

NCO MegaCore ファンクション のシミュレーション手順

ver.14

目次

1. はじめに	3
2. 開発ツールおよびデザイン例	4
2-1. 開発ツール	4
2-2. デザイン例	4
3. シミュレーション手順	11
3-1. シミュレーションの設定	11
3-2. msim_setup.tcl の修正	12
3-3. テストベンチの修正	14
3-4. シミュレーション手順	15
改版履歴	25

1. はじめに

この資料は、アルテラの数値制御発振器用の IP である NCO MegaCore[®] ファンクション(以下、NCO と表記)を使用したデザインにおける RTL シミュレーションの手順を参考用途として説明したものです。

2. 開発ツールおよびデザイン例

開発ツールおよびデザイン例に関して、下記の条件下で説明しています。

2-1. 開発ツール

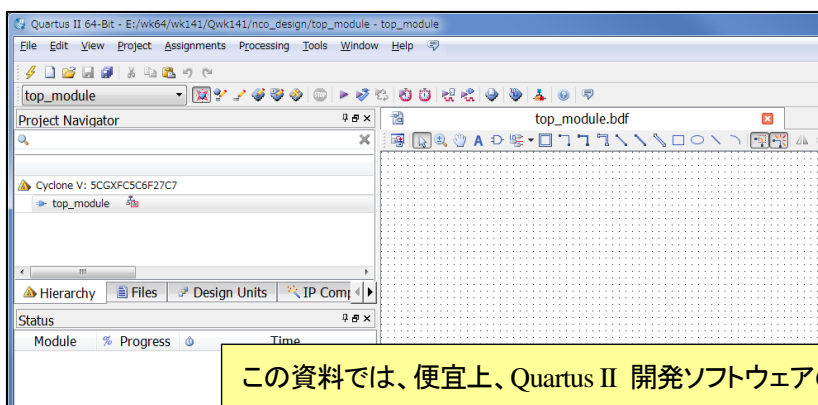
- ・ 論理合成ツール: Quartus® II 開発ソフトウェア 14.1.0
- ・ シミュレーション・ツール: ModelSim® -Altera® 10.3c

2-2. デザイン例

この資料では、次の内容でデザイン例を構築しています。なお、このデザイン例はあらかじめツールやこの資料の付属として用意されているのではなく、NCO MegaCore ファンクションを生成してユーザ自身で構築する必要があります。

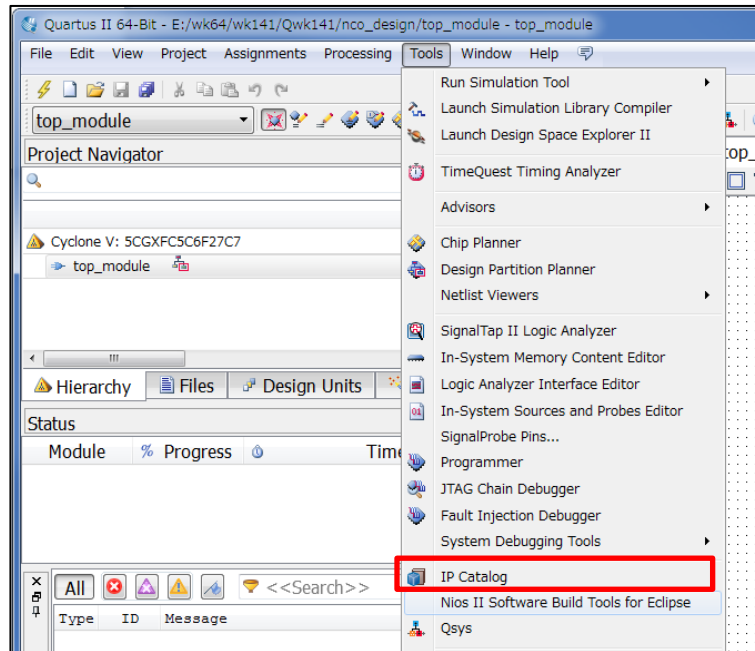
デバイス・ファミリ:	Cyclone® V GX
型名:	5CGXFC5C6F27C7
Quartus II プロジェクト・ディレクトリ:	E:\wk64\wk141\Qwk141\nco_design
※ ユーザ側の環境に合わせて、任意のディレクトリ名を使用しても構いません。	
プロジェクト名:	top_module
最上位階層名:	top_module
NCO インスタンス名:	nco_core

最上位階層のデザインは回路図エディタで作成し、NCO インスタンスはその下の階層に配置されます。

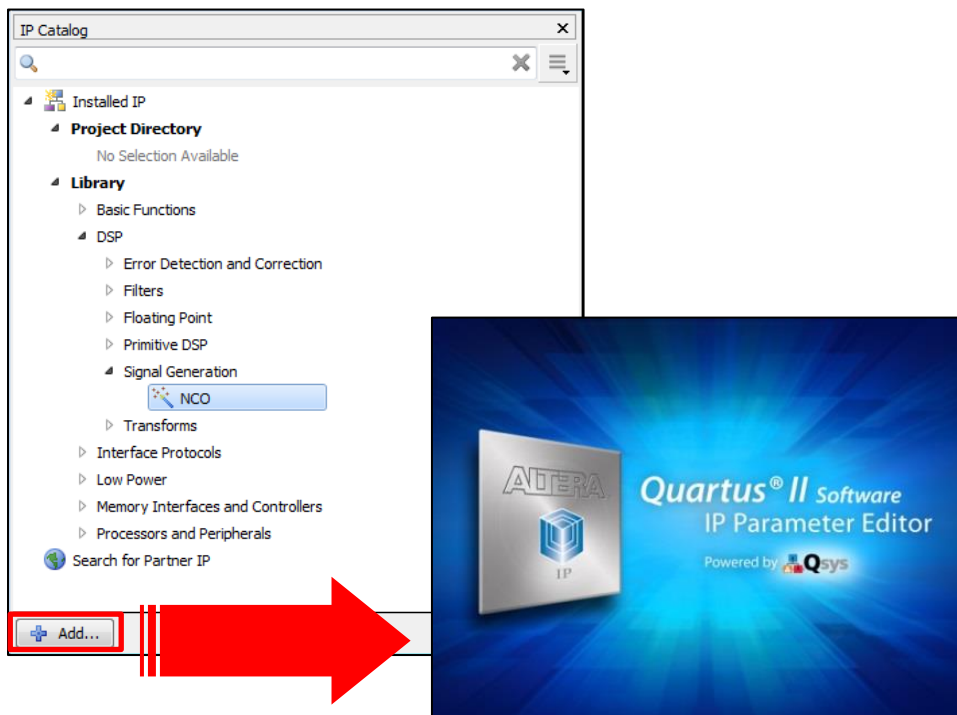


この資料では、便宜上、Quartus II 開発ソフトウェアのプロジェクト作成を行っていますが、シミュレーションだけを実施する場合、Quartus II プロジェクト作成は不要です。

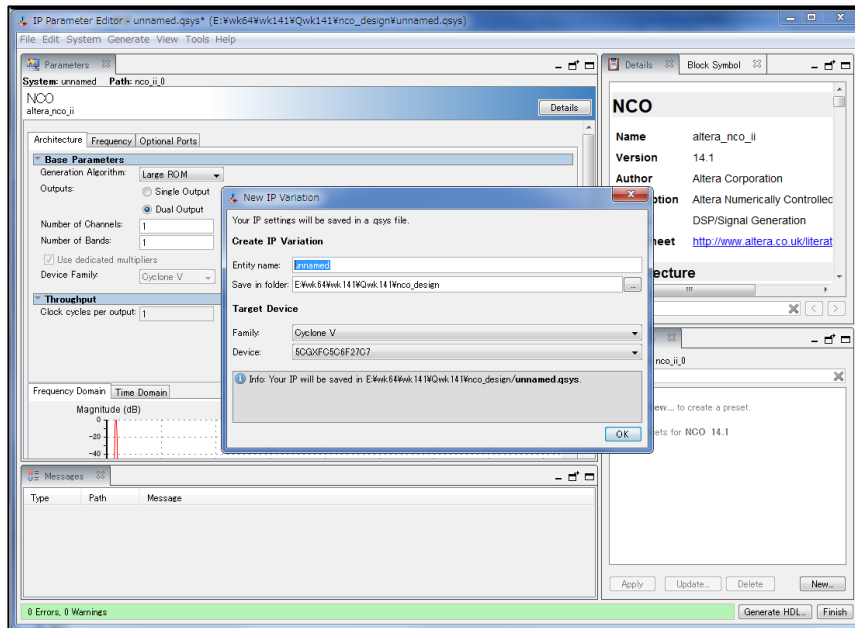
Tools メニュー ⇒ IP Catalog を選択します。



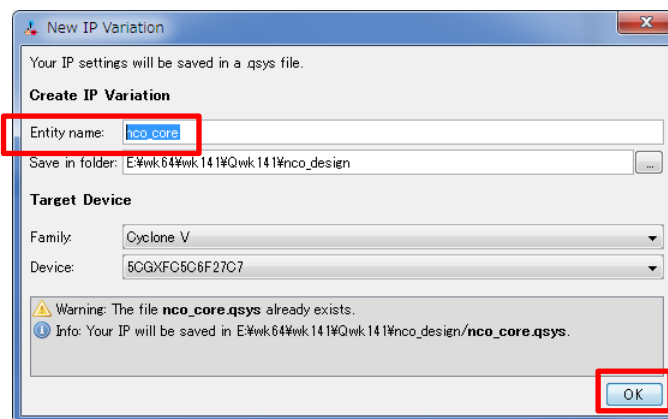
画面右側の IP Catalog 欄の Library ディレクトリから DSP ⇒ Signal Generation ⇒ NCO をハイライトして、左下にある Add ボタンをクリックすると、図のような IP Parameter Editor 画面が起動します。



