Definition of Set-up, Hold and Propagation in Flip-Flops

Figure 1 shows a basic diagram of a D Flip-Flop. Flip-Flops are very common elements in synchronous designs where clock signal provides the timing to various elements and clock domains. **click here if you don't see pictures**

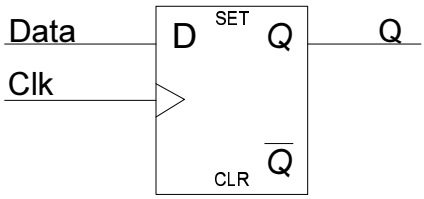


Figure Error! Bookmark not defined.**: D Flip-Flop**

Setup time and hold time describe the timing requirements on the D input of a Flip-Flop with respect to the Clk input. Setup and hold time define a window of time which the D input must be valid and stable in order to assure valid data on the Q output.

Setup Time (Tsu) – Setup time is the time that the D input must be valid before the Flip-Flop samples.

Hold Time (Th) – Hold time is the time that D input must be maintained valid after the Flip-Flop samples.

Propagation Delay (Tpd) – Propagation delay is the time that takes to the sampled D input to propagate to the Q output.

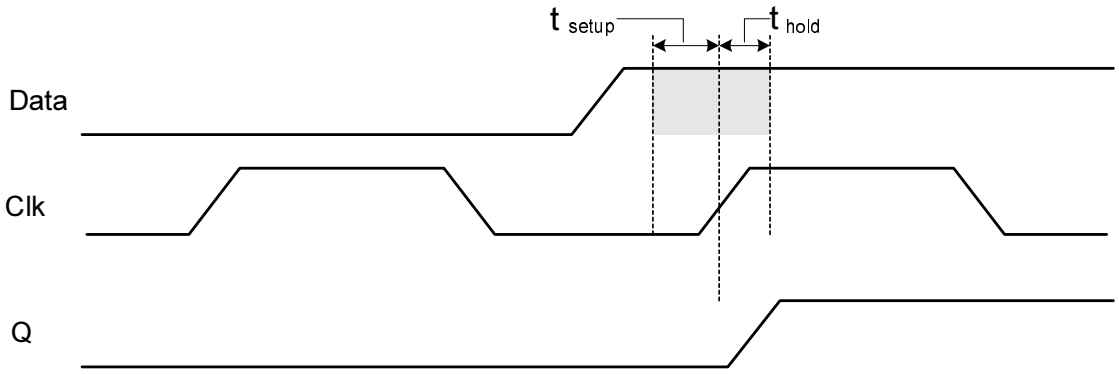


Figure Error! Bookmark not defined.**: Timing Diagram**

Simple Encryption System

The question is to design minimal hardware system, which encrypts 8-bit parallel data. A synchronized clock is provided to this system as well. The output-encrypted data should be at the same rate as the input data but no necessarily with the same phase.

The solution is presented in figure 1. **click here if you don't see pictures**

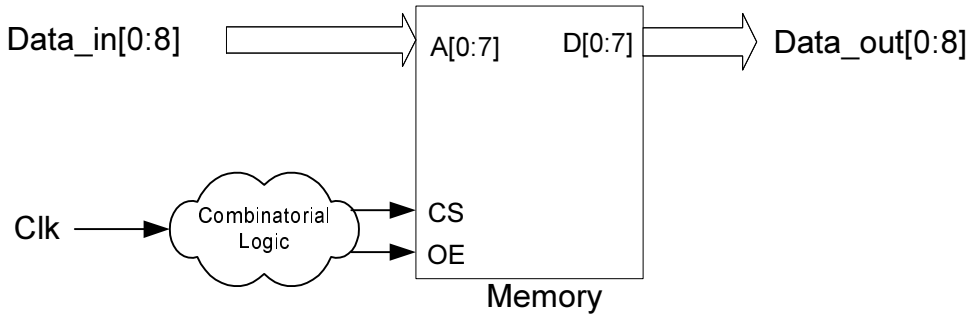


Figure Error! Bookmark not defined.**:**
Block Diagram of Encryption System

The encryption system is centered around a memory device that perform a LUT (Look-Up Table) conversion. This memory functionality can be achieved by using a PROM, EPROM, FLASH and etc. The device contains an encryption code, which may be burned into the device with an external programmer. In encryption operation, the data_in is an address pointer into a memory cell and the combinatorial logic generates the control signals. This creates a read access from the memory. Then the memory device goes to the appropriate address and outputs the associate data. This data represent the data_in after encryption.

Address	Data
01	56
02	12
03	65
04	91
⋮	⋮
FF	03

Table Error! Bookmark not defined.**:**
Example of Memory Content

Pulse Duration Extender

The question is to design a black box that receive a signal input (pulse) and multiply the duration of it by five.

Note: the longer pulse can be transmitted at any time. The length of the longer pulse may not be accurate.
click here if you don't see pictures

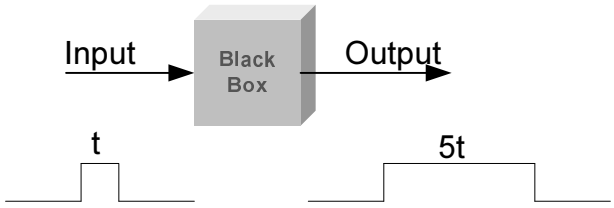


Figure Error! Bookmark not defined.**:**
Pulse Duration Extender

Figure 2 shows a general block diagram of the solution.

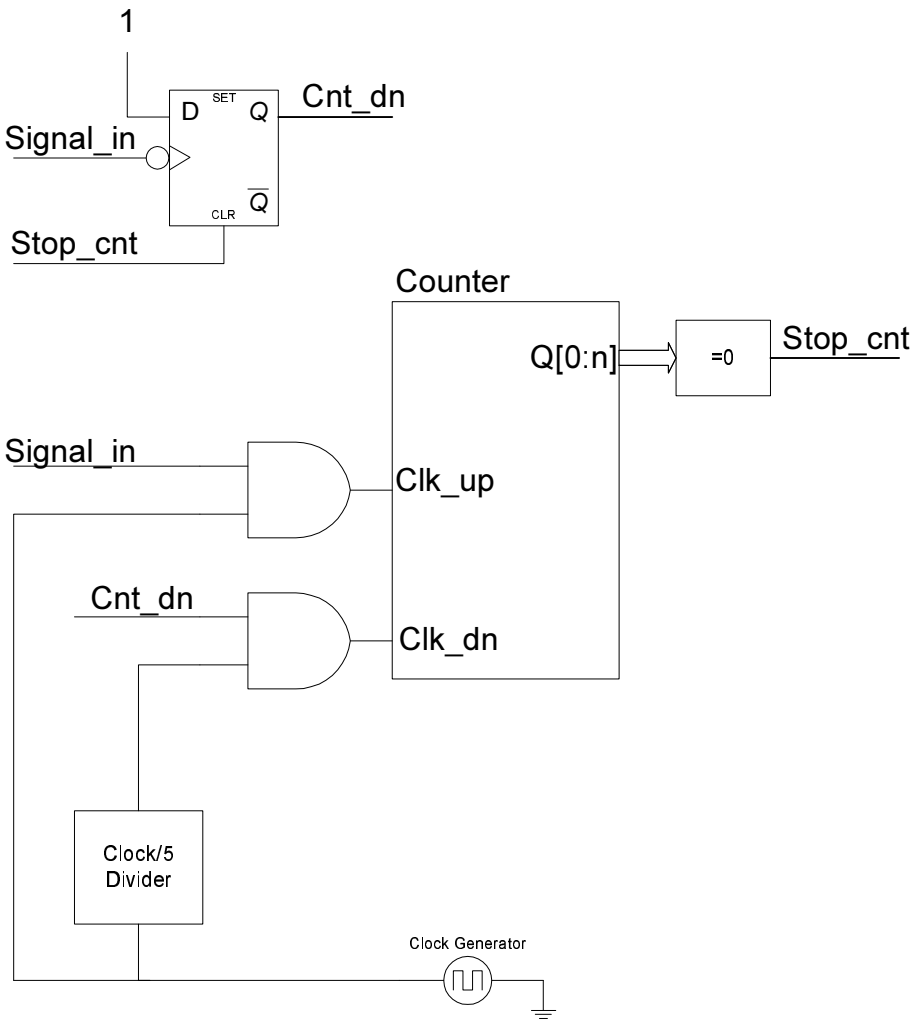


Figure Error! Bookmark not defined.**: General Block Diagram of Duration Extender**

After reset (or power-up), the counter, and Cnt_dn are de-asserted. When the Signal_in set to high (externally), the counter starts receiving clocks and counts up. The clock frequency must by higher then the Signal_in, could be about 100-1000 times to achieve good resolution on the output. When the Signal_in set to low, the counter stop counting up and the Cnt_dn set to high. Now the Clk_dn pin on the counter receives the clocks and start counting down.

When the counter stop counting up and start counting down, the value its' holds represents the number of "Clock Generator" ticks that happened when the Signal_in was high. The idea now is to count down but with slower clock, in this solution we are using a clock divider to divide the "Clock Generator" by 5. Therefore the counter will counter five times slower.

When the counter reaches zero. It means that it finishes to count down and the time passed was five times longer then the Signal_in duration. Stop_cnt create a pulse, which reset the Cnt_dn and put the system in its idle state.

The output of the system can be the Cnt_dn signal.

Note: The above describe general block diagram and general concept. The details are not mention.

Odd Number Clock Divider

This question is really common. Design a clock divider that divides by odd number. The following answer shows how to design a divider by 3 which is asymmetrical. The trivial way is to use a state-machine concept; therefore the answer explains state-machine design step-by-step, from functional specifications towards a complete detailed design and implementation.

A functional description of the asymmetrical divider is shown in Figure 1. **click here if you don't see pictures**

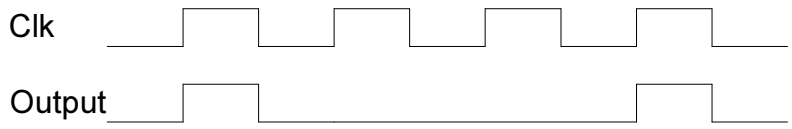


Figure Error! Bookmark not defined.**:**
Functional Description of the Divider

The first step is to draw a state diagram that describes the logical behavior of the circuit. Figure 2 introduces the state diagram of the divider. We can easily see that the divider consist of 3 states which means 2 Flip-Flops. Each step is done every clock cycle.

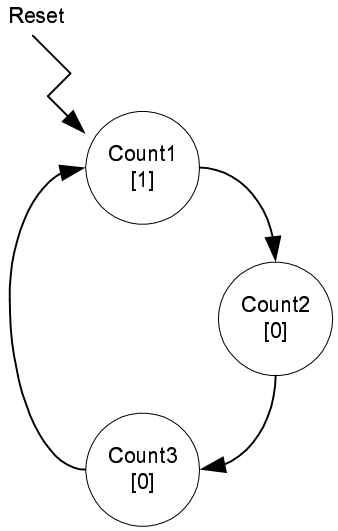


Figure Error! Bookmark not defined.**:**
State diagram of the Divider

We name the state with a unique name and define the outputs in the square brackets. Whenever the state-machine is in Count1, the output shall be 1. Whenever the state-machine is in Count2 or Count3, the output shall be 0.

