Instructions on FPGA Board and Xilinx software

Prepared by Diego Campos (SPRING 2002): 1.0,1.5,2.0 Updated by Matthew Klepeis (SPRING 2003): 3.0 Snehesh Shrestha (SPRING 2004): 4.0 Steve Sanko (SPRING 2005): 4.1 Brian Moran (SPRING 2006): 4.2 Chris Janssen and James Kelly (SPRING 2013) 5.0 Nathan Copros (SPRING 2018) 6.0 Version 6.0 Last update -3/13/13 Edited by Dr. Joseph Wunderlich

I. XILINX AND FPGA INTRODUCTION

Field-Programmable Gate Array (FPGA) - A logic chip which can be programmed to function as an array of computational logic blocks.

Xilinx Software - Xilinx Software allows a designer to graphically create a logic circuit which can be tested and simulated prior to implementation on the FPGA board.

FPGA vs Breadboarding [1]

- FPGA's simplify design, implementation, testing, and debugging
- Reusable Easily reprogrammed and reconfigured
- Most are non volatile (the memory is saved after the device is turned off)
- Useful in prototyping IC designs
- Low power
- High number of gates/parts in the same area
- HDL helps create libraries (new gates or parts) and functional blocks
- Do not need to worry about voltage, current, or power to each gate or circuit part
- Circuit can be simulated and debugged before implementation; therefore saving time and money

FPGAs can be programmed using schematics to lay out a design, or by using an HDL like Verilog or VHDL. As well as being cheap to program for an application, FPGA hardware is relatively inexpensive. FPGAs can be used for complex applications, such as image processing or image compression, and are suitable for use in portable phones, digital cameras, and other complex digital applications. Xilinx, as well as other companies sell intellectual property (IP) cores, and opencores.org provides cores and source code for free under the GPL.

Evaluating the performance of an FPGA is difficult. Most manufacturers don't provide a gate count for their products, because it tends to be misleading. Mostly, FPGA manufacturers give a number of "logic units" or "logic cells," which consist of a lookup table and a flip-flop, as well as assorted connection logic.

Because different manufacturers use different types of basic gates (AND vs. NAND, etc), the gate count of a logic unit isn't the same across different manufacturers, or even different FPGAs.

FPGAs are becoming more sophisticated, and their applications are expanding as they become more capable. When doing calculations that can be made in parallel, the FPGAs are programmed to deal specifically with this problem. Because of the high speed I/O capabilities of the FPGA, it is ideal for acceleration of certain problems.

II. INSTALLING SOFTWARE

Installation Notes:

- 1. With Windows 10, you will need to run the 32-bit version of the ISE <u>Webpack</u> software:
- 2. You will need to create an account to download the software. Then, fill the verification form, where company name is Elizabethtown College.
- 3. Only one license can be activated per account, so use a different account for each computer.
- 4. To activate the license, launch the "Manage Xilinx Licenses" app under the folder Xilinx Design Tools. This can be found by clicking "All Apps" after pressing the Windows button.
- 5. Select the "Get Free Vivado/ISE Webpack License" option to aquire a license. Then fill in the information to download a license folder. Then, go to the "Manage Licenses" tab to load the license file and you can now launch the fully functional free software.

🗶 Xilinx License Configuration Manager	_		\times
Acquire a License Manage Licenses Borrow/Restore Licenses Return Licenses Internet Settings			
Select one of the following options			_
C Start Now! - 30 Day Trial (No Bitstream)			
Get Free Vivado/ISE WebPack License			
C Start 30 Day Evaluation			
C Get My Purchased License(s)			
O Locate Existing License(s)			
Description of the above selected option			
Get a free Vivado/ISE WebPack. license and start using your Xilinx software. You will be taken to the Xilinx website where you can for Vivado/ISE WebPack. Once your licenses file is generated, the "Manage Xilinx Licenses" tab will open to enable you to configuu the license. For more information on Vivado/ISE WebPack, including supported devices and applications, please visit www.xilinx. Note: WebTalk is always enabled for WebPACK users. WebTalk ignores user and install preference when a bitstream is generated license. If a design is using a device contained in WebPACK and a WebPACK license is available, the WebPACK license will always this, please see Answer Record 34746.	re your syst com. d using the !	tem to use WebPACK	
About	lext	Close	

III. USING XILINX ISE PROJECT NAVIGATOR

- a) Start program called "32-bit Project Navigator". It can be found by pressing the start menu, "All Apps", "Xilinx Design Tools"
- b) Create a new project

>	New Project Wizard	1		×
←c	reate New Project Specify project loc			
	Enter a name, locati	ons, and comment for the project		
	Name:	Lab3		
	Location:	U:\Private\TA NAthan\Lab3		
	Working Directory:	U:\Private\TA NAthan\Lab3		
	Description:			
	Select the type of to	p-level source for the project		
	Top-level source typ	e:		
	Schematic			\sim
1	More Info		Next >	Cancel

