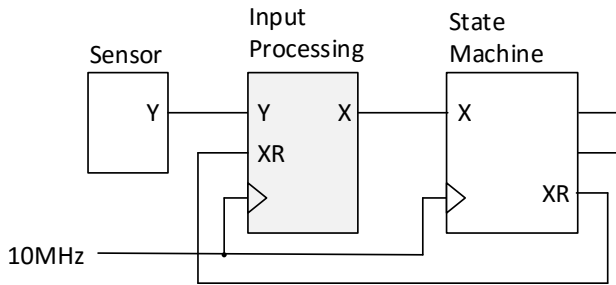


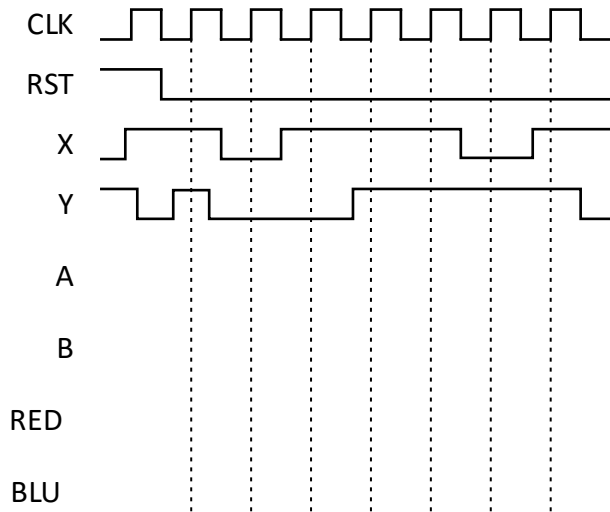
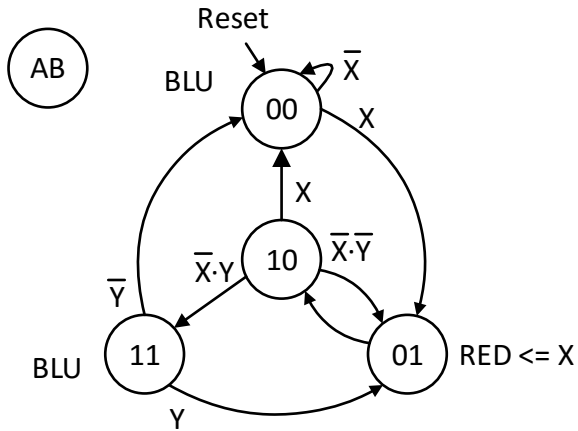
- 1. (8 points) A signal "Y" arising from a sensor can be asserted for an amount of time between 1us and 1ms. Design a state diagram for an input processing circuit that asserts an output signal "X" on the next rising edge of clock after Y has been received. X must remain asserted until after the input signal Y is no longer asserted, and until the feedback signal "XR" has been received to indicate the signal X has been processed by the state machine.



- 2. (9 points) After creating the previous state machine, you learned the sensor signal Y can bounce for up to 10us. Design a state diagram to debounce the signal in addition to meeting the previous requirements.

- 3. (3 points) The figure above shows the input processing block using the same clock as the state machine. Would you continue to use the same clock after adding the debouncing function? If not, what clock would you use?

4. (18 points) In the timing diagram below, show the time courses of the flip-flops (labeled A and B) and output signals defined by the state diagram. Then complete the following sentences.



Can RED suffer from a negative output race glitch? [ Yes / No ]  
 If you circled yes, on what transition? Leaving state \_\_\_\_ and going to state \_\_\_\_.

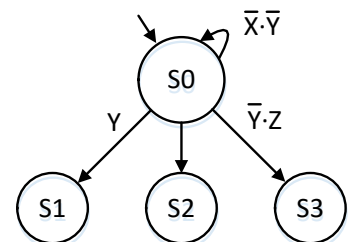
Can RED suffer from a positive output race glitch? [ Yes / No ]  
 If you circled yes, on what transition? Leaving state \_\_\_\_ and going to state \_\_\_\_.

Can BLUE suffer from a negative output race glitch? [ Yes / No ]  
 If you circled yes, on what transition? Leaving state \_\_\_\_ and going to state \_\_\_\_.

