POCO の FPGA 上への実装

1. 論理合成とは

今までは、 Verilog はハードウェア構造を記述し、シミュレーションするだけだった。しかし、記述されたハードウェアは何らかのデバイス上で実現しなければ意味がない。このためには Verilog 記述をゲート部ベルの接続図（ ネットリスト ）に変換し、様々の最適化を行って実際のデバイス上に実装する作業が必要である。これにより POCO の性能、電力、ハードウェア量を評価することができる。

2. FPGA 用の論理合成圧縮ツール

実装対象が FPGA である場合、ベンダが統合設計環境を供給している。この環境を使うと論理合成、最適化、配置配線、構成データの生成までほとんど自動的にやってくれる。 Xilinx 社の ise（最新版は Vivado）、 Altera 社の QuartusII が普及しており、このうち、 web パックと呼ばれる簡易版は無料である。 web パックは一部の機能が制限されているが、 POCO レベルの回路ならば十分合成可能である。

ここでは、タダで誰でも使えるということから Xilinx 社の ise/WebPack を用いる。これは皆さんが自分の PC にダウンロードして使うことができる。設計には制限はあるが、演習程度には十分である。ちなみに Xilinx は Vivado というツールに切り替えようとしているのだが、昨年まで中央大学の環境が超遅かったため、導入できなかった。このドキュメントは ise14.7 を使っているが、来年は Vivado に切り替えが可能になるかもしれない。

3. ise の使い方

synthise.tar をダウンロードして展開する。このディレクトリで、

```
ise
```

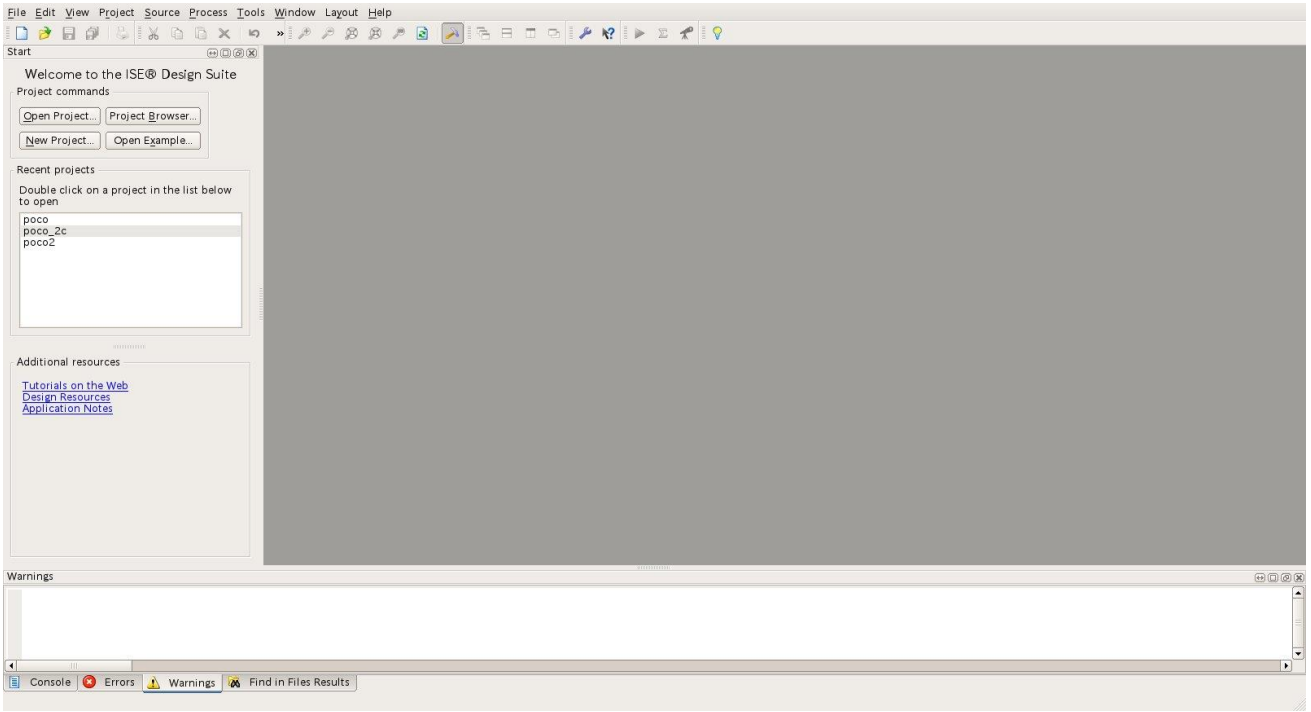
と打ち込むと立ち上がる。 Tip や ise の宣伝の窓が立ち上がるが、これらは全て閉じてしまいメインのウインドウである ISE Project Navigator を前面に出す。

