**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **6/3/20** | **Name:** | **Sathya br** |
| **Course:** | **DIGITAL DESIGN USING HDL** | **USN:** | **4al16ec065** |
| **Topic:** | **EDA Playground Online complier**  **EDA Playground Tutorial Demo Video How to Download And Install Xilinx Vivad**  **Design SuiteVivado Design Suite for implementation of HDL code** | **Semester & Section:** | **6th semister**  **B section** |
| **Github Repository:** | **sathyabr** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| **Report**  **What is EDA Playground? EDA Playground gives engineers immediate hands-on exposure to simulating SystemVerilog, Verilog, VHDL, C++/SystemC, and other HDLs. All you need is a web browser. The goal is to accelerate learning of design/testbench development with easier code sharing and simpler access to EDA tools and libraries.With a simple click, run your code and see console output in real time. View waves for your simulation using EPWave browser-based wave viewer.Save your code snippets (“Playgrounds”).Share your code and simulation results with a web link. Perfect for web forum discussions or emails. Great for asking questions or sharing your knowledge.Quickly try something outTry out a language feature with a small example.Try out a library that you’re thinking of using. Example UsecasesQuick prototyping – try out syntax or a library/language feature.When asking questions on Stack Overflow or other online forums, attach a link to the code and simulation results.Use during technical interviews to test candidates’ SystemVerilog/Verilog coding and debug skills.Try verifying using different verification frameworks: UVM, SVUnit, plain Verilog, or Python.Tools & Simulators For settings and options documentation, see Tools & Simulators OptionsAvailable tools and simulators are below. EDA Playground can support many different tools. Contact us to add your EDA tool to EDA Playground**  **Code:library IEEE;use IEEE.STD\_LOGIC\_1164.ALL;entity mux2\_1 isport(A,B : in STD\_LOGIC;S: in STD\_LOGIC;Z: out STD\_LOGIC);end mux2\_1;architecture Behavioral of mux2\_1 isbeginprocess (A,B,S) isbeginif (S ='0') thenZ <= A;elseZ <= B;end if;end process;end Behavioral;library IEEE;use IEEE.STD\_LOGIC\_1164.ALL;entity mux4\_1 isport(A,B,C,D : in STD\_LOGIC;S0,S1: in STD\_LOGIC;Z: out STD\_LOGIC**  **);end mux4\_1;architecture Behavioral of mux4\_1 iscomponent mux2\_1port( A,B : in STD\_LOGIC;S: in STD\_LOGIC;Z: out STD\_LOGIC);end component;signal temp1, temp2: std\_logic;beginm1: mux2\_1 port map(A,B,S0,temp1);m2: mux2\_1 port map(C,D,S0,temp2);m3: mux2\_1 port map(temp1,temp2,S1,Z);end Behavioral** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date:** | **6/3/20** | **Name:** | **Sathya br** | |
| **Course:** | **Python Core and Advanced** | **USN:** | **4al16ec065** | |
| **Topic:** | **Functions** | **Semester & Section:** | **6th semister**  **B section** | |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session** | | | |
| **Report – Report can be typed or hand written for up to two pages.**   * **Introduction** * **first function** * **returning a result** * **return multiple values** * **local and global variables** * **accessing global variable with the same name** * **assign function to a variable** * **function inside another** * **function as parameter to an other** | | | |