

Tutorial

FPGA Tutorial for Lego Ecar

By Karan Savant



This tutorial provides comprehensive information that will help you understand how to create a FPGA based SOPC system implementing on your FPGA development board and run software upon it.

1.1 Required Features

The Nios II processor core is a soft-core central processing unit that you could program onto an Altera field programmable gate array (FPGA). This tutorial illustrates you to the basic flow covering hardware creation and software building. You are assumed to have the latest Quartus II and NIOS II EDS software installed and quite familiar with the operation of Windows OS. If you use a different Quartus II and NIOS II EDS version, there will have some small difference during the operation. You are also be assumed to possess a DE2-115 development board (other kinds of dev.

Board based on Altera FPGA chip also supported).

The example NIOS II standard hardware system provides the following necessary components:

- * Nios II processor core, that's where the software will be executed
- * On-chip memory to store and run the software
- * JTAG link for communication between the host computer and target hardware (typically using a USB-Blaster cable)
- * LED peripheral I/O (PIO), be used as indicators

1.2 Creation of Hardware Design

This section describes the flow of how to create a hardware system including SOPC feature. 1. Launch Quartus II then select **File**->**New Project Wizard**, start to create a new project. See Figure 1-1 and Figure 1-2.



🖏 Q	uartus II - C:/Users/Karan	/Desktop/Leg	go_Ecar/Leg	o_Ecar - Lego
File	Edit View Project	Assignments	Processing	Tools Wind
Ľ	New	Ctrl+N	Lego	_Ecar
i 🖻	Open	Ctrl+O	a d	9 ×
	Close	Ctrl+F4		
2	New Project Wizard			
\$	Open Project	Ctrl+J		
	Save Project			
	Close Project			
	Save	Ctrl+S		
	Save As			
ø	Save All	Ctrl+Shift	t+S	
	File Properties			
	Create / Update			* ×
	Export		:omize	•••••
	Convert Programming Files			
D.	Page Setup			
ð.	Print Preview			Ξ
6	Print	Ctrl+P	ig file:	s
	Recent Files		•	
	Recent Projects		+ +	•
	Exit	Alt+F4		
C				

Figure 1-1 Start to Create a New Project



•	New Project Wizard	x
	Directory, Name, Top-Level Entity [page 1 of 5]	
	What is the working directory for this project?	
	C:/Users/Karan/Desktop/Lego_Ecar	
	What is the name of this project?	
	Lego_Ecar	
	What is the name of the top-level design entity for this project? This name is case sensitive and must exactly ma the entity name in the design file.	atch
	Lego_Ecar	
	Use Existing Project Settings	
-		
	< Back Next > Finish Cancel Hel	p

Figure 1-2 New Project Wizard

2. Choose a working directory for this project, type project name and top-level entity name as shown in Figure 1-3. Then click **Next**, you will see a window as shown in Figure 1-4.



•	New Project Wizard	x
	Directory, Name, Top-Level Entity [page 1 of 5]	
	what is the working directory for this project?	
	C:/Users/Karan/Desktop/Lego_Ecar	
	what is the name of this project?	_
	Lego_Ecar	
	What is the name of the top-level design entity for this project? This name is case sensitive and must exactly ma the entity name in the design file.	tch
	Lego_Ecar	
	Use Existing Project Settings	
_		
	< Back Next > Finish Cancel Help	

Figure 1-3 Input the working directory, the name of project, top-level design entity



						Add
ile Name	Туре	Library	Design Entry/Synthesis Tool	HDL Version		Add All
						Remove
						Up
						Down
					F	properties

Figure 1-4 New Project Wizard: Add Files [page 2 of 5]

3. Click **Next** to next window. We choose device family and device settings. You should choose settings the same as the Figure 1-5. Then click **Next** to next window as shown in Figure 1-6.



Device runny				Show in '	Availa	ble devices' list	
Family: Cyclone	IV E	_	•	Package		Any	•
Devices: All			Ţ	Pin counts Any		Any	-
Devices. All			Pin count: Any		Arry	_	
Target device				Speed grade: Any		Any	•
Auto device s	selected by the Fitter			Show	adva	nced devices	
	selected by the ritter				Cant	compatible only	A
Specific device	e selected in 'Availab	le devices'	list		Сору	compauble only	
🔘 Other: n/a							
Available devices:							
Name	Core Voltage	LEs	IIe		N	lemory Bits	1
Name	core rollage	11100	03	er 1/0s		1.1.1.6	
EP4CE115F23I8L	1.0V	114480	281	er I/Us	398	1312	5
EP4CE115F23I8L EP4CE115F29C7	1.2V	114480 114480	281 529	er 1/0s	398 398	1312 1312 1312	5
EP4CE115F23I8L EP4CE115F29C7 EP4CE115F29C8	1.0V 1.2V 1.2V	114480 114480 114480	281 529 529	ser 1/0s	398 398 398 398	1312 1312 1312 1312	-
EP4CE115F23I8L EP4CE115F29C7 EP4CE115F29C8 EP4CE115F29C8L	1.2V 1.2V 1.2V 1.2V 1.0V	114480 114480 114480 114480 114480	281 529 529 529	er 1/05	398 398 398 398 398	1312 1312 1312 1312 1312	
EP4CE115F23I8L EP4CE115F29C7 EP4CE115F29C8 EP4CE115F29C8L EP4CE115F29C8L EP4CE115F29C9L	1.0V 1.2V 1.2V 1.2V 1.0V 1.0V	114480 114480 114480 114480 114480 114480	281 529 529 529 529	er 1/05	398 398 398 398 398 398	1312 1312 1312 1312 1312 1312 1312	
EP4CE115F23I8L EP4CE115F29C7 EP4CE115F29C8 EP4CE115F29C8L EP4CE115F29C9L EP4CE115F29C9L	1.0V 1.2V 1.2V 1.2V 1.0V 1.0V 1.0V 1.2V	114480 114480 114480 114480 114480 114480 114480	281 529 529 529 529 529 529	er 1/05	398 398 398 398 398 398 398	1312 1312 1312 1312 1312 1312 1312 1312	
EP4CE115F23I8L EP4CE115F23I8L EP4CE115F29C8 EP4CE115F29C8L EP4CE115F29C9L EP4CE115F29I7 EP4CE115F29I8L	1.0V 1.2V 1.2V 1.2V 1.2V 1.0V 1.0V 1.0V 1.2V 1.0V	114480 114480 114480 114480 114480 114480 114480 114480	201 281 529 529 529 529 529 529 529 529	er 1/05	398 398 398 398 398 398 398 398	1312 1312 1312 1312 1312 1312 1312 1312	
EP4CE115F23I8L EP4CE115F29C7 EP4CE115F29C8 EP4CE115F29C8L EP4CE115F29C9L EP4CE115F29I7 EP4CE115F29I7 EP4CE115F29I8L	1.2V 1.2V 1.2V 1.2V 1.0V 1.0V 1.2V 1.0V 1.2V 1.0V	114480 114480 114480 114480 114480 114480 114480 114480	281 529 529 529 529 529 529 529 529	er 1/05	398 398 398 398 398 398 398 398	1312 1312 1312 1312 1312 1312 1312 1312	
EP4CE115F23I8L EP4CE115F29C7 EP4CE115F29C8L EP4CE115F29C8L EP4CE115F29C9L EP4CE115F29I7 EP4CE115F29I8L < Companion device	1.0V 1.2V 1.2V 1.2V 1.2V 1.0V 1.0V 1.0V 1.2V 1.0V 1.0V 1.0V 1.0V	114480 114480 114480 114480 114480 114480 114480 114480	281 529 529 529 529 529 529 529 529	er 1/05	398 398 398 398 398 398 398 398	1312 1312 1312 1312 1312 1312 1312 1312	
EP4CE115F23I8L EP4CE115F29C7 EP4CE115F29C8 EP4CE115F29C8L EP4CE115F29C9L EP4CE115F29C9L EP4CE115F29I7 EP4CE115F29I8L < Companion device	1.0V 1.2V 1.2V 1.2V 1.2V 1.0V 1.0V 1.0V 1.0V 1.0V 1.0V 1.0V 1.0V 1.2V 1.0V	114480 114480 114480 114480 114480 114480 114480	281 529 529 529 529 529 529 529		398 398 398 398 398 398 398 398	1312 1312 1312 1312 1312 1312 1312 1312	

Figure 1-5 New Project Wizard: Family & Device Settings [page 3 of 5]

4. Click **Next** and will see a window as shown in Figure 1-7. Figure 1-7 is a summary about our new project. Click **Finish** to finish new project. Figure 1-8 show a new complete project.



Teel Type	Teel Name	Ecomot(c)		Due Teel Auto
Design Entry (Synthesis			-	Run this to
Simulation				
Timing Applysis				Dup this to
Formal Verification			Y	
Roard-Level	Timing		-	
board-Lever	Symbol	<none></none>		
	Signal Integrity	<none></none>		
	Boundary Scan			

Figure 1-6 New Project Wizard: EDA Tool Settings [page 4 of 5]



🔇 New Project Wizard	X
Summary [page 5 of 5]	
When you click Finish, the project will be crea	ated with the following settings:
Project directory:	C:/Users/Karan/Desktop/Lego_Ecar
Project name:	Lego_Ecar
Top-level design entity:	Lego_Ecar
Number of files added:	0
Number of user libraries added:	0
Device assignments:	
Family name:	Cyclone IV E
Device:	EP4CE115F29C7
EDA tools:	
Design entry/synthesis:	<none> (<none>)</none></none>
Simulation:	<none> (<none>)</none></none>
Timing analysis:	<none> (<none>)</none></none>
Operating conditions:	
VCCINT voltage:	1.2V
Junction temperature range:	0-85 °C
<back next=""></back>	Finish Cancel Help

Figure 1-7 New Project Wizard: Summary [page 5 of 5]





