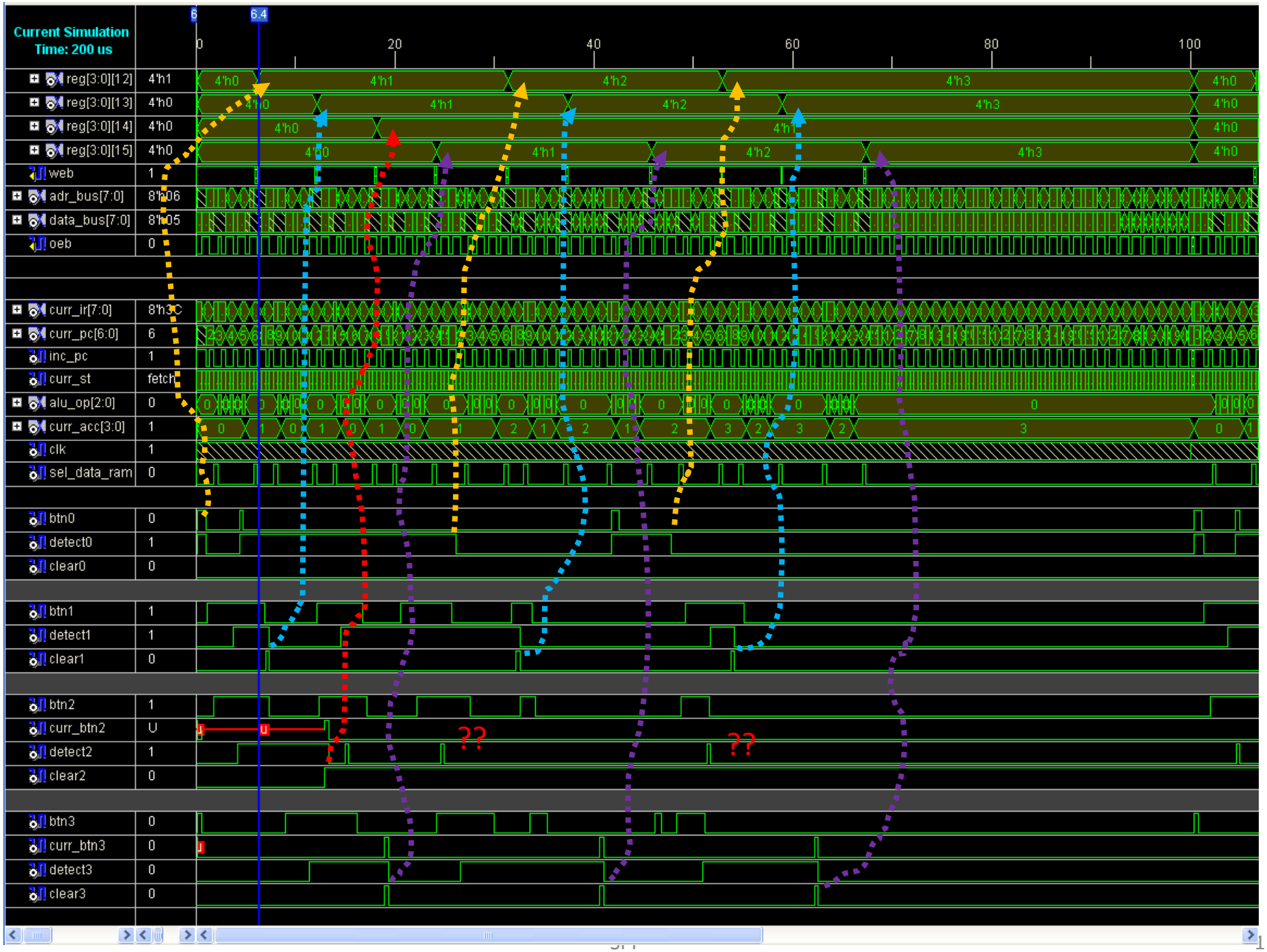
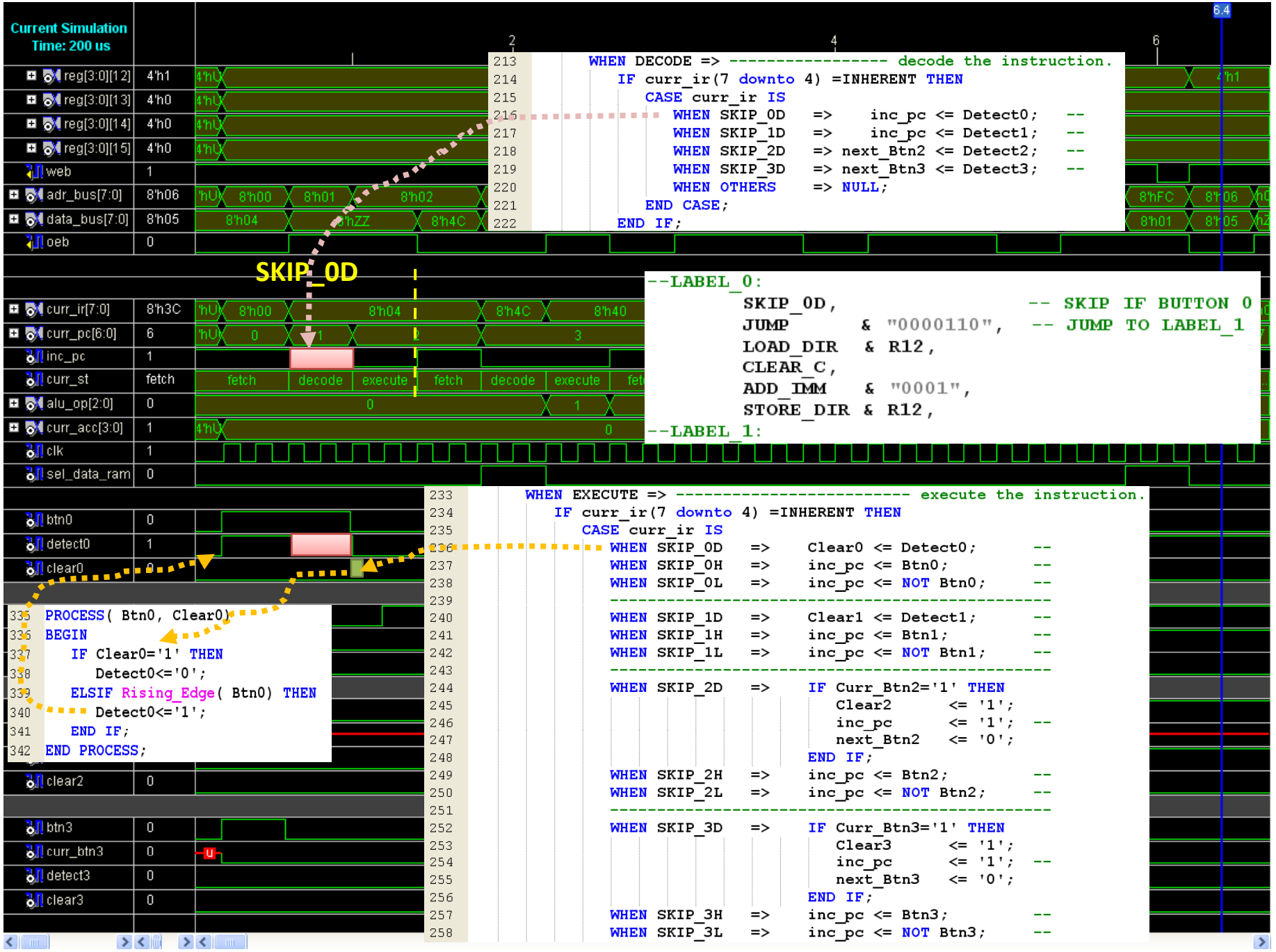
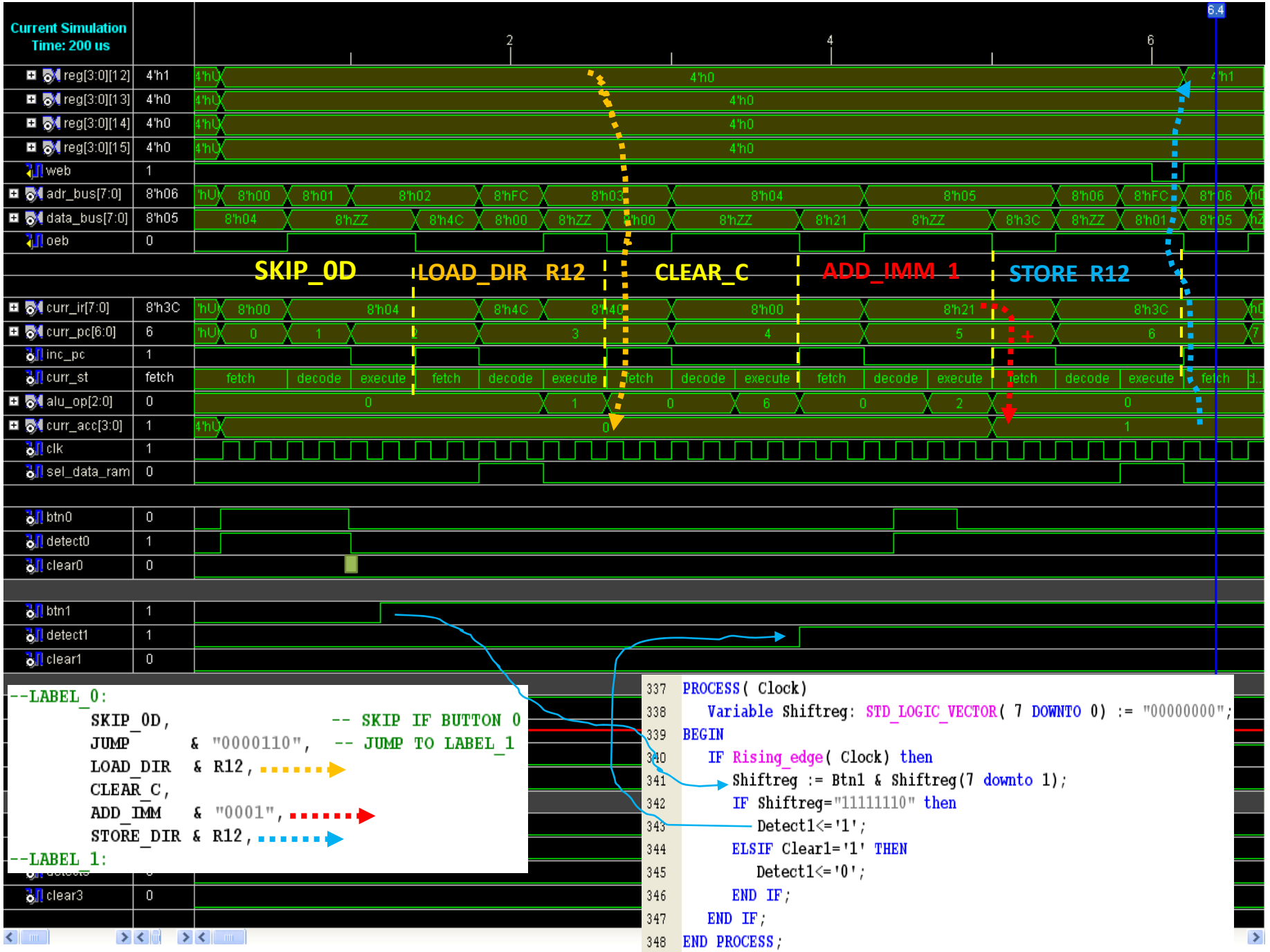
