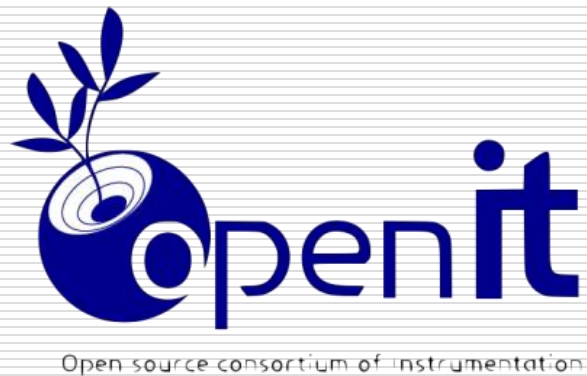


Open-It FPGAトレーニングコース(入門編)

4.3 FPGAへの実装(順序回路)



第3.3版

2016年06月22日

論理合成

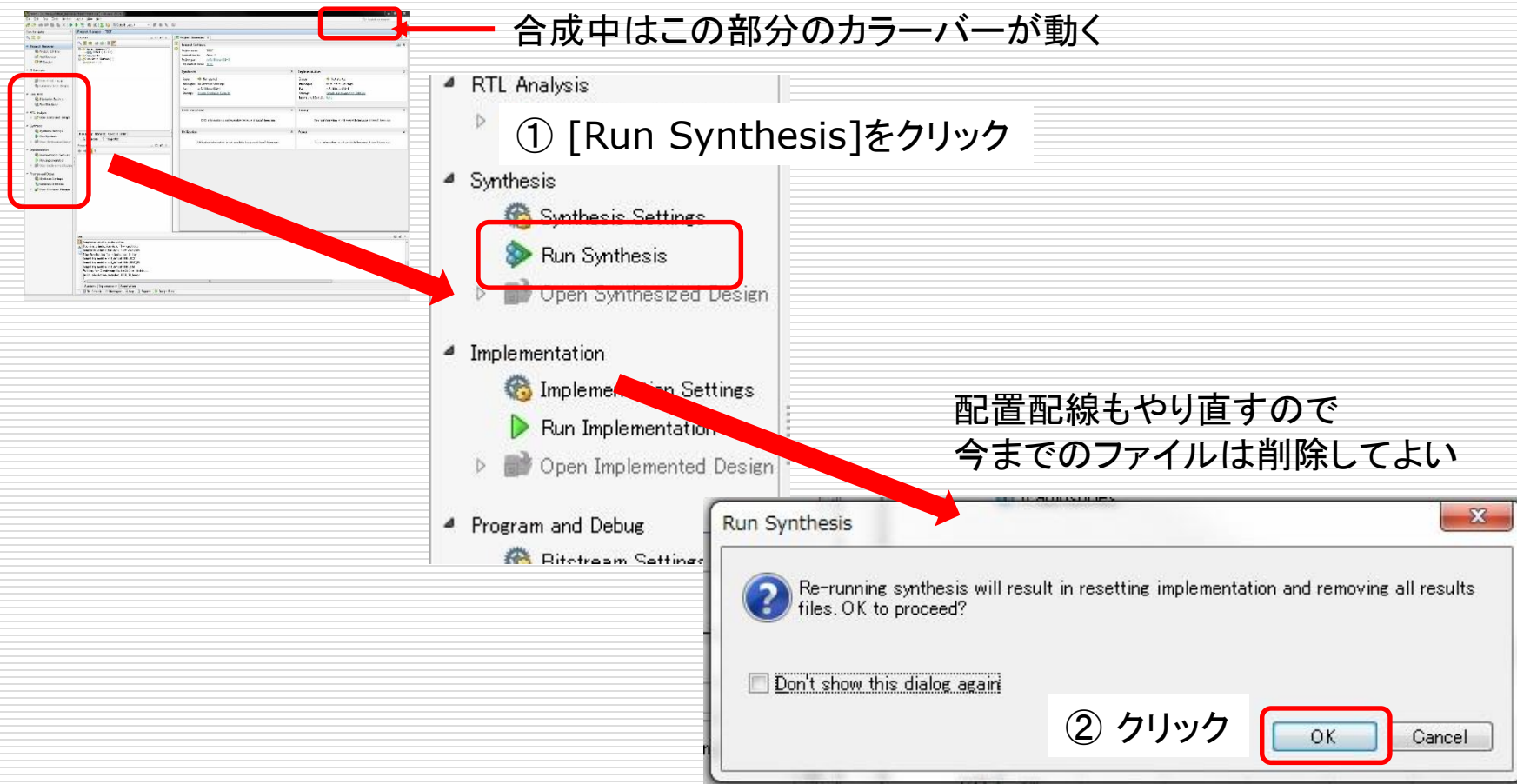
Vivado起動

ソースコードを編集するためにVivadoを起動してください



TESTをクリック

論理合成開始



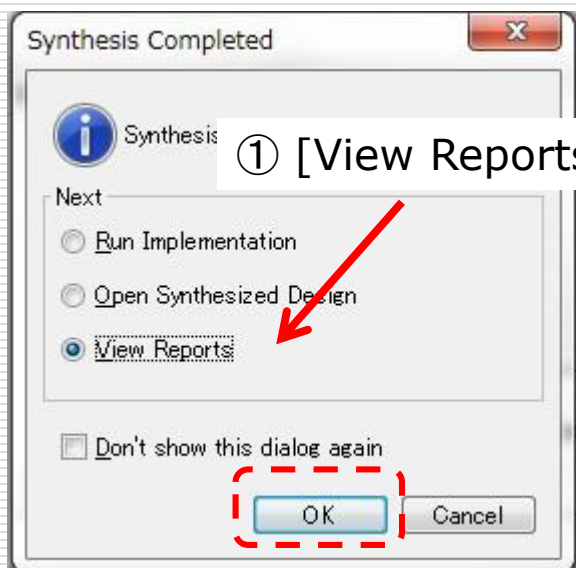
合成中はこの部分のカラーバーが動く

① [Run Synthesis]をクリック

配置配線もやり直すので
今までのファイルは削除してよい

② クリック

論理合成結果の確認



① [View Reports]を選択

②クリック

③エラーが無いことを確認

ここではWarningは問題にせず進めます



Project summaryページのSynthesis項目

ユーザー制約

タイミング制約

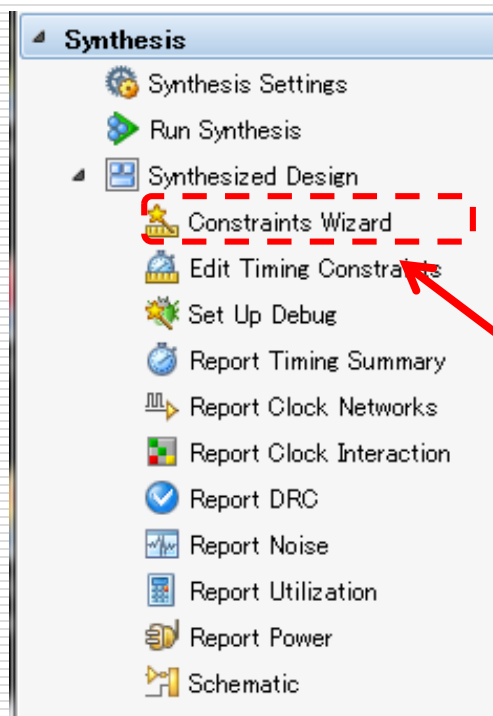
今回は順序回路の配置配線なのでタイミング制約を課しません

□ 動作周波数

- 同期回路でよく使われる制約
- 通常、希望動作周波数がある
- 希望の周波数で回路が動作するかどうかをツールに遅延計算させレポートさせる

Constraints Wizardの起動

昨日はOpen designから画面のレイアウトを切り替えて制約設定しました。
今回はConstraints Wizardを使って入力します

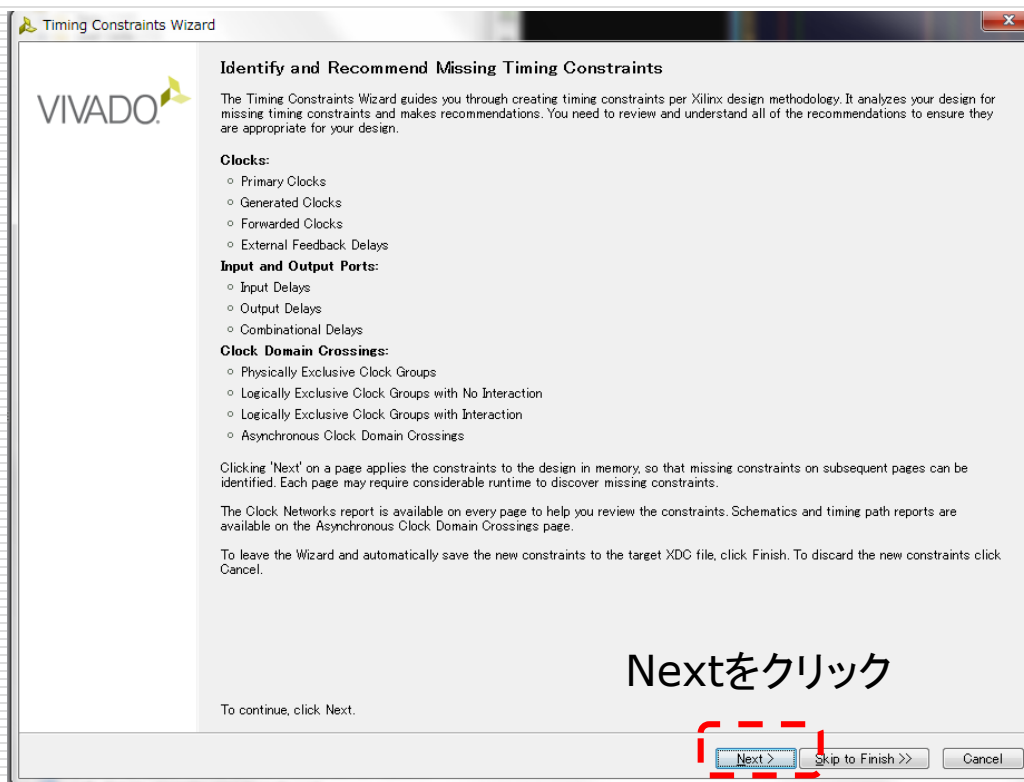


[Constraints Wizard]をクリック

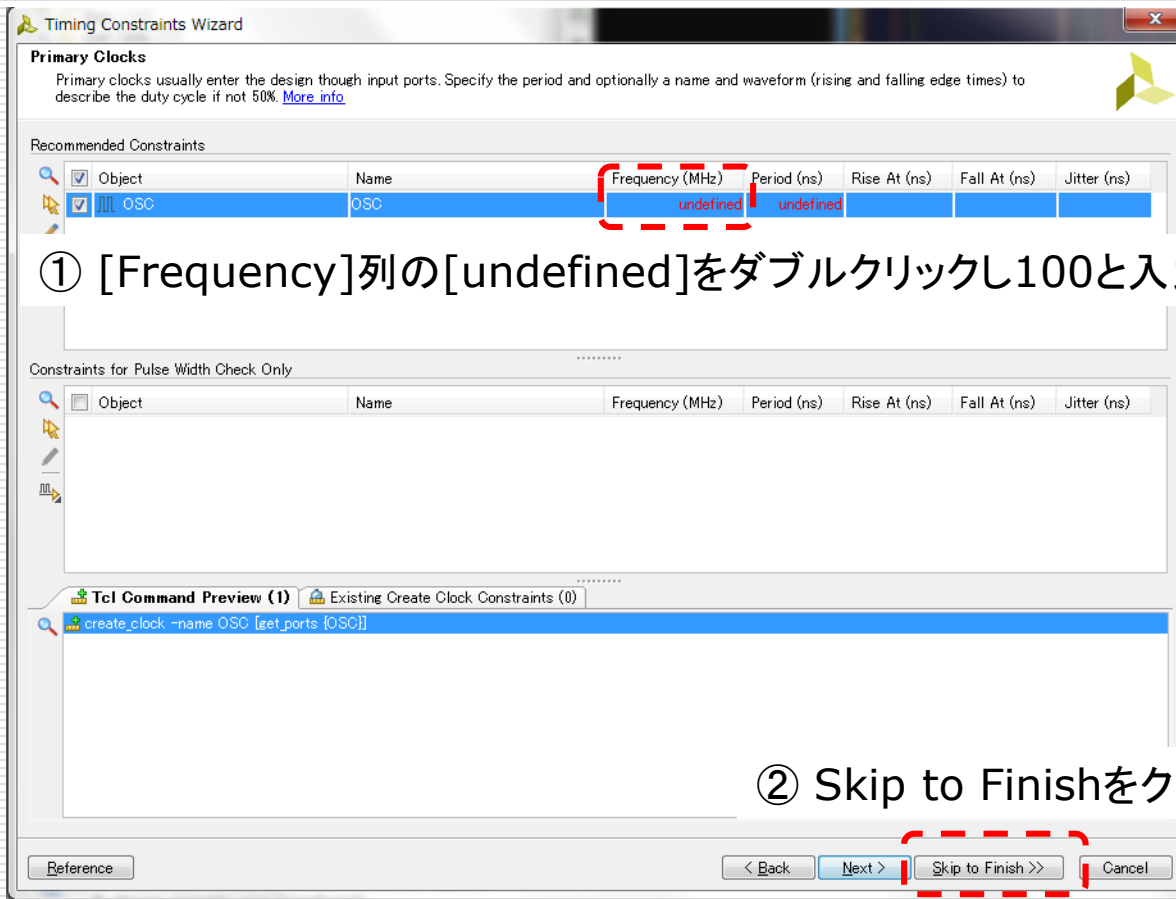
Flow Navigator

