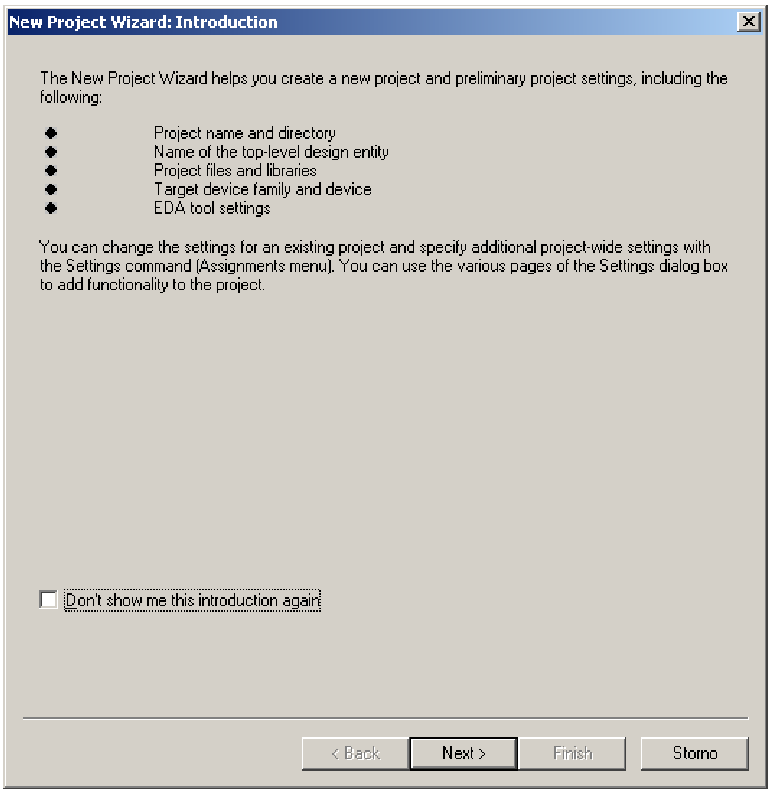
Realizácia hardvéru a softvéru pre NIOS II softcore procesor a pripojenie vlastnej periférie

Úloha: Vytvorte systém na báze hradlových polí od výrobcu Altera, v ktorom použijete softcore procesor NIOS II. Hardvérové požiadavky sú: ekonomická verzia NIOSII, 32kB RAM, JTAG, výstupný port na LED a čítač ako vlastná periféria kompatibilná s Avalon zbernicou typu „Memory Mapped Slave". Pre danú perifériu vytvorte balík základných funkcií v jazyku C pre jej ovládanie. Vlastná periféria má byť napísaná v jazyku VHDL. Ako východzí hardvér použijte „Altera Cyclone III Starter Board". Celý systém má striedavo rozblikať dve LED.

Postup riešenia:

1. Vytvorenie nového projektu v prostredí Quartus II.

   1.a File > New Project Wizard …



New Project Wizard: Introduction

The New Project Wizard helps you create a new project and preliminary project settings, including the following:

- Project name and directory
- Name of the top-level design entity
- Project files and libraries
- Target device family and device
- EDA tool settings

You can change the settings for an existing project and specify additional project-wide settings with the Settings command (Assignments menu). You can use the various pages of the Settings dialog box to add functionality to the project.

☐ Don't show me this introduction again

   1.b Next >

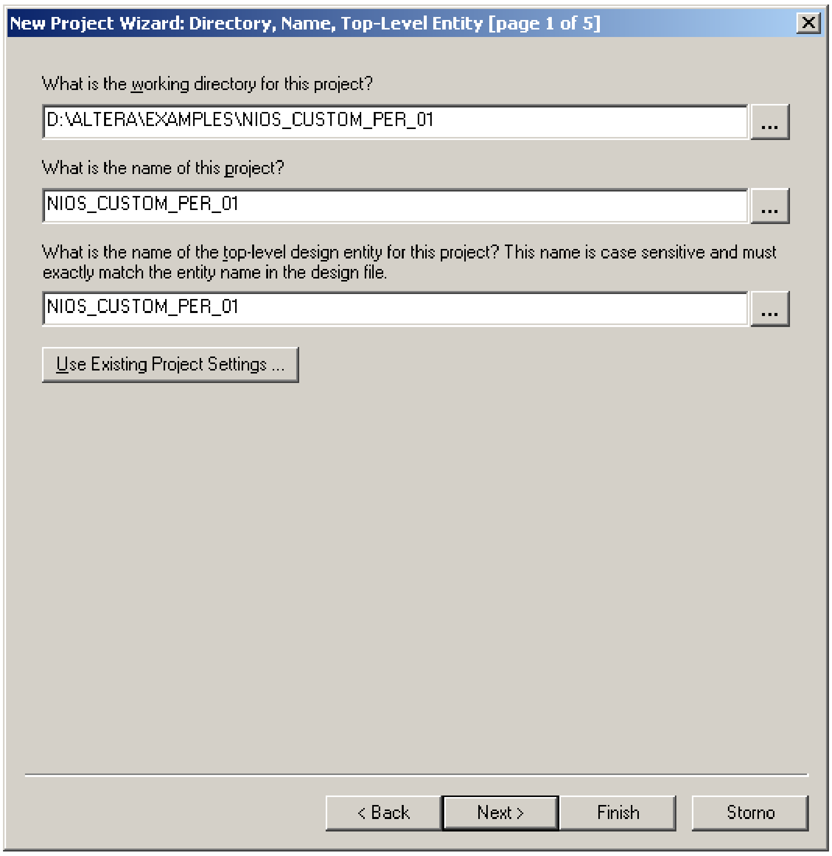
   1.c Dotazník vyplníte nasledovne:

      What is the working directory for this project:

         D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01

      What is the name of the project: NIOS_CUSTOM_PER_01

      Top level entity: NIOS_CUSTOM_PER_01



New Project Wizard: Directory, Name, Top-Level Entity [page 1 of 5]

What is the working directory for this project?

D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01

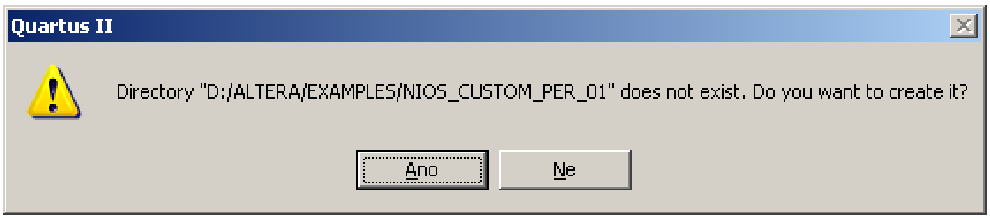
What is the name of this project?

NIOS_CUSTOM_PER_01

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.

NIOS_CUSTOM_PER_01

1.d Next >

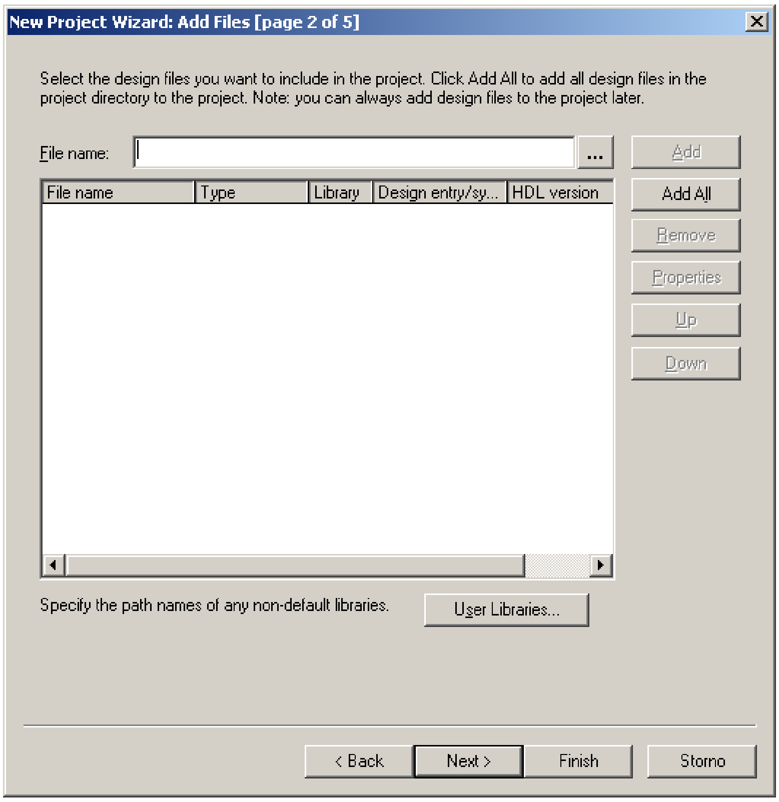


Quartus II

Directory "D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01" does not exist. Do you want to create it?

1.e Ano

1.f Zdrojové súbory budú pridané neskôr a tak pokračujeme Next >



New Project Wizard: Add Files [page 2 of 5]

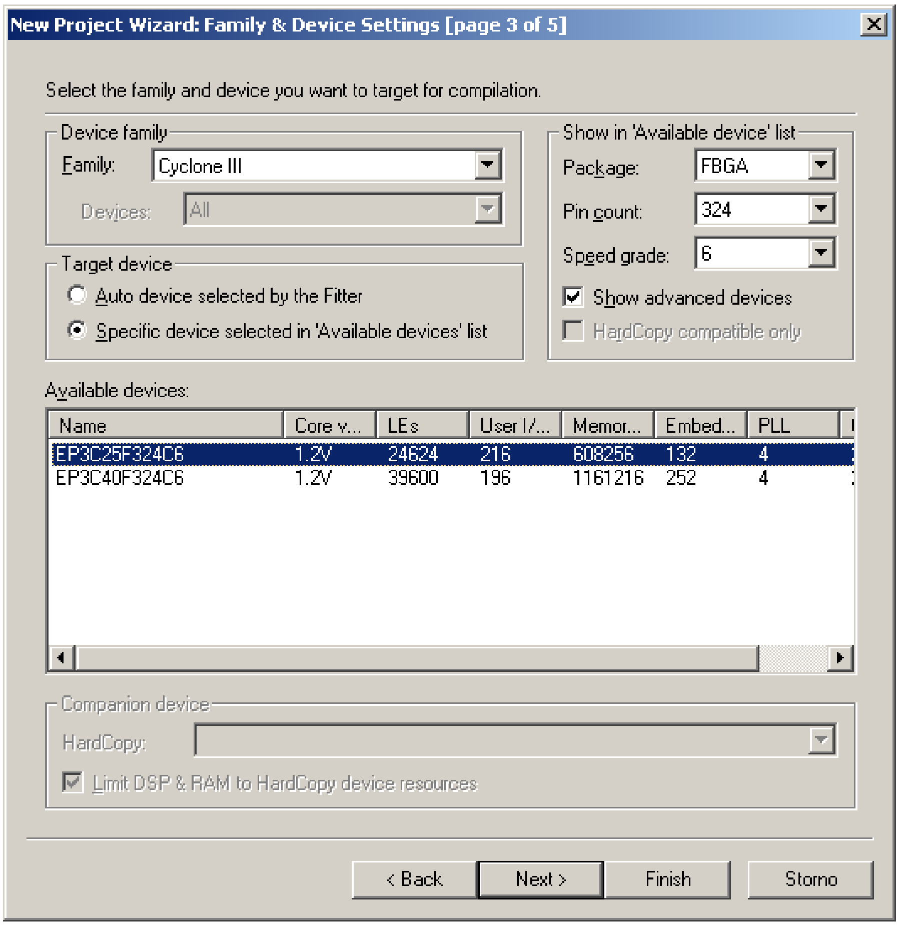
Select the design files you want to include in the project. Click Add All to add all design files in the project directory to the project. Note: you can always add design files to the project later.

File name:

File name	Type	Library	Design entry/sy...	HDL version

Specify the path names of any non-default libraries.

1.g Vyberieme FPGA súčiastku: EP3C25F324C6



New Project Wizard: Family & Device Settings [page 3 of 5]

Select the family and device you want to target for compilation.