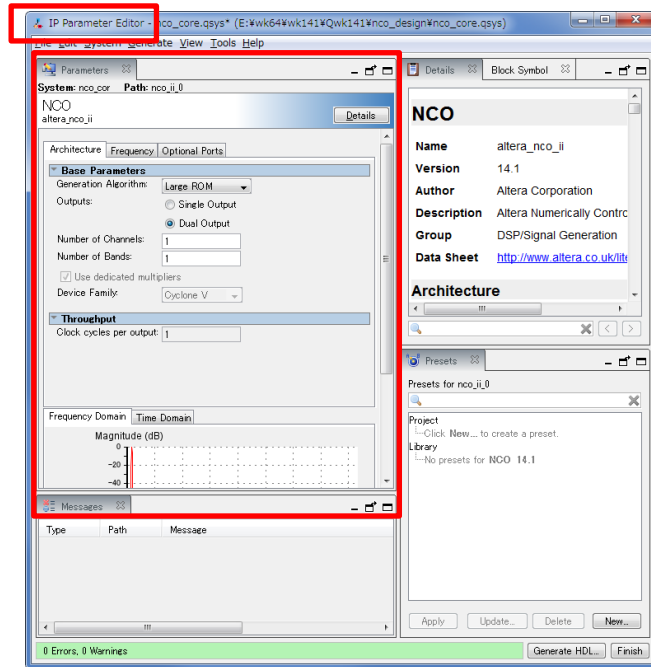
IP Parameter Editor 画面が起動します。



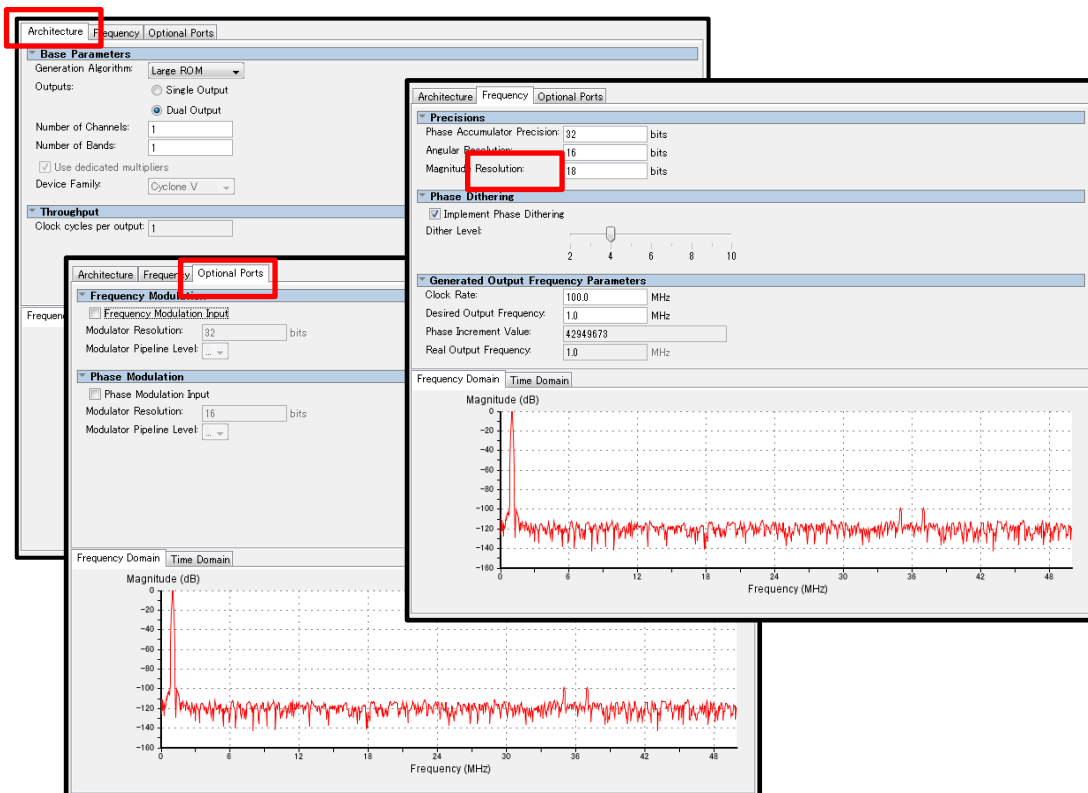
続いて、New IP Variation 画面が起動します。この資料では、Entity name 欄に、nco_core と入力します。その後、OK ボタンをクリックします。



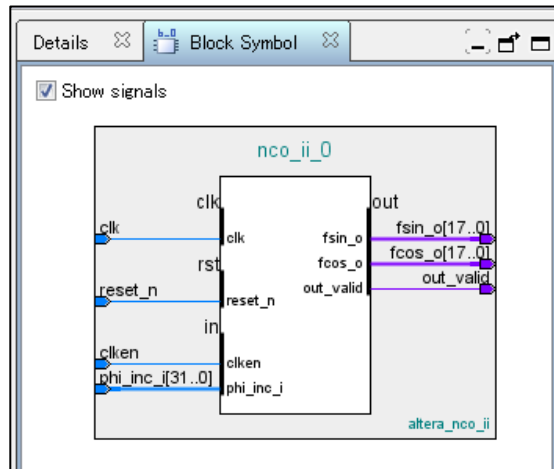
Qsys 画面のフレーム(外枠)では、IP Parameter Editor として表示されます。その中に、NCO の IP Core に対するパラメータ入力画面が表示されます。



この資料では、パラメータの設定内容はデフォルトのものを使用します。



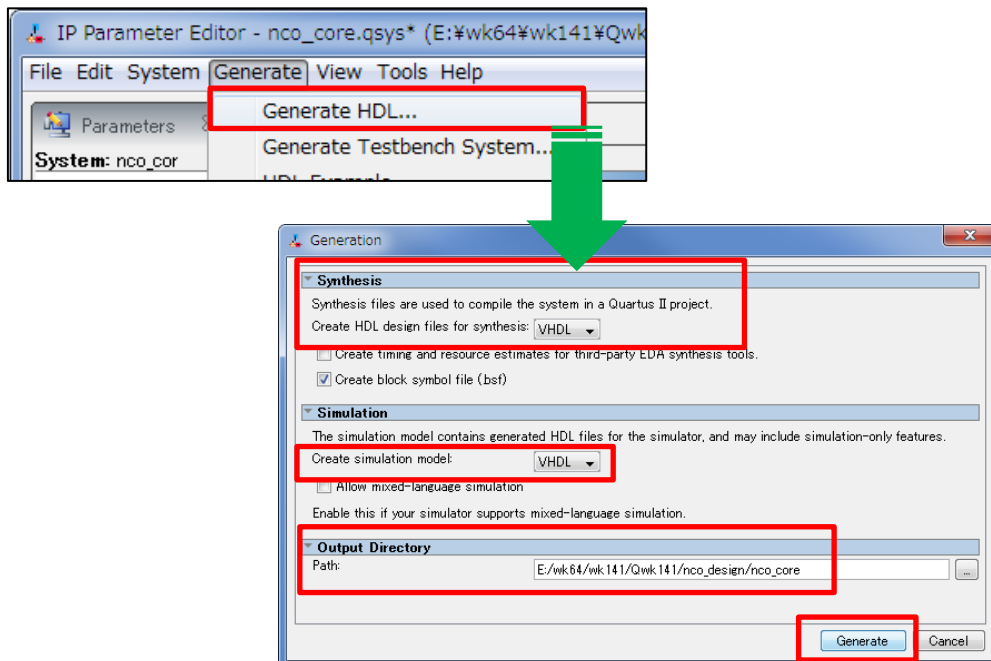
右側の Symbol タブでは、シンボルや信号線の確認ができます。



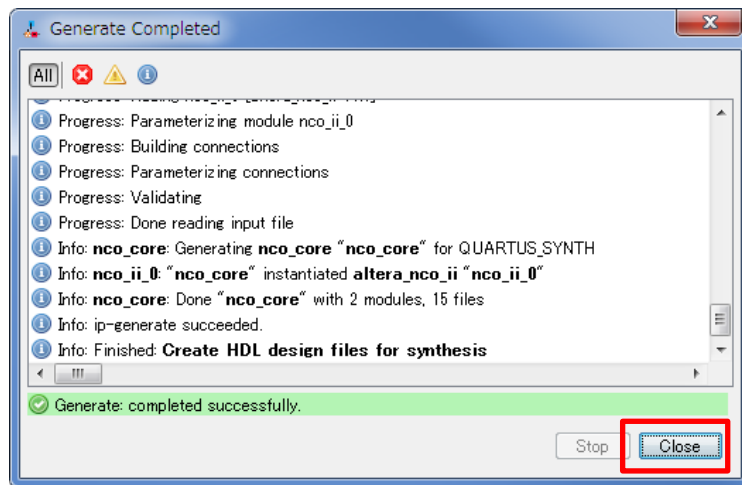
Generate メニュー ⇒ Generate HDL を選択して Generation 画面を起動させたら、Synthesis 欄の Create HDL design file for synthesis を VHDL に設定し、Simulation 欄の Create simulation model を VHDL に設定します。

