Ty Madsen

ECEN 220

12 December 2013

Lab 12 - LC3 FULL

LC3 Controller Verilog:

module LC3Controller(

input [15:0] IR,

input N,

input Z,

input P,

input reset,

input clk,

output reg [1:0] aluControl,

output reg enaALU,

output reg [2:0] SR1,

output reg [2:0] SR2,

output reg [2:0] DR,

output reg regWE,

output reg [1:0] selPC,

output reg enaMARM,

output reg selMAR,

output reg selEAB1,

output reg [1:0] selEAB2,

output reg enaPC,

output reg ldPC,

output reg ldIR,

output reg ldMAR,

output reg ldMDR,

output reg selMDR,

output reg memWE,

output reg flagWE,

output reg enaMDR

);

parameter Fetch0 = 15'b000000000000001;

parameter Fetch1 = 15'b000000000000010;

parameter Fetch2 = 15'b000000000000100;

parameter Decode = 15'b000000000001000;

parameter ALU = 15'b000000000010000;

parameter JSR0 = 15'b000000000100000;

parameter JSR1 = 15'b000000001000000;

parameter BR = 15'b000000010000000;

parameter LD0 = 15'b000000100000000;

parameter LD1 = 15'b000001000000000;

parameter LD2 = 15'b000010000000000;

parameter ST0 = 15'b000100000000000;

parameter ST1 = 15'b001000000000000;

parameter ST2 = 15'b010000000000000;

parameter JMPRET = 15'b100000000000000;

wire [14:0] CS;

reg [14:0] NS;

register\_15 load\_state(CS, clk, NS, reset, 1'b1);

always @(\*)

begin

//SET DEFAULTS FOR ALL SIGNALS ASSIGNED HERE

enaALU = 1'b0;

enaMARM = 1'b0;

enaPC = 1'b0;

enaMDR = 1'b0;

ldIR = 1'b0;

ldPC = 1'b0;

ldMAR = 1'b0;

ldMDR = 1'b0;

regWE = 1'b0;

memWE = 1'b0;

flagWE = 1'b0;

case(CS)

Fetch0:

begin

enaPC = 1'b1;

ldMAR = 1'b1;

NS = Fetch1;

end

Fetch1:

begin

selPC = 2'b00;

ldPC = 1'b1;

ldMDR = 1'b1;

selMDR = 1'b1;

NS = Fetch2;

end

Fetch2:

begin

ldIR = 1'b1;

enaMDR = 1'b1;

NS = Decode;

end

Decode:

begin

NS = (IR[15:12] == 4'b0001) ? ALU :

(IR[15:12] == 4'b0101) ? ALU :

(IR[15:12] == 4'b1001) ? ALU :

(IR[15:12] == 4'b0000) ? BR :

(IR[15:12] == 4'b1100) ? JMPRET :

(IR[15:11] == 5'b01001) ? JSR0 :

(IR[15:12] == 4'b0010) ? LD0 :

(IR[15:12] == 4'b0011) ? ST0 :

Fetch0;

end

ALU:

begin

aluControl = (IR[15:12] == 4'b0001) ? 2'b01 :

(IR[15:12] == 4'b0101) ? 2'b10 :

2'b11;

SR1 = IR[8:6];

SR2 = IR[2:0];

DR = IR[11:9];

regWE = 1'b1;

enaALU = 1'b1;

flagWE = 1'b1;

NS = Fetch0;

end

JSR0:

begin

DR = 3'b111;

regWE = 1'b1;

enaPC = 1'b1;

NS = JSR1;

end

JSR1:

begin

selPC = 2'b01;

selEAB1 = 1'b0;

selEAB2 = 2'b11;

ldPC = 1'b1;

NS = Fetch0;

end

BR:

begin

selPC = 2'b01;

selEAB1 = 1'b0;

selEAB2 = 2'b10;

ldPC = (N & IR[11]) | (Z & IR[10]) | (P & IR[9]);

NS = Fetch0;

end

LD0:

begin

selEAB1 = 1'b0;

selEAB2 = 2'b10;

enaMARM = 1'b1;

selMAR = 1'b0;

ldMAR = 1'b1;