Device family

Family: Cyclone III

Devices: All

Target device

○ Auto device selected by the Fitter

● Specific device selected in 'Available devices' list

Show in 'Available device' list

Package: FBGA

Pin count: 324

Speed grade: 6

☑ Show advanced devices

□ HardCopy compatible only

Available devices:

Name	Core v...	LEs	User I/...	Memor...	Embed...	PLL	I
EP3C25F324C6	1.2V	24624	216	608256	132	4	
EP3C40F324C6	1.2V	39600	196	1161216	252	4	

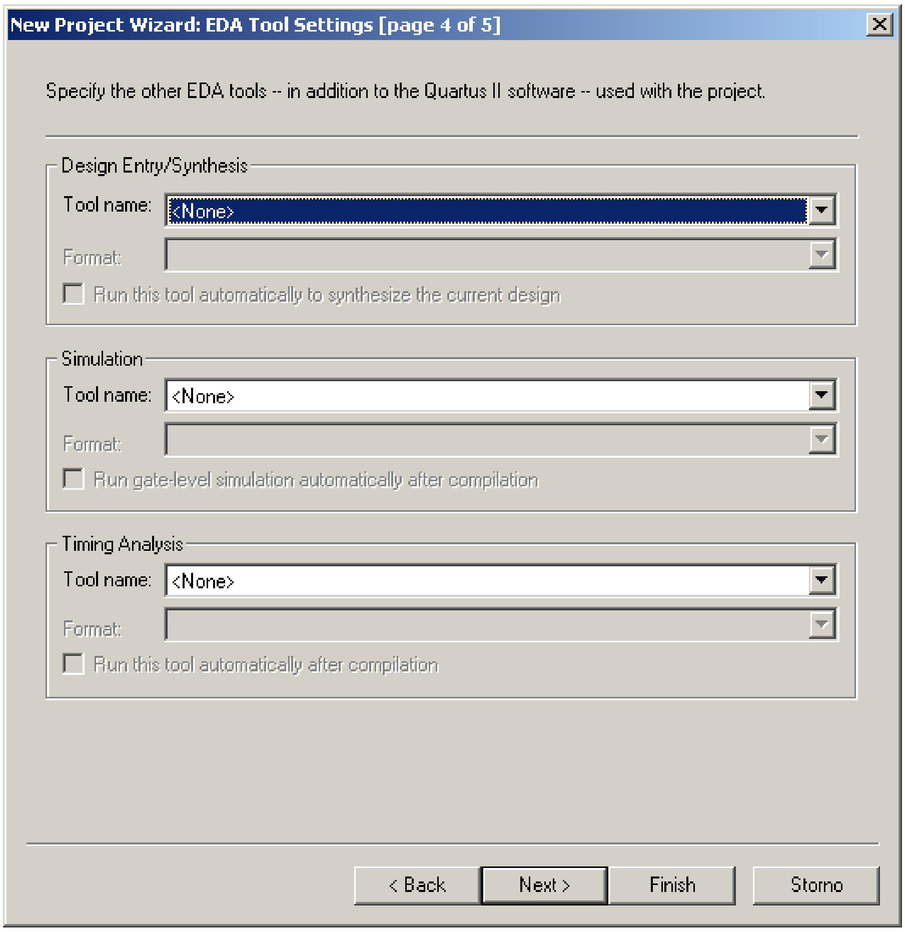
Companion device

HardCopy:

☑ Limit DSP & RAM to HardCopy device resources

1.h Next >

1.i Nevyberáme simuláciu pre ModelSim, nakoľko WebLicencia zrejme nepodporuje tvorbu simulačných súborov a tak kompilácia neprebehne korektne.



New Project Wizard: EDA Tool Settings [page 4 of 5]

Specify the other EDA tools -- in addition to the Quartus II software -- used with the project.

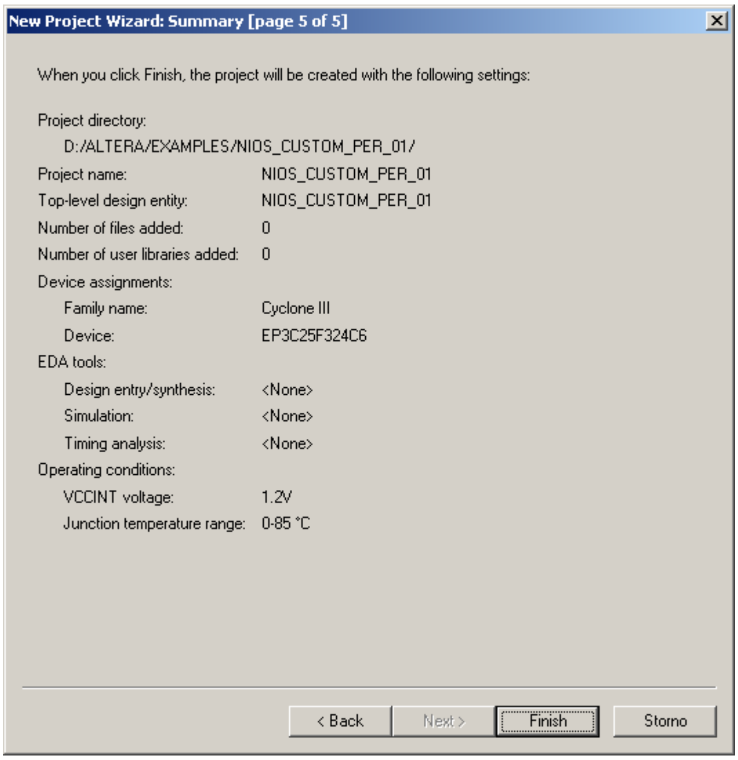
Design Entry/Synthesis
Tool name: <None>
Format:
□ Run this tool automatically to synthesize the current design

Simulation
Tool name: <None>
Format:
□ Run gate-level simulation automatically after compilation

Timing Analysis
Tool name: <None>
Format:
□ Run this tool automatically after compilation

1.j Next >

1.k Finish



New Project Wizard: Summary [page 5 of 5]

When you click Finish, the project will be created with the following settings:

Project directory:
D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01/

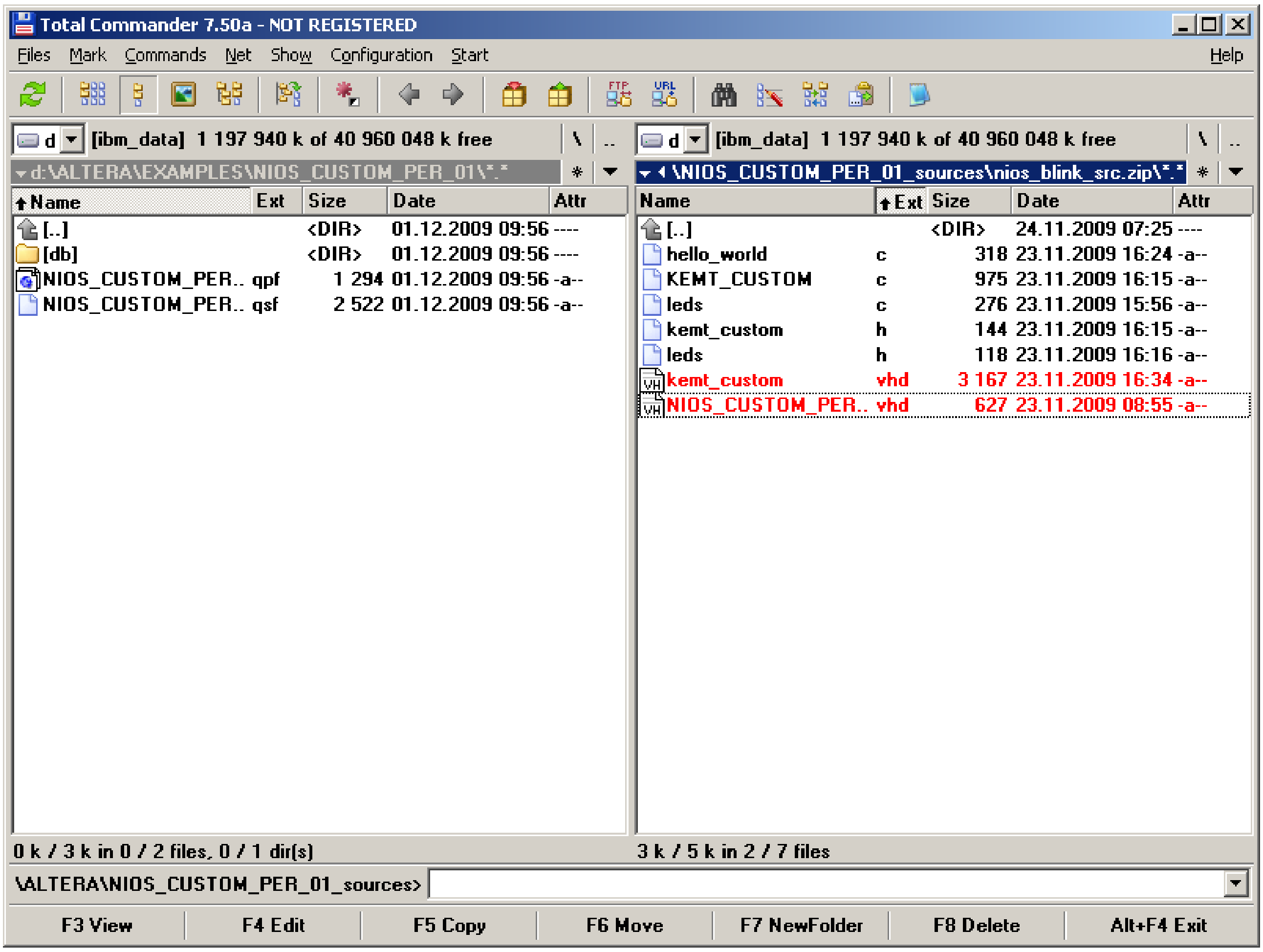
Project name:	NIOS_CUSTOM_PER_01
Top-level design entity:	NIOS_CUSTOM_PER_01
Number of files added:	0
Number of user libraries added:	0
Device assignments:	
Family name:	Cyclone III
Device:	EP3C25F324C6
EDA tools:	
Design entry/synthesis:	<None>
Simulation:	<None>
Timing analysis:	<None>
Operating conditions:	
VCCINT voltage:	1.2V
Junction temperature range:	0-85 °C

2. Pridanie zdrojových súborov

2.a Do projektového adresára nahrajte súbory z balíku nios_blink_src.zip:

kemt_custom.vhd (VHDL popis periférie čítača)

NIOS_CUSTOM_PER_01.vhd (Top Level Entita projektu)



d [ibm_data] 1 197 940 k of 40 960 048 k free

d:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01*.*

Name	Ext	Size	Date	Attr
[..]		<DIR>	01.12.2009 09:56	----
[db]		<DIR>	01.12.2009 09:56	----
NIOS_CUSTOM_PER..	qpf	1 294	01.12.2009 09:56	-a--
NIOS_CUSTOM_PER..	qsf	2 522	01.12.2009 09:56	-a--

0 k / 3 k in 0 / 2 files, 0 / 1 dir(s)

d [ibm_data] 1 197 940 k of 40 960 048 k free

\NIOS_CUSTOM_PER_01_sources\nios_blink_src.zip*.*

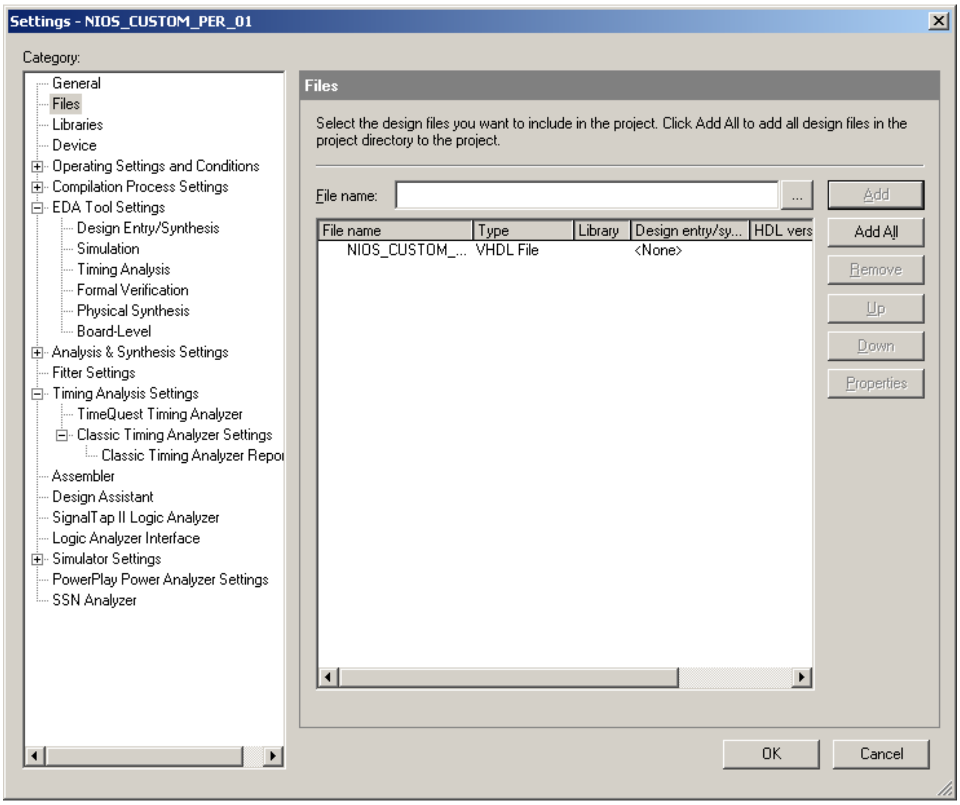
Name	Ext	Size	Date	Attr
[..]		<DIR>	24.11.2009 07:25	----
hello_world	c	318	23.11.2009 16:24	-a--
KEMT_CUSTOM	c	975	23.11.2009 16:15	-a--
leds	c	276	23.11.2009 15:56	-a--
kemt_custom	h	144	23.11.2009 16:15	-a--
leds	h	118	23.11.2009 16:16	-a--
kemt_custom	vhd	3 167	23.11.2009 16:34	-a--
NIOS_CUSTOM_PER..	vhd	627	23.11.2009 08:55	-a--

3 k / 5 k in 2 / 7 files

\ALTERA\NIOS_CUSTOM_PER_01_sources>

2.b V prostredí Quartus II pridajte súbor NIOS_CUSTOM_PER_01.vhd do projektu:

Project > Add/Remove Files in Project …



Settings - NIOS_CUSTOM_PER_01

Files

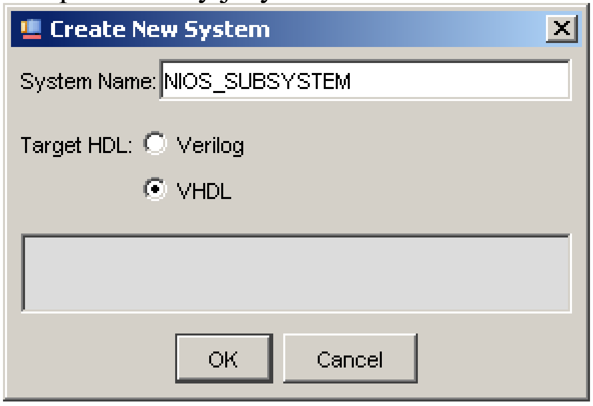
Select the design files you want to include in the project. Click Add All to add all design files in the project directory to the project.

