Prelab

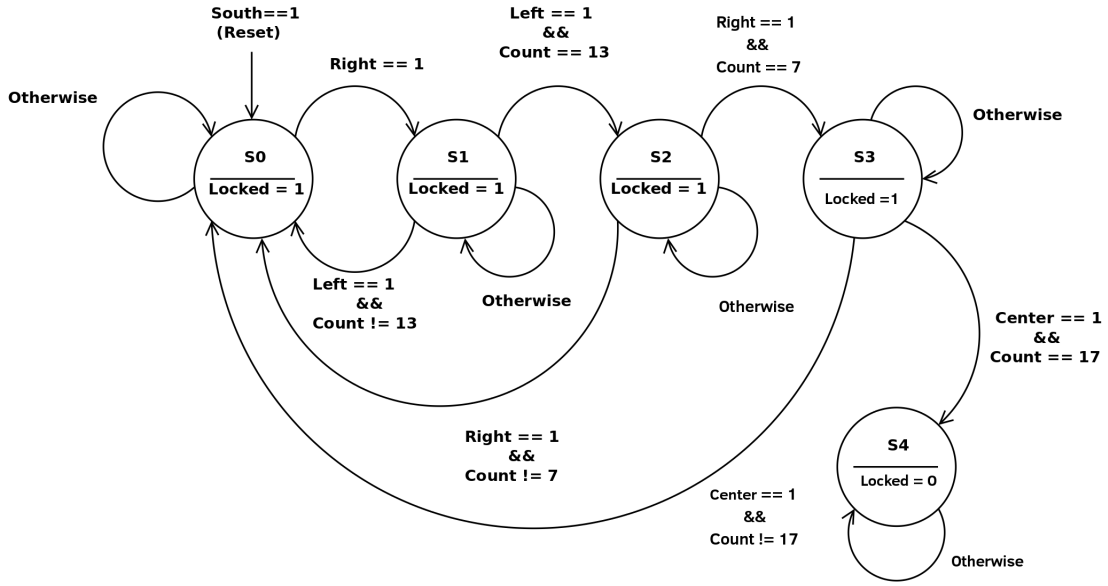

Figure 1: *State Diagram for Combination Lock*

Code Block 1: 4-Bit ALU

```
1  module combination_lock_fsm (
2    output reg [2:0] state,
3    output wire Locked, // asserted when locked
4    input wire Right, Left, // indicate direction
5    input wire [4:0] Count, // indicate position
6    input wire Center, // the unlock button
7    input wire Clk, South // clock and reset
8    );
9
10   always @ ( * ) begin
11     case (state)
12       S0: begin
13         if (condition)
14
15         else
```

```
16
17          end
18        S1: begin
19          if (condition)
20
21          else
22
23          end
24        S2: begin
25          if (condition)
26
27          else
28
29          end
30        S3: begin
31          if (condition)
32
33          else
34
35          end
36        S4: begin
37          if (condition)
38
39          else
40
41          end
42      endcase
43    end
44
45    always @ (posedge Clk) begin
46      if (Rst)
```

```
47          state <= S0;
48        else
49          state <= nextState;
50      end
51
52      if (state == S4)
53        assign Locked = 0;
54      else
55        assign Locked = 1;
56
57      assign Count = state;
58
59  endmodule // combination_lock_fsm
```