Christopher Bero
CPE 322
Homework Simulation #2

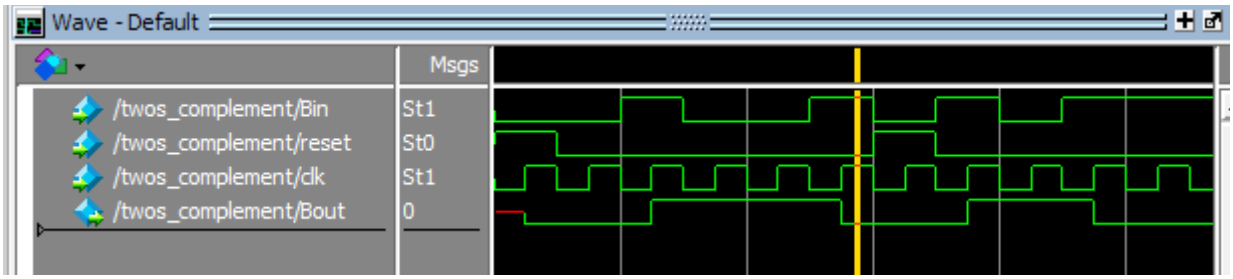
Mealy Simulation

```
// Christopher Bero


// HW2 - Part 1 - Mealy FSM
module twos_complement(input Bin,clk,reset, output reg Bout);

   reg [3:0] state,next_state;

      parameter S0 = 0, S1 = 1, S2 = 2;

      always @ ( posedge clk)
      begin
                  case (state)

                        S0:
                        begin
                          Bout = Bin;
                        if (Bin)
                              next_state = S2;
                        else
                          next_state = S1;
                        end

                        S1:
                        begin
                          Bout = 1;
                        if (Bin)
                              next_state = S1;
                        else
                          next_state = S2;
                        end

                        S2:
                        begin
                          Bout = ~Bin;
                        if (Bin)
                              next_state = S2;
                        else
                          next_state = S2;
                        end

                  endcase

                  if (reset)
                      begin
                        state = S0;
                        Bout = 0;
                      end
                  else
                        state = next_state;
```

```
      end

endmodule
```

Stimulus

```
#
# Stimulus
# [0] is LSB
# SUB_ADD: 0=add (A+B), 1=subtract (A-B)
# Library: cycloneive_ver
#

#add list Bin reset clk Bout

#add wave Bin reset clk Bout

force Bin   0 0ns, 0 50ns, 1 100ns, 1 150ns, 0 200ns, 0 250ns, 1 300ns, 1 350ns, 0
400ns, 0 450ns, 1 500ns

force clk   0 0ns, 1 25ns, 0 50ns, 1 75ns, 0 100ns, 1 125ns, 0 150ns, 1 175ns, 0
200ns, 1 225ns, 0 250ns, 1 275ns, 0 300ns, 1 325ns, 0 350ns, 1 375ns, 0 400ns, 1
425ns, 0 450ns, 1 475ns, 0 500ns, 1 525ns, 0 550ns, 1 575ns, 0 600ns, 1 625ns, 0
650ns, 1 675ns, 0 700ns, 1 725ns, 0 750ns, 1 775ns, 0 800ns, 1 825ns, 0 850ns, 1
875ns

force reset 1 0ns, 0 50ns,
                                          1 450ns, 0 500ns

run 1000ns

# Sample input:   01100110 1
# Returned output:      00111100 0

# Formatted:
# Original input:       1 01100110
# 2s complement:  1 10011011
# Program output:       0 00111100 <- incorrect output
```

