

Name: Adithya M SRN: PES1UG20CS621 SEC: K

mproc.v

```
module ir (input wire clk, reset, load, input wire [15:0] din, output wire [15:0] dout);
    dfr1 dfrl_0 (clk, reset, load, din['h0], dout['h0]);
    dfr1 dfrl_1 (clk, reset, load, din['h1], dout['h1]);
    dfr1 dfrl_2 (clk, reset, load, din['h2], dout['h2]);
    dfr1 dfrl_3 (clk, reset, load, din['h3], dout['h3]);
    dfr1 dfrl_4 (clk, reset, load, din['h4], dout['h4]);
    dfr1 dfrl_5 (clk, reset, load, din['h5], dout['h5]);
    dfr1 dfrl_6 (clk, reset, load, din['h6], dout['h6]);
    dfr1 dfrl_7 (clk, reset, load, din['h7], dout['h7]);
    dfr1 dfrl_8 (clk, reset, load, din['h8], dout['h8]);
    dfr1 dfrl_9 (clk, reset, load, din['h9], dout['h9]);
    dfr1 dfrl_a (clk, reset, load, din['ha], dout['ha]);
    dfr1 dfrl_b (clk, reset, load, din['hb], dout['hb]);
    dfr1 dfrl_c (clk, reset, load, din['hc], dout['hc]);
    dfr1 dfrl_d (clk, reset, load, din['hd], dout['hd]);
    dfr1 dfrl_e (clk, reset, load, din['he], dout['he]);
    dfr1 dfrl_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 pc_inc, load_ir, wr_reg);

    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];

    wire t1,t2,t3;

    or3 o1(cur_ins[11],cur_ins[12],cur_ins[13],t1);
    or3 o2(cur_ins[14],cur_ins[15],t1,t2);
    invert o3(t2,t3);
    dfr1 g1(clk,reset,1'b1,pc_inc,load_ir);
    dfr1 g2(clk,reset,1'b1,load_ir,pc_inc);
    and2 o4(pc_inc, t3,wr_reg);

endmodule

module mproc (input wire clk, reset, input wire [15:0] ins, output wire [15:0] addr);
    wire pc_inc, cout; wire [2:0] rd_addr_a, rd_addr_b, wr_addr; wire [1:0] op; wire [15:0] cur_ins, d_out_a, d_out_b;

    pc pc_0 (clk, reset, pc_inc, 1'b0, 1'b0, 16'b0, addr);
    ir ir_0 (clk, reset, load_ir, ins, cur_ins);
    control_logic control_logic_0 (clk, reset, cur_ins, rd_addr_a, rd_addr_b, wr_addr, op, pc_inc, load_ir, wr_reg);
    reg_alu reg_alu_0 (clk, reset, 1'b1, wr_reg, op, rd_addr_a, rd_addr_b, wr_addr, 16'b0, d_out_a, d_out_b, cout);
endmodule
```

mproc_mem.v

```
module ram_128_16 (input wire clk, reset, wr, input wire [6:0] addr, input wire [15:0] din, output wire [15:0] dout);
    reg [0:127] ram [15:0];

    initial begin
        ram[0]=16'o000100;
        ram[1]=16'o001201;
        ram[2]=16'o002321;
        ram[3]=16'o003432;
    end
    always @(wr) ram[addr]=din;
    assign dout=ram[addr];
endmodule

module mproc_mem (input wire clk, reset);
    wire [15:0] addr; wire [15:0] ins;

    ram_128_16 ram_128_16_0 (clk, reset, 1'b0, addr[6:0], 16'b0, ins);
    mproc mproc_0 (clk, reset, ins, addr);
endmodule
```

Output:

