File: B:/Stony Brook/ESE345/4_stage_multi/ALU_Workspace/ALU/src/alu_tb.vhd

```
1
2    library ieee;
3    use ieee.std_logic_1164.all;
4    use ieee.numeric_std.all;
5    use work.all;
6
7    entity alu_tb is
8    end alu_tb;
9
10
11   architecture tb of alu_tb is
12       signal r1, r2, r3, o : std_logic_vector(127 downto 0);
13       signal instrc : std_logic_vector(24 downto 0);
14       constant clk : time := 5us;
15   begin
16       UUT : entity alu
17           port map (r1 => r1, r2 => r2, r3 => r3, instrc => instrc, o => o);
18
19           process begin
20
21
22   --r4 register set up
23   --r1
24   --r1(127 downto 112)    <= std_logic_vector(to_signed(10, 16));
25   --r1(111 downto 96)     <= std_logic_vector(to_signed(10, 16));
26   --r1(95 downto 80)  <= std_logic_vector(to_signed(10, 16));
27   r1(127 downto 96)   <= std_logic_vector(to_signed(-2147483648, 32));
     --underflow
28
29   r1(95 downto 64)    <= std_logic_vector(to_signed(2147483647, 32));
     --overflow
30
31   r1(63 downto 48)    <= std_logic_vector(to_signed(10, 16));
32   r1(47 downto 32)    <= std_logic_vector(to_signed(10, 16));
33   r1(31 downto 16)    <= std_logic_vector(to_signed(10, 16));
34   r1(15 downto 0)     <= std_logic_vector(to_signed(10, 16));
35
36   --r2
37   r2(127 downto 112)  <= std_logic_vector(to_signed(32767, 16));  --for
     underflow
38   r2(111 downto 96)   <= std_logic_vector(to_signed(32767, 16));  --for
     underflow
39   r2(95 downto 80)    <= std_logic_vector(to_signed(32767, 16));  --for
     overflow
40   r2(79 downto 64)    <= std_logic_vector(to_signed(32767, 16));  --for
     overflow
41   r2(63 downto 48)    <= std_logic_vector(to_signed(-2, 16));
42   r2(47 downto 32)    <= std_logic_vector(to_signed(-2, 16));
43   r2(31 downto 16)    <= std_logic_vector(to_signed(-5, 16));
44   r2(15 downto 0)     <= std_logic_vector(to_signed(2, 16));
45
46   --r2
47   r3(127 downto 112)  <= std_logic_vector(to_signed(-32768, 16));--for
     underflow
48   r3(111 downto 96)   <= std_logic_vector(to_signed(-32768, 16));--for
     underflow
```

```
49    r3(95 downto 80)    <= std_logic_vector(to_signed(32767, 16)); --for
      overflow
50    r3(79 downto 64)    <= std_logic_vector(to_signed(32767, 16)); --for
      overflow
51    r3(63 downto 48)    <= std_logic_vector(to_signed(20, 16));
52    r3(47 downto 32)    <= std_logic_vector(to_signed(20, 16));
53    r3(31 downto 16)    <= std_logic_vector(to_signed(10, 16));
54    r3(15 downto 0)     <= std_logic_vector(to_signed(20, 16));
55
56
57    wait for clk;
58    instrc <= "00000111010010010010000000"; -- imm expected to load
      0111010010010010 = hex 7492, position from pos 16-0
59    wait for clk;
60
61
62    --R4 Instructions
63    --expected result
64    --last 127 to  bit field has overflow
65    --next 32 bit field has underflow
66    --next 32 bit field has regular negative
67    --next regular multiply add
68
69
70    -- 32 bit
71    -- 32 bit Signed Integer Multiply-Add Low with Saturation
72    instrc <= "10000000000000000000000000";
73    wait for clk;
74
75
76    -- 32 bit Signed Integer Multiply-Add high with Saturation
77
78    instrc <= "10001000000000000000000000";
79    wait for clk;

80
81
82
83    -- 32 bit Signed Integer Multiply-sub low with Saturation
84    instrc <= "10010000000000000000000000";
85    wait for clk;
86
87    -- 32 bit Signed Integer Multiply-sub high with Saturation
88    instrc <= "10011000000000000000000000";
89    wait for clk;
90
91
92    -- 64 bit
93    --nop to divide sections
94    instrc <= "11000000000000000000000000";  --"1800000" hex
95    wait for clk;
96
97    -- Long Signed Integer Multiply-Add Low with Saturation
98    instrc <= "10100000000000000000000000";
99    wait for clk;
100
101
```

```
102  -- Long Signed Integer Multiply-Add high with Saturation
103
104  instrc <= "10101000000000000000000000";
105  wait for clk;

106
107
108
