DAILY ASSESSMENT REPORT

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **3-5-2020** | **Name:** | **Rasika Patil** |
| **Course:** | **DIGITAL DESIGN USING HDL** | **USN:** | **4AL16EC057** |
| **Topic:** | EDA Playground Online complier, EDA Playground Tutorial Demo Video,  How to Download And Install Xilinx Vivado Design Suite, Vivado Design Suite for implementation of HDLcode | **Semester & Section:** | **8th B** |
| **Github Repository** | **Rasika B Patil** |  |  |

|  |
| --- |
| **Image of session**    **Report:**   * Vivado DesignSuite for implementation of HDL code * In a separate web browser window, log in to EDA Playground * Log in Click the Log in button (top right) Then eitherclick on Google or Facebook or register by clicking on ‘Register for a full account’ (which enables all the simulators on EDA Playground) * Select ‘Aldec Riviera Pro’ from the Tools & Simulators menu. * This selects the Aldec Riviera Pro simulator, which can be used however you logged in. Using certain other simulators will require you to have registered for a full account. * In either theDesignorTestbenchwindow pane, type in the following code: * Module test; * Initial * $display("Hello World!"); * End module * Click Run(top left)Yes, running a simulation is as simple as that! * In the bottom pane, you should see realtime results as your code is being compiled and then run. A run typically takes 5seconds, depending on network traffic and simulator. Near the bottom of result output, you should see: * Hello World! * Now, let’s save our good work. Click theSharetab near in the bottom pane and then type in a name and description. Then click Save * The browser page will reload and the browser address bar will change. This is a persistent link to your saved code. You can send the link by email, post it on a web page, post it onStack Overflowforums, etc. * Now, let’s try modifying existing code. Load the following example: RAM * On the left editor pane, before then do finite al block, add the following: * write\_enable = 1; * data\_write = 8'h2C; * toggle\_clk\_write; * toggle\_clk\_read; * $display("data[%0h]: %0h", * address\_read, data\_read); * Run the sim. In the results you should see this new message: * data[1b]: 2c * Optional. Click Copy to save a personal version of the modified RAM code, including the simulation results. * Loading Waves from EDA Playground * You can run a simulation on EDA * Playground and load the resulting waves in EPWave. Loading Waves for SystemVerilog and Verilog Simulations * Go to your code on EDA Playground. For example:RAM Design and Test Make sure your code contains appropriate function calls to create a \*.vcd file. * For example:   initial  begin  $dumpfile("dump.vcd");  $dumpvars(1);  End  Select a simulator and check the Open EPWave after run check box.   * Click Run. * After the run completes, the resulting waves will load in a new EPWave window. * Loading Waves for VHDL Simulations check the Open EPWave after run checkbox. * Specify the top entity to simulate. * Click Run   After the run completes, the resulting waves will load in a new EPWave window. (Popups must be enabled.)The waves for all signals in the specified top entity and any of its components will be dumped. In EPWave window, click get Signals to select the signals to view.  **TASK:**   * **Implement 4 to 1 MUX using two 2 to 1 MUX using structural modeling style and test the module in online/offline compiler**   module mux4to1(a,sel,out);  input [3:0] a;  input [1:0] sel;  output out;  wire mux[2:0];  mux2to1 m1 (a[3],a[2],sel[0],mux\_1),  m2 (a[1],a[4],sel[0],mux\_2),  m3 (mux\_1,mux\_2,sel[1],out);  endmodule  **Output:** |