**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **05-06-2020** | **Name:** | **Kavya M M** |
| **Course:** | **HDL** | **USN:** | **4AL17EC040** |
| **Topic:** | **Verilog HDL** | **Semester & Section:** | **6th A** |
| **Github Repository:** | **Kavya\_ECE040** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
|  |
| Design style:   * Bottom−up design * Top−down methodology.   Abstract levels of Verilog:   * Behavioural level * Register−Transfer Level * Gate Level   History Of Verilog  Verilog was started initially as a proprietary hardware modeling language by Gateway Design Automation Inc. around 1984. It is rumored that the original language was designed by taking features from the most popular HDL language of the time, called HiLo as well as from traditional computer language such as C. At that time, Verilog was not standardized and the language modified itself in almost all the revisions that came out within 1984 to 1990.  Various stage of ASIC/FPGA:   * Specification: Word processor like Word, Kwriter, AbiWord, Open Office. * High Level Design: Word processor like Word, Kwriter, AbiWord, for drawing waveform use tools like wave former or testbenches or Word, Open Office. * Micro Design/Low level design: Word processor like Word, Kwriter, AbiWord, for drawing waveform use tools like wave former or test bencher or Word. For FSM State CAD or some similar tool, Open Office. * RTL Coding: Vim, Emacs, context, HDL Turbo Writer * Simulation: Modalism, VCS, Verilog−XL, Veriwell, Finsim, iVerilog, VeriDOS. * Synthesis: Design Compiler, FPGA Compiler, Synplify, Leonardo Spectrum. You can download this from FPGA vendors like Altera and Xilinx for free. * Place & Route: For FPGA use FPGA' vendors P&R tool. ASIC tools require expensive P&R tools like Apollo. Students can use LASI, Magic. * Post Si Validation: For ASIC and FPGA, the chip needs to be tested in real environment. Board design, device drivers needs to be in place.   Verilog operators:  Arithmetic operators:  •Binary: +, −, \*, /, % (the modulus operator)  • Unary: +, − (This is used to specify the sign)  • Integer division truncates any fractional part  • The result of a modulus operation takes the sign of the first operand  • If any operand bit value is the unknown value x, then the entire result value is x  Register data types are used as unsigned values  Relational operators:    Equality operator:  There are two types of Equality operators. Case Equality and Logical Equality.    Logical operator:    Bit wise operator:  Bitwise operators perform a bit wise operation on two operands. They take each bit in one operand and perform the operation with the corresponding bit in the other operand. If one operand is shorter than the other, it will be extended on left side with zeros to match the length of the longer operand.    Reduction operator:     * Reduction operators are unary. * They perform a bit−wise operation on a single operand to produce a single bit result. * Reduction unary NAND and NOR operators operate as AND and OR respectively, but with their outputs negated.   Shift operator:     * The left operand is shifted by the number of bit positions given by the right operand. * The vacated bit positions are filled with zeroes.   Operator precedence:    Verilog HDL abstract level:  • Behavioral Models: Higher level of modeling where behavior of logic is modeled.  • RTL Models: Logic is modeled at register level  • Structural Models: Logic is modeled at both register level and gate level.  Task:  Implement a Verilog module to count number of 0’s in a 16-bit number in compiler.    Output: |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date:** | **05-06-2020** | **Name:** | **Kavya M M** | |
| **Course:** | **Python** | **USN:** | **4AL17EC040** | |
| **Topic:** | **Application** | **Semester & Section:** | **6th A** | |
|  |  |  |  | |
|  |  |  |  | |
|  |  |  |  | |
| **AFTERNOON SESSION DETAILS** | | | |
|  | | | |
| Bokeh:   * Bokeh is python library used in data visualization * We can grab data from various source data file formats such as csv, python list, json file so on * Bokeh produces interactive graph       #Plotting percentage of women who received an engineering degree over years    #importing bokeh and pandas  from bokeh.plotting import figure  from bokeh.io import output\_file, show  import pandas    #prepare some data  df=pandas.read\_csv("http://pythonhow.com/data/bachelors.csv")  x=df["Year"]  y=df["Engineering"]    #prepare the output file  output\_file("Line\_from\_bachelors.html")    #create a figure object  f=figure()    #create line plot  f.line(x,y)    #write the plot in the figure object  Show(f) | | | |