ECE 501 : ASSIGNMENT 8 – ALU TESTER

Roshni Uppala

1011735230

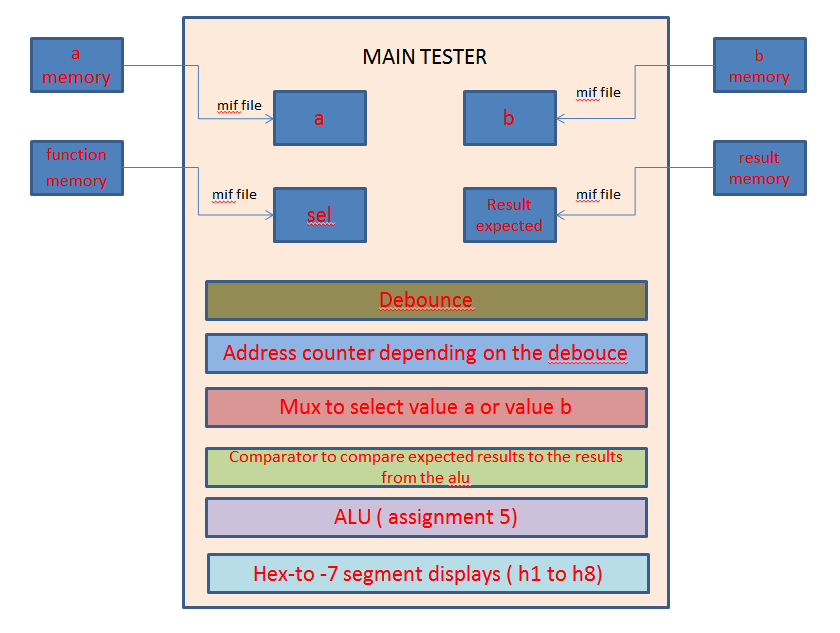
*Objective : To design a system to test the ALU designed in the assignment 5 of this course. The testing is to be done using the DE2 board. Such a test must hold multiple memories to hold test data and the output should be displayed on the 7 segment displayed .*

**Design Specification :**

* Inputs :
* A test memory – 12 bits wide contains data for the A input to the ALU
* B test memory- 12 bits wide for the B input to the ALU
* Function test memory- 4 bits wide contains the function to be executed
* Results test memory- 15 bits wide , these are the expected results to be compared to the ALU results.
* Output:
* The 7 segment display for mux output, results, function and test number on the DE2 board.
* Functional behavior :
* Mux : This is a 2 to 1 multiplexer that selects which 12 bits of data to display
* ALU used here is from the assignment 5.
* Comparator : to compare the expected results to the ALU generated results, lights up the LEDS if a difference between expected and generated results are observed.
* HEX0,1,2 : MUX OUTPUT DISPLAY
* HEX3,4,5 : RESULTS OF THE ALU
* HEX 6: FUNCTION to be executed.
* HEX7 : Test number

**Design Structure:**

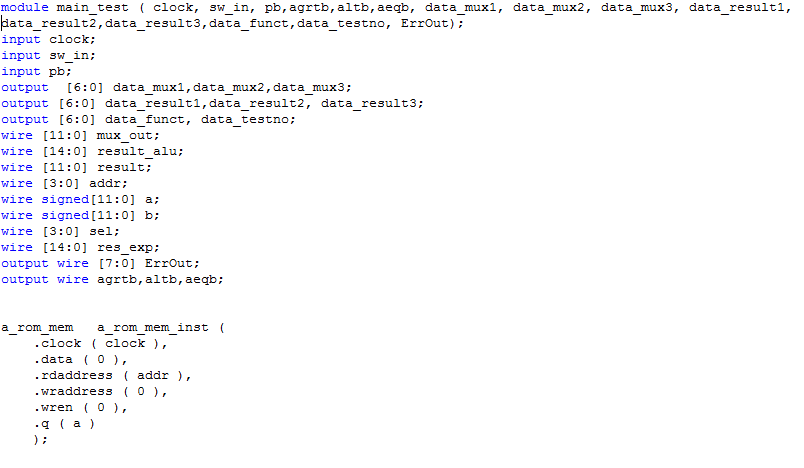
* Inputs :
* Allmemories are 8 words of length . Memories are created for a, b, function, expected results.
* Clock of 50 Mhz
* A push button to change the test data
* A switch sw0 is used to give the mux selections .
* Mif files of all memories are linked in the main\_ test program.
* A test memory – 12 bits wide contains data for the A input to the ALU
* B test memory- 12 bits wide for the B input to the ALU
* Function test memory- 4 bits wide contains the function to be executed
* Results test memory- 15 bits wide , these are the expected results to be compared to the ALU results.
* Output:
* The 7 segment display for mux output, results, function and test number on the DE2 board.
* HEX0,1,2 : MUX OUTPUT DISPLAY
* HEX3,4,5 : RESULTS OF THE ALU
* HEX 6: FUNCTION to be executed.
* HEX7 : Test number
* Functional behavior :
* Mux : This is a 2 to 1 multiplexer that selects which 12 bits of data to display
* ALU used here is from the assignment 5.
* Comparator : to compare the expected results to the ALU generated results, lights up the LEDS if a difference between expected and generated results are observed.
* The ALU follows a set of cases defined in assignment 5 .
* A debounce program is used to count the number of pushes of the push button in a single clock and give a single pulse.
* The address counter code is used to increment the address locations for each test depending upon the push button.

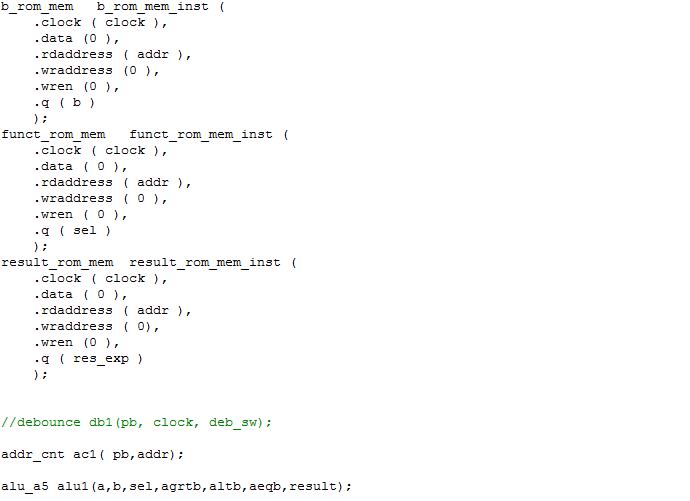


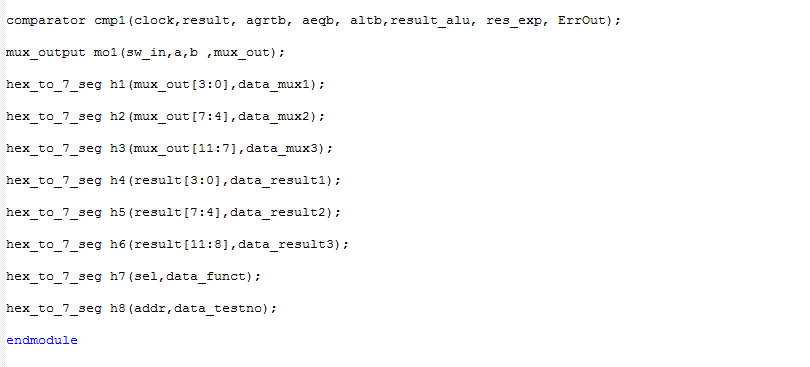
*Fig 8.1 : Tester program overview*

**Design Entry :**

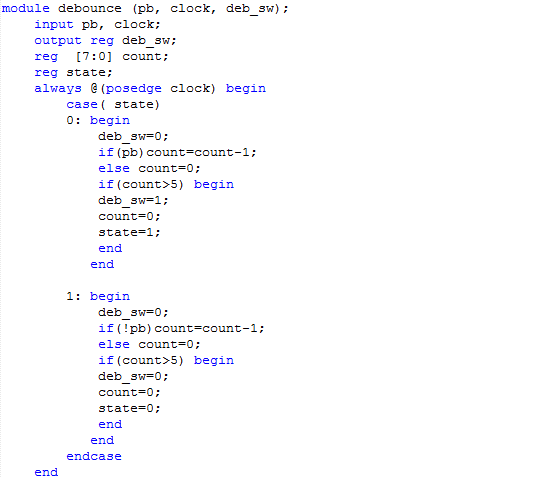
* Verilog code :