- c) Choose a name, and make sure to save it in a location that is not read-only (your public or private folder will work). Change the Top-level source type to "Schematic". Press "Next"
- d) Select "Spartan" in the "Family" box, "XC6SLX45" in the device box and "CSG324" in the "Package" box

elect the device and design flow for the p	roject	
Property Name	Value	
Evaluation Development Board	None Specified	~
Product Category	All	~
Family	Spartan6	~
Device	XC6SLX45	~
Package	CSG324	~
Speed	-3	~
Top-Level Source Type	Schematic	\sim
Synthesis Tool	XST (VHDL/Verilog)	~
Simulator	ISim (VHDL/Verilog)	~
Preferred Language	Verilog	~
Property Specification in Project File	Store all values	~
Manual Compile Order		
VHDL Source Analysis Standard	VHDL-93	~
Enable Message Filtering		

- e) When the "Project Summary" box appears, select "Finish"
- f) Under the Hierarchy window on the top left, right-click on "xc6slx45-3csg324", and choose "New Source". Press "Finish"

Hierarchy 🔄 😭 xcбslx45-3csg324	^
New Source Wizard Select Source Type Select source type, file name and its lease	×
 IP (CORE Generator & Architecture Schematic User Document Verilog Module Verilog Test Fixture VHDL Module VHDL Library VHDL Package VHDL Test Bench Embedded Processor 	Wizard) File name: SchematicLab3 Location: U:\Private\TA NAthan\Lab3
More Info	Next > Cancel

g) The schematic window will appear and you will be able to navigate between the hierarchy window under the "Design" Tab and the logic circuit component under the "Symbols" Tab

Advanced Advanc	ols ↔□♂×	*	(10 - 2
Addressed Addressed Evifies Computer Compu		- 3					
Anthometic Carry, Logic Comparison Comp	<all symbols=""></all>						
Sumbal Info	Advanced						
Carbonics Comparison Comparison Comparison Decoder Properties Decoder Properties Decoder Deco	Anthmetic		-				
Cocostandes Counteres Counteres Counteres Dob Rip, Fripp Decoder file, Fripp Do Di Rip, Fripp Decoder Symbol Info Symbol Info Symbol Info Symbol Info		No.					
Comparison Comparison DOR Inp. Flop Decoder General	- Carry_Logic						
Counter DOR Plip, Frip Prode 100, Flip Frip 100, Flip Fri	Clocking_Kesources						
Piscate piscate 0 Piscate 0 10 Piscate 0 10 <t< td=""><td>Comparator</td><td>TD-</td><td></td><td></td><td></td><td></td><td></td></t<>	Comparator	TD-					
Piscate piscate 0 Piscate 0 10 Piscate 0 10 <t< th=""><th>Counter</th><th></th><th></th><th></th><th></th><th></th><th></th></t<>	Counter						
Pressed Provide Opting Opting Opting Packad accid accid <td< td=""><td>DUR Hip_Hop</td><td>13</td><td></td><td></td><td></td><td></td><td></td></td<>	DUR Hip_Hop	13					
General OD OD Distributions mode/Cutput/functions mode/Cutput/functions mode/Cutput/functions acd3 accd4 accd4 acd4 add18 add31 add32 add32 add32 add32 add31 add32 add31 add32 add32 add31 add32 add32 add33 add34 add35 add31 add32 add33 add34 add35 add34 add35 add34 add35 add34 add35 add34 add35 add36 add37 add38 add39 add40 add54 add54 add54 add54	Decoder						
O_O_TipErp O_Listch modul acc4 acc8 acd3 add1 add3 add3 add4 add3 add3 add3 add1 add2 and21	Flip_Flop						
Or Fightspe Oblishin ImpdVOutput_Functions	General	0					
I cluich information and i	10	1					
In put/Uduput/Unctions	- IO_FIIpFlop						
wrotch ************************************	- IO_Latch						
acc6 acc8 acd3 add3 add4 add3 add4 add4 add4 add3 add4 add4 add3 add1 add2 and2		A					
a col 6 a col 6 a col 8 a col 6		- Telephone					
<pre>scs4 scs6 sdd16 sdd3 sdd8 sdd8 sdd8 sdd8 sdd8 sdd8 sdd8</pre>	mbols						
<pre>scc4 add16 add16 add3 add3 add3 add3 add3 add3 add3 add</pre>	acc16	5					
acc8 add16 add4 add3		10					
add16 add3	- acc8						
dod4 dod3	add16	19					
add3 addu16 - addu16 - addu16 - addu1 - add	add4						
adsul6 -adsul8 -adsul8 -adsul8 -adsul8 -adsul9 adsul9 adsul9 adsul9 adsul9 	add8	A.					
adsud and12 and12 and		8	5-				
adulia and 12 and 13 and 20 and 20 an							
and12 and24 and24 and2b1 and2b10 and2b10 and2b10 and2b10 and2b20 and3 b1 b1 b1 b1 b1 b1 b1 b1 b1 b1 b1 b1 b1	- adsu8						
and16 and20 and201 and201 and2020 and20 bline and2202 and20 bline and20 and	- and12	12					
and2011 and202 and202 and202 and202 and202 and202 and and and and and and and and and and	and16	(TN)					
and2b1 and2b2 and2b2 obliver Filter ntation tation 5 ymbol Info	end2	1					
and2b1i and2b2 bol Name Fiter ntation symbol Info	- and2b1		1				
and2b2 with the second	and2b1I						
and 3 v bol Name Filter Inter on Symbol Infe	and2b2						
hlatori ste 0 vi	and3 v						
hlatori ste 0 vi	hel Name Filter						
te 0 v	100000000000000000000000000000000000000						
Symbol Info	tation						
Speed Info							
🖒 Files 🐚 Libraries 😟 Symbols 📢 🐻 Schematicules).sch 📴 🔟 Design Summary 🛄			<u>[</u>	4		1 I I I	
		125	Subamphiel and sub-		C	Design Summary	