Constraints Wizardの初期画面

タイミング制約を促す画面



外部入力クロック



Primary Clocks
Primary clocks usually enter the design through input ports. Specify the period and optionally a name and waveform (rising and falling edge times) to describe the duty cycle if not 50%. [More info](#)

Recommended Constraints

Object	Name	Frequency (MHz)	Period (ns)	Rise At (ns)	Fall At (ns)	Jitter (ns)
<input checked="" type="checkbox"/>	OSC	undefined	undefined			

① [Frequency]列の[undefined]をダブルクリックし100と入力

Constraints for Pulse Width Check Only

Object	Name	Frequency (MHz)	Period (ns)	Rise At (ns)	Fall At (ns)	Jitter (ns)
--------	------	-----------------	-------------	--------------	--------------	-------------

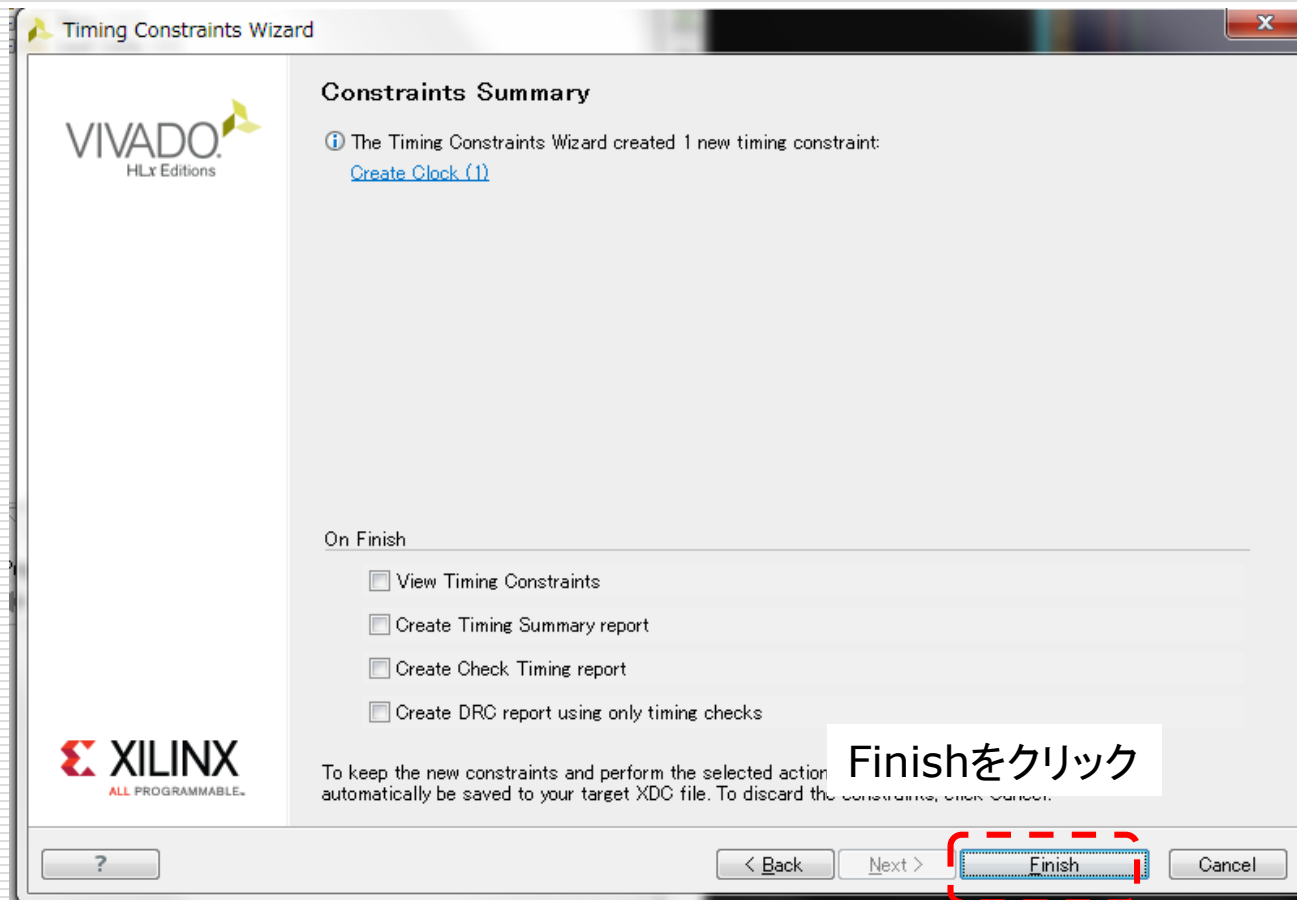
Tcl Command Preview (1) Existing Create Clock Constraints (0)

```
create_clock -name OSC [get_ports {OSC}]
```