Figure 1-8 A New Complete Project

5. Choose **Tools** > **SOPC Builder** to open new SOPC system wizard .See Figure 1-9 and Figure 1-10.





Figure 1-9 SOPC Builder Menu

1 Create New System - Altera SOPC I	Builder - unnamed.sopc (C:\Users\K	aran\Desktop\Lego_B	car\unnamed.sopo	:)			
File Edit Module System View Tools	Help						
System Contents System Generation	System Contents System Generation						
Component Library	Target	Clock Settings					
×	Device Family: Cyclone IV E 👻	Name	Source		MHz	Add	
Project						Remove	
⊕Bridges and Adapters ⊕Debug Components	Use C N 1 Create New S	ystem		Clock	Base	End	
Brigge and Addport Base Base Clock Base System Name:Lego_Ecar_SOPC Target HDL: Verilog Processors Processors SLS Clock Base System Name:Lego_Ecar_SOPC Target HDL: VHDL OK Cancel Clock Base System Name:Lego_Ecar_SOPC Target HDL: OK Cancel Filters Address Map Filters							
	Exit Help	Prev Next	Gene	rate			

Figure 1-10 Create New SOPC System [0]

6. Rename **System Name** as shown in Figure 1-11. Click **OK** and your will see a window as shown in Figure 1-12.



12 Create New System
System Name: Lego_Ecar_SOPC
Target HDL: 💿 Verilog
O VHDL
OK Cancel



12 Altera SOPC Builder			Mile Takent	Manual State	- C X
File Edit Module System View Tools	Help				
System Contents System Generation					
Component Library	Target	Clock Settings			
	Device Family: Cyclone IV E 🗸	Name	Source	MHz	Add
Project		clk_0	External	50.0	Remove
Library					
⊕Debug Components	Use C Name	Description	Clock	Base	End
Herrin Components Herrin Components Herrin Components Herrin Components Herrin Components Herrin L Percessor Additions Herrocessors Herrin L III New Edit Herrin Add	Remove Edit		Address Map	Filters Filte	r: Default
	Exit Help	Prev Next	Generate		

Figure 1-12 Create New System [2]

7. Click the **Name** of the **Clock Settings** table, rename **clk_0** to **clk_50**. Press Enter to complete the update. See Figure 1-13.



🖳 Altera SOPC Builder	-		Million Transaction	Monad Sec.	- 0 X
$\underline{F}ile \underline{E}dit \underline{M}odule \underline{S}ystem \underline{V}iew \underline{T}odule \underline{S}ystem \underline{S}ystem \underline{V}iew \underline{T}odule \underline{S}ystem \underline{S}ystem \underline{V}iew \underline{T}odule \underline{S}ystem \underline{S}yste$	ols <u>H</u> elp				
System Contents System Generation					
Component Library	Target	Clock Settings			
l 🔍 🗙	Device Family: Cyclone IV E 👻	Name	Source	MHz	Add
Project		clk_50	External	50.0	Remove
New component					
Avalon Verification Suite					
Bridges and Adapters	Use C Name	Description	Clock	Base	End
Interface Protocols ≡					
Eegacy Components					

Figure 1-13 Rename Clock Name

8. Choose Library > Processors > Nios II Processor to open wizard of adding cpu component. See Figure 1-14 and Figure 1-15.



🖳 Altera SOPC Builder	
File Edit Module System View Tools	Help
System Contents System Generation	
Component Library	Target
×	Device Family: Cy
New component	
Debug Components	
	Use C Name
Legacy Components	
Nios II Processor	
	•
New Edit 🖶 Add	Remove
To Do: cpu, 0: No reset vector has be	en specified for this (
To Do: cpu_0: No exception vector has	s been specified for
Warning: cpu_0: Reset vector and Ex	ception vector canno

Figure 1-14 Add Nios II Processor

😟 Nios II Processor - cp	u_0			X	
Nios	II Processor			About	
Parameter Settings					
Core Nios II Cache	es and Memory Interfaces $>$	Advanced Features	MMU and MPU Settings 🔰 JTAG Debu	ug Module > Custom Instructions >	
Core Nios II					
Select a Nios II core:					
	○Nios II/e	○Nios II/s	● Nios II/f		
Nios II Selector Guide Family: Cyclone IV E f _{system:} 50.0 MHz cpuid: 0	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 Dvnamic 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: Embe Reset Vector: Mem Exception Vector: Memo	dded Multipliers ory:	Hardware Divide Offset: 0x0 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: Deco					
Include MPU					
🕂 Warning: Reset vector	and Exception vector cannot b	e set until memory devices a	re connected to the Nios II processor		
				Cancel < Back Next > Finish	

Figure 1-15 Nios II Processor

9. Click Finish to return to main window as shown in Figure 1-16.

I Altera SOPC Builder						• X
File Edit Module System View Tools	Nios II Help					
System Contents System Generation						
Component Library	Target	Clock Settings				
×	Device Family: Cyclone IV E 👻	Name	Source	MHz		Add
Ibrary		clk_50	External	50.0		Remove
Avalon Verification Suite						
Bridges and Adapters Debug Components						
Digital Signal Processing	Use Conn Name	Descrip	tion	Clock	Base	End
	C cpu_0 instruction_ma data_master jtag_debug_m	Nios II Pr aster Avalon I Avalon I odule Avalon I	ocessor Aemory Mapped Master Aemory Mapped Master Aemory Mapped Slave	[clk] clk_50 [clk] [clk]	IRQ	0
Peripherals PLL Processor Additions Processors Nios I Processor SLS						
	•					+
New Edit 🕂 Add	K Remove		Address Map.	ү Filter	S Filter: Defau	t
□ To Do: cpu_0: No reset vector has been set vector has been been been been been been been bee	en specified for this CPU. Please param	neterize the CPU to r	esolve this issue			
To Do: cpu_0: No exception vector ha	s been specified for this CPU. Please p	arameterize the CPU	I to resolve this issue			
Warning: cpu_0: Reset vector and Exc	ception vector cannot be set until memo	iry devices are con	nected to the Nios II processo	r		
	Exit Help	Prev	ext 🕨 Generate			

Figure 1-16 Add Nios II CPU completely

10. Choose **cpu_0** and right-click then choose **rename**, after this, you can update **cpu_0** to **cpu**.See Figure 1-17 and Figure 1-18.



I Altera SOPC Builder							• X
File Edit Module System View Tools	Nios II Help						
System Contents System Generation							
Component Library	Target	Clock Set	ings				
	Device Family: Cyclone IV E	▼ Name		Source	M	Hz	Add
New component		clk_50		External	50.	.0	Remove
Library							
Debug Components	Line Conn Name		Depariation		Clock	Rapp	End
⊕…Digital Signal Processing			In a second s	_	CIUCK	Dase	End
⊕-Legacy Components		Connections		▶ ped Master	clk_50		
⊕Memories and Memory Contro		Filter		ped Master	[Clk]	IRC	2 0
⊕Merlin Components		Filler		ped Slave	[Clk]) =` 0x000080	100000x0 00
⊕PLL		Edit	Ctrl+E				
	Ø	Rename	Ctrl+R				
Nios Il Processor	×	Remove					
		Details		•			
	•	Chaus Ashibation	Charas				•
		Show Arbitration	Shares				
New Edit	Remove	Lock Base Addre	ISS	Address Map	Y Fin	Filter: Defa	ult
To Do: cpu, 0: No reset vector has be	en enerified for this CDU.	Expand All					
To Do: cpu 0: No exception vector has been To Do: cpu 0: No exception vector has	s been specified for this C			this issue			
Warning: cpu_0: Reset vector and Exc	ception vector cannot be s	Collapse All		Nios II processor			
	•	Set Color					
		Print					
4		· · · · · · ·			1		

Figure 1-17 Rename CPU name (1)



1 Altera SOPC Builder			1000	and the		- 0 X
File Edit Module System View Tools	Nins II Help					
System Contents System Occurring						
System Contents System Generation		01				
Component Library	Target	Clock Settings				_
	Device Family: Cyclone IV E 🗸	Name	Source	MH	lz	Add
New component		clk_50	External	50.	0	Remove
Library						
⊕ Avalon Verification Suite						
Debug Components						
Digital Signal Processing	Use Conn Name	Description		Clock	Base	End
Interface Protocols	🔽 🗖 сри	Nios II Proce	ssor	[clk]		
Eegacy Components	instruction_ma	aster Avalon Mem	ory Mapped Master	clk_50		
Memories and Memory Contro	data_master	Avaion Mem	ory Mapped Master		- 0x00000	
Peripherals	jiag_debug_iii		ory mapped blave	[Giv]	= 0x00000	000 00000001
Processors						
HIS I Processor						
4 III >						
	•					4
New Edit 🖶 Add	Kemove Edit		Address Map	ү Filt	Filter: De	fault
🗩 To Do: cpu: No reset vector has been	specified for this CPU. Please paramet	erize the CPU to resolve	this issue			
🗩 To Do: cpu: No exception vector has l	been specified for this CPU. Please para	ameterize the CPU to res	solve this issue			
A Warning: cpu: Reset vector and Exce	ption vector cannot be set until memory	devices are connected	to the Nios II processor			
	Exit Help	Prev Next	Generate			

Figure 1-18 Rename CPU Name (2)

11. Choose Library > Interface Protocols > Serial > JTAG UART to open wizard of adding JTAG UART. See Figure 1-19 and Figure 1-20.



<u>File Edit M</u> odule <u>S</u> ystem <u>V</u> iew	Tools	Nic
System Contents System Gener	ation	
Component Library		∩ Te
	×	De
Interface Protocols	*	
⊕⊸Interlaken		Us
⊞PCI ⊞SDI		Z
⊡-Serial	Ξ	
 Avalon-ST JTA(
Avalon-ST Seria		
SPI (3 Wire Seria		
• UART (RS-232 \$		
Memories and Memory Contr Maria Composite	-	
< <u> </u>		•
New Edit 🖶 Add.		

