ECE 510: Pre-Silicon Validation

Laboratory 0



Nilesh Dattani

PSU ID: 958880624

Table of Contents

[Problem Description: 3](#_Toc416378534)

[ALU Implementation: 6](#_Toc416378535)

[Assumptions: 7](#_Toc416378536)

[Conclusion and Results: 7](#_Toc416378537)

[Figure 1: Module Level Block Diagram 4](#_Toc416378464)

[Figure 2: FSM Implementation 5](#_Toc416378465)

[Figure 3: Module Level Block Diagram Tester 6](#_Toc416378466)

[Figure 4: HDL Diagram 7](#_Toc416378467)

# 

# Problem Description:

Design an FSM with Description in the document below:



A short description to the problem is as below:

* Below is the module level block diagram



Figure : Module Level Block Diagram

This is a simple ALU which supports 4 operations. Binary addition (ADD), Binary Subtraction (SUB), Bitwise parity calculation (PAR) and Bitwise comparator (COMP)

Opcode definition:

* ADD: 2'b00
* SUB: 2'b01
* PAR: 2'b10
* COMP: 2'b11

Description of inputs

* clk: Free running clock whose frequency can be set by a parameter (CLOCK\_PERIOD). Default: 10ns
* seset\_n: Active low reset.
* opcode\_valid: active high input to control the internal FSM.
* Opcode: 1 bit opcode field that must be time-multiplexed for two bits needed internally for 4 opcodes.
* Data [7:0]: 8-bit data field that also needs to be time-multiplexed for two operands.

The FSM to be implemented is as below



Figure : FSM Implementation

We are provided with a tester attached and also module level block diagram is as below:





Figure : Module Level Block Diagram Tester

# ALU Implementation:

* The requirement state that we need to use re-usable, synthesizable, modular Verilog code for our design
* The modular design with leaf level modules looks like the hierarchical design below.
* Each functionality has been implemented at gate level using logical operators which synthesizes to gate directly
* FSM has been implemented using 3 always block approach
* Care has been taken which instantiating modules and each signal has been referenced by its name in the design
* Leaf Level modules are re-usable and higher level modules are re-configurable.

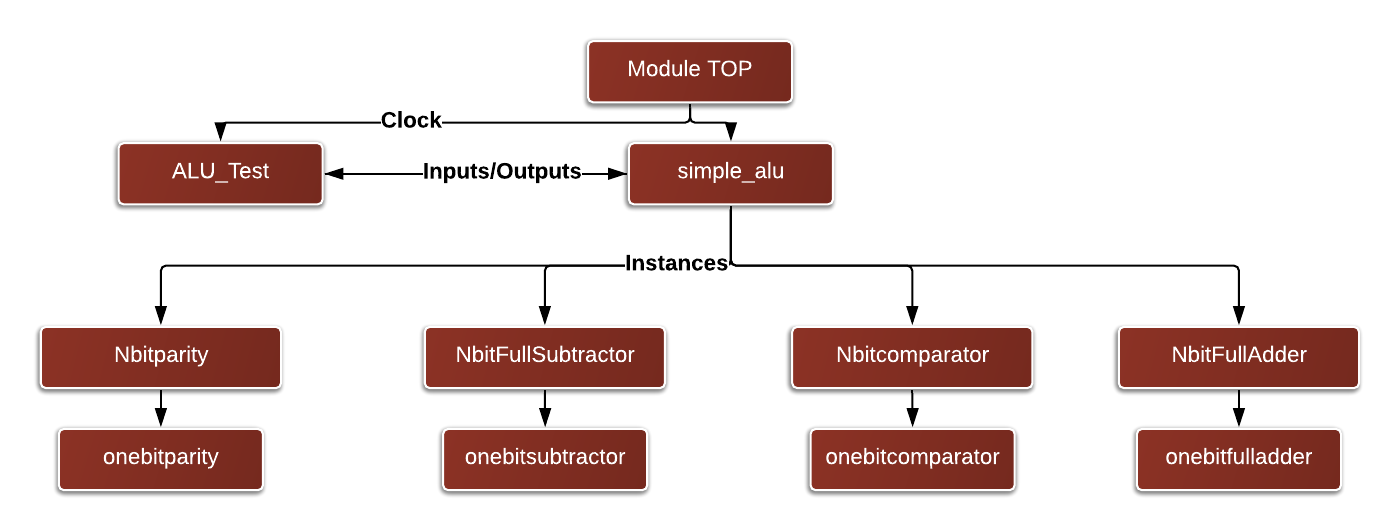


Figure : HDL Diagram

* The gate level schematic is as in the file attached below



* Gate level schematic was generated from code using Xilinx tool
* Care was taken to make the design synthesizable
* Makefile and do files were used for easy simulation, readme.txt have been included for other users

# Assumptions:

* FSM State IDLE should clear the done bit and should not change any other outputs, i.e. hold their values
* Logical Operators are the lowest level of code writing expected
* Delay were not accounted for the simulation
* Race conditions could be ignored
* Arithmetic operations are unsigned operations.ss

# Conclusion and Results:

* The code was simulated and verified with the tester code provided
* Transcript and the source codes are provided with the report in the submission folder
* Files were arranged as required and submitted to d2l.pdx.edu