② Skip to Finishをクリック

Reference < Back Next > Skip to Finish >> Cancel

確認画面



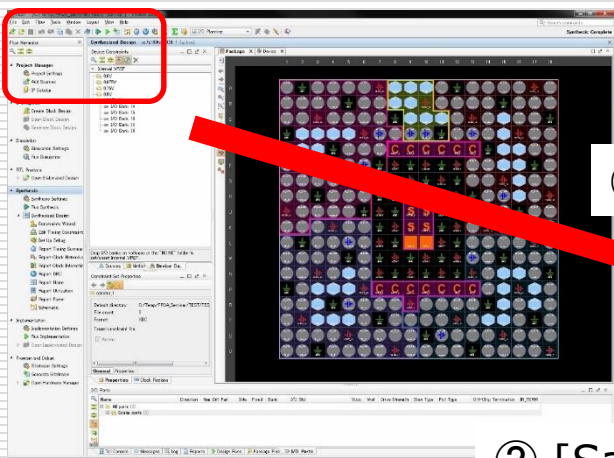
ピンリスト

Name	Package Pin	I/O Std.	Drive Str.	On chip termination	Pull type	Slew
OSC	E3	LVC MOS33		NONE	NONE	
RST_SWn	C12	LVC MOS33		NONE	NONE	
SW_A	U9	LVC MOS33		NONE	NONE	
SW_B	U8	LVC MOS33		NONE	NONE	
LED0	T8	LVC MOS33	12	NONE	NONE	Slow
LED1	V9	LVC MOS33	12	NONE	NONE	Slow
LED2	R8	LVC MOS33	12	NONE	NONE	Slow
LED15	P2	LVC MOS33	12	NONE	NONE	Slow

