

# NCO MegaCore ファンクション のシミュレーション手順

ver.14



2016年3月 Rev.1

ELSENA,Inc.





# NCO MegaCore ファンクションのシミュレーション手順 ver. 14

# <u>目次</u>

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. <b>シミュレーションの</b> 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. <u>はじめに</u>

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<sup>®</sup> 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. <u>開発ツールおよびデザイン例</u>

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<sup>®</sup> II 開発ソフトウェア 14.1.0
- ・ シミュレーション・ツール: ModelSim<sup>®</sup>-Altera<sup>®</sup> 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 <sup>®</sup> 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。

					_			
💱 Quartus II 64-Bit - E:/wk64/wk141/Qw	Quartus II 64-Bit - E://wk64/wk141/nco_design/top_module - top_module							
Eile Edit View Project Assignments	ile Edit View Project Assignments Processing Iools Window Help 即							
🖉 🗋 💕 🗐 🗿 👗 🖻 🛍 🔊 (°								
top_module 🔹 🕅	/ _/ 🏈 🥸 🚸 💷 🕨 🛷 9	s 🙆 🛈 🕅	2 🗞 🥥 🐌 🚣 🥘 🛡					
Project Navigator	1 8 ×	12	top_module.bdf	×				
0,	×	📑 🕟 🔍	○ A ▷ \\$ • □ □ □ □ □ \ < \ < □ ○	> > 📲 🕂 🕼 🗐				
Cyclone V: 5CGXFC5C6F27C7								
▶ top_module								
< III	•							
🔺 Hierarchy 📋 Files 🧬 Desig	an Units 🛛 🖄 IP Comr 🔶							
Status	1.8×							
Module % Progress (a)	Time							
この資料では、便冝上、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								
	シューレーションだけた宇族する場合、 のいっせい H プロジェクト 佐成け 不更です							
	シミュレーションにいて天地。る場合、Quattus II ノロシェクト作成は不安です。							

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。

👃 IP Parameter Editor - unnamed.qsys* (E:	¥wk64¥wk141¥Qwk141¥nco_design¥unnamed.qsys)		- • ×
File Edit System Generate View Tools H	elp		
🙀 Parameters 🐰		📑 Details 🛛	Block Symbol 🛛 🗕 🗗 🗖
System: unnamed Path: nco_ii_0			
NCO	Detaile	NCO	â
altera_nco_ii	Details	neo	
Architecture Frequency Optional Ports	î	Name	altera_nco_li
* Base Parameters		Version	14.1
Generation Algorithm: Large ROM -		Author	Altera Corporation
Outputs: O Single Output	👃 New IP Variation	tion	Altera Numerically Controllec
Dual Output	Your IP settings will be saved in a cases file.		DSP/Signal Generation
Number of Bands: 1	Greate IP Variation	heet	http://www.altera.co.uk/literat
V Use dedicated multipliers		leet	mtp.//www.anera.co.donterat
Device Family. Oyclone V	Entity name: Innamed	ectu	re 🗸
* Throughout	Save in folder: E:#wk.64#wk.141#Q.wk.141#nco_design		m +
Clock cycles per output: 1	Target Device		X < >
	Family: Cyclone V	- 2	
	Device: 5CGXFC5C6F27C7	•	
	Take Very ID will be exceed in Direction (1419/Oct. 1419/Oct. 1419/	nco_ii_	x
Frequency Domain Time Domain	The four in white saved in C+WK0+WK1+1+GWK1+1+Regularity unnamed days.		
Magnitude (dB)		ew to	create a preset.
		OK ets for	NCO 14.1
-40			
8= Manager 52			
Type Path Message			
		Apply U	pdate Delete New
0 Errors, 0 Warnines			Generate HDL. Finish

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。

Í	👃 New IP Va	ariation	X
	Your IP settin	gs will be saved in a qsys file.	
	Create IP V	ariation	
	Entity name:	hco_core	
1	Save in folder	E¥wk64¥wk141¥Qwk141¥nco_design	
	Target Devi	ce	
	Family:	Cyclone V	•]
	Device:	5CGXFC5C6F27C7	•
	📐 Warning: " 🕕 Info: Your	The file <b>nco_core.qsys</b> already exists. IP will be saved in E¥wk64¥wk141¥Qwk141¥nco_design/ <b>nco_core.qsys</b> .	OK