Figure 1-19 Add JTAG UART (1)



I JTAG UART - jtag_uart_0	٢
JTAG UART attera_avalon_itag_uart	
🔻 Block Diagram	
clock ➡ clk irq ➡ interrup1 reset ➡ reset avalon ➡ avalon_jtag_slave	
Write FIFO (Data from Avalon to JTAG)	
Buffer depth (bytes): 64 IRQ threshold: 8 Construct using registers instead of memory blocks	ш
Read FIFO (Data from JTAG to Avalon)	
Buffer depth (bytes): 64 IRQ threshold: 8 Construct using registers instead of memory blocks	
Simulated input character stream	-
Contents:	Ŧ

Figure 1-20 JTAG UART

12. Click Finish to close the wizard and return to the window as shown in Figure 1-21.



🖳 Altera SOPC Builder						
File Edit Module System View Tools	s Nios II Help					
System Contents System Generation						
Component Library	Target	Clock S	ettings			
l X	Device Family: Cyclone IV E	 Name 	Source	МН	z	Add
Ulaterface Protocole		clk_50	External	50.0)	Remove
⊕Ethernet						
	Use Conn Name		Description	Clock	Base	End
			Nios II Processor	[clk]	buse	LING
	instruct	tion_master	Avalon Memory Mapped Master	clk_50		
Avalon-ST JTAC	data_m	naster	Avalon Memory Mapped Master	[clk]	IRQ O	I I
Avalon-ST Seria	itag_de	rt 0	JTAG UART	[CIK]	■ 0x00000800	OXOCOUTI
JTAG UART SPI (3 Wire Serie	avalon_	_jtag_slave	Avalon Memory Mapped Slave	clk_50	e 0x00000000	0x000000
● UART (RS-232 \$						
tegacy Components						
Memories and Memory Conti						
4	•					۰.
New Edit 🖶 Add	🗙 Remove 🛛 🔯 Edit		Address Maj	p ү Filte	rs Filter: Default	
P To Do: cpu: No reset vector has been	n specified for this CPU. Please	parameterize the	e CPU to resolve this issue			
To Do: cpu: No exception vector has	been specified for this CPU. Ple	ease parameteriz	e the CPU to resolve this issue			
warning: cpu: Reset vector and Exce	epuon vector cannot be set until	i memory devices	are connected to the wos ii processo	11		
	Exit Help	Pr	ev Next Deneral	te		

Figure 1-21 JTAG UART

13. Choose jtag_uart_0 and rename it to jtag_uart as shown in Figure 1-22.

🖳 Altera SOPC Builder					
File Edit Module System View Tools	s Nios II Help				
System Contents System Generation					
Component Library	Target	Clock Settings			
×	Device Family: Cyclone IV E 👻	Name	Source	MHz	Add
		clk_50	External	50.0	Remove
⊞High Speed]
	Use Conn Name	Description	Clock	Base	End
⊞PCI ⊕SDI	instruction may	Nios II Processor	[Clk]		
⊖-Serial ≡	data_master	Avalon Memory Ma	pped Master [clk]	IRQ 0	r (
Avalon-ST Seria	jtag_debug_mo	dule Avalon Memory Ma JTAG UART	pped Slave [clk]		0x00000ff
ITAG UART SPI (3 Wire Seria	avalon_jtag_sk	ave Avalon Memory Ma	pped Slave clk_50	e 0x0000000	0x000000
• UART (RS-232 \$					
H-Legacy Components H-Memories and Memory Contr					
Merlin Componente					
					P
New Edit 4 Add	🔀 Remove 🛛 🐼 Edit 2		Address Map	Filter: Default	
🤝 To Do: cpu: No reset vector has been	n specified for this CPU. Please paramet	terize the CPU to resolve this i	ssue		
To Do: cpu: No exception vector has	been specified for this CPU. Please par	ameterize the CPU to resolve	this issue		
Warning: cpu: Reset vector and Exce	eption vector cannot be set until memory	/ devices are connected to the	e Nios II processor		
	Exit Help	Prev Next	Generate		

Figure 1-22 Rename JTAG UART

15. Choose Library > Memories and Memory Controllers > On-Chip > On-Chip Memory (RAM or ROM) to open wizard of adding On-Chip memory. See Figure 1-23 and Figure 1-24.



Component Library			
	×		
Bridges and Adapters			
⊡ Debug Components			
Digital Signal Processing			
Interface Protocols			
iLegacy Components			
-Memories and Memory Controllers			
External Memory Interfaces			
⊡ On-Chip			
Avalon-ST Dual Clock FIFO			
Avalon-ST Multi-Channel Shared Memi			
Avalon-ST Round Robin Scheduler			
Avalon-ST Single Clock FIFO	Ξ		
On-Chip FIFO Memory			
On-Chip Memory (RAM or ROM)			
Processor Additions			
⊕ Processors			
	Ŧ		
4			
New Edit]		

Figure 1-23 Add On-Chip Memory



1 On-Chip Memory (RAM	or ROM) - onchip_memory2_0	X
Mogacore On-Chip M altera_avalon_on	Memory (RAM or ROM) http_memory2	
🔻 Block Diagram		
	clock Clk1 avalon s1 reset reset1	
Memory type		
Туре:	RAM (Writable)	Ξ
Dual-port access		
Single clock operation		
Read During Write Mode:	DONT_CARE -	
Block type:	Auto 🗸	
▼ Size		
Data width:	32 🗸	
Total memory size:	4096 bytes	
Minimize memory block	usage (may impact fmax)	
Read latency		
Slave s1 Latency:		÷
lafe: onchin_memory2 /	0. Memory will be initialized from onchin memory? 0 bey	=
Internoryz_	o. memory will be initialized non-onenip_memoryz_o.nex	
I	Cancal	
	Cancer	

Figure 1-24 On-Chip Memory Box

16. Modify **Total memory size** to **204800** as shown in Figure 1-25. Click **Finish** to return to the window as in Figure 1-26.



📃 On-Chip Memory (RAM	or ROM) - onchip_memory2_0	3
Mogocore On-Chip M attera_avalon_or	Memory (RAM or ROM) hchip_memory2	
Block Diagram		
	onchip_memory2_0 clock ➡ clk1 avalon ➡ s1 reset ➡ reset1	
 Memory type 		
Туре:	RAM (Writable)	Ξ
Dual-port access		
Single clock operation		
Read During Write Mode:	DONT_CARE -	
Block type:	Auto 👻	
▼ Size		
Data width:	32 🗸	
Total memory size:	204800 bytes	
Minimize memory block	usage (may impact fmax)	
Read latency		
Slave s1 Latency:	1 •	÷
		_
Info: onchip_memory2_	0: Memory will be initialized from onchip_memory2_0.hex	
	Cancel	

Figure 1-25 Update Total memory size



Altera SOPC Builder File Edit Module System View Tools System Contents System Generation	s Nios II Help							x
Component Library Component Library Component Library Component Library Component Library Components Component	Target Device Family: Cyclone IV E 🔹	Clock Settings Name clk_50	Clock Settings Name Source MHz clk_50 External 50.0			Remo		
	Use Conn Name C cpu instruction_ma data_master jtag_debug.m instruction_ma data_master jtag_debug.m instruction_ma data_master jtag_uart avalon_ftag_sl S onchip_memore s1	Description Nios II Process Avalon Memor Avalon Memor JTAG UART lave Avalon Memor Avalon Memor Avalon Memor Avalon Memor Avalon Memor	sor y Mapped Master y Mapped Master y Mapped Slave y Mapped Slave ry (RAM or ROM) y Mapped Slave	Clock [clk] clk_50 [clk] [clk] clk_50 [clk]1 clk_50	Base IRQ 0 0x00000800 0x00000000 0x00000000	End IRQ 3 0x00000ff 0x0000007 0x00033ff		Tag
New Edit Add	Remove Edit	X2000 (0x0 or 0x40000 ar	Address Map e acceptable)	Filters.	Filter: Default			+
Error: cpu.data_master: onchip_m To Do: cpu: No reset vector has bee To Do: cpu: No exception vector has To Do: cpu: No exception vector has Info: onchip_memory2_0: Memory	nemory2_0.s1 cannot be at 0x2000 (0 n specified for this CPU. Please parame been specified for this CPU. Please pa will be initialized from onchip_memory2 Exit He	0x0 or 0x40000 are accepteterize the CPU to resolve anameterize the CPU to resolve anameteriz	Atable) this issue solve this issue Next D Ge	enerate				

Figure 1-26 Add On-Chip memory Completely

17. Rename **onchip_memory2_0** to **onchip_memory2** as shown in Figure 1-27.



11 Alters SODC Builder								x
File Edit Module System View Tools	s Nios II Help							
System Contents System Generation								
Component Library	Target	Clock Settings					_	
X	Device Family: Cyclone IV E 🔹	Name	Source		MHz		Add	ור
Avalon Verification Suite		clk_50	External		50.0		Remov	/e
Debug Components								
Interface Protocols	Use Conn Name	Description		Clock	Base	End	IRQ	Tag
-Memories and Memory Contra-	🔽 🗆 сри	Nios II Proces	sor	[clk]				
External Memory Interfa	instruction_mas	ster Avalon Memo	y Mapped Master	clk_50				
⊡On-Chip	data_master	Avalon Memo	ry Mapped Master	[clk]	IRQ 0	IRQ 3	31└──┐	
Avalon-ST Dual	itag_debug_mo	dule Avalon Memo	y Mapped Slave	[clk]	© 0x00000800	0x00000fff		
Avalon-ST Multi-	Jitag_uart	JIAG UARI	w Mannad Clava		0	0		
Avalon-ST Rour Avalas ST Sizel		rv2 On-Chin Memo	y mapped Slave	felk11	0x00000000	0x00000007		
Avaion-ST Singi On-Chin EIFO M	s1	Avalon Memo	v Mapped Slave	clk 50	0x00002000	0x00033fff		
On-Chip Memory								
								- 11
4	•							•
New Edit	Remove Edit		Address Map	Filters	. Filter: Default			
Error: cpu.instruction_master: on	ichip_memory2.s1 cannot be at 0x200	U (UXO or 0x40000 are	acceptable)					
To Day and Na anaster: Onchip_m	nemory2.s1 cannot be at 0x2000 (0x0)	or 0x40000 are accepta	ble)					
To Do: cpu: No reset vector has been	n specified for this CPU. Please parameter	erize the CPU to resolve	triis issue					
To bo, cpu, no exception vector has lafe: enchip, memory?: Memory will	il be initialized from onchin, memory 2 has	ameterize the CPU to res	SUIVE UTIS ISSUE					
wind. onemp_memoryz. wentry wi	in be initialized from offenip_memory2.ne)	^						
	Exit Hel	p Prev	Next 🕨 🤇 🤆	Generate				
L								

Figure 1-27 Rename On-Chip memory

18. Click **cpu** in the component list on the right part to edit the component. Update **Reset vector** and **Exception Vector** as shown in Figure 1-28. Then click **Finish** to return to the window as shown Figure 1-29.



