**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **3 june 2020** | **Name:** | **Shreya poojary** |
| **Course:** | **HDL design** | **USN:** | **4al16ec074** |
| **Topic:** | **How to Download and Install Xilinx Vivado Design Suite for implementation of HDL code** | **Semester & Section:** | **8-B** |
| **Github Repository:** | **shreya-test** |  |  |

|  |
| --- |
|  |
| **Image of session** |
| **Report**  First download Vivado (or at least the web installer for it) from Xilinx’s website, at <http://www.xilinx.com/support/download.html>.  Once you get to the download page, choose the appropriate installer for your system; I’m on a Windows 10 machine and don’t feel the need to get a universal, all OS installer, so I’ll choose the Windows Self Extracting Web Installer. **VHDL Code for Full Adder**  |  | | --- | | library IEEE; | | use IEEE.STD\_LOGIC\_1164.ALL; | |  |  | | --- | |  | | entity full\_adder\_vhdl\_code is | |  |  |  | | --- | --- | | Port ( A : in STD\_LOGIC; | | | B : in STD\_LOGIC; |  |  |  | | --- | --- | | Cin : in STD\_LOGIC; | | | S : out STD\_LOGIC; |  |  | | --- | | Cout : out STD\_LOGIC); | | end full\_adder\_vhdl\_code; | |  |  | | --- | |  | | architecture gate\_level of full\_adder\_vhdl\_code is | |  |  | | --- | |  | | begin | |  |  | | --- | |  | | S <= A XOR B XOR Cin ; | |  |  |  | | --- | --- | | Cout <= (A AND B) OR (Cin AND A) OR (Cin AND B) ; | | |  |  |  | | --- | | end gate\_level; | |  |  **Testbench VHDL Code for Full Adder**  |  | | --- | | LIBRARY ieee; | | USE ieee.std\_logic\_1164.ALL; | |  |  | | --- | |  | | ENTITY Testbench\_full\_adder IS | |  |  |  | | --- | --- | | END Testbench\_full\_adder; | | |  |  |  |  | | --- | --- | | ARCHITECTURE behavior OF Testbench\_full\_adder IS | | |  |  |  |  | | --- | --- | | -- Component Declaration for the Unit Under Test (UUT) | | |  |  |  |  | | --- | --- | | COMPONENT full\_adder\_vhdl\_code | | | PORT( |  |  | | --- | | A : IN std\_logic; | | B : IN std\_logic; |  |  |  | | --- | --- | | Cin : IN std\_logic; | | | S : OUT std\_logic; |  |  |  | | --- | --- | | Cout : OUT std\_logic | | | ); |  |  |  | | --- | --- | | END COMPONENT; | | |  |  |  | | --- | | --Inputs | | signal A : std\_logic := '0'; | |  |  | | --- | | signal B : std\_logic := '0'; | | signal Cin : std\_logic := '0'; | |  |  | | --- | |  | | --Outputs | |  |  | | --- | | signal S : std\_logic; | | signal Cout : std\_logic; | |  |  | | --- | |  | | BEGIN | |  |  | | --- | |  | | -- Instantiate the Unit Under Test (UUT) | |  |  |  | | --- | --- | | uut: full\_adder\_vhdl\_code PORT MAP ( | | | A => A, |  |  | | --- | | B => B, | | Cin => Cin, | |  |  | | --- | | S => S, | | Cout => Cout | |  |  | | --- | | ); |  **Output Waveform for full adder VHDL Code** [Testbench Waveform for full adder VHDL Code](http://allaboutfpga.com/wp-content/uploads/2014/04/Testbench-Waveform-for-full-adder-VHDL-Code.png) |