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。

ore.qsys)
💠 Block Symbol 🔅 🗕 🖬
altera_nco_ii
14.1
Altera Corporation
tion Altera Numerically Contro
DSP/Signal Generation
eet http://www.altera.co.uk/lite
cture
III
₩ < :
~
- =
ico_ii_0
w to create a precet
w to create a preset.
ts for NCO 14.1
Update Delete New

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。

Base Parameters Generation Alcorithms Large ROM Outputs Single Output Dual Output Unuber of Channels: U Use dedicated multipliers Device Family: Cyclone V Throughput Clock, cycles per output	Architecture Frequer Prese Accumulator Angular Basel Line Magnitud Resolution Magnitud Resolution Phase Dithering () Implement Phase Dither Level Clock Rate: Desired Output Free	ncy Optio Precision: on: se Ditherina	32 16 18 5 2 4	bits bits bits						
Generation Algorithm: Large ROM  Output:	Architecture Frequer Precisions Phase Coundlator Magnitud Resolution Magnitud Resolution Phase Dithering Vinplement Phase Dither Level Clock Rate: Desired Output Free	ncy Optio Precision: on: se Ditherin ut Freque	anal Ports 32 16 18 5 2 4	bits bits bits						
Outputs: Single Output © Dual Output Number of Channels 1 Number of Bands: 1 ? Use dedicated multipliers Device Family: Cyclone V Throughput Clock cycles pr output 1 Clock cycles pr output 1 	Architecture Frequer Precisions Phase Accumulator Angule Statistical Phase Dithering Phase Dithering P	ncy Optic Precision: on: se Dithering ut Freque	32 16 18 2 4	bits bits bits 6 8						
Dual Output  Number of Channels:     1  Use dedicated multipliers  Device Family:     Cyclone V =  Throughput  Clock cycles per output  Architecture Frequency Modulation Input  enum	Procisions     Phase Accumulator     Angular Brack down Magnitur Resolutio     Phase Dithering     P	Precision: on: se Ditherina	32 16 18 2 4	bits bits bits						
Number of Channels: 1 Number of Bande: 1 Use dedicated multipliers Device Family:  Optional Ports  Architecture Frequency Modul  Frequency Modul  Frequency Modulation Input	Precisions     Phase Accumulator     Angular Deschation     Magnitud Resolution     Phase Dithering     Implement Phase     Dither Level     Generated Outpo     Clock Rate:     Desired Output Free	Precision: on: se Ditherine <b>ut Freque</b>	82 16 18 2 4	bits bits bits 6 8						
Number of Bands: 1  Use dedicated multipliers Device Frequency Modul	Angular Deschain Maenitur Resolution Phase Dithering Implement Phase Dither Level Clock Rate: Desired Output Free	on: se Ditherin; <b>ut Frequ</b> e	16 18 2 4	bits bits 6 8						
Use dedicated multipliers Device Family: Opcione V +  Throughput Clock cycles per output  Architecture Frequency Module  Frequency Modulation Input	Magnitud Resolution Phase Dithering Implement Phase Dither Level Generated Output Clock Rate: Desired Output Free	on: se Dithering <b>ut Frequ</b> e	18 2 4	bits bits						
Device Family: Cyclone V - Throuchput Clock cycles per output  Architecture   Frequery y Optional Ports  Frequency Modulation Frequency Modulation Frequency	Phase Dithering     Jinplement Phase     Dither Level     Generated Output     Clock Rate:     Desired Output Free	ae Dithering ut Freque	g 2 4	6 8						
Throughput Clock cycles per output 1 Architecture Frequery Optional Ports Frequency Module Frequency Modulation Input	Phase Dithering     Jinplement Phase     Dither Level      Generated Output     Clock Rate:     Desired Output Free	se Dithering aut Freque	g  2 4	6 8	·					_
Architecture Frequery Optional Ports     Frequency Modul 1	Generated Output     Clock Rate:     Desired Output Free	e Ditherine aut Freque	2 4	6 8	10					
Architecture Frequer y Optional Ports	Generated Output     Clock Rate:     Desired Output Free	ut Freque	2 4	6 8						
Architecture   Frequery   Optional Ports       Frequency Modulation     Frequency Modulation Input	<b>Generated Outpu</b> Clock Rate: Desired Output Free	ut Freque	2 4	6 8	10					
Architecture Frequery Optional Ports	Clock Rate: Desired Output Free	ut Freque			10					
Frequency Modulation Input	Clock Rate: Desired Output Free		ency Paramete	rs						
Prequency Modulation Input	Desired Output Free		100.0	MHz						
oquer		quency:	1.0	MHz						
Modulator Resolution: 32 bits	Phase Increment Va	alue:	42949673							
Modulator Pipeline Level:	Real Output Freque	ency:	1.0	MHz						
Phase Modulation	Frequency Domain T	Time Doma	in							
Phase Modulation Input	Magnitude	e (dB)								
Modulator Resolution: 16 bits										1
	-20		:							1
	-40									1
	-60 -									
	-80 -									÷.
	-100	A	الم بدين م يرة							2
	-120 -	1 with the	WYW YYYYYW	al al and a shirt of the	WW WYWW	WWWWWWW	(YYYYYYYYYYYYYYYYY		YY YM WM	MM
Frequency Domain Time Domain	-140		grupidi	a a star a s						1911.
Magnitude (dB)	-160 -		6	12 1	3 2	4	30	36 4	2 '	48
011					Frequ	iency (MHz)				
-20										
-40										
-60										
-80										
-100			1 1 1		· · · · ·					
-120 W. MANY MANY MANY MANY MANY MANY MANY MANY	4KMMM4HWWMMWWWW	MANN +11/11	(hudder here where here where here here here	WAYNAWA	MAYN .					
-140	rutu nutu	1997	a ga ta Mirit	ad and						
-160 6 12 19	24 30		36	42	48					
v 0 12 10	Frequency (MHz)									