1 Nios II Processor - cp	u				×
Mios	II Processor				About Documentation
Parameter Settings					
Core Nios II Cache	es and Memory Interfaces $>$	Advanced Features $>$	MMU and MPU Settings $ ightarrow$ JTA	G Debug Module	Custom Instructions
Core Nios II					
Select a Nios II core:					
Nios II Selector Guide Family: Cyclone IV E f _{system:} 50.0 MHz cpuid: 0	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: Embe	dded Multipliers	Hardware Divide			
Reset Vector: Mem	ory: onchip_memory2		0×0000	2000	
Exception Vector: Memo	ry: onchip_memory2	▼ Offset: 0x20	0×00002	020	
Include MMU Only include the MMU wh East TLB Miss Excention	en using an operating system t	hat explicitly supports an MI	NU		
Include MPU	vector, memory.	Ŧ	01361. 0X0		
				Cancel	Sack Next > Finish

Figure 1-28 Update CPU settings



Itera SOPC Builder File Edit Module System View Tools System Contents System Generation	s Nios II Help	0.10/				×
Component Library	Target Device Family: Cyclone IV E 🔹	Clock Settings Name Clk_50	Source External	MHz 50.0	Ren	.dd nove
Orebug Components Orget also and the state of th	Use Conn Name Control control cont	Description Nos I Processor Avalon Memory Mapi Avalon Memory Mapi JTAG UART ave Avalon Memory Mapi Vry2 On-Chip Memory Mapi Avalon Memory Mapi	Clock [clk] bed Master clk_50 bed Master [clk] [clk] [clk] bed Slave clk_50 M or ROM) [clk1] bed Slave clk_50	Base IRQ 0 0x00000800 0x00000000 0x00000000	End IRQ IRQ 31 0x00000ff 0x0000007 - 6 0x00003ff	Tag
Construction_master: on Fror: cpu.instruction_master: on Error: cpu.data_master: onchip_m Info: onchip_memory2: Memory with	Remove Edit Chip_memory2.s1 cannot be at 0x200 nemory2.s1 cannot be at 0x2000 (0x0 ill be initialized from onchip_memory2.he	00 (0x0 or 0x40000 are accept or 0x40000 are acceptable) ex	Address Map The Filt	Filter: Default		4

Figure 1-29 Update CPU settings Completely

19. Choose Library > Peripherals > Microcontroller Peripherals >PIO (Parallel I/O) to open wizard of adding PIO. See Figure 1-30 and Figure 1-31.

Ш



Component Library	
	X
⊕Bridges and Adapters	
⊡Debug Components	
Interface Protocols	
Egacy Components	
Hemories and Memory Controllers	
Merlin Components	
Peripherals	=
⊕⊡Display	
Microcontroller Peripherals	
Interval Timer	
····· PIO (Parallel I/O)	
⊞ ··Multiprocessor Coordination	
	-
New Edit	

Figure 1-30 Add PIO



및 PIO (Parallel I/O) - pio	_0		×
Megecore PIO (Par attera_avalon_	allel I/O) ^{pio}		Documentation
Block Diagram			
cl re ava con	ock • clk set • reset lon • s1 duit • external_conn) ection	
 Basic Settings 			
Width (1-32 bits):	8		
Direction:	Bidir		
	Input		
	InOut		
	Output		
Output Port Reset Value:	0x0000000000000000		
Output Register			
Enable individual bit s	etting/clearing		
Edge capture register			
Synchronously captu	ire		
	1		
			Cancel Finish

Figure 1-31 Add PIO

20. Click **Finish** to close PIO box and return to the window as shown in Figure 1-32.



🖳 Altera SOPC Builder						x
File Edit Module System View Tools Nios	s II Help					
System Contents Double Double						
System Contents System Generation						
Component Library	Target	Clock Settings				
	Device Family: Cyclone IV E -	Name	Source	MHz	A	dd
Bridges and Adapters		clk_50	External	50.0	Per	20140
H-Debug Components					Keil	love
Interface Protocols		[L				
	Use Conn Name	Description	Clock	Base	End	IRQ
Memories and Memory Controllers	🔽 🗖 сри	Nios II Processor	[clk]			
	instruction_ma	ister Avalon Memory Mapped N	/aster clk_50			
⊕ Debug and Performance	data_master	Avalon Memory Mapped N	laster [clk]	IRQ 0	IRQ 3	1 (~)
⊕⊡Display	jtag_debug_m	odule Avalon Memory Mapped S	Slave [clk]		0x00000fff	
	I I jtag_uart	JTAG UART	[clk]			
Microcontroller Peripherals	avaion_tag_si	2 On-Chip Memory (PAM or	DOM) CK_50		0x00000007	7
Interval I imer BIO (Decallel VO)		Avalon Memory Mapped S	Slave clk 50	0x00002000	0x00033fff	
Multiprocessor Coordination	✓ □ pio_0	PIO (Parallel VO)	[clk]			
⊕ -PLL ▼	s1	Avalon Memory Mapped S	Slave clk_50		0x0000001f	
الله الله الله الله الله الله الله الله						- F
New Edit	🗙 Remove 🛛 🞲 Edit		ddress Map 🛛 🍸 Filter	rs Filter: Default		
😢 Error: cpu.instruction_master: onchip_	_memory2.s1 cannot be at 0x2000 (0x	(0 or 0x40000 are acceptable)				
🔀 Error: cpu.data_master: onchip_memo	ory2.s1 cannot be at 0x2000 (0x0 or 0x4	40000 are acceptable)				
Info: onchip_memory2: Memory will be in	nitialized from onchip_memory2.hex					
	Exit Help	Prev Next 🕨	Generate			

Figure 1-32 PIO

21. Rename **pio_0** to **pio_led** as shown in Figure 1-33.

1 Altera SOPC Builder	the first the				
File Edit Module System View Tools Nios	s II Help				
System Contents System Generation					
Component Library	Target	Clock Settings			
l X	Device Family: Cyclone IV E 👻	Name	Source	MHz	Add
Bridges and Adapters Debug Components Digital Signal Processing Dinterface Protocols		clk_50	External	50.0	Remove
Egacy Components	Use Conn Name	Description	Clo	ck Base	End IRQ
	Image: state stat	Avalon Memory Avalon Memory Avalon Memory Avalon Memory JTAG UART Avalon Memory On-Chip Memory PIO (Parallel VO Avalon Memory PIO (Parallel VO Avalon Memory	Address Map [Cik]	50 IRQ 0	IRQ 31 0x00000fff 0x00000007 0x00003fff 0x000001f
Error: cpu.instruction_master: onchip_ Error: cpu.data_master: onchip_memo Info: onchip_memory2: Memory will be in	memory2.s1 cannot be at 0x2000 (0) ry2.s1 cannot be at 0x2000 (0x0 or 0x itialized from onchip_memory2.hex Exit Help	x0 or 0x40000 are accepta (40000 are acceptable)	ble) ct ▶ Generate		



Figure 1-33 Rename PIO

22. Choose **Library** > **Interface Protocols**> **Serial** >**UART** to open wizard of adding PIO. See Figure 1-34 and Figure 1-35.

Component Library	
	×
-Interface Protocols	
Ethernet	
⊞High Speed	
Interlaken	
	=
···· Avalon-ST JTAG Interface	-
···· Avalon-ST Serial Peripheral	
····· JTAG UART	
SPI (3 Wire Serial)	
UART (RS-232 Serial Port)	
E-Legacy Components	
Memories and Memory Controllers	
	Ψ.
4 III >	
New Edit	

Figure 1-34 Add UART



UART (RS-232 Serial Port) - uart_0	
UART (RS-232 Serial Port) attera_avalon_uart Documentation	
🔻 Block Diagram	
uart_0 clock ■ clk reset avalon ■ s1 conduit ■	
▼ Basic settings	=
Parity: None Data bits: 8 Stop bits: 1 Synchronizer stages: 2 Include CTS/RTS Include end-of-packet	
Baud rate	
Baud error:	
Image: State choice 0.01 Image: State choice 0.01	Ŧ
	=
Cancel	

Figure 1-35 Add UART

23. Click **Finish** to close UART box and return to the window as shown in Figure 1-36.



J Altera SOPC Builder File Edit Module System View Tools Nios II H	telp					
System Contents System Generation						
Component Library	Target	Clock Settings				
	Device Family: Cyclone IV E 👻	Name	Source		MHz	Add
i - Interface Protocols i - ASI - Ethernet - Hinh Speed		clk_50	External		50.0	Remove
	Use Conn Name Istruction_min data_master jtag_debug_m Jtag_uart avalon_itag_s S onchip_memory s1 v	Descripti aster Avaion Me Avaion Me JTAG UAF Jave Avaion Me Ory2 On-Chip M Avaion Me PIO (Paral Avaion Me UART (RS Avaion M	on mory Mapped Master emory Mapped Master emory Mapped Slave emory Mapped Slave emory Mapped Slave el VO) emory Mapped Slave -232 Serial Port) emory Mapped Slave	Clock cik_ov [cik] [cik] [cik] cik_50 [cik1] cik_50 [cik] cik_50 [cik] cik_50	Base IRQ 0 Dx00000800 Dx00000000 Dx00000000 Dx00000000 Dx000000000 Dx000000000	End IRQ 0x00000ff 0x0000007 0x00033ff 0x0000001f 0x0000003f
New Edit	🗶 Remove 🔯 Edit	Z	Address Map.	💡 Filt	ers Filter: Default	
Error: cpu.instruction_master: onchip_mer Error: cpu.data_master: onchip_memory2. Info: onchip_memory2: Memory will be initial:	nory2.s1 cannot be at 0x2000 (0x0 or s1 cannot be at 0x2000 (0x0 or 0x4000) zed from onchip_memory2.hex	0x40000 are accepta 0 are acceptable)	ble)			
	Exit Help	Prev Nex	ct 🕨 Generate			