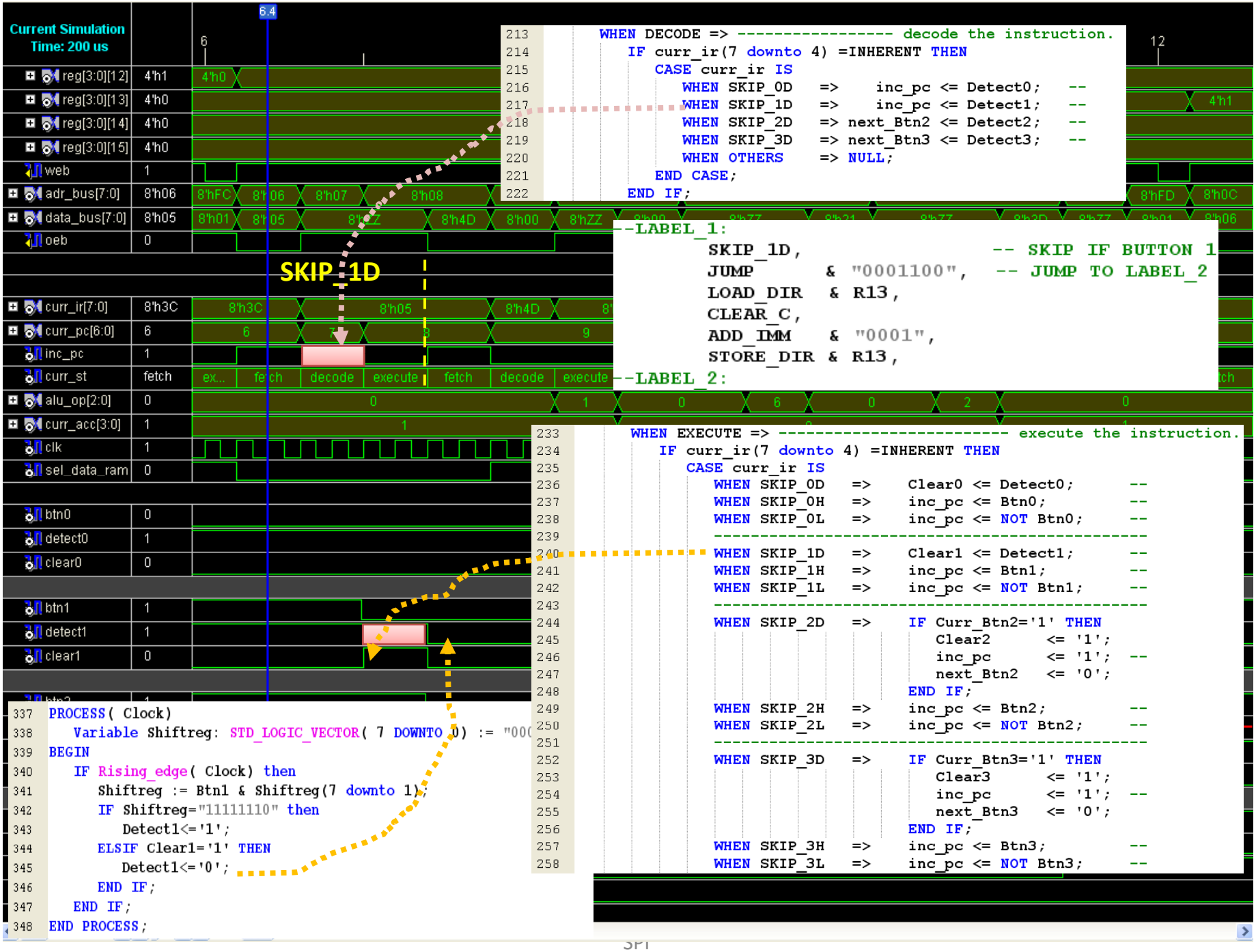
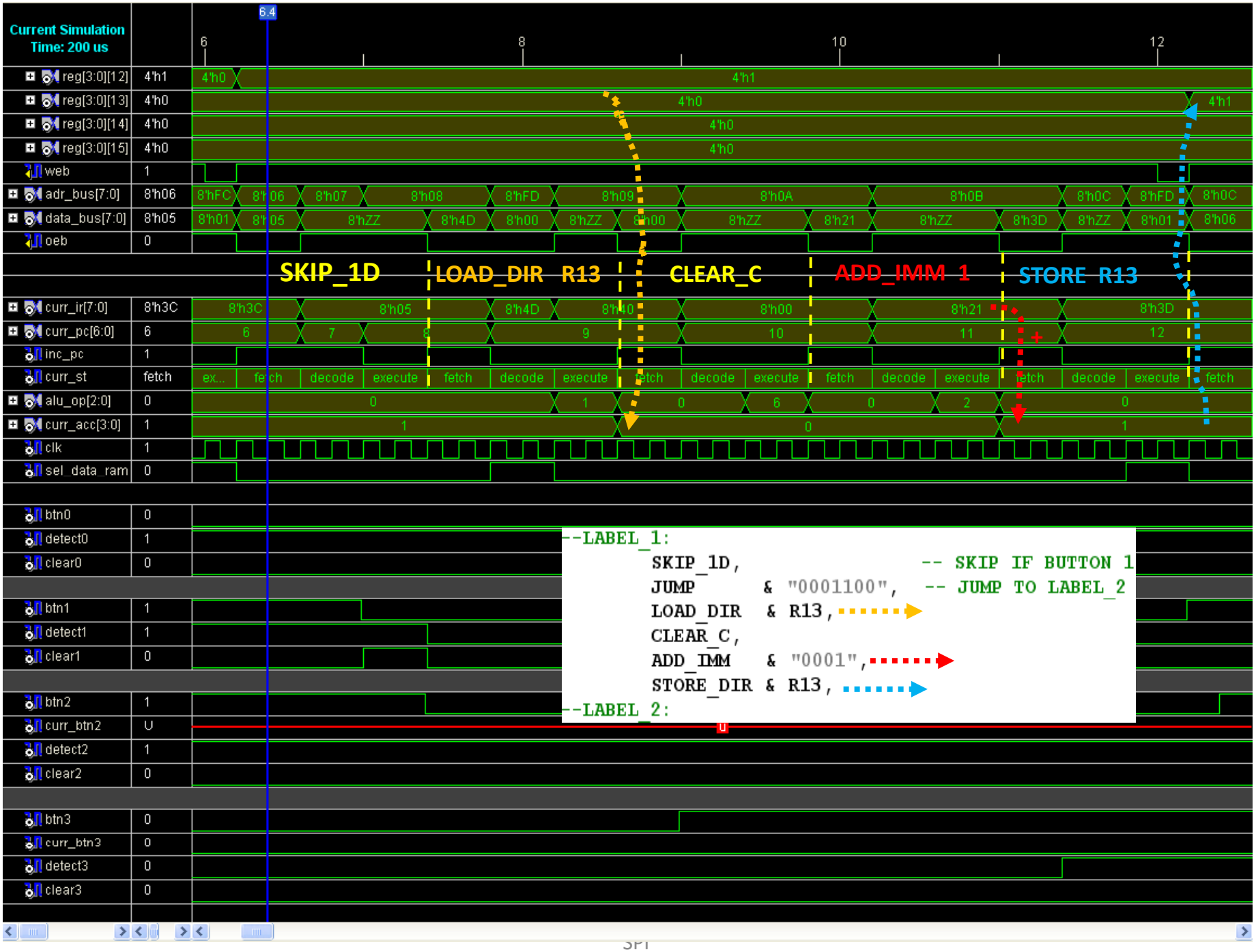
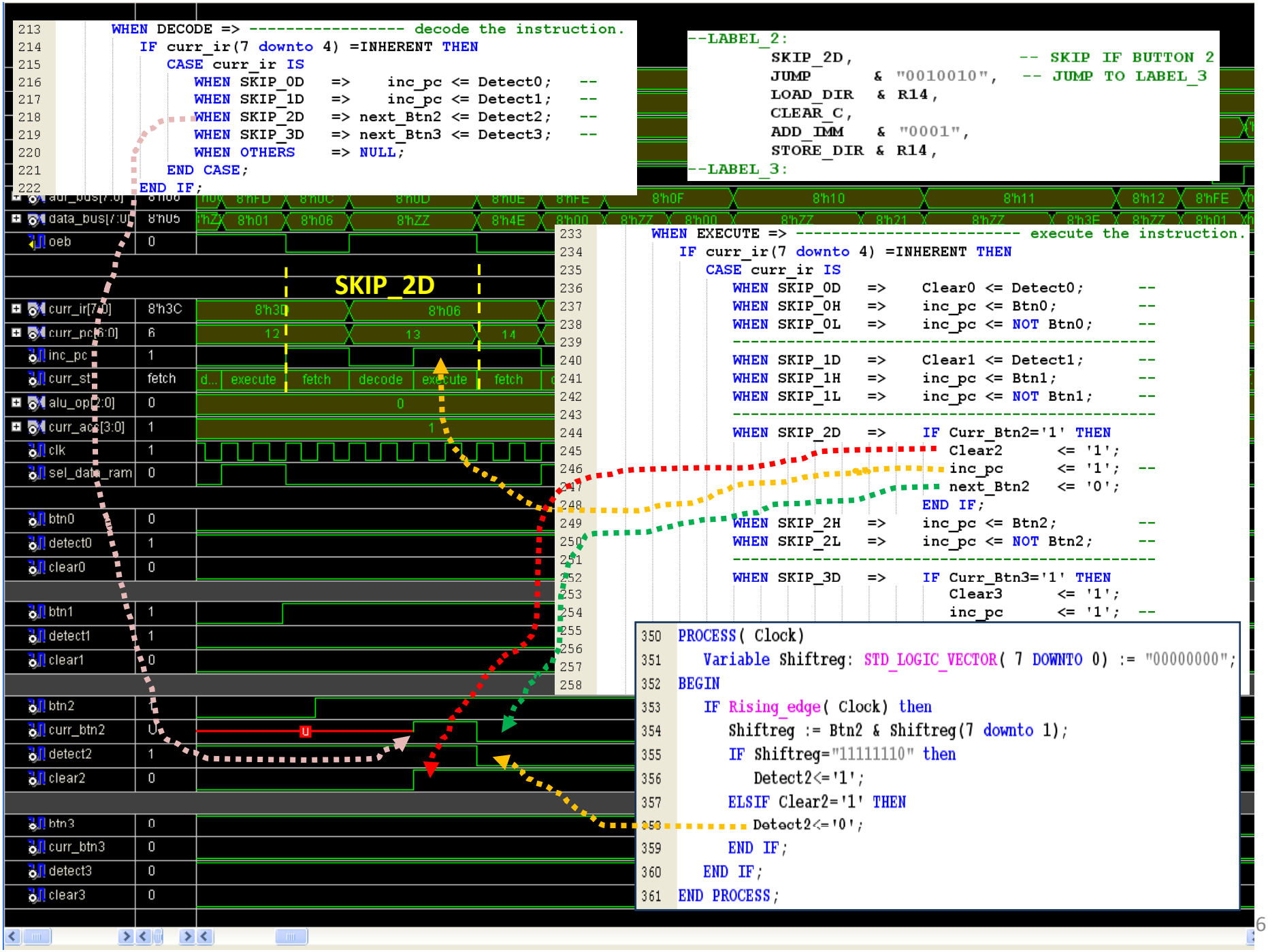