次に、Output Directory 欄で表示されている Path を確認します。デフォルトでは、(Quartus II プロジェクト・ディレクトリ)/nco_core が表示されています。

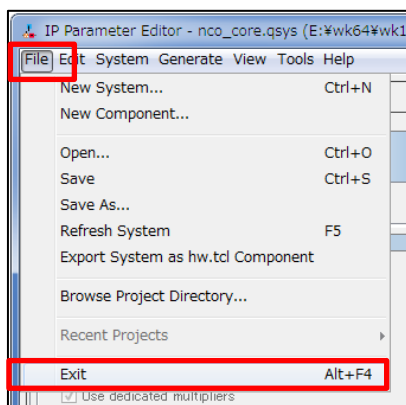
この資料では、Quartus II プロジェクト・ディレクトリは E:/wk64/wk141/Qwk141/nco_design/ を適用していますが、変更せずにデフォルトで表示されている Path を適用します。最後に、Generate ボタンをクリックします。



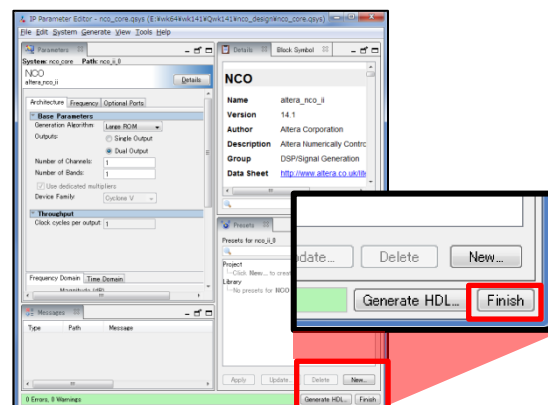
Create HDL design files for synthesis というメッセージが表示されたら、Close ボタンをクリックします。



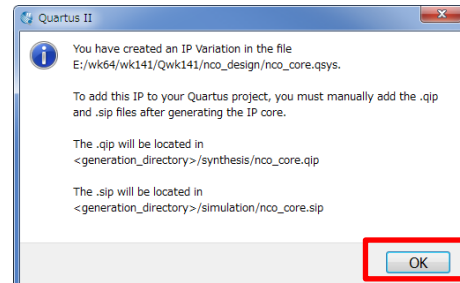
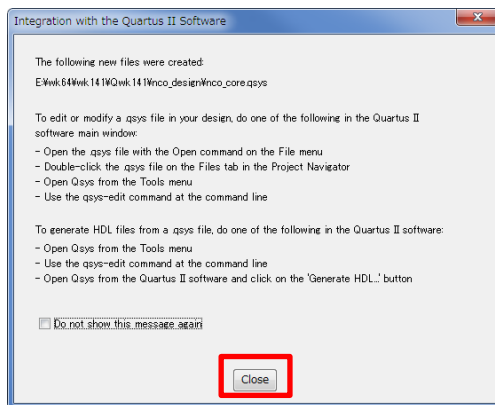
File メニュー ⇒ Exit を選択するか、画面右下の Finish ボタンをクリックして、Qsys 画面を終了します。



OR



次のような画面が起動したら、Close や OK をクリックします。



この操作により、submodule ディレクトリ内に、シミュレーション用のファイルが生成されます。インスタンス名(この資料では、nco_core を適用)の後ろに接尾語 _nco_ii_0 が付加されている点に注意してください。

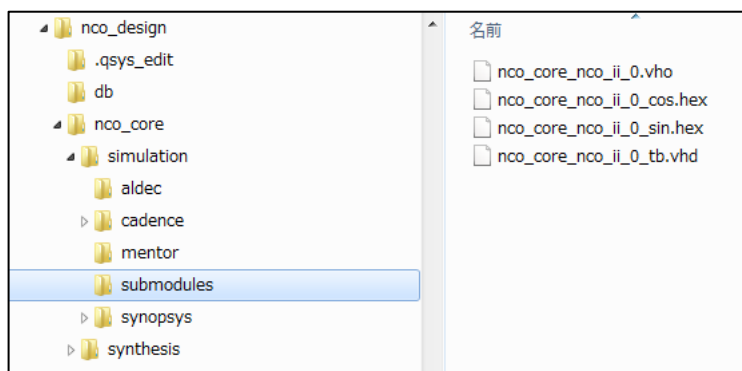
(Quartus II プロジェクト・ディレクトリ)

└ (NCO インスタンス名) ディレクトリ

└ simulation ディレクトリ

└ submodule ディレクトリ

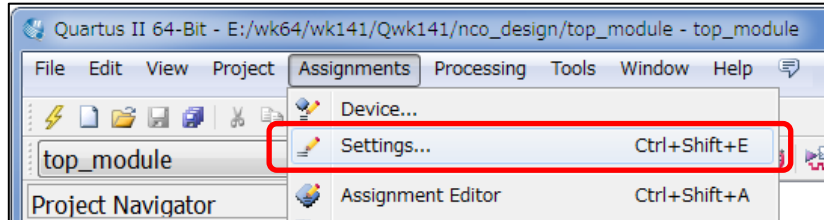
- └ (NCO インスタンス名)_nco_ii_0 .vho ファイル
- └ (NCO インスタンス名)_nco_ii_0_cos .hex ファイル
- └ (NCO インスタンス名)_nco_ii_0_sin .hex ファイル
- └ (NCO インスタンス名)_nco_ii_0_tb.vhd ファイル



3. シミュレーション手順

3-1. シミュレーションの設定

Quartus II 開発ソフトウェアの Assignments メニュー ⇒ Settings を選択します。

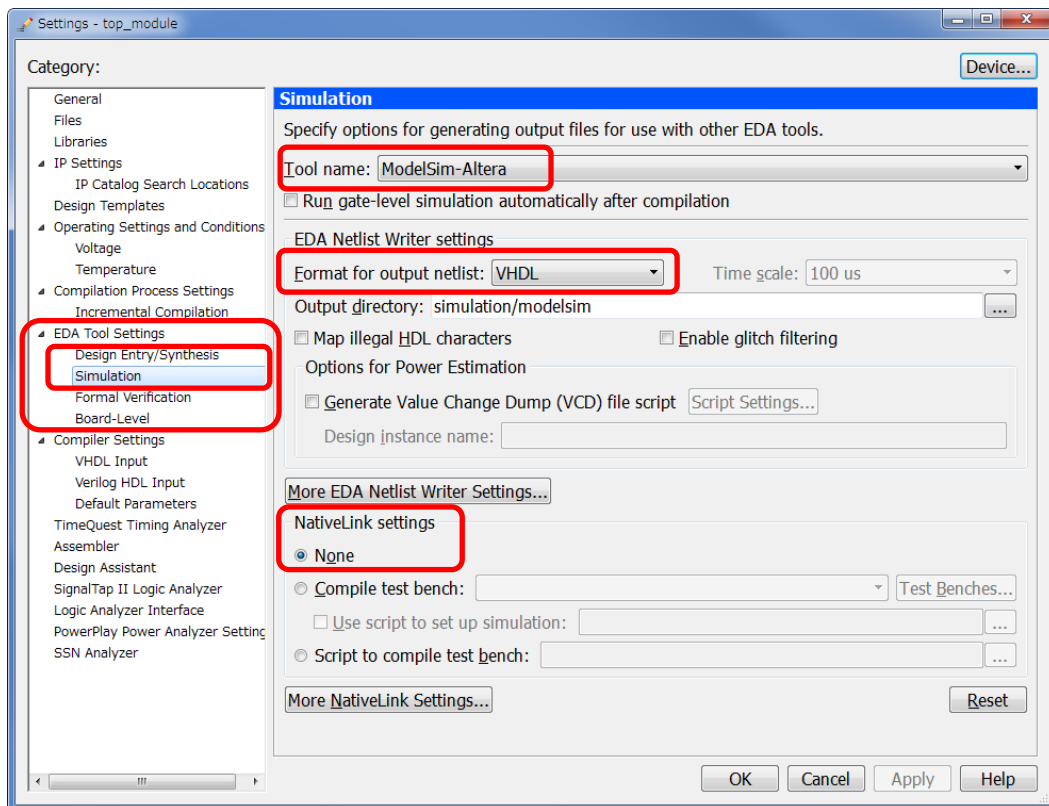


Settings 画面が起動したら、左側の Category 欄から EDA Tool Settings フォルダの Simulation フォルダを選択した後、右欄の Simulation 画面にて、以下の設定が行われているか確認します。

Tool name: ModelSim-Altera を選択

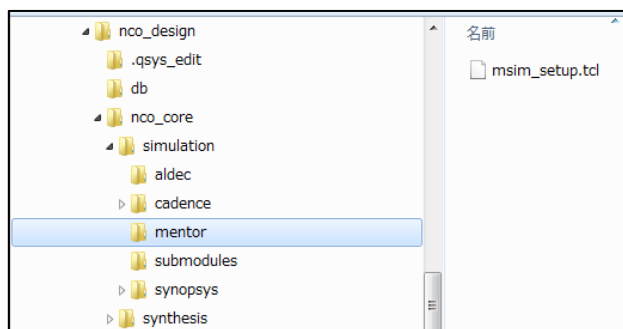
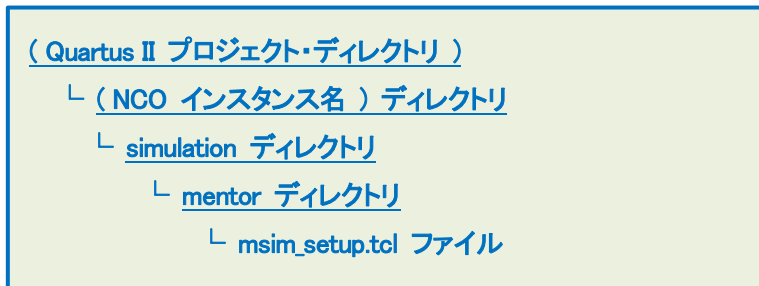
EDA Netlist Writer settings 欄の Format for output netlist: VHDL を選択

