MAX1000

Internal Flash Lab



Software and hardware requirements to complete all exercises Software Requirements: Quartus® Prime Lite or Standard Edition version 18.0 or 18.1 Hardware Requirements: ARROW MAX1000 Board

1. Introduction

This tutorial provides comprehensive information to help you understand how to use the internal flash feature of the MAX10 and how to run it on your MAX1000 board. The Nios II processor is a soft intellectual property (IP) processor that you download (along with other hardware components that comprise the Nios II system) onto an Intel FPGA. At the end, you will know how to setup your Quartus project in order to boot your Nios processor from the internal flash memory.

Lab Notes: Many of the names that the lab asks you to choose for files, components, and other objects in this exercise must be spelled exactly as directed. This nomenclature is necessary because the pre-written software application includes variables that use the names of the hardware peripherals. Naming the components differently can cause the software application to fail. There are also other similar dependencies within the project that require you to enter the correct names.

2. Getting Started

The first objective is to ensure that you have all the necessary hardware items and software installed so that the lab can be completed successfully. Below is a list of items required to complete this lab:

- MAX1000 Board (10M08SAU169C8G)
- USB Cable
- Lab files: Internal_Flash_lab_template: Template files are required to complete the lab. Includes: nios_subsystem.qsys, main.c
- Quartus Prime 18.0 Lite was used for this lab. Previous/newer versions should work (If no Quartus Prime is installed, refer to MAX1000 User Guide for instructions)
- Installed Arrow USB Drivers (If not, refer to MAX1000 User Guide for instructions)
- Personal computer or laptop running 64-bit Linux / Windows 7 or later with at least an Intel i3 core (or equivalent), 4GB RAM and 12 GB of free hard disk space
- A desire to learn!

wow

3. Design Flow

MAX10 devices contain on-chip flash which segmented to two types:

- Configuration Flash Memory (CFM) to store hardware configuration settings for MAX10 FPGA.
- User Flash Memory (UFM) to store user software application.

You can boot and configure the Nios II soft core processor to execute code from the on-chip flash within the FPGA.



The above diagram shows the typical design flow for the system design with the internal flash memory. The system definition is done with Platform Designer. The Nios II IDE uses the system description to create a new project for the software application. The output of the FPGA design is a FPGA image that is used to configure the FPGA. The output of the software flow is an executable which runs on the Nios II processor. At the end, the Quartus puts these two files together in a single configuration image.

4. Project with MAX1000

4.1 Quartus Prime project

4.1.1 Create a new Quartus Prime project

- 4.1.1.1 If not already open, from the Start menu or the Desktop, open the Quartus Prime 18.0 Lite software.
- 4.1.1.2 Create a new project using the New Project Wizard: **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 	
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	
< Back Next > Finish Cancel Help	,

4.1.1.3 Click Next.

4.1.1.4 Configure the New Project Wizard directory, name and top-level entity information:

- Enter a directory in which you will store your Quartus project files for this design, for example, C:/MAX1000/Internal_Flash_lab
- Specify the name of the project: **nios_subsystem**
- Specify the name of the top-level entity: **nios_subsystem**



New Project Wizard	
Directory, Name, Top-Level Entity	
Vhat is the working directory for this project?	
C:/MAX1000/Internal_Flash_lab	
Vhat is the name of this project?	
ios_subsystem	
Vhat is the name of the top-level design entity for this project? This name is case sensitive and must exactly m esign file.	atch the entity name in the
nios_subsystem	

4.1.1.5 Click Next.

4.1.1.6 On the Project Type page, select **"Empty project"** and click **Next**.

New Project Wizard					
Project Type					
Select the type of project to create.					
Empty project					
Create new project by specifying project files and l	ibraries, target (device family an	id device, and E	DA tool settings.	
O Project template					
Create a project from an existing design template. software, or download design templates from the	You can choose <u>Design Store</u> .	from design te	mplates install	ed with the Quartu	s Prime

4.1.1.7 On the Add Files page, add source file to the project by clicking on the button and browse into the lab files folder where you will locate the provided design files and add **nios_subsystem.qsys**.

New Project Wizard								_
Add Files								
Select the design files yo	ou want to include in	the proje	ect. Click Add All to add all desi	gn files in the p	project directory to	the project.		
Note: you can always add design files to the project later. File name: Ad Add Add								
ile name:							Add	
L						×	Add All	
-ile Name	Туре	Library	Design Entry/Synthesis Tool	HDL Version			Remove	
nios_subsystem.qsy	/s Qsys System File						Up	
							Down	
							Propertie	
							Flopente	2
pecify the path names c	of any non-default lib	oraries.	User Libraries					
			< Back	Next >	Finish	Cancel	Help	

4.1.1.8 Click Next.

4.1.1.9 Specify Family and Device Settings. Use pull-down menus to select MAX10 family or enter the part number in the Name Filter text box. The part number is **10M08SAU169C8G**.

Device Board							
elect the family and 'ou can install additi	device you want to t ional device support (arget for co with the In	ompilation. stall Devices com	mand on the	Tools m	ienu.	
o determine the ver	sion of the Quartus P	rime softw	are in which you	r target devic	e is supp	orted, refer to	the <u>Device Support List</u> webpage
Device family				Show in 'A	vailable	devices' list	
Family: MAX 10 (D	A/DF/DC/SA/SC)		•	Package:		UFBGA	•
Device: All			•	Pin count:		169	÷
Target device				Core spee	d grade:	8	•
O Auto device sele	ected by the Fitter			Name filte	r:	10M08SAU1	59C8G
Specific device s	elected in 'Available o	levices' list	t	Show a	dvanced	devices	
Other: n/a							
vailable devices:							
Name	Core Voltage	LEs	Total I/Os	GPIOs	Mer	no ry Bits	Embedded multiplier 9-bit elen
10M08SAU169C8G	3.3V	8064	130	130	38707	2 4	8
10M08SAU169C8G	ES 3.3V	8064	130	130	38707	2 4	18
<							>

4.1.1.10 Click Finish. MAX1000 Internal Flash Lab

4.2 Design entry

Overview: In this module you will generate the basic configuration file for the FPGA.

4.2.1 Platform Designer

4.2.1.1 Double click on nios_subsystem top-level entity in Project Navigator.



4.2.1.2 Platform Designer will open. The Nios processor system is already done, currently no need to change the design.

It is a simple design, which contains a 12MHz clock source, a pll which generates 50MHz for the system, a Nios II/e processor, on-chip RAM which will store the program code in the beginning, jtag-uart and system ID interface for the programming and a parallel I/O interface for the LEDs.