○プロジェクトの生成

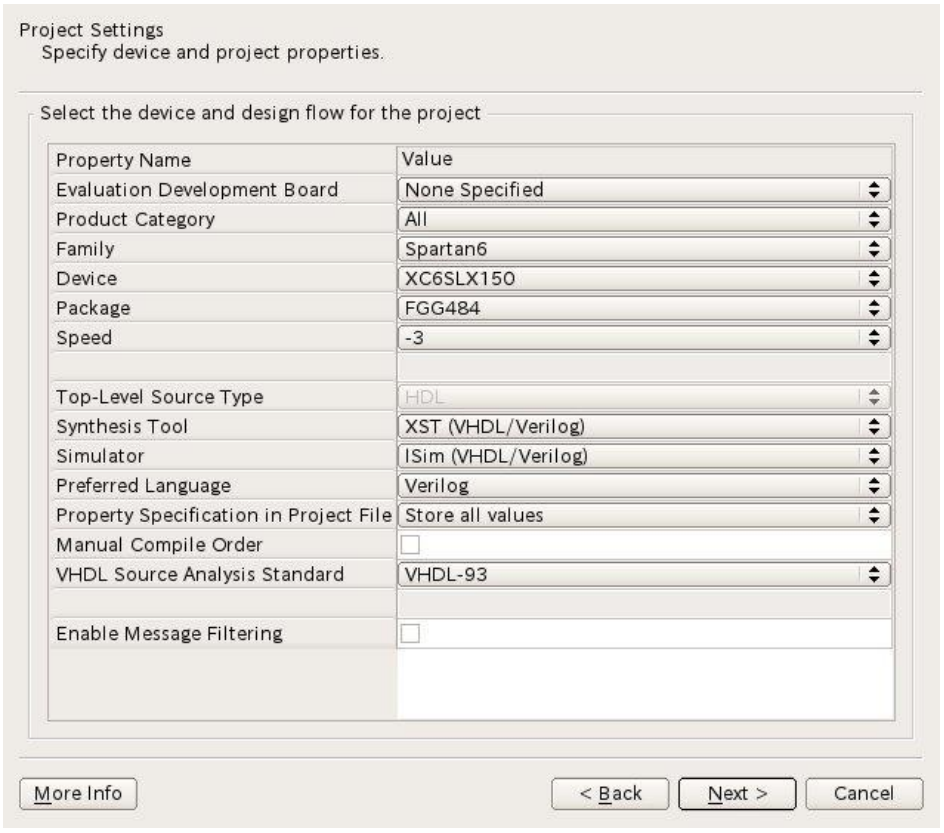
New Project を選択、あるいは File→ New Project を選択

作業を行うディレクトリを指定する。これは皆さんの環境に合わせて決めて欲しい。

Next を選択し、デバイスを指定する。ここでは良く利用する Spartan-6 ファミリのうち、 XC6SLX150 を指定している。パッケージは FGG676 というピン数の多いのを使っている。これは POCO に対しては大きすぎる。Next を選択し、Finish するとプロジェクトができる。