NativeLink settings 欄: None にチェック



3-2. msim_setup.tcl の修正

この資料の執筆時点では、mentor ディレクトリにある msim_setup.tcl を修正する必要があります。修正は、29 行目と 110 行目の 2 箇所あります。



29 行目

変更前

```
set TOP_LEVEL_NAME "nco_core"
```

変更後

```
set TOP_LEVEL_NAME "nco_core_nco_ii_0_tb"
```

110 行目

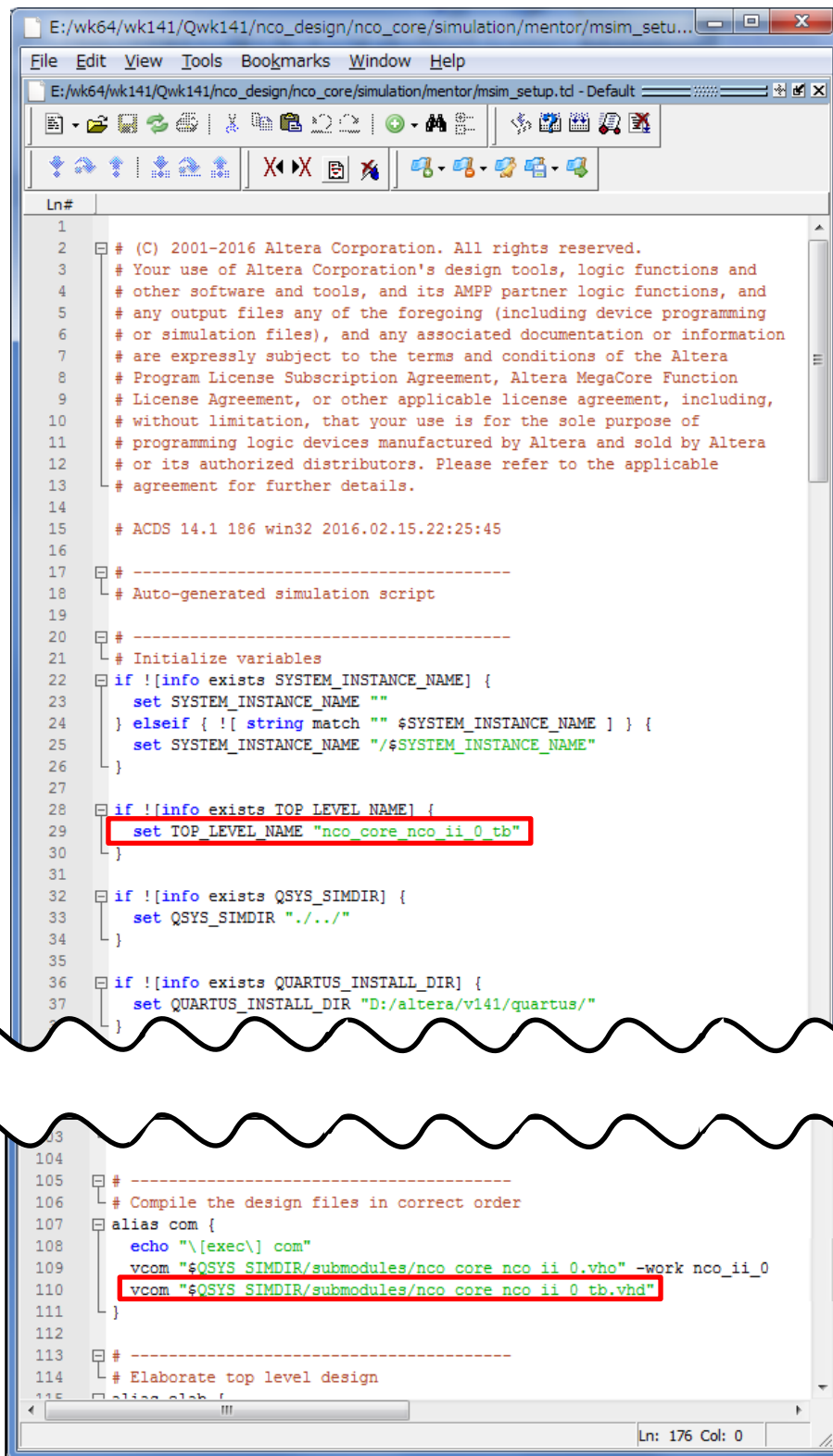
変更前

```
vcom "$QSYS_SIMDIR/nco_core.vhd"
```

変更後

```
vcom "$QSYS_SIMDIR/submodules/nco_core_nco_ii_0_tb.vhd"
```

修正後の msim_setup.tcl は、次のようになります。

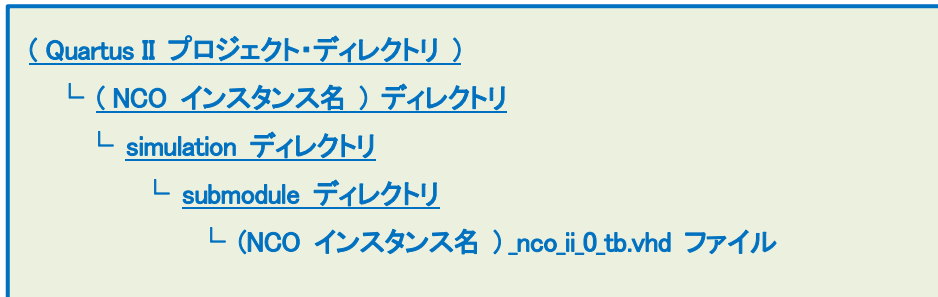


```

1
2 # (C) 2001-2016 Altera Corporation. All rights reserved.
3 # Your use of Altera Corporation's design tools, logic functions and
4 # other software and tools, and its AMPP partner logic functions, and
5 # any output files any of the foregoing (including device programming
6 # or simulation files), and any associated documentation or information
7 # are expressly subject to the terms and conditions of the Altera
8 # Program License Subscription Agreement, Altera MegaCore Function
9 # License Agreement, or other applicable license agreement, including,
10 # without limitation, that your use is for the sole purpose of
11 # programming logic devices manufactured by Altera and sold by Altera
12 # or its authorized distributors. Please refer to the applicable
13 # agreement for further details.
14
15 # ACDS 14.1 186 win32 2016.02.15.22:25:45
16
17 # -----
18 # Auto-generated simulation script
19
20 # -----
21 # Initialize variables
22 if ![info exists SYSTEM_INSTANCE_NAME] {
23     set SYSTEM_INSTANCE_NAME ""
24 } elseif { ![ string match "" $SYSTEM_INSTANCE_NAME ] } {
25     set SYSTEM_INSTANCE_NAME "/$SYSTEM_INSTANCE_NAME"
26 }
27
28 if ![info exists TOP_LEVEL_NAME] {
29     set TOP_LEVEL_NAME "nco_core nco ii_0_tb"
30 }
31
32 if ![info exists QSYS_SIMDIR] {
33     set QSYS_SIMDIR "../.."
34 }
35
36 if ![info exists QUARTUS_INSTALL_DIR] {
37     set QUARTUS_INSTALL_DIR "D:/altera/v141/quartus/"
38 }
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105 # -----
106 # Compile the design files in correct order
107 alias com {
108     echo "[exec\] com"
109     vcom "$QSYS_SIMDIR/submodules/nco_core nco ii_0.vho" -work nco_ii_0
110     vcom "$QSYS_SIMDIR/submodules/nco_core nco ii_0 tb.vhd"
111 }
112
113 # -----
114 # Elaborate top level design
115 alias elab {
116     com
117 }
    
```

3-3. テストベンチの修正

この資料の執筆時点では、submodules ディレクトリにあるテストベンチ <NCO インスタンス名>_nco_ii_0_tb.vhd を修正する必要があります。この資料では、nco_core_nco_ii_0_tb.vhd がテストベンチに該当します。



デフォルトでは、Frequency タブのパラメータ Clock Rate 欄で、100.0 MHz のクロック周波数を与えています。ところが、ツールが自動生成したテストベンチでは、100.0 MHz とは異なる値で、クロック周波数が定義されているので、次のように、半クロック周期の値を修正して、100 MHz のクロックを与えます。