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。

👃 IP Parameter Editor - r	nco_core.qsys* (E:¥wk64¥wk141¥Qwk
File Edit System Gener	ate View Tools Help
Rarameters 8	Generate HDL
System: nco_cor	Generate Testbench System
	Generation     Synthesis     Synthesis files are used to compile the system in a Quartus II project
	Create HDL design files for synthesis: VHDL v Create timing and resource estimates for third-party EDA synthesis tools.
	Simulation     The simulation model contains generated HDL files for the simulator, and may include simulation-only features.     Create simulation model:     VHDL
	Allow mixed-language simulation Enable this if your simulator supports mixed-language simulation.      Output Directory
	Path: E:/wk84/wk141/Qwk141/nco_design/nco_core

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。

🛵 Generate Completed	×
AII) 😫 🔺 🕕	
Progress: Parameterizing module nco_ii_0	*
OProgress: Building connections	
Progress: Parameterizing connections	
🕕 🕕 Progress: Validating	
🕕 Progress: Done reading input file	
Info: nco_core: Generating nco_core "nco_core" for QUARTUS_SYNTH	
🕕 Info: nco_ii_0: "nco_core" instantiated altera_nco_ii "nco_ii_0"	
Info: nco_core: Done "nco_core" with 2 modules, 15 files	
🕕 Info: ip-generate succeeded.	=
Info: Finished: Create HDL design files for synthesis	-
<	- F
S Generate: completed successfully.	
Stop ]	ose

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。





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。

<u>(Quartus II プロジェクト・ディレクトリ)</u>						
└ <u>(NCO インスタンス名 ) ディレクトリ</u>						
L simulation ディレクトリ						
L <u>submodule ディレクトリ</u>						
└ (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。

🔇 Quartus II 64-Bit - E:/wk64/wk141/Qwk141/nco_design/top_module - top_module							lule
File Edit View Project	Ass	ignments	Processing	Tools	Window	Help	P
1 / D 📬 🖬 🗿 🐰 📬	<b>?</b>	Device					
top_module	2	Settings.			Ctrl+Sł	nift+E	
Project Navigator		Assignme	ent Editor		Ctrl+Sł	nift+A	