File name:

File name	Type	Library	Design entry/sy...	HDL vers
NIOS_CUSTOM_...	VHDL File		<None>	

2.c OK

3. Vytvorenie procesora NIOS II a jeho subsystému:
   3.a Tools > SOPC Builder
   3.b V dialógovom okne vyplníme názov Top Level Entity pre procesor – NIOS_SUBSYSTEM a preferovaný jazyk – VHDL



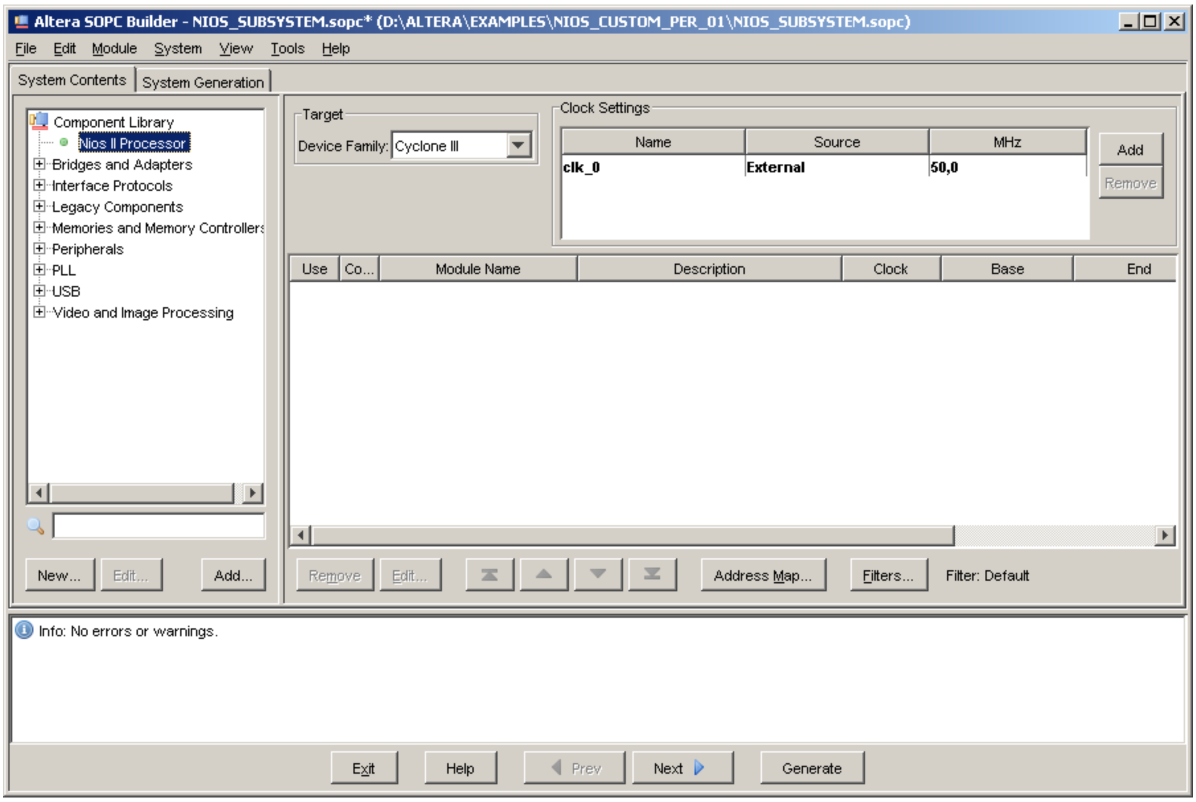
Create New System

System Name: NIOS_SUBSYSTEM

Target HDL: ○ Verilog

● VHDL

   3.c OK
   3.d V prostredí SOPC Builder klikneme na v časti naľavo na NIOS II Processor a stlačíme Add



Altera SOPC Builder - NIOS_SUBSYSTEM.sopc* (D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\NIOS_SUBSYSTEM.sopc)

File Edit Module System View Tools Help
System Contents
System Generation
Component Library
Nios II Processor
Bridges and Adapters
Interface Protocols
Legacy Components
Memories and Memory Controllers
Peripherals
PLL
USB
Video and Image Processing
New... Edit... Add...
Remove Edit... Address Map... Filters...
Exit Help Prev Next Generate

Target

Device Family: Cyclone III

Clock Settings

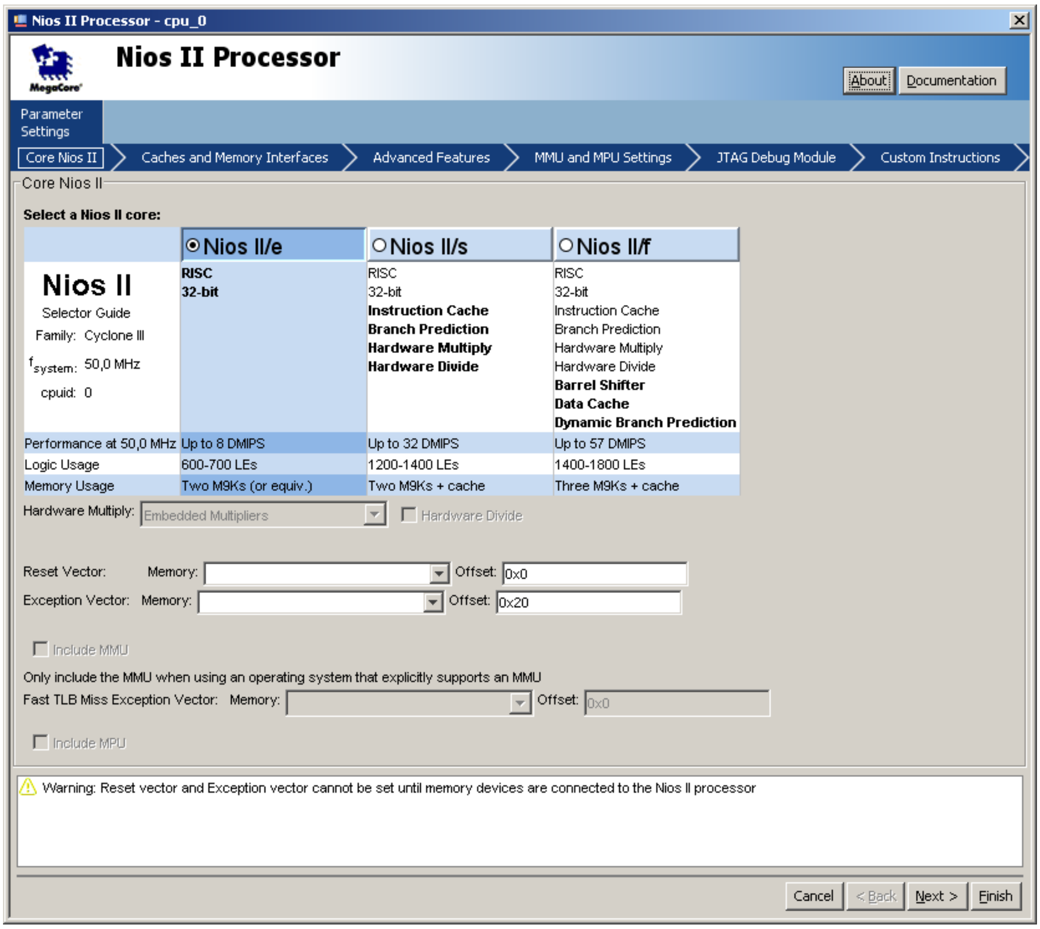
Name	Source	MHz
clk_0	External	50,0

Use	Co...	Module Name	Description	Clock	Base	End

Filter: Default

Info: No errors or warnings.

3.e Následne bude otvorené okno určené ku konfigurácií NIOS II. Vyberieme ekonomickú verziu procesora (najmenšia, najpomalšia) Nios II/e a stlačíme Finish. Zvyšnú konfiguráciu vykonáme neskôr.



Nios II Processor - cpu_0

Nios II Processor

Core Nios II

Select a Nios II core:

Nios II Selector Guide Family: Cyclone III f_{system}: 50,0 MHz cpuid: 0	● Nios II/e	○ Nios II/s	○ Nios II/f
	RISC 32-bit	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide Barrel Shifter Data Cache Dynamic Branch Prediction
Performance at 50,0 MHz	Up to 8 DMIPS	Up to 32 DMIPS	Up to 57 DMIPS
Logic Usage	600-700 LEs	1200-1400 LEs	1400-1800 LEs
Memory Usage	Two M9Ks (or equiv.)	Two M9Ks + cache	Three M9Ks + cache

Hardware Multiply: Embedded Multipliers □ Hardware Divide

Reset Vector: Memory: Offset: 0x0

Exception Vector: Memory: Offset: 0x20

□ Include MMU

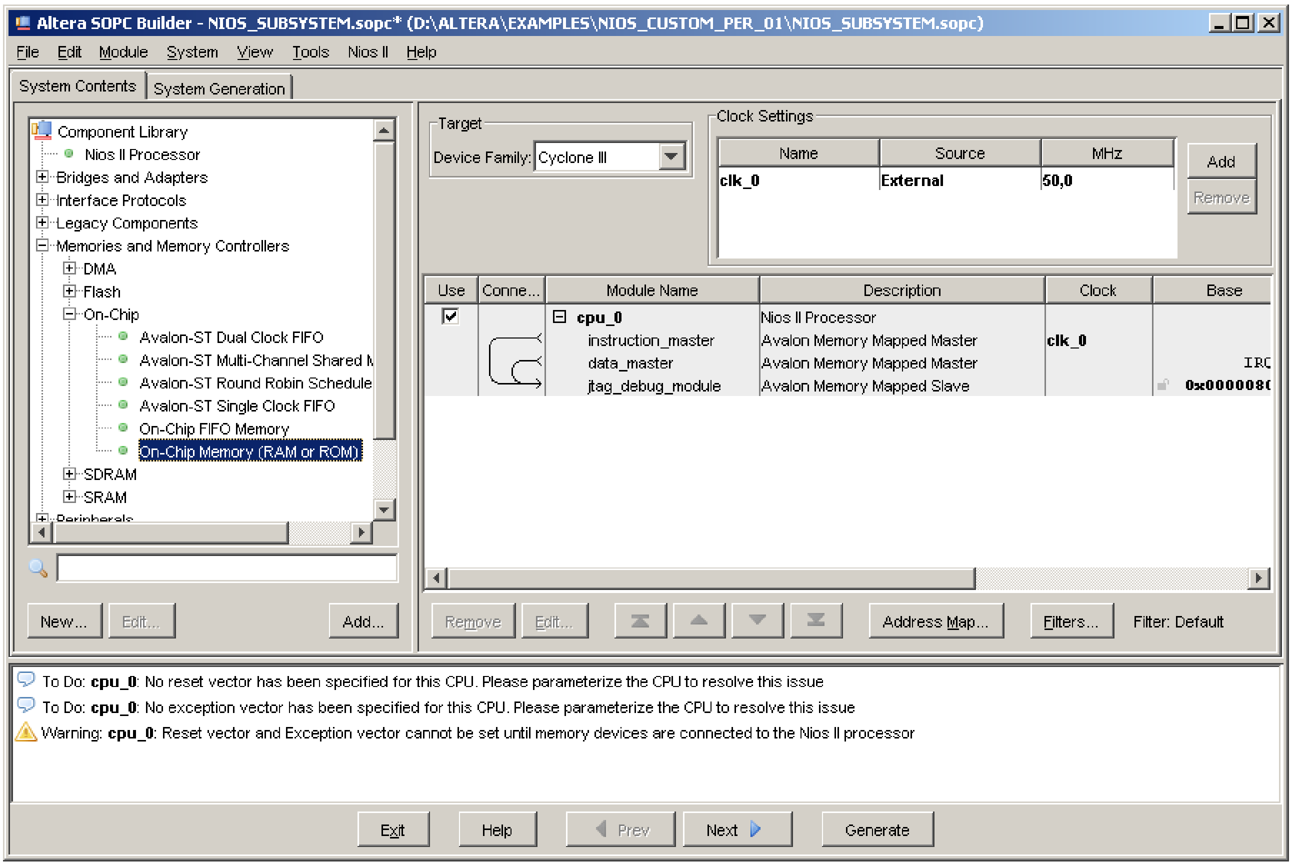
Only include the MMU when using an operating system that explicitly supports an MMU

Fast TLB Miss Exception Vector: Memory: Offset: 0x0

□ Include MPU

Warning: Reset vector and Exception vector cannot be set until memory devices are connected to the Nios II processor

3.f Pridáme pamäť RAM: Memories and Memory Controllers > On-Chip > On-Chip memory (RAM or ROM) a stlašíme add


Altera SOPC Builder - NIOS_SUBSYSTEM.sopc* (D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\NIOS_SUBSYSTEM.sopc)
File Edit Module System View Tools Nios II Help
System Contents
System Generation
Component Library
Nios II Processor
Bridges and Adapters
Interface Protocols
Legacy Components
Memories and Memory Controllers
DMA
Flash
On-Chip
Avalon-ST Dual Clock FIFO
Avalon-ST Multi-Channel Shared M
Avalon-ST Round Robin Schedule
Avalon-ST Single Clock FIFO
On-Chip FIFO Memory
On-Chip Memory (RAM or ROM)
SDRAM
SRAM
Peripherals
New...
Edit...
Add...
Target
Clock Settings
Add
Remove
Remove
Edit...
Address Map...
Filters...
Filter: Default
Exit
Help
Prev
Next
Generate