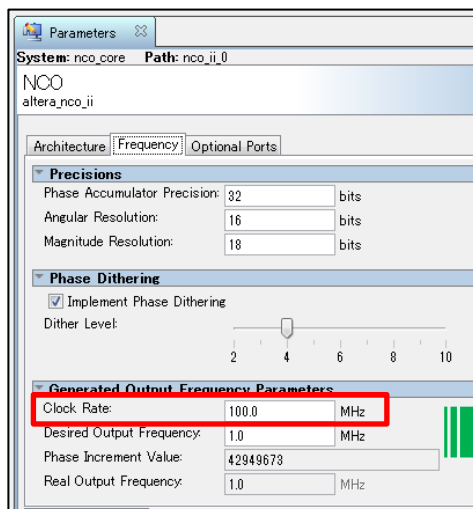
64 行目

変更前

```
constant HALF_CYCLE : time := 5000000 ps;
```

変更後

```
constant HALF_CYCLE : time := 5000 ps;
```

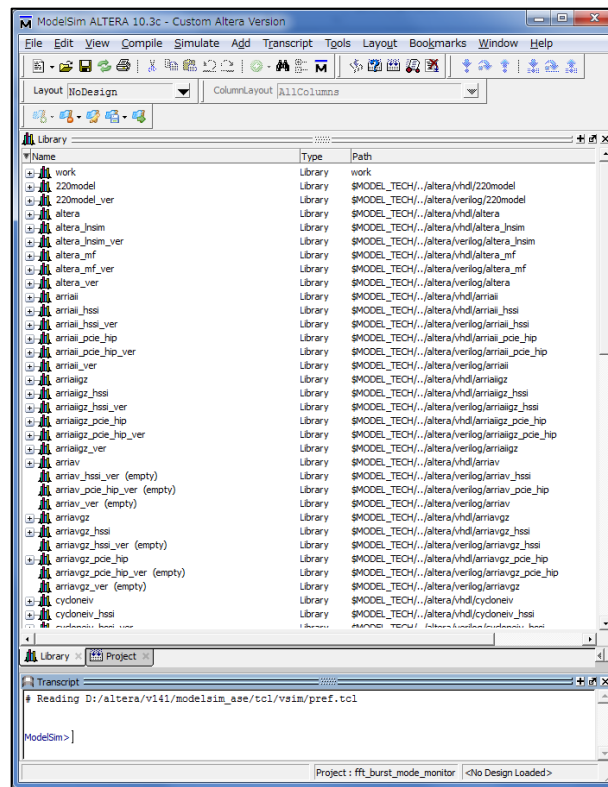


```

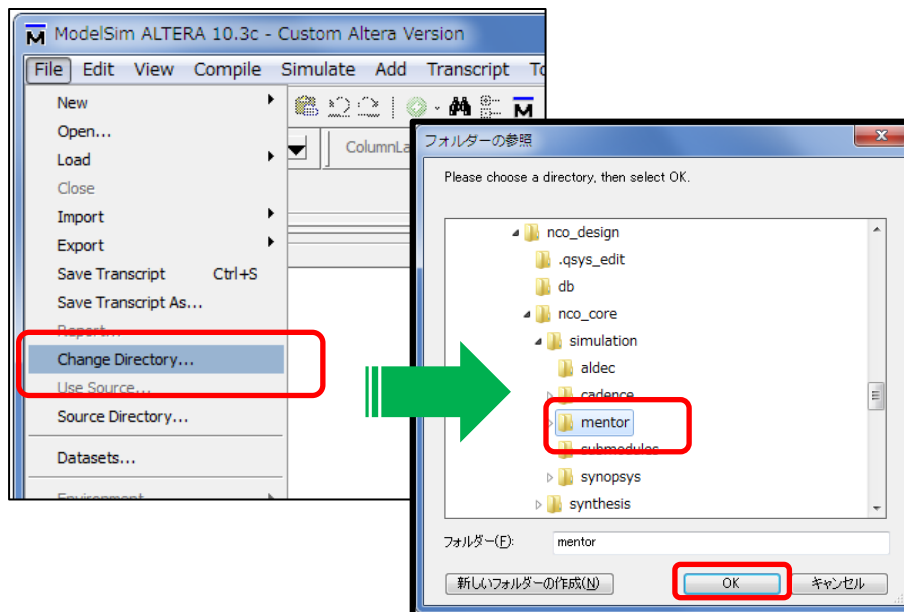
Ln#
44 port(
45   phi_inc_i   : IN STD_LOGIC_VECTOR (APR-1 DOWNT0 0);
46   clken       : IN STD_LOGIC ;
47   clk         : IN STD_LOGIC ;
48   reset_n    : IN STD_LOGIC ;
49   fsin_o     : OUT STD_LOGIC_VECTOR (MPR-1 DOWNT0 0);
50   fcos_o     : OUT STD_LOGIC_VECTOR (MPR-1 DOWNT0 0);
51   out_valid   : OUT STD_LOGIC
52 );
53 end component;
54
55 signal clk       : std_logic;
56 signal reset_n  : std_logic;
57 signal clken    : std_logic;
58 signal sin_val  : std_logic_vector (MPR-1 downto 0);
59 signal cos_val  : std_logic_vector (MPR-1 downto 0);
60 signal phi      : std_logic_vector (APR-1 downto 0);
61 signal sel_phi  : std_logic_vector (2 downto 0);
62 signal sel_output : std_logic_vector (2 downto 0);
63 signal out_valid : std_logic;
64 constant HALF_CYCLE : time := 5000 ps;
65 constant CYCLE      : time := 2*HALF_CYCLE;
66
67
    
```

3-4. シミュレーション手順

ModelSim を起動します。起動したら、次のような画面が表示されます。



ModelSim の File メニュー ⇒ Change Directory を選択して、Quartus II プロジェクト・フォルダから、mentor ディレクトリを選択して、OK ボタンをクリックします。



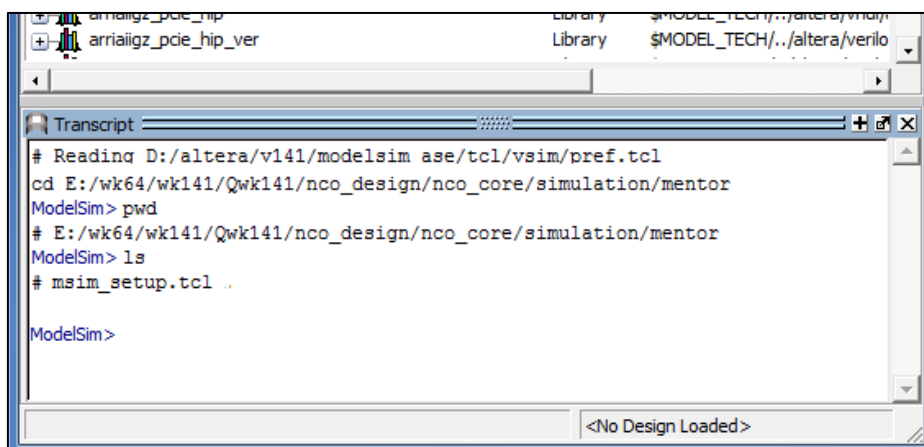
(Quartus II プロジェクト・ディレクトリ)

└ (NCO インスタンス名) ディレクトリ

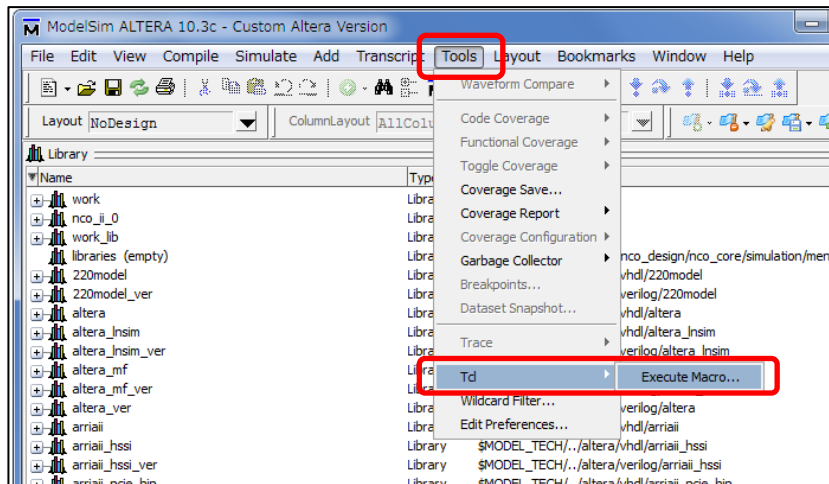
└ simulation ディレクトリ

└ mentor ディレクトリ

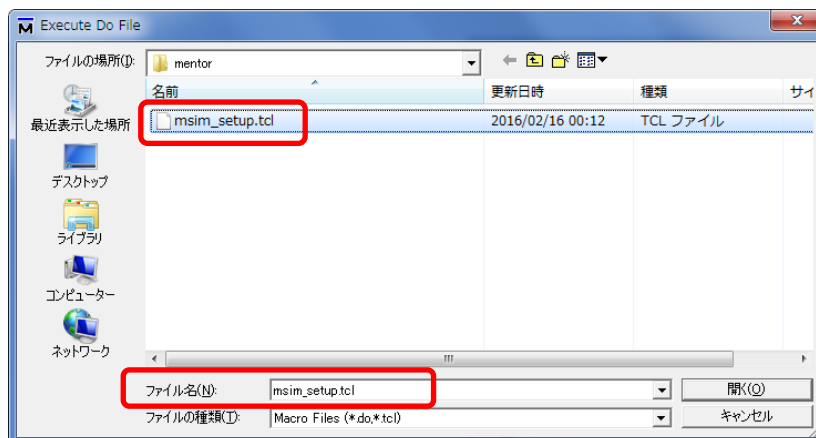
pwd コマンドをタイプして、カレント・ディレクトリが、上記で選択したディレクトリにいることを確認したら、ls コマンドをタイプして、msim_setup.tcl ファイルが見えていることを確認します。