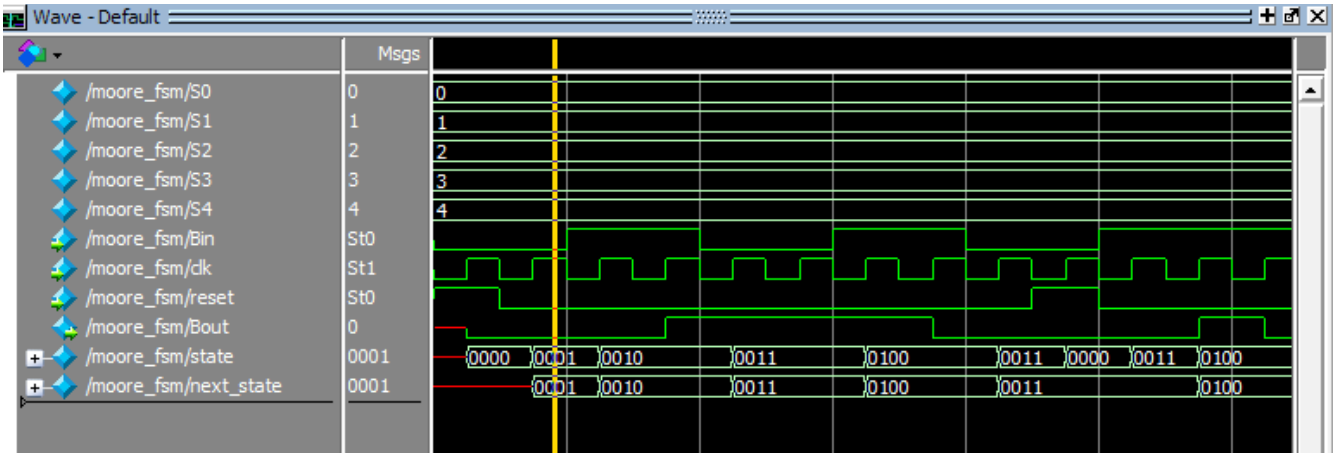


Moore Simulation

```
// Christopher Bero
```

```
// HW2 - Part 2 - Moore FSM
module moore_fsm(input Bin,clk,reset, output reg Bout);

   reg [3:0] state,next_state;

      parameter S0 = 0, S1 = 1, S2 = 2, S3 = 3, S4 = 4;

      always @ ( posedge clk)
      begin
                  case (state)

                        S0:
                        begin
                          Bout = 0;
                        if (Bin)
                              next_state = S3;
                        else
                          next_state = S1;
                        end

                        S1:
                        begin
                          Bout = 0;
                        if (Bin)
                              next_state = S2;
                        else
                          next_state = S3;
                        end

                        S2:
                        begin
                          Bout = 1;
                        if (Bin)
                              next_state = S2;
                        else
                          next_state = S3;
                        end

                        S3:
                        begin
                          Bout = 1;
                        if (Bin)
                              next_state = S4;
                        else
                          next_state = S3;
                        end

                        S4:
                        begin
                          Bout = 0;
                        if (Bin)
                              next_state = S4;
                        else
                          next_state = S3;
                        end

                  endcase
```

```
                  if (reset)
                      begin
                        state = S0;
                        Bout = 0;
                      end
                  else
                        state = next_state;

      end

endmodule



#
# Stimulus
# [0] is LSB
# SUB_ADD: 0=add (A+B), 1=subtract (A-B)
# Library: cycloneive_ver
#

#add list Bin reset clk Bout

#add wave Bin reset clk Bout

force Bin   0 0ns, 0 50ns, 1 100ns, 0 150ns, 0 200ns, 1 250ns, 0 300ns, 1 350ns, 0
400ns, 1 450ns

force clk   0 0ns, 1 25ns, 0 50ns, 1 75ns, 0 100ns, 1 125ns, 0 150ns, 1 175ns, 0
200ns, 1 225ns, 0 250ns, 1 275ns, 0 300ns, 1 325ns, 0 350ns, 1 375ns, 0 400ns, 1
425ns, 0 450ns, 1 475ns, 0 500ns, 1 525ns, 0 550ns, 1 575ns, 0 600ns, 1 625ns, 0
650ns, 1 675ns, 0 700ns, 1 725ns, 0 750ns, 1 775ns, 0 800ns, 1 825ns, 0 850ns, 1
875ns

force reset 1 0ns, 0 50ns,
      1 300ns, 0 350ns

run 1000ns


# Sample input:   01001 101
# Returned output:     01110 010

# Formatted:
# Original input:       101 10010
# 2s complement:  011 01111
# Program output:       011 01110 <- incorrect output
```


Wave - Default
Msgs
/moore_fsm/S0
/moore_fsm/S1
/moore_fsm/S2
/moore_fsm/S3
/moore_fsm/S4
/moore_fsm/Bin
/moore_fsm/clk
/moore_fsm/reset
/moore_fsm/Bout
/moore_fsm/state
/moore_fsm/next_state
0
1
2
3
4
St0
St1
St0
0
0001
0001
0000
0001
0010
0011
0100
0011
0000
0011
0100
0001
0010
0011
0100
0011
0100