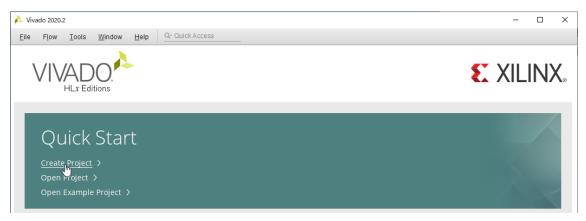
Vivado Design Suite Tutorial

Introduction

This tutorial guides you through the design flow using Xilinx Vivado software to create a simple digital circuit using Verilog HDL. You will go through the typical design flow targeting the Artix-7 based BASYS3 board.

1. Creating a Vivado Project

1.1) Open up Vavado 2020.2 (not Vivado HLS) and click Create Project to start the wizard.



1.2) You will see the "Create a New Vivado Project" dialog box. Click **Next**.

1.3) Enter **adder** in the "Project name" field. Click the browse button of the "Project location" field, browse to your project saving directory, and click **Select**. Make sure that the "Create Project Subdirectory" box is checked. Click **Next**.

🝌 New Project		×
Project Name Enter a name for ye	our project and specify a directory where the project data files will be stored.	A
<u>P</u> roject name:	adder	8
Project location:	D:/Seminar	⊗
🗹 Create proje	ct subdirectory	
Project will be cr	eated at: D:/Seminar/adder	

1.4) Select **RTL Project** option in the "Project Type" form and check the "Do not specify sources at this time" box. Click **Next**.

À New Project	×
Project Type	
Specify the type of project to create.	1
 <u>R</u>TL Project You will be able to add sources, create block designs in IP Integrator, generate IP, run RTL analysis, synthesis, implementation, design planning and analysis. <u>D</u>o not specify sources at this time <u>Project is an extensible Vitis platform</u> 	

1.5) In the "Default Part" form, using the **Parts** option and various drop-down fields of the filter section, select the **xc7a35tcpg236-1** part. Click **Next**.

New Projec	t								
efault Par 100se a def	t ault Xilinx part or board	for your project.							
Parts									
Category:	All		~	Package:	cpg236	~	Temperature	All Remain	ing 🗸
Family:	Artix-7		~	Speed:	eed: -1 🗸		Static power:	All Remaining 🔍	
<u>S</u> earch:	Q	I/O Pin Count	~ Available		Elements	FlipFlops	Block RAMs	Ultra RAMs	505-
Dort								UNIARAMS	
Part xc7a15tc	pg236-1								DSPs 45
Part xc7a15tc xc7a35tc		236 236	106	104	00	20800 41600	25 50	0	45 90

1.6) You will see "New Project Summary" form. Click **Finish** to create the project.

2. Creating a Source File

2.1) Select Add Sources from the File menu.

🝌 ad	🍌 adder - [D:/Seminar/adder/adder.xpr] - Vivado 2020.2						
<u>F</u> ile	<u>E</u> dit F <u>l</u> ow <u>T</u> ools	Rep <u>o</u> rts	<u>Window Layout View Help</u>				
	Project	+					
	Add Sources	Alt+A	PROJECT MANAGER - adder				
	<u>C</u> lose Project		Sources ? _ K Project Summary				
	<u>C</u> onstraints	- F					
	Simulation Waveform	+	Q ★ + 2 ● 0 ♦ Overview Dashboard				
	Chec <u>k</u> point	Þ	☐ Design Sources > ☐ Constraints Settings Edit				
	ĮΡ	+	✓ ☐ Simulation Sources Project name: adder				

2.2) In the "Add Sources" form, select Add or create design sources. Click Next.