NS = LD1;

end

LD1:

begin

ldMDR = 1'b1;

selMDR = 1'b1;

NS = LD2;

end

LD2:

begin

DR = IR[11:9];

regWE = 1'b1;

enaMDR = 1'b1;

flagWE = 1'b1;

NS = Fetch0;

end

ST0:

begin

selEAB1 = 1'b0;

selEAB2 = 2'b10;

enaMARM = 1'b1;

ldMAR = 1'b1;

selMAR = 1'b0;

NS = ST1;

end

ST1:

begin

aluControl = 2'b00;

SR1 = IR[11:9];

enaALU = 1'b1;

ldMDR = 1'b1;

selMDR = 1'b0;

NS = ST2;

end

ST2:

begin

memWE = 1'b1;

NS = Fetch0;

end

JMPRET:

begin

SR1 = IR[8:6];

selPC = 2'b01;

selEAB1 = 1'b1;

selEAB2 = 2'b00;

ldPC = 1'b1;

NS = Fetch0;

end

endcase

//NS = (reset == 1'b1) ? Fetch0 : NS;

end

endmodule

FULL LC3 Verilog:

module FullLC3(

input clk,

input reset

);

wire N, Z, P, enaALU, regWE, enaMARM, selMAR, selEAB1, enaPC, ldPC, ldIR, ldMDR, selMDR, memWE, flagWE, enaMDR;

wire [1:0] aluControl, selPC, selEAB2;

wire [2:0] SR1, SR2, DR;

wire [15:0] IR;

LC3Controller control(IR, N, Z, P, reset, clk, aluControl, enaALU, SR1,

SR2, DR, regWE, selPC, enaMARM, selMAR, selEAB1, selEAB2, enaPC,

ldPC, ldIR, ldMAR, ldMDR,selMDR, memWE, flagWE, enaMDR);

Datapath path(clk, reset, aluControl, enaALU, SR1,

SR2, DR, regWE, selPC, enaMARM, selMAR, selEAB1, selEAB2, enaPC,

ldPC, ldIR, ldMAR, ldMDR, selMDR, memWE, flagWE, enaMDR, N, Z, P, IR);

endmodule

Full TCL:

wave add / -radix hex

wave add /control/CS -radix hex

wave add /control/NS -radix hex

wave add /path/REG/R0 -radix hex

wave add /path/REG/R1 -radix hex

wave add /path/REG/R2 -radix hex

wave add /path/REG/R3 -radix hex

wave add /path/REG/R4 -radix hex

wave add /path/REG/R5 -radix hex

wave add /path/REG/R6 -radix hex

wave add /path/REG/R7 -radix hex

wave add /path/nzp -radix hex

wave add /path/pcModule -radix hex

wave add /path/marmodule -radix hex

wave add /path/mem/MDR0 -radix hex

isim force add reset 1

isim force add clk 0 -time 0 -value 1 -time 5ns -repeat 10ns

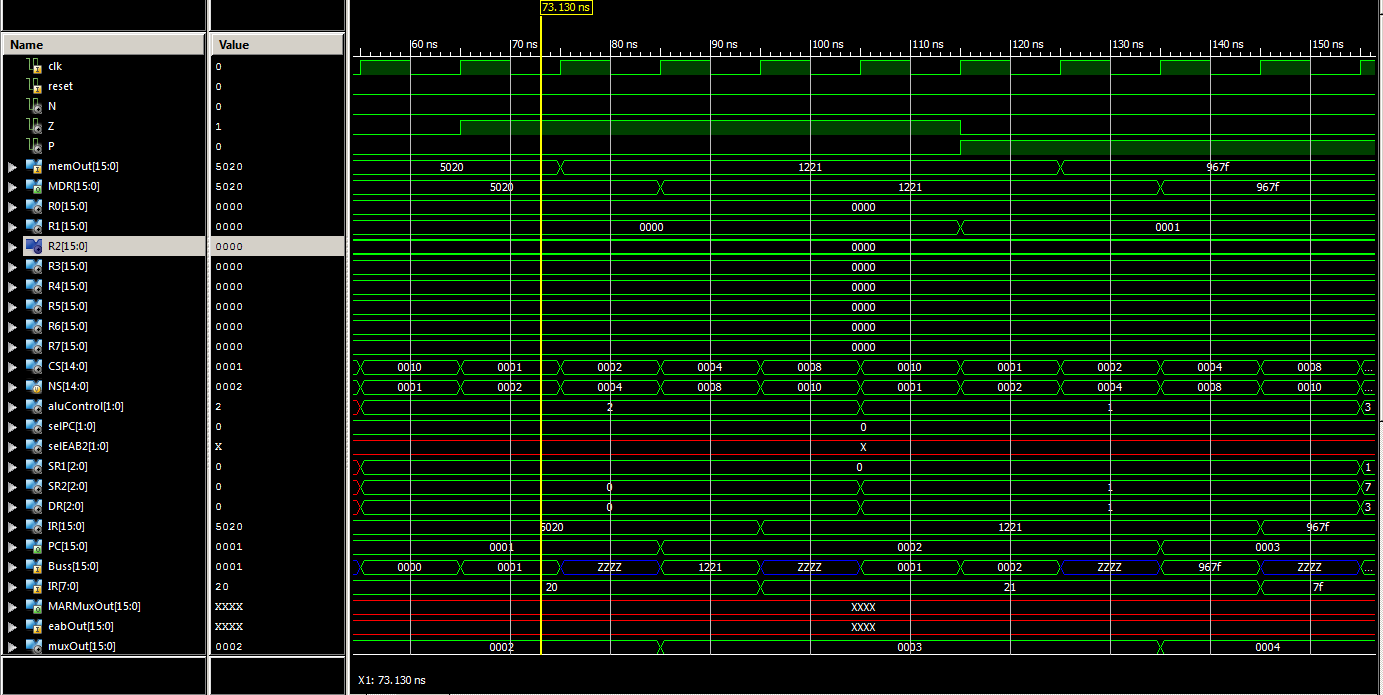
run 23ns

isim force add reset 0

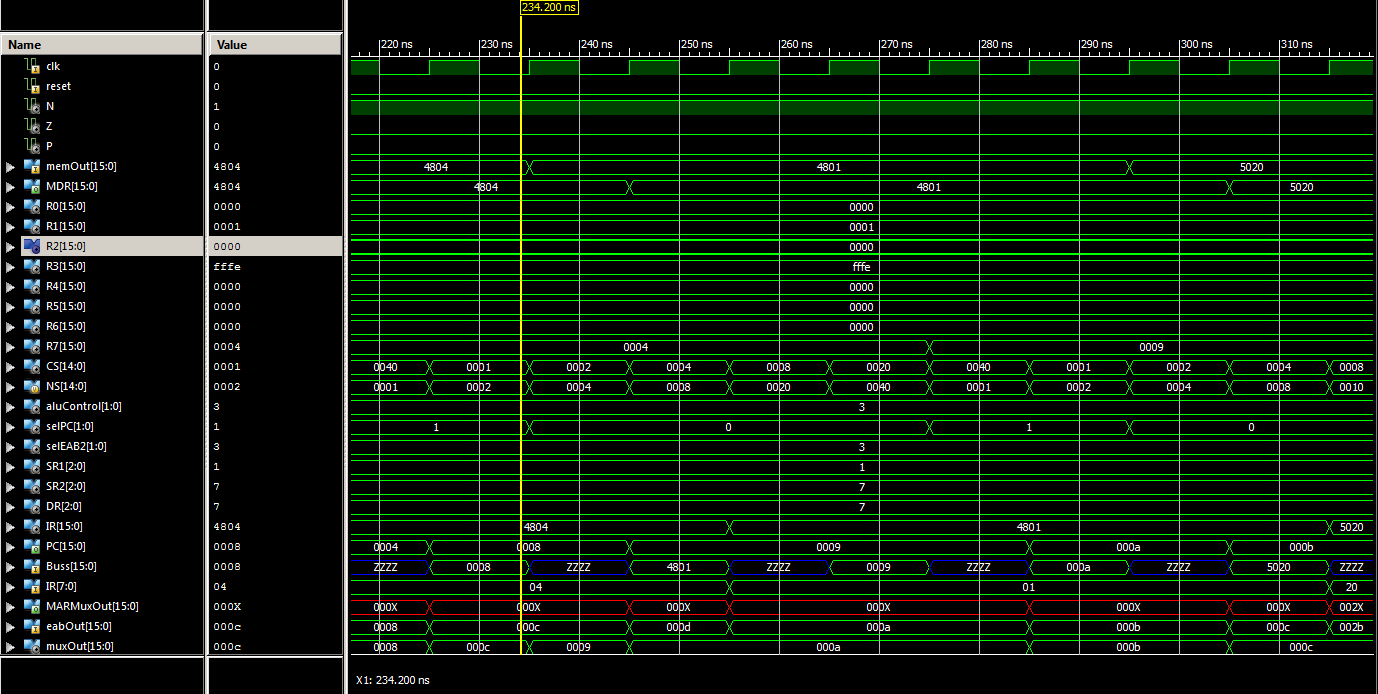
run 1680ns

WaveForms:

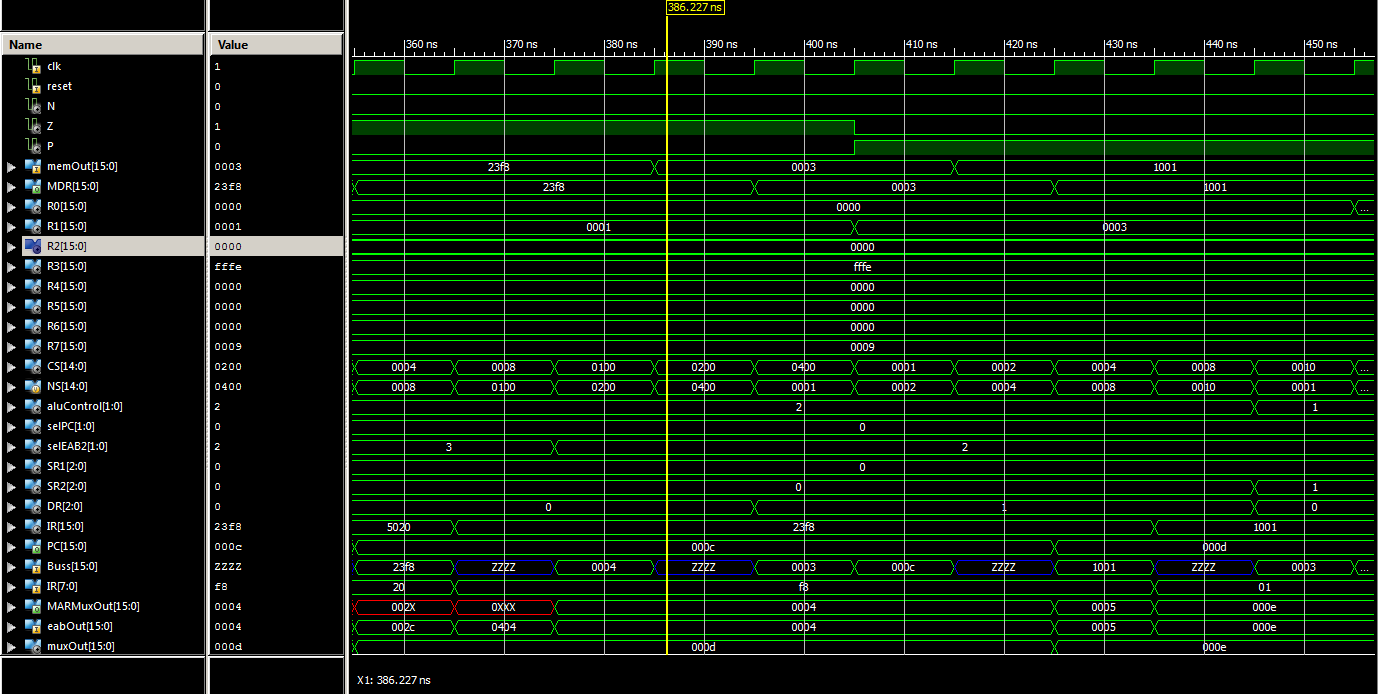
First 3



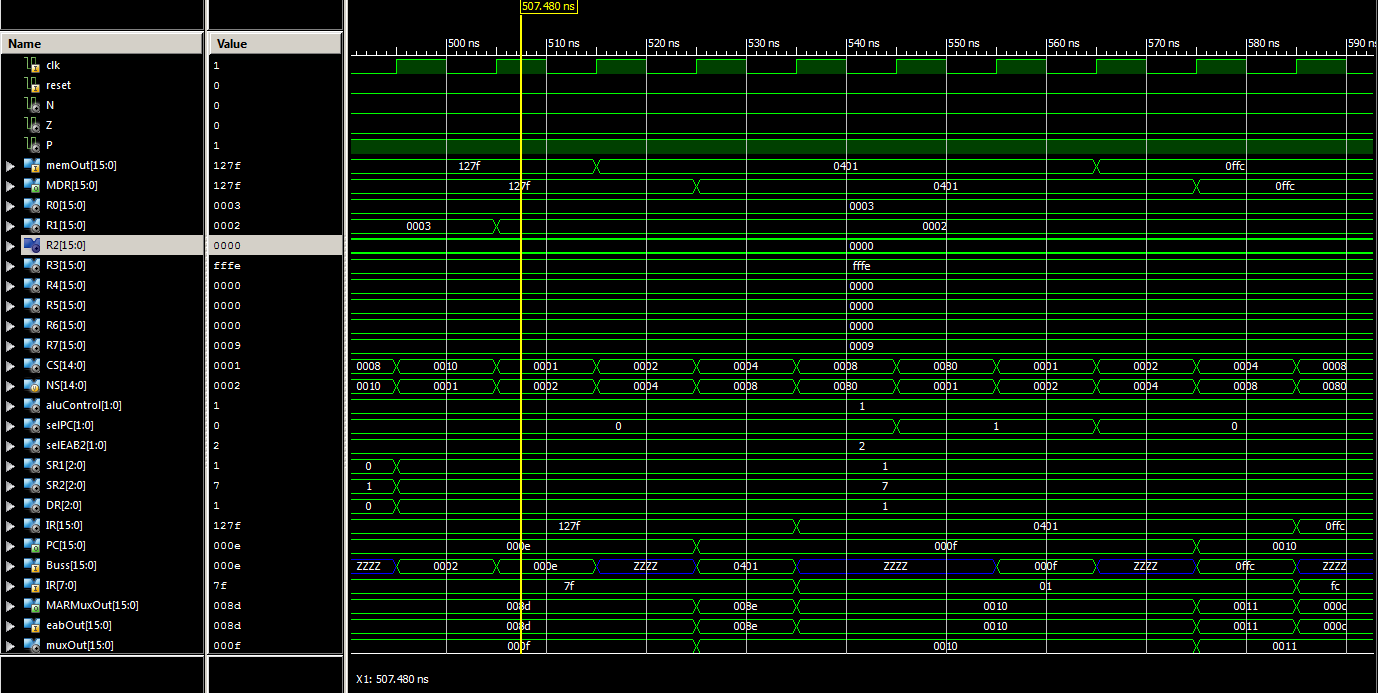
Second 3



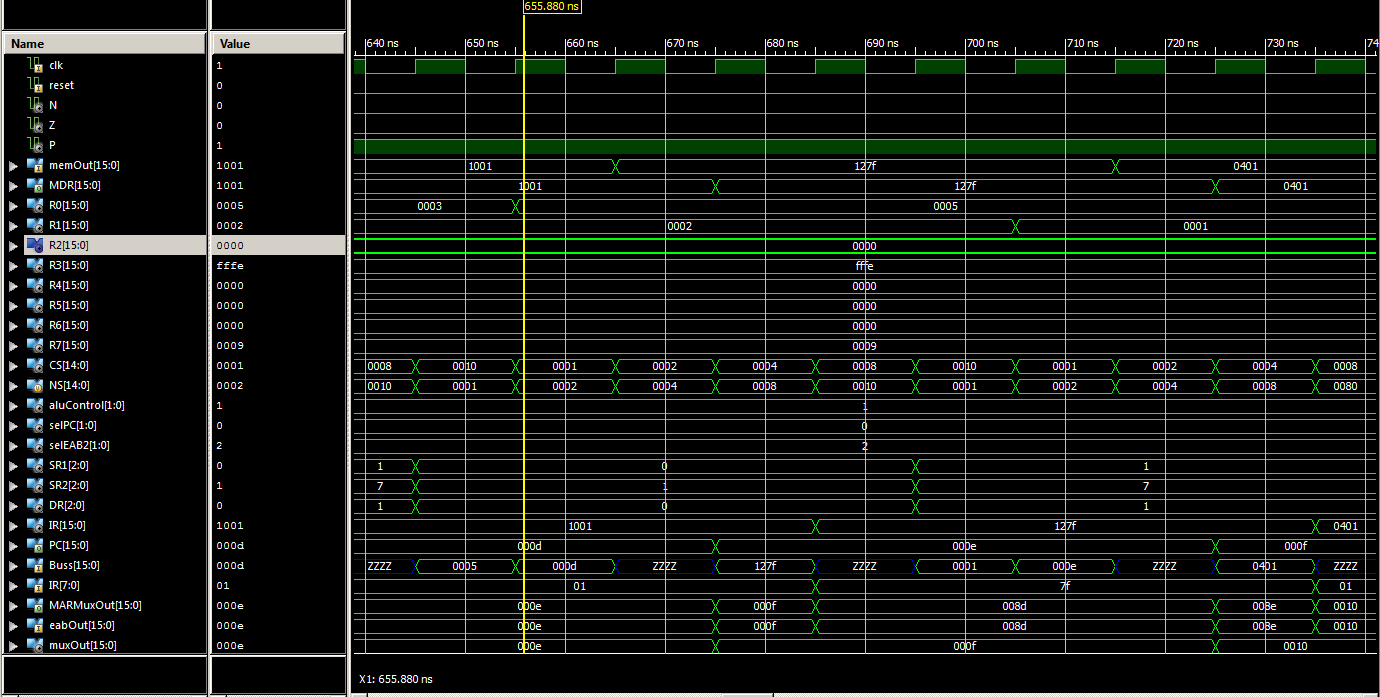