Device Family: Cyclone III

Name	Source	MHz
clk_0	External	50,0

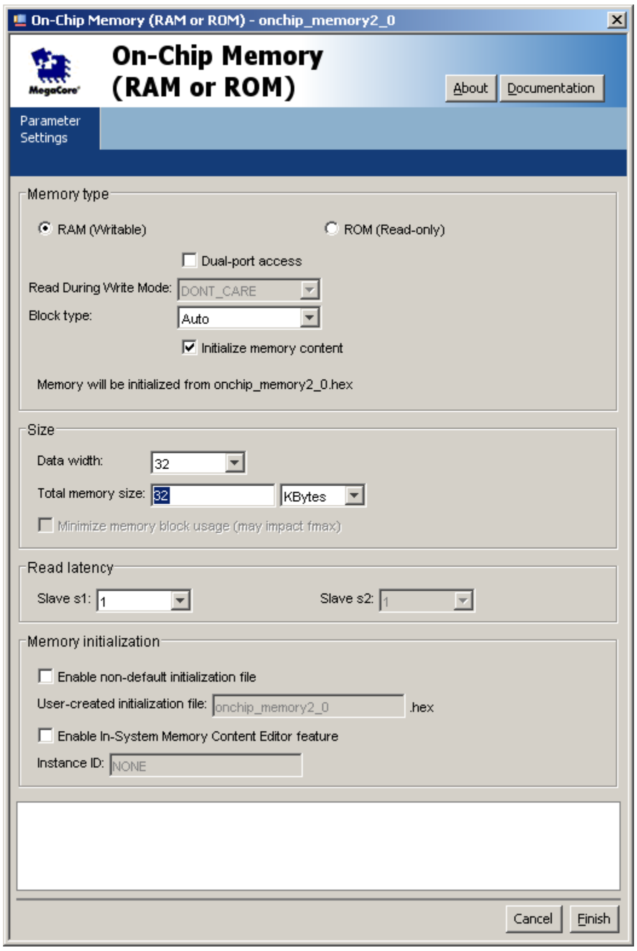
Use	Conne...	Module Name	Description	Clock	Base
☑		cpu_0	Nios II Processor		
		instruction_master	Avalon Memory Mapped Master	clk_0	
		data_master	Avalon Memory Mapped Master		IRQ
		jtag_debug_module	Avalon Memory Mapped Slave		0x0000080

To Do: **cpu_0**: No reset vector has been specified for this CPU. Please parameterize the CPU to resolve this issue

To Do: **cpu_0**: No exception vector has been specified for this CPU. Please parameterize the CPU to resolve this issue

⚠ Warning: **cpu_0**: Reset vector and Exception vector cannot be set until memory devices are connected to the Nios II processor

3.g Následne bude otvorené konfiguračné okno pre pamäť, kde nastavíme jej veľkosť na 32kB a stlačíme Finish.



On-Chip Memory (RAM or ROM) - onchip_memory2_0

On-Chip Memory (RAM or ROM)

Memory type

● RAM (Writable) ○ ROM (Read-only)

□ Dual-port access

Read During Write Mode: DONT_CARE

Block type: Auto

☑ Initialize memory content

Memory will be initialized from onchip_memory2_0.hex

Size

Data width: 32

Total memory size: 32 KBytes

□ Minimize memory block usage (may impact fmax)

Read latency

Slave s1: 1 Slave s2: 1

Memory initialization

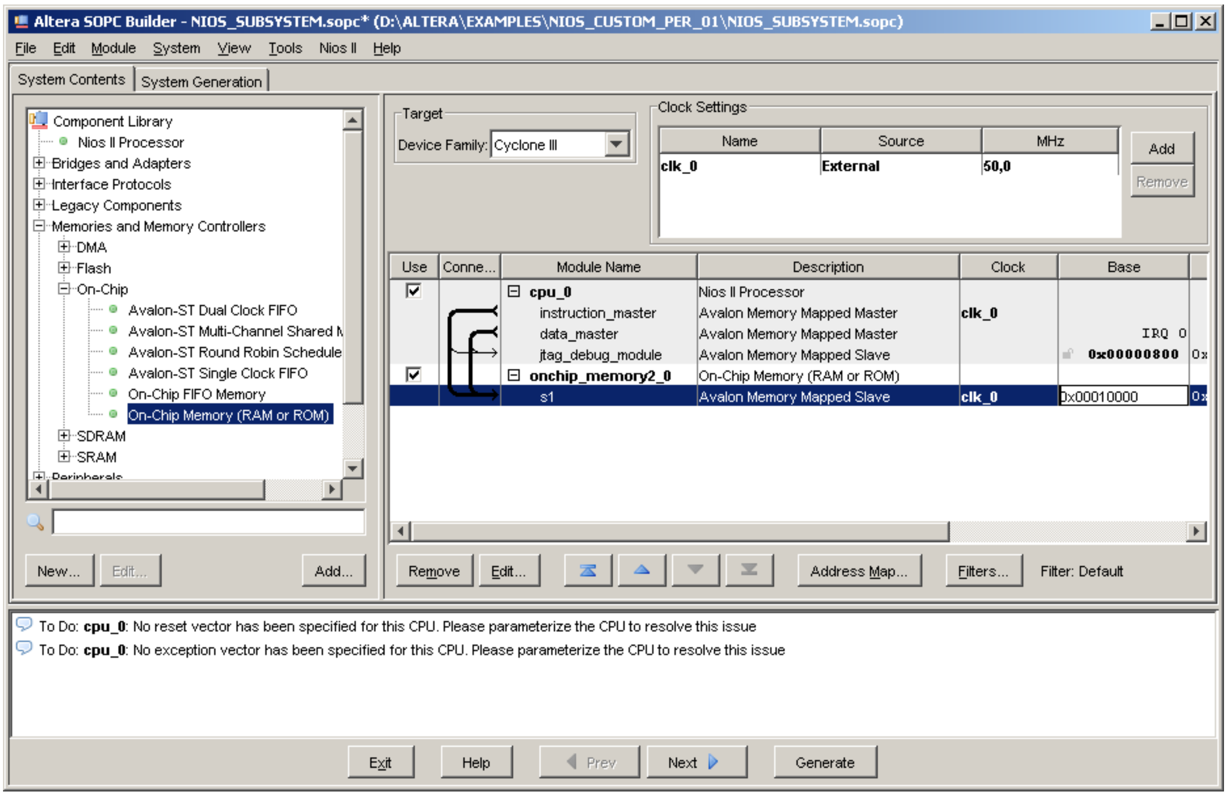
□ Enable non-default initialization file

User-created initialization file: onchip_memory2_0 .hex

□ Enable In-System Memory Content Editor feature

Instance ID: NONE

3.g Upravíme začiatočnú adresu pamäte v adresovom priestore NIOS II na 0x00010000 dvojitým kliknutím na jej pôvodný začiatok:



Altera SOPC Builder - NIOS_SUBSYSTEM.sopc* (D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\NIOS_SUBSYSTEM.sopc)

File Edit Module System View Tools Nios II Help
System Contents
System Generation
Component Library
Nios II Processor
Bridges and Adapters
Interface Protocols
Legacy Components
Memories and Memory Controllers
DMA
Flash
On-Chip
Avalon-ST Dual Clock FIFO
Avalon-ST Multi-Channel Shared M
Avalon-ST Round Robin Schedule
Avalon-ST Single Clock FIFO
On-Chip FIFO Memory
On-Chip Memory (RAM or ROM)
SDRAM
SRAM
Peripherals
New...
Edit...
Add...

Target

Device Family: Cyclone III

Clock Settings

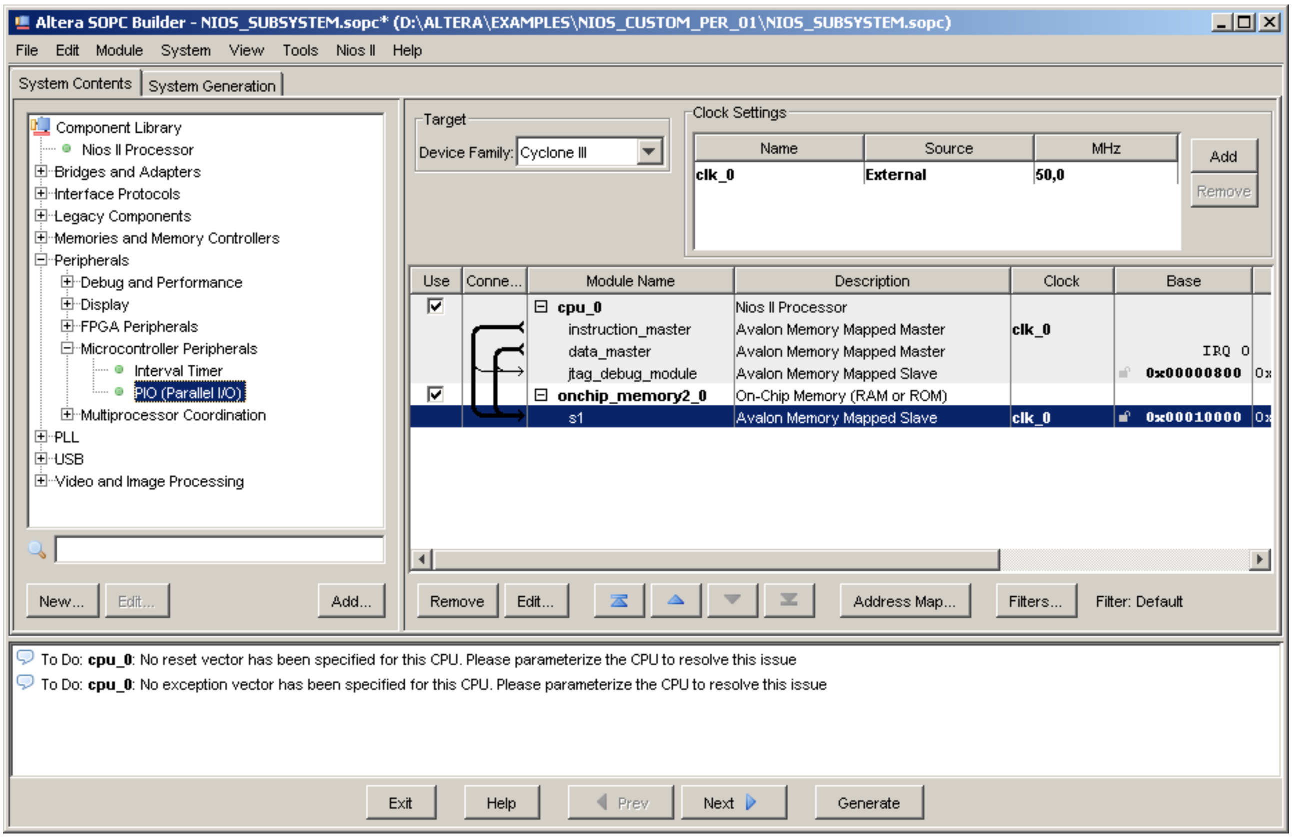
Name	Source	MHz
clk_0	External	50,0

Use	Conne...	Module Name	Description	Clock	Base
☑		⊟ cpu_0	Nios II Processor		
		instruction_master	Avalon Memory Mapped Master	clk_0	
		data_master	Avalon Memory Mapped Master		IRQ 0
		jtag_debug_module	Avalon Memory Mapped Slave		0x00000800
☑		⊟ onchip_memory2_0	On-Chip Memory (RAM or ROM)		
		s1	Avalon Memory Mapped Slave	clk_0	0x00010000

To Do: **cpu_0**: No reset vector has been specified for this CPU. Please parameterize the CPU to resolve this issue

To Do: **cpu_0**: No exception vector has been specified for this CPU. Please parameterize the CPU to resolve this issue

3.h Pridáme periférium PIO (Parallel Input / Output) pre pripojenie dvoch LED. Z ponuky vyberieme: Peripherals > Microcontroller Peripherals > PIO (Parallel Input / Output) a stlačíme Add.


Altera SOPC Builder - NIOS_SUBSYSTEM.sopc* (D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\NIOS_SUBSYSTEM.sopc)
File Edit Module System View Tools Nios II Help
System Contents
System Generation
Component Library
Nios II Processor
Bridges and Adapters
Interface Protocols
Legacy Components
Memories and Memory Controllers
Peripherals
Debug and Performance
Display
FPGA Peripherals
Microcontroller Peripherals
Interval Timer
PIO (Parallel I/O)
Multiprocessor Coordination
PLL
USB
Video and Image Processing
New... Edit... Add...
Target
Clock Settings
Add
Remove
Remove Edit... Address Map... Filters... Filter: Default
Exit Help Prev Next Generate

Device Family: Cyclone III

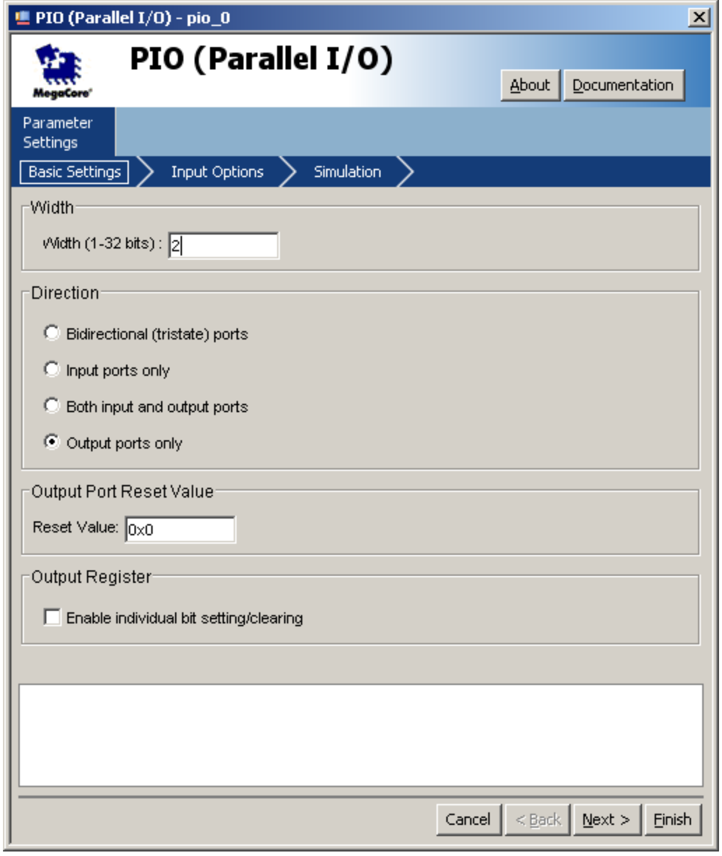
Name	Source	MHz
clk_0	External	50,0

Use	Conne...	Module Name	Description	Clock	Base
☑		⊟ cpu_0	Nios II Processor		
		instruction_master	Avalon Memory Mapped Master	clk_0	
		data_master	Avalon Memory Mapped Master		IRQ 0
		jtag_debug_module	Avalon Memory Mapped Slave		0x00000800
☑		⊟ onchip_memory2_0	On-Chip Memory (RAM or ROM)		
		s1	Avalon Memory Mapped Slave	clk_0	0x00010000

To Do: **cpu_0**: No reset vector has been specified for this CPU. Please parameterize the CPU to resolve this issue

To Do: **cpu_0**: No exception vector has been specified for this CPU. Please parameterize the CPU to resolve this issue

3.i Následne bude otvorené konfiguračné okno. Upravíme šírku výstupu na dva bity. Finish.