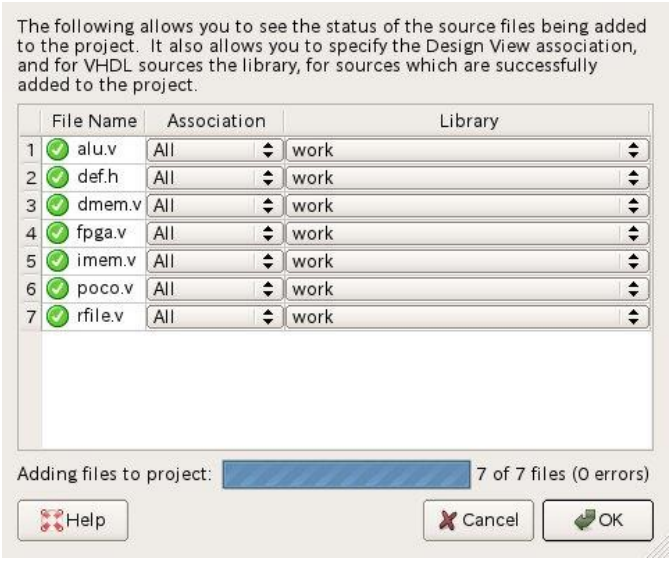
スタート画面



デバイスの設定

〇設計ファイルの読み込み

左上の Hierarchy という窓の xc6slx150-3fgg484 を右クリックし、 Add Source を選択する。Fpga.v、 poco.v 、alu.v 、 rfile.v、imem.v、dmem.v, def.h を全て指定する。緑色のチェックのついた表が出てくるので、 OK を押してやる。