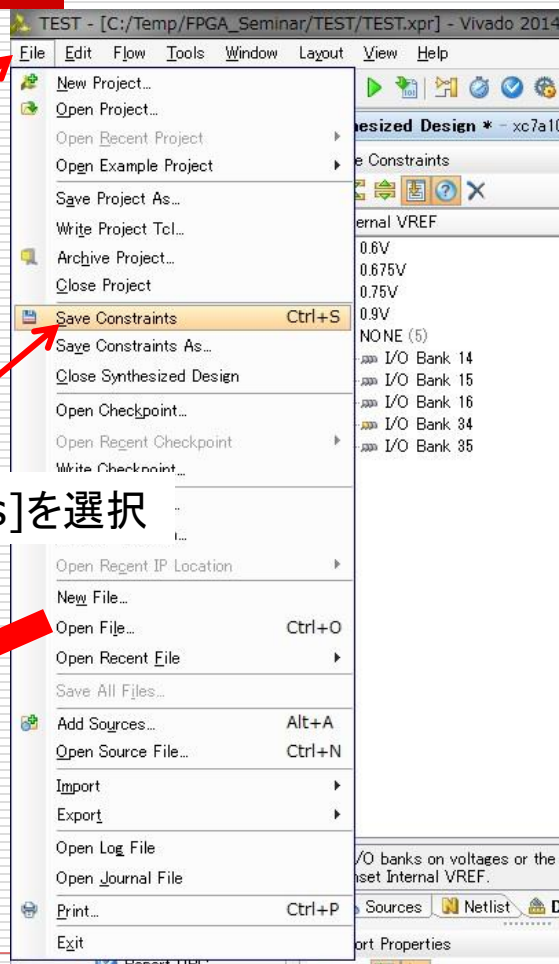
使用している信号のみ設定してください

空欄は設定(選択)不要です

制約ファイルの保存



① [File]から

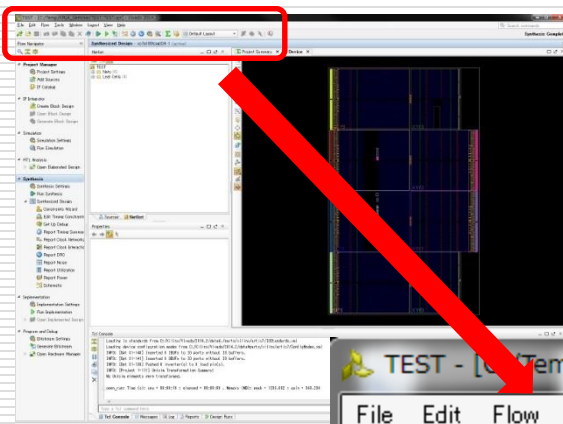


② [Save Constraints]を選択

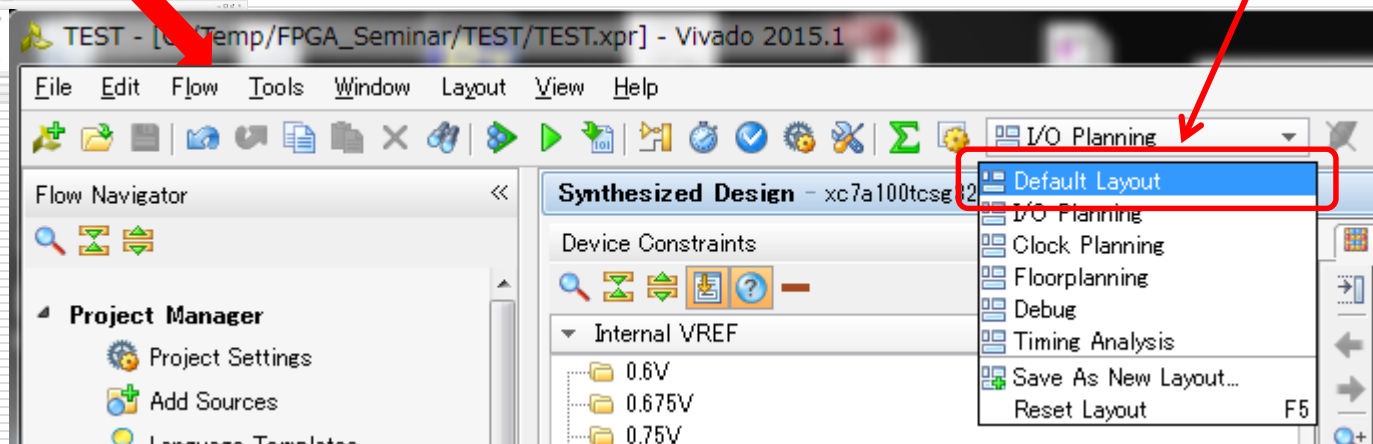


② クリック

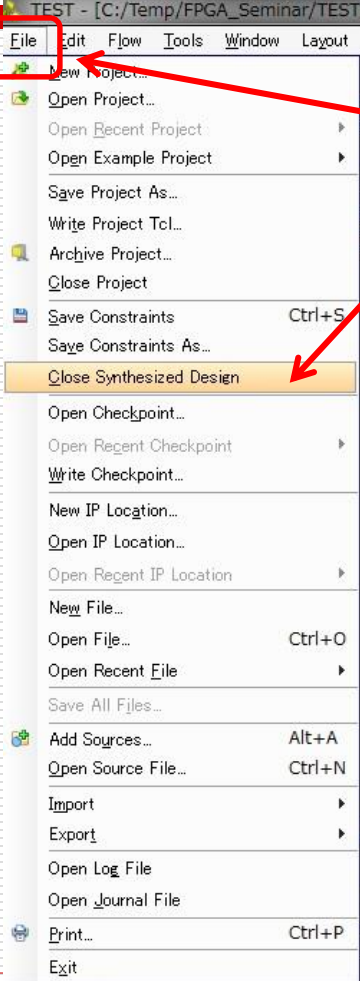
画面レイアウトをDefaultへ戻す



プルダウンメニューから
Default layoutを選択

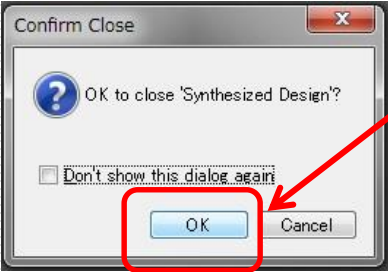


Synthesized Designを閉じる

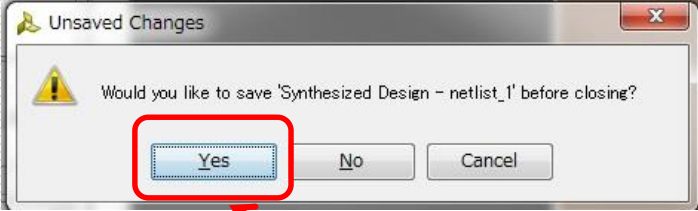


① Fileメニュー

② Close Synthesized Designを選択



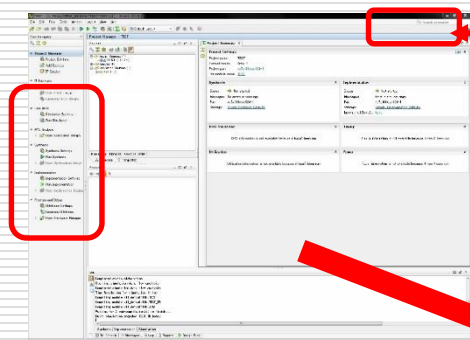
③ OKをクリック



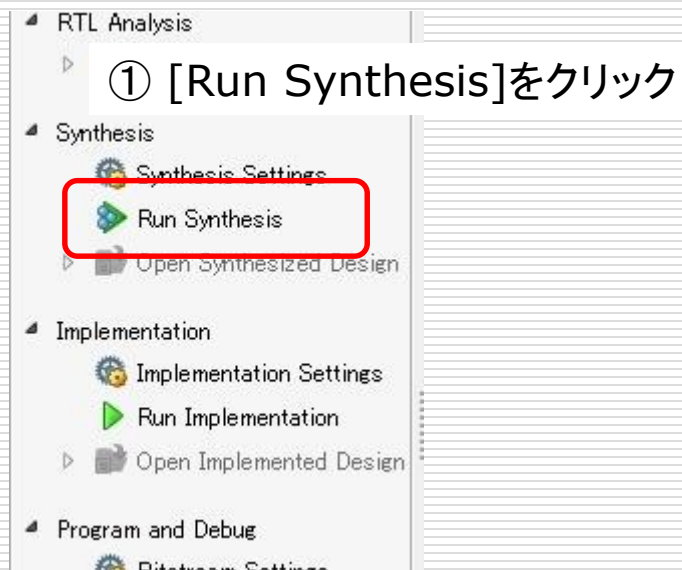
④ Yesをクリック

The image shows a sequence of steps to close a synthesized design. It starts with the File menu open, where 'Close Synthesized Design' is selected. This leads to a 'Confirm Close' dialog box where 'OK' is clicked. Finally, an 'Unsaved Changes' dialog box appears, and 'Yes' is clicked to save the design before closing.

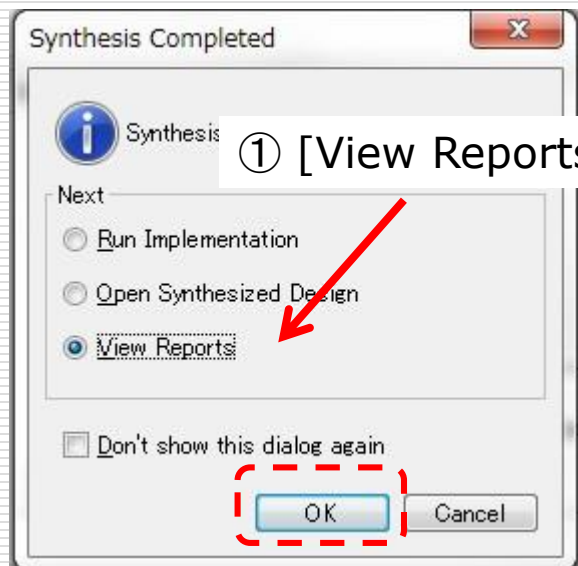
再度、論理合成



合成中はこの部分のカラーバーが動く



論理合成結果の確認

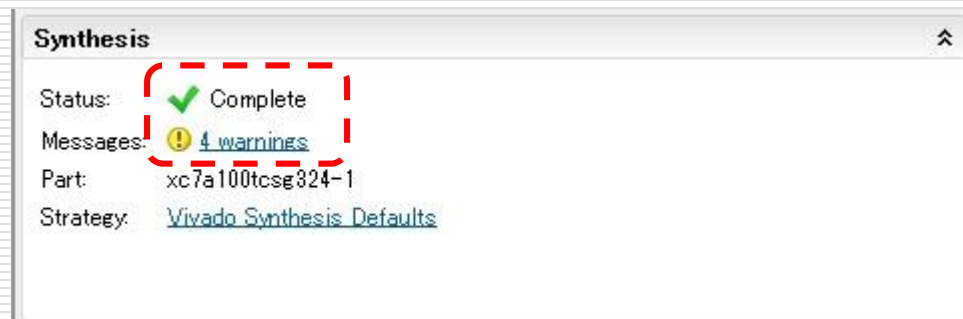


① [View Reports]を選択

②クリック

③エラーが無いことを確認

ここではWarningは問題にせず進めます

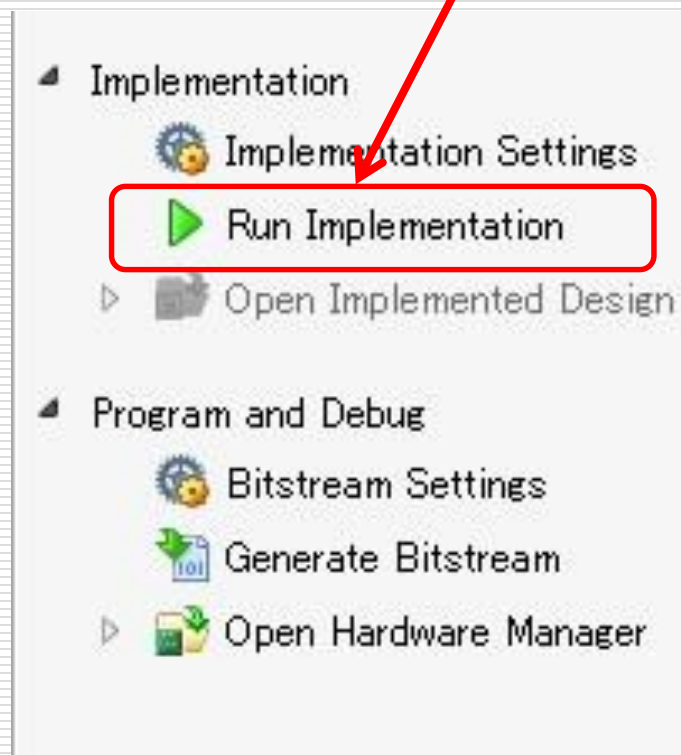
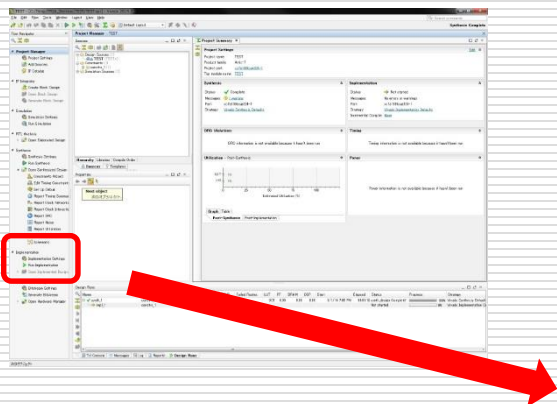


