Review Problem 29

- Draw the state diagram of a machine that continuously outputs a true once at least two 0’s and at least two 1’s (in any order, not necessarily consecutively) have been seen, not including current input.
= vs. <=

- = ("Blocking") assign immediately
- <= ("Non-Blocking") first eval all righthand sides, then do all assignments simultaneously.

module swap1();
    ...
    logic [3:0] val0, val1;
    always_ff @(posedge clk) begin
        if (swap) begin
            val0 = val1;
            val1 = val0;
        end
        out = val1;
    end
endmodule

module swap2();
    ...
    logic [3:0] val0, val1;
    always_ff @(posedge clk) begin
        if (swap) begin
            val0 <= val1;
            val1 <= val0;
        end
        out <= val1;
    end
endmodule
= vs. <= in practice

- = in combinational logic: always_comb, assign
- <= in sequential: always_ff @(posedge clk)
- NEVER mix in one always block!
- Each variable written in only one always block

```vhdl
// Output logic
always_comb begin
  out = (ps == A);
end

// Next State Logic
always_comb begin
  case (ps)
    A: if (w) ns = B;
       else ns = A;
    B: if (w) ns = C;
       else ns = A;
    C: if (w) ns = C;
       else ns = A;
  endcase
end
```

// Sequential Logic
always_ff @(posedge clk) begin
  if (reset)
    ps <= A;
  else
    ps <= ns;
end

keep your always_ffs
Simple
Subdividing FSMs

- Some problems best solved with multiple pieces

- Psychic Tester:
  - Machine generates pattern of 4 values (on or off)
  - If user guesses 8 patterns in a row, they’re psychic

- States?

\[
4^1 = 4 \text{ patterns} \\
2^4 = 16 \text{ patterns}
\]

0.7 guesses = 8

\[128 + 1 = 129\]
Subdividing FSMs (cont.)

- Pieces?

1. **Next Pattern**
   - Pick a Pattern (16 states)
   - Pattern

2. **Count Correct Guesses** (9 state FSM)
   - Right?

3. **Check Guess** (Combinational)
   - User Input (4 switches + 1 go button)

   "So"