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

Settings - top_module	
Category:	Device
General Files Libraries I P Settings IP Catalog Search Locations Design Templates Operating Settings and Conditions Voltage Temperature Compilation Process Settings Incremental Compilation EDA Tool Settings Design Entry/Synthesis Simulation Formal Verification Board-Level Compiler Settings VHDL Input Verilog HDL Input Defigin Assistant SignalTap II Logic Analyzer Logic Analyzer Interface PowerPlay Power Analyzer Setting SSN Analyzer	Simulation Specify options for generating output files for use with other EDA tools. Tool name: ModelSim-Altera  Rug gate-level simulation automatically after compilation EDA Netlist Writer settings Format for output netlist: VHDL  Time scale: 100 us  Output directory: simulation/modelsim  Map illegal HDL characters  Enable glitch filtering Options for Power Estimation  Generate Value Change Dump (VCD) file script Script Settings  More EDA Netlist Writer Settings NativeLink settings  None  Compile test bench:  Use script to set up simulation:  More NativeLink Settings  More NativeLink Settings  Reset
< •	OK Cancel Apply Help

# 

#### 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。

<u>(Quartus II プロジェクト・ディレクトリ)</u> <sup>L</sup> <u>(NCO インスタンス名 ) ディレクトリ</u> <sup>L</sup> <u>simulation ディレクトリ</u> <sup>L</sup> <u>mentor ディレクトリ</u> <sup>L</sup> msim\_setup.tcl ファイル



#### <u> 29 行目</u>

#### 変更前

set TOP\_LEVEL\_NAME "nco\_core"

#### 変更後

set TOP\_LEVEL\_NAME "nco\_core\_nco\_ii\_0\_tb"

#### <u>110 行目</u>

#### 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。



### 

#### 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。

#### <u>64 行目</u>

#### 変更前

constant HALF\_CYCLE : time := 50000000 ps;

#### 変更後

constant HALF\_CYCLE : time := 5000 ps;





#### 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 ALTERA 10.3c - Custom A	ltera Version	
<u>File Edit View Compile Simulate</u>	Add Transcript Tools Layout Boo	<u>k</u> marks <u>W</u> indow <u>H</u> elp
B. ~ D & ALV BODO	〇 1 ヘーム 🖭 💳 🔜 🚓 🖄 🖄 🤅	x & <u> </u>
Layout NoDesign	olumnLayout AllColumns	<u></u>
📲 - 🚜 - 🥵 🖓 - 🥵		
Library		: ± ₫ ×
V Name	Type Path	-
The work	library work	
work	Library WOR	tora (rbd /220model
220model ver	Library MODEL_TECH/ //	stera/verilog/220model
ti altera	Library MODEL_TECH//	altera (vod /altera
ti altera Incim	Library MODEL_TECH/ //	Itera/vhd/altera Incim
altera India ver	Library MODEL_TECH/ //	litera/verilon/altera_Insim
altera mf	Library MODEL_TECH/./	ltera/verlog/artera_nstit
altera mf ver	Library \$MODEL_TECH//	ltera/verilog/altera_mf
L ditera ver	Library \$MODEL_TECH/ //	Itera/verilog/altera
	Library \$MODEL_TECH/ //	stera/veniog/arciai
arriai beel	Library MODEL_TECH/ //	ltera/vho/orriai beei
arriai besi ver	Library \$MODEL_TECH/ /:	ltera/verilon/arriaii. beei
	Library MODEL TECH/ //	ltera/verilog/arriai ocie bio
arriai poe hip ver	Library SMODEL TECH/ //	ltera/verilog/arriaii_pcie_hip
arriai ver	Library \$MODEL_TECH/ //	Itera/verilog/arriaii
	Library SMODEL TECH/. //	ltera/vbdl/arriaioz
A arrialing hesi	Library SMODEL TECH//	ltera/vhd/arriaigz hssi
L dridige issi	Library \$MODEL_TECH//	ltera/verilon/arriaiinz bssi
arrialigz noie bin	Library SMODEL TECH/ //	Itera/veniog/antige_nai
arrialigz pole hip ver	Library SMODEL TECH/ //	Itera/verilog/arriaigz_poe_np
arrialigz ver	Library \$MODEL_TECH//	Itera/verilog/arriainz
arriav	Library SMODEL TECH//	ltera/vbdl/arriav
A arriav hssi ver (empty)	Library \$MODEL_TECH//a	ltera/verilon/arriav_bssi
A arriav prie bin ver (empty)	Library \$MODEL TECH//	ltera/verilog/arriav_prie_bip
A arriav ver (empty)	Library \$MODEL TECH//	ltera/verilon/arriav
+	Library \$MODEL TECH//a	altera/vhdl/arriavoz
+-11 arriavoz hssi	Library \$MODEL TECH//a	ltera/vhd/arriavoz hssi
arriavgz hssi ver (empty)	Library \$MODEL TECH//a	ltera/verilog/arriavgz hssi
+ arriavgz pcie hip	Library \$MODEL TECH//a	ltera/vhdl/arriavgz pcie hip
arriavgz_pcie_hip_ver (empty)	Library \$MODEL_TECH//a	ltera/verilog/arriavgz_pcie_hip
arriavgz_ver (empty)	Library \$MODEL_TECH//a	ltera/verilog/arriavgz
+ 1 cycloneiv	Library \$MODEL_TECH//a	altera/vhdl/cycloneiv
+ t cycloneiv_hssi	Library \$MODEL_TECH//a	altera/vhdl/cycloneiv_hssi
🗇 🖬 eurlanaiu hani yan	Library AMODEL TEOU/ /	ltara harilaa kuudanain hari 📃 💌
•		
👖 Library 🛛 🕮 Project 🛛		4   >
C Transcript		+ # X
A Deading Dr (algens (st 41 (	and (sol (main (see 5 sol	
# Reading D:/aitera/v141/modelsin	_ase/tci/vsim/pref.tcl	-
ModelSim>		
li maccani. I		
0		<u> </u>
	Project : fft_burst_mode_n	ionitor   <no design="" loaded=""></no>