Figure 1-36 UART

24. Rename **uart_0** to **uart_wifi** as shown in Figure 1-37.



Technische	Universität	Münche

Altera SOPC Builder File Edit Module System View Tools Nios II H	alp					
Component Library	Target Device Family: Cyclone IV E 🔹	Clock Settings Name clk_50	Source External	M 50	1Hz 0.0	Add Remove
Herlaken PCI Sol Serial Avalon-ST JTAG Interface Avalon-ST Serial Peripheral SPI (3 Wre Serial) UART (RS-232 Serial Port) Legacy Components Memories and Memory Controllers Merlin Components Merlin Components Merlin Components	Use Conn Name IIISUUCUOI_IIIE data_master jtag_debug_m jtag_uart avaio_jtag_s onchip_memo s1 pio_led s1 v uart_vifi s1	Description Avaion Men JTAG UART Jave Avaion Men JTAG UART Jave Avaion Men Proy On-Chip Me Avaion Men PIO (Paralle Avaion Men UART (RS- Avaion Men UART (RS- Marain Men	Inty mapped master nory Mapped Master nory Mapped Slave r mory Mapped Slave imory (RAM or ROM) nory Mapped Slave 202 Serial Port) mory Mapped Slave	Clock I Cin_sv [cik]	Base IRQ 0 0x00000000 0x00000000 0x000000000 0x000000	End IRO 0x000000ff 0x00000007 0x00033fff 0x0000001f
New Edit Add	Remove Remove	0x40000 are acceptable) 0 are acceptable)	 Address Map e) Generate 	Filters	Filter: Default	

Figure 1-37 Rename UART

25. Choose the Add button to add a new component in our case pwm_gen.





Figure 1-38 Add pwm_gen

Technische Universität München

12 Component Editor - new_component_hw.tcl								
File Templates								
Introduction HDL Files Signals Interfaces HDL Parameters Library Info								
Introduction								
The component editor is a tool for creating and modifying <i>IP components</i> . You can start with an existing Verilog or VHDL design block, and the component editor helps you create an IP component for that block. Alternatively, you can use the component editor to manually define the interfaces to custom logic.								
The output of the component editor is a file with name ending _hw.tcl. For an input HDL file named <i>filename.v</i> or <i>filename.vhd</i> , the component editor saves a component file named <i>filename_hw.tcl</i> in the directory where the source HDL file is located.								
To make your component appear in the component library, configure the IP Search Path to include the path to the *_hw.tcl file. To use your component in other projects or to share your component with other designers, a good practice is to move all HDL source files and the *_hw.tcl file into a separate directory and store it in a central location. You can also include supporting files, such as software drivers and simulation files, in the same directory.								
You can provide software driver files for your component hardware. If you will use your component in Nios II processor systems, you will likely need to create a software driver to integrate into the Nios II board support package (BSP).								
Add a new instance to the system								
Info: Instances of this component will export reversed-direction signals to connect to external modules. Template HDL for the component can Warning: The component has no signals.								
Help Prev Next Finish								

Figure 1-39 Component Editor: Introduction [page 1 of 6]



🗓 Component Editor - new_c	omponent_hw.tcl*				×
File Templates					
Introduction HDL Files Signals	s Interfaces HDL P	arameters Lib	rary Info		
About HDL Files					
	🗓 Open				×
File Name	Look in:	📙 Lego_Eca	ir 🔻) 🤌 📂 🛄 (
HDL Files: (No files) Add Create HD Top Level Module: (No analyzed	Recent Items Desktop My Documents Computer	.sopc_buil db Images pwm_gen pwm_gen	der _vhd _0.vhd		
	Network	File name:	pwm_gen.vhd		Open
		Files of type:	HDL Files	•	Cancel
Error: avalon_slave_0: Canr Warning: avalon_slave_0: W Warning: avalon_slave_0: H	not have both write a laster has no read or las write but no write	nd writebytee write interface edata signal	nable (only one is allowed)		
Warning: avalon_slave_0: S	lave with beginburs nent will export revers	ttransfer also sed-direction sig	needs burstcount for burst transfers gnals to connect to external modules. Template HDI	L for the compon	ent can
•					Þ

Figure 1-40 Component Editor: HDL Files [page 2 of 6]

26. Choose the **Add** button to add a hdl file in our case **pwm_gen.vhd** pwm_gen.vhd : vhdl module which implements the function of pwm generation. Its interface to Nios II is Avalon compatible.



🗓 Componer	nt Editor - pwm_gen_hw.tcl*					23
File Templates						
Introduction	HDL Files Signals Interfaces HDL Para	meters Libra	ary Info			
About HDI	Files					
Audulinou	LTHES					
	File Name	Synth	Sim	Top		
	pwm gen yhd					
HDL File	ae.					
HUL FIR	55.					
(Analyzing pwm gen.vhd Complet	ed			×	
		me incense ar				
	Info: Info: without limitation, that your	use is for the	sole pur	pose of	· · · · · · · · · · · · · · · · · · ·	
	Info: Info: programming logic devices	manufactured	l by Alter	a and sold b	(
Top Level Mode	Info: Info: Altera or its authorized dist	ibutors. Plea	se refer	to the		
	Info: Info: applicable agreement for fu	rther details.	0 2012			
	Info: Info: Command: quartus map not a	n 50 11.20.02	czula nerate h	d interface	C:/Users/Karan/Deskton/Leng	
	Info: Info: Quartus II Generate HDL Inter	ace was suc	cessful.	0 errors, 0 v	arnings	
	Info: Info: Peak virtual memory: 268 m	egabytes			=	
	Info: Info: Processing ended: Wed Jar	n 30 11:28:04	2013			
	Info: Info: Elapsed time: 00:00:02					
	Info: Info: Total CPU time (on all proce	ssors): 00:00	:01		Ţ	
	Analyzing pwm_gen.vhd Completed	. 0 Errors, 0 V	Varnings			
					Close	
			-	_		
Error avala	an elave 0: Cannot have both write and	writebuteer	able (cr	alv one is ell	(ed)	
Warning: a	valon_slave_0: Master has no read or wr	writebyteen	iable (0	ny one is all	weu)	
Warning: av	valon_slave_0. Master has no read of wr	ata sional				
Warning: av	valon_slave_0. Has write but no writed	ansfer also r	needs hu	urstcount fr	hurst transfers	
warning. at	valon_slave_o. slave with beginbursta	anarer alaor		in steount in	bulat dunatera	
		1 -				
	Help	Prev	Next		Finish	

Figure 1-41 Component Editor: HDL Files [page 2 of 6]



12 Component Editor - pwm_gen_hw.tcl*								
File Templates								
Introduction HDL Files Signals Interfaces HDL Parameters Library Info								
▶ About Signals								
Name	Interface	Signal Type	Width	Direction				
csi_l_clk	1	clk	1	input				
csi_l_rst	clock_reset	reset_n	1	input				
avs_address	avalon_slave_0	address	3	input				
avs_write	avalon_slave_0	write	1	input				
avs_l_cs	avalon_slave_0	chipselect	1	input				
avs_l_wr_data	avalon_slave_0	writedata	32	input				
coe_0_pwm1	conduit_end_0	export	1	output				
coe_0_pwm2	conduit_end_0	export	1	output				
	Add Signal Remove S	Signal						
Info: No errors or warnings								
	Help Vrev Next 🕨	Finish						

Figure 1-42 Component Editor: Signals [page 3 of 6]

27. Please check the signal type as it might not be correct initially. Change it to make it exactly as shown in the above figure.



1 Component Editor - pwm gen hw.tcl*	×
File Templates	
Introduction HDL Files Signals Interfaces HDL Parameters Library Info	
About Interfaces	
Vrite Waveforms	Î
write	
chipselect	
address A0	
writedata D0	
▶ Deprecated	
"conduit_end_0" (Conduit)	
Name: conduit_end_0 Documentation	
Type: Conduit	
Rlock Diagram	
export coe O pwm1 [1]	Ξ
export coe_O_pwm2 [1]	
	+
Add Interface Remove Interfaces With No Signals	
Info: No errors or warnings.	
Help Prev Next Finish	

Figure 1-43 Component Editor: Interfaces [page 4 of 6]



🖳 Compon	ent Editor - pwm	_gen_hw.tcl*	turne.			×	
File Templat	tes						
Introduction	HDL Files Signa	als Interfaces	HDL Parameters Lib	rary Info			
► About H	IDL Parameters						
	Name	Default Value	Edita Type	Group	Tooltip		
Parameters:	(No top level V	erilog/VHDL para	meters)				
	Add Parameter	Remove P	arameter				
	Description the OII						
	Preview the GU	l					
	errors or warnings.						
							_
		Help	Prev	Next 🕨	Finish		

Figure 1-44 Component Editor: HDL Parameters [page 5 of 6]



🗓 Component Ed	ditor - pwm_gen_h	iw.tcl*			x
File Templates					
Introduction HDL	Files Signals Inte	erfaces HDL Parameters Librar	y Info		
About Library	Info				
Parameters					
Name:	pwm_gen				
Display name:	new_component				
Version:	1.0				
Group:					•
Description:					
Created by:					
Icon:					
Documentation:	Title	URL]	
	+ -				
Into: No errors	or warnings.				
		Help Prev	Next Fir	nish	

Figure 1-45 Component Editor: Library Info [page 6 of 6]