109  -- Long Signed Integer Multiply-sub low with Saturation
110  instrc <= "10110000000000000000000000";
111  wait for clk;
112
113  -- Long Signed Integer Multiply-sub high with Saturation
114  instrc <= "10111000000000000000000000";
115  wait for clk;
116
117
118
119  --r3(instr) register set up
120  --r1
121  r1(127 downto 112)  <= std_logic_vector(to_signed(0, 16)); --for counting
     zeros
122  r1(111 downto 96)   <= std_logic_vector(to_signed(0, 16));
123  r1(95 downto 80)    <= std_logic_vector(to_signed(-32768, 16));
124  r1(79 downto 64)    <= std_logic_vector(to_signed(10, 16));
125  r1(63 downto 48)    <= std_logic_vector(to_signed(10, 16));
126  r1(47 downto 32)    <= std_logic_vector(to_signed(10, 16));
127  r1(31 downto 16)    <= std_logic_vector(to_signed(10, 16));
128  r1(15 downto 0)     <= std_logic_vector(to_signed(10, 16));
129
130  --r2
131  r2(127 downto 112)  <= std_logic_vector(to_signed(0, 16));
132  r2(111 downto 96)   <= std_logic_vector(to_signed(10, 16));
133  r2(95 downto 80)    <= std_logic_vector(to_signed(32767, 16));
134  r2(79 downto 64)    <= std_logic_vector(to_signed(32767, 16));
135  r2(63 downto 48)    <= std_logic_vector(to_signed(-2, 16));
136  r2(47 downto 32)    <= std_logic_vector(to_signed(-2, 16));
137  r2(31 downto 16)    <= std_logic_vector(to_signed(2, 16));
138  r2(15 downto 0)     <= std_logic_vector(to_signed(2, 16));
139
140  --r3
141  r3(127 downto 112)  <= std_logic_vector(to_signed(-32768, 16));
142  r3(111 downto 96)   <= std_logic_vector(to_signed(-32768, 16));
143  r3(95 downto 80)    <= std_logic_vector(to_signed(32767, 16));
144  r3(79 downto 64)    <= std_logic_vector(to_signed(32767, 16));
145  r3(63 downto 48)    <= std_logic_vector(to_signed(20, 16));
146  r3(47 downto 32)    <= std_logic_vector(to_signed(20, 16));
147  r3(31 downto 16)    <= std_logic_vector(to_signed(20, 16));
148  r3(15 downto 0)     <= std_logic_vector(to_signed(20, 16));
149
150
151
152
153
154  --or
155  instrc <= "11000010110000000000000000";    --1858000
156  wait for clk;
```

```
157
158   --and
159   instrc <= "11000001010000000000000000";  -- 1828000
160   wait for clk;
161
162
163   --broadcast word
164   instrc <= "11000001100000000000000000"; -- 1830000
165   wait for clk;
166
167
168
169   ----max signed word
170   instrc <= "11000001110000000000000000"; --1838000
171   wait for clk;
172
173   ----min signed word
174   instrc <= "11000010000000000000000000";  --1840000
175   wait for clk;
176
177   --leading zeroes
178   instrc <= "11000000010000000000000000";   ---1800000
179   wait for clk;
180
181
182   --count 1s
183   instrc <= "11000011000000000000000000";   ---1860000
184   wait for clk;
185
186   --add word
187   instrc <= "11000000100000000000000000";   ---1810000
188   wait for clk;
189
190   --add half word
191   instrc <= "11000000110000000000000000";   ---1818000
192   wait for clk;
193
194   --add half word saturated
195   instrc <= "11000001000000000000000000";   ---1820000
196   wait for clk;
197
198   --sub half word
199   instrc <= "11000011100000000000000000";   ---1870000
200   wait for clk;
201
202   --sub half word saturated
203   instrc <= "11000011110000000000000000";   ---1878000
204   wait for clk;
205
206   --r1 values
207   r1(127 downto 96)   <= std_logic_vector(to_signed(2147483647, 32));

208   r1(95 downto 64)    <= std_logic_vector(to_signed(2147483647, 32));
209   r1(95) <= '1';
210
211   r2(111 downto 96)   <= std_logic_vector(to_signed(65535, 16));  --for
      underflow
```

```
212  r3(111 downto 96)   <= std_logic_vector(to_signed(65535, 16));  --for
     underflow
213  r3(111 downto 96)   <= std_logic_vector(to_signed(0, 16));  --for
     underflow
214  -- 64 bit signed   Integer Multiply-Add low with Saturation
215  instrc <= "10100000000000000000000000";
216  wait for clk;
217
218
219
220
221
222
223      std.env.finish;
224
225      end process;
226
227  end tb;
228
229
230
231
```