ModelSim の File メニュー  $\Rightarrow$  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。



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。

⊕ arrialigz_pde_np	Library	\$MODEL_TECH//altera/verilo
•		<u> </u>
A Transcript	****	+ ₫ ×
# Reading D:/altera/v141/modelsim ase/to	l/vsim/pref.	tcl 🔺
<pre>cd E:/wk64/wk141/Qwk141/nco_design/nco_c ModelSim&gt; pwd # E:/wk64/wk141/Qwk141/nco_design/nco_cc ModelSim&gt; ls # msim_setup.tcl</pre>	core/simulati	on/mentor n/mentor
ModelSim>		-
	<no [<="" td=""><td>Design Loaded &gt; //</td></no>	Design Loaded > //

Tools メニュー  $\Rightarrow$  Tcl  $\Rightarrow$  Execute Macro を選択します。

ModelSim ALTERA 10.3c - Custom Altera Version	_		
File Edit View Compile Simulate Add Transcr	ip: (T	ools Layout Bookm	arks Window Help
📗 • 😅 🖬 🧇 🎒   🗼 🐚 🏙 💭 🗘   🔕 - 🗛		Waveform Compare	****
Layout NoDesign 🗸 🛛 ColumnLayout All	Colu	Code Coverage	· 💌 ] 🦓 - 🖧 - 🤣 🐴 - 🥰
Library		Functional Coverage	
*Name	Type	Toggle Coverage	
	Libra	Coverage Save	
	Libra	Coverage Report	•
+-	Libra	Coverage Configuration	•
libraries (empty)	Libra	Garbage Collector	nco_design/nco_core/simulation/ment
+-1 220model	Libra	Breakpoints	vhdl/220model
220model_ver	Libra	Diedopolitis	verilog/220model
altera	Libra	Dataset Snapshot	vhdl/altera
	Libra	Trace	vhdl/altera_Insim
	Libra		verilog/altera Insim
	Linra	Td	Execute Macro
	Libra	Wildcard Filter	verilog/altera
	Libra	Edit Preferences	vhdl/arriai
	Library	\$MODEL_TECH//alter	a/vhdl/arriaii hssi
++	Library	\$MODEL_TECH//alter	a/verilog/arriaii hssi
	Library	MODEL TECH/ /alter	a/vhdl/arriaii.ncie.hin