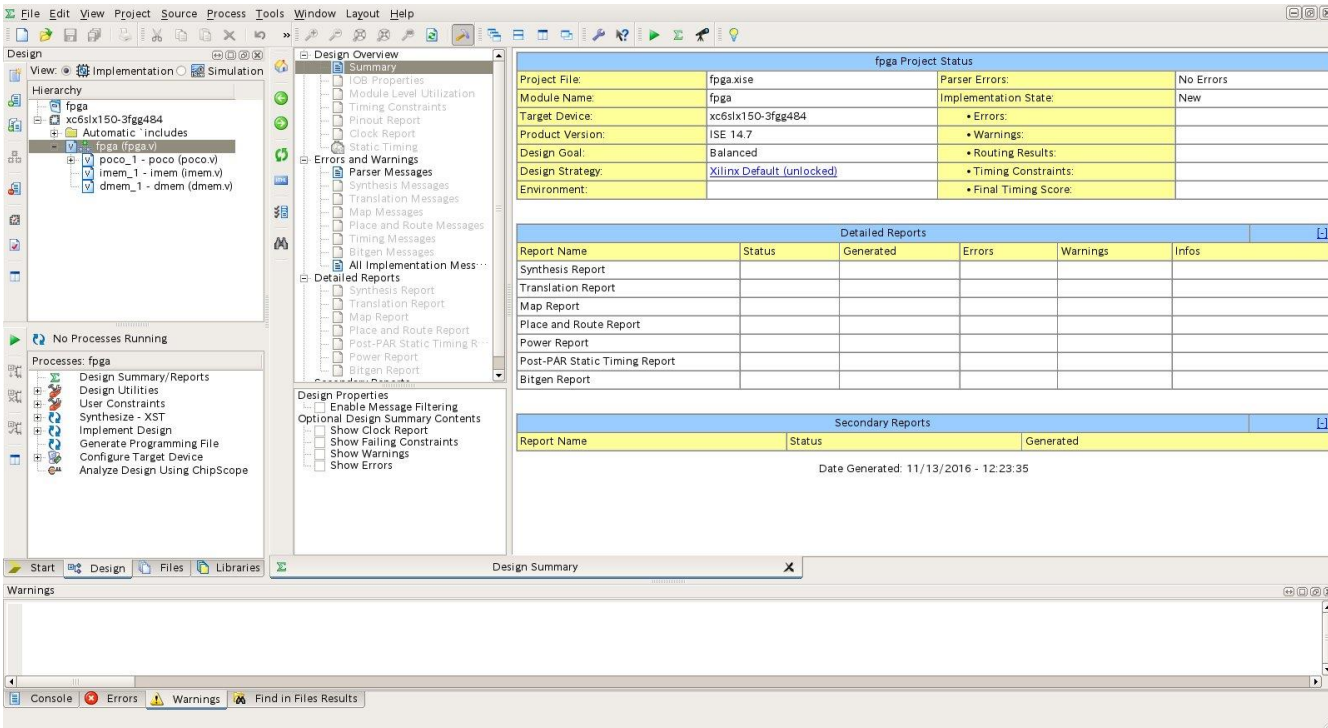


ファイルの読み込み

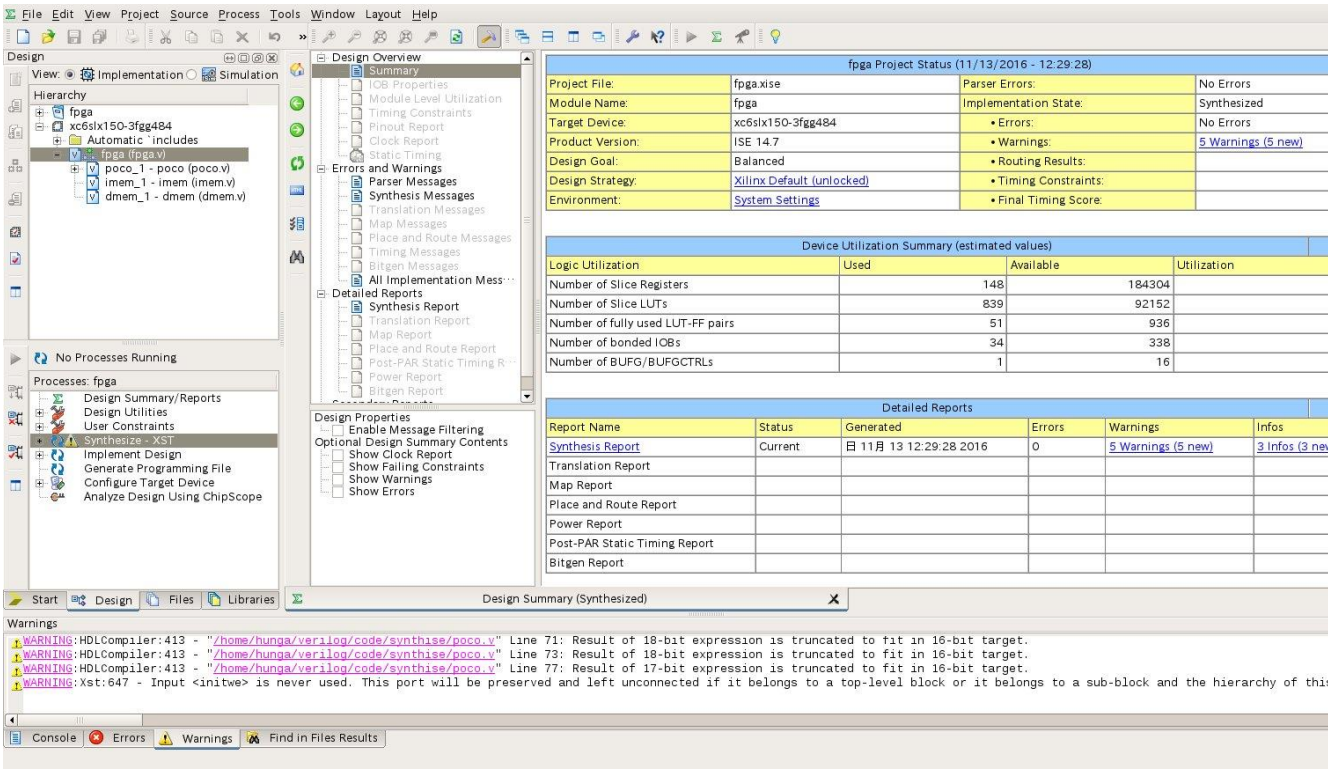
これで準備は完了である。左上の窓に設計階層が見える。FPGA の＋をクリックすると中身が見える。この設計は、FPGA の上にメモリを含めた POCO 全体が載って動作するようになっている。ちゃんと動作させるためには、プログラム入力も必要なので、再び Add Source して、全てのファイルを選択し、imem.dat を選んで OK をして付け加えてやる。

