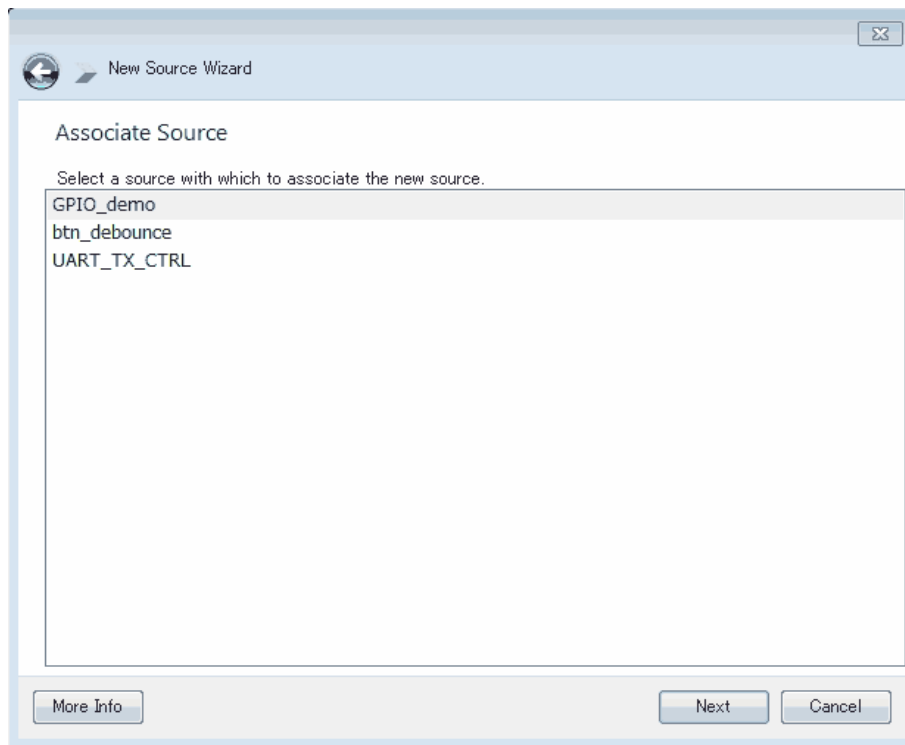
2. **VHDL Test Bench** を選択し、File Name、Location を入力し、[Next]をクリックします。
Project Navigator では、拡張子「.vhd」として認識されます。



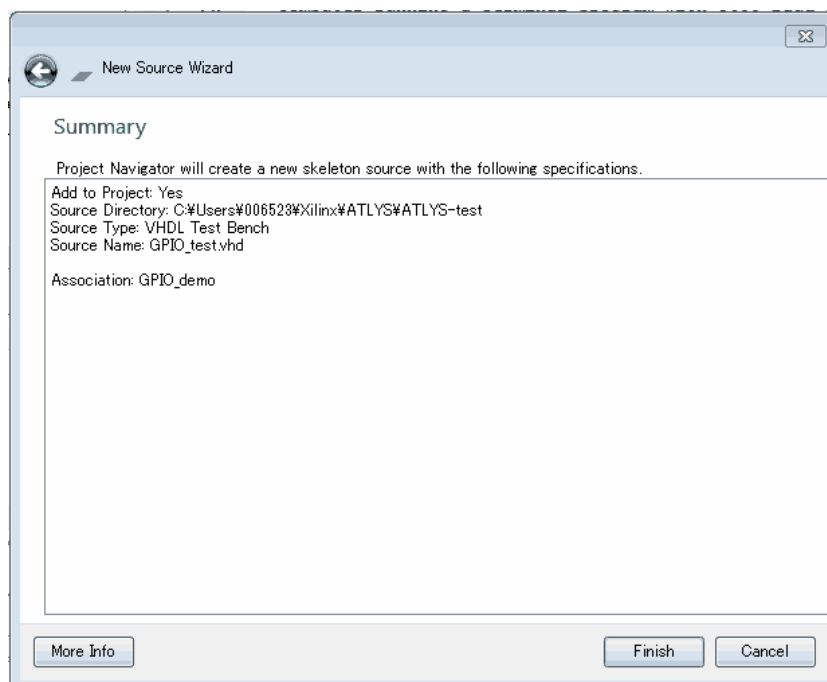
【One-Point】

Location ですが、デフォルト設定では、Work ディレクトリが設定されています。
Work ディレクトリ内は、様々なファイルが存在しますので、別ディレクトリを指定した方が
その後のファイル管理が楽になります。