After obtaining the state diagram it is possible to describe the design with a symbolic state transition table. In this step we put all the information we gathered as shown in the following table.

Present State	Next Step	Output
Count1	*Count2*	1
Count2	*Count3*	0
Count3	*Count1*	0

Table Error! Bookmark not defined.**:**
Symbolic State Transition Table

The next step is to go into details. We have 2 Flip-Flops and one output. This information is entered into an encoded state transition table. The functions can be extracted from a Karnaugh map, or in this case, use the table as a truth table.

Present State		Next Step		Output
Q0	***Q1***	***D0***	***D1***	
0	0	0	1	1
0	1	1	0	0
1	0	0	0	0

Table Error! Bookmark not defined.**:**
Encoded State Transition Table

We can write the functions as:

- D0 = Q1
- D1 = NOT(Q0+Q1)
- OUT = D1

The implement of the divider by 3 is shown in Figure 3. The output can be connected to Q1 pin.

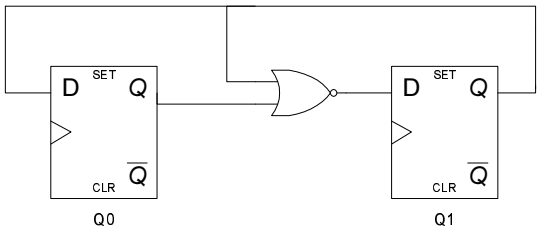


Figure Error! Bookmark not defined.**:**
Implementation of the Divider

Comments and suggestions: interview@hardware-guru.com

Digital One Shot

This "one-shot" shall produce a single output pulse for any long pulse in the input. The length of the output pulse shall be one clock cycle. Assume that the input pulse is as you see in the following figure. **click here if you don't see pictures**

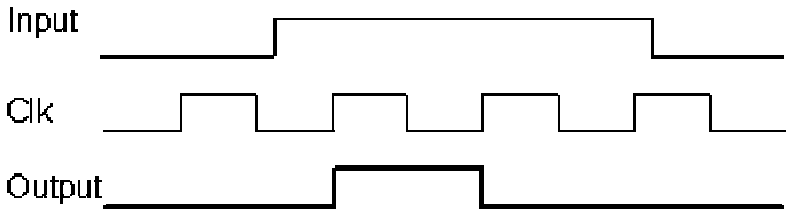


Figure Error! Bookmark not defined.**:**
One Shot Timing Diagram

The answer is showed in Figure 2. It's based on two flip-flops, which create a delay on the signal input. Then the result of the outputs (*Q0, Q1*) are logically AND, and output the result.

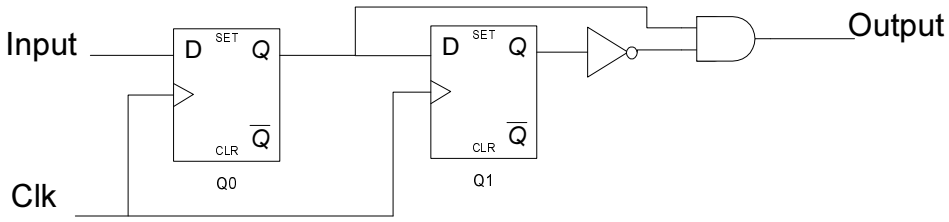


Figure Error! Bookmark not defined.**:**
One Shot Schematics

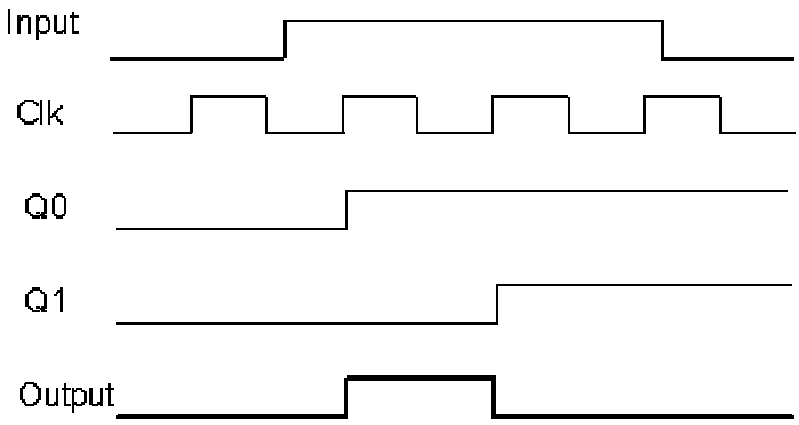


Figure Error! Bookmark not defined.**:**
One Shot Detailed Timing Diagram

This is a simplified design and thus has some problems (hint: asynchronous input). Please write to us with your improvement ideas and we will update the entire solution with your inputs.

Comments and suggestions: *interview@hardware-guru.com*

The following are some of the questions I was asked in my interviews. The questions of course, depend on the position *you* are being interviewed and also on *your* Resume. So if you find any questions *not* relevant to your Resume, you can safely ignore them. Also, these questions are limited to VLSI Design, Computer Architeture and some basic Programming. If you are looking for something in Analog, RF etc, this is NOT the place.

Okay alright...that makes sense...now lets get going...

VLSI Design:

1) Explain why & how a MOSFET works

2) Draw Vds-Ids curve for a MOSFET. Now, show how this curve changes (a) with increasing Vgs (b) with increasing transistor width (c) considering Channel Length Modulation

3) Explain the various MOSFET Capacitances & their significance

4) Draw a CMOS Inverter. Explain its transfer characteristics

5) Explain sizing of the inverter

6)How do you size NMOS and PMOS transistors to increase the threshold voltage?

7) What is Noise Margin? Explain the procedure to determine Noise Margin

8) Give the expression for CMOS switching power dissipation

9) What is Body Effect?

10) Describe the various effects of scaling

11) Give the expression for calculating Delay in CMOS circuit

12) What happens to delay if you increase load capacitance?

13) What happens to delay if we include a resistance at the output of a CMOS circuit?

14) What are the limitations in increasing the power supply to reduce delay?

15) How does Resistance of the metal lines vary with increasing thickness and increasing length?

16) You have three adjacent parallel metal lines. Two out of phase signals pass through the outer two metal lines. Draw the waveforms in the center metal line due to interference. Now, draw the signals if the signals in outer metal lines are in phase with each other

17) What happens if we increase the number of contacts or via from one metal layer to the next?

18) Draw a transistor level two input NAND gate. Explain its sizing (a) considering Vth (b) for equal rise and fall times

19) Let A & B be two inputs of the NAND gate. Say signal A arrives at the NAND gate later than signal B. To optimize delay, of the two series NMOS inputs A & B, which one would you place near the output?

20) Draw the stick diagram of a NOR gate. Optimize it

21) For CMOS logic, give the various techniques you know to minimize power consumption

22) What is Charge Sharing? Explain the Charge Sharing problem while sampling data from a Bus

23) Why do we gradually increase the size of inverters in buffer design? Why not give the output of a circuit to one large inverter?

24) In the design of a large inverter, why do we prefer to connect small transistors in parallel (thus increasing effective width) rather than lay out one transistor with large width?

25) Given a layout, draw its transistor level circuit. (I was given a 3 input AND gate and a 2 input Multiplexer. You can expect any simple 2 or 3 input gates)

26) Give the logic expression for an AOI gate. Draw its transistor level equivalent. Draw its stick diagram

27) Why don't we use just one NMOS or PMOS transistor as a transmission gate?

28) For a NMOS transistor acting as a pass transistor, say the gate is connected to VDD, give the output for a square pulse input going from 0 to VDD

29) Draw a 6-T SRAM Cell and explain the Read and Write operations

30) Draw the Differential Sense Amplifier and explain its working. Any idea how to size this circuit? (Consider Channel Length Modulation)

31) What happens if we use an Inverter instead of the Differential Sense Amplifier?

32) Draw the SRAM Write Circuitry

33) Approximately, what were the sizes of your transistors in the SRAM cell? How did you arrive at those sizes?

34) How does the size of PMOS Pull Up transistors (for bit & bit- lines) affect SRAM's performance?

35) What's the critical path in a SRAM?

36) Draw the timing diagram for a SRAM Read. What happens if we delay the enabling of Clock signal?

37) Give a big picture of the entire SRAM Layout showing your placements of SRAM Cells, Row Decoders, Column Decoders, Read Circuit, Write Circuit and Buffers

38) In a SRAM layout, which metal layers would you prefer for Word Lines and Bit Lines? Why?

39) How can you model a SRAM at RTL Level?

40) What's the difference between Testing & Verification?

41) For an AND-OR implementation of a two input Mux, how do you test for Stuck-At-0 and Stuck-At-1 faults at the internal nodes? (You can expect a circuit with some redundant logic)

42) What is Latch Up? Explain Latch Up with cross section of a CMOS Inverter. How do you avoid Latch Up?

Digital Design:

1) Give two ways of converting a two input NAND gate to an inverter

2) Given a circuit, draw its exact timing response. (I was given a Pseudo Random Signal Generator; you can expect any sequential ckt)

3) What are set up time & hold time constraints? What do they signify? Which one is critical for estimating maximum clock frequency of a circuit?

4) Give a circuit to divide frequency of clock cycle by two

5) Design a divide-by-3 sequential circuit with 50% duty circle. (Hint: Double the Clock)

6) Suppose you have a combinational circuit between two registers driven by a clock. What will you do if the delay of the combinational circuit is greater than your clock signal? (You can't resize the combinational circuit transistors)

7) The answer to the above question is breaking the combinational circuit and pipelining it. What will be affected if you do this?

8) What are the different Adder circuits you studied?

9) Give the truth table for a Half Adder. Give a gate level implementation of the same.

10) Draw a Transmission Gate-based D-Latch.

11) Design a Transmission Gate based XOR. Now, how do you convert it to XNOR? (Without inverting the output)

12) How do you detect if two 8-bit signals are same?

13) How do you detect a sequence of "1101" arriving serially from a signal line?

14) Design any FSM in VHDL or Verilog.

Computer Architecture:

1) What is pipelining?

2) What are the five stages in a DLX pipeline?

3) For a pipeline with 'n' stages, what's the ideal throughput? What prevents us from achieving this ideal throughput?

4) What are the different hazards? How do you avoid them?

5) Instead of just 5-8 pipe stages why not have, say, a pipeline with 50 pipe stages?

6) What are Branch Prediction and Branch Target Buffers?

7) How do you handle precise exceptions or interrupts?

8) What is a cache?

9) What's the difference between Write-Through and Write-Back Caches? Explain advantages and disadvantages of each.