〇論理合成

次に左上の窓の fpga(fpga.v)をクリックし、左下の窓の Synthesize-XST をダブルクリックすると論理合成がスタートする。Verilog 記述に問題があるとここでエラーになる場合がある。Warning がいくつか出るが気にしない。



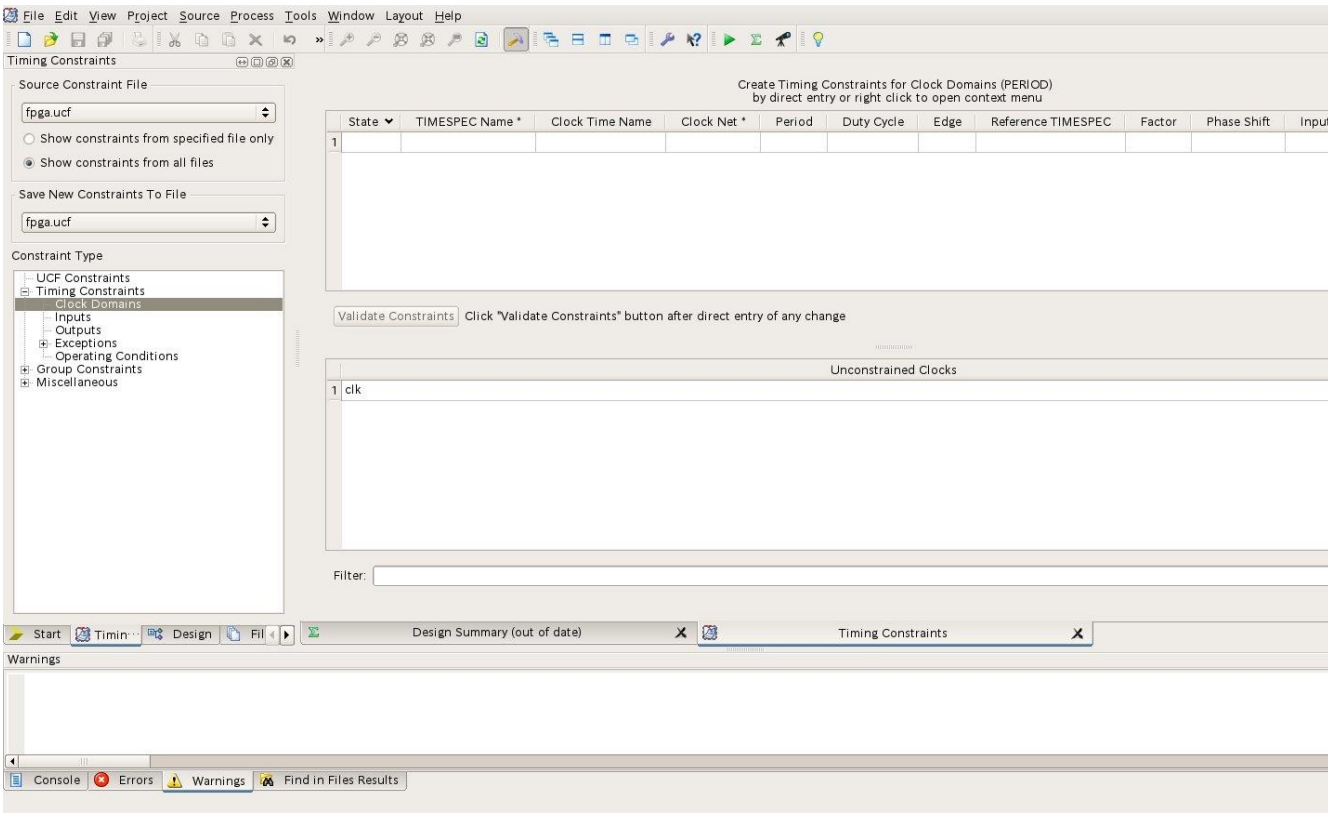
プロジェクトのセットアップ終了時の画面



