**Digital Design and Computer Organization Laboratory**

**UE19CS206**

**3rd Semester, Academic Year 2020-21**

Date: 28-09-2020

|  |  |  |
| --- | --- | --- |
| Name: Pranav R Hegde | SRN: PES1UG19CS343 | Section: F |

Experiment Number: 5 Week # :5

**Title of the Program: Register with ALU**

**Code:**

module reg16(input wire clk,reset,load,input wire [15:0]din,output wire [15:0]r);

dfrl my\_d0(clk,reset,load,din[0],r[0]);

dfrl my\_d1(clk,reset,load,din[1],r[1]);

dfrl my\_d2(clk,reset,load,din[2],r[2]);

dfrl my\_d3(clk,reset,load,din[3],r[3]);

dfrl my\_d4(clk,reset,load,din[4],r[4]);

dfrl my\_d5(clk,reset,load,din[5],r[5]);

dfrl my\_d6(clk,reset,load,din[6],r[6]);

dfrl my\_d7(clk,reset,load,din[7],r[7]);

dfrl my\_d8(clk,reset,load,din[8],r[8]);

dfrl my\_d9(clk,reset,load,din[9],r[9]);

dfrl my\_d10(clk,reset,load,din[10],r[10]);

dfrl my\_d11(clk,reset,load,din[11],r[11]);

dfrl my\_d12(clk,reset,load,din[12],r[12]);

dfrl my\_d13(clk,reset,load,din[13],r[13]);

dfrl my\_d14(clk,reset,load,din[14],r[14]);

dfrl my\_d15(clk,reset,load,din[15],r[15]);

endmodule

module mux128\_16(input wire [15:0]i0,i1,i2,i3,i4,i5,i6,i7,input wire s0,s1,s2,output wire [15:0]o);

mux8 my\_mux0({i0[0],i1[0],i2[0],i3[0],i4[0],i5[0],i6[0],i7[0]},s2,s1,s0,o[0]);

mux8 my\_mux1({i0[1],i1[1],i2[1],i3[1],i4[1],i5[1],i6[1],i7[1]},s2,s1,s0,o[1]);

mux8 my\_mux2({i0[2],i1[2],i2[2],i3[2],i4[2],i5[2],i6[2],i7[2]},s2,s1,s0,o[2]);

mux8 my\_mux3({i0[3],i1[3],i2[3],i3[3],i4[3],i5[3],i6[3],i7[3]},s2,s1,s0,o[3]);

mux8 my\_mux4({i0[4],i1[4],i2[4],i3[4],i4[4],i5[4],i6[4],i7[4]},s2,s1,s0,o[4]);

mux8 my\_mux5({i0[5],i1[5],i2[5],i3[5],i4[5],i5[5],i6[5],i7[5]},s2,s1,s0,o[5]);

mux8 my\_mux6({i0[6],i1[6],i2[6],i3[6],i4[6],i5[6],i6[6],i7[6]},s2,s1,s0,o[6]);

mux8 my\_mux7({i0[7],i1[7],i2[7],i3[7],i4[7],i5[7],i6[7],i7[7]},s2,s1,s0,o[7]);

mux8 my\_mux8({i0[8],i1[8],i2[8],i3[8],i4[8],i5[8],i6[8],i7[8]},s2,s1,s0,o[8]);

mux8 my\_mux9({i0[9],i1[9],i2[9],i3[9],i4[9],i5[9],i6[9],i7[9]},s2,s1,s0,o[9]);

mux8 my\_mux10({i0[10],i1[10],i2[10],i3[10],i4[10],i5[10],i6[10],i7[10]},s2,s1,s0,o[10]);

mux8 my\_mux11({i0[11],i1[11],i2[11],i3[11],i4[11],i5[11],i6[11],i7[11]},s2,s1,s0,o[11]);

mux8 my\_mux12({i0[12],i1[12],i2[12],i3[12],i4[12],i5[12],i6[12],i7[12]},s2,s1,s0,o[12]);

mux8 my\_mux13({i0[13],i1[13],i2[13],i3[13],i4[13],i5[13],i6[13],i7[13]},s2,s1,s0,o[13]);

mux8 my\_mux14({i0[14],i1[14],i2[14],i3[14],i4[14],i5[14],i6[14],i7[14]},s2,s1,s0,o[14]);

mux8 my\_mux15({i0[15],i1[15],i2[15],i3[15],i4[15],i5[15],i6[15],i7[15]},s2,s1,s0,o[15]);

endmodule

module reg\_file (input wire clk, reset, wr, input wire [2:0] rd\_addr\_a, rd\_addr\_b, wr\_addr, input wire [15:0] d\_in, output wire [15:0] d\_out\_a, d\_out\_b);

wire [0:7]load;

wire [0:15]r0,r1,r2,r3,r4,r5,r6,r7;

demux8 dmx(wr,wr\_addr[2],wr\_addr[1],wr\_addr[0],load);

reg16 my\_reg0(clk,reset,load[0],d\_in,r0);

reg16 my\_reg1(clk,reset,load[1],d\_in,r1);

reg16 my\_reg2(clk,reset,load[2],d\_in,r2);

reg16 reg3(clk,reset,load[3],d\_in,r3);

reg16 reg4(clk,reset,load[4],d\_in,r4);

reg16 reg5(clk,reset,load[5],d\_in,r5);

reg16 reg6(clk,reset,load[6],d\_in,r6);

reg16 reg7(clk,reset,load[7],d\_in,r7);

mux128\_16 mm0(r0,r1,r2,r3,r4,r5,r6,r7,rd\_addr\_a[0],rd\_addr\_a[1],rd\_addr\_a[2],d\_out\_a);

mux128\_16 mm1(r0,r1,r2,r3,r4,r5,r6,r7,rd\_addr\_b[0],rd\_addr\_b[1],rd\_addr\_b[2],d\_out\_b);

endmodule

**Output waveform**

