

タイミング・シミュレーションの手順

動作環境

小局では、以下の環境で動作の確認をしています。

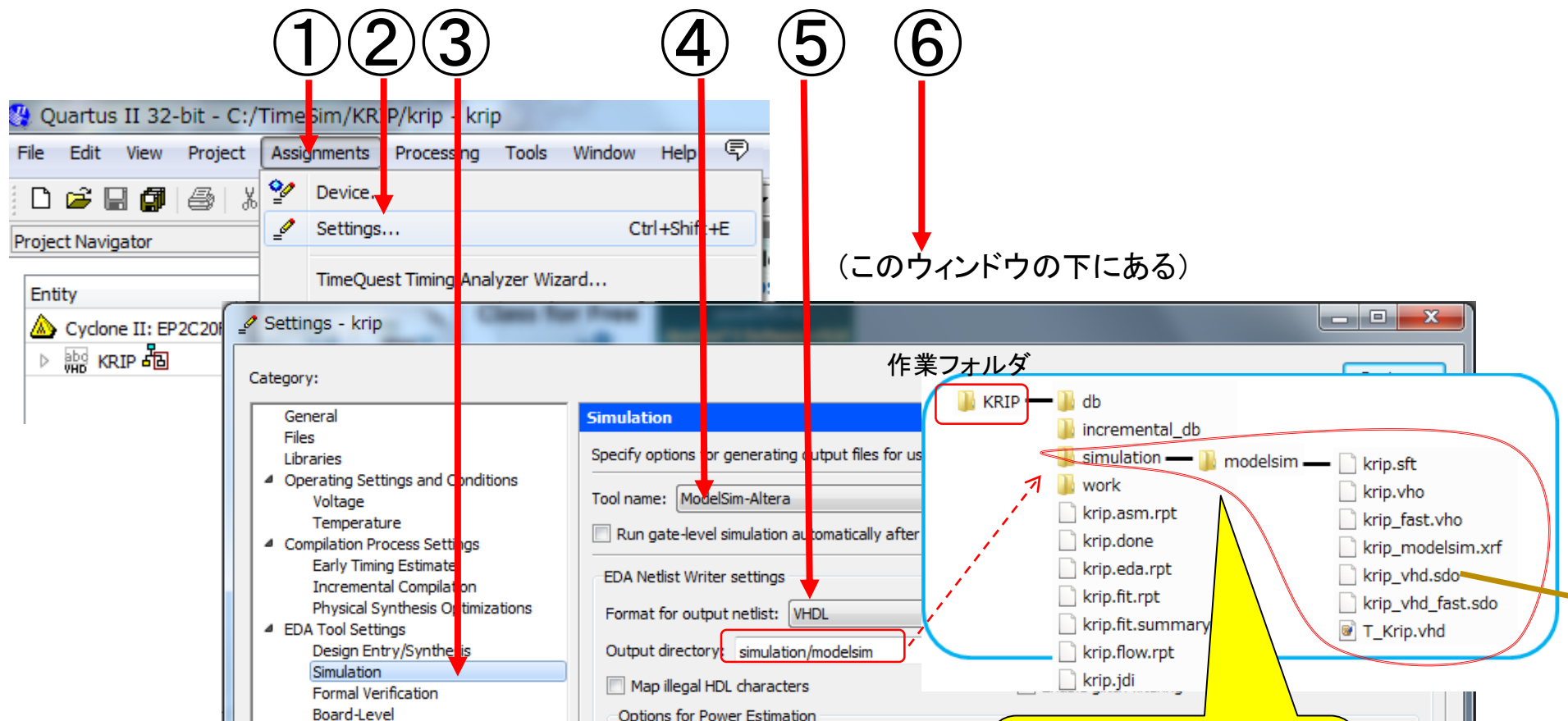
Quartus II のバージョン＋デバイスの選択

Quartus II ver13.1＋Cyclone III

Quartus II ver13.1＋Cyclone IV