論理合成終了時の画面

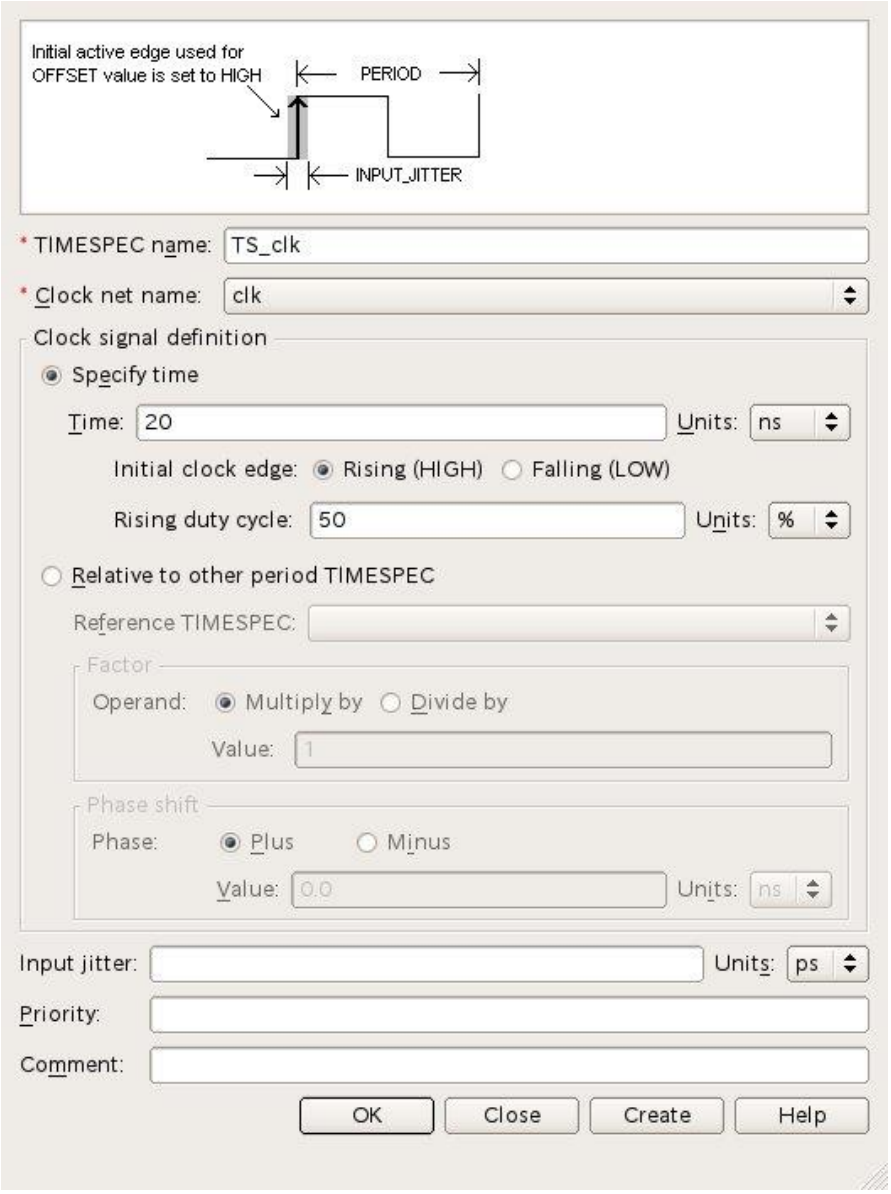
〇制約ファイルの生成

左下の窓の User Constraints の＋をクリックし、一番上の Create Timing Constraints をクリック。メッセージボックスが出るので Yes を押すと Timing Constraints Editor が立ち上がる。