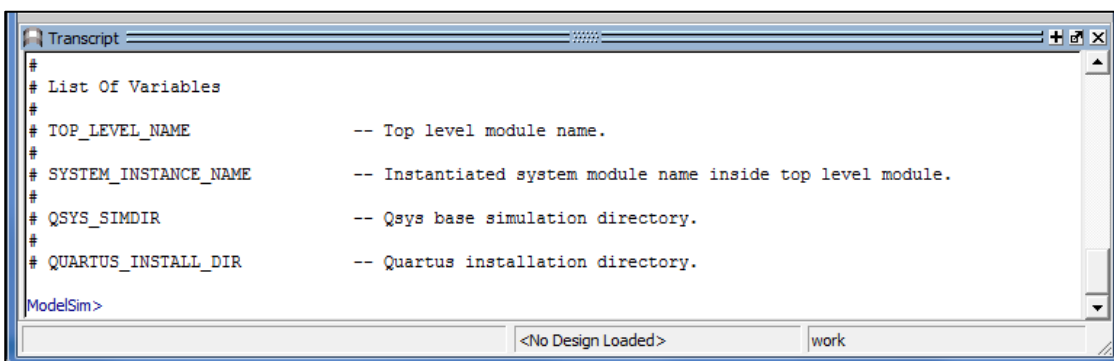
Tools メニュー ⇒ Tcl ⇒ Execute Macro を選択します。



前述の、msim_setup.tcl ファイルを選択します。



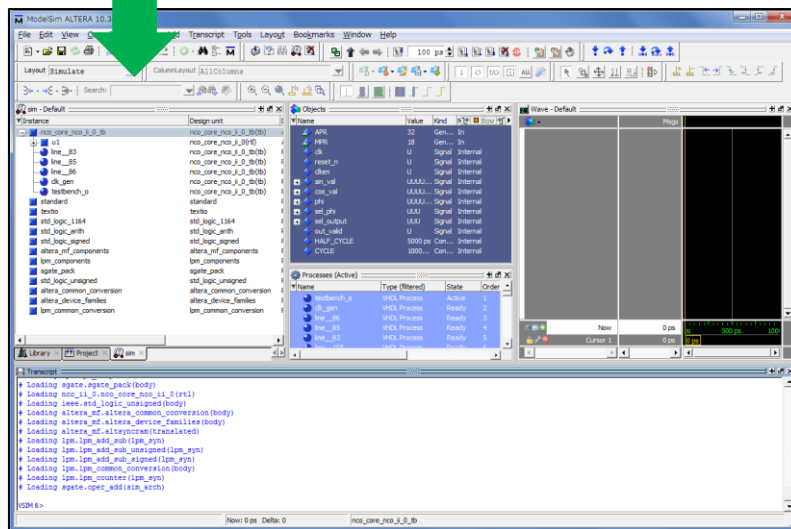
Tcl スクリプトが実行された後、ModelSim> プロンプトが表示されます。



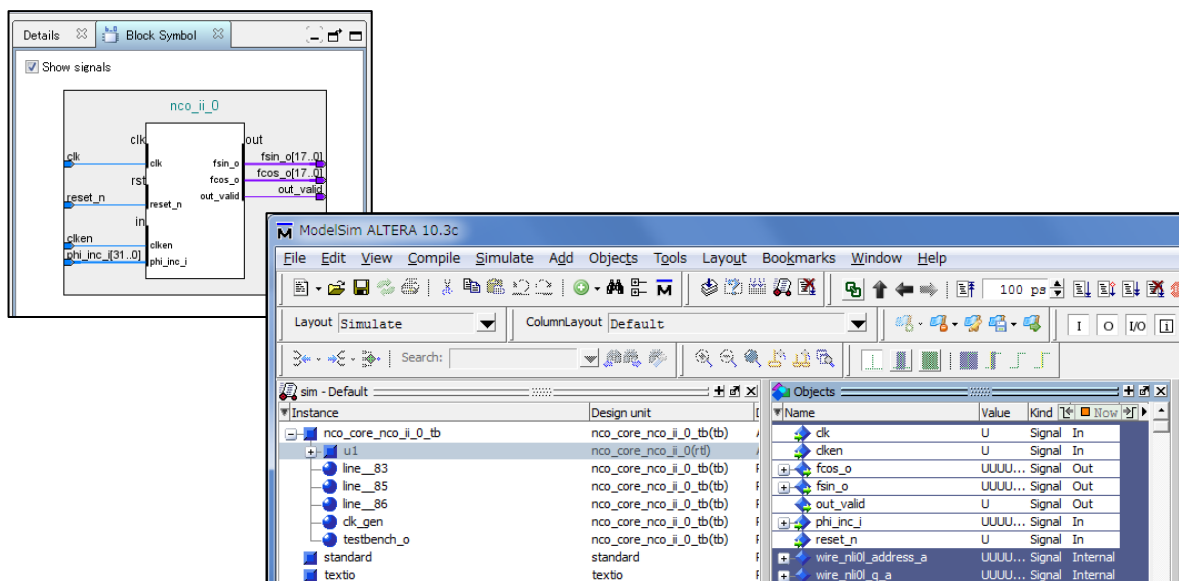
ld_debug とタイプします。この操作により、コンパイル ⇒ ロード が自動的に行われます。

```

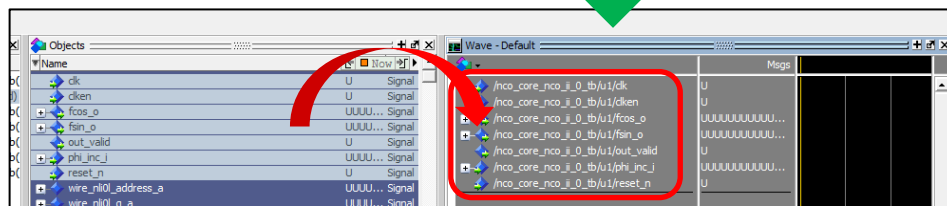
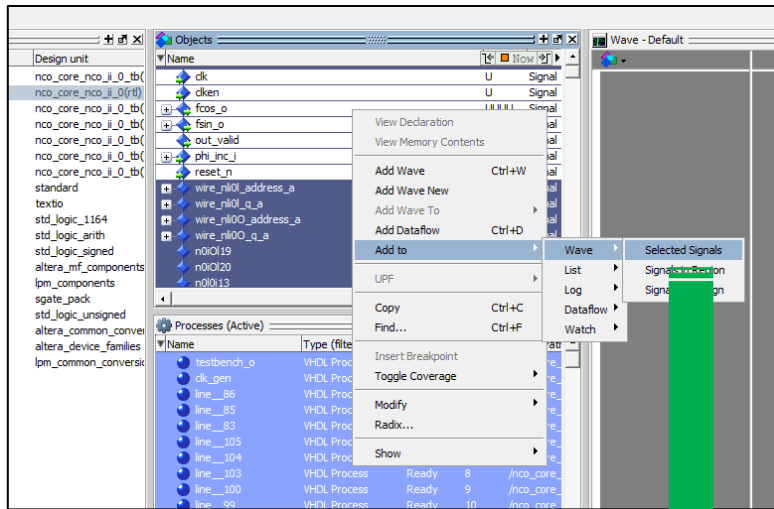
Transcript
#
# TOP_LEVEL_NAME          -- Top level module name.
#
# SYSTEM_INSTANCE_NAME    -- Instantiated system module name inside top level module.
#
# QSYS_SIMDIR             -- Qsys base simulation directory.
#
# QUARTUS_INSTALL_DIR     -- Quartus installation directory.
ModelSim: ld_debug
    
```