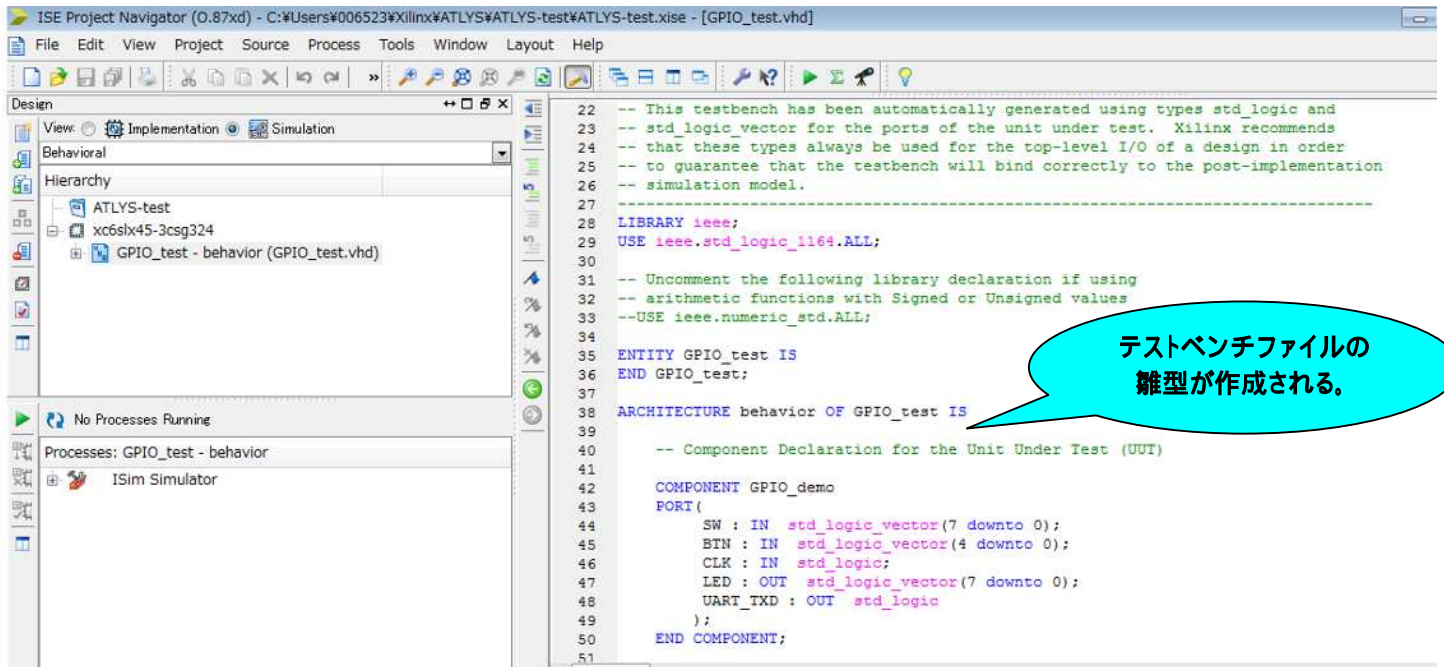
3. 関連付けるソースファイルを選択し、[Next]をクリックします。
本デザインでは、GPIO_demo が TOP モジュールになります。