Project summaryページのSynthesis項目

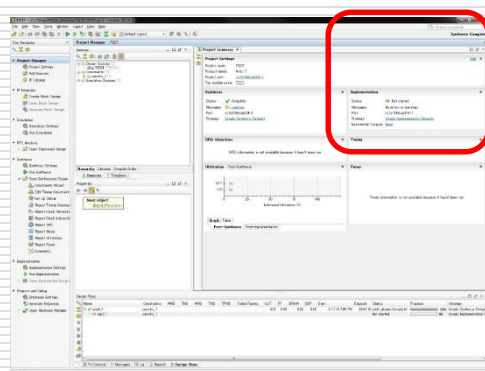
配置配線

配置配線の実行

[Run Implementation]をクリック



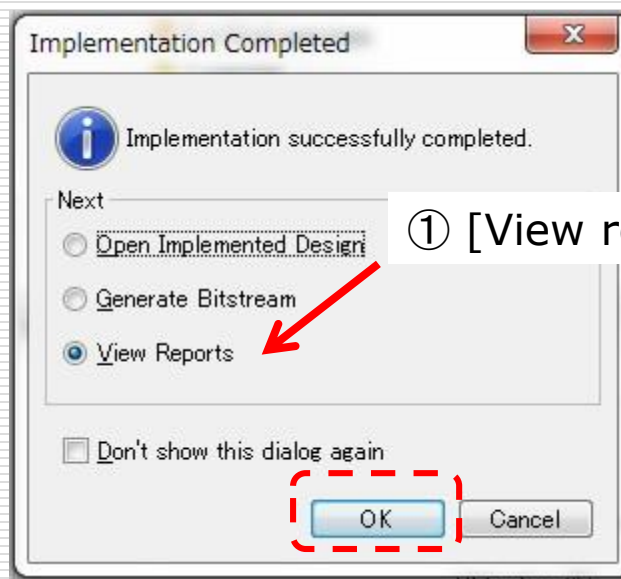
配置配線中の画面



StatusがRunningになる

配置配線結果の確認

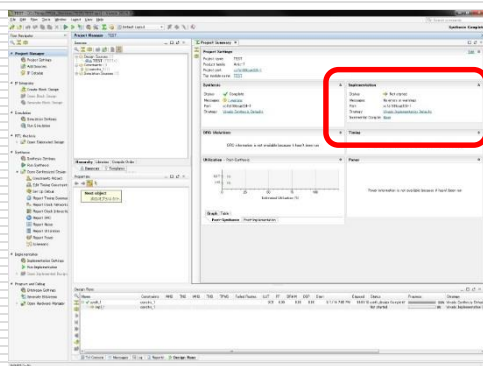
配置配線が終わると表示される



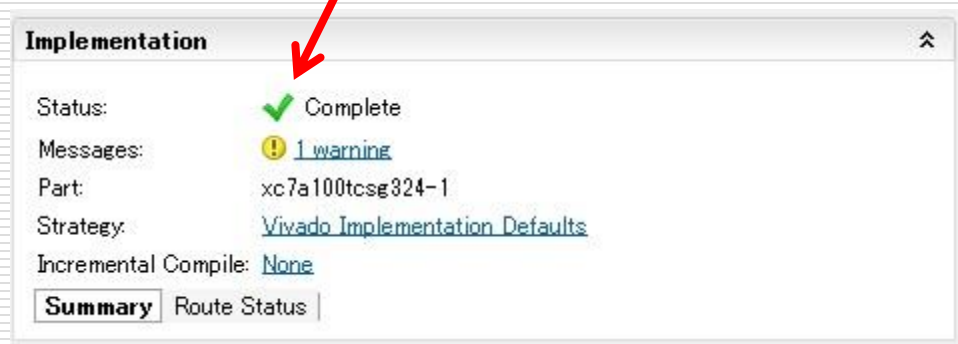
① [View reports]を選択

②クリック


配置配線結果を確認




Completeになっていることを確認
ここではWarningは問題にせず進めます



Implementation

Status:  Complete

Messages:  [1 warning](#)

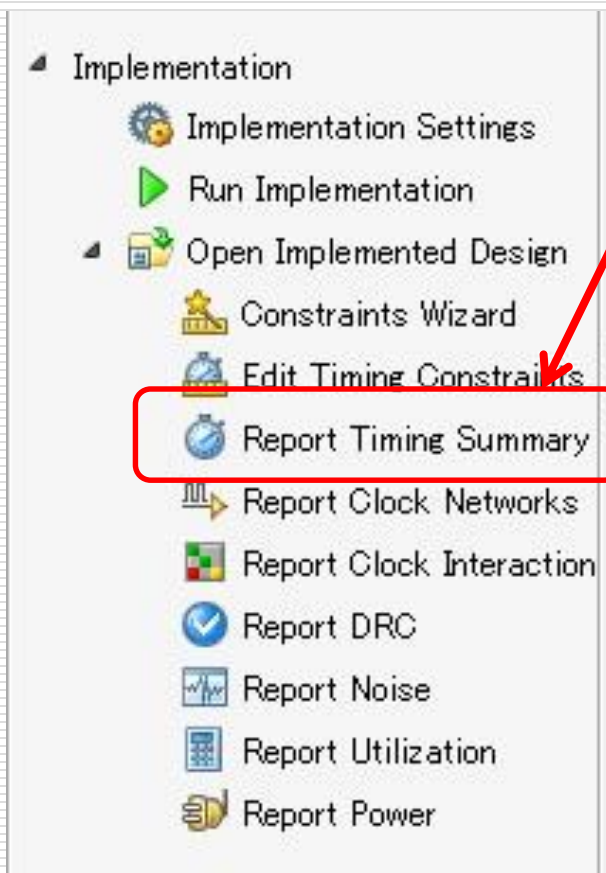
Part: [xc7a100tcsq324-1](#)

Strategy: [Vivado Implementation Defaults](#)

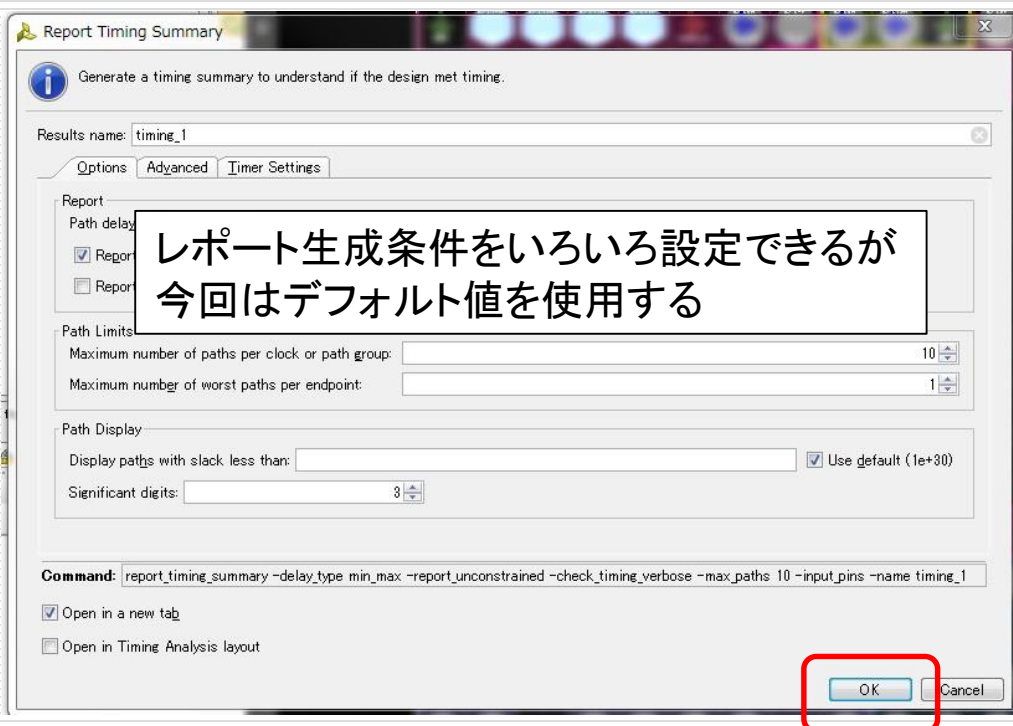
Incremental Compile: [None](#)

Summary | [Route Status](#)

タイミングレポートの生成

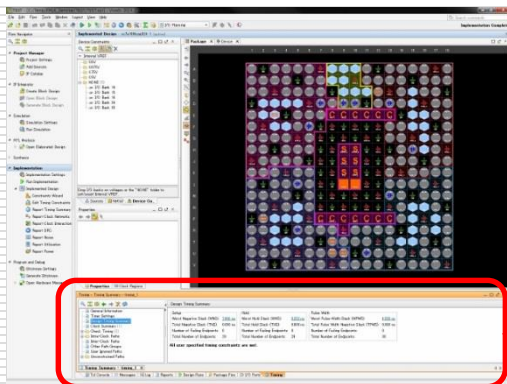


① [Report Timing Summary]を左クリック



② [OK]をクリック

Timing summaryの確認



Timing - Timing Summary - timing_1

Design Timing Summary

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 7.410 ns	Worst Hold Slack (WHS): 0.254 ns	Worst Pulse Width Slack (WPWS): 4.500 ns
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total Pulse Width Negative Slack (TPWS): 0.000 ns
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of Failing Endpoints: 0
Total Number of Endpoints: 29	Total Number of Endpoints: 29	Total Number of Endpoints: 30

All user specified timing constraints are met.