P Catalog 💠 🗕 🖬 🗖	12 -	iystem Co	ntents 88	Address Map	23 Interconnect Requi	irements 83								
× 🐼		× *	System	nics_subsyste	m Path: dk12mhz									
ject	+	Use C	Connections		Name	Description	Export	Clock	Base	End	IRQ	Tags	Opcode Name	
👷 New Component	💷				clk12mhz	Clock Source						((,
System	×			0	⊢ dk_in	Clock Input	clk	exported						
asic Functions					dk_in_reset	Reset Input	reset	di Darka						
9 9	x		_		dk remet	Report Output	Double-click to export Double-click to export	OKTSMUS						
terface Protocols		м	1		E el	ALTPLL Intel EPGA IP	Double Cack to Caport	_						
enory Interfaces and Controllers	-	- I			inck_interface	Clock Input	Double-click to export	clk12mhz						
ocessors and Peripherals	z		+	+	inck_interface_rese	et Reset Input	Double-click to export	(inck_interf						
sys Interconnect			- H	-	+ pl_slave	Avaion Memory Mapped Slave	Double-click to export	Inck_interf	₽ 0x5020	0x502f				
1-State Components		_			< 00	Clock Output	Double-click to export	pl_c0						
and any Program						Nos II Processor								
					+ dk	Clock Input	Double-click to export	pll_c0						
			TU		 reset data master 	Keset unput	Double-click to export Double click to export	[cik]						
					< cata_master	Avaion Memory Mapped Master	Double-click to export	[dk]						
				-	+ ing	Interrupt Receiver	Double-click to export	[dk]		TRO O	180 31			
				\rightarrow	< debug_reset_reque	st Reset Output	Double-click to export	[dk]						
			11++		debug_mem_slave	Avaion Memory Mapped Slave	Double-click to export	[dk]	# 0x4800	0x4fff				
				~	custom_instruction_	m Custom Instruction Master	Double-click to export							
					onchip	On-Chip Memory (RAM or ROM) Intel .								
					+ dk1	Clock Input	Double-click to export	pll_c0						
Edt 🕂 Add					* \$1	Avaion Memory Mapped Slave	Double-click to export	[dk1]	# 0x2000	0x3fff				
				T	Teleci	TAC LIART Intel EDCA 70	Double-cack to export	fox 1						
erarchy 💠 Device Family 🖏 💶 🗖 🗖		۳I.			the second	Clock Input	Double-click to export	oll c0						
a aubavaten (nice aubavaten cava)			+	++	+ reset	Reset Input	Double-click to export	(dk)						
dk					avaion_jtag_slave	Avaion Memory Mapped Slave	Double-click to export	[dk]	@ 0x5038	0x503£				
led				•	< irg	Interrupt Sender	Double-click to export	[dk]						
reset					sysid	System ID Peripheral Intel FPGA IP								
itan uart			1+		+ dk	Clock Input	Double-click to export	pll_c0						
leds				+	+ reset	Reset Input	Double-click to export	[dk]						
nios					 control_save 	Avaion Memory Mapped Slave	Double-click to export	[ok]	- 0x5030	0x5037				
onchip					a dk	Clock Target	Double click to expect	01.00						
pil					+ reset	Reset locut	Double-click to export	Telk1						
Connections			· •	-	* s1	Avaion Memory Mapped Slave	Double-click to export	(dk)	# 0x5000	0x501f				
				୍ୟ	external_connection	n Conduit	led							
	· ۱	de alto	ү 🗑 Cur	ent filter:										
	× .	Accesses	22											_
	0=	-teasingles	nut.											-
	i ype		3 Info Messar		Message									
	E.		nine subsy	tem itan ua	et ITAG LIART IR input clock	open to be at least double (2v) the operation	in frequency of ITAG TCK on ho	ard						
		,	nics_subsy	term enabled	Susteen 10 is not assigned	a domatically. Edit the Sectors 10 narranal	s to provide a unique TD							
			mus_subsy	seen.sysid	aystem to is not assigned	a accomencemy, cost the system (D paramete	i w provide a unique au							
	0	9	nios_subsy	tem.sysid	Time stamp will be automa	atically updated when this component is ger	erated.							

MAX1000 Internal Flash Lab

www.arrow.com January 2019

- 4.2.1.3 Select **Generate → Generate HDL...** from the menu or alternately click **Generate HDL...** button on the bottom right of the Platform Designer window.
- 4.2.1.4 On the Generation window, enter the following information.
 - Create HDL design files for synthesis: VHDL
 - Uncheck Create timing and resource estimates for third-party EDA synthesis tools.
 - Uncheck Create block symbol file (.bsf)
 - Create simulation model: None

🐇 Generation		×
▼ Synthesis		
Synthesis files are used to compile the system in a Quartus project. Create HDL design files for synthesis: VHDL TCreate timing and resource estimates for third-party EDA synthesis tools.		
Create block symbol file (.bsf)		
* Simulation		
The simulation model contains generated HDL files for the simulator, and may include simulation-only features.		
Simulation scripts for this component will be generated in a vendor-specific sub-directory in the specified output directory.		
Follow the guidance in the generated simulation scripts about how to structure your design's simulation scripts and how to use the <i>ip-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-setup-set</i>	<i>imulation</i> and	
Create simulation model: None V		
* Output Directory		
Path: C:/MAX1000/Internal_Flash_lab/nios_subsystem		
	Generate	Cancel

4.2.1.5 Click Generate.

4.2.1.6 When the generate process completed, click Close.



4.2.1.7 Click **Finish** button on the bottom right of the Platform Designer window.

4.3 Compile design

4.3.1 Analysis and Synthesis

4.3.1.1 Run Analysis and Synthesis by clicking on ^k button on the toolbars, or **Processing** → **Start** → **Analysis and Synthesis**.

There should be no errors. If there are errors, they should be fixed before continuing. If there are no errors the compilation task windows should look like this:



4.3.2 Pin Assignments

4.3.2.1 Open **Pin Planner** by clicking on [⋖] button on the toolbars, or **Assignments** → **Pin Planner**.



www.arrow.com January 2019

4.3.2.2 In the bottom table, type **PIN_H6** in Location column of the clk_clk.

Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	Jurrent Strengt	Slew Rate	Differential Pair	trict Preservatio
altera_reserved_tck	Input	Π			2.5 V (default)		12mA (default)			
in altera_reserved_tdi	Input				2.5 V (default)		12mA (default)			
altera_reserved_tdo	Output				2.5 V (default)		12mA (default)	2 (default)		
in_ altera_rrved_tms	Input				2.5 V (default)		12mA (default)			
	Input	PIN_H6	2	B2_N0	2.5 V (default)		12mA (default)			
led_export[7]	Output				2.5 V (default)		12mA (default)	2 (default)		
led_export[6]	Output				2.5 V (default)		12mA (default)	2 (default)		
led_export[5]	Output				2.5 V (default)		12mA (default)	2 (default)		
led_export[4]	Output				2.5 V (default)		12mA (default)	2 (default)		
led_export[3]	Output				2.5 V (default)		12mA (default)	2 (default)		
led_export[2]	Output				2.5 V (default)		12mA (default)	2 (default)		
led_export[1]	Output				2.5 V (default)		12mA (default)	2 (default)		
led_export[0]	Output				2.5 V (default)		12mA (default)	2 (default)		
inreset_reset_n	Input				2.5 V (default)		12mA (default)			
< <new node="">></new>										

4.3.2.3 Repeat the previous step with the following assignments:

Node Name	Pin Location
altera_reserved_tck	PIN_G2
altera_reserved_tdi	PIN_F5
altera_reserved_tdo	PIN_F6
altera_reserved_tms	PIN_G1
led_export[0]	PIN_A8
led_export[1]	PIN_A9
led_export[2]	PIN_A11
led_export[3]	PIN_A10
led_export[4]	PIN_B10
led_export[5]	PIN_C9
led_export[6]	PIN_C10
led_export[7]	PIN_D8
reset_reset_n	PIN_E6

Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	Jurrent Strengt	Slew Rate	Differential Pair	trict Preservatio
in altera_reserved_tck	Input	PIN_g2	1B	B1_N0	2.5 V (default)		12mA (default)			
🖳 altera_reserved_tdi	Input	PIN_f5	1B	B1_N0	2.5 V (default)		12mA (default)			
altera_reserved_tdo	Output	PIN_f6	1B	B1_N0	2.5 V (default)		12mA (default)	2 (default)		
altera_reserved_tms	Input	PIN_g1	1B	B1_N0	2.5 V (default)		12mA (default)			
💾 clk_clk	Input	PIN_H6	2	B2_N0	2.5 V (default)		12mA (default)			
led_export[0]	Output	PIN_a8	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
led_export[1]	Output	PIN_a9	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
led_export[2]	Output	PIN_a11	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
led_export[3]	Output	PIN_a10	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
led_export[4]	Output	PIN_b10	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
led_export[5]	Output	PIN_c9	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
led_export[6]	Output	PIN_c10	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
led_export[7]	Output	PIN_d8	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
reset_reset_n	Input	PIN_e6	8	B8_N0	2.5 V (default)		12mA (default)			
< <new node="">></new>										

4.3.2.4 Double click in the I/O Standard column for any pins to open a drop-down list and change the 2.5V (default) to the specific I/O standard.

Node Name	Pin I/O Standard
altera_reserved_tck	3.3-V LVTTL
altera_reserved_tdi	3.3-V LVTTL
altera_reserved_tdo	3.3-V LVTTL
altera_reserved_tms	3.3-V LVTTL
clk_clk	3.3-V LVTTL
led_export[0]	3.3-V LVTTL
led_export[1]	3.3-V LVTTL
led_export[2]	3.3-V LVTTL
led_export[3]	3.3-V LVTTL
led_export[4]	3.3-V LVTTL
led_export[5]	3.3-V LVTTL
led_export[6]	3.3-V LVTTL
led_export[7]	3.3-V LVTTL
reset reset n	3.3-V Schmitt Trigger



4.3.2.5 Close the Pin Planner, the settings are automatically saved.

4.3.3 Compiling the Design

- 4.3.3.1 Open the device settings window from **Assignments** → **Device...** and click **Device and Pin Options**.
- 4.3.3.2 Click to the **Configuration** category.
- 4.3.3.3 Set configuration mode to Single Uncompressed Image with Memory Initialization (256kbits UFM).