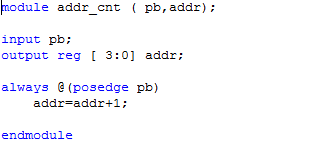


*Fig 8.2: Main tester program to test the alu*

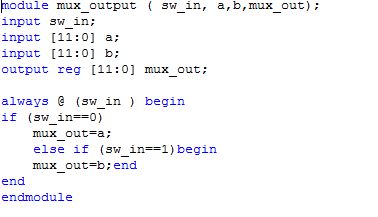




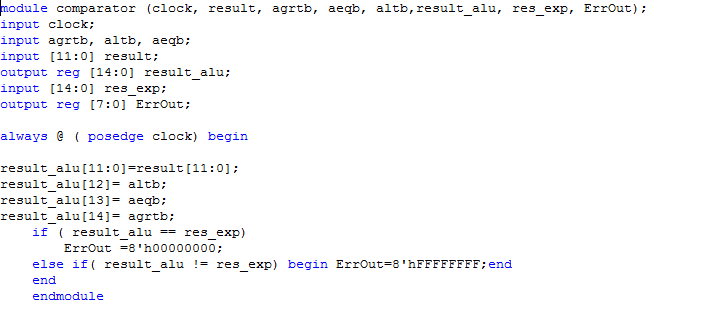
*Fig 8.3 : Debounce code*



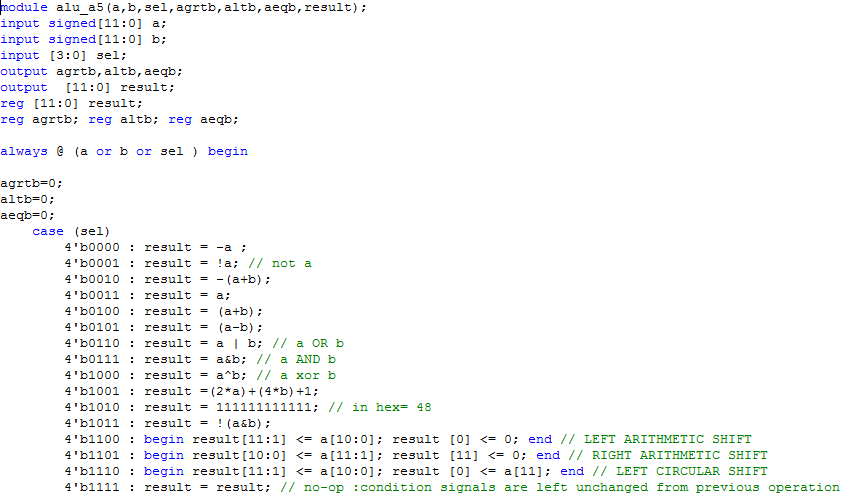
*Fig 8.4 : address counter code*

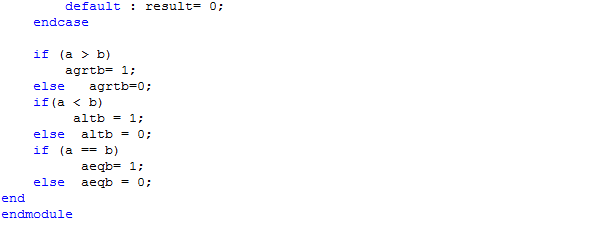


*Fig 8.5 : Mux code*