Third 3



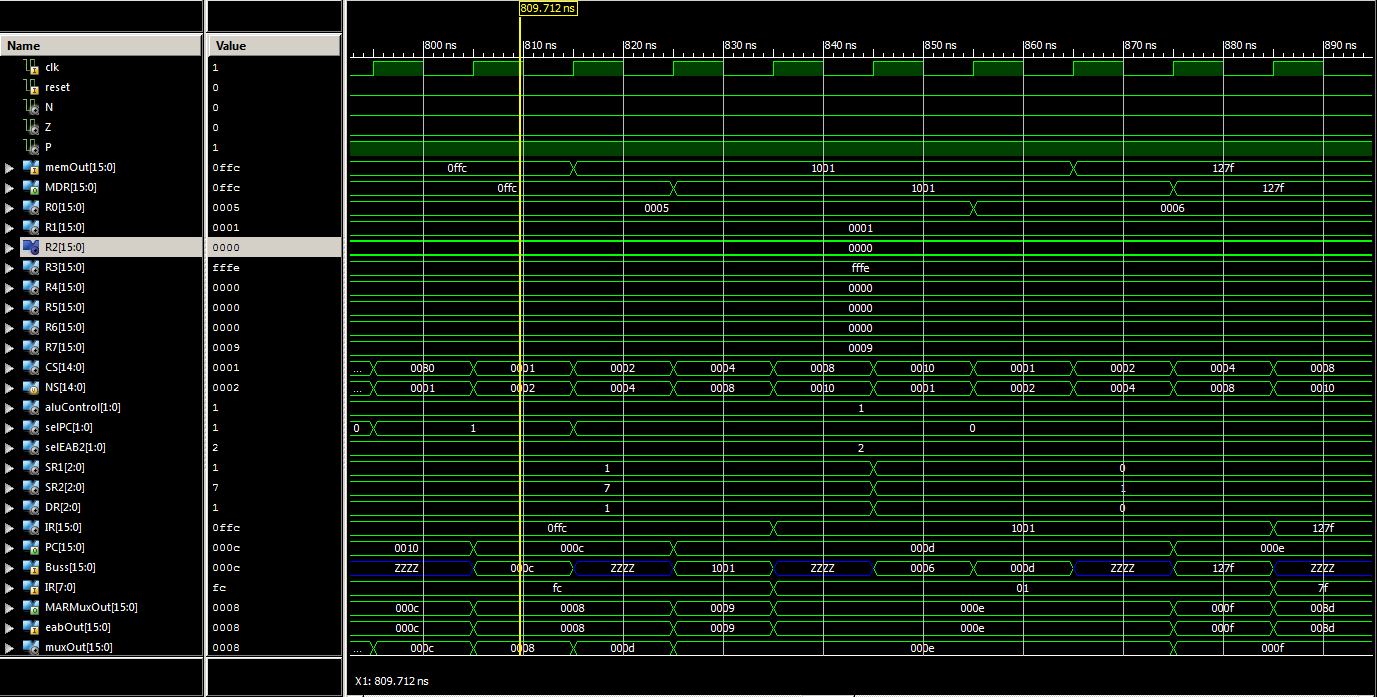
Fourth 3