10) Cache Size is 64KB, Block size is 32B and the cache is Two-Way Set Associative. For a 32-bit physical address, give the division between Block Offset, Index and Tag.

11) What is Virtual Memory?

12) What is Cache Coherency?

13) What is MESI?

14) What is a Snooping cache?

15) What are the components in a Microprocessor?

16) What is ACBF(Hex) divided by 16?

17) Convert 65(Hex) to Binary

18) Convert a number to its two's compliment and back

19) The CPU is busy but you want to stop and do some other task. How do you do it?

C/C++, Perl & Unix:

1) How would you decide weather to use C, C++ or Perl for a particular project?

2) What are pointers? Why do we use them?

3) What are the benefits of having Global & Local Variables?

4) What is 'malloc'? Why do we need to use it?

5) Write a C program to compare two arrays and write the common elements in another array

6) Write a function in C to accept two integers and return the bigger integer

7) What are the advantages of C over Perl and vice versa?

8) What does '@' and '&' mean in Perl?

9) What is a 'Package' in Perl?

10) What are Perl Regular Expressions?

11) Perl Regular Expressions are greedy. What does that mean?

12) What are Associative arrays in Perl?

13) Suppose a Perl variable has your name stored in it. Now, how can you define an array by the name? (i.e., you have $a="Adarsh"; now you want @Adarsh=[.....])

14) Write a Perl script to parse a particular txt file and output to another file in a desired format. (You can expect the file to have some data arranged rows & columns)

15) Suppose you have the outputs of a test program in some big test file. In Perl, how can you test if all the outputs match a particular string?

16) What are Data Abstraction and Data Encapsulation?

17) Explain Friend Functions and Polymorphism with examples

18) Commands for changing directory, making directory, going up one directory, knowing the file permissions and changing file permissions.

19) How do you search for a particular string in all the text files in current directory from command line?

20) How do you sort a file alphabetically from command line?

Other Simple Questions:

1) What is j to the power j?

2) What is Normal Distribution? Where is the Mean and Median on the graph for Normal Distribution?

3) Draw a simple RC-Low pass circuit.

Some General Questions:

1) Tell me something about yourself and your interests

2) Tell me something about some problems you faced in a project and how did you handle it?

3) Give one instance where you were criticised by your Professor

4) Where do you see yourself five years from now?

5) What salary are you expecting?

6) Any Questions for me regarding the position or the company?

7) Finally, does this position sound interesting? :-)

Frequently Asked Interview Questions

I gathered these questions from several emails, sent to me by students who attended on-site interviews at varios different companies. I shall try to add more of them in near future.

1. What is the difference between a latch and a flip flop. For the same input, how
would the output look for a latch and for a flip-flop.

2. Finite state machines:
(2.1)Design a state-machine (or draw a state-diagram) to give an output '1' when the # of A's are even
and # of B's are odd. The input is in the form of a serial-stream (one-bit per clock cycle). The inputs could be of the type A, B or C. At any given clock cycle, the output is a '1', provided the # of A's are even and # of B's are odd. At any given clock cycle, the output is a '0', if the above condition is not satisfied.

(2.2). To detect the sequence "abca" when the inputs can be a b c d.

3. minimize a boolean expression.

4. Draw transistor level nand gate.

5. Draw the cross-section of a CMOS inverter.

6. Deriving the vectors for the stuck at 0 and stuck at 1 faults.

7. Given a boolean expression he asked me to implement just with muxes but nothing else.

8. Draw Id Vds curves for mosfets and explain different regions.

9. Given the transfer characteristics of a black box draw the circuit for the black box.

10. Given a circuit and its inputs draw the outputs exact to the timing.

11. Given an inverter with a particular timing derive an inverter using the previous one but with the required timing other than the previous one.

12. Change the rise time and fall time of a given circuit by not changing the transistor sizes but by using current mirrors.

13. Some problems on clamping diodes.

These are some of the questions asked by Microsoft.
(I feel that these type of questions are asked even in Electrical Engineering interviews. Make sure you browse them.)

1. Given a rectangular (cuboidal for the puritans) cake with a rectangular piece removed (any size or orientation), how would you cut the remainder of the cake into two equal halves with one straight cut of a knife ?

2. You're given an array containing both positive and negative integers and required to find the sub-array with the largest sum (O(N) a la KBL). Write a routine in C for the above.

3. Given an array of size N in which every number is between 1 and N, determine if there are any duplicates in it. You are allowed to destroy the array if you like.

4. Write a routine to draw a circle (x ** 2 + y ** 2 = r ** 2) without making use of any floating point computations at all.

5. Given only putchar (no sprintf, itoa, etc.) write a routine putlon the prints out an unsigned long in decimal.

6. Give a one-line C expression to test whether a number is a power of 2. [No loops allowed - it's a simple test.]

7. Given an array of characters which form a sentence of words, give an efficient algorithm to reverse the order of the words (no characters) in it.

8. How many points are there on the globe where by walking one mile south, one mile east and one mile north you reach the place where you started.

9. Give a very good method to count the number of ones in a 32 bit number. (caution: looping through testing each bit is not a solution)

10. What are the different ways to say, the value of x can be either a 0 or a 1. Apparently the if then else solution has a jump when written

```
out in assembly.
if (x == 0)
  y=0
else
  y =x
```

There is a logical, arithmetic and a datastructure soln to the above problem.

Logic design:

1. Draw the transistor level CMOS # input NAND or NOR gate.
After drawing it lot of qestions on that ckt will be asked.

2. Transistor sizing for given rise time and fall time. How do you size it for equal rise and fall time.

3. Given a function whose inputs are dependent on its outputs. Design a sequential circuit.

4. Design a finite state machine to give a modulo 3 counter when x=0 and modulo 4 counter when x=1.

5. Given a boolean equation minimize it.

6. Given a boolean equation draw the transistor level minimum transistor circuit.

7. What is the function of a D-flipflop, whose inverted outputs are connected to its input ?

8. What will you do if you want to drive a large capacitance ?

Layout related questions:

1. asked me to layout the 3 input nand gate.

2. Later he asked me to modify it to consume as much less space as we can.

3. He also asked me about the transistor sizing.

---

1. He asked me to draw the cross section of an inverter and asked me to show all the capacitances on it and reason for those capacitances.

2. Describe the latchup effect.

3. More about the tristate buffers.

3. What will be the voltage at the output node of a triostate buffer in its high impedence state. He gave a waveform for the input and asked me to draw the output waveform for that.

4. Posed a lot of questions on charge sharing problems and keeper circuits.

5. Asked me to draw the I d Vds curves for mosfet. Asked me to explain the regions and some couses for that curve like channel width modulation.

6. He asked me about the electron migration effect and methods to avoid it.

7. Asked me to draw the dynamic logic of a particular gate and then posed lots of tricky questions from the previous discussion.

8. He asked me to draw the 6 transistor contemporary sram cell and asked me to explain how the reading and writing is done in it.

9. Something about trip point.

Computer Architecture Questions:

1. Explain what is DMA?
2. what is pipelining?
3. what are superscalar machines and vliw machines?
4. what is cache?
5. what is cache coherency and how is it eliminated?
6. what is write back and write through caches?
7. what are different pipelining hazards and how are they eliminated.
8. what are different stages of a pipe?
9. eplain more about branch prediction in controlling the control hazards
10. Give examples of data hazards with pseudo codes.
11. Caluculating the number of sets given its way and size in a cache?
12. How is a block found in a cache?
13. scoreboard analysis.
14. What is miss penalty and give your own ideas to eliminate it.
15. How do you improve the cache performance.
16. Different addressing modes.
17. Computer arithmetic with two's complements.
18. About hardware and software interrupts.
19. What is bus contention and how do you eliminate it.

20. What is aliasing?
21) What is the difference between a latch and a flip flop?
22) What is the race around condition? How can it be overcome?
23) What is the purpose of cache? How is it used?
24) What are the types of memory management?

Introduction :

A fresh graduate faces some tough questions in his first job interview. The questions themselves are simple but require practical and innovative approach to solve them. I started collecting some questions from my own experience and from my friends. Answers to most questions are not given. Spend some time to solve these and let me know if you have some more interesting ones.

Please do not send me emails asking for solutions. You are not supposed to answer these questions in 10 seconds like some university multiple choice questions. Some questions may have more than correct answers and some may not even have correct answer :)

What matters is your approach to solution and understanding of basic hardware design principles.

Recently added questions

Q. Design a logic which mimics a infinite width register. It takes input serially 1 bit at a time. Output is asserted high when this register holds a value which is divisible by 5.

For example:

Input	Sequence	Value	Output
1	1	1	0
0	10	2	0
1	101	5	1
0	1010	10	1
1	10101	21	0

(Hint: Use a FSM to create this)

Q.Design a block which has 3 inputs as followed.

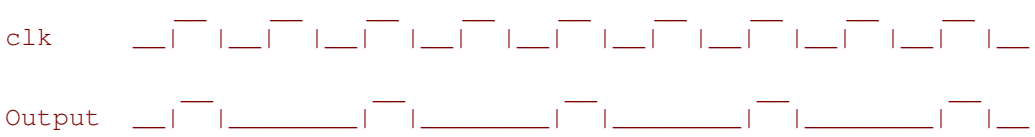
1. system clock of pretty high freq
2. asynch clock input P
3. asynch clock input Q

P and Q clocks have 50% duty cycle each. Their frequencies are close enough and they have phase difference. Design the block to generate these outputs.

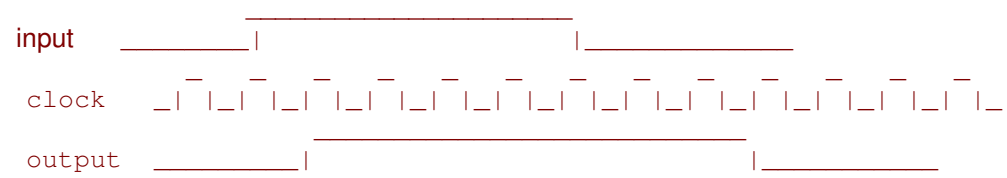
1. PeqQ : goes high if periods of P and Q are same
2. PleQ : goes high if P's period is less than that of Q.
3. PgrQ : goes high if P's period is greater than that of Q.

Q. What's the difference between a latch and a flip-flop? Write Verilog RTL code for each. (This is one of the most common questions but still some EE's don't know how to explain it correctly!)