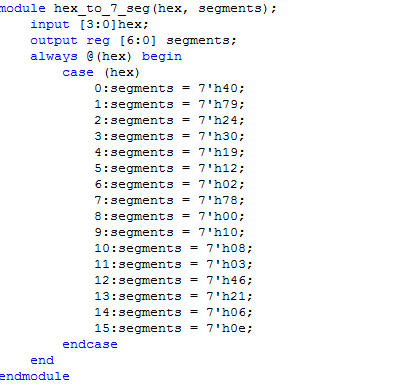


*Fig 8.6 : Comparator code*



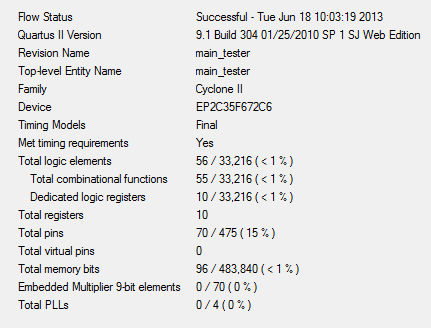


*Fig 8.7 : ALU from assignment 5*



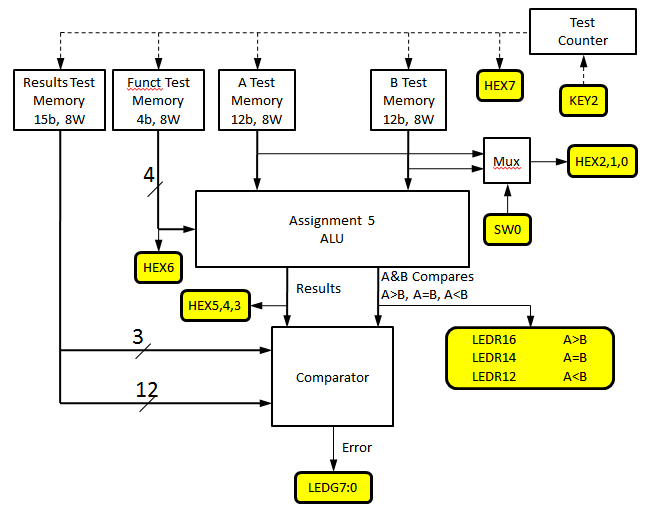
*Fig 8.8 : Hex- to –7 segment display code*

* Compilation report :



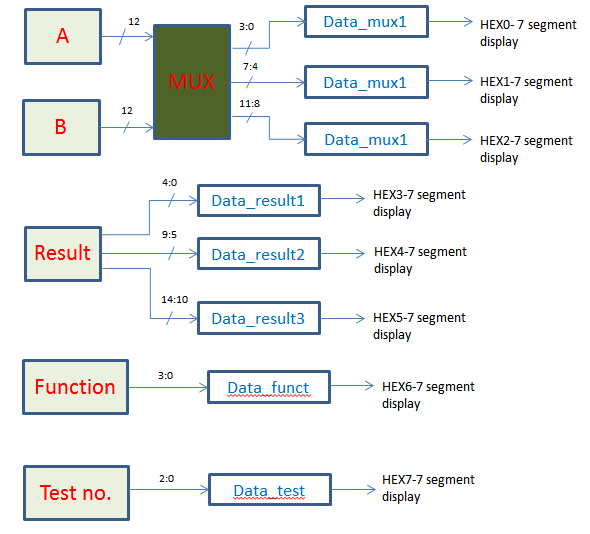
*Fig 8.9 : ALU tester compilation report*

* Top level block diagram :