Quartus II ver14.0＋Cyclone IV

2014.12.10.現在

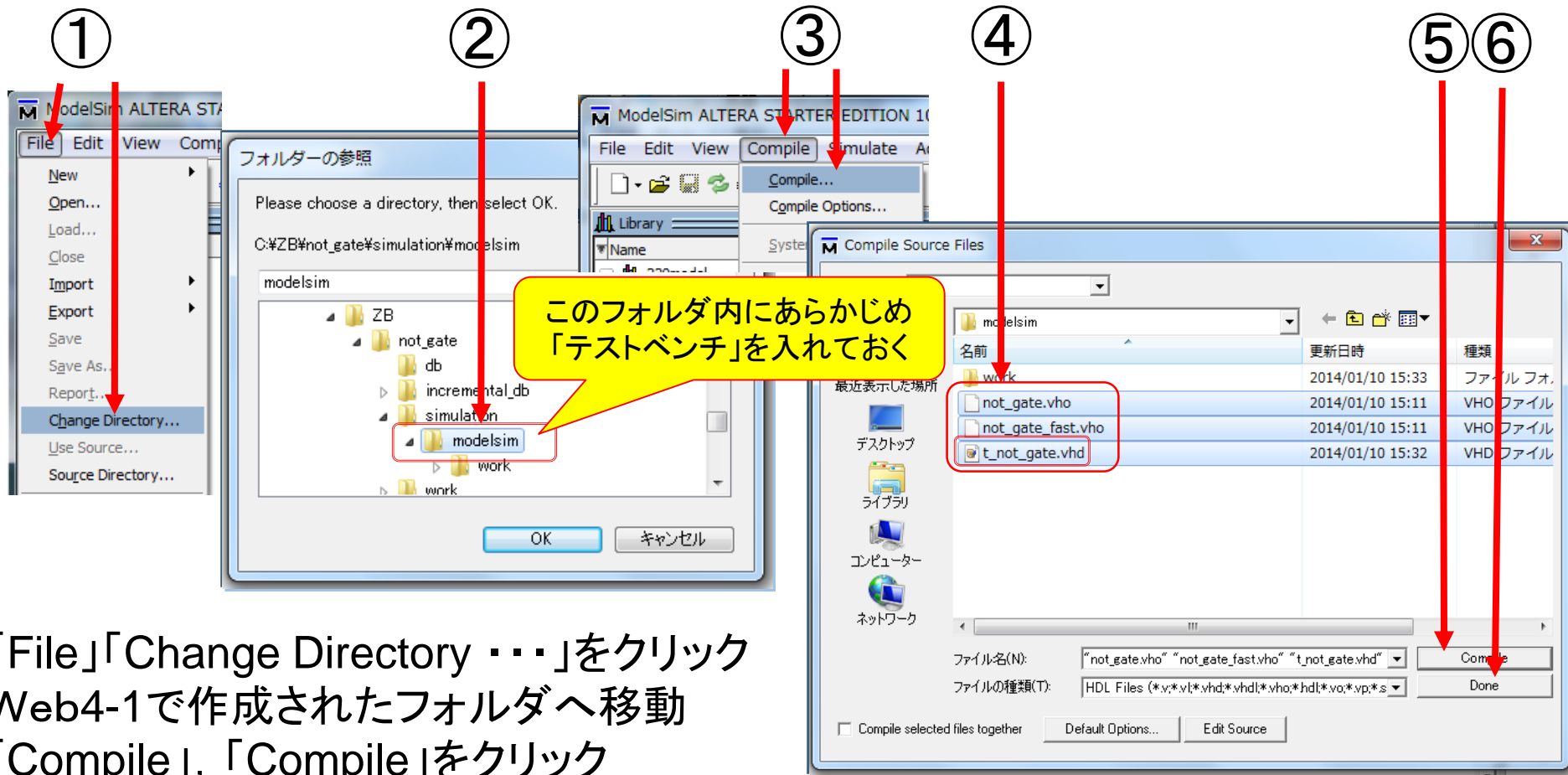


- ①「Assignments」をクリック
- ②「Setting...」をクリック
- ③「Simulation」をクリック
- ④「ModelSim-Altera」を選択
- ⑤「VHDL」を選択
- ⑥「OK」をクリック