4. 確認画面で設定内容を確認し、[Finish]をクリックします。

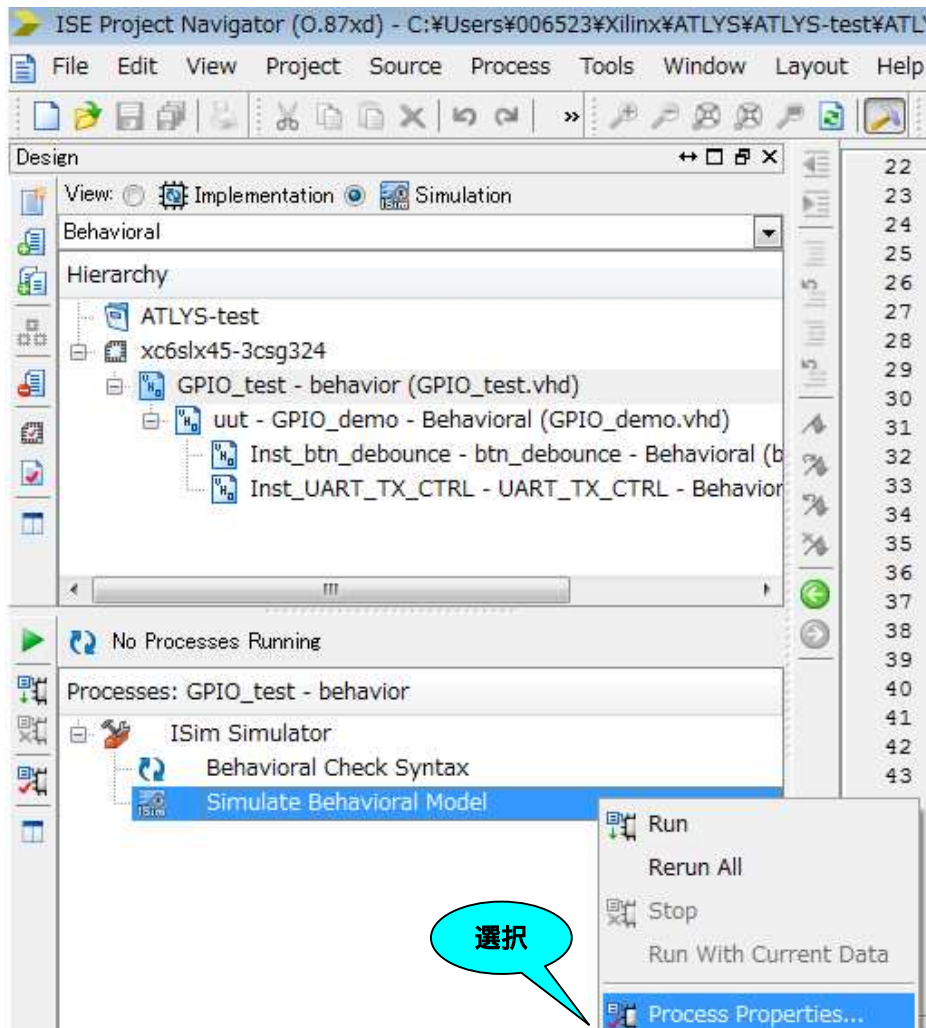


Project Navigator のソースウィンドウにテストベンチファイルが追加され、また、エディタウィンドウにテストベンチの雛型が表示されますので、これを元にテストベンチを完成させます。



◆ ファンクションシミュレーションの実行

1. ソースウィンドウでテストベンチ (GPIO_test.vhd) のファイルを選択します。
Implementation から Simulation にボタンチェックを変更します。
2. プロセスウィンドウの[ISim Simulator] - [Simulate Behavioral Model]を選択し、マウスの右クリックで表示されるメニューの中の[Process Properties]を選択します。

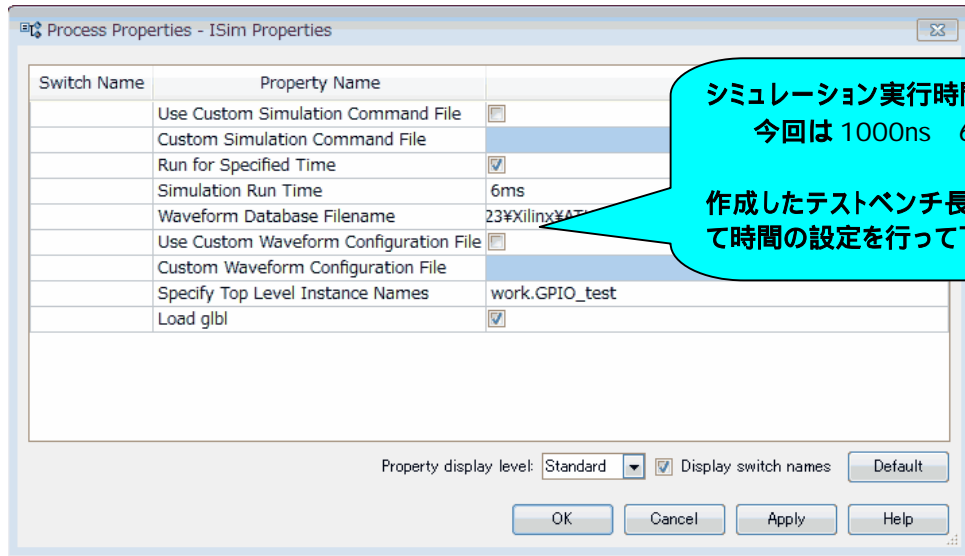


【One-Point】

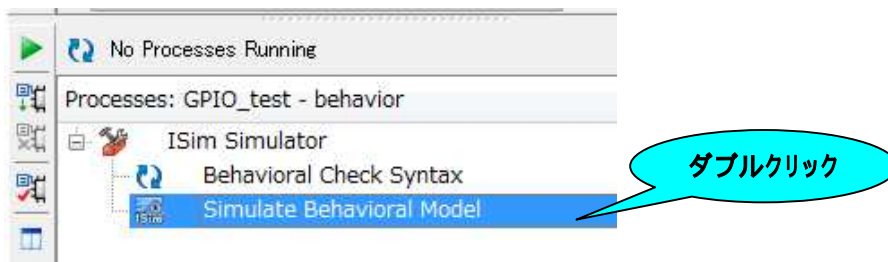
Process Properties は、Synthesize-XST などにも存在します。