General	Configuration								
Configuration	Specify the device config	uration scheme and the config	guration device.						
Programming Files									
Unused Pins	Configuration scheme:	Internal Configuration							
Dual-Purpose Pins	Configuration mode:	Single Uncompressed Image v	with Memory Initialization (256Kbits UFM)						
Capacitive Loading	Configuration device								
I/O Timing	comparation acres								
/oltage	Use configuration d	Auto		~					
Pin Placement	ose configuration d	evice.	Device Options						
Error Detection CRC	Configuration device I/	Queltage		~					
CvP Settings	Configuration device i/	O voltage:							
Partial Reconfiguration	Force VCCIO to be o	compatible with configuration	I/O voltage						
	V/D Operation mode								
	VID Operation mode								
	Configuration pin:		Configuration Pin Options						
	Generate compressed	bitstreams							
	A stive seriel shark serves								
	Active senal clock source								
	Enable input tri-state	on active configuration pins in	n user mode						
	Description:								
	Specifies the configurati	ion mode used with the config	guration scheme for configuring the device.						

- 4.3.3.4 Press **OK** to close Device and Pin Options window. Press again **OK** to close Device window.
- 4.3.3.5 Start Compilation by clicking on ► button on the toolbars, or Processing → Start Compilation.

wow

There should be no errors. If there are errors, they should be fixed before re-compilating. The 100% in the lower right corner or a green checkmark next to the Compile Design in the Compilation task window indicates that the compilation was successful.

😘 Quartus Prime I	Lite Edition - C:/MAX	(1000/Internal_Flas	h_lab/nios_subsystem - nios_subsyster	n					- ø ×
File Edit View	Project Assign	ments Processie	ng Tools Window Help					Search a	altera.com 🤇
🗋 🗖 🖳 🗲		nios_subsystem	•	- I III + + + + + + + + + + + + + + + +		_			
Project Navigator	A Hierarchy	▼ Q # # ×	~	Compliation Report - nios_subsyste	em	×		IP Catalog	0 8 ×
A MAX 10: 10M	Entity:Instance		Flow Summary	Flow Summary <	C	- 2010		🗙 🕌 Installed IP	~ =
 A nos_subs Tacks Y > b Co Y > b Co<!--</td--><td>Compliation Task mpile Design Analysis & Synthesi Filter Pilez & Route Assembler (Gonrad Toring Analysis EDA Netlett Witer Setting and Device (Open F</td><td>> • = = 0.5 × s programming file Programme)</td><td>The Setting The Non-Orbari Global Se The Non-Orbari Global Se The Non-Orbari Global Se The OS Summary The Non Lig Analysis S Synthesis The Non Lig Set Analysis S Synthesis The Non-National Section 2015 The Non-National Section 2015 The Non-National Section 2015 The Non-National National Section 2015 The National Nationa</td><td>New Salsa Benston Name Topshend Erikky Name Fandy Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Jans Toski Ryke Genenis Toski Ryke Holdes Toski Ryke Holdes Toski Ryke Holdes Toski Ryke Holdes ADC Macks</td><td>Successful - Fri Jan 16 (60)05 Iniotublystem Iniotublystem MAX:10 10M055AUG6CG Final 1,714 / 8,064 (21 %) 947 10 / 130 (8%) 0 75,800 (357,072 (20 %) 0 / 10 (0%) 1 / 1 (0 %) 0 / 1 (0 %)</td><td>2019 Lits Edison</td><td></td><td>Project Directory System System Sectors Sectors Sectors Momony interfaces and C Processors and Peripher Momony interfaces and C Processors and Peripher Sectors for Partner IP Search for Partner IP</td><td>iontrollers de</td>	Compliation Task mpile Design Analysis & Synthesi Filter Pilez & Route Assembler (Gonrad Toring Analysis EDA Netlett Witer Setting and Device (Open F	> • = = 0.5 × s programming file Programme)	The Setting The Non-Orbari Global Se The Non-Orbari Global Se The Non-Orbari Global Se The OS Summary The Non Lig Analysis S Synthesis The Non Lig Set Analysis S Synthesis The Non-National Section 2015 The Non-National Section 2015 The Non-National Section 2015 The Non-National National Section 2015 The National Nationa	New Salsa Benston Name Topshend Erikky Name Fandy Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Genenis Toski Ryke Jans Toski Ryke Genenis Toski Ryke Holdes Toski Ryke Holdes Toski Ryke Holdes Toski Ryke Holdes ADC Macks	Successful - Fri Jan 16 (60)05 Iniotublystem Iniotublystem MAX:10 10M055AUG6CG Final 1,714 / 8,064 (21 %) 947 10 / 130 (8%) 0 75,800 (357,072 (20 %) 0 / 10 (0%) 1 / 1 (0 %) 0 / 1 (0 %)	2019 Lits Edison		Project Directory System System Sectors Sectors Sectors Momony interfaces and C Processors and Peripher Momony interfaces and C Processors and Peripher Sectors for Partner IP Search for Partner IP	iontrollers de
< v		>	< >					+ Add	
5 All O	🔬 👗 📐 🕥	< <filter>></filter>		66 Find 66 Find Next					
Type T > 0 332 > 0 322 > 0 322 > 0 322 > 0 322 0 322 > 0 322 > 0 293 *	v Message 146 Worst-case 146 Worst-case 146 Worst-case 114 Report Met 102 Design is 102 Design is Quartus Pr 000 Quartus Pr	recovery sl removal sla atminum pul astability: not fully co not fully co rime Timing A rime Full Com	ack is 49.275 ck is 0.463 se width slack is 49.423 Found 3 synchronizer chai mstrained for setup requi mstrained for hold requi malyzer was successful. 0 pilation was successful.	ns. rements eenets errors, 22 warnings 0 errors, 50 warnings					Ŷ
System	Processing (439)								>

4.3.4 Configuration

4.3.4.1 Connect your MAX1000 board to your PC using an USB cable. Since the Arrow USB Blaster should be already installed, the Window's Device Manager should display the following entries are highlighted in red (port number may differ depending on your PC):



4.3.4.2 Open the Quartus Prime Programmer from **Tools** → **Programmer** or double click on Program Device (Open Programmer) from the Task window.

Programmer - C:/M/ File Edit View Pro	AX1000/Internal_Flash_lab ocessing Tools Wind	o/nios_subsystem - ni low Help	os_subsystem -	[nios_subsyste	m.cdf]*						Search al	L tera.com	×
Aardware Setup	Arrow-USB-Blaster [USB	10]			Mode:	JTAG		Ŧ	Progress	s:			
Enable real-time ISP	• to allow background pro File	gramming when avai Device	Checksum	Usercode	Program/	Verify	Blank-	Examine	Security	Erase	ISP		
≣™ Stop					Configure		Check		Bit		CLAMP		
Huto Detect													
Add File													
Save File													
T ^h Up													
¹ ™ Down													

4.3.4.3 Click **Hardware Setup...** and double click **Arrow-USB-Blaster** entry in the Hardware Setup tab. The Currently selected hardware should now show Arrow-USB-Blaster [USB0] (depending on your PC, the USB port number may variant).

lardware Settings JTAG Set	tings		
elect a programming hardware ardware setup applies only to t	setup to use the current pr	when programmin ogrammer windov	ng devices. This programming w.
urrently selected hardware:	Arrow-USB-BI	aster [USB0]	
Hardware Arrow-USB-Blaster	Server Local	Port USB0	Add Hardware Remove Hardware

4.3.4.4 Click Close.



4.3.4.5 Make sure the hardware setup is Arrow-USB-Blaster [USB0] and the mode is JTAG. Click **Auto Detect**.

Programmer - C:/1 File Edit View	MAX1000/Internal_Flash_lab Processing Tools W	o/nios_subsystem - ı 'indow Help	nios_subsystem -	[nios_subsyste	m.cdf]*					Search	altera.con	×
🚖 Hardware Setu	p Arrow-USB-Blaster	[USB0]		Mode:	JTAG			• Prog	ress:			
Enable real-time	ISP to allow backgroun	d programming w	hen available									
⊨ ³ b Start	File	Device	Checksum	Usercode	Program/	Verify	Blank-	Examine	Security	Erase	ISP	
🛍 Stop					compare		CHECK		Dit		CLAMP	
💏 Auto Detect												
× Delete												
Add File												
Change File												
Add Device												
1 ^ካ Up												
ી ^પ ે Down												

- 4.3.4.6 If the configuration has been added by default, you can skip the following steps and continue with the 4.3.4.11 point.
- 4.3.4.7 Select **10M08SA** device and click **OK** in the pop-up window.