28. Click **Finish** and you will find the **pwm_gen** component in the Library as shown below.



 Altera SOPC Builder File Edit Module System View Tools Nios II H 	elp						
System Contents System Generation							
Component Library	Target	Clock Settings					
×	Device Family: Cyclone IV E 👻	Name	Source		MHz	Add	
Project		clk_50	External		50.0	Remove	
Avalon Verification Suite	Use Conn Name	Descripti	on emory mapped master	Clock	Base	End	
Debug Components Digital Signal Processing Interface Protocols Components Legacy Components	data_master jtag_debug_m itag_uart avalon_itag_s	Avalon M Avalon M JTAG UA slave Avalon M	emory Mapped Master emory Mapped Slave RT emory Mapped Slave	_ [clk] [clk] [clk] clk_50	IRQ 0 ■ 0x00000800 ■ 0x00000000	IRQ 0x00000fff 0x0000007	
Merin Components Peripherals PLL	▼ → onchip_memory s1 ⇒ pio_led	ory2 On-Chip M Avalon M PIO (Para	Nemory (RAM or ROM) emory Mapped Slave liel VO) emory Mapped Slave	[clk1] clk_50 [clk]		0x00033fff E	
Processor Additions	v si v uart_wifi	UART (R	S-232 Serial Port)	[clk]		0x00000011	
Therefore Sols Therefore Sols There	→ s1 <	Avalon M	emory Mapped Slave	clk_50		0x000003f -	
New Edit	🗙 Remove		Address Map	Filter	s Filter: Default		
Error: cpu.instruction_master: onchip_memory2.s1 cannot be at 0x2000 (0x0 or 0x40000 are acceptable) Error: cpu.data_master: onchip_memory2.s1 cannot be at 0x2000 (0x0 or 0x40000 are acceptable) Info: onchip_memory2: Memory will be initialized from onchip_memory2.hex							
Exit Help I Prev Next December Generate							

Figure 1-46 Add pwm_gen

29. Click Finish to close pwm_gen box and return to the window as shown in Figure 1-47.



Edit Module - Altera SOPC Builder File Edit Module System View Tools Nios II He	elp							
System Contents System Generation	<u>г</u>							
Component Library	Target Device Family: Cyclone IV E -	Name	Source	MHz	Add			
Project Project New component Library Project	I pwm_gen - pwm	gen_0	Evtornal	50.0	Remove			
Avalon Verification Suite Bridges and Adapters Debug Components Digital Signal Processing Interface Protocols Legacy Components Memories and Memory Controllers Memories and Memory Controllers Peripherals PLL Processor Additions Processors SLS New Edit Add	Use Con pwm_ge Block Diagram Clock reset avalon conduit	Documen Documen pwm_gen_0 I clock_reset avalon_slave_0 conduit_end_0	tation Clock Cm_30 [Ck] [ck] [ck] [ck] ck_50 [ck] ck_50 [ck] ck_50 [ck] ck_50 [ck] ck_50 [ck]	Base IRQ 0 # 0x00000800 0 # 0x000002000 0 # 0x00000000 0	Ind IRQ x000000ff x000033fff x0000001f x0000001f x0000003f x0000003f			

Figure 1-47 Add pwm_gen

System Contents System Generation Component Library Go Y Project	Target Device Family: Cyclone IV E 🔹	Clock Settings Name clk_50	Source		MHz 50.0	Add		
New component Library Second Sec	Use Conn Name jiag_deoug_m jiag_aurt avalon_itag_si Ø ⇒ orchip_memo \$1 ⇒ pio_led \$1 > uart_wiffi \$1 > uart_wiffi \$1 > avalon_slave_	Descriptio JTAG UAR JTAG UAR lave Avalon Me PiO (Parali Avalon Me PIO (Parali Avalon Me UART (RS Avalon Me pwm_gen 0 Avalon Me	n mory mapped Slave T mory Mapped Slave emory (RAM or ROM) mory Mapped Slave el VO) mory Mapped Slave -232 Serial Port) mory Mapped Slave mory Mapped Slave	Clock [Uny [ck] clk_50 [ck1] clk_50 [ck] clk_50 [ck] clk_50 U clk_50	Base UXUUUUUUUU UXUUUUUUUU UXUUUUUUUU UXUUUUUUUU UXUUUUUUUU UXUUUUUUUU UXUUUUUUUUU UXUUUUUUUU UXUUUUUUUU UXUUUUUUUU UXUUUUUUUU UXUUUUUUU UXUUUUUU UXUUUUUU UXUUUUUU UXUUUUUUU UXUUUUUUU UXUUUUUU UXUUUUUUU UXUUUUUUU UXUUUUUUU UXUUUUUUUU UXUUUUUUUUUUU UXUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU	End UX00000077 0x0000007 0x00003ff 0x0000003f 0x0000003f		
P-SLS Image: Add::::::::::::::::::::::::::::::::::								

Figure 1-48 pwm_gen



30. Choose **System > Auto-Assign Base Addresses** as shown in Figure 1-48. After that, you will find that there is no error in the message window as shown in Figure 1-49.

Altera SOPC Builder File Edt Module System View Tools Nios I He System Contents Assign Base Addresses	p							
Component Libra Project Library New component	Target Device Family: Cyclone IV E	Clock Settings Name clk_50	Source External	MHz 50.0	Add Remove			
A-valor Verification Suite Bridges and Adapters Debug Components Digital Signal Processing Interface Protocols Legacy Components Memories and Memory Controllers Merlin Components Peripherals PLL Processor Additions Processors SLS	Use Conn Name µay_ueouy_m □ itag_uart avalon_itag_si □ onchip_memod s1 □ pio_led s1 □ uart_wifi s1 □ pwm_gen avalon_slave_	Description JTAG UART JTAG UART lave Avaion Memory Mappe Pry2 On-Chip Memory (RAM Avaion Memory Mappe PIO (Parallel VO) Avaion Memory Mappe UART (RS-232 Serial P Avaion Memory Mappe pwm_gen 0 Avaion Memory Mappe	Clock LUM] [LUM] [CR] [CR] d Slave clk_50 or ROM) [clk] d Slave clk_50 [ck] [ck] d Slave clk_50 ort) [clk] d Slave clk_50 utild Slave clk_50 utild Slave clk_50 utild Slave clk_50	Base UX000000000 CX00000000000000000000000000	End vxvvvvvvrr 0x0000007 0x00033fff 0x0000001f 0x0000003f 0x0000005f *			
New Edit Edit Edit Edit Edit Edit Filters Filter: Edit Edit Edit Edit Edit Filters Error: pwm_gen: Can't have same name as its top level hdl module Error: Error: cpu.instruction_master: onchip_memory2.s1 Error: cpu.data_master: onchip_memory2.s1 cannot be at 0x2000 (0x0 or 0x40000 are acceptable) Error: cpu.data_master: onchip_memory2.s1 cannot be at 0x2000 (0x0 or 0x40000 are acceptable) Info: onchip_memory2.s1 cannot be at 0x2000 (0x0 or 0x40000 are acceptable) Info: onchip_memory2.s1 cannot be at 0x2000 (0x0 or 0x40000 are acceptable) Info: onchip_memory2.s1 cannot be at 0x2000 (0x0 or 0x40000 are acceptable)								

1 Altera SOPC Builder						
File Edit Module System View Tools Nios II He	alp					
System Contents System Generation						
Component Library	Target	Clock Settings				
	Device Family: Cyclone IV E 👻	Name	Source		MHz	Add
Project		clk_50	External		50.0	Remove
New component						
Library						
		Description		011	Deer	5 -1
⊞Bridges and Adapters	Use Conn Name	Description		Clock	Base	End
Debug Components Digital Signal Processing	instruction ma	aster Avalon Memor	v Mapped Master	clk 50		Â
Interface Protocols	data_master	Avalon Memor	y Mapped Master	[clk]	IRQ 0	IRQ
Legacy Components	jtag_debug_m	nodule Avalon Memor	y Mapped Slave	[clk]		0x00080fff
Memories and Memory Controllers	I I I I I I I I I I I I I I I I I I I	JTAG UART		[clk]		=
Merlin Components	→ avaion_tag_s	slave Avalon Memor	y Mapped Slave	CIK_50	⇒ 0x00081050	0x00081057
		Avalon Memor	v Manned Slave	clk 50	0-0-0040000	0x00071fff
Processor Additions	☑ □ pio_led	PIO (Parallel V))	[clk]		
+Processors	→ s1	Avalon Memor	y Mapped Slave	clk_50	Ox00081040 ■ Ox0081040 ■ Ox008104 ■ Ox08104 ■ Ox008104	0x0008104f
terest state stat	I uart wifi	UIADT (DS 23)	Carial Dort)	Tel M		
New Edit	🗙 Remove 🔯 Edit		Address Map	Filter	s Filter: Default	
Info: onchip_memory2: Memory will be initialized	d from onchip_memory2.hex					
	Exit Help	Prev Next	Generate			

Figure 1-49 No errors or warnings



31. Click **Generate** and then pop a window as shown in Figure 1-50. Click **Save** and the generation start. If there is no error in the generation, the window will show successful as shown in Figure 1-51.

File Edit Module System View Tools Nos I Help System Contents System Contents Component Library Ibrary Save in: Save in: <th>🖳 Altera SOPC Builder</th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>	🖳 Altera SOPC Builder								
System Contents System Generation Component Library If Save in: Image: Save in: Lego_Ecar Save in: Lego_Ecar Image: Save in: Save_Duilder Ibrary Image: Save_Duilder Ibrary Ibrary Ibrary Ibrary Ibrary Ibrary Ibrary Ibrary Ibrary Ibrary Ibrary Ibr	File Edit Module System View Too	ls Nios II Help							
Component Lbrary Save Project Add Wer component Lbrary Lbrary Save in: Lego_Ecar Before component Before component Lbrary Before component Bridges and Adapters Body Signal Processing Builder Spinal Processing Desktop Before and Memory Controller Desktop Before	System Contents System Generation								
	System Contents System Generation Component Library Project Wew component Library Solution Verification Suite Object O	2 Save Save in: Recent Items Desktop My Documents Computer Computer	Lego_Ec	ar ilder Lego_Ecar SOPC Builder System Fil	les (* sopc)		Save Cancel	z se IRQ 0 0x00080800 0x00040000 0x00040000 0x00081040 Filter: Default	Add Remove
Exit Help I Prev Next December 2010			Exit	Help Prev	Next 🕨	Generate			