Width

Width (1-32 bits) : 2

Direction

○ Bidirectional (tristate) ports

○ Input ports only

○ Both input and output ports

● Output ports only

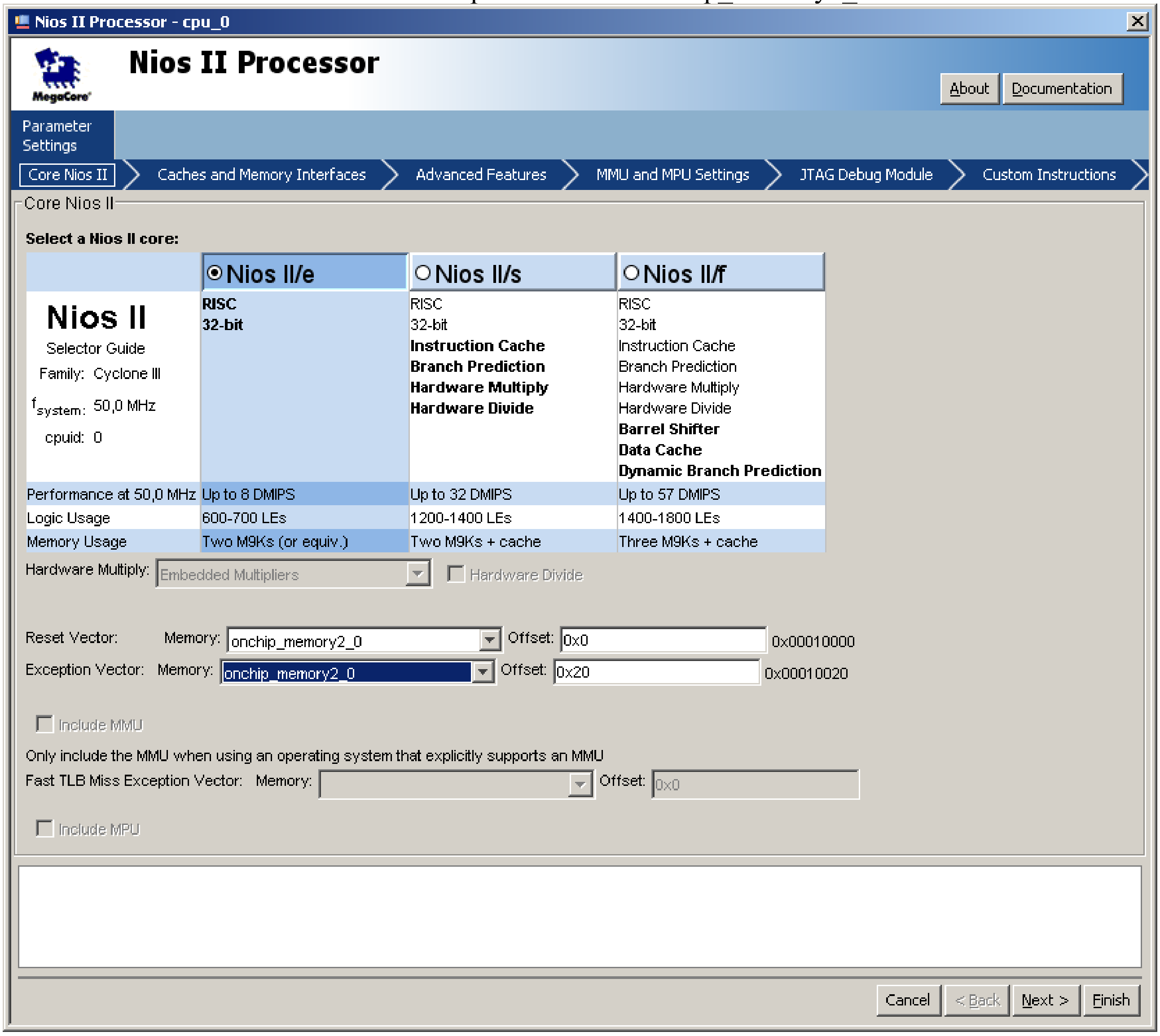
Output Port Reset Value

Reset Value: 0x0

Output Register

□ Enable individual bit setting/clearing

3.j Dvojitým kliknutím na NIOS II otvoríme konfiguračné okno procesora. Vyberieme Umiestnenie Reset Vector a Exception Vector v onchip_memory2_0. Finish.



Nios II Processor - cpu_0

Nios II Processor

Select a Nios II core:

Nios II Selector Guide Family: Cyclone III f_{system}: 50,0 MHz cpuid: 0	● Nios II/e	○ Nios II/s	○ Nios II/f
	RISC **32-bit**	RISC 32-bit **Instruction Cache** **Branch Prediction** **Hardware Multiply** **Hardware Divide**	RISC 32-bit Instruction Cache Branch Prediction Hardware Multiply Hardware Divide **Barrel Shifter** **Data Cache** **Dynamic Branch Prediction**
Performance at 50,0 MHz	Up to 8 DMIPS	Up to 32 DMIPS	Up to 57 DMIPS
Logic Usage	600-700 LEs	1200-1400 LEs	1400-1800 LEs
Memory Usage	Two M9Ks (or equiv.)	Two M9Ks + cache	Three M9Ks + cache

Hardware Multiply: Embedded Multipliers □ Hardware Divide

Reset Vector: Memory: onchip_memory2_0 Offset: 0x0 0x00010000

Exception Vector: Memory: onchip_memory2_0 Offset: 0x20 0x00010020

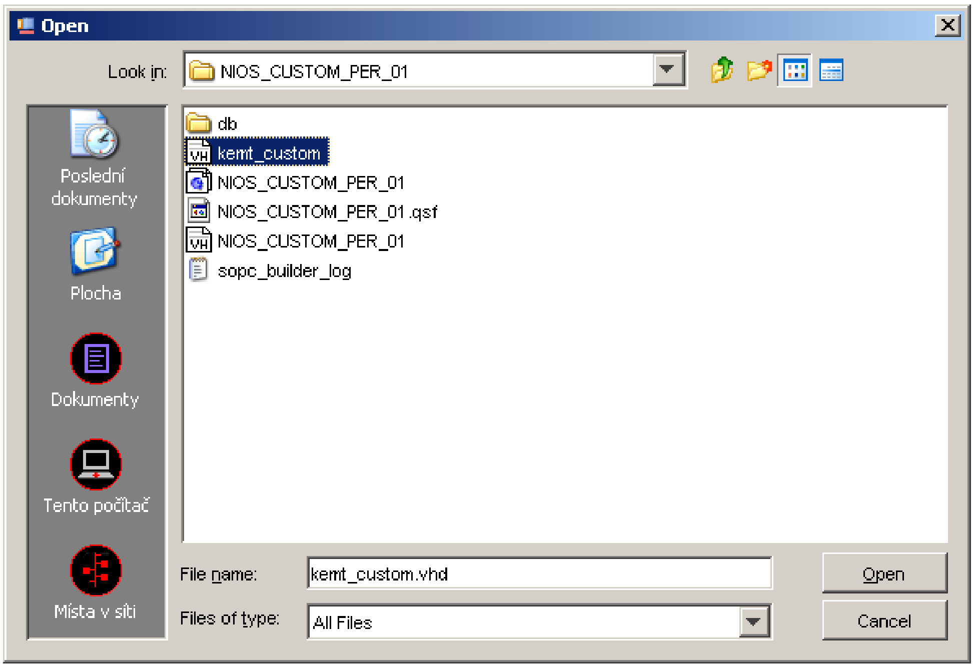
□ Include MMU

Only include the MMU when using an operating system that explicitly supports an MMU

Fast TLB Miss Exception Vector: Memory: Offset: 0x0

□ Include MPU

4. Vytvorenie vlastnej periférie
   4.a V SOPC Builderi v záložke System Contents kliknite na New.
   4.b Otvorí sa okno Component Editor s aktívnou záložnou Introduction. Pokračujeme stlačením Next >
   4.c Teraz je otvorená záložka, kde môžeme pridať HDL zdrojový súbor pre vlastnú periférius. Stlačíme Add.



Open

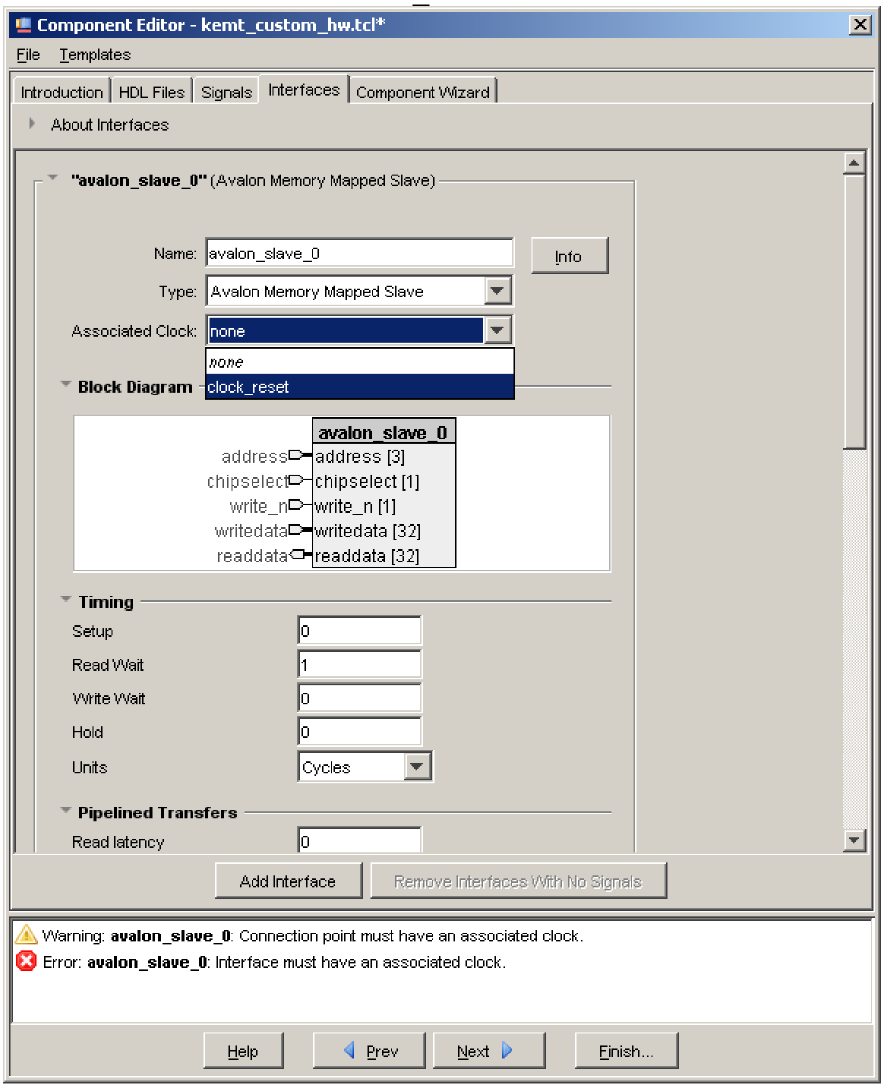
Look in: NIOS_CUSTOM_PER_01

db
kemt_custom
NIOS_CUSTOM_PER_01
NIOS_CUSTOM_PER_01.qsf
NIOS_CUSTOM_PER_01
sopc_builder_log

File name: kemt_custom.vhd

Files of type: All Files

4.d stlačíme Open
4.e Po analýze súboru stlačíme Close.
4.f Tlačidlom Next > pokračujeme do záložky Signals.
4.g Tlačidlom Next > pokračujeme do záložky Interfaces
4.h Vyberieme Associated Clock – clock_reset



Component Editor - kemt_custom_hw.tcl*

"avalon_slave_0" (Avalon Memory Mapped Slave)