Current Simulation Time: 200 us
0
20
40
60
80
100
reg[3:0][12]
reg[3:0][13]
reg[3:0][14]
reg[3:0][15]
web
adr_bus[7:0]
data_bus[7:0]
oeb
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
btn0
detect0
clear0
btn1
detect1
clear1
btn2
curr_btn2
detect2
clear2
btn3
curr_btn3
detect3
clear3
??
??

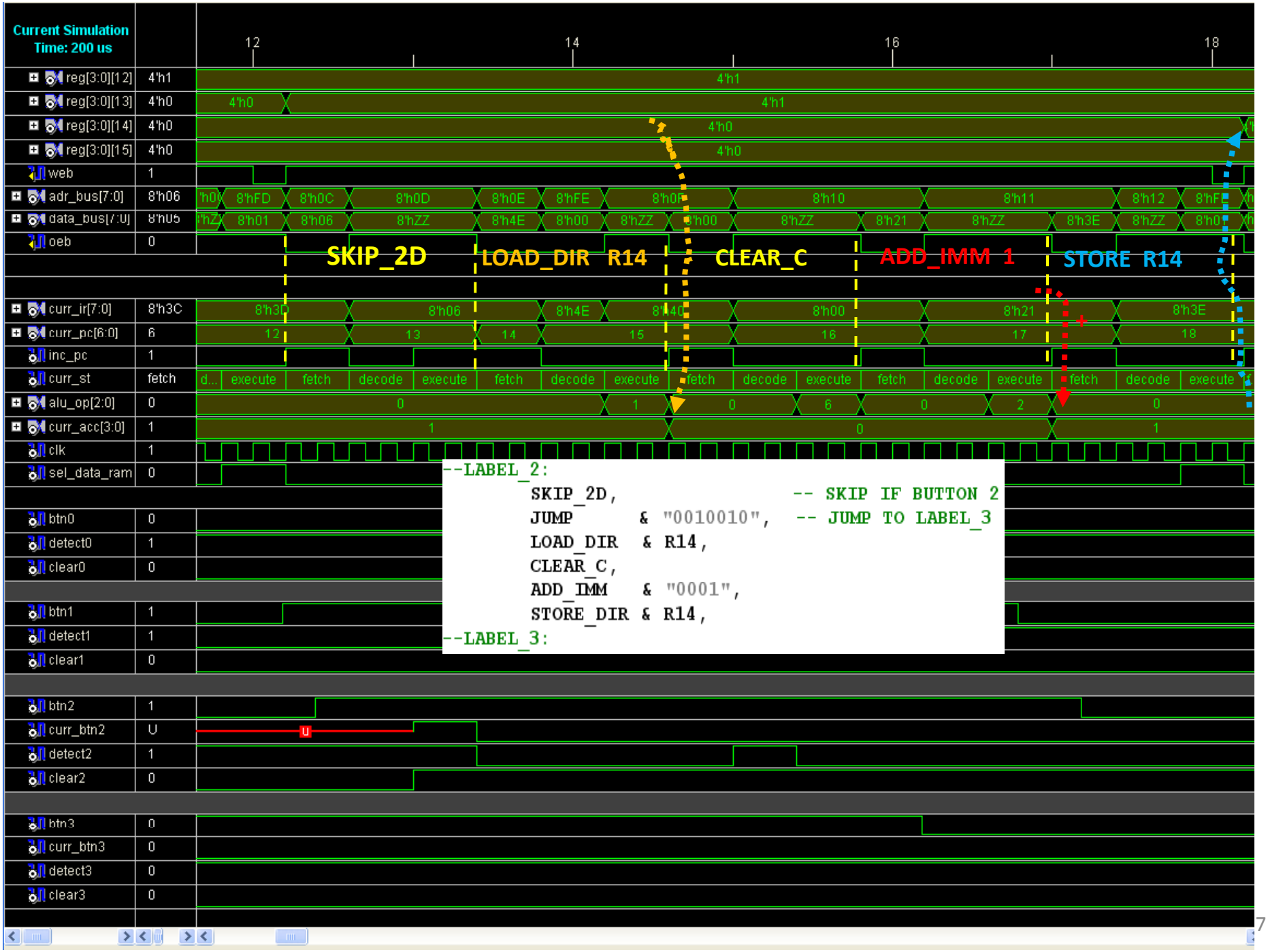

Current Simulation Time: 200 us
reg[3:0][12]
reg[3:0][13]
reg[3:0][14]
reg[3:0][15]
web
adr_bus[7:0]
data_bus[7:0]
oeb
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
btn0
detect0
clear0
clear2
btn3
curr_btn3
detect3
clear3
SKIP_0D
fetch
decode
execute
WHEN DECODE => ---------- decode the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => inc_pc <= Detect0;
WHEN SKIP_1D => inc_pc <= Detect1;
WHEN SKIP_2D => next_Btn2 <= Detect2;
WHEN SKIP_3D => next_Btn3 <= Detect3;
WHEN OTHERS => NULL;
END CASE;
END IF;
--LABEL_0:
SKIP_0D, -- SKIP IF BUTTON 0
JUMP & "0000110", -- JUMP TO LABEL_1
LOAD_DIR & R12,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R12,
--LABEL_1:
PROCESS(Btn0, Clear0)
BEGIN
IF Clear0='1' THEN
Detect0<='0';
ELSIF Rising_Edge(Btn0) THEN
Detect0<='1';
END IF;
END PROCESS;
WHEN EXECUTE => ---------- execute the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => Clear0 <= Detect0;
WHEN SKIP_0H => inc_pc <= Btn0;
WHEN SKIP_0L => inc_pc <= NOT Btn0;
WHEN SKIP_1D => Clear1 <= Detect1;
WHEN SKIP_1H => inc_pc <= Btn1;
WHEN SKIP_1L => inc_pc <= NOT Btn1;
WHEN SKIP_2D => IF Curr_Btn2='1' THEN
Clear2 <= '1';
inc_pc <= '1';
next_Btn2 <= '0';
END IF;
WHEN SKIP_2H => inc_pc <= Btn2;
WHEN SKIP_2L => inc_pc <= NOT Btn2;
WHEN SKIP_3D => IF Curr_Btn3='1' THEN
Clear3 <= '1';
inc_pc <= '1';
next_Btn3 <= '0';
END IF;
WHEN SKIP_3H => inc_pc <= Btn3;
WHEN SKIP_3L => inc_pc <= NOT Btn3;

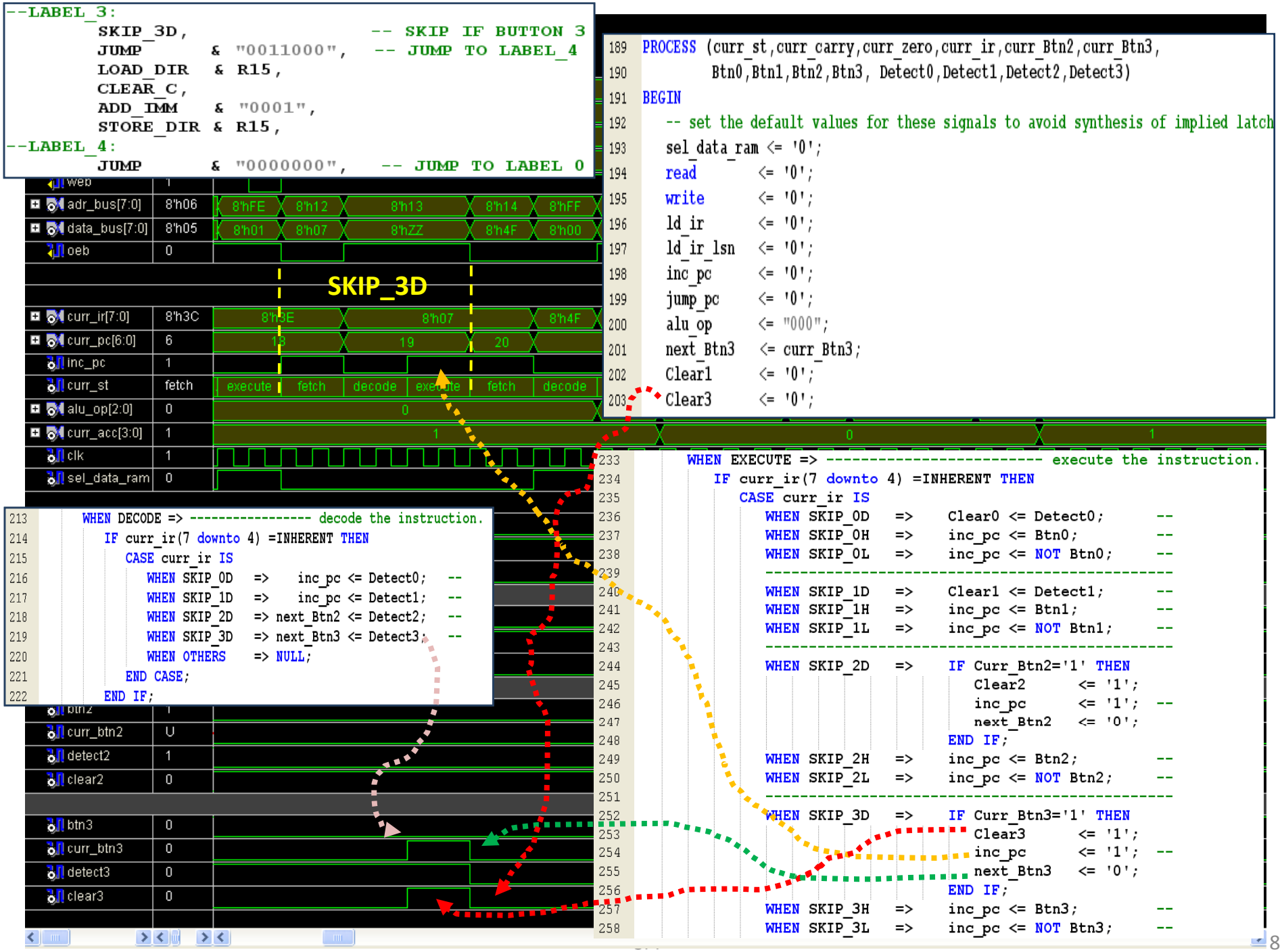

Current Simulation Time: 200 us
reg[3:0][12]
reg[3:0][13]
reg[3:0][14]
reg[3:0][15]
web
adr_bus[7:0]
data_bus[7:0]
oeb
SKIP_0D
LOAD_DIR R12
CLEAR_C
ADD_IMM 1
STORE R12
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
btn0
detect0
clear0
btn1
detect1
clear1
clear3
--LABEL_0:
SKIP_0D, -- SKIP IF BUTTON 0
JUMP & "0000110", -- JUMP TO LABEL_1
LOAD_DIR & R12,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R12,
--LABEL_1:
PROCESS(Clock)
Variable Shiftreg: STD_LOGIC_VECTOR(7 DOWNTO 0) := "00000000";
BEGIN
IF Rising_edge(Clock) then
Shiftreg := Btn1 & Shiftreg(7 downto 1);
IF Shiftreg="11111110" then
Detect1<='1';
ELSIF Clear1='1' THEN
Detect1<='0';
END IF;
END IF;
END PROCESS;