🝌 Add Sources		×
HLx Editions	Add Sources This guides you through the process of adding and creating sources for your project Add or greate constraints Add or create design sources Add or create gimulation sources	

2.3) In the "Add or Create Design Sources" form, click **Create File** button.

dd or Create Design Sources pecify HDL, netlist, Block Design, and IP files isk and add it to your project.	s, or directories containing those file types to add to your project. Create a new source file on	1
+, - + +		
U	Jse Add Files, Add Directories or Create File buttons below	
Scan and add RTL include files into proj	Add Files Add Directories Create File	
Copy <u>sources into project</u> Add so <u>u</u> rces from subdirectories		

2.4) In the "Create Source File" form, select **Verilog** in the "File type" field. Enter **half_adder** in the "File name" field. Click **OK**.

À Create Source File X					
Create a new source file and add it to your project.					
<u>F</u> ile type:	Verilog	*			
F <u>i</u> le name:	half_adder	\otimes			
Fil <u>e</u> location:	😜 <local project="" to=""></local>	~			
?	ОК	ncel			

2.5) You will see the created file in the "Add or Create Design Sources" form. Click **Finish**.

A 4	Add Sources							×
Sp	ecify HDL, ne	e Design So tlist, Block Des o your project.	sign, and IP files	s, or directories c	containing those file ty	pes to add to your proje	ct. Create a new source	file on
	+, -	+ +						
		Index	Name	Library	Location			
	•	1	half_adder.v	xil_defaultlib	<local project="" to=""></local>			
					1		1	
				Add Files	A <u>d</u> d Directories	<u>C</u> reate File		
(Scan and	add RTL <u>i</u> nclu	de files into proj	ect				
(Copy <u>s</u> our	rces into proje	ct					
(Add so <u>u</u> rc	es from subdir	rectories					
	Ð				<	<u>B</u> ack <u>N</u> ext ≥	<u>F</u> inish	Cancel

2.6) Skip the "Module Definition" form by clicking **OK**.

2.7) You will see the "Define Module" dialog box. Click Yes.

2.8) The created file will be added to the "Design Sources" folder in the "Sources" pane of the "PROJECT MANAGER".

À adder - [D:/Seminar/adder/adder.xpr] - Vi	vado 2020.2			
<u>F</u> ile <u>E</u> dit F <u>l</u> ow <u>T</u> ools Rep <u>o</u> rts	s <u>W</u> indow La <u>v</u> out <u>V</u> iew <u>H</u> elp	Q- Quick Access		
- 🕒 i 🔺 i 🕒 i 🖬 i 🗙 i 🕨	👬 🔅 ∑ 😿 🔌 🎘			
Flow Navigator 😤 🌲 ? 🔔	PROJECT MANAGER - adder			
✓ PROJECT MANAGER	Sources ? _ 🗆 🖒 X	Project Summary × half_adder.v ×		
Settings	Q ≍ ≑ + ? ● 0 *	D:/Seminar/adder/adder.srcs/sources_1/new/half_ac		
Add Sources Language Templates	V Design Sources (1)	Q, ■ ← → X, ■ ■ X . 1 : `timescale ins / ips 2 □		
₽ IP Catalog	half_adder (half_adder.v) Constraints			
V IP INTEGRATOR	✓ □ Simulation Sources (1) > □ sim_1 (1)	<pre>3 // Company: 4 // Engineer: 5 //</pre>		

3. Verilog Coding

3.1) Double click on the **half_adder.v** file in the "Sources" pane to open it.

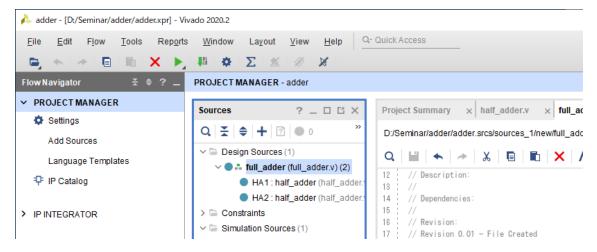
3.2) Write the half adder Verilog code into the text editor.