Name: avalon_slave_0

Type: Avalon Memory Mapped Slave

Associated Clock: none

none
clock_reset

Block Diagram

Timing

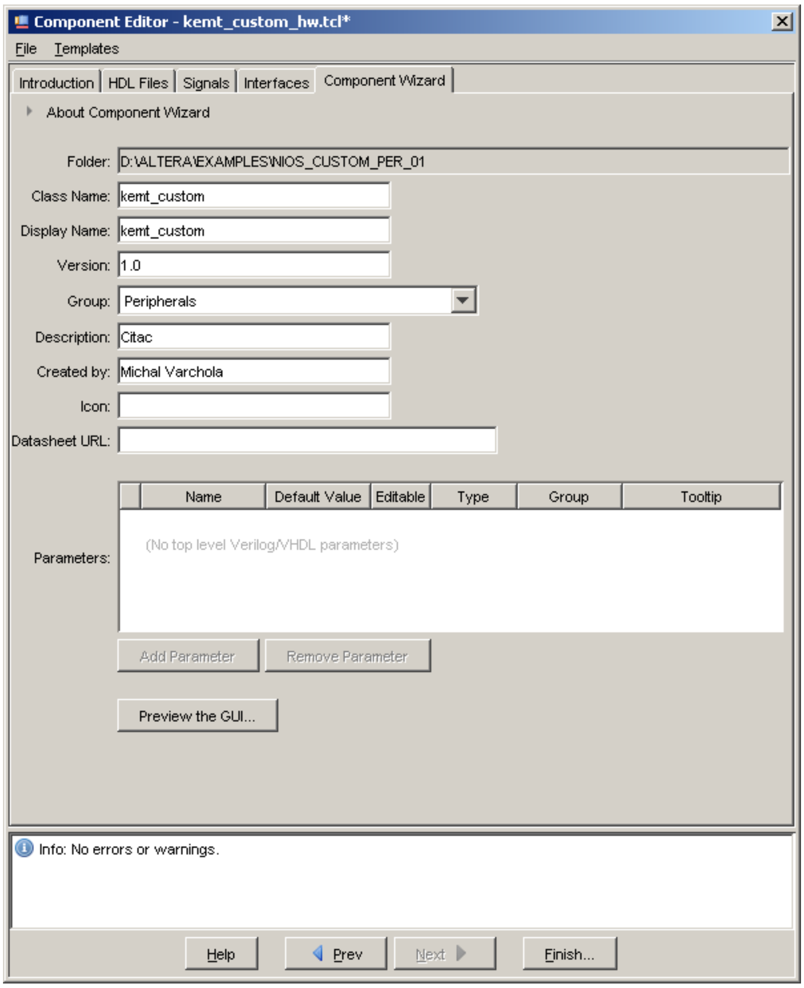
Setup	0
Read Wait	1
Write Wait	0
Hold	0
Units	Cycles

Pipelined Transfers

Read latency 0

Warning: **avalon_slave_0**: Connection point must have an associated clock.
Error: **avalon_slave_0**: Interface must have an associated clock.

4.i Pokračujeme tlačidlom Next > do záložky Component Wizard. Vyberieme Group - Peripherals a prípadne vyplníme zvyšok. Pokračujeme stlačením Finish...



Folder: D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01

Class Name: kemt_custom

Display Name: kemt_custom

Version: 1.0

Group: Peripherals

Description: Citac

Created by: Michal Varchola

Icon:

Datasheet URL:

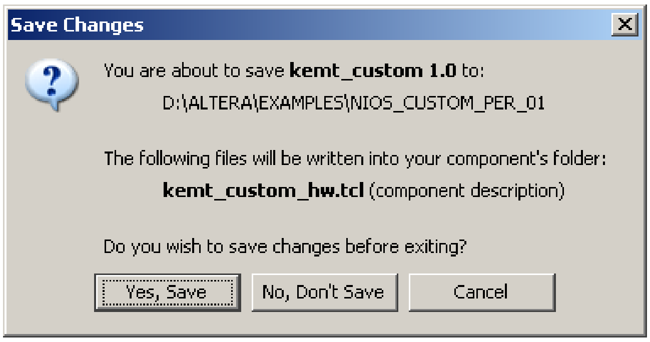
Parameters:

	Name	Default Value	Editable	Type	Group	Tooltip

(No top level Verilog/VHDL parameters)

Info: No errors or warnings.

4.j Komponent uložíme.

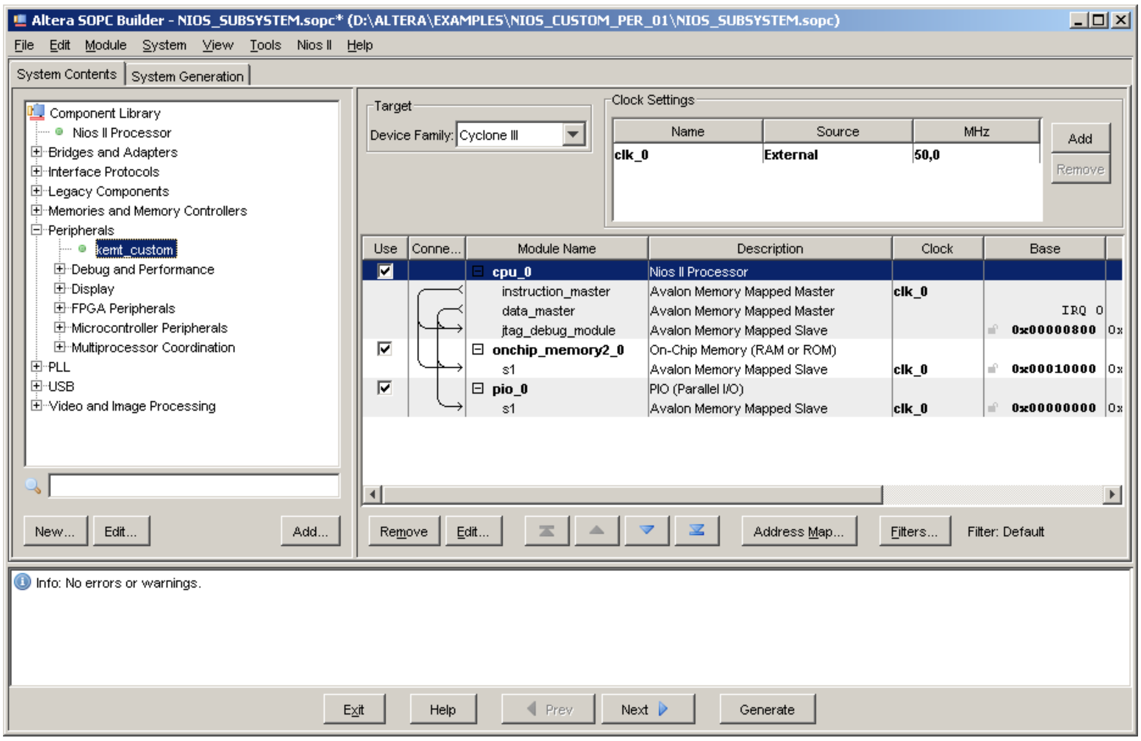


You are about to save **kemt_custom 1.0** to:
D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01

The following files will be written into your component's folder:
kemt_custom_hw.tcl (component description)

Do you wish to save changes before exiting?

4.k Komponent kemt_custom pridáme do projektu:



Altera SOPC Builder - NIOS_SUBSYSTEM.sopc* (D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\NIOS_SUBSYSTEM.sopc)

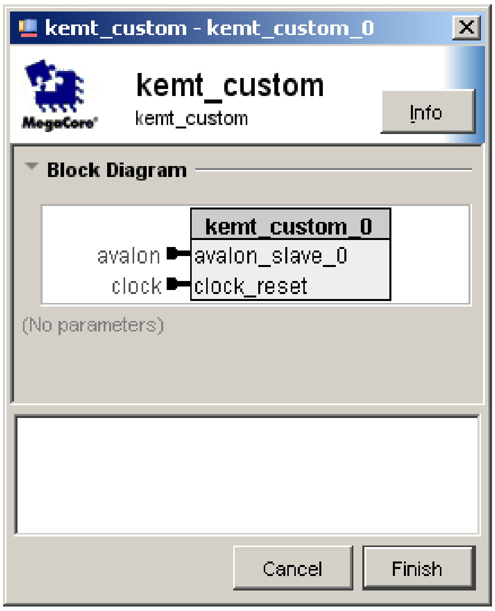
Device Family: Cyclone III

Name	Source	MHz
clk_0	External	50,0

Use	Conne...	Module Name	Description	Clock	Base	
☑		cpu_0	Nios II Processor			
		instruction_master	Avalon Memory Mapped Master	clk_0		
		data_master	Avalon Memory Mapped Master		IRQ 0	
		jtag_debug_module	Avalon Memory Mapped Slave		0x00000800	0x
☑		onchip_memory2_0	On-Chip Memory (RAM or ROM)			
		s1	Avalon Memory Mapped Slave	clk_0	0x00010000	0x
☑		pio_0	PIO (Parallel I/O)			
		s1	Avalon Memory Mapped Slave	clk_0	0x00000000	0x

Info: No errors or warnings.

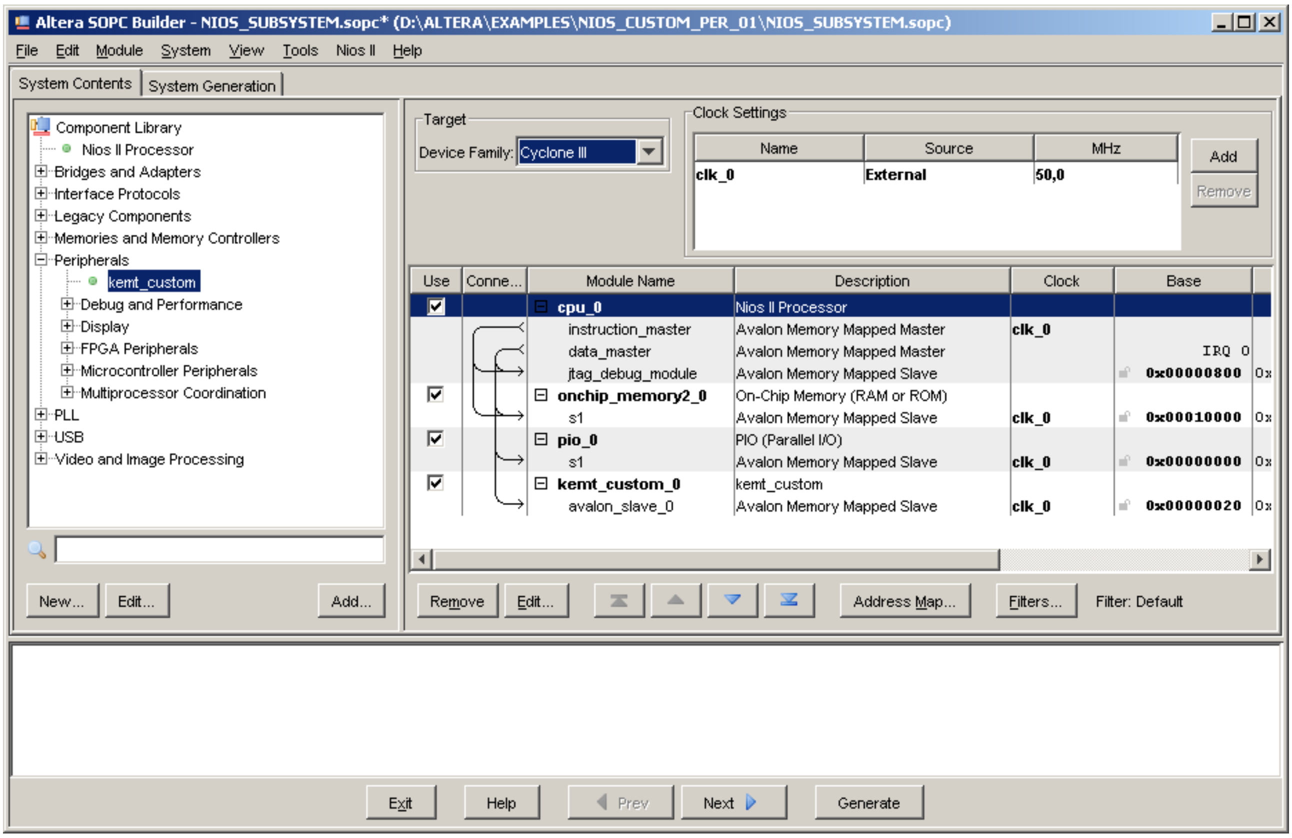
4.l Stlačíme Add
4.m Stlačíme Finish



kemt_custom - kemt_custom_0

(No parameters)