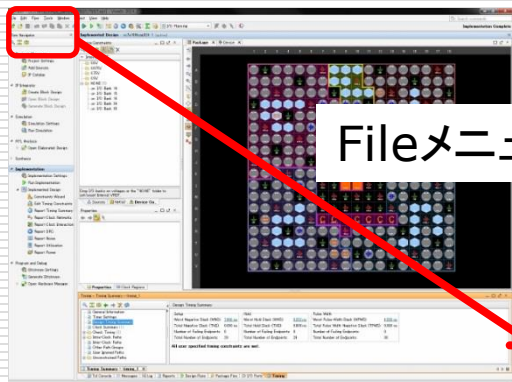
全ての制約が満たされていることを確認

Timing Summary - impl_1 Timing Summary - timing_1

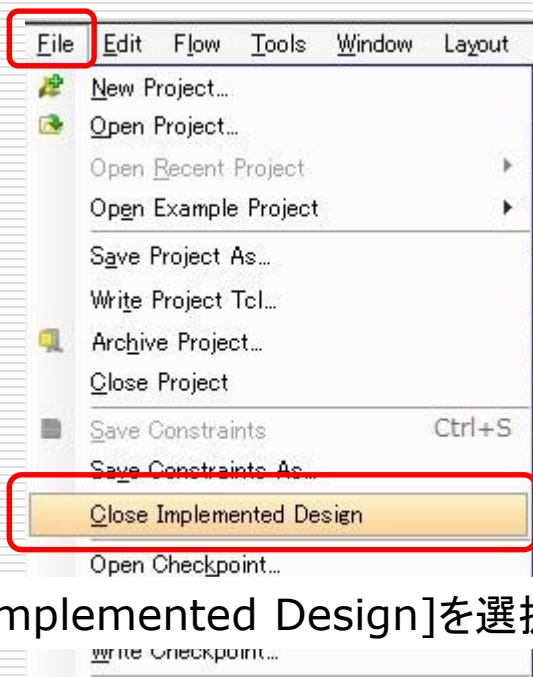
Tool Console Messages Log Reports Package Pins Design Runs Power Timing

今回はWarningがあってもOK

Implemented designを閉じる



Fileメニュー



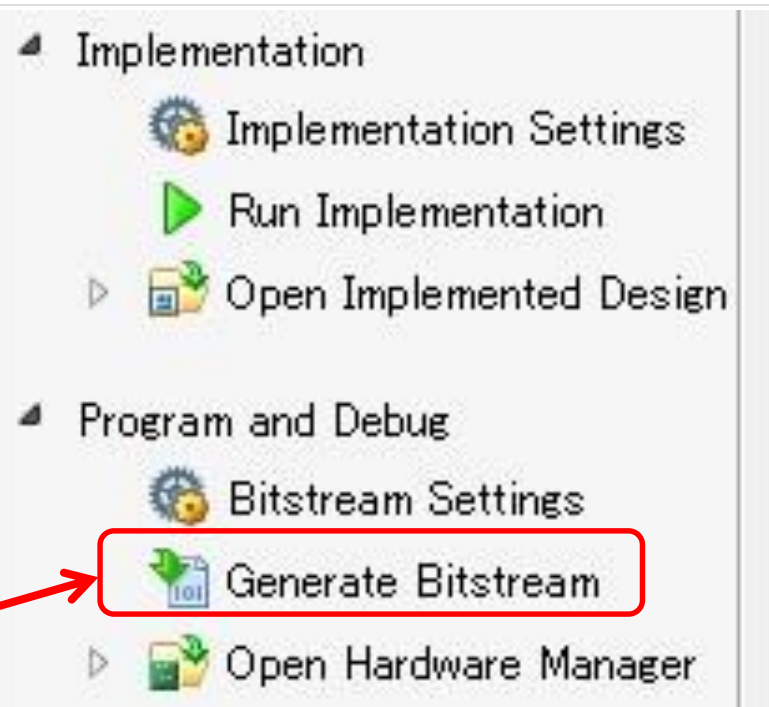
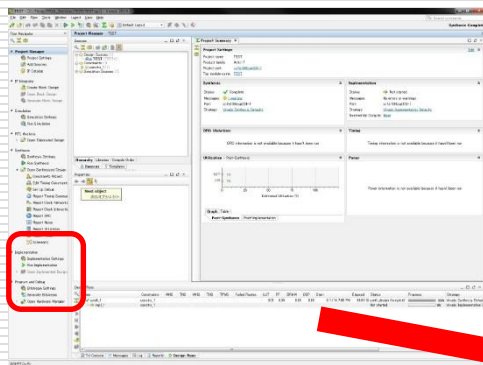
[Close Implemented Design]を選択



