Digital Design and Computer Organization

Laboratory UE19CS206

3rd Semester, Academic Year 2020-21

Date:01 /11 /2020

NAME : SUHAN B REVANKAR	SRN : PES2UG19CS412	Section : G

Experiment Number: 7
Week #: 7

Title of the Program:

16 Bit Microprocessor Control Logic

Aim of the Program:

To Design and Implement the Microprocessor Control logic, which is essentially a finite state machine

Code (mproc.v)

mproc - Notepad

File Edit Format View Help

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

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);
    //assignment 3 logic here
    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);
    dfsl g1(clk,reset,1'b1,pc_inc,load_ir);
    dfrl 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
```

Ln 1, Col 1

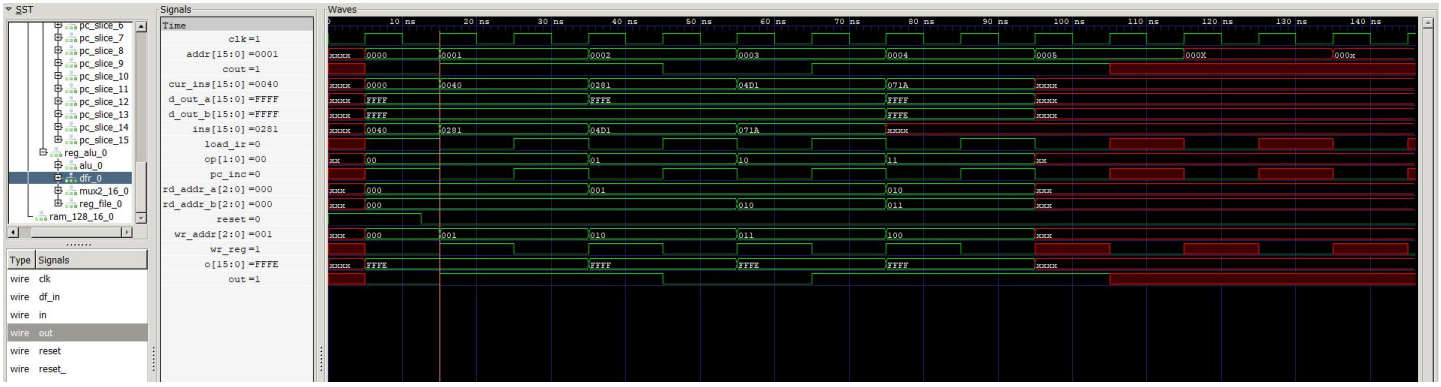
TABLE

cur_ins [15:0]	ram	Octal Value Of cur_ins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0040 (Hex)	ram [0]	16‘o 000100;	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	d_out_a+ d_out_b= FFFF+FFFF =FFFE with carry =1
0281 (Hex)	ram [1]	16'o 001100;	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	d_out_a- d_out_b= FFFE-FFFF =FFFF with borrow=1
04D1 (Hex)	ram [2]	16'o 002100	0	0	0	0	0	1	0	0	1	1	0	1	0	0	0	1	d_out_a AND d_out_b= FFFE AND FFFF=FFFE