Figure 1-50 Generate SOPC



🖳 Altera SOPC Builder - Lego Ecar.sopc* (C:\Users\Karan\Desktop\Lego Ecar\Lego Ecar.sopc)	_ O _ X
File Frit Module System View Tools Nins II Help	
The Columnodule system view hous most net	
System Contents System Generation	
Options	
System module logic will be created in Verilog.	
Simulation. Create project simulator files. Run Simulator	
Nios II Tools	
Nios II Software Build Tools for Eclipse	
א בעריכטי דבטי דבטי דער אוווין ערטוויענטורעווט איז אוויענער נעטין דער איז איז אוויענער איז איז איז איז איז איז	
# 2013.01.30 12:01:09 (*) Generating Quartus symbol for top level: Lego_Ecar_SOPC	<u> </u>
# 2013.01.30 12:01:09 (*) Generating Symbol C:/Users/Karan/Desktop/Lego_Ecar/Lego_Ecar_SOPC.bsf	
# 2013.01.30 12:01:09 (*) Creating command-line system-generation script: C:/Users/Karan/Desktop/Lego_Ecar/Lego_Ecar_SOPC_generation_script	
# 2013.01.30 12:01:09 (*) Running setup for HDL simulator: modelsim	
# 2013.01.30 12:01:10 (*) Completed generation for system: Lego_Ecar_SOPC.	
# 2013.01.30 12:01:10 (*) THE FOLLOWING SYSTEM ITEMS HAVE BEEN GENERATED:	
SOPC Builder database : C:/Users/Karan/Desktop/Lego_Ecar/Lego_Ecar_SOPC.ptf	
System HDL Model : C:/Users/Karan/Desktop/Lego_Ecar/Lego_Ecar_SOPC.v	
System Generation Script : C:/Users/Karan/Desktop/Lego_Ecar/Lego_Ecar_SOPC_generation_script	
# 2013.01.30 12:01:10 (*) SUCCESS: SYSTEM GENERATION COMPLETED.	E
🚯 Info: System generation was successful.	-
	4
Info: onchip memory2: Memory will be initialized from onchip memory2 hex	
Exit Help I Prev Next D Generate	

Figure 1-51 Generate SOPC Complete

32. Click Exit to exit the SOPC Builder and return to the window as shown in Figure 1-52.



Figure 1-52

33. Choose File > New to open new files wizard. See Figure 1-53 and Figure 1-54. (E) File Edit View Project Assignments Processing Tools Window Help 🗋 🚅 🔚 🎒 🎒 🐰 🖻 🛍 🗠 🗠 🗍 niosi_uart - 🕱 🔮 🖉 🥙 🚳 📼 🕨 🤻 🍗 🝈 🗶 👟 🌘 roject Navigator ₽× New Quartus[®] II Software v12.1: Entity Download Subscription Edition Free 30-Day Trial Cyclone IV E: EP4CE115F29C7 🖻 🔜 niosii_top 品 x 🙁 New SEE V New Quartus II Project Design Files AHDL File Block Diagram/Schematic File EDIF File Qsys System File State Machine File 🖹 Files d Design Units A Hierarchy SystemVerilog HDL File Tcl Script File ₽× asks VHDL File Verilog HDL File =low: Compilation Customize... Memory Files Hexadecimal (Intel-Format) File Task . Memory Initialization File ? 🔺 🕨 Compile Design Verification/Debugging Files ? In-System Sources and Probes File \triangleright Analysis & Synthesis = ? Fitter (Place & Route) Logic Analyzer Interface File SignalTap II Logic Analyzer File Assembler (Generate programming files) A Other Files ? TimeQuest Timing Analysis AHDL Include File EDA Netlist Writer Block Symbol File Device (Open Programmer) Chain Description File • Synopsys Design Constraints File Text File × Type Message

Figure 1-53 New Block Diagram File

34. Choose **New Block Diagram File** and click **OK** to return to the window as shown in Figure 1-54. Figure 1-54 show a blank Block Diagram File



🐫 Quartus II - C:/Users/Karan/Desktop/niosii_uart/niosii_u	rt - niosii_uart	
File Edit View Project Assignments Processing To	ils Window Help 😨	
Project Navigator 🗗 🗙	Tel Block1.bdf	
Entity	│ 🖳 🚔 ¼ № 🖻 🗠 ♀ │ 🖓 ℝ ♥ ♥ A ♀ 炎 ▾ 🔲 ᄀ ᄀ ᄀ " " ヽ ヽ ヽ ⊡ ○ ヽ ヽ !?!?! / 4 ╡ &	
Cyclone IV E: EP4CE115E29C7		
•		
Hierarchy		
		E
Tasks 🗗 🛪		
Elour Compilation		
Costomzett		
Task		
2 4 Comple Design		
Analysis & Synthesis		
b Fitter (Place & Route)		
Accombler (Generate programming files)		
Assembler (denied de programming nes		
EDA Netliet Writer		
Program Device (Open Programmer)		
< III >		•
*		
Type Message		
-		
×Ω <		۱.
System / Processing / Extra Info / Info / Warnin	2 / Critical Warning / Error / Suppressed / Flag /	
Message:		▼ Locate
		£ 40 08/ 00:00:00
		0,40 0% 00:00:00

Figure 1-54 A blank Block Diagram File

35. Double click on the empty block diagram file to add the Symbols. Add the **Lego_Ecar_SOPC** from the Project folder.



braries:		
 ✓ Project ☑ Lego_Ecar_SOPC ▷ □ c:/altera/11.0sp1/quartus/libraries/ 	Lego_Ecar_SOPC clk_50 reset_n	
۲	out_port_from_	the_pio_led[70]
ame:	coe_o_pwmz_trom_	me_pwm_gen_u
.ego_Ecar_SOPC	rxd_to_the_uart_wifi txd_fr	om_the_uart_wifi
Repeat-insert mode	inst	
Insert symbol as block		
Launch MegaWizard Plug-In		
MegaWizard Plug-In Manager		

Figure 1-55

36. Add other symbols to make the following block diagram



Figure 1-56

37. Choose **Save** Icon in the tool bar. There will appear a window as shown in Figure 1-57. Click **Save.**



Save As	0.90			1		×
	Save in:	Documents		•	← 🗈 💣 📰▼	
C		Name	*		Date modified	Туре
Recent	Places	퉬 Atmel Stud	lio		12/28/2012 7:08 AM	File folder
necon	10003	📗 Outlook Fil	les		1/12/2013 11:34 AM	File folder
		i The KMPla	yer		12/28/2012 1:52 PM	File folder
Desk	top	퉬 Updater5			12/21/2012 1:17 PM	File folder
<u> </u>	-	📗 Visual Stud	lio 2010		12/27/2012 7:05 PM	File folder
6		Lego_Ecar_	Тор		1/30/2013 12:12 PM	BDF File
Libra	aries					
Comp	outer					
Netu	V					
INCLW	VOIK					
		•				•
		File name:	Lego Ecar Top		•	Save
		<u> </u>				Connel
		Save as type:	Block Diagram/Schemati	c Files (*.b	dt) 🔽	Cancel
			Add file to current proj	ect		
·						

Figure 1-57 Save Bdf File

38. Choose **Processing** > **Start Compilation**. Figure 1-58 shows the compilation process. Note: You must select **Lego_Ecar_Top.bdf** as the top level entity. This can be done by right clicking and selecting the same.





Figure 1-58: Compilation

Note: In the compilation, if there is the error which shows "Error: The core supply voltage of '1.0v' is illegal for the currently selected part.", you should modify the text "set_global_assignment –name NOMINAL_CORE_SUPPLY_VOLTAGE 1.0V" to "set_global_assignment –name NOMINAL_CORE_SUPPLY_VOLTAGE 1.2V"

39. A window that shows successful compilation will appear as shown in Figure 1-59.



Figure 1-59: Successful

40. Choose **Assignments > Pin Planner** to open pin planner as shown in Figure 1-60. Figure 1-61 show blank pins.



😤 Quartus II - C:/Users/Kar	an/De	esktop/Lego	o_Ecar/Lego	o_Ecar	- Lego_E	tar	
<u>File E</u> dit <u>V</u> iew <u>P</u> roject	Assi	gnments	P <u>r</u> ocessing	<u>T</u> ools	<u>W</u> indov	v <u>H</u> elp	P
🗅 🖻 🗑 🎒 🖓	2	Device					- 🐹
Project Navigator	2	<u>S</u> ettings				Ctrl+Shift	+E ments/L
G Files		TimeQuest	t Timing Anal	yzer <u>W</u>	/izard		5
El Lego_Ecar_SOPC.q//Documents/Lego_SOPC.q	1	<u>A</u> ssignmen	t Editor			Ctrl+Shift	+A
bbr / /	1	Pi <u>n</u> Planne	r			Ctrl+Shift	+N :::::
	₿.	<u>R</u> emove A <u>B</u> ack-Anno Imp <u>o</u> rt Ass Export Ass	ssignments otate Assignr signments	 nents			_50
Hierarchy D Eilee	-	Assignmen	t <u>G</u> roups				
Tasks	_ 品	<u>L</u> ogicLock I Design Par	Regions Wine	dow ow		Alt+L Alt+D	
Flow: Compilation		-	Customize	1t			P
Task	_						
✓ ▲ ► Compile Desi	an					· · · · · · ·	
V 🕨 🕨 Analysis	- & Syn	thesis					· · · · · · · · · · · ·
V 🕨 🕨 Fitter (P	ace &	Route)			:::::	: : : : : :	
V 🕨 🕨 Assembl	er (Ger	nerate progr	amming files				
V 🕨 🕨 TimeQue	st Tim	ing Analysis					
🔰 🕨 EDA Net	list Wri	iter			:::::	: : : : : :	
Program Dev	ice (O	<u>nen Pr</u> ogram	nmer)				

Figure 1-60 Pins menu

0-0 <mark>19</mark>	•	4							
۲ <u>۹</u>	ž	Named: * 🔻 🗮 E	dit: 🗶 🗹 📃				Filter: F	Pins: all	_
<u>ња</u>	ď	Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	-
		CLOCK_50	Input				2.5 V (default)		
P		🐵 ENA	Output				2.5 V (default)		
3		KEY[0]	Input				2.5 V (default)		=
1		LEDR[7]	Output				2.5 V (default)		
8		LEDR[6]	Output				2.5 V (default)		
20		LEDR[5]	Output				2.5 V (default)		
		LEDR[4]	Output				2.5 V (default)		
0		LEDR[3]	Output				2.5 V (default)		
1	2	LEDR[2]	Output				2.5 V (default)		
	II P		0.12.11	111			5 E 0 / J. C. M		
•	~						0	% 00:00:00	

Figure 1-61 Blank Pins

41. Input Location value as shown in Figure 1-62.





Figure 1-62

42. Close the **pin planner**. Restart compilation the project.



1.3 Download Hardware Design to Target FPGA

This section describes how to download the configuration file to the board.