Q. Design a black box whose input clock and output relationship as shown in diagram.



Q. Design a digital circuit to delay the negative edge of the input signal by 2 clock cycles.

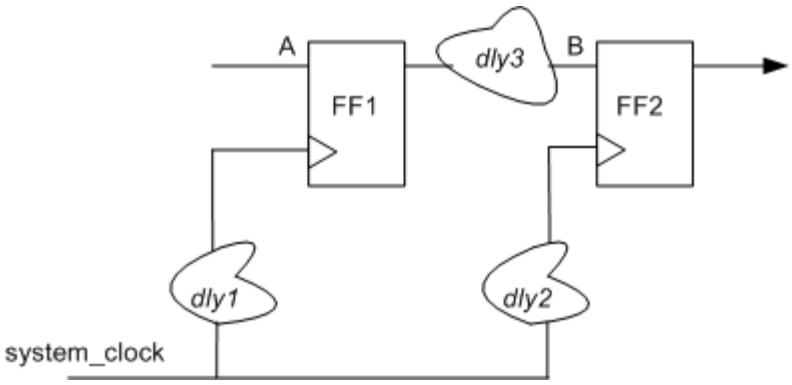


Q. Design a Pattern matching block

- Output is asserted if pattern "101" is detected in last 4 inputs.
- How will you modify this design if it is required to detect same "101" pattern anywhere in last 8 samples?

Questions:

Q1.

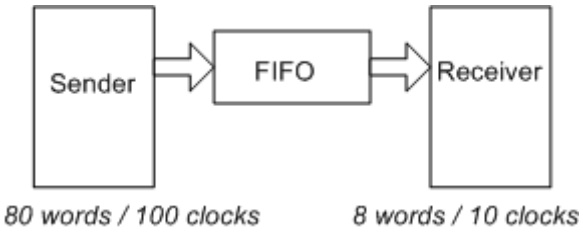


The digital circuit is shown with logic delay (dly3) and two clock buffer delays (dly1, dly2).

- How will you fix setup timing violations occurring at pin B?
- How will you fix hold violations occurring at pin B?

(Hint: Change the values of three delays to get desired effect)

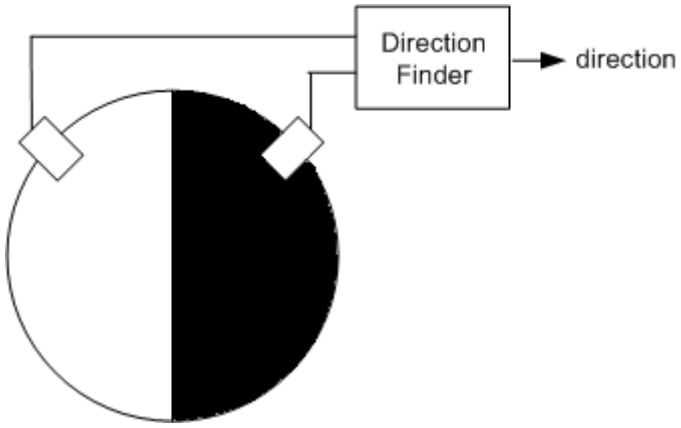
Q2.



Sender sends data at the rate of 80 words / 100 clocks
Receiver can consume at the rate of 8 words / 10 clocks

Calculate the depth of FIFO so that no data is dropped.
Assumptions: There is no feedback or handshake mechanism. Occurrence of data in that time period is guaranteed but exact place in those clock cycles is indeterminate.

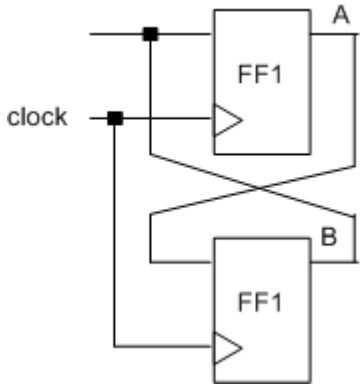
Q3



Optical sensors A and B are positioned at 90 degrees to each other as shown in Figure. Half od the disc is white and remaining is black. When black portion is under sensor it generates logic 0 and logic 1 when white portion is under sensor.

Design Direction finder block using digital components (flip flops and gates) to indicate speed. Logic 0 for clockwise and Logic 1 for counter clockwise.

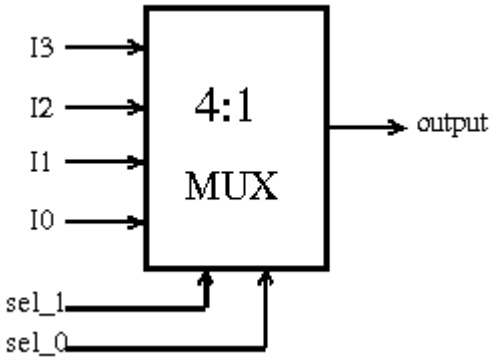
Q4



Will this design work satisfactorily?
Assumptions: thold = tsetup = tclock_out = tclock_skew = 1ns.
After reset A = 0, B = 1

---

1. Design a 4:1 mux in Verilog.



- Multiple styles of coding. e.g.
  Using **if-else** statements

```
if(sel_1 == 0 && sel_0 == 0) output = I0;
else if(sel_1 == 0 && sel_0 == 1) output = I1;
else if(sel_1 == 1 && sel_0 == 0) output = I2;
else if(sel_1 == 1 && sel_0 == 1) output = I3;
```

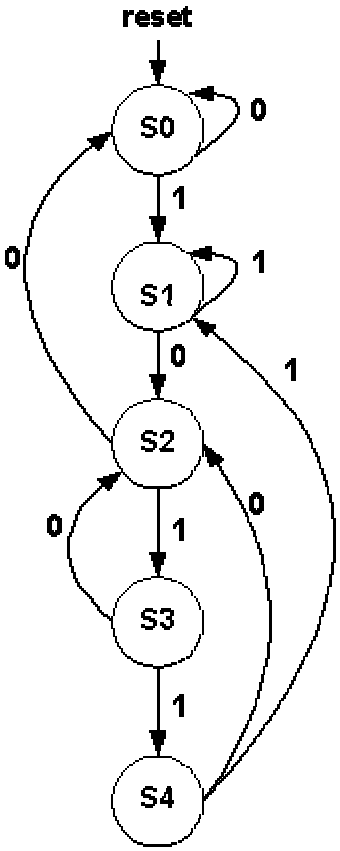
  Using **case** statement

```
case ({sel_1, sel_0})
     00 : output = I0;
     01 : output = I1;
     10 : output = I2;
     11 : output = I3;
     default : output = I0;
endcase
```

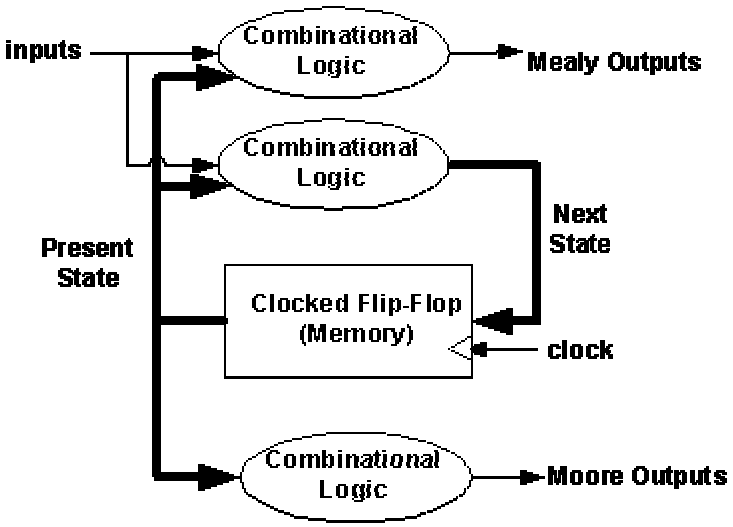
- What are the advantages / disadvantages of each coding style shown above?
- How Synthesis tool will give result for above codes?
- What happens if default statement is removed in case statement?
- What happens if combination 11 and default statement is removed? (Hint Latch inference)
  (Comments : Though this questions looks simple and out of text books, the answers to supporting questions can come only after some experience / experimentation.)

---

2. Design a FSM (Finite State Machine) to detect a sequence 10110.



State Transition Diagram



FSM Block Diagram

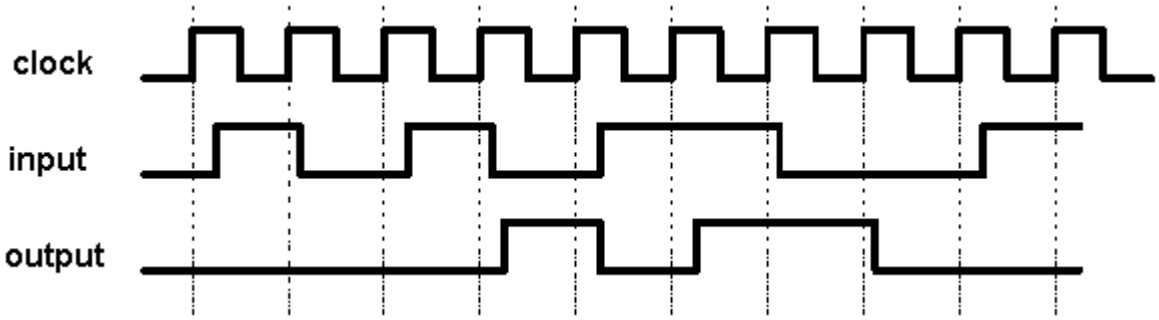
- Have a good approach to solve the design problem.
- Know the difference between Mealy, Moore, 1-Hot type of state encoding.
- Each state should have output transitions for all combinations of inputs.
- All states make transition to appropriate states and not to default if sequence is broken. e.g. S3 makes transition to S2 in example shown.
- Take help of FSM block diagram to write Verilog code.

---

3. One more sequence detector:

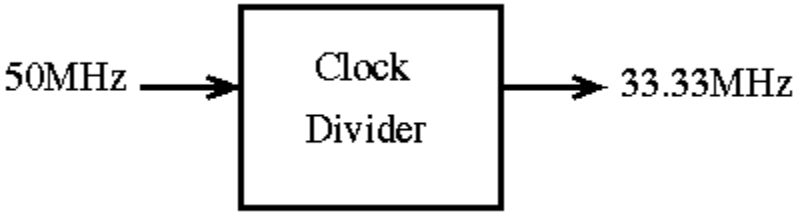
Design a FSM (Finite State Machine) to detect more than one "1"s in last 3 samples.
For example: If the input sampled at clock edges is 0 1 0 1 0 1 1 0 0 1
then output should be 0 0 0 1 0 1 1 1 0 0 as shown in timing diagram.

And yes, you have to design this FSM using not more than 4 states!!



---

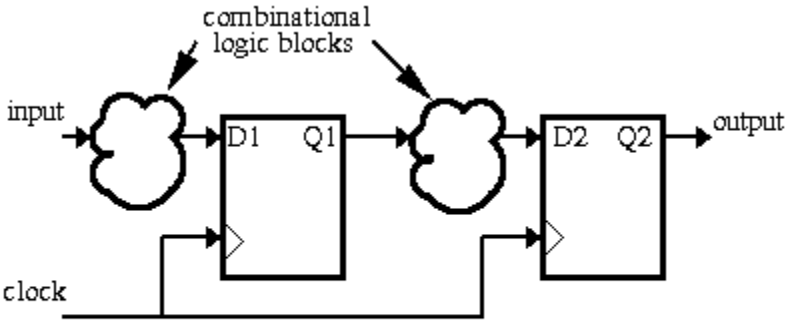
4. Design a state machine to divide the clock by 3/2.