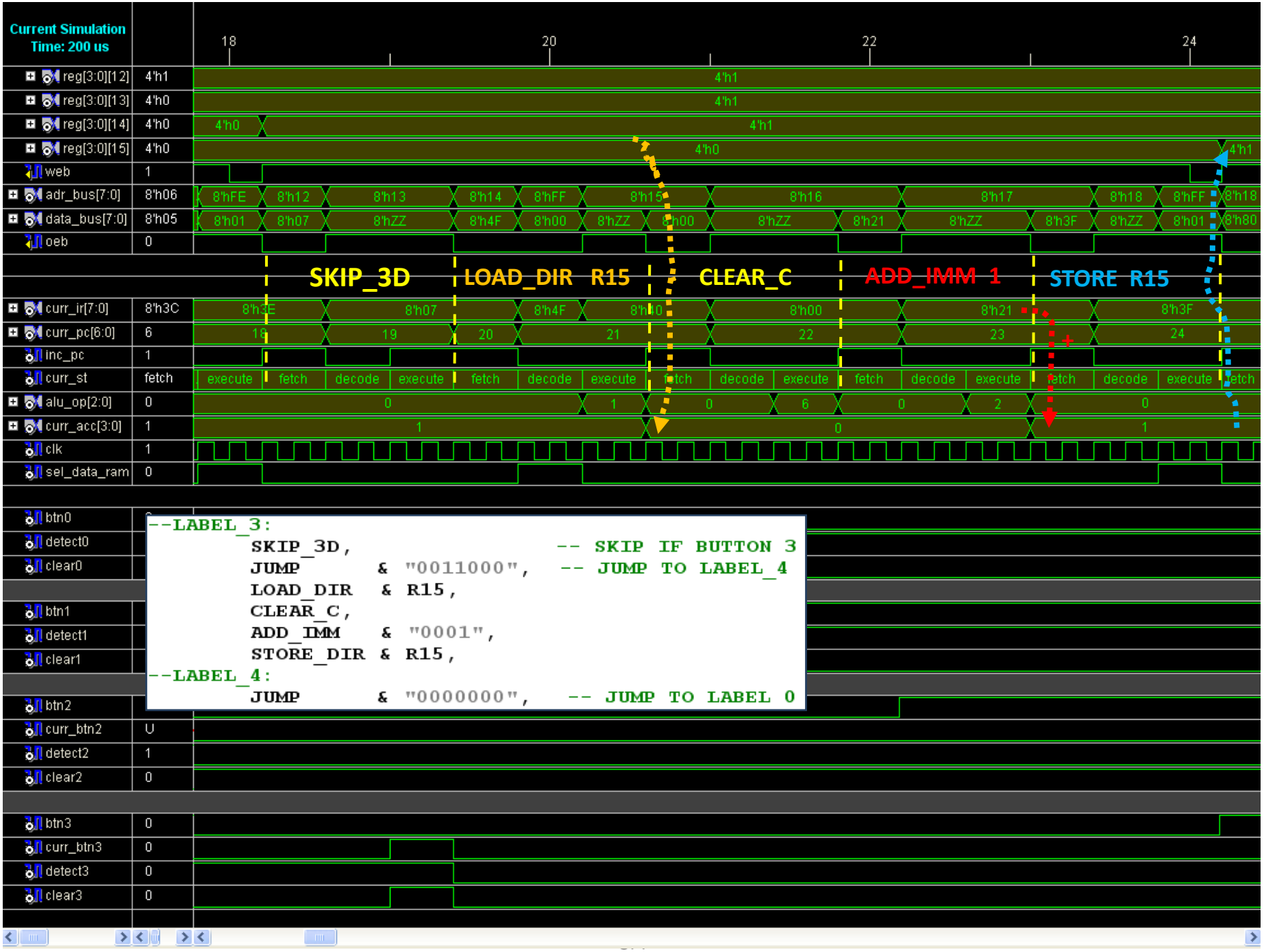

Current Simulation Time: 200 us
reg[3:0][12]
reg[3:0][13]
reg[3:0][14]
reg[3:0][15]
web
adr_bus[7:0]
data_bus[7:0]
oeb
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
btn0
detect0
clear0
btn1
detect1
clear1
SKIP_1D
WHEN DECODE => ---------- decode the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => inc_pc <= Detect0;
WHEN SKIP_1D => inc_pc <= Detect1;
WHEN SKIP_2D => next_Btn2 <= Detect2;
WHEN SKIP_3D => next_Btn3 <= Detect3;
WHEN OTHERS => NULL;
END CASE;
END IF;
--LABEL_1:
SKIP_1D, -- SKIP IF BUTTON 1
JUMP & "0001100", -- JUMP TO LABEL_2
LOAD_DIR & R13,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R13,
--LABEL_2:
WHEN EXECUTE => ---------- execute the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => Clear0 <= Detect0;
WHEN SKIP_0H => inc_pc <= Btn0;
WHEN SKIP_0L => inc_pc <= NOT Btn0;
WHEN SKIP_1D => Clear1 <= Detect1;
WHEN SKIP_1H => inc_pc <= Btn1;
WHEN SKIP_1L => inc_pc <= NOT Btn1;
WHEN SKIP_2D => IF Curr_Btn2='1' THEN
Clear2 <= '1';
inc_pc <= '1';
next_Btn2 <= '0';
END IF;
WHEN SKIP_2H => inc_pc <= Btn2;
WHEN SKIP_2L => inc_pc <= NOT Btn2;
WHEN SKIP_3D => IF Curr_Btn3='1' THEN
Clear3 <= '1';
inc_pc <= '1';
next_Btn3 <= '0';
END IF;
WHEN SKIP_3H => inc_pc <= Btn3;
WHEN SKIP_3L => inc_pc <= NOT Btn3;
PROCESS(Clock)
Variable Shiftreg: STD_LOGIC_VECTOR(7 DOWNTO 0) := "000
BEGIN
IF Rising_edge(Clock) then
Shiftreg := Btn1 & Shiftreg(7 downto 1);
IF Shiftreg="11111110" then
Detect1<='1';
ELSIF Clear1='1' THEN
Detect1<='0';
END IF;
END IF;
END PROCESS;

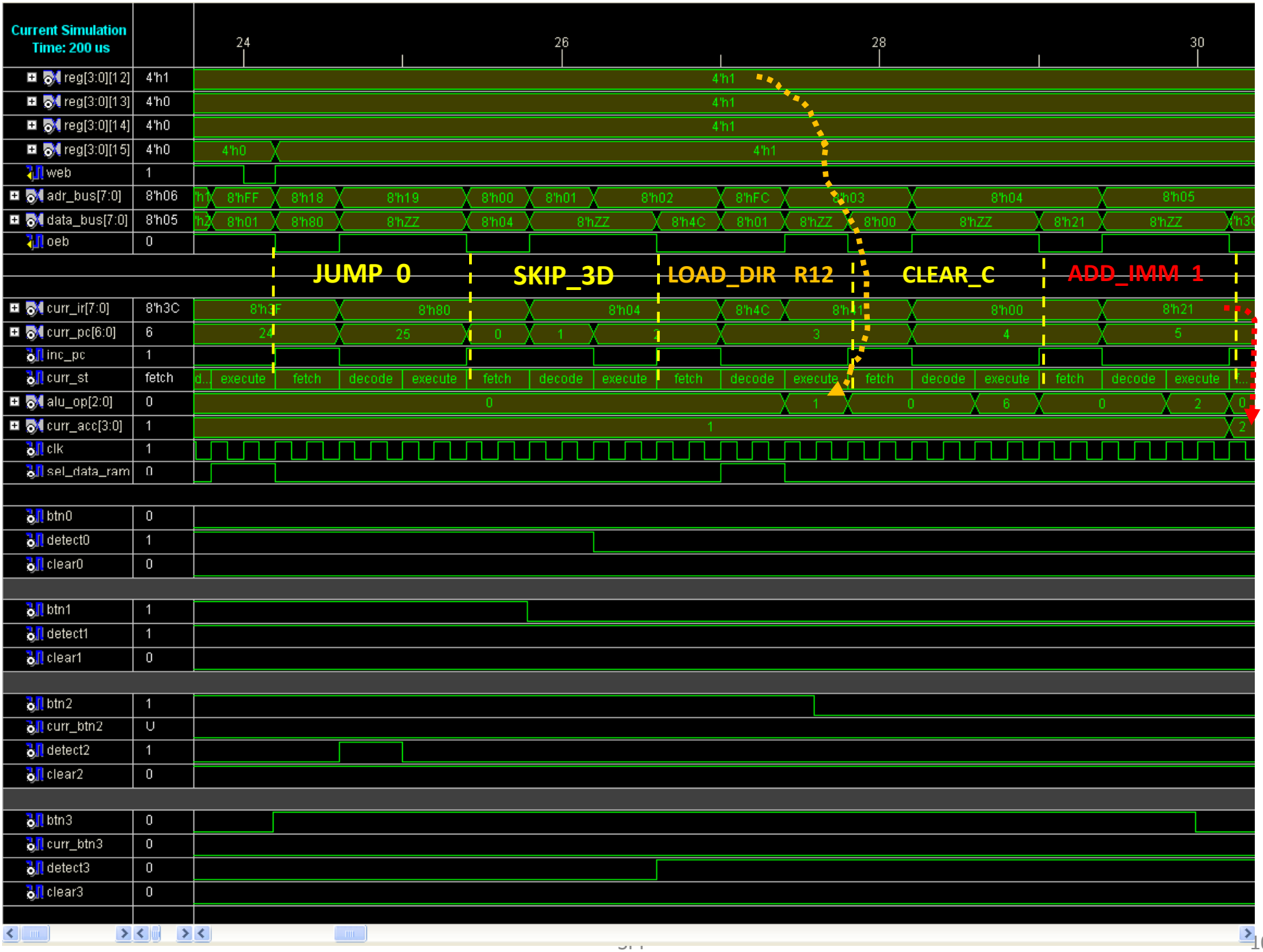

Current Simulation Time: 200 us
reg[3:0][12]
reg[3:0][13]
reg[3:0][14]
reg[3:0][15]
web
adr_bus[7:0]
data_bus[7:0]
oeb
SKIP_1D
LOAD_DIR R13
CLEAR_C
ADD_IMM 1
STORE R13
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
btn0
detect0
clear0
btn1
detect1
clear1
btn2
curr_btn2
detect2
clear2
btn3
curr_btn3
detect3
clear3
--LABEL_1:
SKIP_1D, -- SKIP IF BUTTON 1
JUMP & "0001100", -- JUMP TO LABEL_2
LOAD_DIR & R13,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R13,
--LABEL_2:

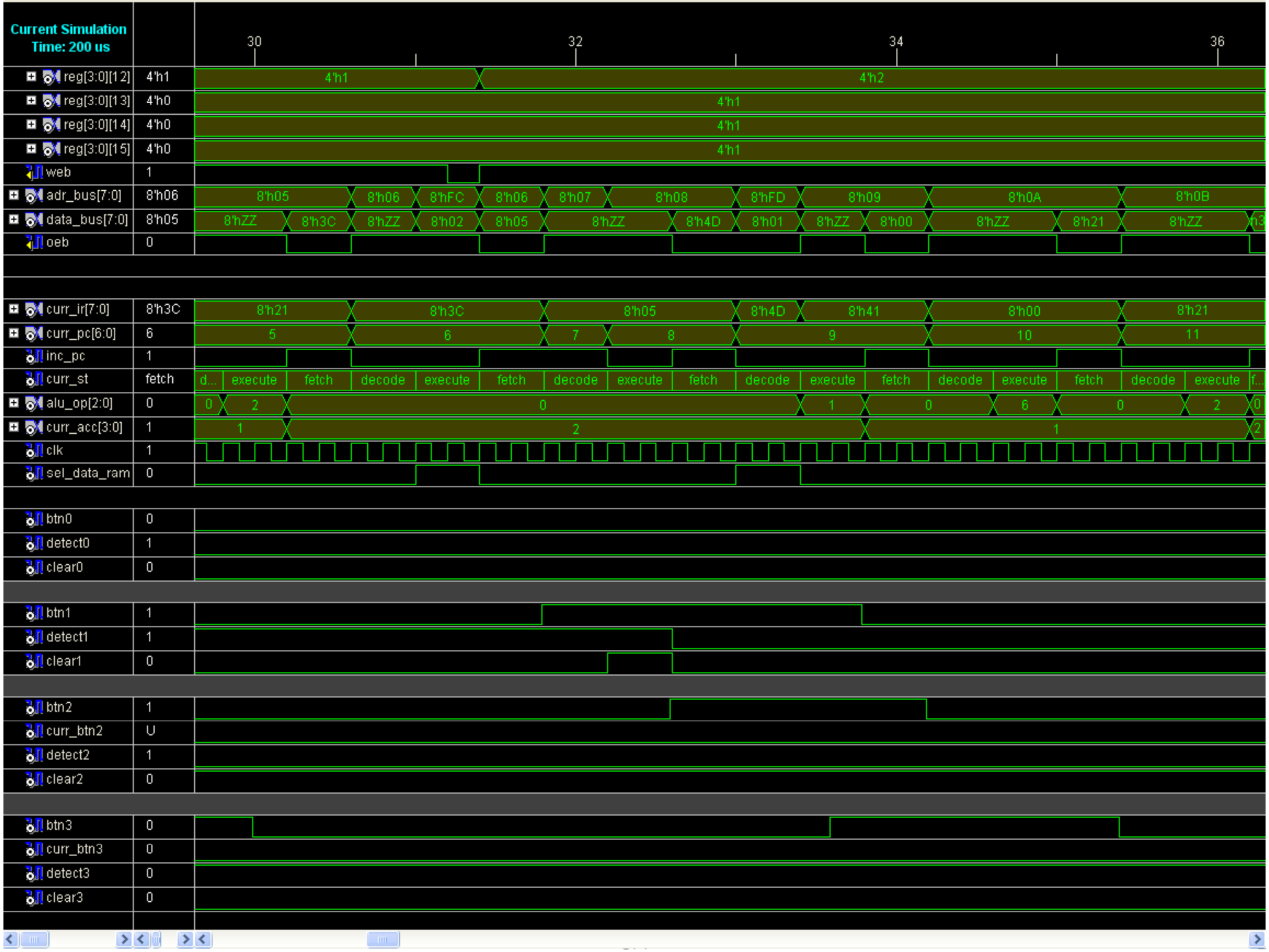

SKIP_2D
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
btn0
detect0
clear0
btn1
detect1
clear1
btn2
curr_btn2
detect2
clear2
btn3
curr_btn3
detect3
clear3
data_bus[7:0]
oeb
WHEN DECODE => ---------- decode the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => inc_pc <= Detect0;
WHEN SKIP_1D => inc_pc <= Detect1;
WHEN SKIP_2D => next_Btn2 <= Detect2;
WHEN SKIP_3D => next_Btn3 <= Detect3;
WHEN OTHERS => NULL;
END CASE;
END IF;
--LABEL_2:
SKIP_2D, -- SKIP IF BUTTON 2
JUMP & "0010010", -- JUMP TO LABEL_3
LOAD_DIR & R14,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R14,
--LABEL_3:
WHEN EXECUTE => ---------- execute the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => Clear0 <= Detect0;
WHEN SKIP_0H => inc_pc <= Btn0;
WHEN SKIP_0L => inc_pc <= NOT Btn0;
WHEN SKIP_1D => Clear1 <= Detect1;
WHEN SKIP_1H => inc_pc <= Btn1;
WHEN SKIP_1L => inc_pc <= NOT Btn1;
WHEN SKIP_2D => IF Curr_Btn2='1' THEN
Clear2 <= '1';
inc_pc <= '1';
next_Btn2 <= '0';
END IF;
WHEN SKIP_2H => inc_pc <= Btn2;
WHEN SKIP_2L => inc_pc <= NOT Btn2;
WHEN SKIP_3D => IF Curr_Btn3='1' THEN
Clear3 <= '1';
inc_pc <= '1';
PROCESS(Clock)
Variable Shiftreg: STD_LOGIC_VECTOR(7 DOWNTO 0) := "00000000";
BEGIN
IF Rising_edge(Clock) then
Shiftreg := Btn2 & Shiftreg(7 downto 1);
IF Shiftreg="11111110" then
Detect2<='1';
ELSIF Clear2='1' THEN
Detect2<='0';
END IF;
END IF;
END PROCESS;

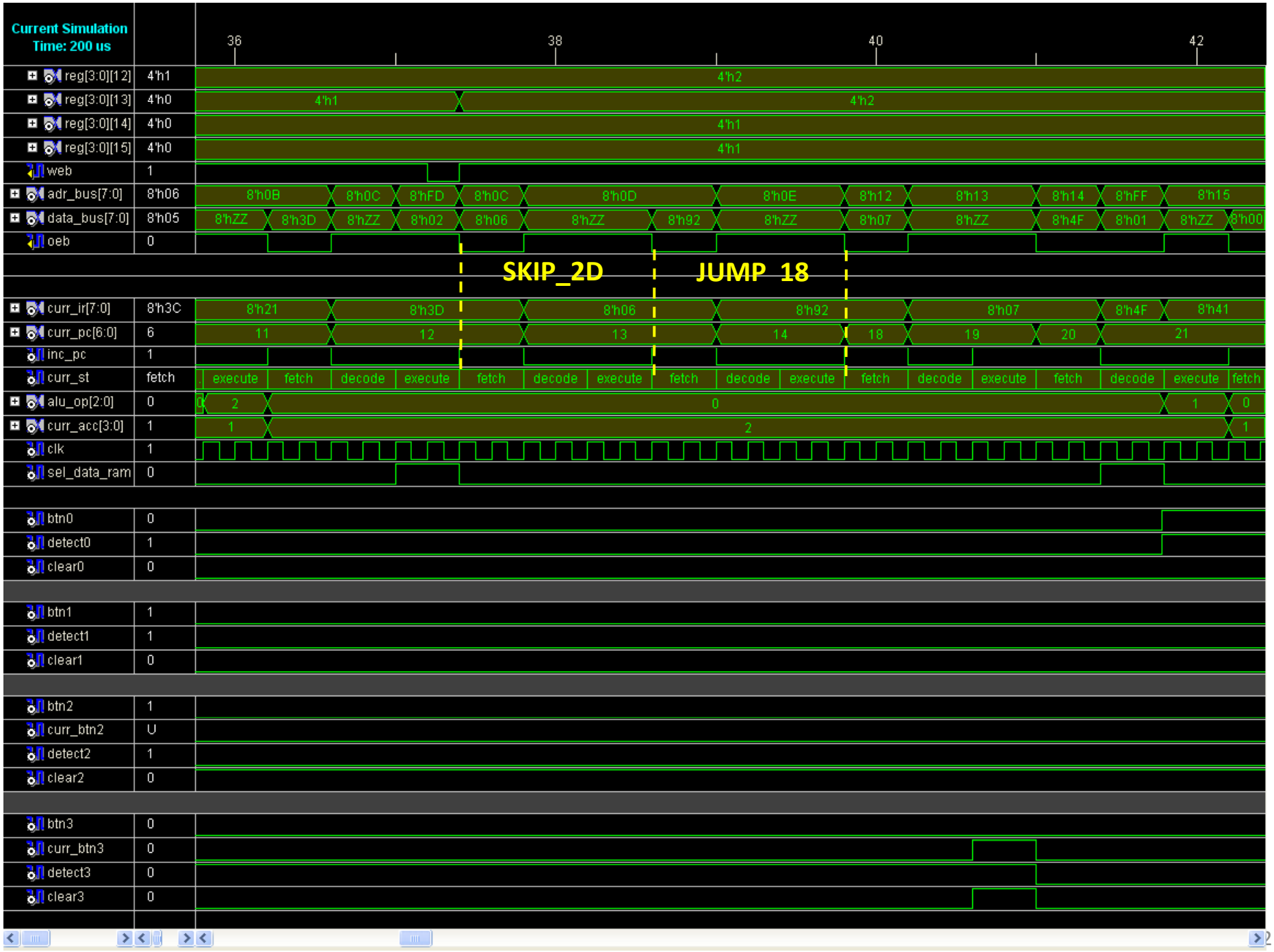

Current Simulation Time: 200 us
SKIP_2D
LOAD_DIR R14
CLEAR_C
ADD_IMM 1
STORE R14
--LABEL_2:
SKIP_2D, -- SKIP IF BUTTON 2
JUMP & "0010010", -- JUMP TO LABEL_3
LOAD_DIR & R14,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R14,
--LABEL_3:

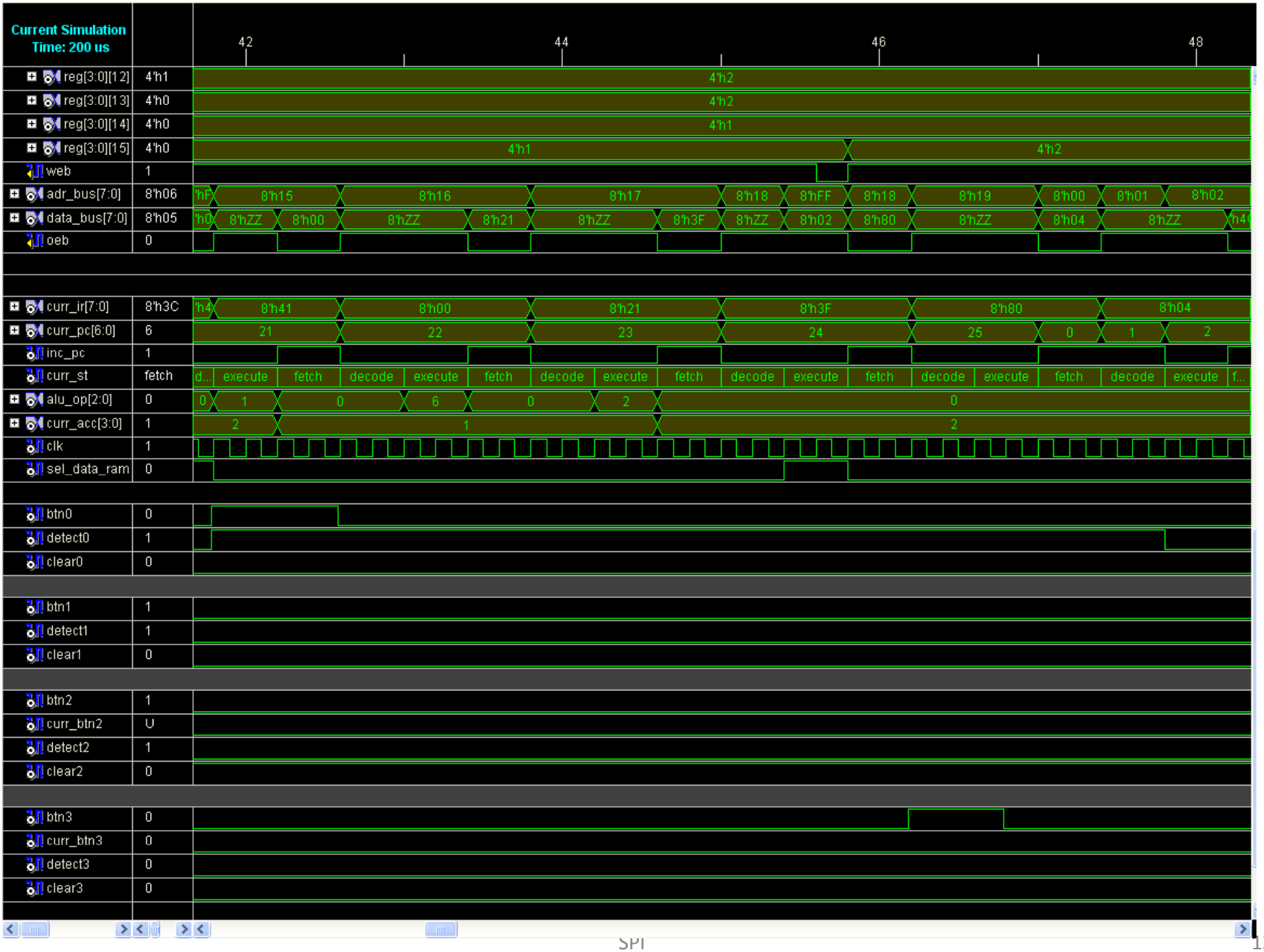

--LABEL_3:
SKIP_3D, -- SKIP IF BUTTON 3
JUMP & "0011000", -- JUMP TO LABEL_4
LOAD_DIR & R15,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R15,
--LABEL_4:
JUMP & "0000000", -- JUMP TO LABEL 0
web
adr_bus[7:0]
data_bus[7:0]
oeb
SKIP_3D
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
curr_btn2
detect2
clear2
btn3
curr_btn3
detect3
clear3
PROCESS (curr_st,curr_carry,curr_zero,curr_ir,curr_Btn2,curr_Btn3,
Btn0,Btn1,Btn2,Btn3, Detect0,Detect1,Detect2,Detect3)
BEGIN
-- set the default values for these signals to avoid synthesis of implied latch
sel_data_ram <= '0';
read <= '0';
write <= '0';
ld_ir <= '0';
ld_ir_lsn <= '0';
inc_pc <= '0';
jump_pc <= '0';
alu_op <= "000";
next_Btn3 <= curr_Btn3;
Clear1 <= '0';
Clear3 <= '0';
WHEN DECODE => ---------- decode the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => inc_pc <= Detect0; --
WHEN SKIP_1D => inc_pc <= Detect1; --
WHEN SKIP_2D => next_Btn2 <= Detect2; --
WHEN SKIP_3D => next_Btn3 <= Detect3; --
WHEN OTHERS => NULL;
END CASE;
END IF;
WHEN EXECUTE => ---------- execute the instruction.
IF curr_ir(7 downto 4) =INHERENT THEN
CASE curr_ir IS
WHEN SKIP_0D => Clear0 <= Detect0; --
WHEN SKIP_0H => inc_pc <= Btn0; --
WHEN SKIP_0L => inc_pc <= NOT Btn0; --
WHEN SKIP_1D => Clear1 <= Detect1; --
WHEN SKIP_1H => inc_pc <= Btn1; --
WHEN SKIP_1L => inc_pc <= NOT Btn1; --
WHEN SKIP_2D => IF Curr_Btn2='1' THEN
Clear2 <= '1';
inc_pc <= '1'; --
next_Btn2 <= '0';
END IF;
WHEN SKIP_2H => inc_pc <= Btn2; --
WHEN SKIP_2L => inc_pc <= NOT Btn2; --
WHEN SKIP_3D => IF Curr_Btn3='1' THEN
Clear3 <= '1';
inc_pc <= '1'; --
next_Btn3 <= '0';
END IF;
WHEN SKIP_3H => inc_pc <= Btn3; --
WHEN SKIP_3L => inc_pc <= NOT Btn3; --

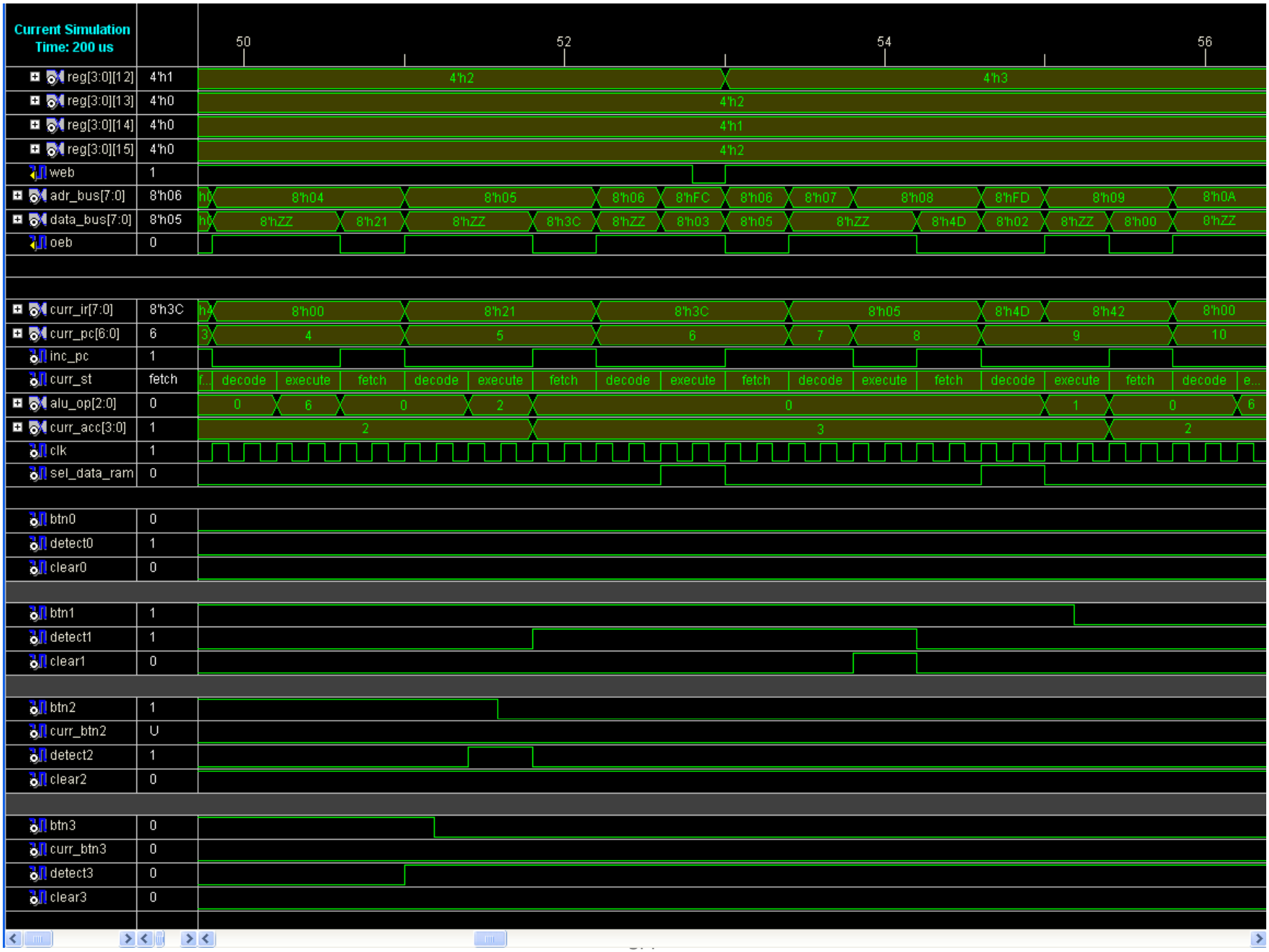

Current Simulation Time: 200 us
18
20
22
24
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
SKIP_3D
LOAD_DIR R15
CLEAR_C
ADD_IMM 1
STORE R15
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0
detect0
clear0
btn1
detect1
clear1
btn2
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0
--LABEL_3:
SKIP_3D, -- SKIP IF BUTTON 3
JUMP & "0011000", -- JUMP TO LABEL_4
LOAD_DIR & R15,
CLEAR_C,
ADD_IMM & "0001",
STORE_DIR & R15,
--LABEL_4:
JUMP & "0000000", -- JUMP TO LABEL 0

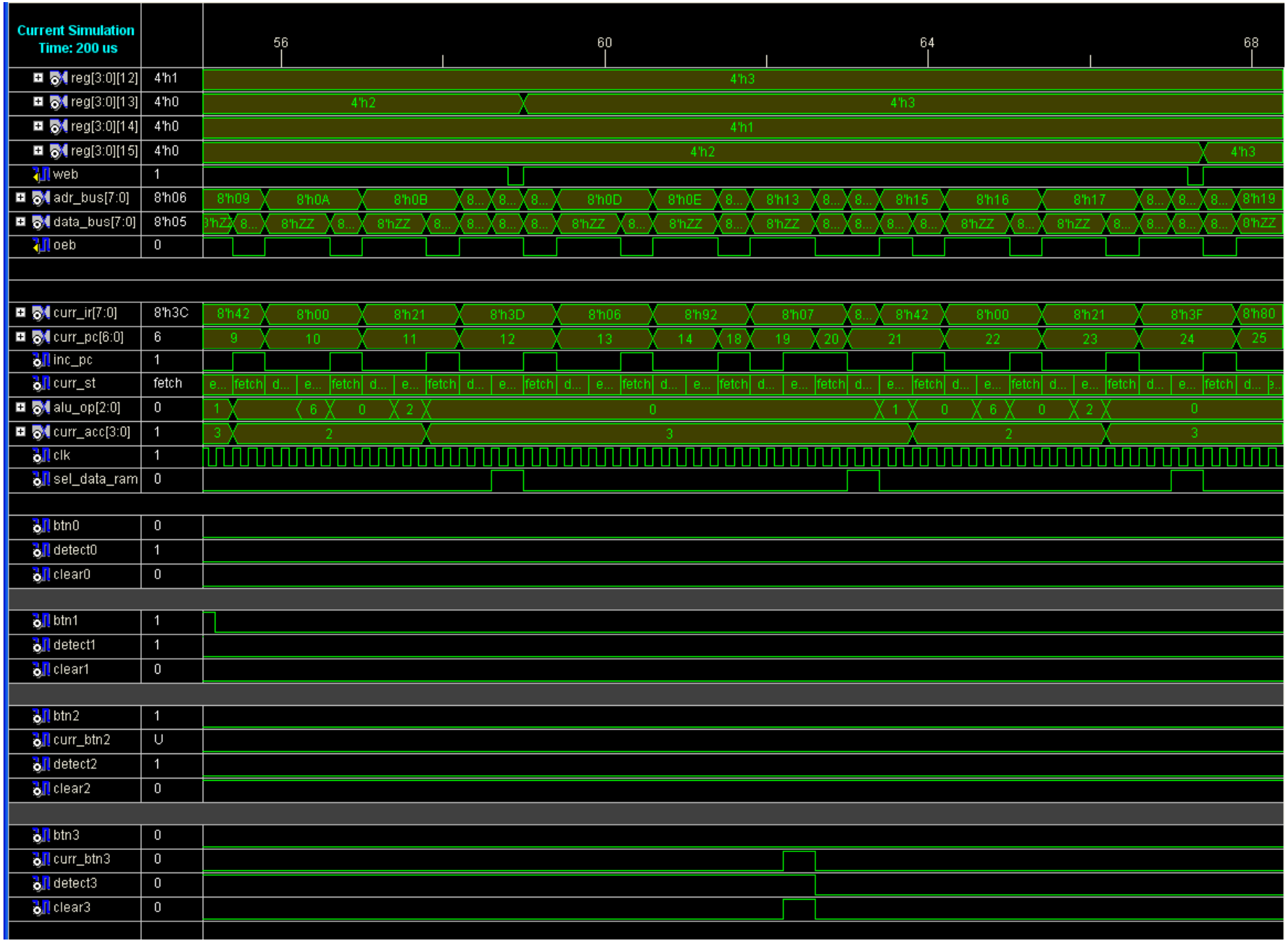

Current Simulation Time: 200 us
24
26
28
30
reg[3:0][12]
reg[3:0][13]
reg[3:0][14]
reg[3:0][15]
web
adr_bus[7:0]
data_bus[7:0]
oeb
JUMP 0
SKIP_3D
LOAD_DIR R12
CLEAR_C
ADD_IMM 1
curr_ir[7:0]
curr_pc[6:0]
inc_pc
curr_st
alu_op[2:0]
curr_acc[3:0]
clk
sel_data_ram
btn0
detect0
clear0
btn1
detect1
clear1
btn2
curr_btn2
detect2
clear2
btn3
curr_btn3
detect3
clear3