OKをクリック

データファイル生成

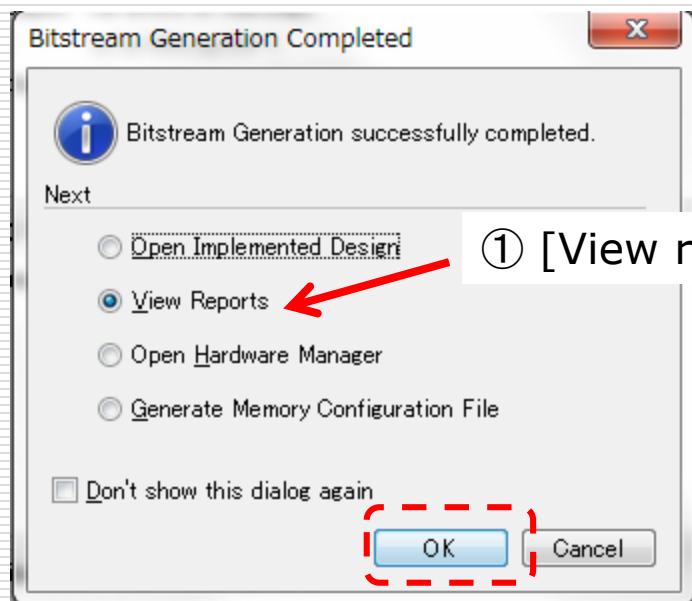
デーファイル生成



[Generate Bitstream]をクリック

結果確認

データ生成が終わると表示される



① [View reports]を選択

②クリック

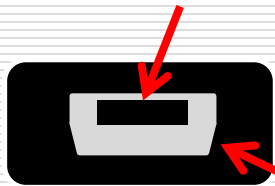
FPGAへダウンロード JTAG mode

ハードウェアの準備

① PCとUSBケーブルで接続

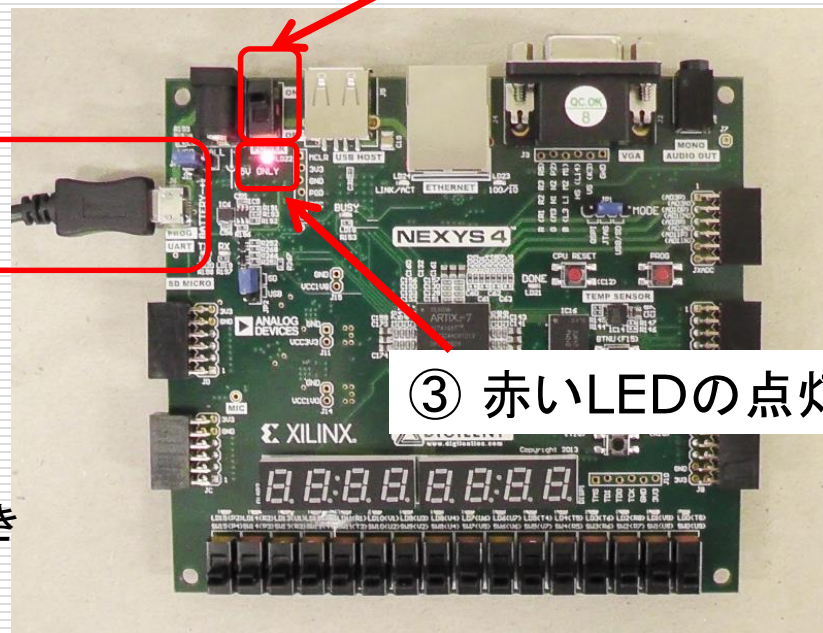
② POWER SWをON

へこんでいる部分を上にして挿す



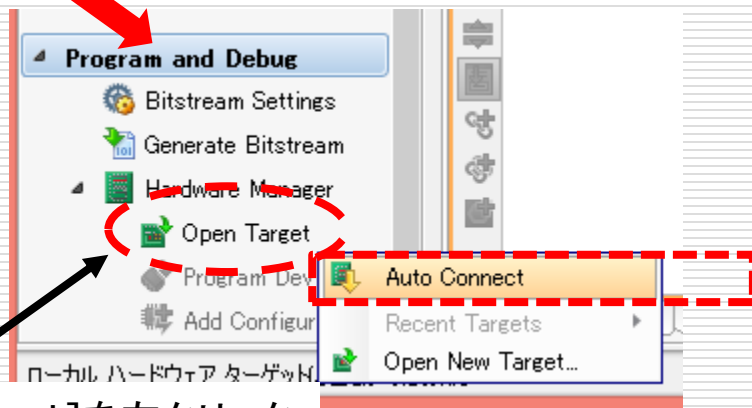
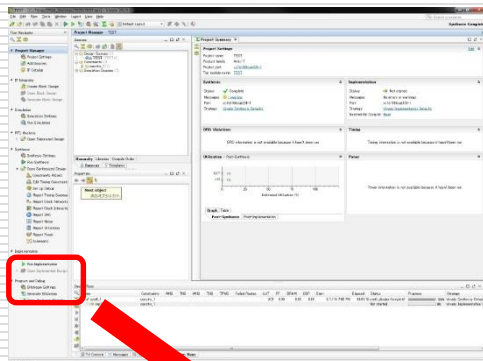
横面が斜めになっている部分を下向き

USBケーブルの向き



③ 赤いLEDの点灯確認

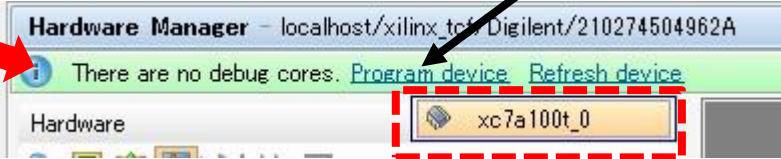
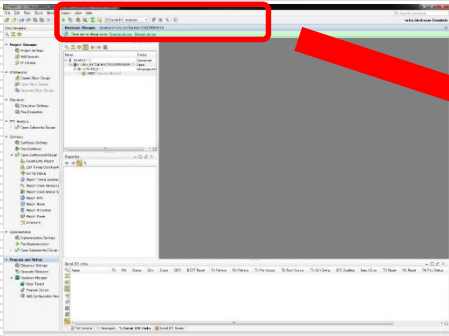
ダウンロードツールの起動