EDA Toolを設定してコンパイルすると、このフォルダが作成されデータが書き込まれる

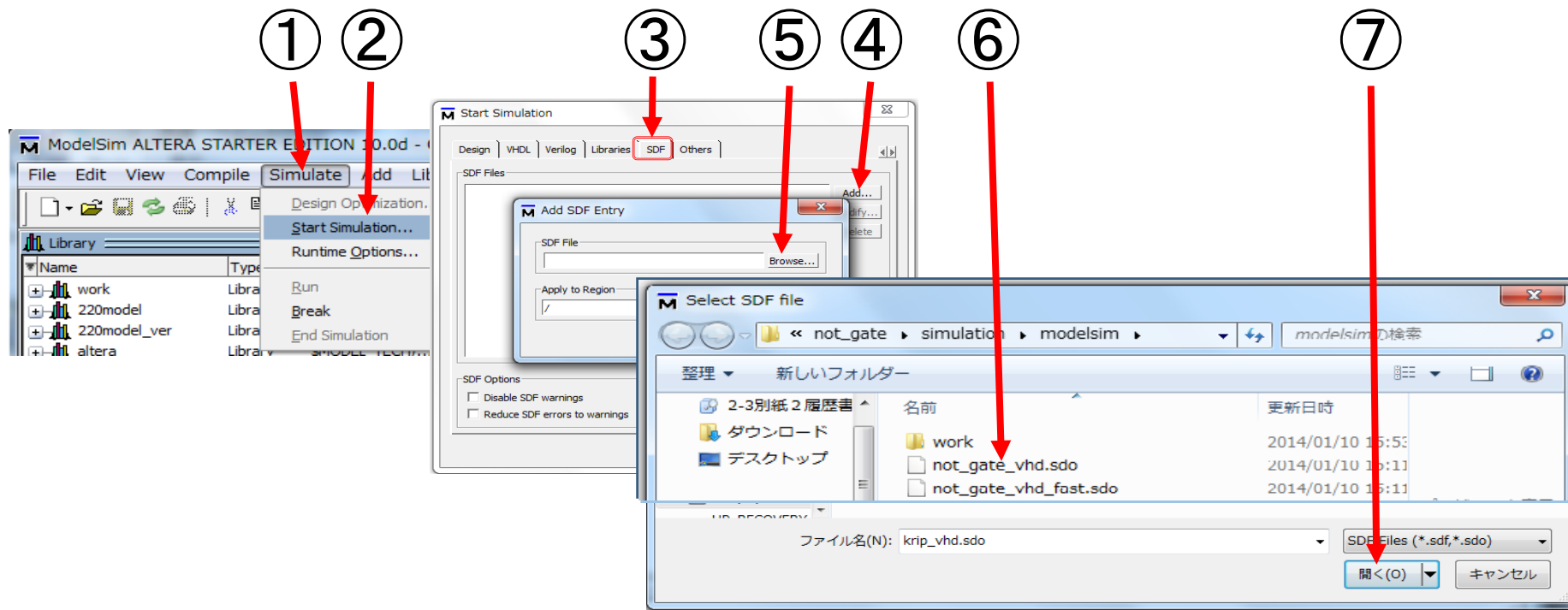
フォルダ内に.sdoファイルが生成されない場合があります。Quartus IIのバージョンやデバイスの種類を変えてみたり、ご自身で試してみてください

Web4-1. Quartus II での事前設定 (EDAツールの追加)



