EC &EI

VLSI

1. what is the difference between mealy and moore state-machines

2. How to solve setup & Hold violations in the design

To solve setup violation

1. optimizing/restructuring combination logic between the flops.
2. Tweak flops to offer lesser setup delay [DFFX1 -> DFFXx]
3. Tweak launch-flop to have better slew at the clock pin, this will make CK->Q of launch flop to be fast there by helping fixing setup violations
4. Play with skew [tweak clock network delay, slow-down clock to capturing flop and fasten the clock to launch-flop](otherwise called as Useful-skews)

To solve Hold Violations

1. Adding delay/buffer[as buffer offers lesser delay, we go for spl Delay cells whose functionality Y=A, but with more delay]
2. Making the launch flop clock reaching delayed

3. Also, one can add lockup-latches [in cases where the hold time requirement is very huge, basically to avoid data slip]

3. What is antenna Violation & ways to prevent it

During the process of plasma etching, charges accumulate along the metal strips. The longer the strips are, the more charges are accumulated. IF a small transistor gate connected to these long metal strips, the gate oxide can be destroyed (large electric field over a very thin electric) , This is called as Antenna violation.

The ways to prevent is , by making jogging the metal line, which is atleast one metal above the layer to be protected. If we want to remove antenna violation in metal2 then need to jog it in metal3 not in metal1. The reason being while we are etching metal2, metal3 layer is not laid out. So the two pieces of metal2 got disconnected. Only the piece of metal connected to gate have charge to gate. When we lay down metal3, the remaining portion of metal

got charge added to metal3. This is called
accumulative antenna effect.
Another way of preventing is adding reverse Diodes at the gates

4. We have multiple instances in RTL(Register Transfer Language), do you do anything special during synthesis stage?

While writing RTL(Register Transfer language),say in verilog or in VHDL language, we don't write the same module functionality again and again, we use a concept called as instantiation, where in as per the language, the instanciation of a module will behave like the parent module in terms of functionality, where during synthesis stage we need the full code so that the synthesis tool can study the logic , structure and map it to the library cells, so we use a command in synthesis , called as "UNIQUIFY" which will replace the instantiations with the real logic, because once we are in a synthesis stages we have to visualize as real cells and no more modeling just for functionality alone, we need to visualize in-terms of physical world as well.

5. what is tie-high and tie-low cells and where it is used

Tie-high and Tie-Low cells are used to connect the gate of the transistor to either power or ground. In deep sub micron processes, if the gate is connected to power/ground the transistor might be turned on/off due to power or ground bounce. The suggestion from foundry is to use tie cells for this purpose. These cells are part of standard-cell library. The cells which require Vdd, comes and connect to Tie high...(so tie high is a power supply cell)...while the cells which wants Vss connects itself to Tie-low.

6. what is the difference between latches and flip-flops based designs

Latches are level-sensitive and flip-flops are edge sensitive. latch based design and flop based design is that latch allows time borrowing which a tradition flop does not. That makes latch based design more efficient. But at the same time, latch based design is more complicated and has more issues in min timing (races). Its STA with time borrowing in deep pipelining can be quite complex.

7. What is High-Vt and Low-Vt cells.

Hvt cells are MOS devices with less leakage due to high Vt but they have higher delay than low VT, where as the low Vt cells are devices which have less delay but leakage is high. The threshold(t) voltage dictates the transistor switching speed , it matters how much minimum threshold voltage applied can make the transistor switching to active state which results to how fast we can switch the trasistor. disadvantage is it needs to maintain the transistor in a minimum subthreshold voltage level to make ir switch fast so it leads to leakage of current inturn loss of power.

8. What is LEF mean?

LEF is an ASCII data format from Cadence Design inc, to describe a standard cell library. It includes the design rules for routing and the Abstract layout of the cells. LEF file contains the following,

Technology: layer, design rules, via-definitions, metal-capacitance

Site : Site extension

Macros : cell descriptions, cell dimensions, layout of pins and blockages, capacitances

To get further insight to the topic, please check this
http://www.csee.umbc.edu/~cpatel2/links/414/slides/lect03_LEF.pdf

9. what is DEF mean?

DEF is an ASCII data format from Cadence Design inc., to describe Design related information.

