From Harris & Harris Textbook: "Computer Architecture & Digital Design"

Combinational Logic

1. Draw a two-input XOR gate using only NAND gates. What is the minimum number of gates required?

2. Design a circuit that determines whether a month has 31 days based on a 4-bit input.

3. What is a tristate buffer? Explain its function and typical use cases.

4. Why are NAND gates considered universal? Justify your answer.

5. Why can a circuit's contamination delay be shorter than its propagation delay?

Sequential Logic

1. Create an FSM that detects the input sequence 1010.

2. Design a serial FSM that performs two's complement bit-by-bit.

3. What is the difference between a latch and a flip-flop? When should each be used?

4. Design an FSM that functions as a 5-bit counter.

5. Implement an edge detector that outputs HIGH on a rising edge (0 to 1 transition).

6. What is pipelining, and why is it useful in digital systems?

7. Define negative hold time in the context of flip-flops.

Timing and Metastability

1. Explain timing constraints for logic between two registers.

2. If a buffer is added to the clock input of the second flip-flop, how does that affect the setup time requirement?

3. What is metastability, and how do synchronizers mitigate its effects?

Verilog/SystemVerilog

1. What is the difference between blocking and non-blocking assignments in Verilog?

2. When should you use `always_comb` vs. `always_ff`?

3. How do you write a clean and efficient testbench in SystemVerilog?

4. Compare `case` statements with `if-else` constructs in Verilog.

5. How do you define parameterized modules in Verilog?

Microarchitecture & FPGA Design

1. What are pipeline hazards, and how can they be resolved?
2. Why don't modern CPUs use extremely deep pipelines (e.g., 100 stages)?
3. Compare cache organizations: direct-mapped, set-associative, and fully-associative.
4. What are the key differences in design approach between FPGAs and ASICs?
5. Discuss the trade-offs involved in implementing FSMs on an FPGA.

Miscellaneous Digital Design Topics

1. Compare clock gating and power gating. When is each used?
2. How does clock skew impact setup and hold timing?
3. What is the difference between synchronous and asynchronous resets?
4. How can multiplication be implemented efficiently in digital circuits?
5. What is the maximum possible result from multiplying two N-bit numbers?

Interview Questions I Was Asked (Various Companies)

Signal Processing

1. How is an analog signal converted to digital? (Cover discretization and quantization)
2. What are sampling and aliasing, both in the time and frequency domains?
3. How many bits are needed to represent a signal with N distinct values? (e.g., $\log_2(N)$)
4. Compare FIR and IIR filters: benefits, limitations, and their phase characteristics.

Digital Design

1. How can you build an OR gate using only NAND gates?
2. How do you construct a 2x1 multiplexer using basic gates?
3. Write the truth tables for JK and D flip-flops.
4. Design a D flip-flop using a JK flip-flop.
5. Design a 2-bit counter using JK flip-flops (use LSB as the clock for MSB).
6. What are setup time and hold time in sequential circuits?
7. **Bonus:** How do you derive the transfer function of an IIR filter?

Verilog

1. What is the difference between synchronous and asynchronous resets? Provide Verilog examples.

2. How do you swap two variables in Verilog with and without a temporary register?

3. Compare blocking and non-blocking assignments in Verilog.

4. What distinguishes a latch from a flip-flop?

5. Which types of Verilog code will infer a latch, and why?

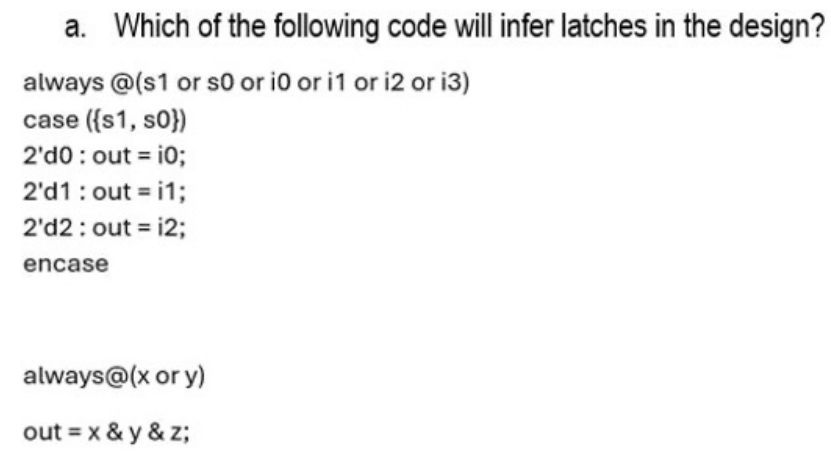


Figure 1: Latch Inference in Verilog.