- 4.3.4.8 Click Change File... or double click <none> to choose the programming file.
- 4.3.4.9 Navigate to <project_directory>/output_files/ and select the nios_subsystem.sof file.

4.3.4.10 Click Open.

4.3.4.11 Make sure the Programmer shows the correct file and the correct part in the JTAG chain and check the Program/Configure checkbox.

Programmer - C:	/MAX1000/Internal_Flash_la	b/nios_subsystem -	nios_subsystem	- [nios_subsyste	em.cdf]*						_		×
File Edit View	Processing Tools Win	dow Help									Search al	tera.com	6
🔔 Hardware Setup	Arrow-USB-Blaster [USI	80]			Mode:	JTAG		-	Progress	s:			
Enable real-time	ISP to allow background pro	ogramming when av	vailable										
M Start	File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMP		
Stop	output_files/nios_subs	10M08SAU169	00194C19	00194C19									
💏 Auto Detect													
🗙 Delete													
📩 Add File													
隆 Change File													
Save File													
Add Device													
1 ¹⁰ Up													
^{‡%} Down													
	10M08SAU TDO	169											
	•												

4.3.4.12 Click **Start** to program the board. When the configuration is complete, the Progress bar should show 100% (Successful).

Progress: 100% (Successful)	
-----------------------------	--

4.4 Software Design

Overview: In this section, you will use the Nios II Software Build Tools (SBT) for Eclipse to quickly create a board support package (BSP) and a C software application to run on the Nios II processor.

4.4.1 Create a new software project

- 4.4.1.1 From the main Quartus Prime window, start STB from **Tools** → **Nios II Software Build Tools** for Eclipse.
- 4.4.1.2 The Eclipse Workspace Launcher will open. Click **Browse...** and choose the directory of your project. In this case it was C:\MAX1000\Internal_Flash_lab.

🖨 Workspac	e Launcher	Х
Select a w	orkspace	
Eclipse store Choose a w	es your projects in a folder called a workspace. orkspace folder to use for this session.	
Workspace:	C:\MAX1000\Internal_Flash_lab V Browse	
Use this a	s the default and do not ask again OK Cancel	
	OK Cancel	

4.4.1.3 Click **OK** and the Eclipse will open.



MAX1000 Internal Flash Lab

- 4.4.1.4 Select File → New → Nios II Application and BSP from Template.
- 4.4.1.5 Click to select the **nios_subsystem.sopcinfo** from your project directory and name the project **internal_flash_lab**. Make sure you select **Hello World** from the Templates.

Nios II Application and BSP from	n Template	_ [ı ×			
Nios II Software Examples						
Create a new application and boa template	rd support package based on a software example					
Target hardware information						
SOPC Information File name:	C:\MAX1000\Internal_Flash_lab\nios_subsystem.sopcinfo					
CPU name:	CPU name: v					
Application project						
Project name: internal_flash	lab					
Use default location						
			_			
Project location: C:\MA	1000\Internal_Flash_lab\software\internal_flash_lab					
Project template						
Templates	Template description					
Blank Project Board Diagnostics	Hello World prints 'Hello from Nios II' to STDOUT.	^				
Count Binary	This example runs with or without the MicroC/OS-II R	TOS				
Float2 Functionality	and requires an STDOUT device in your system's hard	ware.				
Float2 Performance	For details, click Finish to create the project and refer	to the				
Hello Freestanding	readme.txt file in the project directory.					
Hello MicroC/OS-II Hello World	The BSP for this template is based on the Altera HAL					
Hello World Small	operating system.					
Memory Test						
Kiemory Test Small	For information about how this software example rela	tes to				
()	< Back Next > Finish	Ci	ancel			

4.4.1.6 Click Finish.

4.4.1.7 Eclipse will create two directories in the workspace, one for the application project and one for the BSP. The application directory (internal_flash_lab) contains a hello_world.c file while the BSP directory (internal_flash_lab_bsp) contains software drivers, a system.h, header file and other software infrastructure.



- 4.4.1.8 The C source file have been provided for you in this lab. Right click on hello_world.c and **delete** it. In the Delete Resources window click **OK**.
- 4.4.1.9 From Windows Explorer, navigate to your main project directory. There you will find a file named **main.c** that you will need to copy to this project.
- 4.4.1.10 Select main.c file and drag it into the internal_flash_lab directory in Eclipse. Select Copy files option in the pop-up and click **OK**.

File Operation	×
Select how files should be imported into the project:	
Opy files	
◯ Link to files	
✓ Create link locations relative to: PROJECT_LOC ∨	
Configure Drag and Drop Settings	
OK Cancel	

You should see the new file appear under the internal_flash_lab project in the Project Explorer.

4.4.2 Build the software

- 4.4.2.1 Right click on the internal_flash_lab_bsp project and select **Properties** from the pop-up menu.
- 4.4.2.2 In the Properties window select the **Nios II BSP Properties** tab. It may take a moment to load the settings.

MAX1000 Internal Flash Lab

4.4.2.3 To keep the software footprint small so it fits our device, open the drop-down menu for Optimization level and change it to Level 2. Enable Reduced device drivers and Small C library options. As there is no C++ code, uncheck the Support C++ option.

Properties for internal_flash_	lab_bsp	— 🗆 X
type filter text	Nios II BSP Properties	<-> < <
 > Resource Builders > C/C++ Build > C/C++ General Linux Tools Path Nios II BSP Properties Project References Run/Debug Settings > Task Repository WikiText 	SopcInfo: \\nios_subsystem.sopcinfo Flags Defined symbols: Defined symbols: none Undefined symbols: none Assembler flags: -Wa,-gdwarf2 Warning flags: -Wall User flags: none Debug level: On Optimization level: Level 2 P Reduced device drivers Support C++ GPROF support Small C library ModelSim only, no hardware support	BSP Editor
?		OK Cancel

4.4.2.4 Click **Apply**, and when it finished with Applying BSP Settings click **OK** to close Properties window.



4.4.2.5 Right click on the internal_flash_lab_bsp project and select **Build Project** from the pop-up menu.

🖨 Build Project			_		×
Building project					
Always run in background					
	Run in Background	Cancel	D	etails >>	

4.4.2.6 When it finished, repeat the previous step for the internal_flash_lab application project.

🖹 Problems 🧔 Tasks 📮 Console 🕱 🔲 Properties
CDT Build Console [internal_flash_lab]
Info: Linking internal_flash_lab.elf
<pre>nios2-elf-g++ -T'/internal_flash_lab_bsp//linker.x' -msys-crt0='</pre>
nios2-elf-insert internal_flash_lab.elfthread_model halcpu_na
<pre>Info: (internal_flash_lab.elf) 4360 Bytes program size (code + init</pre>
Info: 3108 Bytes free for stack + heap.
Info: Creating internal_flash_lab.objdump
nios2-elf-objdumpdisassemblesymsall-headersource intern
[internal_flash_lab build complete]
16:25:32 Build Finished (took 4s.620ms)
<

4.4.3 Run the application

4.4.3.1 Select internal_flash_lab and go to **Run** → **Run Configurations...** and double click to **Nios II Hardware** to add a new configuration.

Run Configurations		×
Create, manage, and run configurations A project name must be selected.		
Image: The second s	Name: New_configuration Project L Target Connection 券 Debugger C Common S Source Project LF file name: C Enable browse for file system ELF file	×
im Nios II Hardware v2 (beta) I Nios II ModelSim I Nios II ModelSim v2 (beta)	File system ELF file name: Advance	
Filter matched 8 of 8 items	Revert App	bly
?	Run Cl	ose

4.4.3.2 Rename it to **Internal Flash configuration** and on the Project tab select **internal_flash_lab** from the drop-down menu for the Project name.