タイミング制約設定画面

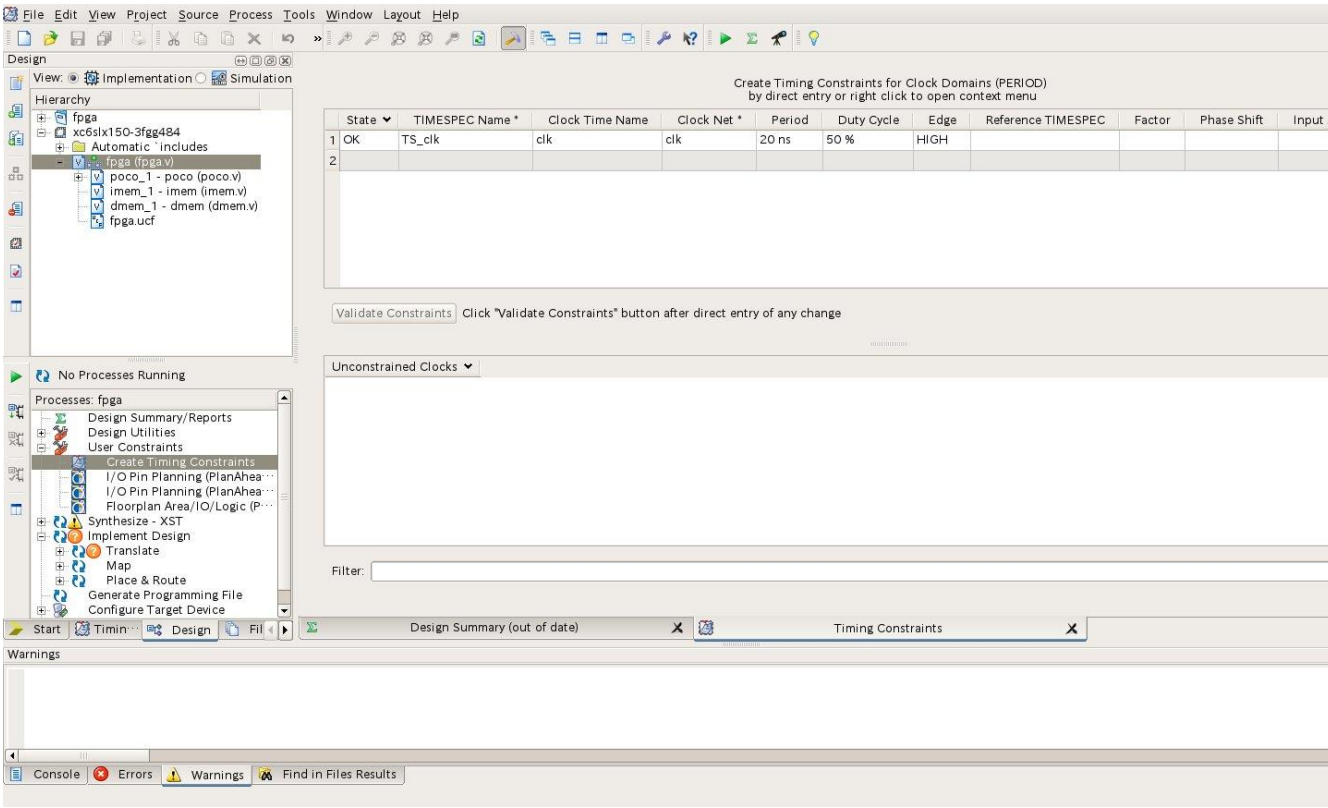
下の窓の Unconstrainted Clock を右クリックし Create Constraint をクリックすると、クロック設定画面が出るので、クロック周波数とデューティ比を設定する。ここでは 50MHz(20nsec)、50%を設定する。OK を押すとクロックが上の画面に移動する。この画面で入力遅延、出力遅延の設定ができるが、今回はこれらは行わず、このまま設定を保存（左上のフロッピー印をクリック）し、左下窓枠の Design をクリックして元の画面に戻す。