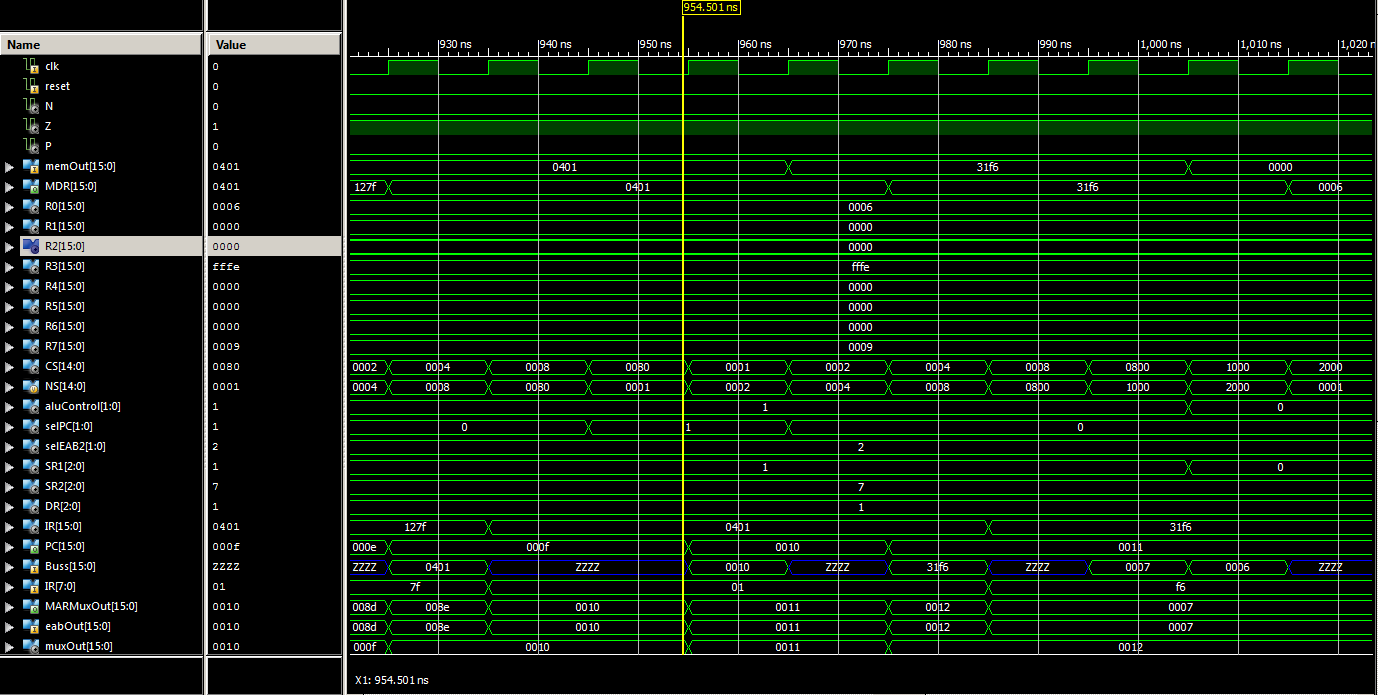
Fifth 3



Sixth 3



Seventh 3



Last 3

