Subtype StateType is std\_logic\_vector(2 downto 0);

Signal CS, XS: StateType;

StateReg: Process(Clk)

Begin

If(Clk = '1' And Clk'Event) Then

If(Rst = '1') Then

CS <= "000" after 4 ns;

Else

CS <= XS after 4 ns;

end if;

end if;

end process;

CombLogic: Process(CS, Start)

Begin

W\_en <= '1';

R\_en <= '1';

OutReg\_Ld <= '0';

Case CS is

When "000" => --start

W\_en <= '0';

If (Start = '1') Then

XS <= "001" after 5 ns;

Else

XS <= "000" after 5 ns;

end if;

When "001" => --vt

XS <= "010" after 5 ns;

When "010" => --at

XS <= "011" after 5 ns;

When "011" => --at2

XS <= "100" after 5 ns;

When "100" => --at2/2

XS <= "101" after 5 ns;

When "101" => --sum1

XS <= "110" after 5 ns;

When "110" => --sum2

OutReg\_Ld <= '1';

XS <= "111" after 5 ns;

When "111" => --end

XS <= "000" after 5 ns;

end case;

W\_en <= '1';

R\_en <= '1';

--R\_Addr1

R\_Addr1(2) <= (CS(2) AND (NOT CS(0))) OR ((NOT CS(2)) AND CS(1) AND CS(0)) AFTER 18 NS;

R\_Addr1(1) <= (CS(2) XOR CS(0)) OR (CS(2) AND (NOT CS(1))) AFTER 18 NS;

R\_Addr1(0) <= (CS(1) AND (NOT CS(0))) OR (CS(2) AND (NOT CS(1)) AND CS(0)) AFTER 18 NS;

--R\_Addr2

R\_Addr2(2) <= (NOT CS(2) AND CS(0)) OR (CS(1) XOR CS(0)) AFTER 186 NS;

R\_Addr2(1) <= CS(2) AND CS(1) AND (NOT CS(0)) AFTER 18 NS;

R\_Addr2(0) <= CS(2) AND (NOT CS(1)) AND CS(0) AFTER 18 NS;

--W\_Addr

W\_Addr(2) <= ((NOT CS(2)) AND CS(0)) OR (CS(1) AND (NOT CS(0))) OR (CS(2) AND (NOT CS(1))) AFTER 18 NS;

W\_Addr(1) <= (CS(2) XOR CS(1)) OR (CS(1) AND (NOT CS(0))) AFTER 18 NS;

W\_Addr(0) <= ((NOT CS(1)) AND CS(0)) OR (CS(2) AND CS(1) AND (NOT CS(0))) AFTER 18 NS;

ALU\_sel <= ((not CS(2)) and CS(1)) or ((not CS(2)) and CS(0)) after 5.6 ns;

Shifter\_sel <= '0';

Selector\_sel <= ((not CS(2)) and CS(0)) or (CS(1) xor CS(0)) after 5.6 ns;

OutReg\_Ld <= CS(2) and CS(1) and (not CS(0)) after 5.6 ns;

Oe <= CS(2) and CS(1) and CS(0) after 5.6 ns;

Done <= CS(2) and CS(1) and CS(0) after 5.6 ns;