様々のオプション設定が可能となっており、デザインに適した設定を行うことで、FPGA の能力を最大限まで発揮させることが可能です。

3. Process Properties でシミュレーション実行のオプションを指定します。

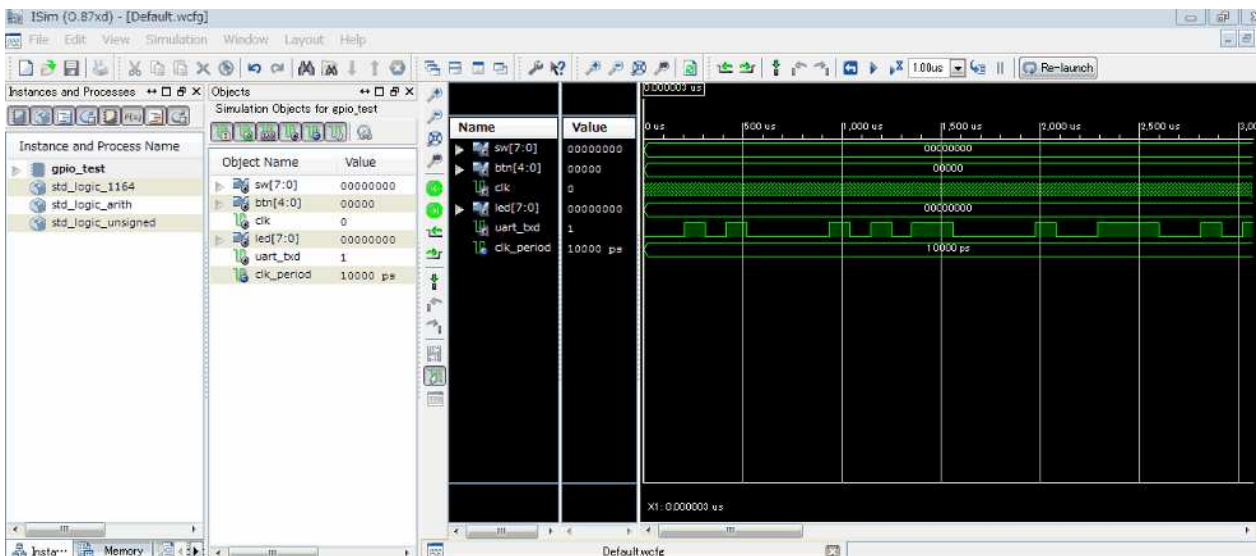


オプションの設定が終了したら、プロセスウィンドウの[ISim Simulator] - [Simulate Behavioral Model]をダブルクリックします。



[One-Point]
 自動的に ISim が立ち上がり、ソースファイルのコンパイル、デザインのロードを行ない、プロパティウィンドウで指定した時間でシミュレーションが実行されます。

4. 正しく動作しているか波形を確認します。



◆ ISim操作ガイド(補足)

詳細は、Xilinx 社が提供している ISim ユーザーガイドを参照して下さい。

1. ソースファイルの変更について

以下の手順に従いますと、ソースファイルを安全に変更出来ます。デザインの競合を回避するには、ISim 外で編集を行って下さい。

1. ISE の ISE Text Editor 又はサードパーティのテキストエディタを用いて、ソースファイルを編集します。
2. ISE ツールでデザインを実行して、デザインをアップデートし、シミュレーションを実行します。

2. 信号の追加 と シミュレーションの再実行

信号の追加

「Instances and Processes」パネルに Module が階層構造で表示されます。

追加したい Module を選択するとその Module の信号が「Object パネル」に表示されるので、追加したい信号をドラッグ&ドロップで「波形ウィンドウ」に追加できます。

但し、これだけでは値は表示されませんので シミュレーションの再実行を行って下さい。

シミュレーションの再実行

下図のようにメインメニューの[Restart] アイコンをクリックします。シミュレーション実行時間を設定後に[Run]アイコンをクリックします。

