

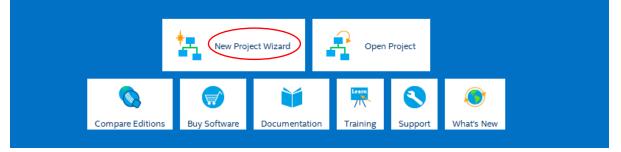
Quartus Prime – simulation

Example of configuring and running the Questa simulator from the Quartus Prime project. Two types of simulations will be performed: RTL and Gate-level.

The description of the activities provided assumes the existence of sources for synthesis (**encoder**.vhd) and testbench (**encoder_tb**.vhd).

1. Preparing the project

Launch Quartus Prime. In the start window, select New Project Wizard



Select any working directory, enter the project name and the name of the main project unit

🗑 New Project Wizard	>
Directory, Name, Top-Level Entity	
What is the working directory for this project?	
D:/semestry/Quartus/first_qsim]
What is the name of this project?	
qsim_rtl	
What is the name of the top-level design entity for this project? This name is case sensitive and must exactly match the entity name in the design file.	e
lencoder	

Choose to create a new project

S New Project Wizard	×
Project Type	
Select the type of project to create.	
Empty project	
Create new project by specifying project files and libraries, target device family and device, and EDA tool settings.	
O Project template	
Create a project from an existing design template. You can choose from design templates installed with the Quartus Prime software, o download design templates from the <u>Design Store</u> .	or

Add VHDL source files to the project. Make sure to set the appropriate file type

New Project Wizard						
Add Files						
elect the design files you war	nt to include in the projec	t. Click Add All	to add all design files in the p	roject directory to the	e project	
ote: you can always add desi	gn files to the project late	er.				
ile name:						<u>A</u> dd
<					×	Add A <u>l</u> l
ile Name	Туре	Library	Design Entry/Synthesis To	ol HDL Version	Ĩ	Remove
/first_qsim/encoder.vhd	VHDL File			VHDL_2008		_
/first_qsim/encoder_tb.v	hd VHDL Test Bench	File				Up
						<u>D</u> own
					1	Properties

2. Configuring EDA tools

Open Tools Configuration from the Tools menu

	<u>T</u> ools	<u>W</u> indow <u>H</u> elp						
	Run Sim <u>u</u> lation Tool							
🐍 Launch Simulation Library Compiler								
	🗞 Launch Design Space E <u>x</u> plorer II							
	S Timing Analyzer							
•	••							
	<u>// 1</u>	u ocijpis	- 11					
	Customize							
1	Options							
	License Setup							
	🕥 In	stall Devices						

Check the correct settings of the EDA tools, if necessary, set the appropriate path to the Questa simulator

itegory:			
General	EDA Tool Options		
EDA Tool Options Fonts	Specify the directory t	hat contains the tool executable for each third-party ED	A tool:
Headers & Footers Settings	EDA Tool	Directory Containing Tool Executable	
Internet Connectivity	Precision Synthesis		
Libraries V IP Settings	Synplify		
IP Catalog Search Locations	Synplify Pro		
Design Templates			
License Setup	Active-HDL		
Preferred Text Editor	Riviera-PRO		
Processing	ModelSim		
Tooltip Settings			
Messages	QuestaSim		

3. Simulation setup

Select project settings from the Assignments menu

:	<u>A</u> ssignments	P <u>r</u> ocessing	<u>T</u> ools	<u>W</u> indow	<u>H</u> elp		
	<u> ▶ Device</u>						
Ć	<u>S</u> ettings	\supset			Ctrl+Shift+E		
	∉ <u>A</u> ssignme	ent Editor			Ctrl+Shift+A		
E	∉ Pi <u>n</u> Planner Ctrl+Shift+N						
łC	<u>R</u> emove Assignments						
	Back-Annotate Assignments						
	Imp <u>o</u> rt Assignments						
	E <u>x</u> port Assignments						
	Assignment <u>G</u> roups						
	▲ Logic Lock Regions Window Alt+L						
	📥 Design Pa	rtitions <u>W</u> indo	w		Alt+D		

In the settings window, find the simulation parameters (1). Select the Quest Intel FPGA simulator (2) and go to the testbench configuration in the NativeLink settings group (3)

Settings - encoder	– – ×
Category:	Device/Board
General	Simulation
Files Libraries	Specify options for generating output files for use with other EDA tools.
 IP Settings IP Catalog Search Locations 	Tool name: Questa Intel FPGA 2
Design Templates • Operating Settings and Conditions	Run gate-level simulation automatically after compilation
Voltage Temperature	EDA Netlist Writer settings
 Compilation Process Settings Incremental Compilation 	Eormat for output netlist: VHDL Time scale: 100 us
✓ EDA Tool Settings	Output directory: simulation/questa
Design Entry/Synthesis	Map illegal <u>H</u> DL characters <u>E</u> nable glitch filtering
Board-Level	Options for Power Estimation
Compiler Settings VHDL Input	Generate Value Change Dump (VCD) file script Script Settings
Verilog HDL Input Default Parameters	Design instance name:
Timing Analyzer Assembler	
Design Assistant	More EDA Netlist Writer Settings
Signal Tap Logic Analyzer Logic Analyzer Interface	NativeLink settings
Power Analyzer Settings SSN Analyzer	O None
	<u>C</u> ompile test bench: <u>Test Benches</u>
	Use script to set up simulation:
	O Script to compile test <u>b</u> ench:
	More NativeLink Settings
	Weight Buy Software OK Cancel Apply Help

Select a new configuration

🚽 Test Benche	25				×
Specify setting	s for each test bench.				
Existing test b	ench settings:				<u>N</u> ew
Name	Top Level Module	Design Instance	Run For	Test Bench File(s)	Edit

Find the testbench file on the disk (1), Enter the name of the main unit defined in this file (2), Select the use of testbench also for time simulation (3), Set the simulation duration (4).