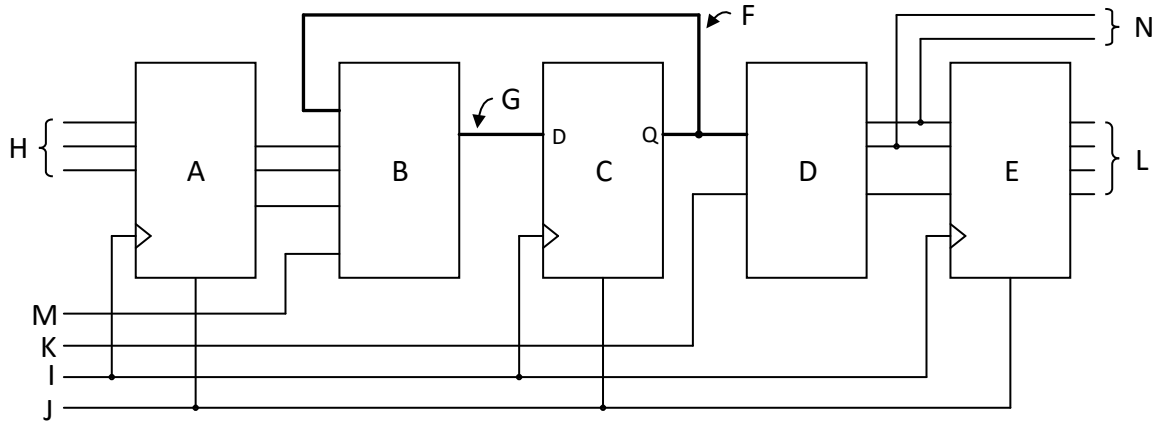
Can BLUE suffer from a positive output race glitch? [ Yes / No ]  
 If you circled yes, on what transition? Leaving state \_\_\_\_ and going to state \_\_\_\_.

5. (4 points) No branch condition is shown for the S2 branch. If some combination of branch conditions exists for which no next state is specified, then make that combination the S2 branch condition. Then, modify the hold condition so that only one next state is specified for all combinations of branch conditions. Enter the branch conditions below.

To S2: \_\_\_\_\_ Holding condition: \_\_\_\_\_



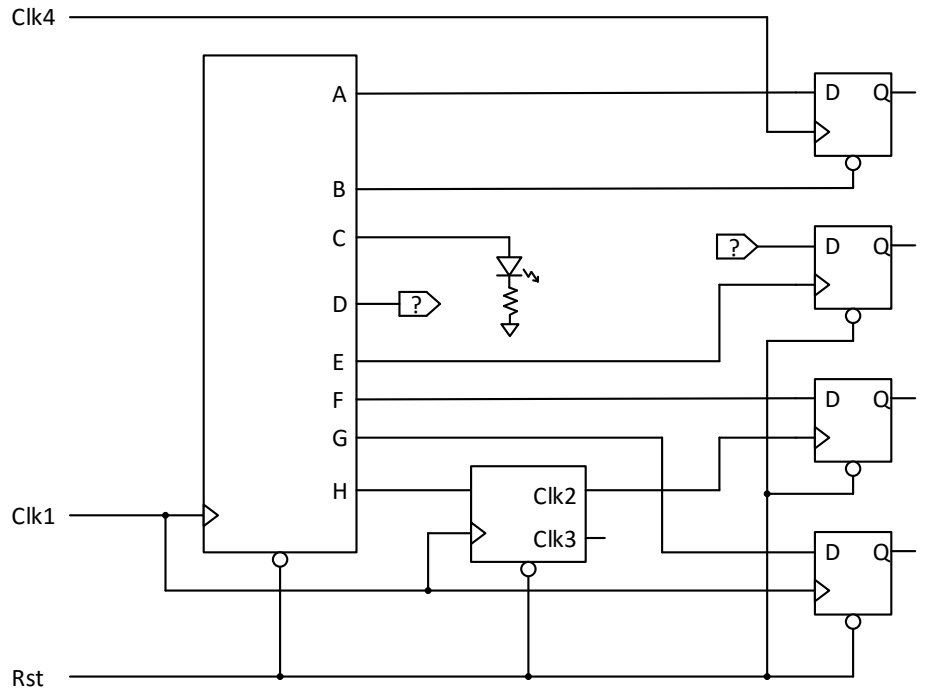
6. (23 points) Place letters in the blanks below to match the circuit blocks with the the best description. Some blanks may be left blank (because no letter identifies the block or signal), and some letters may be used more than once.