6. Describe the differences between `case`, `casez`, and `casex`.

7. Determine the output of the following `casez` and `casex` statements:

Figure 2: Verilog Case Statements.

8. What is a glitch in digital circuits?

9. How would you eliminate the glitch shown in this waveform and code?

a. How do you fix the glitch in the waveform shown below?

```
module count(
input clk,
output reg equal3,
output reg [1:0] counter);

always@(posedge clk) begin
counter <= counter +1;
end

always@* begin
if (counter == 3)
        equal3 = 1;
else
        equal3 = 0;
end
endmodule
```

```
`timescale 1 ns/1 ns
module clktb();
reg clk_s;
wire [1:0]counter_s;
wire equal3_s;

count
t1(.clk(clk_s),.counter(counter_s),.eq
ual3(equal3_s));

always begin
clk_s <=0; #10;
clk_s <=1; #10;
end
endmodule
```

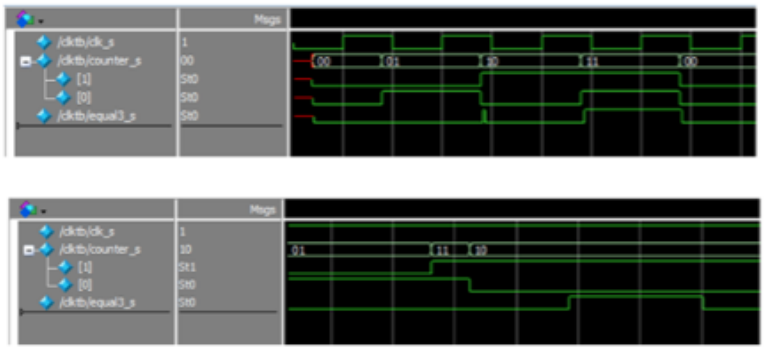


Figure 3: Glitch Waveform Example.

10. Write Verilog for a circuit with three input gates, three D flip-flops, and a single output gate:

17. Write the Verilog code for this diagram.

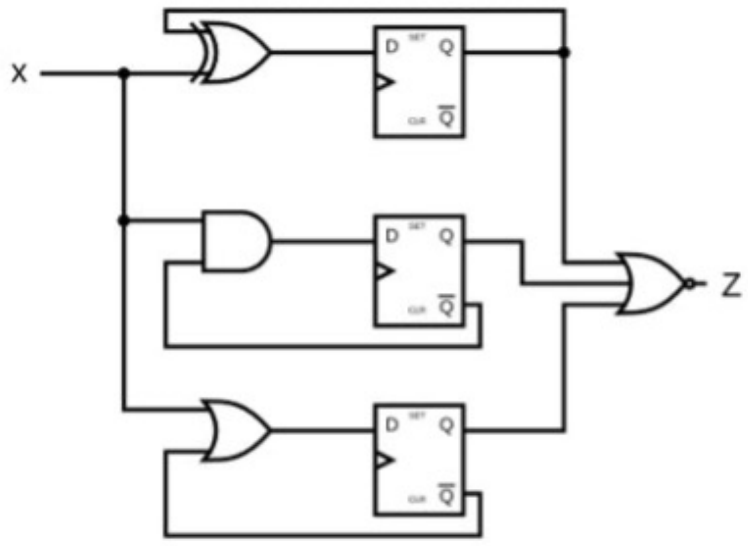


Figure 4: Circuit Diagram for Verilog Module.

11. Design an FSM that detects the sequence “011” and write the Verilog code.

12. What is the difference between inter-delay and intra-delay statements?

13. Given the code and delays below, what are the resulting logic outputs?

a. What is the output of this logic at time 0 and time 5?

```
module tb;
  reg  a, b, c, q;

  initial begin
    $monitor("[%0t] a=%0b b=%0b c=%0b q=%0b", $time, a, b, c, q);

    // Initialize all signals to 0 at time 0
    a <= 0;
    b <= 0;
    c <= 0;
    q <= 0;

    // Inter-assignment delay: Wait for #5 time units
    // and then assign a and c to 1. Note that 'a' and 'c'
    // gets updated at the end of current timestep
    #5  a <= 1;
        c <= 1;

    // Intra-assignment delay: First execute the statement
    // then wait for 5 time units and then assign the evaluated
    // value to q
    q <= #5 a & b | c;

    #20;
  end
endmodule
```

Figure 5: Verilog Delay Statements.