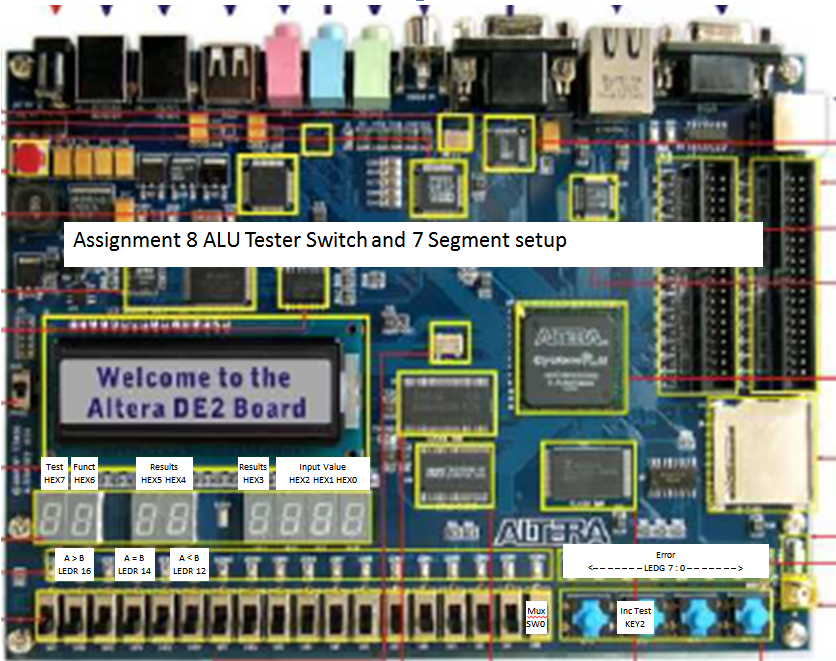
*Fig 8.10: Top level block diagram*

* 7 segment display block diagram :



*Fig 8.11: 7-segment display block diagram*

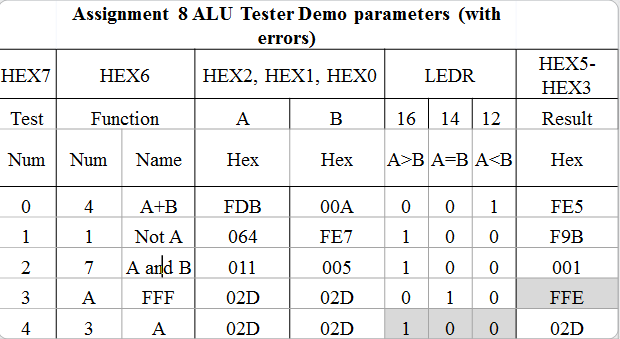
* DE2 diagram :



*Fig 8.12 : DE2 board*

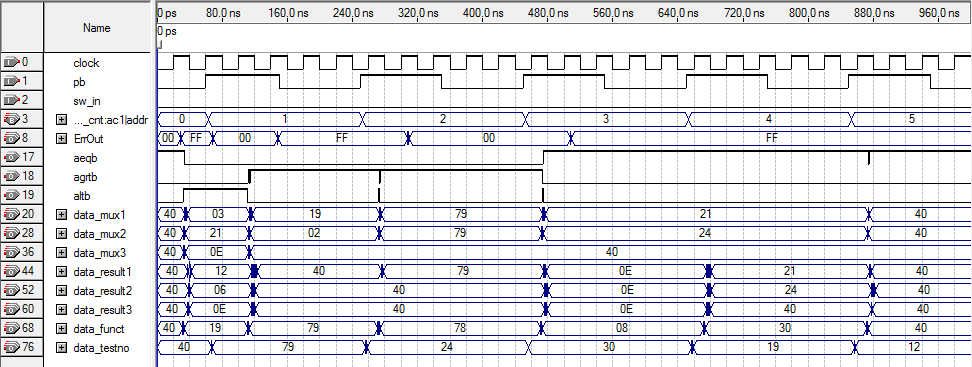
**Design Verification :**

Test Plan :



*Fig 8.13 : the verification of the design is done using the values given in class*

Simulation results :



Data\_results1, 2, 3 = 02d represents the values in table above

Data\_results1, 2, 3 = 001 represents the values in table above

Data\_mux1, mux2, mux3 represents the value in memory a = 064

*Fig 8.14 : Simulation results from the above test data*

**Result :**

Overall Result : The ALU test designed has been designed and is successfully tested on the ALU of the assignment 5. From the reports available in the Altera Quartus it takes 10.8 ns for the process. The results are successfully seen on the 7 segment display of the DE2 Board. The debounce program used here , partially worked since it only took 5 counts per clocks. The output was partially right using the debounce code. This code with the debounce code included needs more future work.