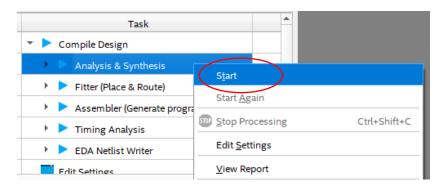
	New Test Bench S	ettings				×		
	Create new test bend	Create new test bench settings.						
	Test bench name: e	ncoder_tb 2)					
	Top level <u>m</u> odule in t	test bench: encod	er_tb					
3	✓ Use test bench to	perform VHDL tim	ning simulation					
	Design instance	name in test bench	: encoder					
	Simulation period							
	O Run simulation	n until all <u>v</u> ector sti	muli are used					
	• End simulation	n at: 1	us 🔹 4					
	Test bench and simu	lation files						
	<u>F</u> ile name:				Add			
	<u>File Name</u>	Library	HDL Version		<u>R</u> emove			
Ç	1 encoder_tb.vhd		VHDL_2008		Up			
					Down			
					Properties			

🥜 Test i	Test Benches ×							
Specify s	settings fo	r each test bench.						
Existing	test bench	settings:				<u>N</u> ew		
Na	ame	Top Level Module	Design Instance	Run For	Test Bench File(s)	<u>E</u> dit		
encoder	r_tb	encoder_tb	encoder	1 us	encoder_tb.vhd			
						<u>D</u> elete		
					OK Cancel	Help		

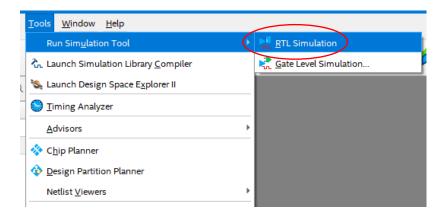
4. Starting the RTL simulation

Before performing RTL simulation, the project must be compiled.

From the Task window, from the Compile Design group, run the Analusis & Synthesis command



If the compilation was successful, run the RTL simulation from the Tools menu



As a result of this command, an external simulator will be opened and the configuration/simula-tion script defined in point 3 will be executed. The effects of the simulator operation can be observed in the "wave" time waveform window.

Verify the structure of the created project - the content of the "structure" (UUT) window should present the elements of the behavioral description from the encoder.vhd file

🗸 sim - Default 🚞				= + 🗗 ×
₹ Instance	Design unit	Design unit type	Top Category	Visibilit
 encoder_tb	encoder_tb(behav)	Architecture	DU Instance	+acc=
🚊 🗾 υυτ	encoder(with_delay)	Architecture	DU Instance	+acc=
- () line_18	encoder(with_delay)	Process	-	+acc=
p_enc	encoder(with_delay)	Process	-	+acc=
test_proc	encoder_tb(behav)	Process	-	+acc=
🗾 standard	standard	Package	Package	+acc=
🗾 textio	textio	Package	Package	+acc=
🗾 std_logic_1164	std_logic_1164	Package	Package	+acc=
🗾 std_logic_arith	std_logic_arith	Package	Package	+acc=
🗾 std_logic_unsigned	std_logic_unsigned	Package	Package	+acc=
•				
👖 Library 🗶 🎬 Project 🗶	🔊 sim 🗙			*

5. Starting the gate-level simulation

Before performing gate-level simulation, the project must pass the routing stage (Fitter – Place&Route). Then, start creating a model for simulation.

From the Task window, from the Compile Design group, run the EDA Netlist Writer command



If the netlist creation was successful, run the gate-level simulation from the Tools menu

<u>Tools</u> <u>W</u> indow <u>H</u> elp	
Run Sim <u>u</u> lation Tool	RTL Simulation
€. Launch Simulation Library <u>C</u> ompiler	Gate Level Simulation
🗞 Launch Design Space E <u>x</u> plorer II	encoder tb.vhd 🗙

As a result of this command, the external simulator will be opened and the configuration/simulation script defined in point 3 will be executed.

Verify the structure of the created project - the content of the "structure" window should present blocks of the automatically created structural model from the encoder.vho file

sim - Default				: + 2
Instance	Design unit	Design unit type	Top Category	Visib
H encoder_tb	encoder_tb(behav)	Architecture	DU Instance	+ao
ட்- 🛒 யா 🔍 🔇	encoder(structure)	Architecture	DU Instance	+ao
🗄 🚽 🖬 auto_generated_i	hard_block(structure)	Architecture	DU Instance	+ao
+- Vartus_CRE	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+ao
⊕-	fiftyfivenm_io_obuf(behavi	Architecture	DU Instance	+aci
🕂 🛒 \y[1]~output	fiftyfivenm_io_obuf(behavi	Architecture	DU Instance	+aci
+- 🗾 \y[2]~output\	fiftyfivenm_io_obuf(behavi	Architecture	DU Instance	+aci
+- 🗾 \y[3]~output\	fiftyfivenm_io_obuf(behavi	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
+- _ \s[5]~input\	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
∰- WideOr2~7\	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+aci
	fiftyfivenm_io_ibuf(behavior)	Architecture	DU Instance	+ao
🕂 🗾 WideOr2	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
₩ideOr1~3	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
🕂 🗾 WideOr 1	fiftyfivenm_lcell_comb(vital	Architecture	DU Instance	+aci
	fiftyfiveom Icell comb(vital	Architecture	DI I Instance	+ao