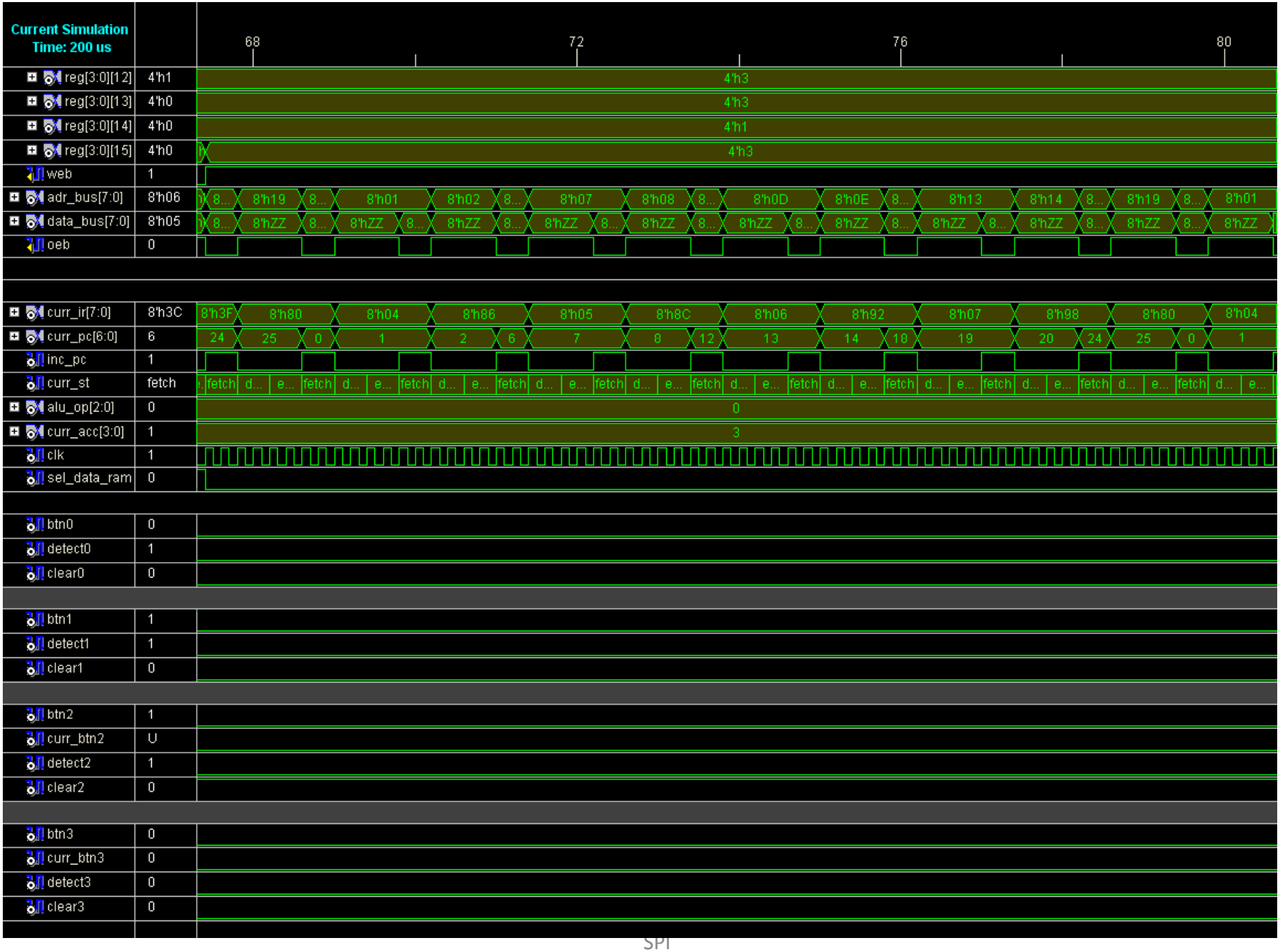

Current Simulation Time: 200 us
30
32
34
36
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0

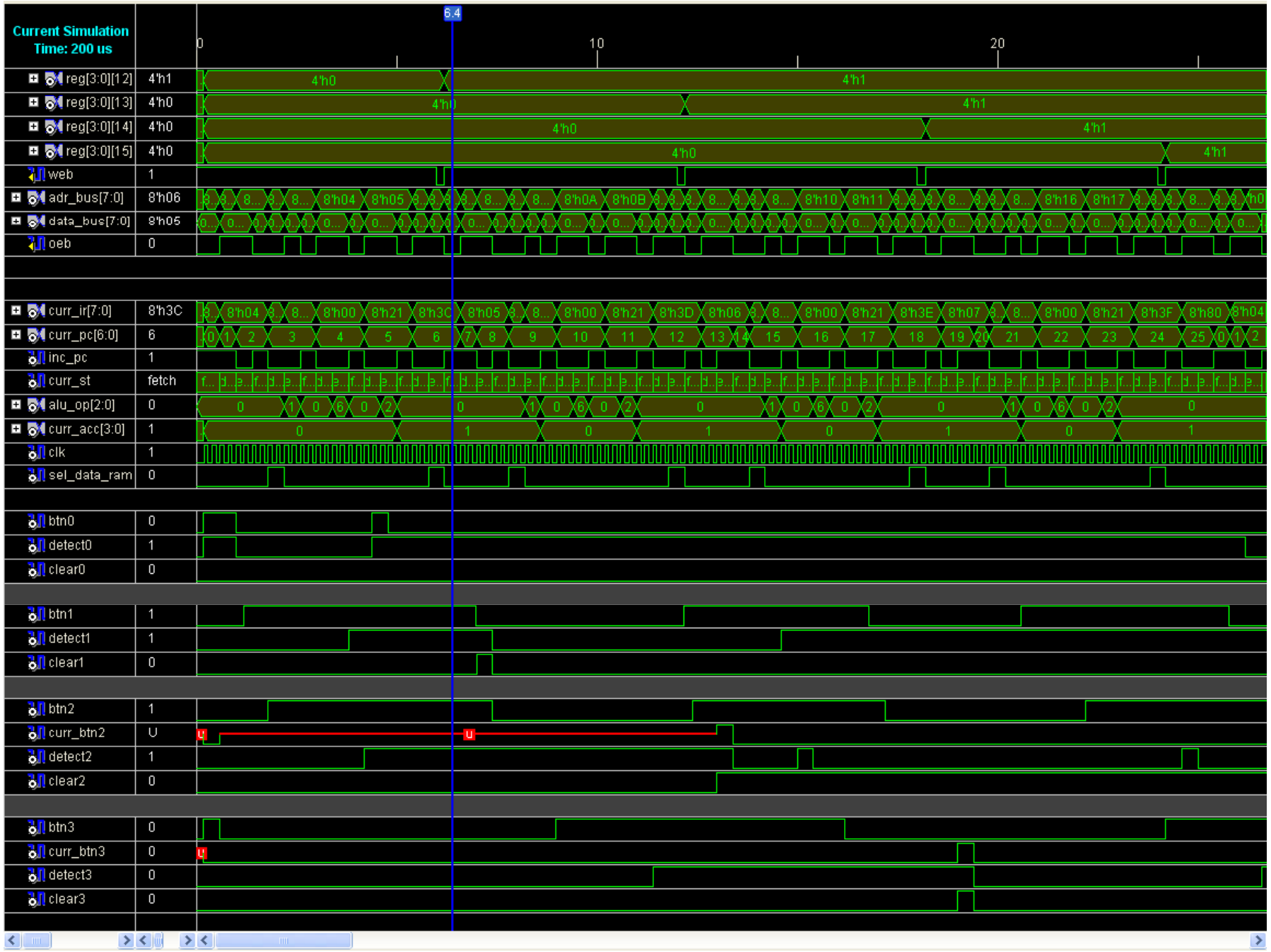

Current Simulation Time: 200 us
36
38
40
42
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
SKIP_2D
JUMP_18
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0

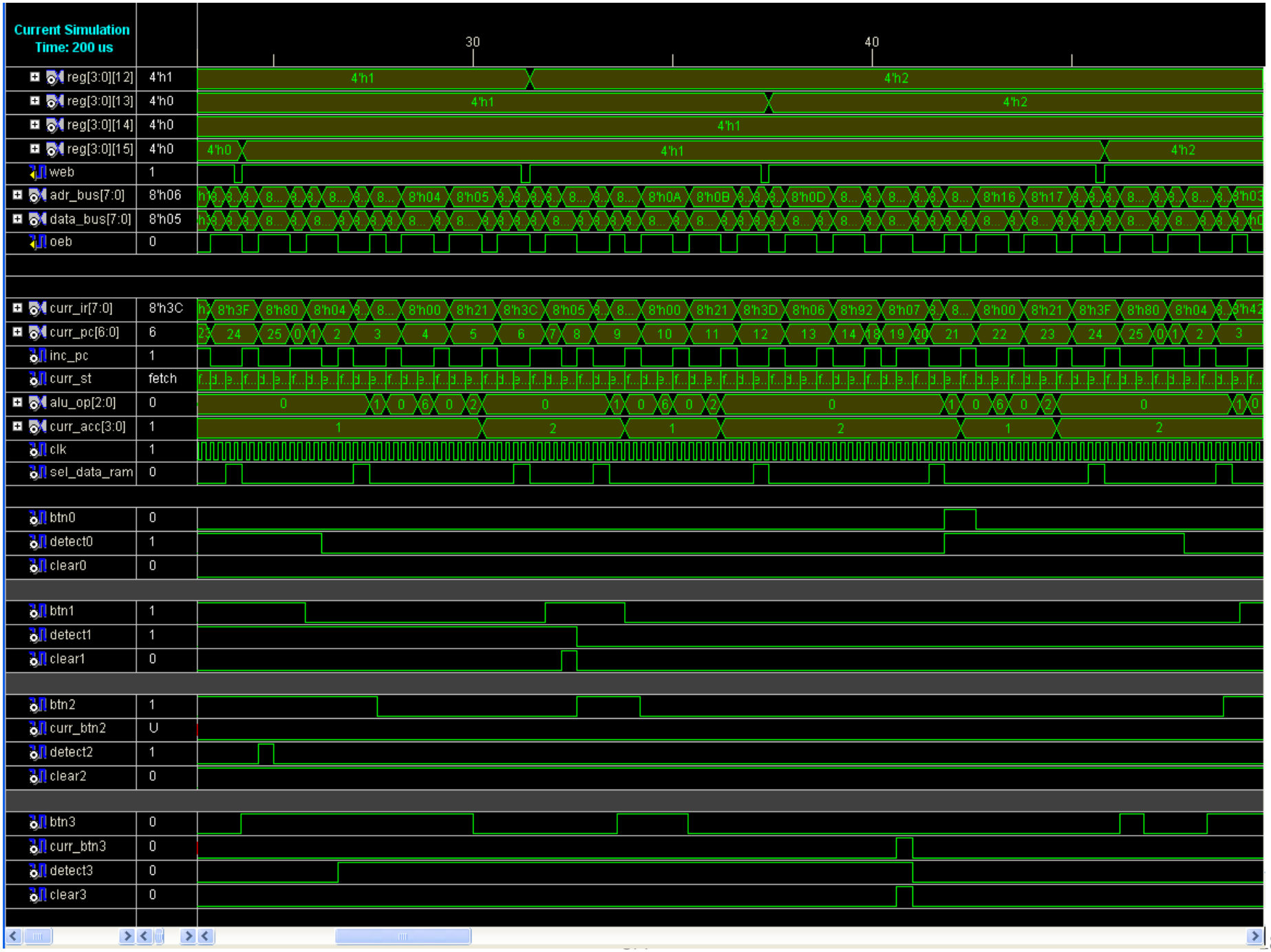

Current Simulation Time: 200 us
42
44
46
48
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0
4'h2
4'h1
8'h15
8'h16
8'h17
8'h18
8'hFF
8'h19
8'h00
8'h01
8'h02
8'hZZ
8'h21
8'h3F
8'h80
8'h04
8'h41
execute
fetch
decode