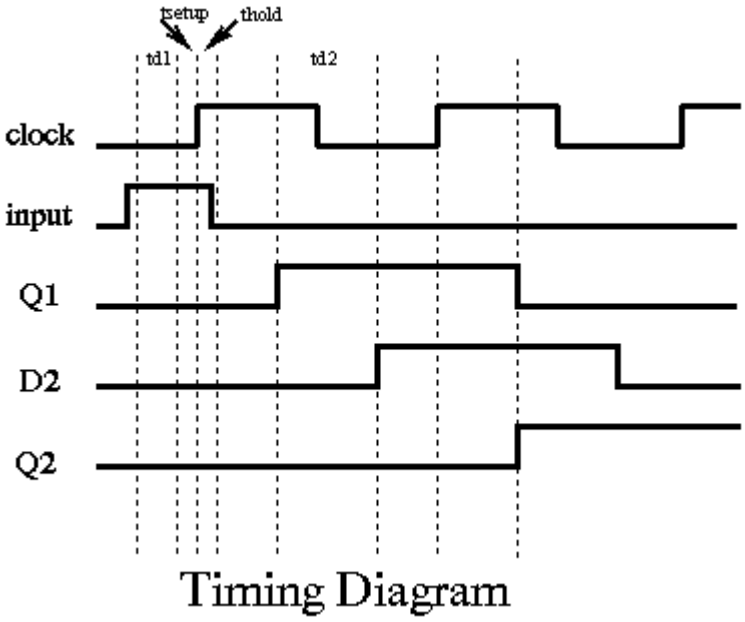
(Hint: 2 FSMs working on posedge and negedge)

---

5. Draw timing diagrams for following circuit.



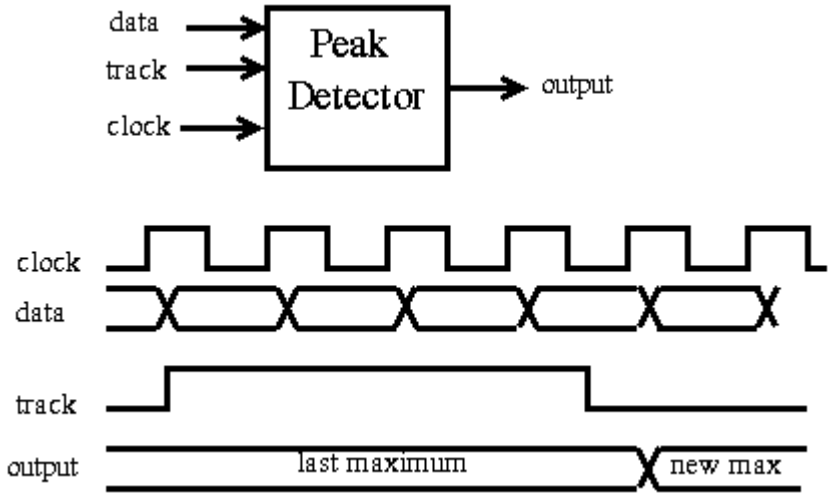
Schematic of two cascaded flip–flops



- What is the maximum frequency at which this circuit can operate?
- What is the minimum width of input pulse and position?
- Problem can be given interesting twist by specifying all delays in min and max types.

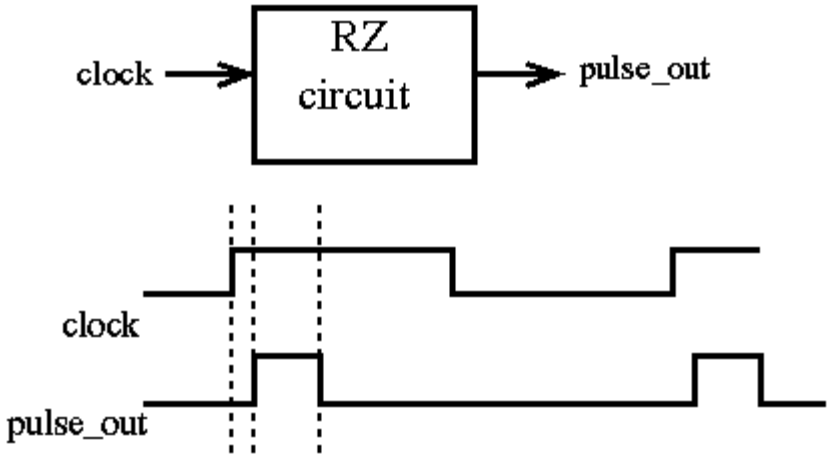
---

6. Design a Digital Peak Detector in Verilog.



---

7. Design a RZ (return to zero)circuit. Design a clock to pulse circuit in Verilog / hardware gates.



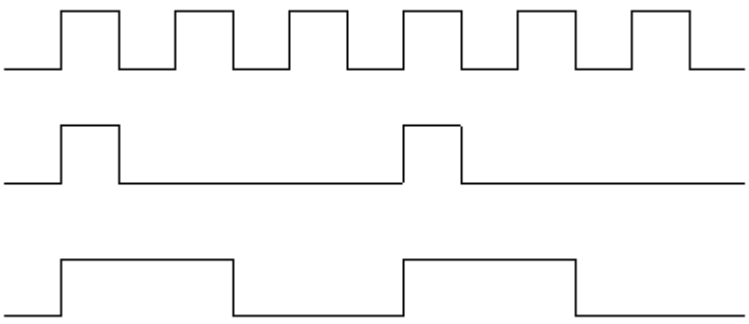
---

8. Miscellaneous Basic Verilog Questions:

- What is the difference between Behavior modeling and RTL modeling?
- What is the benefit of using Behavior modeling style over RTL modeling?
- What is the difference between blocking assignments and non-blocking assignments ?
- How do you implement the bi-directional ports in Verilog HDL
- How to model inertial and transport delay using Verilog?
- How to synchronize control signals and data between two different clock domains?

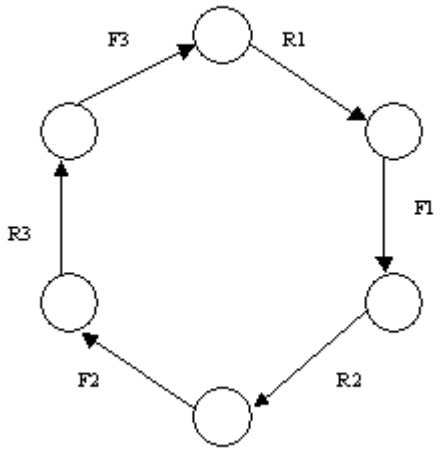
Q: how to design a divide-by-3 counter with equal duty cycle ?

Here is one of the solutions...
Start with the timing diagram. It shows an input clock, an output of a regular divide-by-3 counter and an output of divide-by-3 counter with 50% duty cycle.



It is obvious from the diagram, that we have to use both rising and falling edges of the

input clock.
The next drawing is a state diagram.

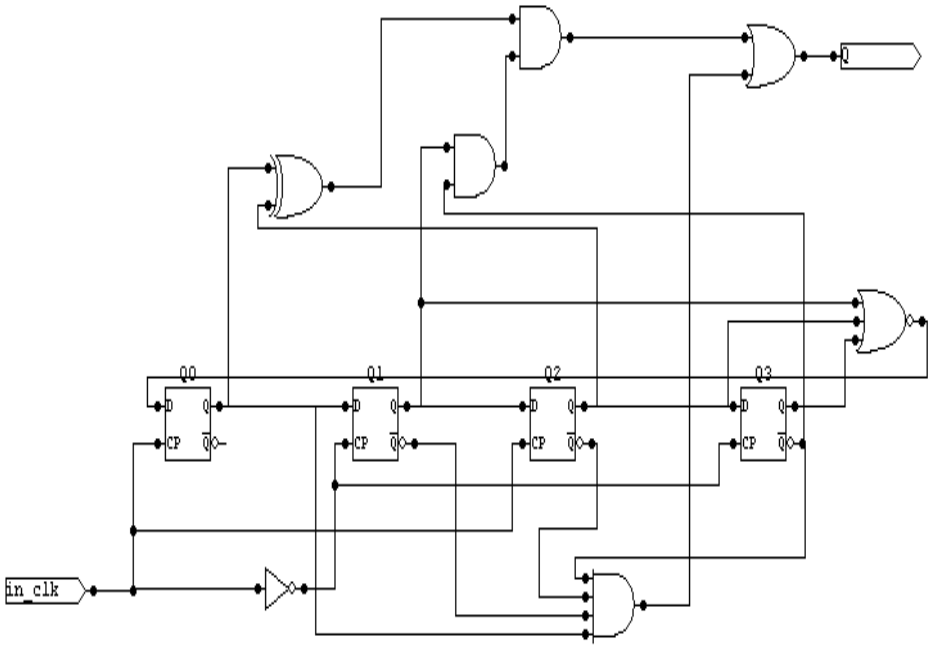


On this diagram R - is a rising edge of input clock, and F - is a falling edge.
How many FF do we need to implement 6 states? At least 3. In this example I am going to use 4 D-type FF just to simplify the design.
Now, look at the table below. Q0 ... Q3 are the outputs of FFs. Q - is the output of the devider with 50% duty cycle. In the first raw the outputs are in the initial state: 0000. In the second raw - the data after the first rising edge and so on.The status of the FFs' ouputs is changing on every rising or falling edge of the input clock according to the information on D-inputs. So, D-inputs should have the data before the clock edge.

in_clk	Q	Q0	Q1	Q2	Q3	D0	D1	D2	D3
R1	1	1	0	0	0	1	1	0	0
F1	1	1	1	0	0	0	1	1	0
R2	1	0	1	1	0	0	0	1	1
F2	0	0	0	1	1	0	0	0	1
R3	0	0	0	0	1	0	0	0	0
F3	0	0	0	0	0	1	0	0	0

These equations are resulting from the table analysis:
D1 = Q0
D2 = Q1
D3 = Q2
D0 = (Q1+Q2+Q3)'
Q = Q0*Q1'*Q2'*Q3'+Q0*Q1*Q2'*Q3'+Q0'*Q1*Q2*Q3' =
Q1*Q3'(Q0*Q2'+Q0'*Q2)+Q0*Q1'*Q2'*Q3'

Now it is the time for the circuit diagram:



Din contributed to this article. Your comments are welcome at eclub@hitequest.com
We will gladly post your design as well.

Q:I swapped transistors in CMOS inverter (put n-transistor at the top and p-transistor at the bottom).
Can I use this circuit as a noninverting buffer?

Not really. High input level can't open n-transistor,because it's source has the same high potential (Vdd) as a gate. By the same reason low level will not open p-transistor.