Run Configurations		×
Create, manage, and run co Nios II Hardware Tab Group	nfigurations	
Comparison of the second	Name: Internal Flash co	Infiguration t Connection * Debugger * Source Common internal_flash_lab C:\MAX1000\Unternal_Flash_lab\voftware\\nternal_flash_lab\vhoternal_flash_lab.elf ie system ELF file Advanced Revert Apply
?		Run Close

4.4.3.3 Click on the **Target Connection** tab and click **Refresh Connections** button. The configured MAX1000 board should appear.

Run Configurations							×
Create, manage, and run configurati The expected Stdout device name does not n	ions natch the selected target byte	stream device name.					
Image: Second system Image: Second system <th>Name: Internal Flash com Project Image Co Connections Processors: Cable Image Co Strow-USB-Blaster Image Co Cable Image Co Strow-USB-Blaster Image Co Disable Nios II Console Quartus Project File name: System ID checks Ignore mismatched sys Download IP oselet Start processor Reset the selected targe</th> <th>figuration innection interview cusing default .sopcinfo & .jd tem ID tem timestamp ed target system yet system</th> <th>Ege Source Device ID 1 Device ID 1 files extracted fr</th> <th>Common Instance ID 0 Instance ID 0</th> <th>Name nios Name Ttag War</th> <th>Architecture Nios2:3 Version . 1</th> <th>Refresh Connections Resolve Names System ID Properties</th>	Name: Internal Flash com Project Image Co Connections Processors: Cable Image Co Strow-USB-Blaster Image Co Cable Image Co Strow-USB-Blaster Image Co Disable Nios II Console Quartus Project File name: System ID checks Ignore mismatched sys Download IP oselet Start processor Reset the selected targe	figuration innection interview cusing default .sopcinfo & .jd tem ID tem timestamp ed target system yet system	Ege Source Device ID 1 Device ID 1 files extracted fr	Common Instance ID 0 Instance ID 0	Name nios Name Ttag War	Architecture Nios2:3 Version . 1	Refresh Connections Resolve Names System ID Properties
Filter matched 8 of 8 items							Revert Apply
?							Run Close

4.4.3.4 Click **Apply** and **Run**.

MAX1000
Internal Flash Lab
WDVIA

4.4.3.5 After a few second, the Nios II Console should open at the bottom of the Eclipse and the LEDs should run on the MAX1000 board.



4.5 Program and boot On-Chip Flash

Overview: In this module you will modify the existing design to ensure that the Nios processor is able to boot from the internal Flash.

4.5.1 Hardware modification

4.5.1.1 In Quartus Prime, double click on **nios_subsystem** top-level entity in Project Navigator. The Platform Designer will open.



4.5.1.2 In the search bar of the IP Catalog, type "onchip," and add On-Chip Flash Intel FPGA IP.

IP Catalog 🛛	- d 🗆
🔍 onchip	×
Project	
Library	
Basic Functions	
On Chip Memory On-Chip Flash Intel FPGA IP On-Chip Memory (RAM or ROM) Intel FPGA IP On-Chip Memory (RAM or ROM) Intel FPGA IP	
New Edit	🛉 Add

4.5.1.3 Change the Configuration Mode to **Single Uncompressed Image with Memory Initialization**, under Flash Memory change the Access Mode to **Read and write** for each Sector and set the Clock frequency to **50 MHz**.

ock Diagraffi	T Davamators				
1	Data interface:	Decellel			
now signais	Read burst mode	Parallel V			
anabia dash O		Incrementing] ~		
onchip_flash_U	Read burst count:	8 🗸			
-lest-	Configuration Mode				
Clock	Configuration Scheme:	Internal Con	figuration 🧹		
reset	Configuration Mode:	Single Uncon	pressed Image with Memory Ini	tialization 🗸	
avalon					
avalon	* Flash Memory				
altera_onchip_flash	Sector ID	Access Mode	Address Mapping	Туре	
	1	Read and write	0x00000 - 0x03fff	UFM	
	2	Read and write	0x04000 - 0x07fff	UFM	
	3	Read and write	0x08000 - 0x1c7ff	CFM	
	4	Read and write	0x1c800 - 0x2afff	CFM	
	+ -				
	* Clock Source				
	Clock frequency:	50.0	MHz		
	User is required to pu The on-chip flash me	rovide the clock freque gafunction will be run	ency. with 50000000.0 Hz clo	ck frequency.	
	Enable non-default initi	alization file			
	User created bey or r	mif file:	0.1.1		
		altera_onchi	o_masn.nex		
	User created dat file	for simulation: altera_onchi	o_flash.dat		

- 4.5.1.4 Accept the defaults for the remaining fields and click **Finish** to add the component to the system.
- 4.5.1.5 Rename the new onchip_flash_0 component to **onchip_flash**.
- 4.5.1.6 In the Connections column, hover over the connections and you will then be able to fill in dots to make the connections by clicking them.
- 4.5.1.7 Make the following connections:

Component A		Component B
onchip_flash clk	\leftrightarrow	pll c0
onchip_flash data	\leftrightarrow	nios data_master
onchip_flash data	\leftrightarrow	nios instruction_master
onchip_flash csr	\leftrightarrow	nios data_master

wow

- 4.5.1.8 Automatically create global reset by selecting System → Create Global Reset Network from the menu.
- 4.5.1.9 Automatically assign base addresses for the peripherals by selecting System -> Assign Base Addresses from the menu.
- 4.5.1.10 Verify that your system is the same as below:



- 4.5.1.11 Double click on the Nios II component nios. The Nios II Processor parameter editor will open.
- 4.5.1.12 Click on Vectors tab and set Reset Vector to onchip_flash.data.

eset Vector set vector memory: onchip_flash.data set vector offset: 0x0000000 oxception Vector seption vector memory: onchip.s1 veption vector offset: 0x00000020 set TLB Miss Exception Vector ti TLB Miss Exception vector offset: 0x0000000 ti TLB Miss Exception vector offset: 0x00000000 ti TLB Miss Exception vector offset: 0x00000000 ti TLB Miss Exception vector offset: 0x00000000 ti TLB Miss Exception vector : 0x00000000 ti TLB Miss Exception vector : 0x00000000 ti TLB Miss Exception vector : 0x000000000 ti TLB Miss Exception vector : 0x00000000 ti TLB Miss Exception vector : 0x000000000 ti TLB Miss Exception vector : 0x00000000000 ti TLB Miss Exception vector : 0x000000000 ti TLB Miss Exception vector : 0x00000000 ti tLB Miss Exception vector : 0x000000000 ti tLB Miss Exception vector : 0x000000000000000000 ti tLB Miss Exception vector : 0x0000000000000000000000000000000000	~				
ext vector offset: 0x0000000 set vector: 0x0000000 set vector: 0x0000000 set vector: 0x0000000 seption vector memory: 0x00000020 seption vector offset: 0x0000020 set TLB Miss Exception Vector st TLB Miss Exception vector memory: None ♥ st TLB Miss Exception vector offset: 0x00000000 st TLB Miss Exception vector offset: 0x00000000	~				
bx0000000 set vector: 0x0000000 exeption Vector reption vector memory: onchip.s1 ✓ reption vector offset: 0x0000020 reption vector: 0x00102020 set TLB Miss Exception Vector st st TLB Miss Exception vector offset: 0x00000000					
sect vector: 0x00080000 sception Vector section vector memory: seption vector offset: 0x0000020 seption vector: 0x00102020 set TLB Miss Exception Vector st st TLB Miss Exception vector memory: None st TLB Miss Exception vector offset: 0x00000000 st TLB Miss Exception vector offset: 0x00000000 st TLB Miss Exception vector offset: 0x00000000					
sception Vector seption vector memory: onchip.s1 v seption vector offset: 0x0000020 seption vector: 0x00102020 sest TLB Miss Exception Vector st st TLB Miss Exception vector memory: None st TLB Miss Exception vector offset: 0x00000000 st TLB Miss Exception vector offset: 0x00000000 st TLB Miss Exception vector: 0x00000000					
seption vector memory: onchip.s1 seption vector offset: 0x0000020 seption vector: 0x00102020 set TLB Miss Exception Vector st TLB Miss Exception vector memory: None vector offset: 0x00000000 st TLB Miss Exception vector offset: 0x00000000 vector 0x00000000 vector 0x00000000					
seption vector offset: 0x0000020 seption vector: 0x00102020 set TLB Miss Exception Vector st st TLB Miss Exception vector memory: None vector 0x00000000 st TLB Miss Exception vector offset: 0x00000000 vector 0x00000000	~				
seption vector: 0x00102020 set TLB Miss Exception Vector					
ast TLB Miss Exception Vector tTLB Miss Exception vector memory: tTLB Miss Exception vector offset: 0x00000000 st TLB Miss Exception vector: 0x00000000					
st TLB Miss Exception vector memory: None					
st TLB Miss Exception vector offset: 0x00000000 st TLB Miss Exception vector: 0x0000000	\sim				
st TLB Miss Exception vector:					
		>	~	v V	v V

MAX1000 Internal Flash Lab wow

wow

4.5.1.13 Review message window for remains errors.

At this point should be no remaining errors in the message window. If there are, please refer again to the previous steps to resolve them.

