Project 1: Environment Setup and FSM

Objectives:

1) Build your Verilog environment
2) Explore a simple FSM

Part A): Sample FSM

Using the SampleFSM Verilog files, build and successfully run the FSM sample using the Getting Started and Lab 1 documents as a guide. Make sure you understand what's going on in the Verilog by reviewing the syntax AND reviewing waveform outputs. Learn how to scale and show results on Waveform viewers that are meaningful.

Repeat the exercise using the FIFO Verilog files.

Results:

1) Show results of SampleFSM and Testbench on Waveform Viewer.
    a. Show all inputs and outputs
    b. Show State
    c. Place one blank row between each waveform for easier viewing
    d. Show all results until steady state
2) Build a FSM to count from 0 to 7 three times.
    a. Have a reset signal and use three bits for the count. If reset is ever active (high), counter goes back to zero on next clock. If reset is low, the FSM counts based on the rules.
    b. The counter will count from 0 to 7 only three times. It will hold the value 7 on the third count's completion. In other words, if reset stays low long enough, the counter will count 0 to 7, 0 to 7, and 0 to 7 and hold 7 until reset goes high.
    c. Show your Verilog working on a waveform viewer (gtkwave).

Test Checks

Part A): Sample FSM

Using the SampleFSM Verilog files, build and successfully run the FSM sample using the Getting Started and Lab 1 documents as a guide. Make sure you understand what's going on in the Verilog by reviewing the syntax AND reviewing waveform outputs. Learn how to scale and show results on Waveform viewers that are meaningful.

Repeat the exercise using the FIFO Verilog files.

Results:

1) Show results of SampleFSM and Testbench on Waveform Viewer.
    a. Show all inputs and outputs
    b. Show State
    c. Place one blank row between each waveform for easier viewing
    d. Show all results until steady state
2) Build a FSM to count from 0 to 7 three times.
    a. Have a reset signal and use three bits for the count t.
    b. The counter will count from 0 to 7 only three times. It will hold the value 7 on the third count's completion
    c. Show your Verilog working on a waveform viewer (gtkwave).