Single Register Test Plan for ALU Right

Test:	ALU Right Register On Reset
Section:	5.6, 2.0
What is being tested:	● On rst_b the ALU right register is reset to all 0's ○ Start with nonzero values (F's, As, and/or 5's) before resetting in each state ○ Reset ○ Read back to ensure the values got resetted, all values are 0

Test:	Reset State Read/Write
Section:	5.6, 6.1
What is being tested:	● When in reset state be able to read and write all F's, 5's, 0's and A's ○ Reset the register ■ Write all F's into the register ■ Read that value to see if the value written was what was asserted ■ Repeat with 5's, 0's, then A's

Test:	Normal State Read/Write
Section:	5.6, 6.2
What is being tested:	● Be able to read and write all F's, 5's, 0's and A's in this state ○ Reset the register and then transition into the Normal State ■ Write all F's into the ALU right register ■ Read that value to see if the value written was what was asserted ■ Repeat with 5's, 0's, then A's

Test:	Export Violation State Read/Write
Section:	5.6, 6.3
What is being tested:	● Try to read in the Export Violation State ○ Write all F's to the ALU Right register ○ Transition to Export Violation State ○ Try to read from Export Violation State ■ Should see only 0's, not the value that was written beforehand ○ Repeat with 5's and A's ● Try to write in the Export Violation State ○ In the Export Violation State, write all F's into the ALU right register

	○ Try to read the ALU right register ■ Should see only 0’s, not the value written ○ Repeat with 5’s and A’s

Test:	Error State Read/Write
Section:	5.6, 6.4
What is being tested:	● When in this state, try to read all values: F's, 5's, 0's, and A's ○ Write all F’s to the ALU Right register ○ Transition to Error State ○ Read from the Right ALU register and check to see if the value written was what was asserted ○ Should return the written value for reading each time ○ Repeat with 5’s, 0’s and A’s ● Try to write in the Error State for all values: F's, 5's, 0's, and A's ○ Write a different value besides the one being test ■ Since testing F, write 5, 0, or A ○ Transition to Error State ○ Write to ALU Right register the value F to try to overwrite the previous write ○ Read from Right ALU register to see if the value was overwritten

	○ Should return the first write before transitioning to the Error State; not the value (F in this case) that was trying to overwrite ○ Repeat with 5's, 0's and A's ■ Changing the value written before transitioning to Error State to be different from the value trying to overwrite

Test:	Byte Enables
Section:	5.6, 3.0
What is being tested:	● Only need to do in reset and normal states ● Write with byte enables both zero and make sure contents don't change ○ Write 5555 into the register and then change the byte enables to 00 and try to write AAAA into the register. Then read the register. ■ The pattern read should be 5555 not AAAA ● Do the same thing for both byte enables 01 and 10, ensure only one byte changes ○ (01) Put 5555 in the register and write AAAA and should read 55AA ○ (10) Put FFFF in the register and write AAAA and

	should read AAFF

Test:	Chip Selects
Section:	5.6, 3.0
What is being tested:	● Turn the chip_select signal off, try to read and write all values: F's, 5's, and A's ○ Should be unable to read or write ■ Write all F's into the ALU right register ■ Read that value to see if the value written was what was asserted ■ A value of 0 should return, since unable to write ■ Repeat with 5's, 0's, then A's ● Ensure chip_select signal is on for all other tests ○ Write all F's into the ALU right register ○ Read that value to see if the value written was what was asserted ○ The value should return, since chip select was on ○ Repeat with 5's, 0's, then A's

Test:	Aliasing
Section:	5.6, 3.0, 5.0
What is being tested:	● Address bus is 7 bits, but internal addressing only uses 6 bits. Check that the right ALU register is only read when using the 6'h14 and not the 7'h54 (where the 7th bit is a 1 and not 0) ○ Write a non-zero value to the right ALU register then attempt to read it from the address 7'h54 ■ Ensure that the value read from the address 7'h54 is all 0's and not the written value ○ Reset the right ALU register, then write a non-zero value to the address 7'h54 then attempt to read it from the right ALU register ■ Ensure that the value read from the right ALU register is all 0's and not the written value