4.n Takto máme vytvorený NIOS II procesor so všetkými potrebnými perifériami

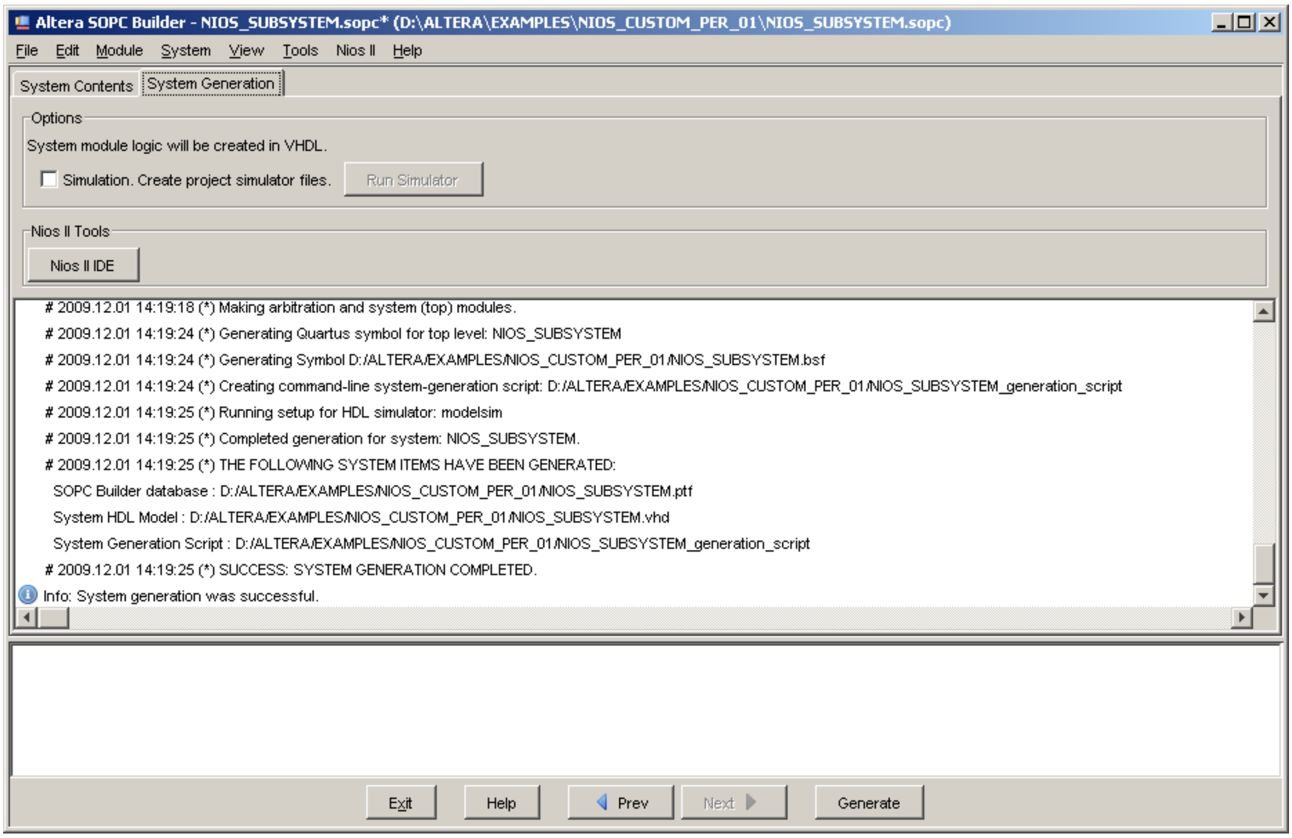


Device Family: Cyclone III

Name	Source	MHz
clk_0	External	50,0

Use	Conne...	Module Name	Description	Clock	Base
☑		⊟ **cpu_0**	Nios II Processor		
		instruction_master	Avalon Memory Mapped Master	clk_0	
		data_master	Avalon Memory Mapped Master		IRQ 0
		jtag_debug_module	Avalon Memory Mapped Slave		0x00000800
☑		⊟ **onchip_memory2_0**	On-Chip Memory (RAM or ROM)		
		s1	Avalon Memory Mapped Slave	clk_0	0x00010000
☑		⊟ **pio_0**	PIO (Parallel I/O)		
		s1	Avalon Memory Mapped Slave	clk_0	0x00000000
☑		⊟ **kemt_custom_0**	kemt_custom		
		avalon_slave_0	Avalon Memory Mapped Slave	clk_0	0x00000020

4.o Stlačíme tlačidlo Generate, pri výzve uložíme projekt.


Altera SOPC Builder - NIOS_SUBSYSTEM.sopc* (D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\NIOS_SUBSYSTEM.sopc)
File Edit Module System View Tools Nios II Help
System Contents
System Generation
Options
Run Simulator
Nios II Tools
Nios II IDE
Exit
Help
Prev
Next
Generate

System module logic will be created in VHDL.

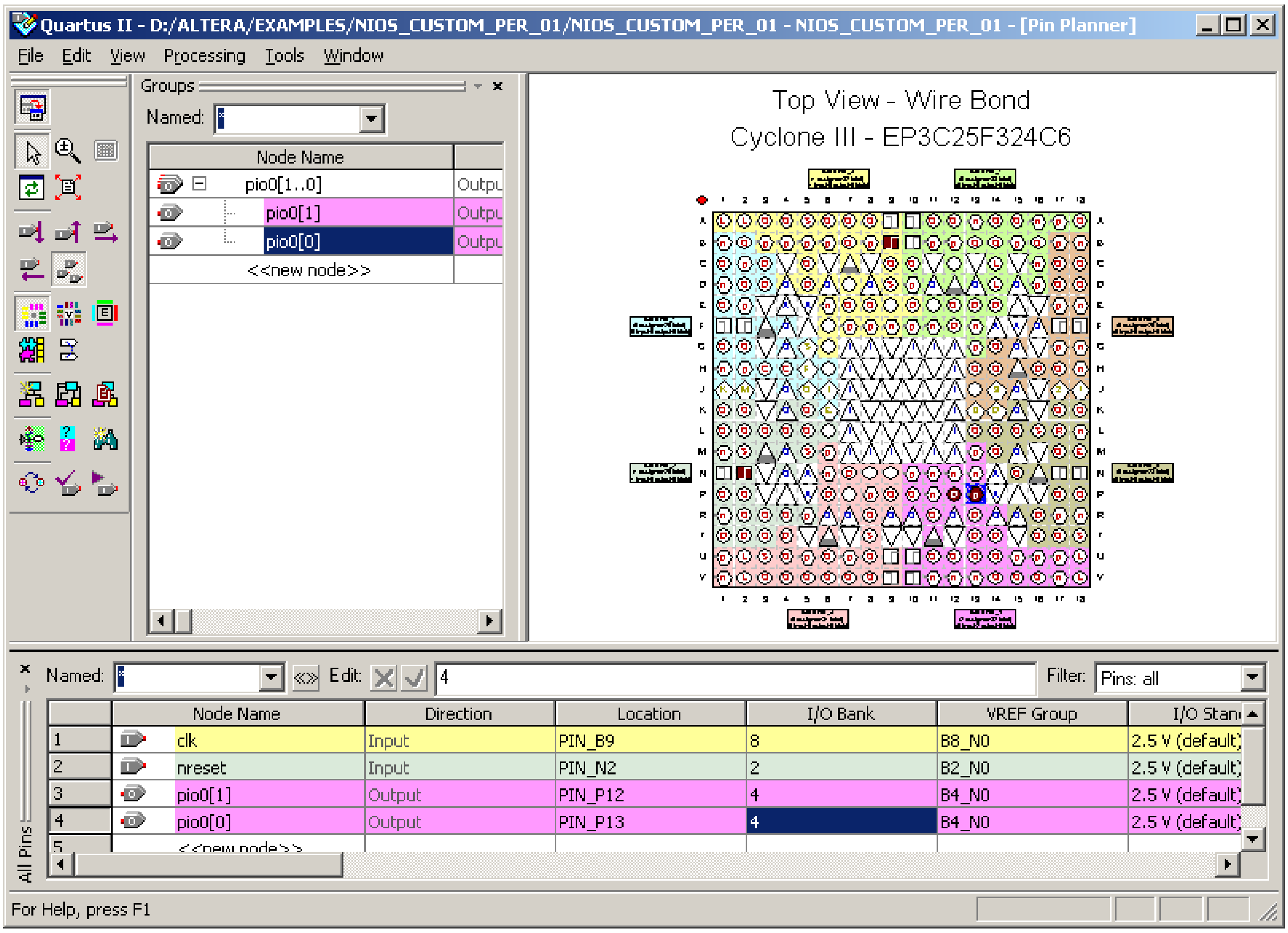
☐ Simulation. Create project simulator files.

```
# 2009.12.01 14:19:18 (*) Making arbitration and system (top) modules.
# 2009.12.01 14:19:24 (*) Generating Quartus symbol for top level: NIOS_SUBSYSTEM
# 2009.12.01 14:19:24 (*) Generating Symbol D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01/NIOS_SUBSYSTEM.bsf
# 2009.12.01 14:19:24 (*) Creating command-line system-generation script: D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01/NIOS_SUBSYSTEM_generation_script
# 2009.12.01 14:19:25 (*) Running setup for HDL simulator: modelsim
# 2009.12.01 14:19:25 (*) Completed generation for system: NIOS_SUBSYSTEM.
# 2009.12.01 14:19:25 (*) THE FOLLOWING SYSTEM ITEMS HAVE BEEN GENERATED:
  SOPC Builder database : D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01/NIOS_SUBSYSTEM.ptf
  System HDL Model : D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01/NIOS_SUBSYSTEM.vhd
  System Generation Script : D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01/NIOS_SUBSYSTEM_generation_script
# 2009.12.01 14:19:25 (*) SUCCESS: SYSTEM GENERATION COMPLETED.
Info: System generation was successful.
```

4.p Po úspešnom vygenerovaní systému stlačíme Exit a pri výzve uložíme projekt

5. Kompilácia celého projektu v prostredí Quartus.
   5.a Projekt skompilujeme: Processing > Start Compilation
   5.b Priradíme piny: Assigments > Pin Planner:

clk:	B9
nreset:	N2
pio0[1]:	P12
pio0[0]:	P13


Quartus II - D:/ALTERA/EXAMPLES/NIOS_CUSTOM_PER_01/NIOS_CUSTOM_PER_01 - NIOS_CUSTOM_PER_01 - [Pin Planner]
File
Edit
View
Processing
Tools
Window
Groups
Named:
Top View - Wire Bond
Cyclone III - EP3C25F324C6
Named:
Edit:
Filter:
Pins: all
All Pins
For Help, press F1

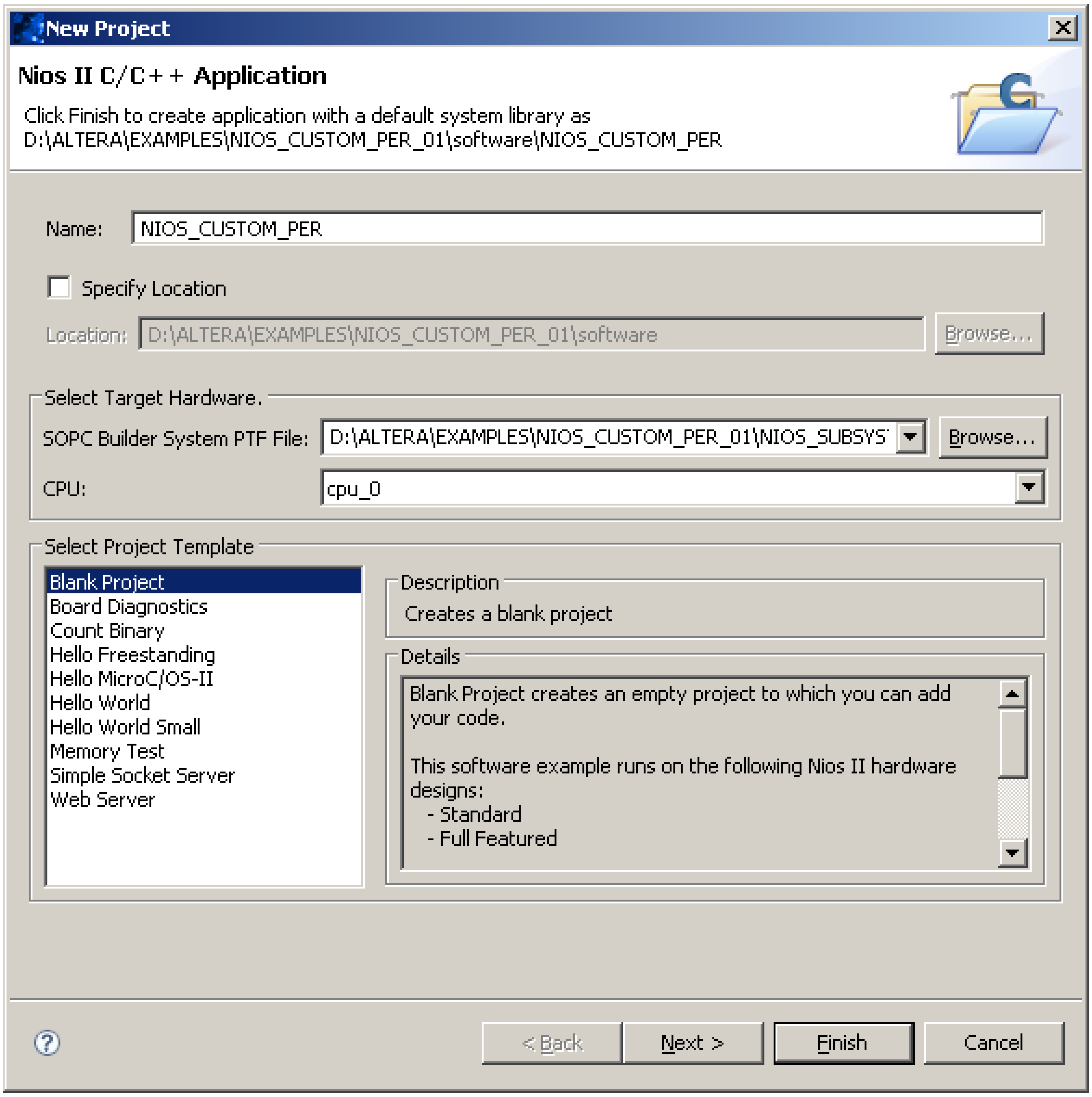
Node Name	
pio0[1..0]	Outpu
pio0[1]	Outpu
pio0[0]	Outpu
<<new node>>	

Edit: 4

	Node Name	Direction	Location	I/O Bank	VREF Group	I/O Stan
1	clk	Input	PIN_B9	8	B8_N0	2.5 V (default)
2	nreset	Input	PIN_N2	2	B2_N0	2.5 V (default)
3	pio0[1]	Output	PIN_P12	4	B4_N0	2.5 V (default)
4	pio0[0]	Output	PIN_P13	4	B4_N0	2.5 V (default)
5	<<new node>>					

   5.c Projekt skompilujeme ešte raz: Processing > Start Compilation
   5.d Tools > Programmer, hradlové pole nakonfigurujeme.
   5.e Týmto máme hardvér prichystaný.

