Birla Institute of Technology and Science Pilani
CS F342 Computer Architecture
Second Semester 2020-21
Lab Sheet– 4 (18th March 2021)

FSM & Memory in Verilog HDL

In today's lab we shall implement FSMs in Verilog HDL and write a suitable test bench to check its functionality. Along with this, we shall implement Memories in Verilog HDL and learn how to write and read data into them using system tasks.

Learning Outcome: By the end of this lab. session the learner will be able to

- Implement an FSM using Verilog HDL, given its state diagram.
- Write an effective test bench to verify the FSM.
- Use ***$random*** system task to generate random test vectors.
- Implement memory in Verilog HDL.
- Use ***$readmemh*** system task to populate a large memory instantaneously.
- Write effective test cases to verify the functionality of the memory implemented.

Exercise Problems

For Q1 & Q2: The clock (CLK) signal is positive edge triggered.
Reset (RST) signal is active high and synchronous.

1. Implement a *Moore* type FSM to detect a palindrome of length 3-bits in a serial input bit stream. Example of three bit palindromes 111,101,010,000.
2. Implement a *Mealy* type FSM to detect a palindrome of length 3-bits in a serial input bit stream. Example of three bit palindromes 111,101,010,000.
3. Implement a Memory which has a synchronous write and asynchronous read capability. The writing of data takes place on the negative edge of the CLK signal. The memory is a 64 Byte memory with a 8-bit data-bus. In the test bench, initialize the memory using a file named *dummy.dat (the data in this file is in hex format and can be random values of your choice).* Read the data from each address-location and replace it with it grey code equivalent. To convert binary to grey you can use data-flow modeling.

Submission Link: https://forms.gle/pGKsytyfwygKUXXP6

****** The End ******

-Prepared By
K. Babu Ravi Teja
Assistant Professor , EEE Department
BITS Pilani, Pilani Campus