① [Open Target]を左クリック

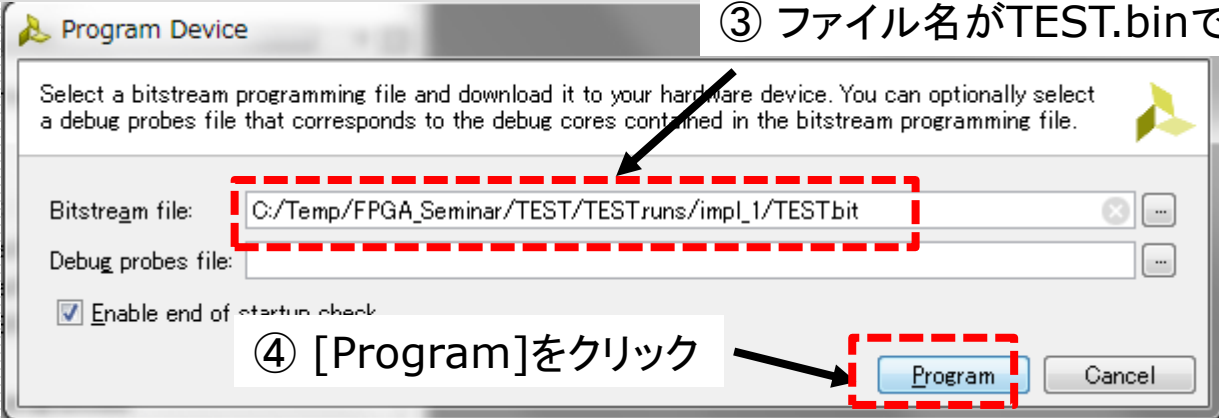
② [Auto connect]を選択

書き込み



① [Program device]をクリック

② [xc7a100t]を選択

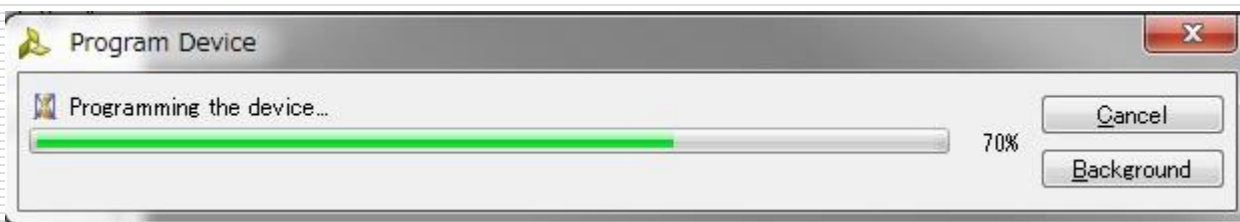


③ ファイル名がTEST.binであることを確認

④ [Program]をクリック

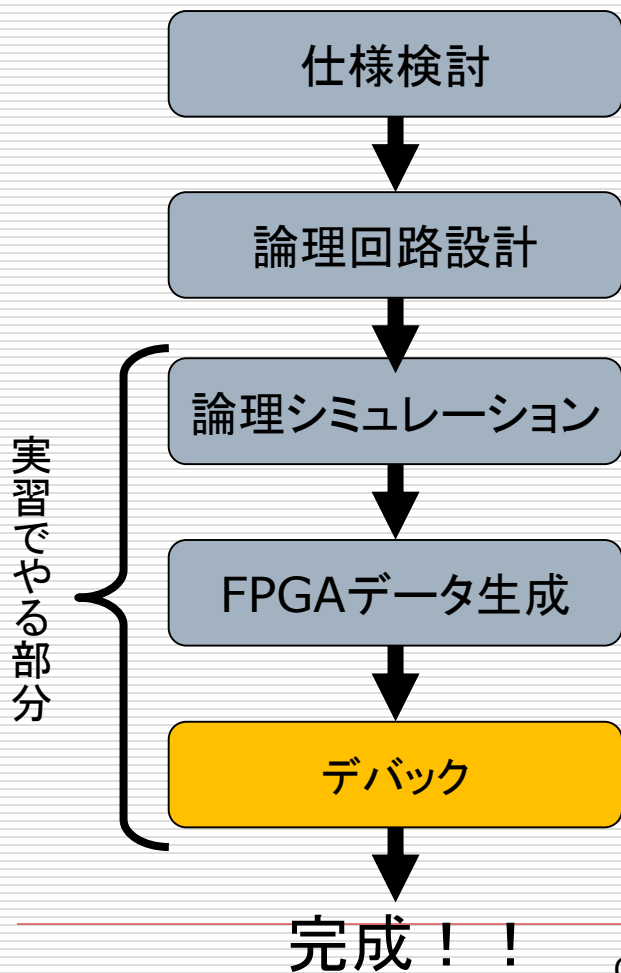
データ書き込みの確認

書き込み中の表示



動作確認

デバック

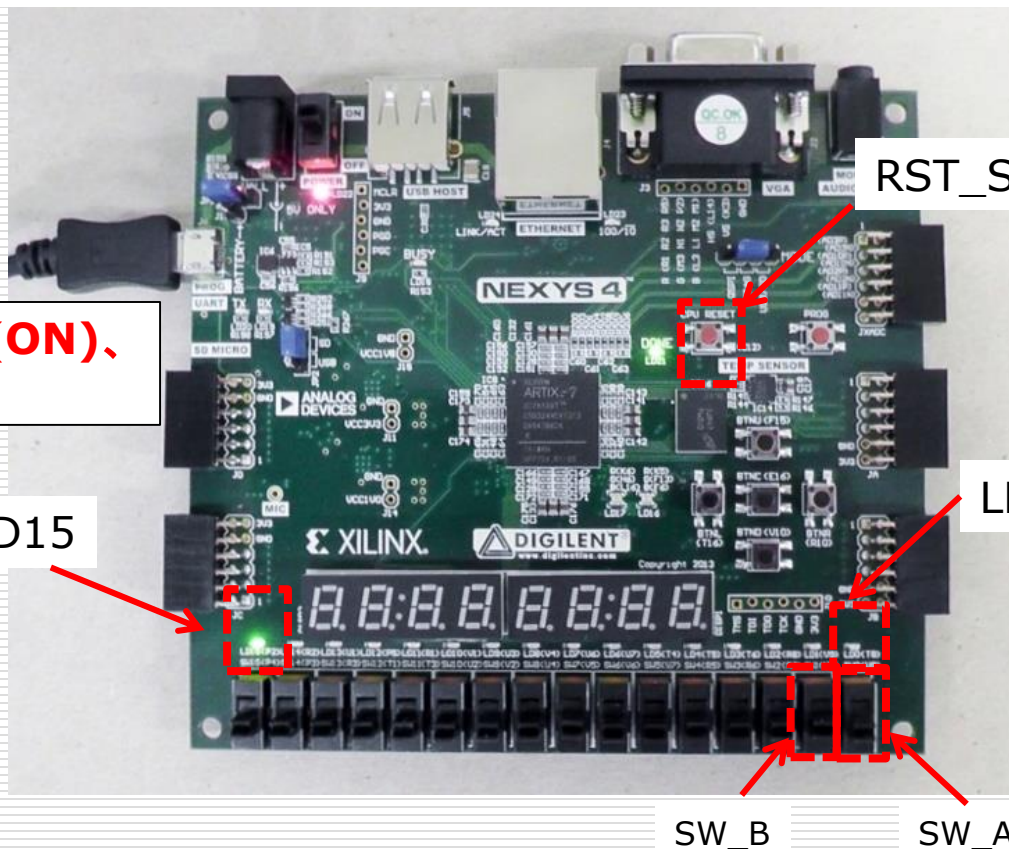


多くの場合、最初は動かないので問題を特定して解決する過程です

今日はSWを動かして動作確認します

動作確認

SWを操作して設計した通りに動作するか試してください



(注意) SWは上で1(ON)、
下で0(OFF)になる

完了！！

動きましたか？

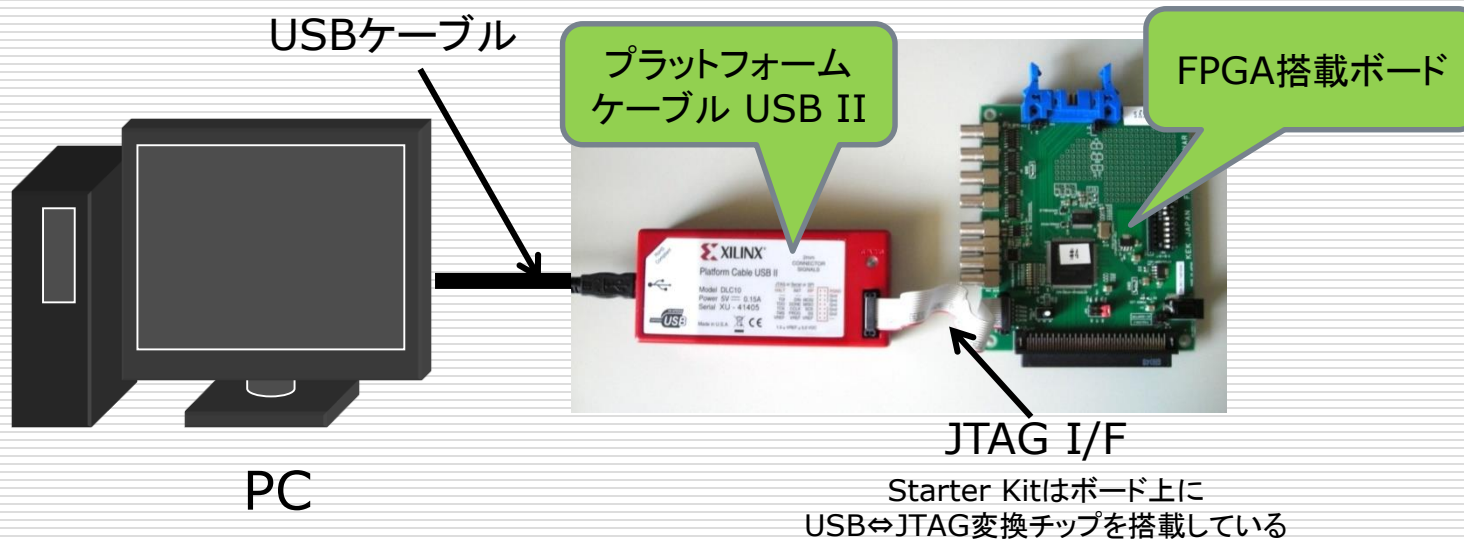
付録

JTAG I/Fによるダウンロード

- 様々なダウンロードI/Fの一つ
- 長所
 - コネクタが小さい(信号線数が少ない)
 - 複数チップを同一コネクタからプログラム可
 - FPGAに直接書き込み可(デバック時に有効)
- 短所
 - 書き込み時間が長い(転送速度が遅い)

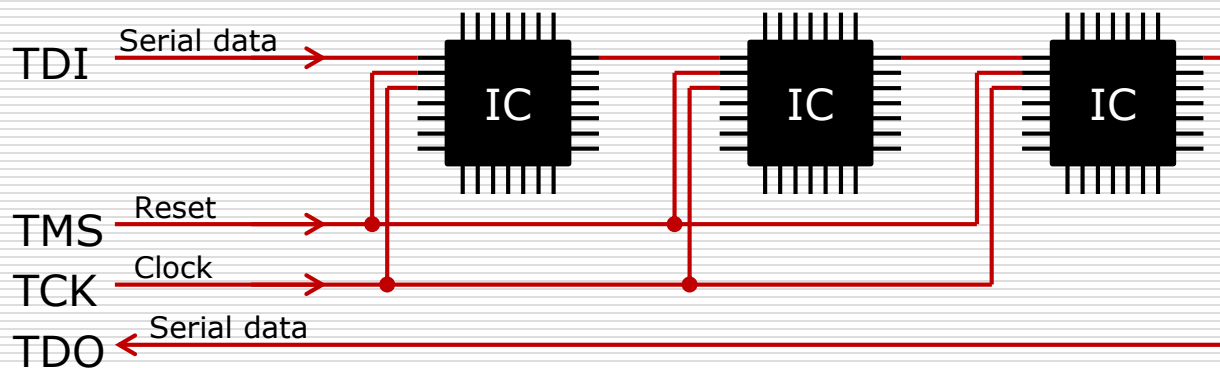
JTAG I/Fによるダウンロード方法

JTAG I/Fを用いた



JTAG I/F

- I/F規格の一つ
- シリアルI/F (信号数4本、電源除く)
 - 1ビットずつ順番に送る
- ボードや実装済みチップのテストの為に開発
- 複数のチップを直列接続可 (Daisy Chain接続)



履歴

- 2012/5/17 第1.0版 ISE13.4対応 内田智久(Esys, KEK/総研大), 林達也(大阪大学)
- 2014/8/7 第2.0版 Vivado2014対応、章構成変更 内田智久(Esys, KEK/総研大)
- 2015/7/31 第3.0版 Vivado2015対応、章構成変更 内田智久(Esys, KEK/総研大)
- 2015/12/04 第3.1版 ハードウェア接続方法を変更 内田智久(Esys, KEK/総研大)
- 2016/01/27 第3.2版 Vivado 2015.4対応 内田智久(Esys, KEK/総研大)
- 2016/06/22 第3.2版 Vivado 2016.2対応 内田智久(Esys, KEK/総研大)