クロックの設定

〇配置配線

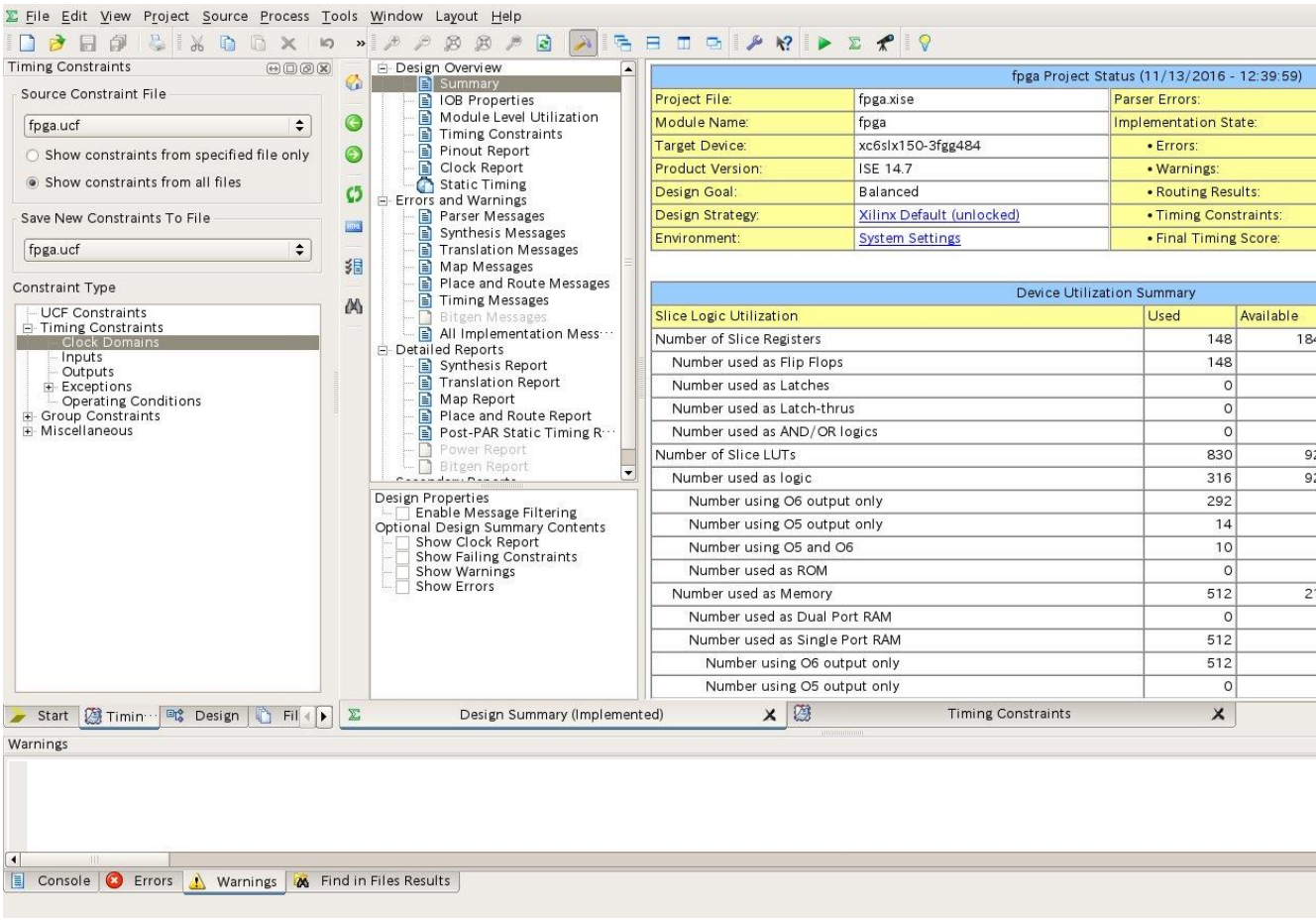
Implement Design をダブルクリックすると配置配線が始まる。すべて終わると緑色の〇が出る。ここは少し時間が掛かる。



配置配線実行前の画面

〇結果の確認

右の窓枠の Design Summary をクリックすると、結果が表れるので、数値を確認する。重要なのは、Number of Slice LUTｓでこれが利用した LUT の数を示す。今回は１%程度である。右の窓中の左の脇窓の Post-PAR Static Timing Report をクリックするとクリティカルパス、最大動作周波数が出てくる。最後のところのまとめを見ると良い。



レポート画面

演習

ハイエンドのデバイスである Virtex-6 XC6VCX130T FF484 を使って合成して利用スライス数と動作周波数を求めよ。デバイスを変えるにはデバイスの所をダブルクリックして必要なものを選べばよい。