6. Tvorba Softvéru pre NIOS II.
   6.a Spustíme Start > Programs > Altera > NIOS II EDS > NIOS II IDE
   6.b Vytvoríme nový projekt: File > New > Nios II C/C++ Application
   6.c Nakonfigurujeme Projekt:
      Zadáme názov napr.: NIOS_CUSTOM_PER
      Zadáme cieľový hardvér: Nabrowsuejeme .ptf súbor vygenerovaný SOPC builderom
      Vyberieme Blank Project



New Project

Nios II C/C++ Application

Click Finish to create application with a default system library as D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\software\NIOS_CUSTOM_PER

Name: NIOS_CUSTOM_PER

☐ Specify Location

Location: D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\software

Select Target Hardware.

SOPC Builder System PTF File: D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\NIOS_SUBSYS'

CPU: cpu_0

Select Project Template

- Blank Project
- Board Diagnostics
- Count Binary
- Hello Freestanding
- Hello MicroC/OS-II
- Hello World
- Hello World Small
- Memory Test
- Simple Socket Server
- Web Server

Description

Creates a blank project

Details

Blank Project creates an empty project to which you can add your code.

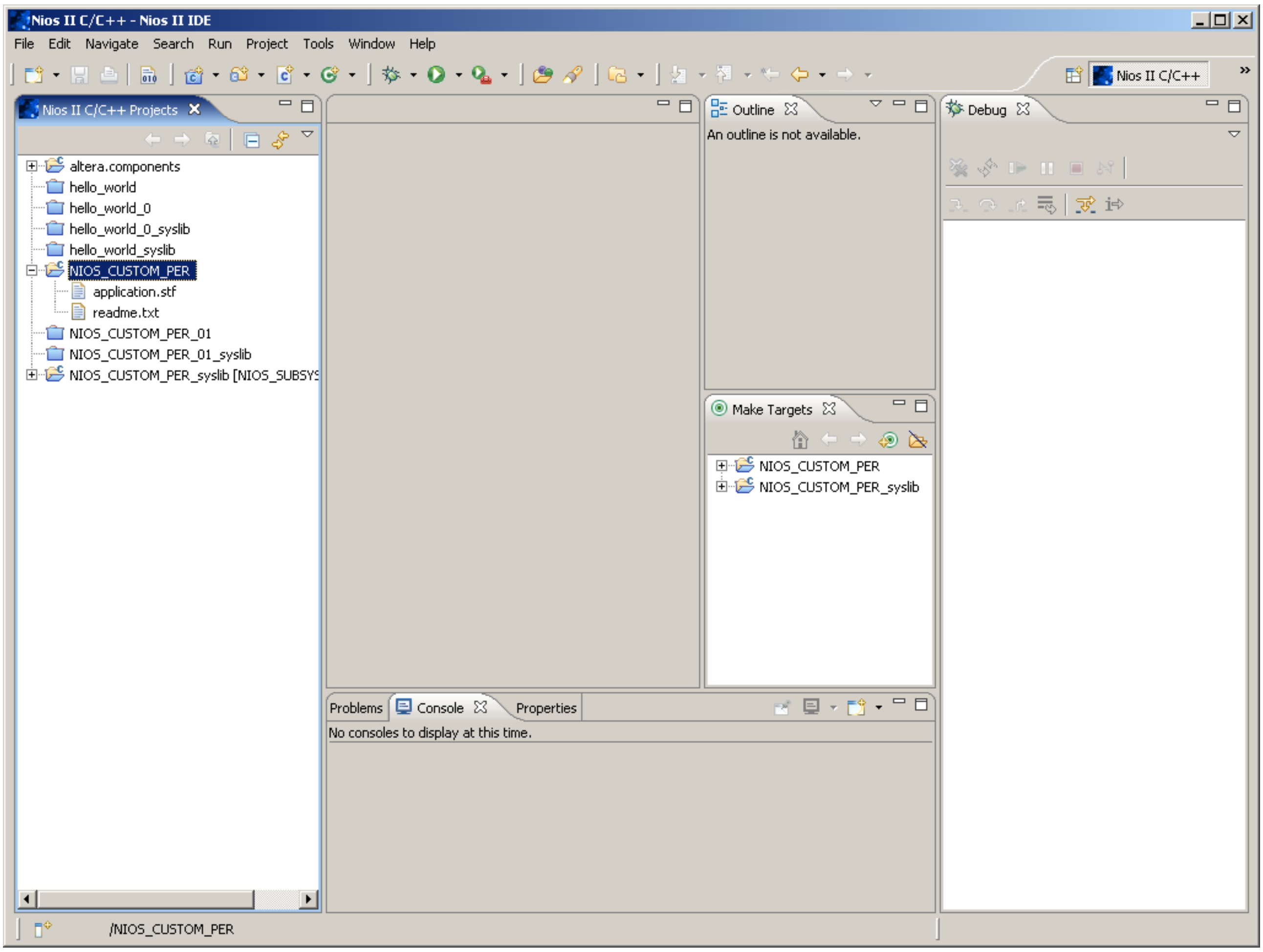
This software example runs on the following Nios II hardware designs:
- Standard
- Full Featured

6.d Stlačíme Next >

6.e Stlačíme Finish

6.f Z balíku nios_blink_src.zip skopírujeme všetky .c a .h súbory do: d:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\software\src\

6.g dané súbory pridáme do projektu: Klikneme pravým tlačidlom na myši na NIOS_CUSTOM_PER v záložke Nios II C/C++ Projects

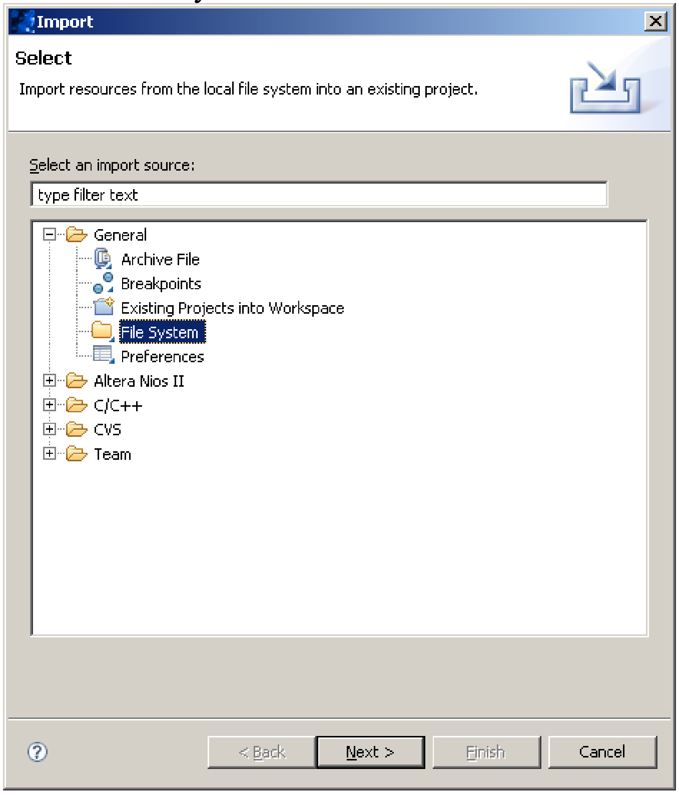


An outline is not available.

No consoles to display at this time.

6.h Vyberieme Import...

6.i Vyberieme General > File system

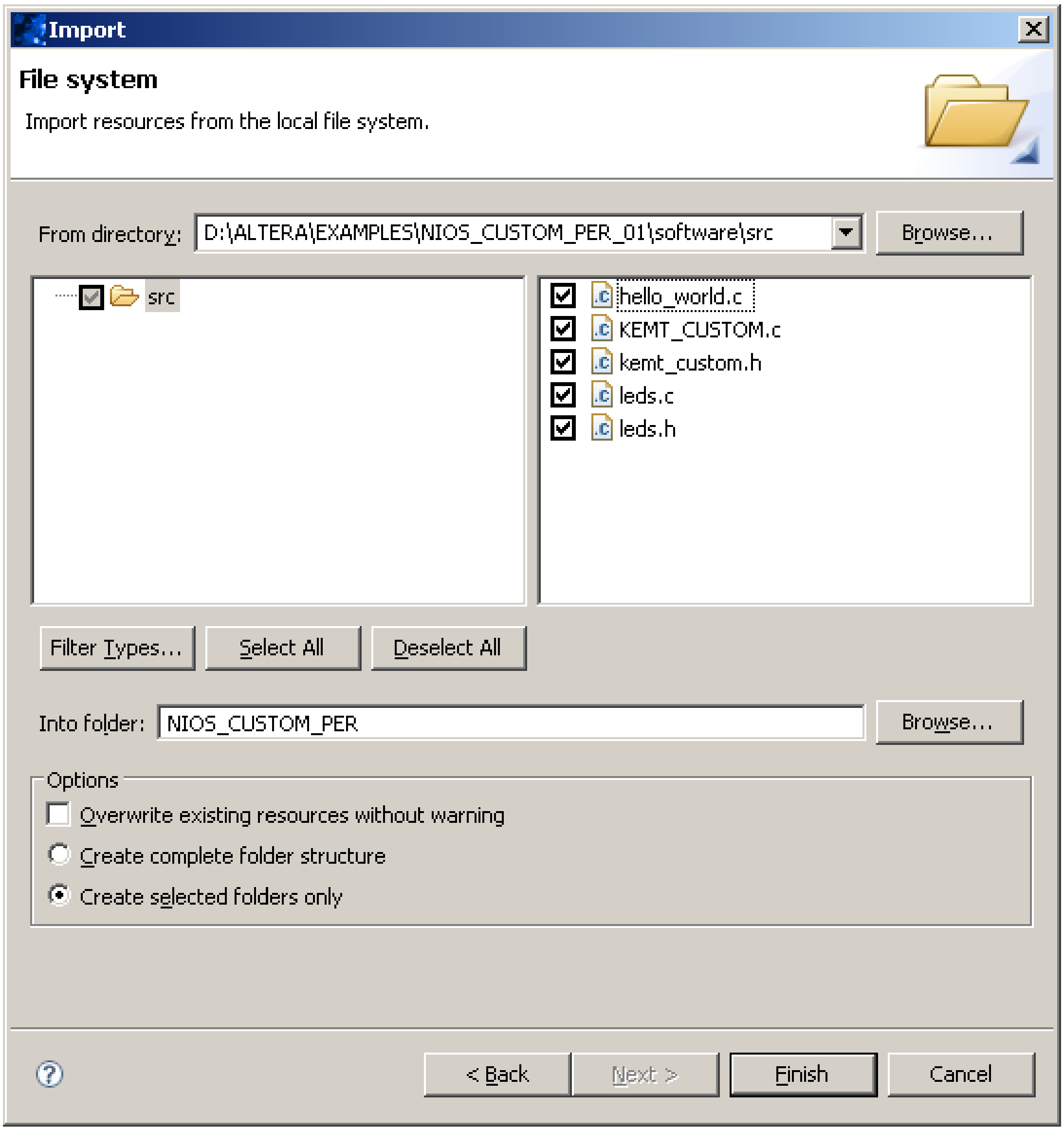


Import resources from the local file system into an existing project.

Select an import source:

type filter text

6.j Stlačíme Next >
6.k Nabrowsujeme zdrojové súbory v zložke, kam boli nahraté v kroku 6.f
Vyberieme všetky zdrojové súbory.



Import

File system

Import resources from the local file system.

From directory: D:\ALTERA\EXAMPLES\NIOS_CUSTOM_PER_01\software\src

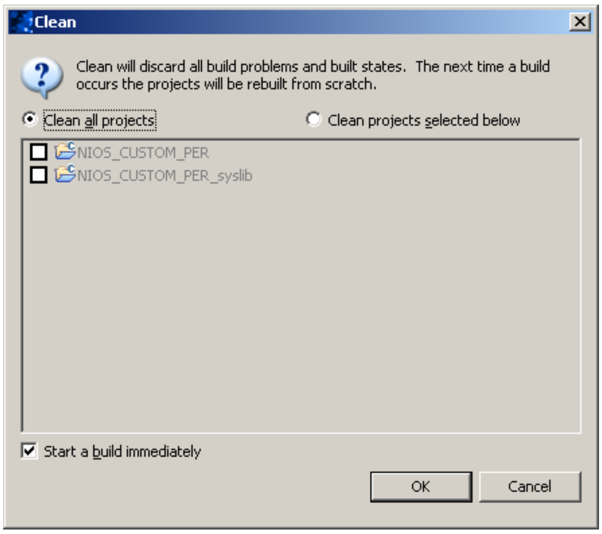
☑ src

☑ hello_world.c
☑ KEMT_CUSTOM.c
☑ kemt_custom.h
☑ leds.c
☑ leds.h

Into folder: NIOS_CUSTOM_PER

Options
☐ Overwrite existing resources without warning
○ Create complete folder structure
● Create selected folders only

6.1 Stlačíme Finish
6.m Project > Clean...



Clean

Clean will discard all build problems and built states. The next time a build occurs the projects will be rebuilt from scratch.

● Clean all projects ○ Clean projects selected below

□ NIOS_CUSTOM_PER
□ NIOS_CUSTOM_PER_syslib

☑ Start a build immediately

6.n Stlačíme OK
6.o Project > Build Project
6.p Projekt nahráme do FPGA tak, že klikneme pravým tlačidlom na myši na NIOS_CUSTOM_PER v záložke Nios II C/C++ Projects a vyberieme Run As > Nios II Hardware
6.q LED sa teraz musia rozblikať
6.r. Program môžeme ladiť (krokovať) tak, že klikneme pravým tlačidlom na myši na NIOS_CUSTOM_PER v záložke Nios II C/C++ Projects a vyberieme Debug As > Nios II Hardware