4.5.1.14 Save your design.

4.5.2 Generate the Platform Designer System

- 4.5.2.1 Select **Generate → Generate HDL...** from the menu or alternately click **Generate HDL...** button on the bottom right of the Platform Designer window.
- 4.5.2.2 On the Generation window, enter the following information.
 - Create HDL design files for synthesis: VHDL
 - Uncheck Create timing and resource estimates for third-party EDA synthesis tools.
 - Uncheck Create block symbol file (.bsf)
 - Create simulation model: None

- Generation		×
▼ Synthesis		
Synthesis files are used to compile the system in a Quartus proje	ect.	
Create HDL design files for synthesis: VHDL \sim		
Create timing and resource estimates for third-party EDA sy	inthesis tools.	
Create block symbol file (.bsf)		
 Simulation 		
The simulation model contains generated HDL files for the simula	tor, and may include simulation-only features.	
Simulation scripts for this component will be generated in a vend	or-specific sub-directory in the specified output directory.	
Follow the guidance in the generated simulation scripts about ho ip-make-simscript command-line utilities to compile all of the files	w to structure your design's simulation scripts and how to use the <i>ip-setup-simulation</i> and needed for simulating all of the IP in your design.	
Create simulation model: None 🗸		
Output Directory		
Path: C:/MAX1000/Internal_Fla	ash_lab/nios_subsystem	
L	Generate	incel

4.5.2.3 Click Generate.

4.5.2.4 When the generate process completed, click Close.

Info: rsp mux: "mm interconnect 0" insta	antiated altera merlin multiplexer "rsp m
Info: Reusing file C:/MAX1000/Internal Flat	sh lab/nios subsystem/synthesis/subm
Info: rsp mux 001: "mm interconnect 0)" instantiated altera merlin multiplexer "r
Into: Reusing file C:/MAX1000/Internal_Flag	SILIAD/INOS SUDSVSLEIN/SVILLIESIS/SUDIT
Info: Reusing file C:/MAX1000/Internal_Flat Info: crosser: "mm_interconnect_0" instan	ntiated altera avalon st handshake clock
Info: Reusing file C:/MAX1000/Internal_Fla: Info: crosser: "mm_interconnect_0" instan Info: avalon st adapter: "mm interconn	ntiated altera_avalon_st_handshake_clock ect 0"instantiated altera_avalon_st_dag
Into: Reusing file C:/MAX1000/Internal_Fla: Info: crosser: "mm_interconnect_0" instan Info: avalon_st_adapter: "mm_interconn. Info: error adapter 0: "avalon st adapt	sti_iau/inos_subsystem/synchesis/subin ntiated altera_avalon_st_handshake_cloci eect_0" instantiated altera_avalon_st_adap ere" instantiated error_adapter "error_adap
Into: Reusing the Cr/MAX1000/Internal_Flax Info: crosser: "mm_interconnect_0"instan Info: avalon_st_adapter: "mm_interconn Info: error_adapter_0: "avalon_st_adapt Ninfo: nios_subsystem: Done "nios_subsyst	SII_lady/Inos_Subsystem/Synthesis/Subin itiated altera_avalon_st_handshake_cloci leect_0" instantiated altera_avalon_st_adap ier" instantiated error_adapter "error_adap tem" with 33 modules. 57 files
 Info: Reusing lie C:/MAX1000/Internal_Flat Info: crosser: "mm_interconnect_0" instan Info: avalon_st_adapter: "mm_interconn Info: error_adapter_0: "avalon_st_adapt Info: nios_subsystem: Done "nios_subsyst Info: overnerate succeeded. 	sn_ady/mos_subsystem/symmess/submit nitated altera_avalon_st_handshake_docl ect_0° instantiated altera_avalon_st_adap ere' instantiated error_adapter "error_adap tem" with 33 modules, 57 files
 Info: Reusing lie C:/MAX1000/Internal_lia. Info: crosser: "mm_interconnect_0" instan Info: avalon_st_adapter: "mm_interconn Info: error_adapter_0: "avalon_st_adapt Info: nios_subsystem: Done "nios_subsyst Info: rays-generate succeeded. Info: rays-generate succeeded. 	sm_ady/mos_subsystem/symmess/subm titated altera_avalon_st_handshake_docl eet_0°instantiated altera_avalon_st_adap eer instantiated error_adapter "error_adap tem" with 33 modules, 57 files
 Info: Reusing lie C:/HAX1000/Internal_Fla: Info: crosser: "mm_interconnect_0" instan Info: avalon_st_adapter: "mm_interconn Info: roro_adapter_0: "avalon_st_adapt Info: nios_subsystem: Done "nios_subsyst Info: rays-generate succeeded. Info: Finished: Create HDL design files for s 	sin_adv/mos_subsystem/synthesis/subm titated altera_avalon_st_handshake_cloci dect_0"instantiated altera_avalon_st_adap ter" instantiated error_adapter "error_adap term" with 33 modules, 57 files
Info: Reusing file C:/HAX1000/Internal_Flat Info: crosser: "mm_interconnect_0" instan Info: avalon_st_adapter: "mm_interconn Info: error_adapter_0: "avalon_st_adapt Info: nios_subsystem: Done "nios_subsyst Info: finished: Create HDL design files for s	sn_ady mos_subsystem/syntness/submit tated altera_avalon_st_handshake_docl ect_0" instantiated altera_avalon_st_adap ter" instantiated error_adapter "error_adap tem" with 33 modules, 57 files synthesis

4.5.2.5 Click **Finish** button on the bottom right of the Platform Designer window.

4.5.3 Recompilation

4.5.3.1 Start Compilation by clicking on ► button on the toolbars, or **Processing** → Start Compilation.

There should be no errors. If there are errors, they should be fixed before re-compilating. The 100% in the lower right corner or a green checkmark next to the Compile Design in the Compilation task window indicates that the compilation was successful.



4.5.4 Software reconfiguration

- 4.5.4.1 In the Eclipse, right click on internal_flash_lab_bsp and choose Nios II \rightarrow Generate BSP. With this step you update the BSP files for the modified Nios system.
- 4.5.4.2 Right click on internal_flash_lab_bsp and select **Nios II** → **BSP Editor**.
- 4.5.4.3 On the left side of BSP Editor select Settings/Advanced/hal/linker and set all options.

🗄 BSP Editor - settings.bsp		-		×
File Edit Tools Help				
Main Software Packages Drivers Linker Script Enable File G	eneration Target BSP Directory			
SOPC Information file:\. /pios_subsystem.sopcinfo CPU name: nios Operating system: Altera HAL BSP target directory: C:\WAX1000\Internal_Flash_lab\softw	Version: [default ~] are\nternal flash_lab_bop			
Settings Settings Common Default Default	hallinker allow_code_at_reset enable_alt_load neable_alt_load.copy_rodata enable_alt_load_copy_rwdata enable_alt_load_copy_exceptions			
S >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	: to "andip". it to "andip". ian.			^
-				
	Ge	nerate	Exi	t -

4.5.4.4 Click on the Linker Script tab and set onchip_flash_data from the drop-down menu for the Linker Region Name of '.text'.