071A (Hex)	ram [3]	16'o 003100;	0	0	0	0	0	1	1	1	0	0	0	1	1	0	1	0	d_out_a OR d_out_b= FFFF OR FFFE=FFFF

Output waveform

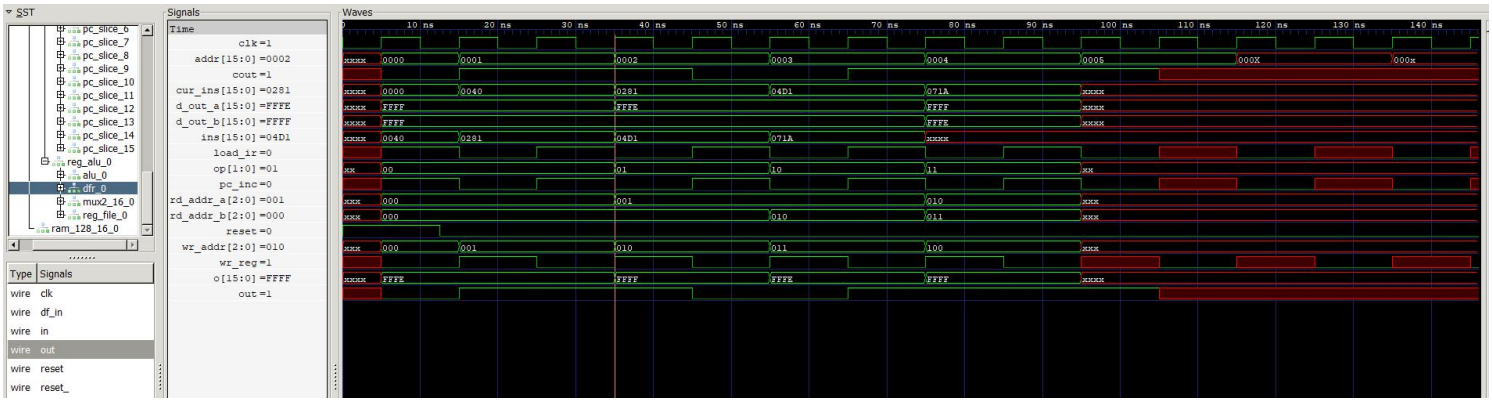
SCREENSHOT1(ADD Instruction)

cur_ins [15:0]	ram	Octal Value Of cur_ins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0040 (Hex)	ram [0]	16'o 000100;	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	d_out_a+ d_out_b= FFFF+FFFF =FFFE with carry =1



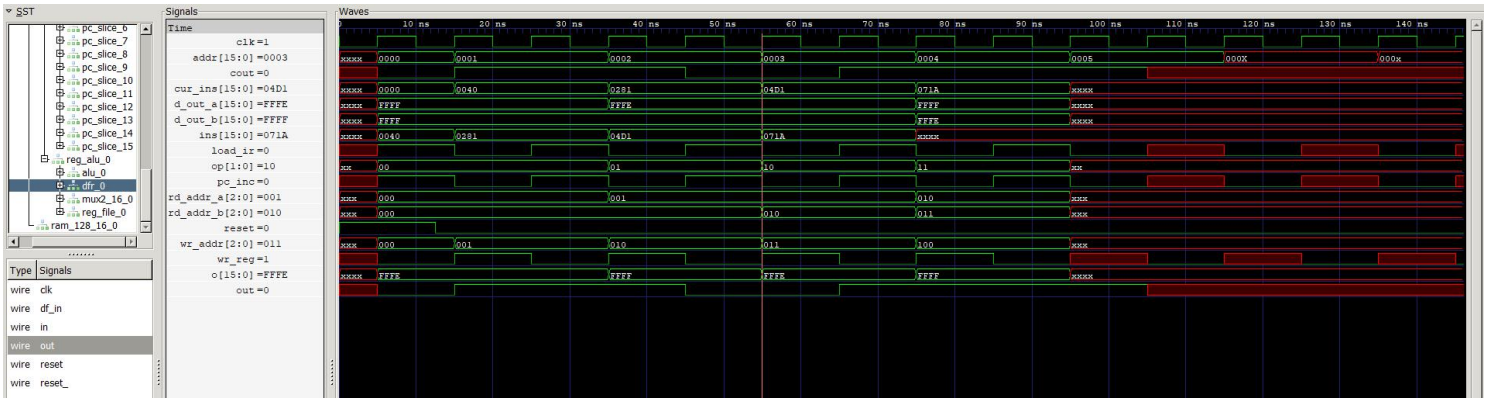
SCREENSHOT 02 (SUBTRACT Instruction)

cur_ins [15:0]	ram	Octal Value Of cur_ins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0281 (Hex)	ram [1]	16'o 001100;	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	d_out_a- d_out_b= FFFE-FFFF =FFFF with borrow=1



SCREENSHOT 3(AND Instruction)

cur_ins [15:0]	ram	Octal Value Of cur_ins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
04D1 (Hex)	ram [2]	16'o 002100	0	0	0	0	0	1	0	0	1	1	0	1	0	0	0	1	d_out_a AND d_out_b= FFFE AND FFFF=FFFE



SCREENSHOT 4 (OR Instruction)

cur_ins [15:0]	ram	Octal Value of cur_ins	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
071A (Hex)	ram [3]	16'o 003100;	0	0	0	0	0	1	1	1	0	0	0	1	1	0	1	0	d_out_a OR d_out_b= FFFF OR FFFE=FFFF