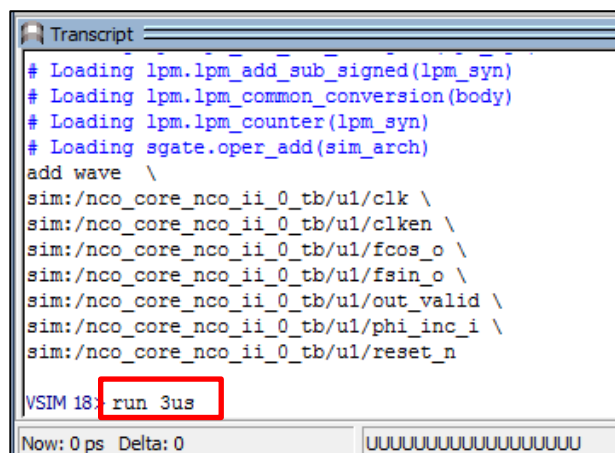
前述のシンボル情報を参考にして、観測対象の信号をハイライトします。




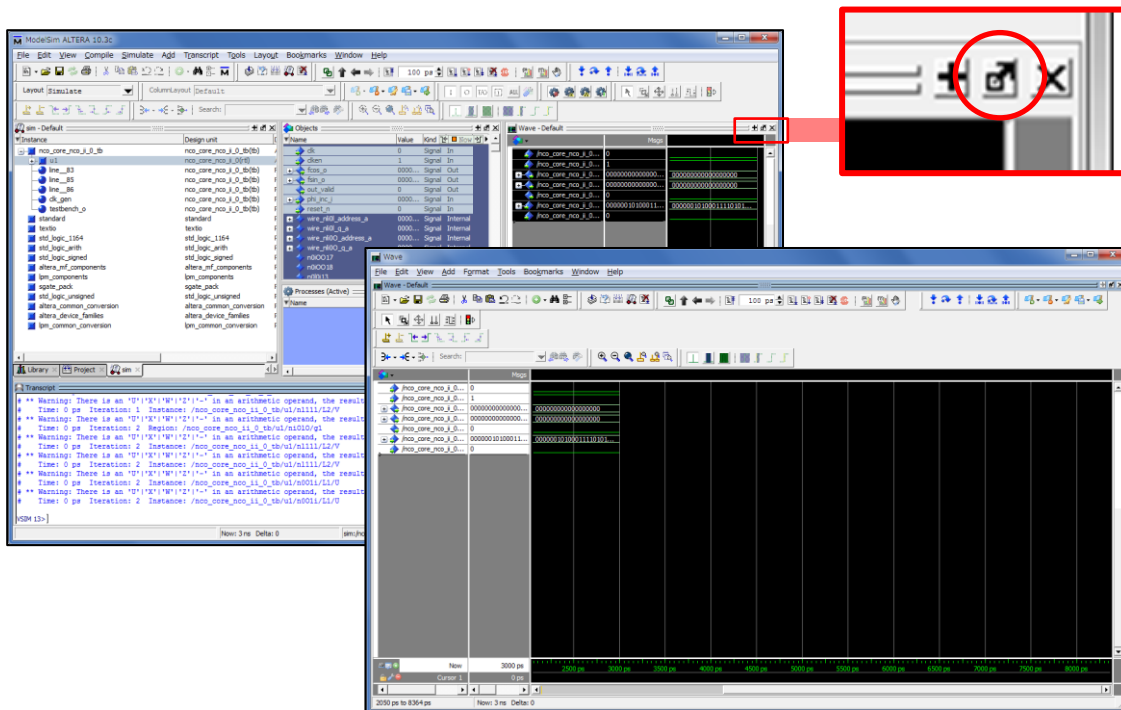
右クリックを行い、Add to ⇒ Wave ⇒ Selected Signals を選択すると、Wave 画面にハイライトした観測対象の信号が移動されます。




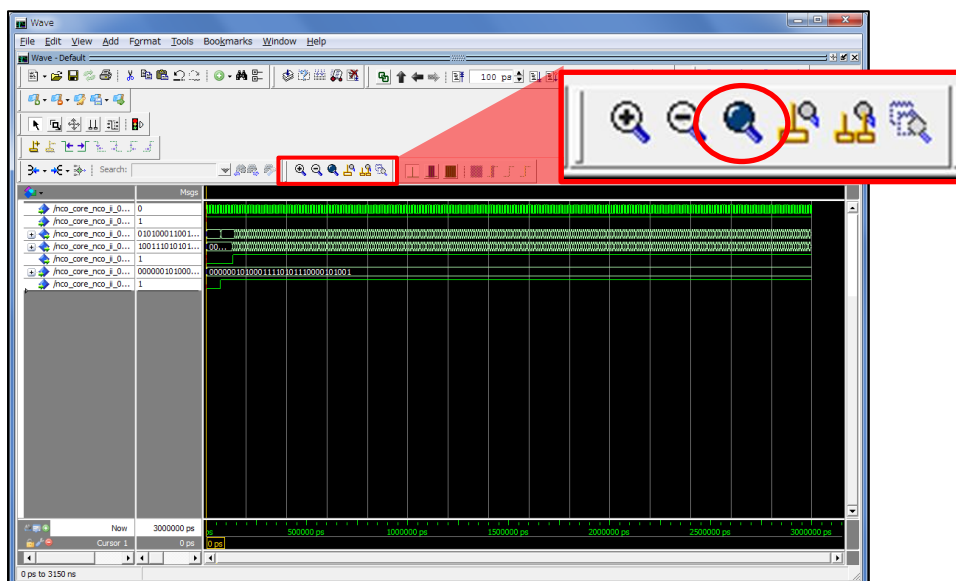
この資料では、例として、3us のシミュレーションを実行しますので、run 3 us とタイプします。




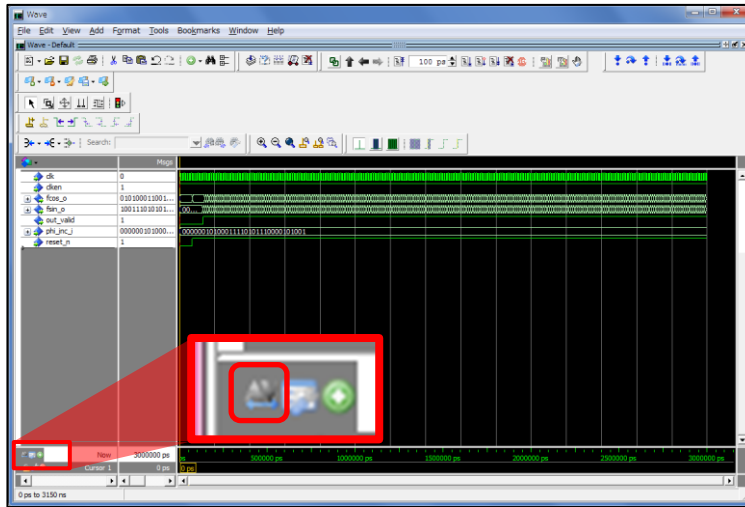
ModelSim の Wave 画面の右上の  アイコンをクリックして、Wave 画面を前面に拡大します。(Wave 画面は、ツールの使用履歴によっては既に全面に出ているケースもあります。)



Wave 画面内の  アイコンをクリックすると、画面内に実行したシミュレーション時間全体の波形が表示されます。

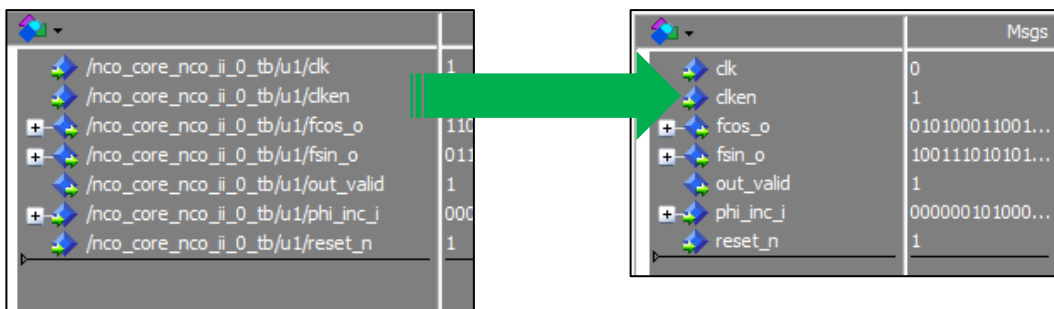


Wave 画面内の  アイコンをクリックすると、パスの表示が省略されて、選択した信号線が見やすくなります。

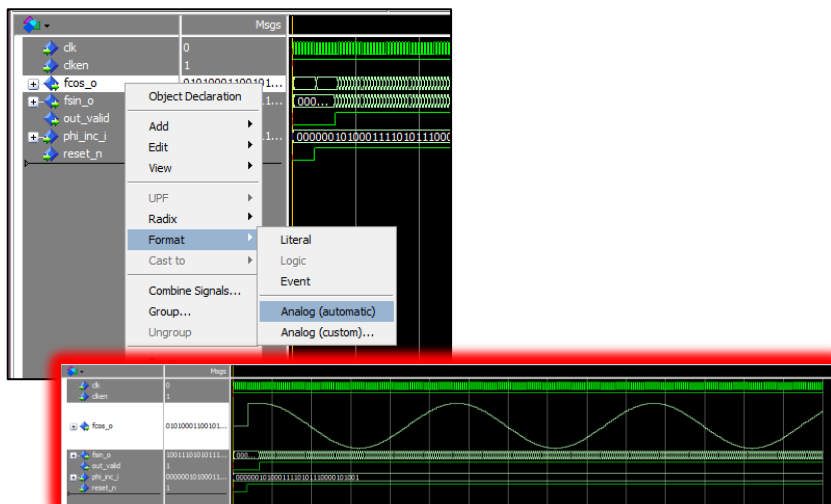


クリック前

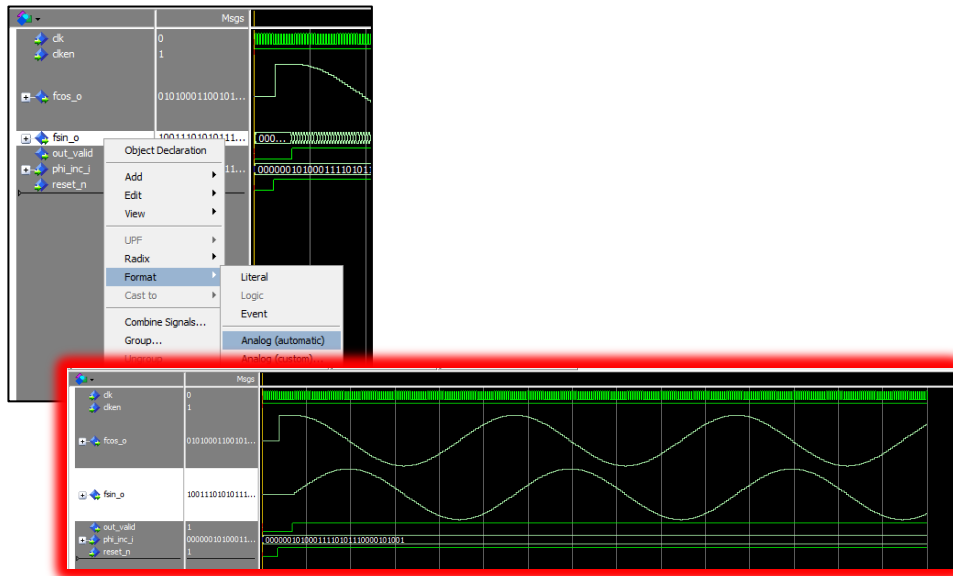
クリック後



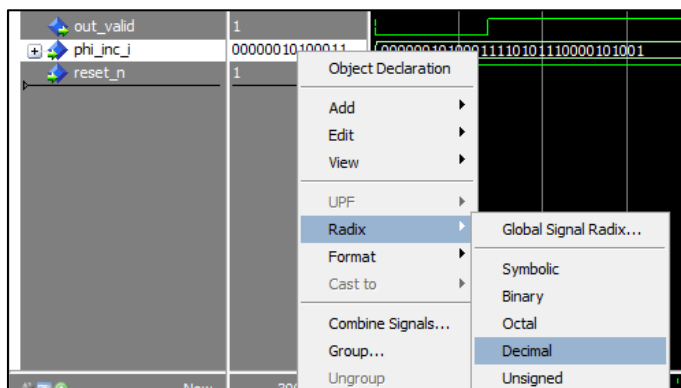
fcos_o をハイライトして、右クリック ⇒ Format ⇒ Analog (automatic) を選択すると、シミュレーション波形がアナログ風にプロットアウトされて表示されます。



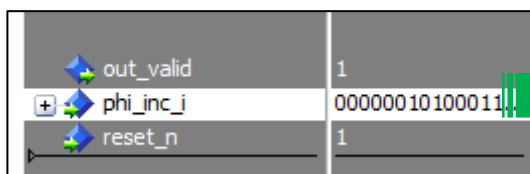
同様に、fsin_o をハイライトして、右クリック ⇒ Format ⇒ Analog (automatic) を選択すると、シミュレーション波形がアナログ風にプロットアウトされて表示されます。



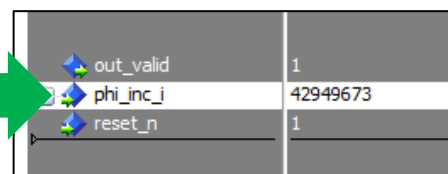
phi_inc_i をハイライトして、右クリック ⇒ Radix ⇒ Decimal を選択すると、値が 10 進数で表示されます。



デフォルト

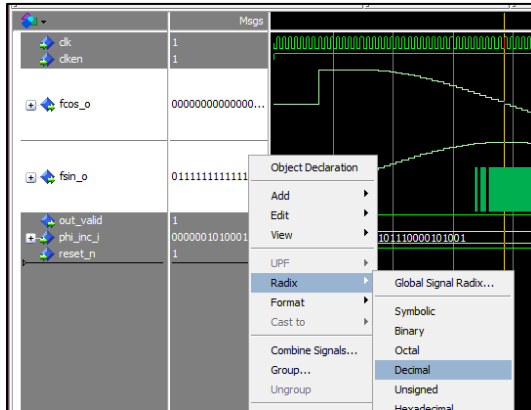


10 進数表示

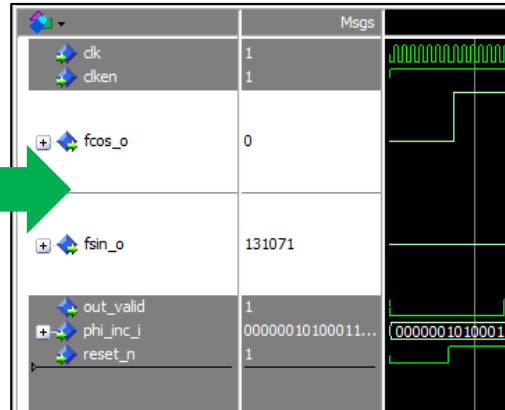


同様の操作を fcos_o と fsin_o に対しても行います。

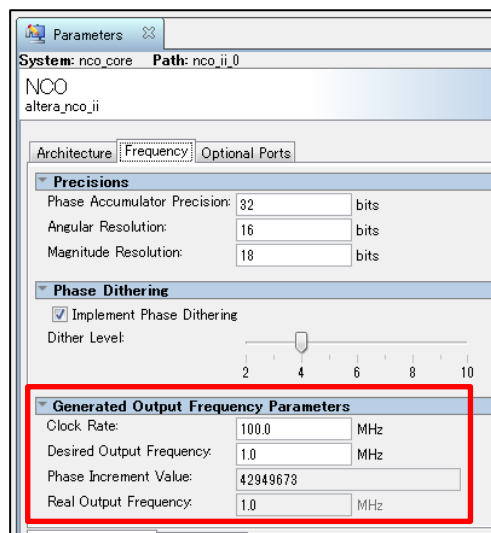
デフォルト



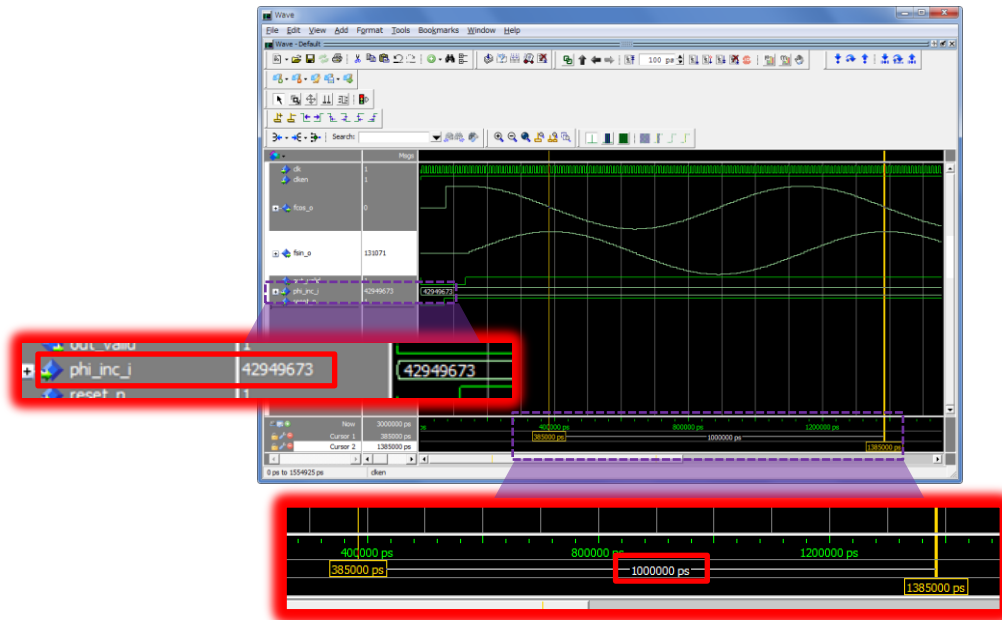
10進数表示



デフォルトでは、Frequency タブのパラメータ Clock Rate 欄で指定した 100.0 MHz のクロック周波数を与えて、NCO (数値制御発振器) から 1 MHz の正弦波および余弦波を生成するように、Desired Output Frequency 欄に与えています。これらの値を元にして、ツール側では、位相増加量および、実際に出力される正弦波および余弦波の周波数を計算します。前者の位相増加量の計算結果は、Phase Increment Value 欄に表示され、後者の正弦波および余弦波周波数の計算結果は、Real Output Frequency 欄で表示されます。これらは、グレイアウトで表示されるので、ユーザ側で直接値を変更することはできません。



シミュレーション結果として、phi_inc_i 信号に位相増加量 Phase Increment Value と同じ 42,949,673 が与えられ、正弦波が fsin_o 信号から表示されていることが視覚的に確認できます。また、その周期が 1,000,000 ps で出力されていることも確認できます。周期が 1,000,000 ps なので、周波数は、その逆数の 1MHz になります。よって、シミュレーションの結果が、設定内容と一致していることが把握できます。



このように、NCO(数値制御発振器)の動作を確認する際、アルテラ提供の自動生成テストベンチを使用したシミュレーションを利用することで、各種パラメータと動作結果との関係性を把握することができます。

改版履歴

Revision	年月	概要
1	2016年3月	初版

免責およびご利用上の注意

弊社より資料を入手されましたお客様におかれましては、下記の使用上の注意を一読いただいた上でご使用ください。

1. 本資料は非売品です。許可無く転売することや無断複製することを禁じます。
2. 本資料は予告なく変更することがあります。
3. 本資料の作成には万全を期していますが、万一ご不明な点や誤り、記載漏れなどお気づきの点がありましたら、本資料を入手されました下記代理店までご一報いただければ幸いです。
 株式会社アルティマ ホームページ: <http://www.altima.co.jp> 技術情報サイト EDISON: <https://www.altima.jp/members/index.cfm>
 株式会社エルセナ ホームページ: <http://www.elsena.co.jp> 技術情報サイト ETS : <https://www.elsena.co.jp/elspear/members/index.cfm>
4. 本資料で取り扱っている回路、技術、プログラムに関して運用した結果の影響については、責任を負いかねますのであらかじめご了承ください。
5. 本資料は製品を利用する際の補助的な資料です。製品をご使用になる際は、各メーカー発行の英語版の資料もあわせてご利用ください。