Project Summary × half_adder.v ×	? 🗆 🖒
D:/Seminar/adder/adder.srcs/sources_1/new/half_adder.v	×
Q ★ → & ■ ■ X // ■ ♀	٥
9 // Project Name: 10 // Target Devices: 11 // Tool Versions: 12 // Description: 13 // 14 // Dependencies: 15 // 16 // Revision: 17 // Revision 0.01 - File Created 18 // Additional Comments: 19 // 20 ////////////////////////////////////	
23	
31 assign C = A & B; 32 - 33 (a) endmodule 34 - -	~

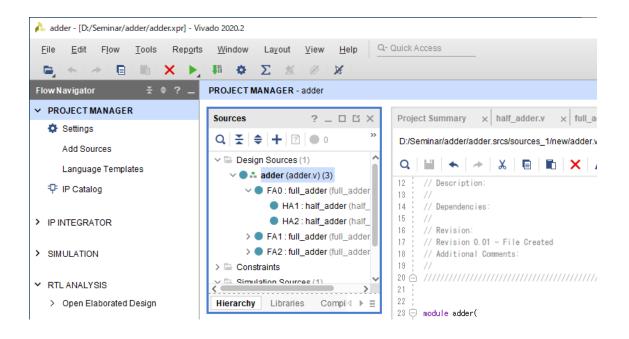
3.3) Ctrl + S to save the file.

3.4) Create and add the **full_adder.v** file into the project and write the full adder Verilog code.

3.5) Vivado automatically updates the hierarchy of the modules and instances and detects the top module of the project. The top module defines the hierarchy of the design for synthesis and implementation.



3.6) Create and add the **adder.v** file into the project and write the 3-bit adder Verilog code. It will be the top module of the project.



4. Synthesis

4.1) Click Run Synthesis in the "Flow Navigator".

> RTL ANALYSIS	Simulation Sources (1) 20 ⊕ Hierarchy Libraries Compid ▶ ≡
✓ SYNTHESIS	23 🖗 module adder(
Run Synthesis	Source File Proper ? _ C X 24 A, B, Sum, Cout 25);
> Open Synthesized Design	● adder.v ← → ☆ 26 27 input [2:0] A, B;
 IMPLEMENTATION Run Implementation 	Contraction: Cont

4.2) You will see the "Launch Runs" form. Click OK.

🝌 Launch Runs				×	;
Launch the selected s	ynthesis or imp	plementation runs.		4	
Launch directory:	🗟 <default la<="" td=""><td>unch Directory></td><td></td><td>~</td><td></td></default>	unch Directory>		~	
Options					
• Launch runs	on local host:	Number of jobs:	10	~	
◯ <u>G</u> enerate scri	pts only				
Don't show this d	ialog again				
		ОК	6	Cancel	

4.3) You can see the run status and information in the "Design Runs" tab.

Q 素 ♦ I4 ≪ ▶ ≫ + %										
Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT
C synth_1	constrs_1	Running synth_design								
▷ impl_1	constrs_1	Not started								

5. Implementation

5.1) Upon completion of the synthesis process, you will see the "Synthesis Completed" dialogue. Select **Run Implementation** and click **OK**.

Synthesis Completed				
Synthesis successfully completed.				
Next				
<u> R</u> un Implementation				
Open Synthesized Design				
◯ <u>V</u> iew Reports				
Don't show this dialog again				
OK Cancel				

5.2) You will see the "Launch Runs" form. Click OK.

6. Pin Assignment

6.1) Upon completion of the implementation process, you will see the "Implementation Completed" dialogue. Select **Open Implemented Design** and click **OK**.

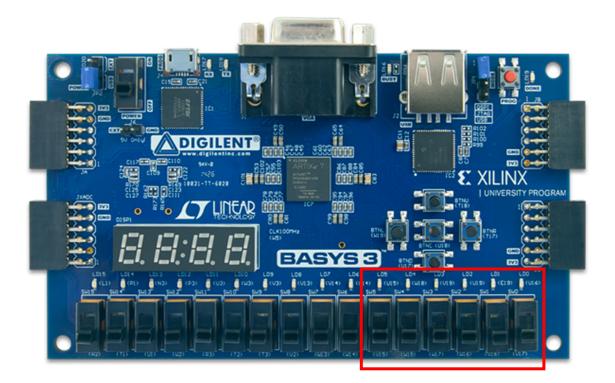
Implementation Completed				
implementation successfully completed.				
Next				
Open Implemented Design				
Generate Bitstream				
○ <u>V</u> iew Reports				
Don't show this dialog again				
OK Cancel				

6.2) To launch the I/O Planning view layout, select **I/O planning** in the **Layout** menu.