Main Software Packages Drivers Li	nker Script Enable File Generation Target	t BSP Directory			
Linker Section Mappings					
Linker Section Name	Linker Region Name	Men	ory Device Name		Add
bas	onchin	ong	nin		Remove
entry	reset	onc	hip flash data		Restore Defaults
exceptions	onchip	onc	nip		
.heap	onchip	onc	nip		
.rodata	onchip	ond	nip		
.rwdata	onchip	onc	nip		
.stack	onchip	onc	nip		
.text	onchip flash data	🗸 onc	hip flash data		
Linker Region Name	Address Range	Memory Device Name	Size (bytes)	Offset (bytes)	Add
Linker Region Name	Address Range	Memory Device Name	Size (bytes)	Offset (bytes)	Add
onchip	0x00102020 - 0x00103FFF	onchip	8160	32	Remove
onchip_BEFORE_EXCEPTION	0x00102000 - 0x0010201F	onchip	32	0	Restore Defaults
onchip_flash_data	0x00080020 - 0x000CDFFF	onchip_flash_data	319456	32	
reset	0x00080000 - 0x0008001F	onchip flash data	32	0	Add Memory Device
					Remove Memory Device
					Memory Ligges
					Memory Usage
					Memory Map
			00		
And the state of the second state of the secon	y created at generate time. They are no	t editable of persisted in the t	SP settings file.		
Grayed out entries are automatical					
Arayed out entries are automatical					
Arayed out entries are automatical nformation Problems Processing mapped module: onchip_nastrice of	se trie detadit univer version.				
Stayed out entries are automatical nformation Problems Processing growpped module: "oncop_nastrono Mapped module: "leds" to use the d	se the default onliver version. :fault driver version.				^
rayed out entries are automatical nformation Problems Processing mapped module: "leds" to use the d Mapped module: "sysid" to use the d	se the default driver version. Fault driver version. Jefault driver version.				î
Srayed out entries are automatical nformation Problems Processing mapped module: "leds" to use the d Mapped module: "sysid" to use the d Mapped module: "sysid" to use the d Mapped module: "sysid" to use the d	se the denatic univer version. :fault driver version. lefault driver version. he default driver version.				Ŷ
rrayed out entries are automatical nformation Problems Processing Mapped module: "oricinp_nasm to co Mapped module: "sysid" to use the d Mapped module: "sysid" to use the d Mapped module: "itag_uart" to use Finished loading drivers from ensem	se the denatic univer version. :fault driver version. lefault driver version. the default driver version. xle report.				^
srayed out entries are automatical nformation Problems Processing ymapped module: "on the processing Mapped module: "add" to use the d Mapped module: "sysid" to use the Mapped module: "tag_uart" to use Finished loading driving, uart" to use Finished loading driving settings from setting Loading 59° settings from settings	se the denadic driver version. fault driver version. lefault driver version. the default driver version. kle report. le.				Â
Srayed out entries are automatical information Problems Processing Imapper mouser rennip_inset to Mapped module: "sigal" to use the d Mapped module: "sigal" to use the Mapped module: "sigal" to use the Enished loading drivers from estings I Loading SSP settings from settings Distributed sources Source States and the sources Finished loading SSPC Builder soute	se ure ueraur unver version. ifault driver version. lefault driver version. the default driver version. ke report. le. n info file "	slative to settings file]"			Â
Srayed out entries are automatical information Problems Processing y mapper mouncer roring-main or o Mapped module: "load" to use the i Mapped module: "tytag_uart" to use p Finished loading drivers from ensem j Loading SPS betings from settings j Finished loading SOPC Builder syste Changed maged section 'text' fro	se the denaut on ver version. If ault driver version. Iefault driver version. the default driver version. Jac report. Ie. n info file ", 'pios_suboystem.sopcinfo [re m memory region "onchip" to memory region	slative to settings file]" "onchip. filsh. data".			Ĵ
Grayed out entries are automatical Information Problems Processing yr mappen mocauser running maner und Mapped module: "leval" to use the d Mapped module: "sysial" to use the d Mapped module: "sysial" to use the d Mapped module: "leval" to use the d Finathed loading drivers from enterm Loading BSP settings from settings f Di Finathed loading SOC Builder systal Changed mapped section ".text" fro	se the behauit unver version. efault driver version. lefault driver version. he default driver version. he report. le. info file " 'nica _subaystem.sopcinfo [re n memory region" onchip" to memory region	slative to settings file]" "onchip_flash_data".			~

MAX1000 Internal Flash Lab WDW

4.5.4.5 Click Generate and when finished click Exit.

4.5.4.6 Right click on internal_flash_lab and choose **Make Targets** → **Build...**

Target mem init install	Location	Add
mem_init_generate		Remov
elp		Edit

4.5.4.7 In the Make Targets window select mem_init_generate and click Build.

4.5.5 Configuration image

4.5.5.1 In Quartus Prime, go to File → Convert Programming File.

4.5.5.2 In the Convert Programming File window set Mode to Internal Configuration.

Open 0	Conversion Setup	Data	Save (Conversion	Setup	
Output programming f	file					
Programming file type	Programmer O	bject File (.pof)				-
Options/Boot info	Configuration d	levice: EPCE16	Mode:	Internal	Configuration	n 🔻
File name:	output_files/ou	utput_file.pof				
Advanced	Remote/Local u	pdate difference file: NONE				~
	_					
	Create Mem	ory Map File (Generate outpu	ut_file.map)			
	Create Mem	ory Map File (Generate outpu iles (Generate output_file.pe	ut_file.map) riph.pof and output_	file.core.rbf)	
	Create Memo	ory Map File (Generate outpu iiles (Generate output_file.pe g data RPD (Generate output	ut_file.map) riph.pof and output_ :_file_auto.rpd)	file.core.rbf)	
nput files to convert	Create Memo	ory Map File (Generate outpu illes (Generate output_file.pe g data RPD (Generate output	ut_file.map) riph.pof and output_ :_file_auto.rpd)	file.core.rbf)	
nput files to convert File/Data	Create CvP f Create config Create config area	ory Map File (Generate outpu illes (Generate output_file.pe g data RPD (Generate output Properties	ut_file.map) riph.pof and output_ _file_auto.rpd) Start Address	file.core.rbf		Add Hex Data
nput files to convert File/Data SOF Data	Create CvP f Create confi	ory Map File (Generate output files (Generate output_file.pe g data RPD (Generate output Properties Page_0	it_file.map) riph.pof and output	file.core.rbf		Add Hex Data
nput files to convert File/Data SOF Data	Create CvP f Create CvP f Create config area	ory Map File (Generate outpu files (Generate output_file.pe g data RPD (Generate output Properties Page_0	it_file.map) riph.pof and output_ file_auto.rpd) 	file.core.rbf		Add Hex Data Add Sof Page Add File
nput files to convert File/Data SOF Data	Create Mem Create CvP f Create confi	ory Map File (Generate outpu files (Generate output_file.pe g data RPD (Generate output Properties Page_0	it_file.map) riph.pof and output_ _file_auto.rpd) 	file.core.rbf		Add Hex Data Add Sof Page Add File Remove
nput files to convert File/Data SOF Data	Create Memu	ory Map File (Generate outpu files (Generate output_file.pe g data RPD (Generate output Properties Page_0	ıt_file.map) riph.pof and output_ _file_auto.rpd) 	file.core.rbf		Add Hex Data Add Sof Page Add File Remove Up
nput files to convert File/Data SOF Data	Create Memu	ory Map File (Generate output files (Generate output_file.pe ig data RPD (Generate output Properties Page_0	it_file.map) riph.pof and output_ file_auto.rpd) 	file.core.rbf		Add Hex Data Add Sof Page Add File Remove Up Down

MAX1000 Internal Flash Lab WDW

4.5.5.3 Click on Options/Boot info...

- 4.5.5.4 In the MAX10 Device Options window set **Load memory file** from the drop-down menu for UFM source.

