

Digital Design and Computer Organization Laboratory

UE19CS206

3rd Semester, Academic Year 2020-21

Date:

Name : Adithya M	SRN : PES1UG20CS621	Section : K
------------------	---------------------	----------------

Experiment Number: 8

Week # : 8

Title of the Program: Load and Jump Instructions

Code:

```
module nor5 (input wire [0:4] i, output wire o);  
  
    wire t;  
  
    or3 or3_0 (i[0], i[1], i[2], t);  
  
    nor3 nor3_0 (t, i[3], i[4], o);  
  
endmodule
```

```
module ir (input wire clk, reset, load, input wire [15:0] din, output wire [15:0] dout);  
  
    dfri dfri_0 (clk, reset, load, din['h0], dout['h0]);  
  
    dfri dfri_1 (clk, reset, load, din['h1], dout['h1]);  
  
    dfri dfri_2 (clk, reset, load, din['h2], dout['h2]);  
  
    dfri dfri_3 (clk, reset, load, din['h3], dout['h3]);  
  
    dfri dfri_4 (clk, reset, load, din['h4], dout['h4]);  
  
    dfri dfri_5 (clk, reset, load, din['h5], dout['h5]);  
  
    dfri dfri_6 (clk, reset, load, din['h6], dout['h6]);  
  
    dfri dfri_7 (clk, reset, load, din['h7], dout['h7]);
```

```

dfri dfri_8 (clk, reset, load, din['h8], dout['h8]);
dfri dfri_9 (clk, reset, load, din['h9], dout['h9]);
dfri dfri_a (clk, reset, load, din['ha], dout['ha]);
dfri dfri_b (clk, reset, load, din['hb], dout['hb]);
dfri dfri_c (clk, reset, load, din['hc], dout['hc]);
dfri dfri_d (clk, reset, load, din['hd], dout['hd]);
dfri dfri_e (clk, reset, load, din['he], dout['he]);
dfri dfri_f (clk, reset, load, din['hf], dout['hf]);

endmodule

```

```

module control_logic (input wire clk, reset, input wire [15:0] cur_ins, output wire [2:0] rd_addr_a, rd_addr_b, wr_addr,
output wire [1:0] op, output wire sel, jump, pc_inc, load_ir, wr_reg);

```

```

    wire u,w,s,wr_reg1,wr_reg2,alu_ins,ld_ins,ld_ins_,fi,fo,el,eo,ef;

```

```

    assign rd_addr_a[0] = cur_ins[0];

```

```

    assign rd_addr_a[1] = cur_ins[1];

```

```

    assign rd_addr_a[2] = cur_ins[2];

```

```

    assign rd_addr_b[0] = cur_ins[3];

```

```

    assign rd_addr_b[1] = cur_ins[4];

```

```

    assign rd_addr_b[2] = cur_ins[5];

```

```

    assign wr_addr[0] = cur_ins[6];

```

```

    assign wr_addr[1] = cur_ins[7];

```

```

    assign wr_addr[2] = cur_ins[8];

```

```

    assign op[0] = cur_ins[9];

```

```

    assign op[1] = cur_ins[10];

```

```

    invert i1(ld_ins,ld_ins_);

```

```

    nor5 n1({cur_ins[15],cur_ins[14],cur_ins[13],cur_ins[12],cur_ins[11]},alu_ins);

```

```
invert i2(cur_ins[14],s);
```

```
and2 a1(cur_ins[15],s,ld_ins);
```

```
invert i3(cur_ins[10],w);
```

```
invert i4(cur_ins[15],u);
```

```
and3 a2(cur_ins[14],u,ef,jump);
```

```
dfri d1(clk,reset,1'b1,fo,eo);
```

```
and2 a3(ld_ins_,eo,ef);
```

```
and2 a4(ef,alu_ins,wr_reg1);
```

```
or2 o3(wr_reg1,wr_reg2,wr_reg);
```

```
and2 a5(eo,ld_ins,el);
```

```
and2 a6(ld_ins,el,wr_reg2);
```

```
nand2 n2(el,ld_ins,sel);
```

```
dfri d2(clk,reset,1'b1,el,lo);
```

```
or2 o1(lo,ef,fi);
```

```
dfsl d3(clk,reset,1'b1,fi,fo);
```

```
assign load_ir = fo;
```

```
or2 o2(load_ir,el,pc_inc);
```

```
endmodule
```

```
module mproc (input wire clk, reset, input wire [15:0] d_in, output wire [6:0] addr, output wire [15:0] d_out);
```

```
  wire pc_inc, cout, cout_, sub, sel, sel_addr; wire [2:0] rd_addr_a, rd_addr_b, wr_addr; wire [1:0] op; wire [8:0] _addr;
```

```
  wire [15:0] cur_ins, d_out_a, d_out_b;
```

and2 and2_0 (jump, cout, sub);

pc pc_0 (clk, reset, pc_inc, 1'b0, sub, {8'b0, cur_ins[7:0]}, {_addr, addr});

ir ir_0 (clk, reset, load_ir, d_in, cur_ins);

control_logic control_logic_0 (clk, reset, cur_ins, rd_addr_a, rd_addr_b, wr_addr, op, sel, jump, pc_inc, load_ir, wr_reg);

reg_alu reg_alu_0 (clk, reset, sel, wr_reg, op, rd_addr_a, rd_addr_b, wr_addr, d_in, d_out_a, d_out_b, cout);

assign d_out = d_out_a;

endmodule

Output waveform