Further discussion:

```
> > Dear Mr.Martovetsky,
First off, I'd like to thank you and Hitequest for hosting an excellent
website
which has been very useful to me.Great work!I have a question about
 the solution about the noninverting buffer.


I didn't quite understand this because I thought the source and drain
were
 interchangeable depending on which one is at a higher potential.In the
case of
 NMOSFET, if one of the 2 terminals is tied to VDD,then, doesn't that
become the
 drain since it is at a higher potential? I'd really appreciate if you
could
```

enlighten me on this whenever you get time.Thank you and have a great day:-)

Thanks,Sriram

I still am not clear with one thing.Perhaps,I should rephrase my question:
Consider the swapped circuit with NMOS on top and PMOS below.One terminal of
the NMOS is connected to Vdd(this becomes the drain due to its higher potential).The gate is connected to Vdd as well.So shouldn't it act like a pass
transistor and conduct current? there is a voltage difference between source
and drain since drain is connected to Vdd,right?

I'm really curious to know the answer.Thanks for your time and have a great
day:-)

Thanks Sriram

> > OK, think about it this way: what does the current between drain and
> > source depend on? (Vgs - Vt) - any book at the first page,right?
> > If you put nmos transistor at the top, it's Vs = Vdd/2. You can barely
> > open it applying Vg=Vdd. In order to open it you need to apply Vg
> > greater than Vdd.
> > Same story with pmos tarnsistor.
> > Does it make sense?
> > Take care.
> > alex

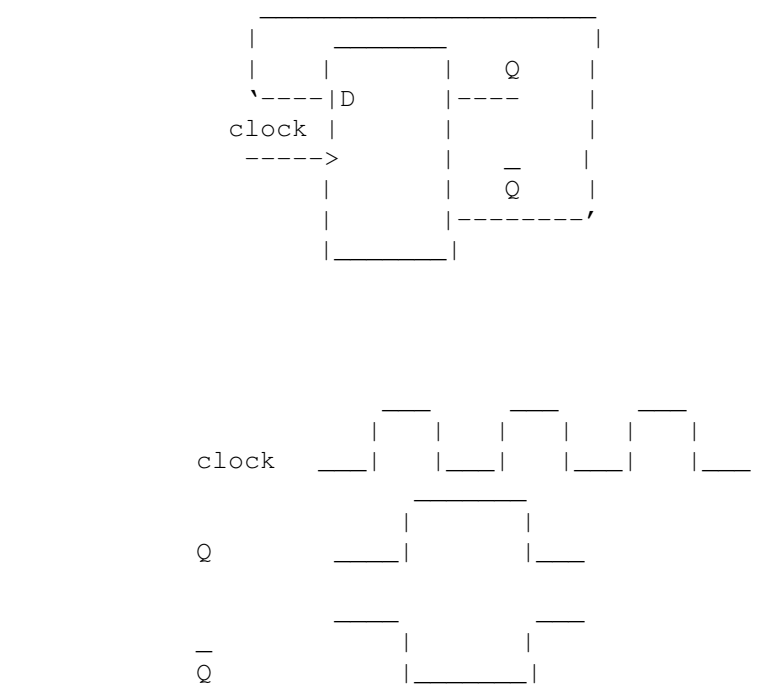
> Dear Mr.Paikin,
> Once again,thanks for responding immediately.I don't know what I'm missing but
> how is the Vs of the NMOS equal to vdd/2.I thought the source of the NMOS is
> basically the output node of the new arrangement and its source voltage is
> indeterminate at the beginning.Look forward to your reply.I appreciate your
> help.
>
> Cheers
> Sriram

---
hello Sriram,
if both transistors are closed,the output is in the tristate,which is Vdd/2 (think about these transistors like of 2 large value resistors).If
you apply "1" to n_mos transistor, it is trying to open ,and it's Vs

would be close to Vdd.But it can't happen,because your Vg is not high enough.
alex

Q: Convert D-latch into divider by 2.
   What is the max clock frequency the circuit can handle ?
                T_setup= 6nS
                T_hold = 2nS
                T_propagation = 10nS

A:



Any system with clock should meet setup and hold time conditions.
Besides since there is a feedback from !Q to D, we should take care of D input timing: the data on D input should not change while clock is high!
Otherwise the results are unpredictable.
To meet these conditions:

t_clock_high <= T_prop

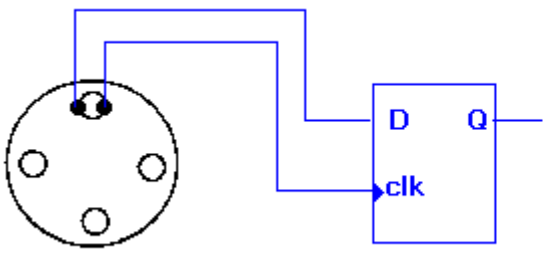
t_clock_low >= T_setup

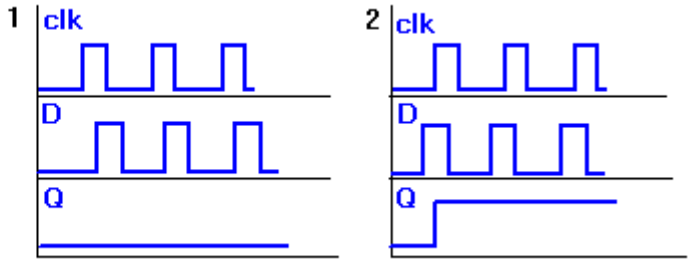
T_hold <= T_prop

For example if we take t_clock_high= t_clock_low = 6nS
Then clock period = 12nS,i.e max Freq = 80MHz

Q: The circle can rotate clockwise and back. Use minimum hardware to build
a circuit to indicate the direction of rotating.

A: 2 sensors are required to find out the direction of rotating.
They are placed like at the drawing. One of them is connected to the data input of D flip-flop,
and a second one - to the clock input. If the circle rotates the way clock sensor sees the light
first while D input (second sensor) is zero - the output of the flip-flop equals zero, and if D input
sensor "fires" first - the output of the flip-flop becomes high.






```
module fifo1 (full,empty,clk,clkb,ain,bout,rst_N)
output [7:0] bout;
input [7:0] ain;
input clk,clkb,rst_N;
output empty, full;
reg [3:0] wptr, rptr;
...

endmodule
```

wptr -->

0
1
2
3
4
5
6
7
8

<-- rptr

9
10
11
12

Multiple clocks add complexity to this design. We need to define conditions for Empty and Full signals, take care of WR and RD pointers. Here is one of the solutions.

Empty and Full flags:
assign empty=((wptr == rptr) && (w_flag == r_flag);
assign full=((wptr == rptr) && (w_flag == ~r_flag);

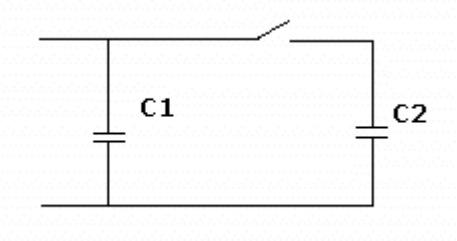
where w_flag is set when wptr =12 (end of FIFO). After that wptr is reset to 0. The same for r_flag and rptr.

Pointer handling:
if (wptr == 12) {w_flag,wptr} <= {~w_flag,4'b0000};
else wptr <= wptr+1;
if (rptr == 12) {r_flag,rptr} <= {~r_flag,4'b0000};
else rptr <= rptr+1;

Q:What does CMOS stand for? VLSI?

A:VLSI - means a Very Large-Scale Integration.
CMOS stands for Complementary Metal-Oxide Semiconductor technology... Don't stop here, draw a complementary transistor pair and tell them how it works!

Q: Two capacitors are connected in parallel through a switch. C1= 1uF, C2=0.25uF. Initially switch is open,C1 is charged to 10V. What happens if we close the switch? No loss in the wires and capacitors.



A:
Since no loss in the circuit the charge remains the same:
U1C1 + U2C2 = U3(C1+C2)
U3 = (U1C1+U2C2)/(C1+C2) = (10*1 + 0*0.25)/1+0.25 = 8
U3= 8V

Q: You work on a specification of a system with some digital parameters. Each parameter has Min,Typ and Max colomns.
What colomn would you put setup and hold time?

A: put SETUP time into the Min colomn, put HOLD time into the Min colomn too.
Example:
usually the data must be set at least (minimum) X nS before clock and being held at least Y nS after the clock. You need to specify Min setup and Min hold time.

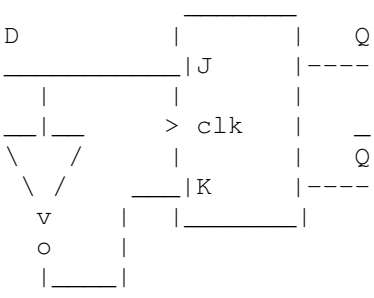
Q:How to convert D-latch into JK-latch and JK-latch into D-latch?

Compare the truth tables of D-latch and JK-latch accordingly:

clk	D	Q
+	0	0
+	1	1

clk	J	K	Q
+	0	0	hold
+	0	1	0
+	1	0	1
+	1	1	switch to opposite

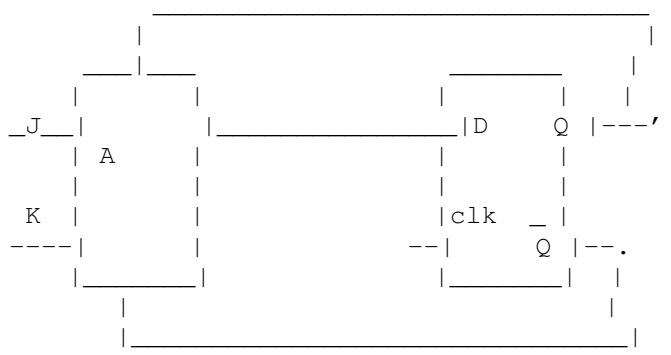
Using these truth tables it is easy to convert JK-latch into D-latch.



To convert D-latch into JK-latch some extra logic is required.
The following table shows the relation between J,K and D

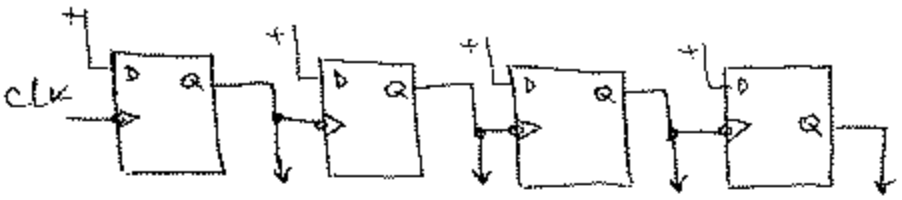
J	K	D

0	0	Q
0	1	0
1	0	1
1	1	!Q

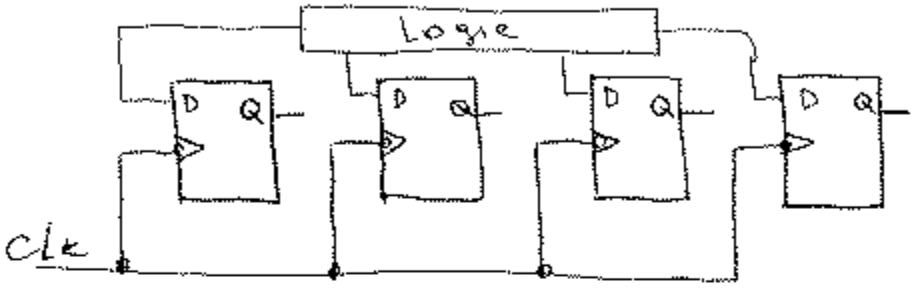


Looking at the drawing and the table it is not a problem to implement block A.
Probably the easiest way is to use a MUX.
J and K are control signals and 1,0,Q,!Q are data inputs.