Download the FPGA configuration file (i.e. the SRAM Object File (.sof) that contains the NIOS II standard system) to the board by performing the following steps:

- 1. Connect the board to the host computer via the USB download cable.
- 2. Apply power to the board.
- 3. Start the NIOS II IDE.
- 4. After the welcome page appears, click **Workbench**.
- 5. Choose **Tools->Quartus II Programmer**.
- 6. Click Auto Detect. The device on your development board should be detected automatically.
- 7. Click the top row to highlight it.
- 8. Click Change File.
- 9. Browse to the myfirst_niosii project directory.
- 10. Select the programming file (myfirst_niosii.sof) for your board.

11. Click **OK**.

12. Click **Hardware Setup** in the top, left comer of the Quartus II programmer window. The Hardware Setup dialog box appears.

13. Select USB-Blaster from the Currently selected hardware drop-down list box.

Note: If the appropriate download cable does not appear in the list, you must first install drivers for the cable. Refer to Quartus II Help for information on how to install the driver. See Figure 1-63.

lardware Settings elect a programming	JTAG S	ettings e setup to u	ise when prog	ramming devices	. This programming
ardware setup appli Currently selected ha Available hardware	es only to rdware: items	USB-Blast	t programmer er [USB-0] vare er [USB-0]	window.	
Hardware		Relatediation	Server	Port	Add Hardware
USB-blaster			LOCAI	USB-0	Remove Hardware



Figure 1-63 Hardware Setup Window

14. Click Close.

15. Turn on the **Program/Configure** option for the programming file.(See Figure 1-64 for an example).

16. Click Start.

File Edit View Processing Tools Window Help 🗣	
Aradware Setup USB-Blaster [USB-0] Mode: JTAG Progress:	
Enable real-time ISP to allow background programming (for MAX II and MAX V devices)	
File Device Checksum Usercode Program/ Configure Verify Blank- Check	Examine
Lego_Ecar_time_limited.sof EP4CE115F29 0089E54F FFFFFFFF	
Auto Detect	
X Delete	
Add File	
Change File	•
Save File	A
Add Device	
	E
TDO	
	+

Figure 1-64 Quartus II Programmer

The Progress meter sweeps to 100% after the configuration finished. When configuration is complete, the FPGA is configured with the Nios II system, but it does not yet have a C program in memory to execute.



Chapter 2 NIOS II IDE Build Flow

This Chapter covers build flow of Nios II C coded software program. The Nios II IDE build flow is an easy-to-use graphical user interface (GUI) that automates build and makefile management. The Nios II IDE integrates a text editor, debugger, the Nios II flash programmer, the Quartus II Programmer, and the Nios II C-to-Hardware (C2H) compiler GUI. The included example software application templates make it easy for new software programmers to get started quickly. In this section you will use the Nios II IDE to compile a simple C language example software program to run on the Nios II standard system configured onto the FPGA on your development board. You will create a new software project, build it, and run it on the target hardware. You will also edit the project, re-build it, and set up a debug session.

2.1 Create the hello_world Example Project

In this section you will create a new NIOS II C/C++ application project based on an installed example. To begin, perform the following steps in the NIOS II IDE:

1. Return to the NIOS II IDE.

Note: you can close the Quartus II Programmer or leave it open in the background if you want to reload the processor system onto your development board quickly.

2. Choose File > Switch Workspace to switch workspace. See Figure 2-1 and Figure 2-2.



Edit Refactor Navigate	e Search Run Proj
New	Alt+Shift+N
Open File	
Close	Ctrl+W
Close All	Ctrl+Shift+W
Save	Ctrl+S
Save As	
Save All	Ctrl+Shift+S
Revert	
Move	
Rename	
Refresh	F5
Convert Line Delimiters To	I
Print	Ctrl+P
Switch Workspace	
Import	
Export	
Properties	Alt+Enter
1 hello_world.c [hello_world]	_0]
2 motor_setting.h [hello_wo	rld_0]
3 motor_setting.c [hello_wor	'ld_0]
4 motor_settings.c [hello_wo	orld_0]
Evit	

💽 Workspace	e Launcher
Select a w	orkspace
Nios II IDE sto Choose a wor	ores your projects in a folder called a workspace. rkspace folder to use for this session.
Workspace:	C:\Users\Karan\Desktop\Lego_Ecar\Software Browse
?	OK Cancel



Figure 2-2 Switch Workspace (2)

3. Choose File->New->NIOS II C/C++ Application to open the New Project Wizard.

4. In the New Project wizard, make sure the following things:

Select the **Hello World** project template.

• Give the project a name. (hello_world_0 is default name)

• Select the target hardware system PTF file that locates in where the previously created hardware project resides as shown in Figure 2-3.

New Project	
Nios II C/C++ Application	- G
Click Finish to create application \Lego_Ecar\software\Lego_Ecar_	with a default system library as C:\Users\Karan\Desktop
Namer Lego Fear piecijann	
Rame. Lego_ccar_mosnapp	
Location: C:\Users\Karan\De	sktop\Lego Ecar\software Browse
Select Target Hardware.	
SOPC Builder System PTF File:	C:\Users\Karan\Desktop\Lego_Ecar\Lego_Ecar_SOPC.ptf 👻 Browse
CPU:	cpu 🔹
Select Project Template	
Count Binary Hello Freestanding	Description Prints 'Hello from Nios II'
Hello MicroC/OS-II Hello World	Details
Hello World Small Memory Test	Hello World prints 'Hello from Nios II' to STDOUT.
Memory Test Small Simple Socket Server	This example runs with or without the MicroC/OS-II RTOS and requires an STDOUT device in your system's hardware.
Web Server (RGMII)	For details, click Finish to create the project and refer to the
?	< Back Next > Finish Cancel

Figure 2-3 Nios II IDE New Project Wizard



5. Click **Finish**. The Nios II IDE creates the **Lego_Ecar_niosiiapp** project and returns to the Nios II C/C++ project perspective. See Figure 2-4.

🛃 Nios II C/C++ - hello_world.c - Nios II IDE			
File Edit Refactor Navigate Search Project Tools Run Window Help			
🔁 - 🛛 🚔 🖬 🔯 - 😋 -	G • 🕸 • O • 94 • 😕 🖋 🙃 •	🐓 🔹 🖓 🔹 🏷 🔶 🔹	🖹 🂽 Nios II C/C++
💽 Nios II C/C++ Projects 🛛 📃 🗖	c hello_world.c ⋈	🗖 🗄 Outline 🛛 👘 🗖	□ Welcome 🛛 💧 🗇 🖓 🗖
 	<pre>/* * "Hello World" example. * * This example prints 'Hello f * the Nios II 'standard', 'ful * designs. It runs with or wit * device in your system's hard * The memory footprint of this</pre>	Jª R S o ⊽ Stdich o main	Welcome to the Altera Nios II IDE Overview Overview of the Nios II IDE.
	<pre>* using the standard reference * * For a reduced footprint vers * to reduce the memory footpri * "small_hello_world" template * */</pre>	● Make Targets ☆ □ □ ☆ ⇔ ⇒ ⊛ ≽ □ € Large Forz pignijane	Tutorials Tutorials and cheat sheets to get started quickly.
	<pre>#include <stdio.h> int main() { printf("Wello from Nice III); }</stdio.h></pre>	u csgo_ccar_niosiiapp_syslib	What's New New features in this release.
	return 0; Problems 🖾 Console Properties	· · · · · · · · · · · · · · · · · · ·	
	0 errors, 0 warnings, 0 infos	Dath	
	Description F	lesource Path	eclipse
4 III >	<	•	
□◆	Writable Sm	art Insert 1:1	

Figure 2-4 Nios II IDE C++ Project Perspective for hello_world_0

When you create a new project, the NIOS II IDE creates two new projects in the NIOS II C/C++ Projects tab:

■ Lego_Ecar_niosiiapp is your C/C++ application project. This project contains the source and header

files for your application.

■ Lego_Ecar_niosiiapp _syslib is a system library that encapsulates the details of the Nios II system hardware.

Note: When you build the system library for the first time the NIOS II IDE automatically generates files useful for software development, including:

- Installed IP device drivers, including SOPC component device drivers for the NIOS II hardware system
- Newlib C library, which is a richly featured C library for the NIOS II processor.
- NIOS software packages which includes NIOS II hardware abstraction layer, NicheStack TCP/IP



Network stack, NIOS II host file system, NIOS II read-only zip file system and Micrium's μ C/OS-II real time operating system(RTOS).

- **system.h**, which is a header file that encapsulates your hardware system.
- alt_sys_init.c, which is an initialization file that initializes the devices in the system.
- Lego_Ecar_niosiiapp.elf, which is an executable and linked format file for the application located in Lego_Ecar_niosiiapp folder under Debug.

2.2 Build and Run the Program

In this section you will build and run the program to execute the compiled code.

To build the program, right-click the **Lego_Ecar_niosiiapp** project in the Nios II C/C++ Projects tab and

choose **Build Project**. The **Build Project** dialog box appears and the IDE begins compiling the project. When compilation completes, a message 'Build complete' will appear in the Console tab.

The compilation time varies depending on your system. See Figure 2-5 for an example.



Figure 2-5 Nios II IDE Build Completed

After compilation complete, right-click the **hello_world_0** project, choose **Run As**, and choose **NIOS II Hardware**. The IDE begins to download the program to the target FPGA development board and begins execution. When the target hardware begins executing the program, the message '**Hello from Nios II!**' appears in the NIOS II IDE Console tab.