· ······] - Vivado 2020.2	- □ >
<u>File Edit Flow T</u> ools Rep <u>o</u>	ports Window Layout View Help Q- Quick Access	Implementation Complete
	× ∞ • • • • • • • • • • • • • • • • • •	🔛 I/O Planning
- FlowNavigator 🛨 🔶 ? _	IMPLEMENTED DE SIGN - xc7a35tcpg236-1	?
PROJECT MANAGER	Sources Netlist Device Constraints × ? _ □ Ľ Package × Device ×	0.0.0
🔅 Settings		? 🗆 🖸
Add Sources	$ \mathbf{Q} \underbrace{\mathbf{X}} \mathbf{\varphi} = \mathbf{Q} \mathbf{Q} \underbrace{\mathbf{X}} \underbrace{\mathbf{Q}} \mathbf{Q} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \mathbf{Q} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \mathbf{Q} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \mathbf{Q} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{X}} \underbrace{\mathbf{Q}} \underbrace{\mathbf{X}} $	\$
Language Templates	✓ Internal VREF 1 2 3 4 5 6 7 8 9 10 11 12	2 13 14 15 16 17 18 19
👎 IP Catalog	C 0.675V C C C C C C C C C C C C C C C C C C C	
	© 0.75V	
IP INTEGRATOR		
	ч 🕞 NONE (4)	
SIMULATION	🦘 I/O Bank 14 🗸 🗸 🕹 🐇 🕹 🐇	
RTL ANALYSIS	Drop I/O banks on voltages or the "NONE" folder to set/unset	
RILANALYSIS	Internal VREF.	
SYNTHESIS		
Run Synthesis	Properties × Clock Regions ? _ C	
> Open Synthesized Design	Select an object to see properties	
	Tcl Console Messages Log Reports Design Runs Timing Power DRC Package Pins	
IMPLEMENTATION Run Implementation		VO Ports × ? _ D
IMPLEMENTATION Run Implementation Open Implemented Design	Tcl Console Messages Log Reports Design Runs Timing Power DRC Package Pins	
IMPLEMENTATION Run Implementation Open Implemented Design Constraints Wizard 	Tcl Console Messages Log Reports Design Runs Timing Power DRC Package Pins Q X Image: Console	Vcco Vre
IMPLEMENTATION Run Implementation Open Implemented Design	Tcl Console Messages Log Reports Design Runs Timing Power DRC Package Pins Q X Image:	Vcco Vre CMOS18) • 1.800
IMPLEMENTATION Run Implementation Open Implemented Design Constraints Wizard 	Tcl Console Messages Log Reports Design Runs Timing Power DRC Package Pins Q ★	Vcco Vre CMOS18) • 1.800 CMOS18) • 1.800
IMPLEMENTATION Run Implementation Open Implemented Design Constraints Wizard Edit Timing Constraints 	Tcl Console Messages Log Reports Design Runs Timing Power DRC Package Pins Q X Image:	Vcco Vre CMOS18) • 1.800 CMOS18) • 1.800
IMPLEMENTATION Run Implementation Open Implemented Design Constraints Wizard Edit Timing Constraints Timing Constraints	Tcl Console Messages Log Reports Design Runs Timing Power DRC Package Pins Q ★	Vcco Vre CMOS18) • 1.800 CMOS18) • 1.800

6.3) In the "I/O Ports" tab, set the **Package Pin** and **I/O Std** for the BASYS3 board.