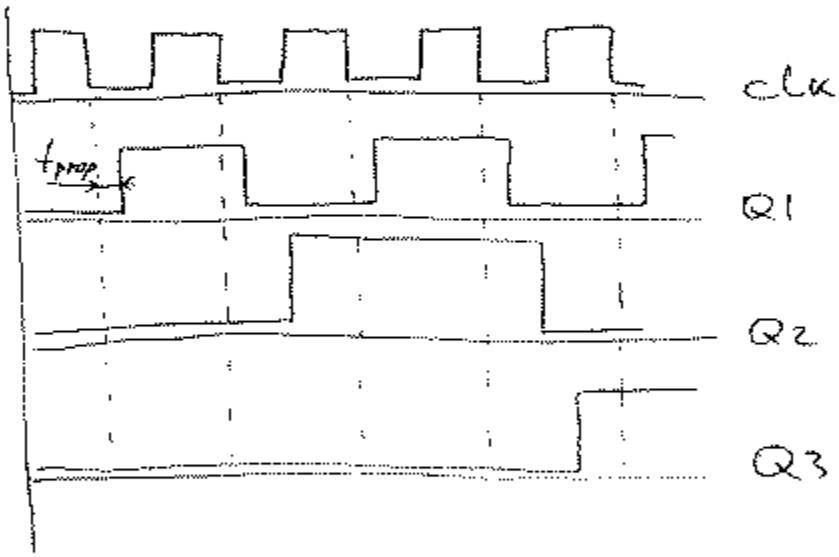
Q:You have two counters to 16 built from negedge D- FF . First circuit is synchronous and second is "ripple" (cascading). Which circuit has less delay?



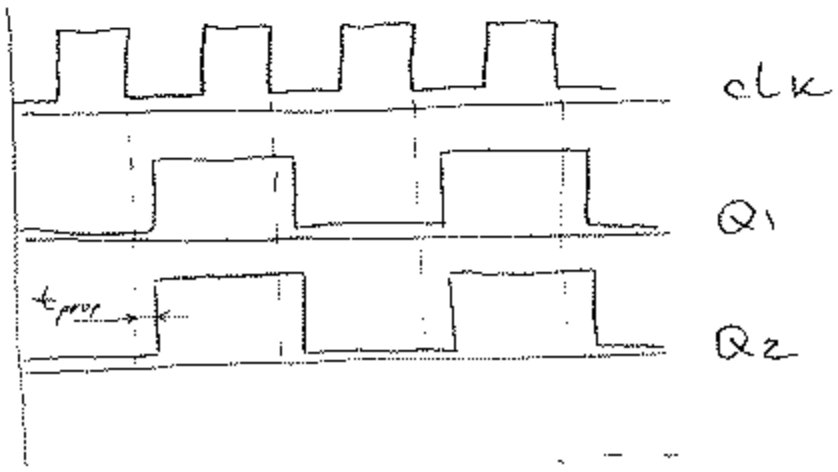
1



2



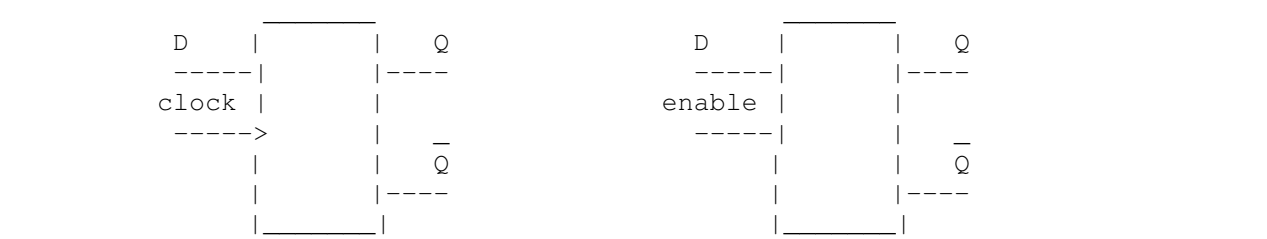
3



4

1 - is ripple counter;
2 - synchronous.
Both consist of 4 FF, synchronous counter also has some logic to control it's operation. From diagram 3 (for ripple) and 4 (for synchronous) it is seen that propagation delay of ripple counter is 4* t_prop , while synchronous counter has only 1* t_prop.

Q:What is the difference between flip-flop and latch?



The example shows D-latch and D-FF.
The simplest form of data storage is latch.It's output responds immidiately
to changes at the input and the input state will be remembered, or "latched" onto.
While "enable" input is active the input of the latch is transparant to the output,
once "enable" is disactivated the output remains locked.
Flip flops use clock as a control input. The transition in output Q occurs only at the edge
of the clock pulse. Input data must present T_setup time before clock edge and remain T_hold time after.

* RESET input, while it is not shown, is present in most FF.

```
module DFF (Q,_Q,D,clk,rst);
output Q,_Q;
input D,clk,rst;
reg Q,_Q;

always @(posedge clk or posedge rst)
 begin
  if (rst) Q  <= 0;
  else     Q  <= D;
  _Q <= !Q;
 end

endmodule

module DLatch (Q,_Q,D,en,rst);
output Q,_Q;
input D,en,rst;
reg Q,_Q;

always @(en or D or posedge rst)
```

```
 begin
  if (rst)      Q  <= 0;
  else if (en) Q  <= D;
  _Q <= !Q;
 end

endmodule
```

Here are the most typical interview questions about your personality.
Nevertheless a proper answer to these questions is just as important, as a technical qualification.

So, tell me something about yourself?
Remember, this question is totally job-related. It is designed for the interviewer to hear you talk and see how you express yourself. Don't talk long about your marital status, your hobbies or go through your whole resume. Instead, you need to summarize your response and talk about key accomplishments in your career: " These are the things I am good at ..., these are the things I can do for the company ..." You can briefly show your professional identity, and what you are looking for (professionally).

Name 3 positive and 3 negative qualities of yours.
Positive qualities: get along with people, high motivation to learn, never give up facing a problem , responsibility
(at least not switching jobs in the middle of the project) etc.
Negative: Hey, nobody is perfect. You have to give something.But don't go too far, you don't want to look as a bad person either.

Where do you see yourself professionally, in 3-5 years?
Some people talk about their professional development.
It is a good idea to tell if you want to take any particular classes or explore any particular areas in the professional environment.

What is the most exciting and boring part of your job?
There is no 100% exciting job. Every job has it's boring part.

Do you prefer team work or individual?

Be carefull. If you say "individual" - it may sound like you are not a team player, if you say you prefer team work - it means you can't work alone.

What is the most significant responsibility you have ever had in your life?
Matovolwa Peter, email: matovolwap@scitz.com

You are assigned to work on an important project containing 7 design modules and you are short of time.
By the end of the month all you can do is either to have 3 modules accomplished or to have all 7 modules started in parallel but not finished.
What strategy will you choose?
You definitely want to have at least part of your work finished.You can demonstrate it, explain your problems and ask some more time to complete the project.

Why do you want to leave your present job?
Do not blame your company,your job,managers. The interviewer may think you are not getting along with your supervisor or coworkers. In this case you are not someone they would want to hire. Instead, you can say it is a time for you to move on, try yourself in the new area, etc

What do you know about our company?
You have to do some homework before the interview. At least look at the company's WEB page.

What particularly would you like to work on?
Some people say:"What ever you want me to do!" In most cases this answer will not be appreciated, especially in start up companies.
It may sound like you have no any other interest in this job but money.

Tell us about one of the technical problems you had to solve recently
Bill Benson, technical recruter from Silicon Valley, says this question is quite frequent on interviews. Don't miss this chance ! Tell how good you are in solving technical problems.

What is the earth?
Sometimes people ask strange questions to see your reaction .Just stay cool and bring your sense of humor.
A friend of mine who really likes talking answered to the above question:

"How much time do you have?"

The next few questions are reproduced from www.microsearchsf.com with kind permission of Janice Schooler Litvin, executive search consultant

---

Tell me about a conflict you encountered and how you handled it.
HINT : This is one of the toughest interview questions of all. It's sort of a trick question, as a matter of fact. Never speak negatively about anyone. The ability to successfully resolve conflicts is important for all members of an IS team.. It may be the most important factor if you're working in a service environment, such as a large consulting firm that deals with outside clients. The answer you give here could go a long way toward getting you a job offer. Managers want to see that you are mature and unselfish. The answer should involve proof of your maturity level. They are looking for your ability to handle conflict. Compromise and working it out without external intervention are the keys. A disgruntled person is not going to be productive, and tends to bring down coworkers' morale as well.

What changes have you made in your life that you are most proud of?
HINT : This tells the manager more about your ability to take control of your life. It illustrates your leadership potential, and suggests just how promotable you might be. After all, if he produces a star, he looks good.

What are your salary requirements?
HINT : The use of the word "offer" is critical. It's a subliminal message that an actual job offer is what you are discussing, not just your salary needs in general.
A : "Salary is not my primary consideration. Of course, I have to pay the bills. I'd be open to any reasonable offer." Pause and maintain direct eye contact, even if it seems like forever. Do not be the first one to flinch. Do not over-talk. Be prepared for a long silence. Let the manager be the first to present a figure. It will give you power and control.
If forced to give a specific number, never give a broad range -- you will usually be offered the low end. Instead, be as precise as possible: "I'd be open to something in the low-fifties (or mid-forties, high-seventies, whatever)." Giving such a specific number presumes you've researched

the local job market and know what people with your skills are making.

Are you interviewing at any other companies?
HINT : You want the manager to know that you're extremely interested in his opportunity, but are keeping your options open.
A : "Yes, Mr. X, but at this point XYZ is my first choice."

Remember, all of these interview questions have more than one appropriate answer.
If you are feeling nervous about an upcoming interview, keep in mind that the hiring manager gets just as excited about a potentially strong candidate as the candidate does about him or her.
Strong, qualified, motivated technical people are very hard to find. Be direct, but think before you speak, and you will surely get an offer.

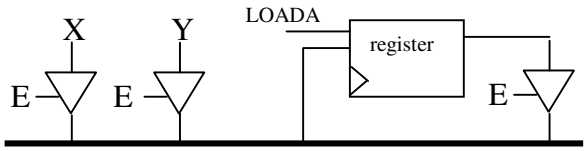
MT1 practice questions. These questions aren't to test your reasoning capabilities, just your knowledge of the basics.