- ①「File」「Change Directory・・・」をクリック
- ②Web4-1で作成されたフォルダへ移動
- ③「Compile」, 「Compile」をクリック
- ④すべてのファイルを選択
- ⑤「Compile」をクリック
- ⑥エラーがなければ「Done」をクリックして終了

Web4-2. ModelSimでコンパイル



- ①「Simulation」をクリック
- ②「Start Simulation...」をクリック
- ③「SDF」タブをクリック
- ④「ADD」をクリックするとAdd SDF Entryが開く
- ⑤「Browse」をクリックするとSelect SDF fileが開く
- ⑥回路名_vhd. sdoファイルを選択
- ⑦「開く」をクリック

⑧

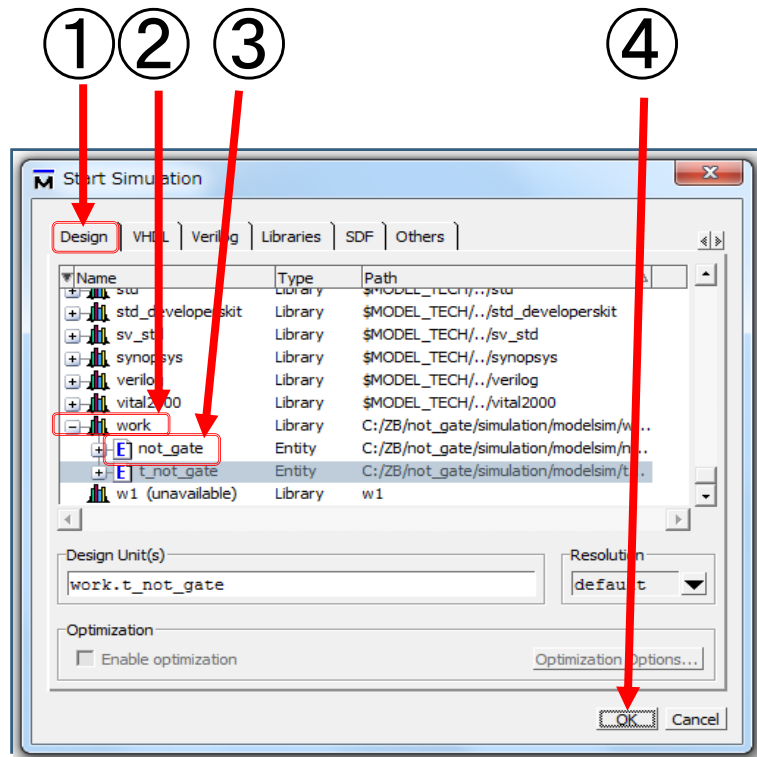
⑨

インスタンス名の先頭の「/」は消さないこと

```
Ln# |  
1 | library ieee;  
2 | use ieee.std_logic_1164.all;  
3 | use ieee.std_logic_unsigned.all;  
4 |  
5 | entity T_Krip is  
6 | end T_Krip;  
7 |  
8 | architecture SIM of T_Krip is  
9 | component Krip  
10 | port ( CLK: in std_logic;  
11 | LED0, LED1, LED2, LED3,  
12 | end component;  
13 |  
14 | signal CLK: std_logic:= '0';  
15 | signal LED0, LED1, LED2, LED3, I  
16 | begin  
17 | U01:Krip port map( CLK=>CLK, LEI  
18 |  
19 | process begin  
20 | for i in 1 to 30 loop  
21 | CLK <= not CLK ; wait for  
22 | end loop;  
23 |
```

⑧シミュレート対象のインスタンス名を入力する

⑨「OK」をクリック



- ①「Design」タブを選択
- ②workフォルダをクリック
- ③テストベンチを選択
- ④「OK」をクリック



(本書掲載の図6-5と比較してみよう！)

- ⑤「Simulate」をクリック
- ⑥「Run」「Run-All」の順に選択
- ⑦結果が表示される

Web4-5. タイミング・シミュレーションの実行