cl Console Me	ssages Log Rep	orts Design	Runs Timing	Pov	wer D	RC	Package Pins	I/O Ports	×
Q ¥ ♦ [4 + 5				_				
Name	Direction	Neg Diff Pair	Package Pin		Fixed	Bank	I/O Std		Vcco
🗸 🕞 All ports (10)									
🗸 🧕 A (3)	IN					14	LVCMOS33*	*	3.30
🕑 A[2]	IN		W16	\sim		14	LVCMOS33*	*	3.30
🕑 A[1]	IN		V16	~		14	LVCMOS33*	*	3.3
🕑 A[0]	IN		V17	~		14	LVCMOS33*	*	3.3
🗸 💁 В (З)	IN					14	LVCMOS33*	*	3.3
▶ B[2]	IN		V15	~		14	LVCMOS33*	*	3.3
▶ B[1]	IN		W15	~		14	LVCMOS33*	*	3.3
▶ B[0]	IN		W17	\sim		14	LVCMOS33*	*	3.3
🗸 🍯 Sum (3)	OUT					14	LVCMOS33*	*	3.3
🕑 Sum[2]	OUT		U19	~		14	LVCMOS33*	*	3.3
🕑 Sum[1]	OUT		E19	~		14	LVCMOS33*	*	3.3
🛃 Sum[0]	OUT		U16	~		14	LVCMOS33*	*	3.30
🗸 🕞 Scalar port	ts (1)								
Cout	OUT		V19	~		14	LVCMOS33*	v	3.3



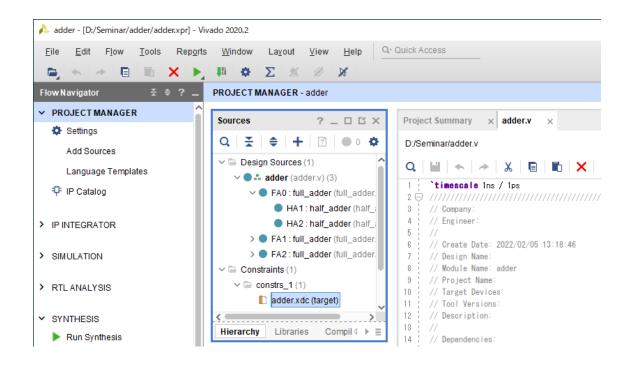
6.3) Ctrl + S to save the pin assignment. You will see the "Out of Date Design" dialog. Click OK.



6.4) You will see the "Save Constraints" form. Enter adder in the "File name" field and click OK.

🝌 Save Constraints		×
Select a target file to write r Choosing an existing file w constraints.		
• <u>C</u> reate a new file		
<u>F</u> ile type:	XDC	~
F <u>i</u> le name:	adder	\otimes
Fil <u>e</u> location:	<local project="" to=""></local>	~
Select an existing fil	e	
<select a="" targ<="" td=""><td>et file></td><td>\mathbf{v}</td></select>	et file>	\mathbf{v}
?	ок	Cancel

6.5) The constraint file **adder.xdc** will be added to the "Constraints" folder in the "Sources" pane of the "PROJECT MANAGER".



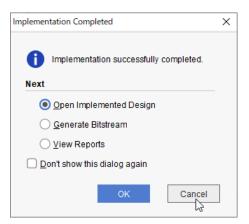
6.6) Click Run Implementation in the "Flow Navigator" to rearrange the layout.

 ✓ SYNTHESIS ▶ Run Synthesis 	↓ 12 // Description: Hierarchy Libraries Compil ⊲ ▶ ≡ 13 14 // Dependencies:
> Open Synthesized Design	Source File Propert ? _ 🗆 🖾 × 15 // 18 // Revision: 17 // Revision 0.01 - File Created
	□ adder.xdc ← ⇒ ☆ 17 // Nevision 0.01 = Price Greated 18 // Additional Comments: 18 // Additional Comments:
 <u>Run Implementation</u> Open Implemented Design 	

6.7) You will see the "Synthesis is Out-of-date" dialogue. Click Yes.