- |   |  |  |
|---|--|--|
| <input type="checkbox"/> Next state bus                 | <input type="checkbox"/> State register          | <input type="checkbox"/> Output logic                  |
| <input type="checkbox"/> UART                           | <input type="checkbox"/> Output filtering        | <input type="checkbox"/> Double-NAND stack             |
| <input type="checkbox"/> Input processing/synchronizing | <input type="checkbox"/> Next-state logic        | <input type="checkbox"/> Flux gate filter              |
| <input type="checkbox"/> Conditional ("Mealy") input    | <input type="checkbox"/> Current state bus       | <input type="checkbox"/> Clock                         |
| <input type="checkbox"/> Asynch. input – use "go/no-go" | <input type="checkbox"/> Outputs that may glitch | <input type="checkbox"/> Combinational logic block (2) |
| <input type="checkbox"/> Glitch-free outputs            | <input type="checkbox"/> Reset                   | <input type="checkbox"/> Could be a counter            |
- The number of flip-flops here determine how many states are possible  
 Glitches and hazards in this circuit can generate "logic noise" on outputs  
 This signal needs to be the lowest-skew signal, or else this circuit might not function  
 Delay here is primary determinant of overall clock speed  
 Best place for a de-metastabilizing and/or debouncing circuit  
 This circuit can eliminate/filter potential output race glitches and other logic noise  
 The binary number stored here defines the state of the state machine  
 Delay here may skew the outputs, but won't limit how fast the state machine can run

7. (5 points) If you bring asynchronous inputs into a state machine, what should you do to make sure differential delays through next state logic do not create unanticipated or incorrect branches?

8. (7 points) The figure shows a state machine with outputs going to various destinations. Circle the letter of any output that should have logic noise (glitches) removed.



9. (10 points) Sketch a block diagram, state diagram, and circuit for the Verilog code shown below. Your circuit should show a D flip-flop together with any other needed logic circuits.

```

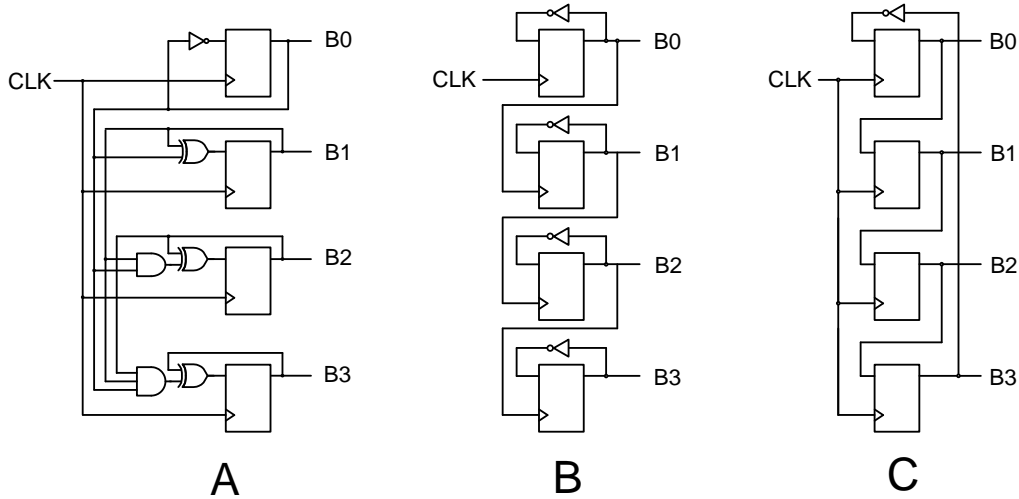
module FSM (
    input clk, rst, J, K,
    output Q);

    reg Q;

    always @ (posedge(clk))
    begin
        if (rst) Q <= 0;
        else Q = (J&~Q) + (~K&Q);
    end

endmodule
    
```

10. (10 points) The figures below show three different types of 4-bit counters. Circle the letters to indicate which counters (if any) the statements describe - you may circle 0, 1, 2, or all 3 letters for any row.



- |   |   |   |  |
|---|---|---|--|
| A | B | C | Creates $2^N$ binary numbers from N flip-flops                               |
| A | B | C | Suffers from output-bit skew problems in higher-order bits                   |
| A | B | C | Is not limited in operating frequency by next-state logic delays             |
| A | B | C | Is commonly called an Asynchronous Counter                                   |
| A | B | C | Is commonly called a Ring or Johnson Counter                                 |
| A | B | C | Is commonly called a Binary Counter  |
| A | B | C | Creates $(2 \times N)$ binary numbers from N flip-flops                      |
| A | B | C | Outputs (bit signals B0 – B3) can have glitches                              |
| A | B | C | Outputs (bit signals B0 – B3) can be used as clocks for other circuit blocks |
| A | B | C | Generates numbers in a natural counting sequence (0,1, 2, 3, etc)            |
| A | B | C | Can be used as a memory register in a state machine                          |
| A | B | C | Can be limited in operating frequency by next-state logic delays             |