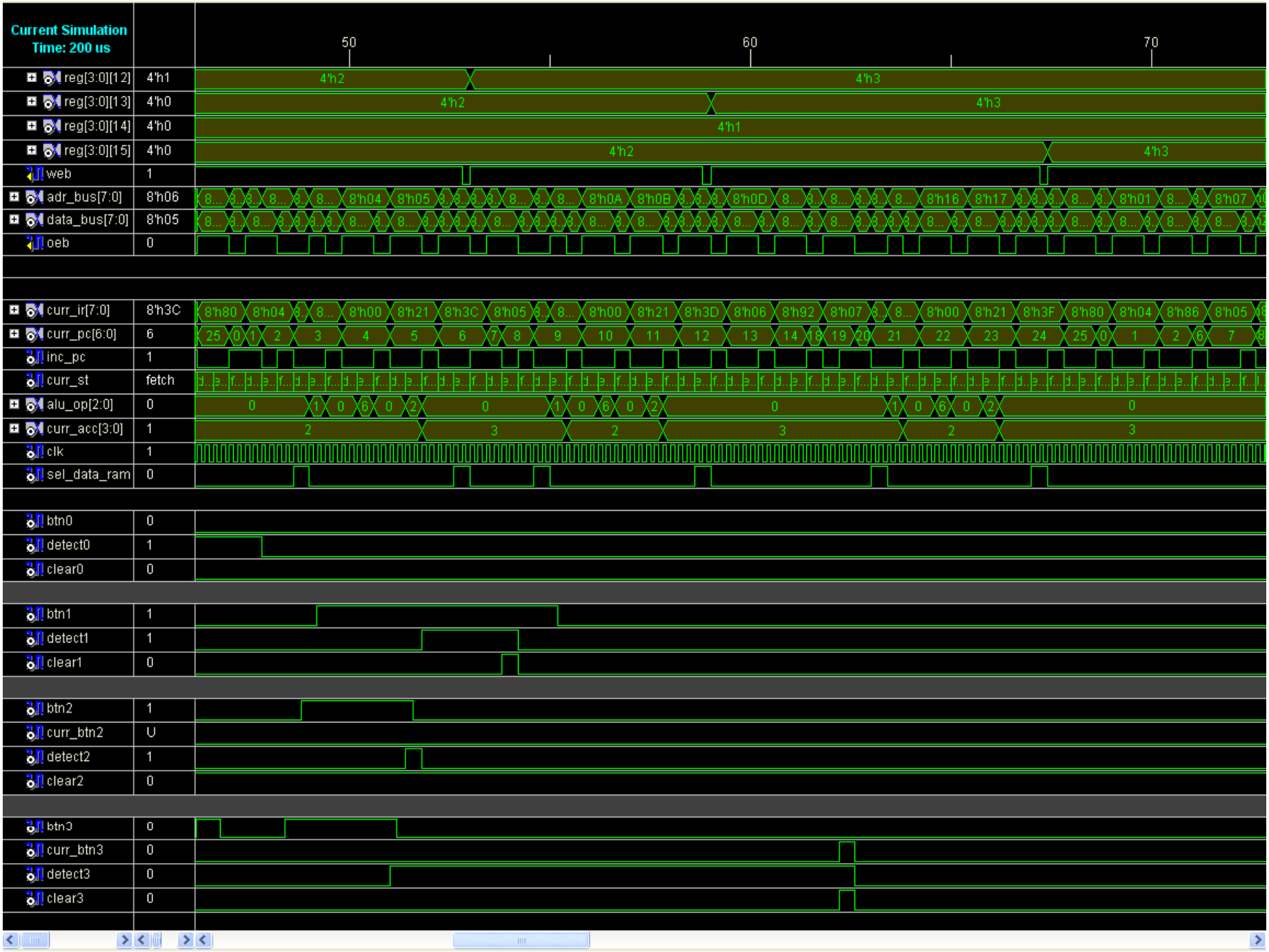

Current Simulation Time: 200 us
50
52
54
56
reg[3:0][12] 4'h1 4'h2 4'h3
reg[3:0][13] 4'h0 4'h2
reg[3:0][14] 4'h0 4'h1
reg[3:0][15] 4'h0 4'h2
web 1
adr_bus[7:0] 8'h06 8'h04 8'h05 8'h06 8'hFC 8'h06 8'h07 8'h08 8'hFD 8'h09 8'h0A
data_bus[7:0] 8'h05 8'hZZ 8'h21 8'hZZ 8'h3C 8'hZZ 8'h03 8'h05 8'hZZ 8'h4D 8'h02 8'hZZ 8'h00 8'hZZ
oeb 0
curr_ir[7:0] 8'h3C 8'h00 8'h21 8'h3C 8'h05 8'h4D 8'h42 8'h00
curr_pc[6:0] 6 4 5 6 7 8 9 10
inc_pc 1
curr_st fetch decode execute fetch decode execute fetch decode execute fetch decode execute fetch decode execute fetch decode
alu_op[2:0] 0 0 6 0 2 0 1 0 6
curr_acc[3:0] 1 2 3 2
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0

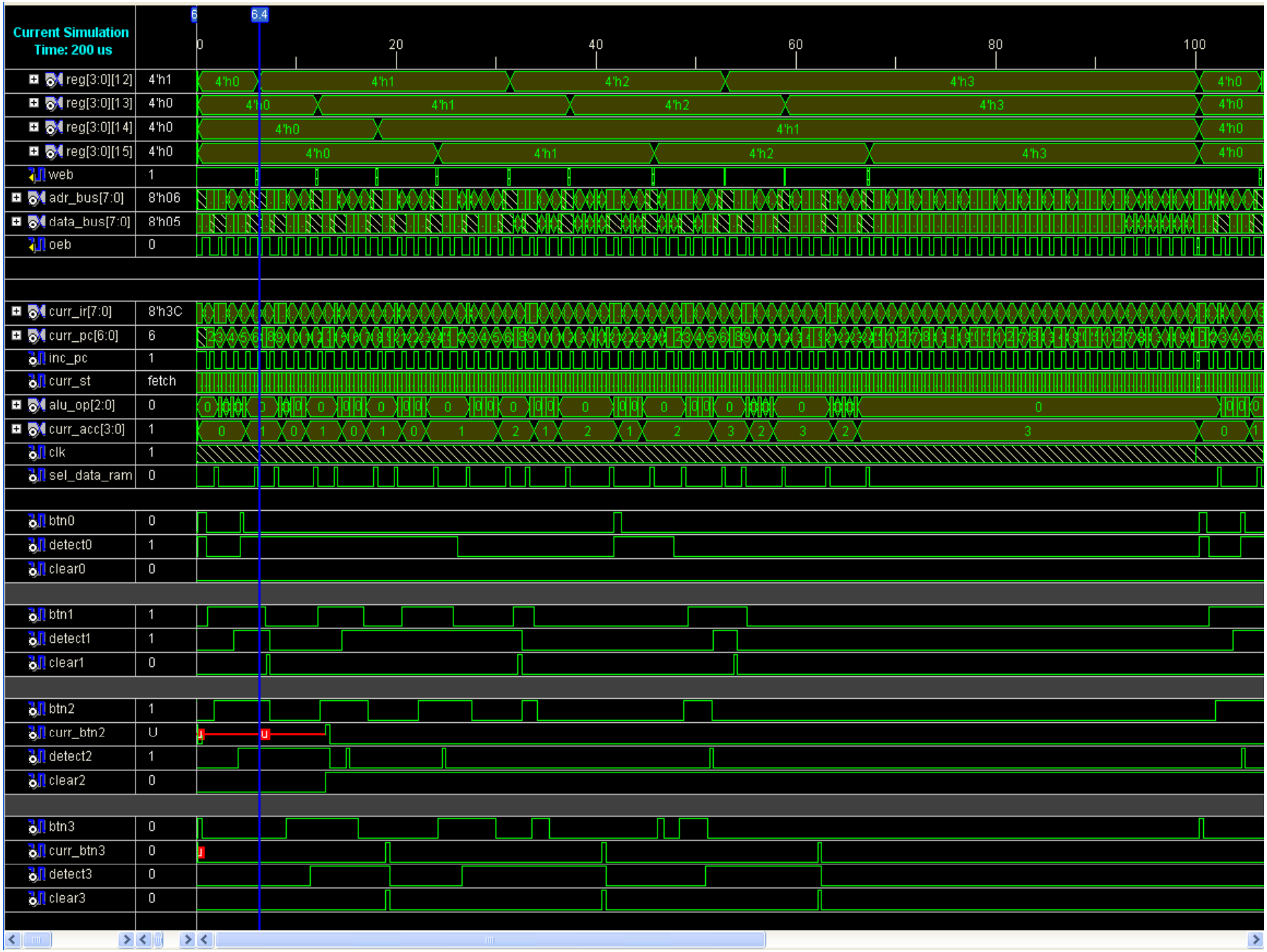

Current Simulation Time: 200 us
56
60
64
68
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0


Current Simulation Time: 200 us
68
72
76
80
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0


Current Simulation Time: 200 us
6.4
0
10
20
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0


Current Simulation Time: 200 us
30
40
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0


Current Simulation Time: 200 us
50
60
70
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0


Current Simulation Time: 200 us
0
20
40
60
80
100
reg[3:0][12] 4'h1
reg[3:0][13] 4'h0
reg[3:0][14] 4'h0
reg[3:0][15] 4'h0
web 1
adr_bus[7:0] 8'h06
data_bus[7:0] 8'h05
oeb 0
curr_ir[7:0] 8'h3C
curr_pc[6:0] 6
inc_pc 1
curr_st fetch
alu_op[2:0] 0
curr_acc[3:0] 1
clk 1
sel_data_ram 0
btn0 0
detect0 1
clear0 0
btn1 1
detect1 1
clear1 0
btn2 1
curr_btn2 U
detect2 1
clear2 0
btn3 0
curr_btn3 0
detect3 0
clear3 0

```
66  --LABEL_0:
67          SKIP_0D,                    -- SKIP IF BUTTON 0 ...
68          JUMP      & "0000110",      -- JUMP TO LABEL_1
69          LOAD_DIR  & R12,
70          CLEAR_C,
71          ADD_IMM   & "0001",
72          STORE_DIR & R12,
73  --LABEL_1:
74          SKIP_1D,                    -- SKIP IF BUTTON 1 ...
75          JUMP      & "0001100",      -- JUMP TO LABEL_2
76          LOAD_DIR  & R13,
77          CLEAR_C,
78          ADD_IMM   & "0001",
79          STORE_DIR & R13,
80  --LABEL_2:
81          SKIP_2D,                    -- SKIP IF BUTTON 2 ...
82          JUMP      & "0010010",      -- JUMP TO LABEL_3
83          LOAD_DIR  & R14,
84          CLEAR_C,
85          ADD_IMM   & "0001",
86          STORE_DIR & R14,
87  --LABEL_3:
88          SKIP_3D,                    -- SKIP IF BUTTON 3 ...
89          JUMP      & "0011000",      -- JUMP TO LABEL_4
90          LOAD_DIR  & R15,
91          CLEAR_C,
92          ADD_IMM   & "0001",
93          STORE_DIR & R15,
94  --LABEL_4:
95          JUMP      & "0000000",      -- JUMP TO LABEL 0
96          JUMP      & "0000000",
97          NOP,
98          NOP,
99          NOP,
100         NOP,
101         NOP,
102         NOP
103         );
```