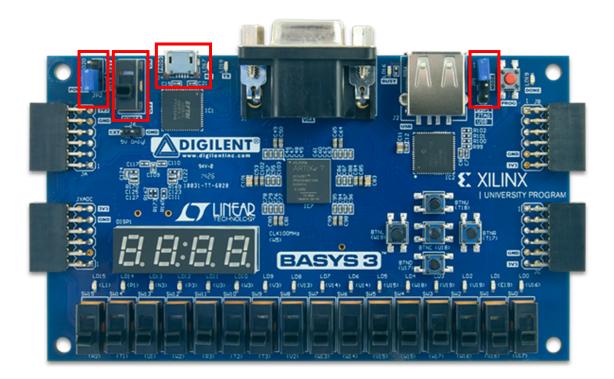


6.8) Upon completion of the implementation process, you will see the "Implementation Completed" dialogue. Click **Cancel** to ignore.



7. Programming FPGA

7.1) Make sure that JP2 of the BASYS3 board is configured to use USB as power source. Furthermore, make sure JP1 is configured to the QSPI mode. Insert a micro USB cable to J4 and connect it to your PC. Finally, switch SW16 to the ON position. You will see the demo design running.



7.1) Click Generating Bitstream in the "PROGRAM AND DEBUG" section of the "Flow Navigator".

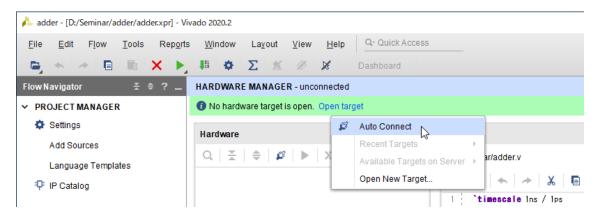


7.2) You will see the "Launch Runs" form. Click OK.

7.3) Upon completion of the bitstream generation, you will see the "Bitstream Generation Completed" dialogue. Select **Open Hardware Manager** and click **OK**.

Bitstream Generation Completed				
Bitstream Generation successfully completed.				
Next				
Open Implemented Design				
<u>View Reports</u>				
Open <u>H</u> ardware Manager				
<u>G</u> enerate Memory Configuration File				
Don't show this dialog again				
OK Cancel				

7.4) Click Open target in the "HARDWARE MANAGER" and select Auto Connect.



7.5) Click Program device.

🍌 adder - [D:/Seminar/adder/adder.xpr] - Viv	rado 2020.2		
<u>F</u> ile <u>E</u> dit F <u>l</u> ow <u>T</u> ools Rep <u>o</u> rts	<u>W</u> indow La <u>v</u> out <u>V</u> iew <u>H</u> elp	Q- Quick Access	
🕒 🐟 🕗 🗉 🐘 🗙 🕨	👫 🏟 Σ 🖄 🖉 😹 Da	ishboard 👻	
Flow Navigator 😤 🌲 ? 🔔	HARDWARE MANAGER - localhost/xilinx_tcl	/Digilent/210183B1	5F12A
✓ PROJECT MANAGER	There are no debug cores. Program devic	e Refresh device	
Settings		_ 0 6 X	adder.v
Add Sources		ð	
Language Templates		*	D:/Seminar/adder.v
후 IP Catalog	Name	Status	Q 🔛 🛧 🥕 🗶 🗉
Y IP Catalog	V I localhost (1)	Connected	1 `timescale 1ns / 1ps
	✓ ■ ✓ xilinx_tcf/Digilent/210183B15F12	Open	
> IP INTEGRATOR	✓ ∰ xc7a35t_0 (1)	Programmed	3 // Company:

7.6) You will see the "Program Device" form. Click Program.

🍐 Program Device		×
	ramming file and download it to your hardware device. You can optionally file that corresponds to the debug cores contained in the bitstream	4
Bitstre <u>a</u> m file:	D:/Seminar/adder/adder.runs/impl_1/adder.bit	•••
Debu <u>a</u> probes file:		•••
✓ Enable end of s	artup check	
?	Program Can	cel

7.7) The BASYS3 is now running your 3-bit adder!

7.8) After testing your design, press **PROG** button to refresh the device.