1. 3 Bit Up Counter (Moore FSM Design, Setup/Hold Time, Logic Simplification)
   1a) Build a 3 bit up counter Moore machine using only AND, OR, NOT, and XOR gates. Simplify all logic. (Use the fewest number of gates possible.)

   1b) Will the circuit have any problems if:
   2ns < T_{pINV} < 4ns
   3ns < T_{pAND} < 6ns
   3ns < T_{pOR} < 5ns
   3ns < T_{pXOR} < 4ns
   3ns < Tcko < 8ns
   Tsetup = 4ns
   Thold = 6ns
   CLK = 40 MHz

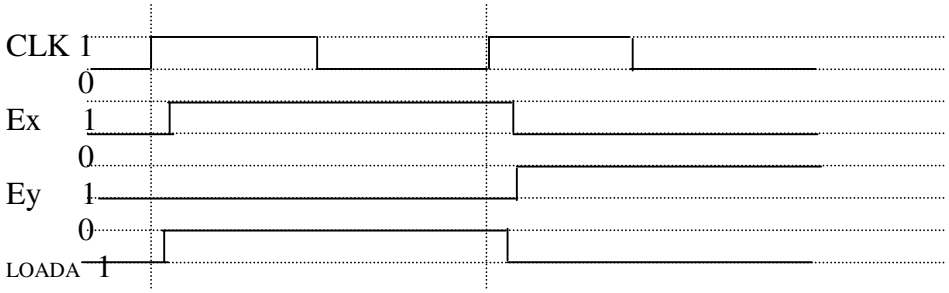
2. Timing problem… (All questions use homework 5, problem 2)
   2a) (HW5, 2b) Without doing any math, do you think increasing T_{DELAY} will increase $(T1\text{-}T0)_{MAX}$ or decrease it.
   2b) (HW5, 2c) Without doing any math, do you think increasing T_{DELAY} will increase $(T1\text{-}T0)_{MIN}$ or decrease it.
   2c) What is the formula for finding a setup time violation on HW #5's 2b? Use 'T' for the period.
   2d) What is the formula for finding a setup time violation on HW #5's 2c? Use 'T' for the period.

3. BUFEs and timing



(Note: E_{XAZ} means the time it takes the output of E_X to go from active to tristated. The turn-off time.)

The controller sends the following signals to the above datapath:



Are there any problems when:

3a) E_{XAZ} = 10ns, E_{YZA} = 5ns, T_{setup} = 10ns, T_{hold} = 4ns, Period = 30ns?

3b) E_{XAZ} = 9ns, E_{YZA} = 10ns, T_{setup} = 10ns, T_{hold} = 10ns, Period = 30ns?

3c) E_{XAZ} = 10ns, E_{YZA} = 12ns, T_{setup} = 10ns, T_{hold} = 4ns, Period = 20ns?

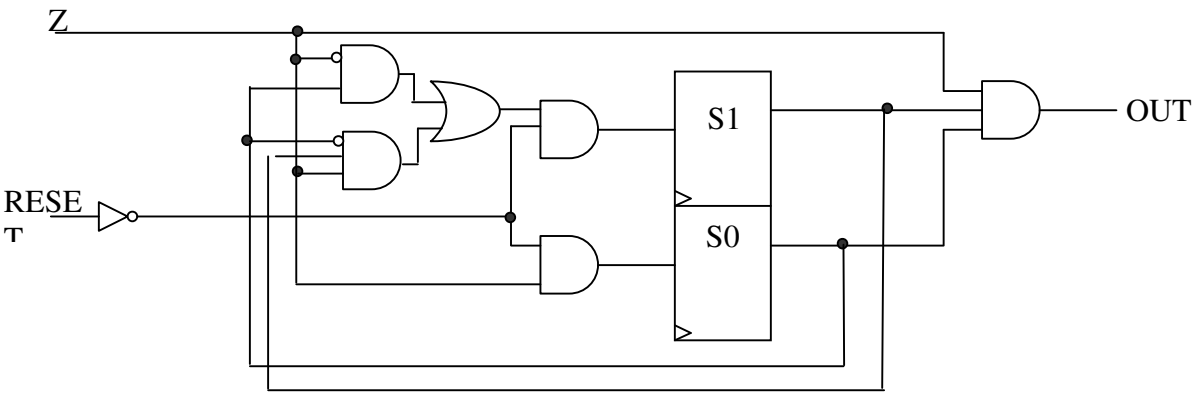
3d) E_{XAZ} = 14ns, E_{YZA} = 6ns, T_{setup} = 12ns, T_{hold} = 8ns, Period = 25ns?

4. Clock divider (FSM Design, Counters, Clocks and Glitches)

Design a clock divider which runs on a 200MHz clock and produces a 1MHz clock. Make the duty cycle of the 1MHz clock 50%. Should you design this to be a Moore or a Mealy machine?

5. FSM Analysis

What does this circuit do? (Is it a Mealy or Moore? Where are the NSD, OD & State FFs.)



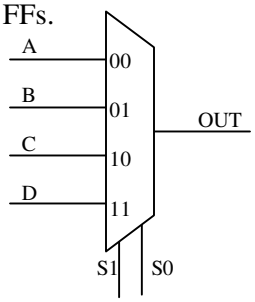
6. STD & STT
   Draw a Moore STD for a circuit that does the same thing as the circuit in problem 5 and then make it's STT.

7. Gate Delay Timing Diagram (Cross Coupled Gates, Gate Delays)
   It's too much to try create a problem like the one on last year's MT. Try to do last year's (and if you can get it, the year before's timing problem might be good practice).

8. Toggle Flip Flops (FSM Design)
   Design a 2 Bit Down Counter that is implemented with T FFs.

9. JK Flip Flops (FSM Design)
   Design a 2 Bit Up Counter that is implemented with JK FFs.

10. Use 3 BUFTs and 3 BUFEs to build a 4-1 MUX.



Also review asserted high, asserted low, & equation simplfication.

Solutions:

1. a) STT:

PS2	PS1	PS0	NS2	NS1	NS0
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0

Equations:

$NS0 = \overline{PS0}$

$NS1 = PS1 \oplus PS0$

$NS2 = PS2\ \overline{PS0} + PS2\ \overline{PS1} + \overline{PS2}\ PS1\ PS0$

b) Any problems?

Check for hold time violations (Can value from flip-flop output propagate back to the flip-flop inputs so quickly that the hold time is violated?):

> Find the shortest propagation path: Looks like NS0 (XOR gates have bigger delay than inverters).

$Tckp_{min} + Tinv_{min} \stackrel{?}{>} T_{hold}$

3ns + 2ns $\stackrel{?}{>}$ 6ns This isn't true → hold time violation possible.

Check for setup time violations (Can value from flip-flop output propagate back to the flip-flop inputs so slowly that the next clock's setup time is violated?)

> Find the longest propagation path: Looks like NS2.

$Tcko_{max} + Tinv_{max} + Tand_{max} + Tor_{max} + Tsetup \stackrel{?}{<} Period$ (1/40MHz = 25ns)

8ns + 4ns + 6ns + 5ns + 4ns < 25ns False → setup time violation possible.

2. 2a) By increasing the delay, you're delaying the change in D1's value. Which means that you can increase T1-T0.

2b) The delay won't affect the minimum T1-T0.

2c) Tckomax + INVTpmax + Tsumax < T – (T1-T0).

2d) Tckomax + Tdelaymax + Tsumax < T – (T1-T0).

3. 3a) Bus conflict (X & Y both on bus at same time).

3b) Hold time violation. Turn-off time (E_{XAZ}) is less than Thold of flip-flop.

3c) Setup time violation. E_{YZA} + Tsetup > Period.

3d) Setup time violation, hold time violation & bus conflict.

4. This should be a Moore machine with two states and no logic on the output – the output should come directly from the state bits. Because... any combinational logic, no matter how simple may cause glitches. It should change states when a counter counts up to 100. Since the output of this circuit is a clock, we have to make especially sure that there aren't glitches on the output. Moore is safer than a Mealy in this case. Why?

5. It's a Mealy pattern detector (1011).

Equations:

$NS1 = \overline{R}\,(\,PS0\,\overline{Z} + PS1\,\overline{PS0\,Z}\,)$

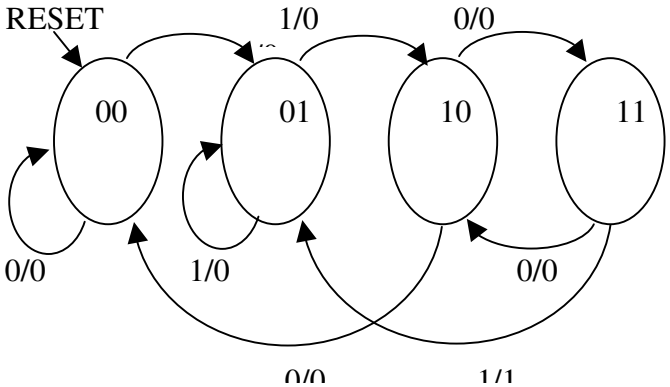
$NS0 = \overline{Z}\,\overline{R}$

$OUT = Z\,PS1\,PS0$

STT:

RESET	PS1	PS0	Z	NS1	NS0
	OUT				
0	0	0	0	0	0	0
0	0	0	1	0	1	0
0	0	1	0	1	0	0
0	0	1	1	0	1	0
0	1	0	0	0	0	0
0	1	0	1	1	1	0
0	1	1	0	1	0	0
0	1	1	1	0	1	1
1	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	0	0	0
1	0	1	1	0	0	0
1	1	0	0	0	0	0

STD:



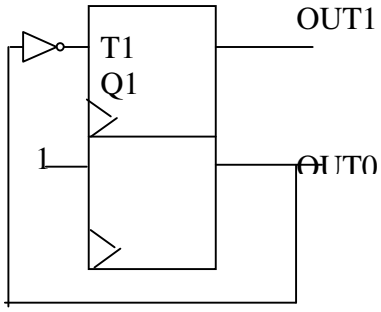
6. The Moore STD has 5 states. 1 is output in the 5th state.
On the STT, make sure that the output is the same for the same PS even though the input changes.

7. ------

8. STT:

PS1	PS0	Bookkeeping T1	Bookkeeping T0	NS1	NS0
0	0	1	1	1	1
0	1	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0

Equations found for T1 and T0:



9. STT:

Equations:

PS1	PS0	Bookkeeping J1K1	Bookkeeping J0K0	NS1	NS0
0	0	0X	1X	0	1
0	1	1X	X1	1	0
1	0	X0	1X	1	1
1	1	X1	X1	0	0

$NS1 = PS1 \cdot \overline{PS0} + PS1 \cdot \overline{K1}$

$NS0 = PS1 \cdot PS0 + \overline{PS1} \cdot J0$

$J0 = \overline{PS0}$

$K0 = PS0$

X means “don’t care”.

10.