Digital/Physical

1. Identify and explain timing violations in the following waveform:

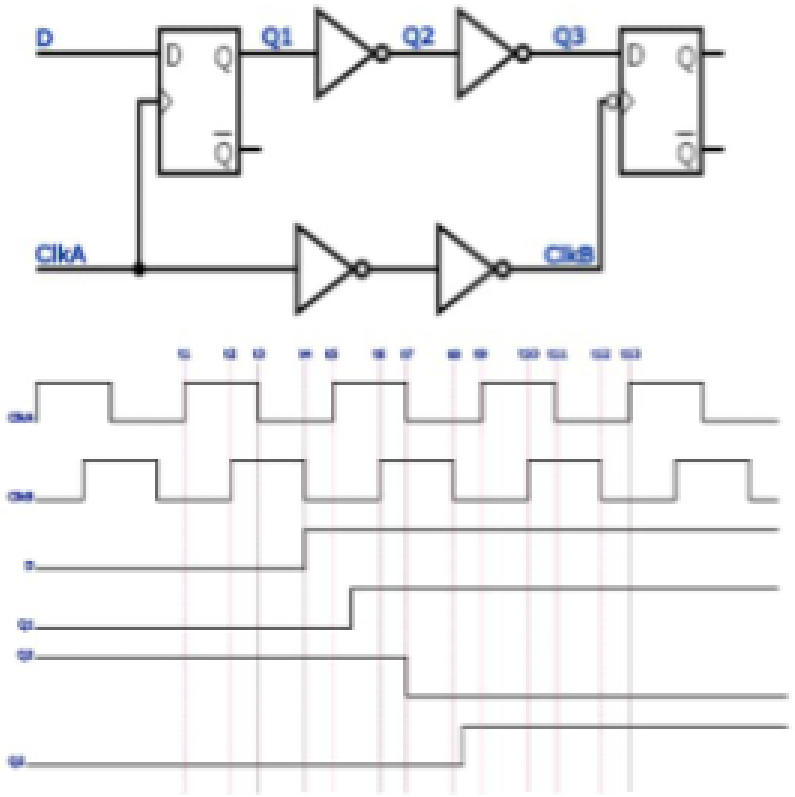
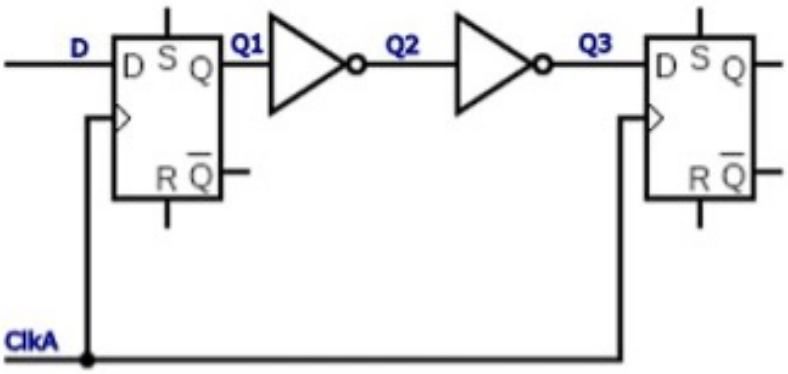


Figure 6: Timing Violation Waveform.

2. Using given setup times, calculate the minimum viable clock period:



i. $T_{Q1-Q3} = 5ns$
ii. $T_{ClkA-Q} = 1ns$
iii. $T_{Setup\ (Both\ FF)} = 2ns$

Figure 7: Timing Calculation Example.

3. Describe the design handoff from Verilog to floorplanning (e.g., 45nm project).
4. How were power (VDD) and ground (GND) nets routed in your project?
5. Explain the process and goals of clock tree synthesis.
6. What makes clock lines unique in digital layouts?

7. What are the typical challenges during place and route/layout?
8. What file types (e.g., SDC) were used in your flow?
9. Which static timing analysis tool was used to close timing?

Miscellaneous

1. Describe how an FPGA functions.
2. What is a Look-Up Table (LUT) in the context of FPGAs?
3. Recall key topics from computer architecture (e.g., jump, branch, C-level mappings).
4. What is a Phase-Locked Loop (PLL)?
5. Define clock jitter and its impact on performance.
6. Review sampling and aliasing — what issues can arise?
7. What are the Fourier and Laplace transforms used for?
8. Walk through a typical ASIC design flow and project life cycle.
9. What is the difference between DRC and LVS checks?
10. What are DACs and ADCs, and how are they applied in systems?