<project_directory>/software/internal_flash_lab/mem_init and open onchip_flash.hex

🛍 Max 10 Device Options	×				
Power On Reset scheme: Instant ON	~				
Set I/O to weak pull-up prior usermode					
Configure device from CFM0 only					
Use secondary image ISP data as default setting when availabl	e				
Security					
Verify protect					
Allow encrypted POF only					
Dual Config					
Enable watchdog					
Watch value:					
User Elash Memory					
,					
UFM source: Load memory file					
File path: ernal_flash_lab/mem_init/onchip_flash.hex					
RPD File Endianness					
🗹 Little endian					
Big endian					
Description:					
New memory file path used as UFM data					
Ok Cancel	Ī				

4.5.5.6 Click **OK**.

4.5.5.7 Under the Input files to convert Click on SOF DATA and then click on Add File...

WUW

4.5.5.8 Go to: <project_directory>/output_files/ and open nios_subsystem.sof.

🛍 Convert Programming	g File - C:/MAX1000/In	ternal_Flash_lab/n	ios_subsyst	em - nios_	subsyster	n	_		Х
File Tools Window							Search al	tera.com	6
Specify the input files to You can also import inpu future use. Conversion setup files Open C	convert and the type o tt file information from onversion Setup Data.	f programming fil other files and sa 	e to generat	te. ersion sett Sa	up inform	nation	created he	ere for	
Output programming fi	ile								
Programming file type:	Programmer Object	File (.pof)							•
Options/Boot info	Configuration device:	EPCE16	v	Mode:	Int	ernal	Configurat	tion	•
File name:	output_files/output_	file.pof							
Advanced	Remote/Local update	difference file: N	IONE						~
Input files to convert	Create CvP files (G	enerate output_fi	le.periph.po itput_file_a	f and outp uto.rpd)	out_file.co	ore.rbf)		
File/Data	area	Properties		Start Addr	ress			Add He	x Data
✓ SOF Data nios_subsystem	Page m.sof 10M	2_0 085AU169	<	auto>				Add So Add F Rem Uj	f Page Tile ove
				[Generate	e	Close	Prope	rties Help

4.5.5.9 Click Generate.

4.5.5.10 Click **OK** in the pop-up window and **close** the Convert Programming File window.

4.5.5.11 Open the Programmer window.

4.5.5.12 Click on Change File... button and open output_file.pof.

4.5.5.13 Check the Program/Configure checkbox for the .pof file, CFM0 and UFM.

Programmer - C:/ File Edit View	ner - C:/MAX1000/Internal_Flash_lab/nios_subsystem - nios_subsystem - [nios_subsystem.cdf]* View Processing Tools Window Help								s	– 🗆 X Search altera.com		
Hardware Setup.	Arrow-USB-Blaster [USB0] ISP to allow background progr	amming when avail	able		Mode: J	ITAG		•	Progress:		100% (Succ	essful)
▶ [™] Start	File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMP	IPS File
Contract Stop Contr	output_files/output_file.pof CFM0 UFM	10M085AU169	012DF50D	001F964C	N N							
P** Add File P** Change File Change File P** Add Device P** Up P** Down												>

4.5.5.14 Click **Start** to program the board. When the configuration is complete, the Progress bar should show 100% (Successful).

Progress: 100% (Successful)

After this point, the FPGA configuration file and the program code for Nios soft processor are stored in the internal Flash memory of MAX10. When you power-on this board, this configuration file will load automatically.

4.5.6 Testing the project

- 4.5.6.1 **Close** Programmer, Quartus Prime and Nios II Eclipse.
- 4.5.6.2 Remove MAX1000 board from your PC in order to power-off the device. In this way we can check the non-volatile configuration.

WUW

4.5.6.3 Open Nios II Command Shell from the **Start menu** → **All Programs** → **Intel FPGA 18.0.0.614 Lite Edition**. This command line tool is useful for a range of activities, from board and system-level debugging to programming an FPGA configuration file.



4.5.6.4 Connect again your MAX1000 board to your PC.

4.5.6.5 In the terminal window type the following command and press enter:

nios2-terminal

This command establishes contact with stdin, stdout, and stderr in the Nios II processor subsystem and allows you to provide input and monitor them. These standard streams are routed through the JTAG UART module within this system.

4.5.6.6 In the terminal window you should see 'Hello from Nios II!' that we saw in the Eclipse Nios terminal. Every time, when you press the Reset button this sentence is printed.



CONGRATULATIONS! YOU HAVE SUCCESSFULLY COMPLETED THE INTERNAL FLASH LAB!

5 Revision History

Version	Change Log	Date of Change
V1.0	Initial Version	18/01/2019

6 Legal Disclaimer

ARROW ELECTRONICS

EVALUATION BOARD LICENSE AGREEMENT

By using this evaluation board or kit (together with all related software, firmware, components, and documentation provided by Arrow, "Evaluation Board"), You ("You") are agreeing to be bound by the terms and conditions of this Evaluation Board License Agreement ("Agreement"). Do not use the Evaluation Board until You have read and agreed to this Agreement. Your use of the Evaluation Board constitutes Your acceptance of this Agreement.

PURPOSE

The purpose of this evaluation board is solely intended for evaluation purposes. Any use of the Board beyond these purposes is on your own risk. Furthermore, according the applicable law, the offering Arrow entity explicitly does not warrant, guarantee or provide any remedies to you with regard to the board.

LICENSE

Arrow grants You a non-exclusive, limited right to use the enclosed Evaluation Board offering limited features only for Your evaluation and testing purposes in a research and development setting. Usage in a live environment is prohibited. The Evaluation Board shall not be, in any case, directly or indirectly assembled as a part in any production of Yours as it is solely developed to serve evaluation purposes and has no direct function and is not a finished product.

EVALUATION BOARD STATUS

The Evaluation Board offers limited features allowing You only to evaluate and test purposes. The Evaluation Board is not intended for consumer or household use. You are not authorized to use the Evaluation Board in any production system, and it may not be offered for sale or lease, or sold, leased or otherwise distributed for commercial purposes.

OWNERSHIP AND COPYRIGHT

Title to the Evaluation Board remains with Arrow and/or its licensors. This Agreement does not involve any transfer of intellectual property rights ("IPR) for evaluation board. You may not remove any copyright or other proprietary rights notices without prior written authorization from Arrow or it licensors.

RESTRICTIONS AND WARNINGS

Before You handle or use the Evaluation Board, You shall comply with all such warnings and other instructions and employ reasonable safety precautions in using the Evaluation Board. Failure to do so may result in death, personal injury, or property damage.

You shall not use the Evaluation Board in any safety critical or functional safety testing, including but not limited to testing of life supporting, military or nuclear applications. Arrow expressly disclaims any responsibility for such usage which shall be made at Your sole risk.

WARRANTY

Arrow warrants that it has the right to provide the evaluation board to you. This warranty is provided by Arrow in lieu of all other warranties, written or oral, statutory, express or implied, including any warranty as to merchantability, non-infringement, fitness for any particular purpose, or uninterrupted or error-free operation, all of which are expressly disclaimed. The evaluation board is provided "as is" without any other rights or warranties, directly or indirectly.

You warrant to Arrow that the evaluation board is used only by electronics experts who understand the dangers of handling and using such items, you assume all responsibility and liability for any improper or unsafe handling or use of the evaluation board by you, your employees, affiliates, contractors, and designees.

wow

LIMITATION OF LIABILITIES

In no event shall Arrow be liable to you, whether in contract, tort (including negligence), strict liability, or any other legal theory, for any direct, indirect, special, consequential, incidental, punitive, or exemplary damages with respect to any matters relating to this agreement. In no event shall arrow's liability arising out of this agreement in the aggregate exceed the amount paid by you under this agreement for the purchase of the evaluation board.

IDENTIFICATION

You shall, at Your expense, defend Arrow and its Affiliates and Licensors against a claim or action brought by a third party for infringement or misappropriation of any patent, copyright, trade secret or other intellectual property right of a third party to the extent resulting from (1) Your combination of the Evaluation Board with any other component, system, software, or firmware, (2) Your modification of the Evaluation Board, or (3) Your use of the Evaluation Board in a manner not permitted under this Agreement. You shall indemnify Arrow and its Affiliates and Licensors against and pay any resulting costs and damages finally awarded against Arrow and its Affiliates and Licensors or agreed to in any settlement, provided that You have sole control of the defense and settlement of the claim or action, and Arrow cooperates in the defense and furnishes all related evidence under its control at Your expense. Arrow will be entitled to participate in the defense of such claim or action and to employ counsel at its own expense.

RECYCLING

The Evaluation Board is not to be disposed as an urban waste. At the end of its life cycle, differentiated waste collection must be followed, as stated in the directive 2002/96/EC. In all the countries belonging to the European Union (EU Dir. 2002/96/EC) and those following differentiated recycling, the Evaluation Board is subject to differentiated recycling at the end of its life cycle, therefore: It is forbidden to dispose the Evaluation Board as an undifferentiated waste or with other domestic wastes. Consult the local authorities for more information on the proper disposal channels. An incorrect Evaluation Board disposal may cause damage to the environment and is punishable by the law.