前述の、msim\_setup.tcl ファイルを選択します。

M Execute Do File	2					X
ファイルの場所(]):	鷆 mentor		-	← 🗈 💣 📰▼		
Ca	名前	^		更新日時	種類	サイ
最近表示した場所	msim_setup.to	3		2016/02/16 00:12	TCL ファイル	
デスクトップ						
<b>(</b> ) ライブラリ						
3281-9-						
ネットワーク	•		m			F
	ファイル名(N):	msim_setup.tcl			▼ 開K(0)	
	ファイルの種類(工):	Macro Files (*.do,*.tcl)			<ul> <li></li></ul>	ıı 🔤

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。

F	Transcript ======		-+ • ×
ŧ	Tint Of Hamishing		
Ŧ	LIST OF VARIABLES		
ŧ	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.	
M	lodelSim>		<u> </u>
		<no design="" loaded=""> work</no>	

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。



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。

🔒 Transcript 💳		
+ Loading lpm lpm add sub signed (lpm sup)		
# Loading ipm.ipm_add_sdb_signed(ipm_syn)		
# Loading 1pm.1pm_common_conversion(body)		
# Loading lpm.lpm_counter(lpm_syn)		
<pre># Loading sgate.oper_add(sim_arch)</pre>		
add wave \		
sim:/nco_core_nco_ii_0_tb/u1/clk \		
sim:/nco_core_nco_ii_0_tb/u1/clken \		
sim:/nco_core_nco_ii_0_tb/u1/fcos_o \		
<pre>sim:/nco_core_nco_ii_0_tb/u1/fsin_o \</pre>		
<pre>sim:/nco_core_nco_ii_0_tb/u1/out_valid \</pre>		
sim:/nco_core_nco_ii_0_tb/u1/phi_inc_i \		
sim:/nco_core_nco_ii_0_tb/u1/reset_n		
VSIM 182 run 3us		
Now: 0 ps Delta: 0 UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU		

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 マ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 Prd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。



<u>クリック前</u>

<u>クリック後</u>



 $fcos_o$ をハイライトして、右クリック  $\Rightarrow$  Format  $\Rightarrow$  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、 $fsin_o e \wedge format \Rightarrow Analog (automatic) を選択すると、シミュレーション$ 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。

<pre>out_valid     out_inc_i</pre>	1 00000010 1	100011 Object Declarat	ion	11110101110000101001
		Add Edit	+	
		View	•	
		Radix		Global Signal Radix
		Format Cast to	•	Symbolic Binary
		Combine Signals	i	Octal
		Group		Decimal
Al 🗖 🔿 🛛 Noui	20(	Ungroup		Unsigned

#### <u>デフォルト</u>

<u>10 進数表示</u>

📥 out_valid	1	🖕 out_valid	1
🕀 📣 phi inc i	00000010100011	≥-🌧 phi_inc_i	42949673
🔥 reset n	1	set_n	1
	<u>*</u>	•	

同様の操作をfcos\_oとfsin\_oに対しても行います。



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、グレー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。

💐 Parameters 🛛		
System: nco_core Path: nco_ii_	)	
NCO		
altera_nco_ii		
Architecture Frequency Optio	nal Ports	
Precisions		
Phase Accumulator Precision:	32	bits
Angular Resolution:	16	bits
Magnitude Resolution:	18	bits
* Phase Dithering		
📝 Implement Phase Dithering	5	
Dither Level:		
	1 · Y ·	
	2 4	6 8 10
Cenerated Output Freque	ency Parameters	
Clock Rate:	100.0	MHz
Desired Output Frequency	10	MHz
Phase Increment Value:	42949673	
Real Output Frequency:	1.0	MHz

デフォルト

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。



### <u> 改版履歴</u>

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: <u>http://www.altima.co.jp</u>
   技術情報サイト EDISON:

   株式会社エルセナ
   ホームページ: <u>http://www.elsena.co.jp</u>
   技術情報サイト ETS :
  - 技術情報サイト EDISON: <u>https://www.altima.jp/members/index.cfm</u> 技術情報サイト ETS : <u>https://www.elsena.co.jp/elspear/members/index.cfm</u>

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、各メーカ発行の英語版の資料もあわせてご利用ください。