Started : "Launching Schematic Editor to edit SchematicLab3.sch". Launching Design Summary/Report Viewer...

h) Like Logisim, the Symbol tool lets you add parts by connecting the respective leads together with the mouse or with the wire drawing tool. Common gates like the AND gate are located under the "Logic" category. To create an Input, you must first add an "IBUF" from the IO

category. Then, click "I/O Marker" and click the first leg. For an output, add an Obuf and add the I/O marker to the second leg.

iii. Input(s) & Output(s): Must be assigned PIN LOCALIZATION NUMBERS. (see step l)



- i. Inputs and outputs should be named in such a way to help illustrate their purpose
 - i. To input a name, double click on the symbol, click "Nets" in the category on the left of the window that appears and type the desired name in the value field under the "Name" row

cadegory .	Ten and edit the	attributes of the	selected nets		
 1/0 Mailers SW1 Nets 	Name	Value		Visible	Rev
SAL	Name	SM1		Add	Edt. Turks
	PortPolarity	legent	•	Add	
					Delete

- j. Assigning pin locations
 - i. Create a constraint file by clicking on the "design" at the bottom of the design window (if the tap is not visible click the arrows in the bottom right corner of the window to scroll through the different tabs)
 - ii. Select your schematic in the hierarch window
 - iii. Click the "New Source" button
 - iv. Select the "implementation constraint file" under "source type" and input a name
 - v. To assign a pin to an input/output, type "NET "<block name>"
 LOC="<pin location>";" into the text file that was created when

you created the constraint file

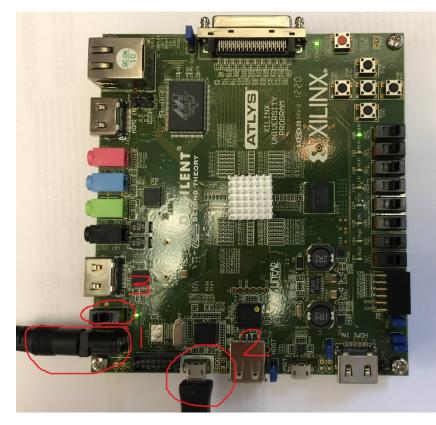
Pushbuttons	Slide Switches	LEDs
BTNU: N4	SW0: A10	LD0: U18
BTNC: F5	SW1: D14	LD1: M14
BTNR: F6	SW2: C14	LD2: N14
BTNL: P4	SW3: P15	LD3: L14
BTND: P3	SW4: P12	LD4: M13
BRST: T15	SW5: R5	LD5: D4
	SW6: T5	LD6: P16
	SW7: E4	LD7: N12

Important Pin Locations (for full list see Appendix A)

IV. IMPLEMENTATION/PROGRAMMING

Make sure to use the power cord from the Xilinx box. It should be a 5V output. Any voltage superior to this could permanently break the FPGA board.

- a) Once the circuit is properly designed, you can proceed to implement your circuit on the FPGA.
- b) Setup the FPGA board in three steps as illustrated in the picture below:



Plug in the power cord, then the usb cable from the board to the computer, and finally turn on the power switch.

- c) Click on the "Implement" button (green arrow on the middle left of the design window)
- d) In the hierarchy, click on the schematic file. On the bottom left, you should see the list of processes. Click on "Synthesize –XST" and "Generate Programming File". After that, you should see only green status like in the picture below:

Processes: manuaschem

- Design Summary/Reports Σ ÷... Design Utilities 🛓 -- 😼 User Constraints 🗄 🔁 🕜 Synthesize - XST 🖶 🍋 🖉 Implement Design 🗄 🚺 🕢 Translate 🗄 🔁 🕗 Map 🗄 🔁 🕢 Place & Route 🖓 🕜 Generate Programming File Configure Target Device 🕀 - 🕀 <u>е</u>ш Analyze Design Using ChipScope
- e) The next step is to download the circuit into the FPGA. To accomplish this, press the Tools tab then click on "Impact", and press "OK" to the warning message.
- f) Double-click "Boudary Scan", then right-click and select "Initialize Chain". A message should pop-up to assign configuration files, press "Yes". Then open the saved schematic. Another message should pop-up about Flash PROMs, you should press "No". Then the Device Programming Properties window will come up and click on Device 1 under Boundary-Scan and press "OK".
- g) Finally, if everything went well, you should see the equivalent of the picture below. To launch the program, press "Program" under Available Operations. It should display "Program Succeeded"