10. Steps involved in designing an optimal padring

1. Make sure you have corner-pads, across all the corners of the padring, This is mainly to have the power-continuity as well as the resistance is less .
2. Ensure that the Padring ful-fills the ESD requirement, Identifyh the power-domains, split the domains, Ensure common ground across all the domains.
3. Ensure the padring has ful-filled the SSN(Simultaneous Switching Noise) requirement.
4. Placing Transfer-cell Pads in the cross power-domains, for different height pads, to have rail connectivity.

5. Ensure that the design has sufficient core power-pads.
6. Choose the Drive-strenght of the pads based on the current requirements, timing.
7. Ensure that there is seperate analog ground and power pads.
8. A No-Connection Pad is used to fill out the pad-frame if there is no requirement for I/O's.Extra VDD/GND pads also could be used. Ensure that no Input/output pads are used with un-connected inputs, as they consume power if the inputs float.
9. Ensure that oscillator-pads are used for clock inputs.
10. In-case if the design requirement for source synchronous circuits, make sure that the clock and data pads are of same drive-strength.
11. Breaker-pads are used to break the power-ring, and to isolate the power-structure across the pads.
12. Ensure that the metal-wire connected to the pin can carry sufficient amount of the current, check if more than one metal-layer is necessary to carry the maximum current provided at the pin.
13. In case if required , place pads with capacitance.
related information.

11. What is metastability and steps to prevent it.

Metastability is an unknown state it is neither Zero nor One.Metastability happens for the design systems violating setup or hole time requirements. Setup time is a requirement , that the data has to be stable before the clock-edge and hold time is a requirement , that the data has to be stable after the clock-edge. The potential violation of the setup and hold violation can happen when the data is purely asynchronous and clocked synchronously.

Steps to prevent Metastability.

1. Using proper synchronizers(two-stage or three stage), as soon as the data is coming from the asynchronous domain. Using Synchronizers, recovers from the metastable event.

2. Use synchronizers between cross-clocking domains to reduce the possibility from metastability.

3. Using Faster flip-flops (which has narrower Metastable Window).

12. what is local-skew, global-skew,useful-skew mean?

Local skew : The difference between the clock reaching at the launching flop vs the clock reaching the destination flip-flop of a timing-path.

Global skew : The difference between the earliest reaching flip-flop and latest reaching flip-flop for a same clock-domain.

Useful skew: Useful skew is a concept of delaying the capturing flip-flop clock path, this approach helps in meeting setup requirement with in the launch and capture timing path. But the hold-requirement has to be met for the design.

13. What are the various timing-paths which i should take care in my STA runs?

1. Timing path starting from an input-port and ending at the output port(purely combinational path).

2. Timing path starting from an input-port and ending at the register.

3. Timing path starting from an Register and ending at the output-port.

4. Timing path starting from an register and ending at the register.

14. What are the various components of Leakage-power?

1. sub-threshold leakage

- Subthreshold leakage exponential with V_{gs}

$$I_{ds} = I_{ds0} e^{\frac{V_{gs} - V_t}{n v_T}} \left(1 - e^{\frac{-V_{ds}}{v_T}}\right) \qquad I_{ds0} = \beta v_T^2 e^{1.8}$$

Drain-Induced Barrier Lowering

- Drain voltage also affect V_t

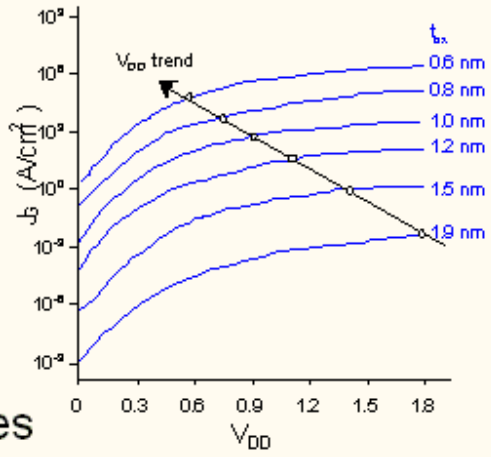
$$V_t' = V_t - \eta V_{ds}$$

- High drain voltage causes subthreshold leakage to increase.

-courtesy Khondker

2. gate leakage

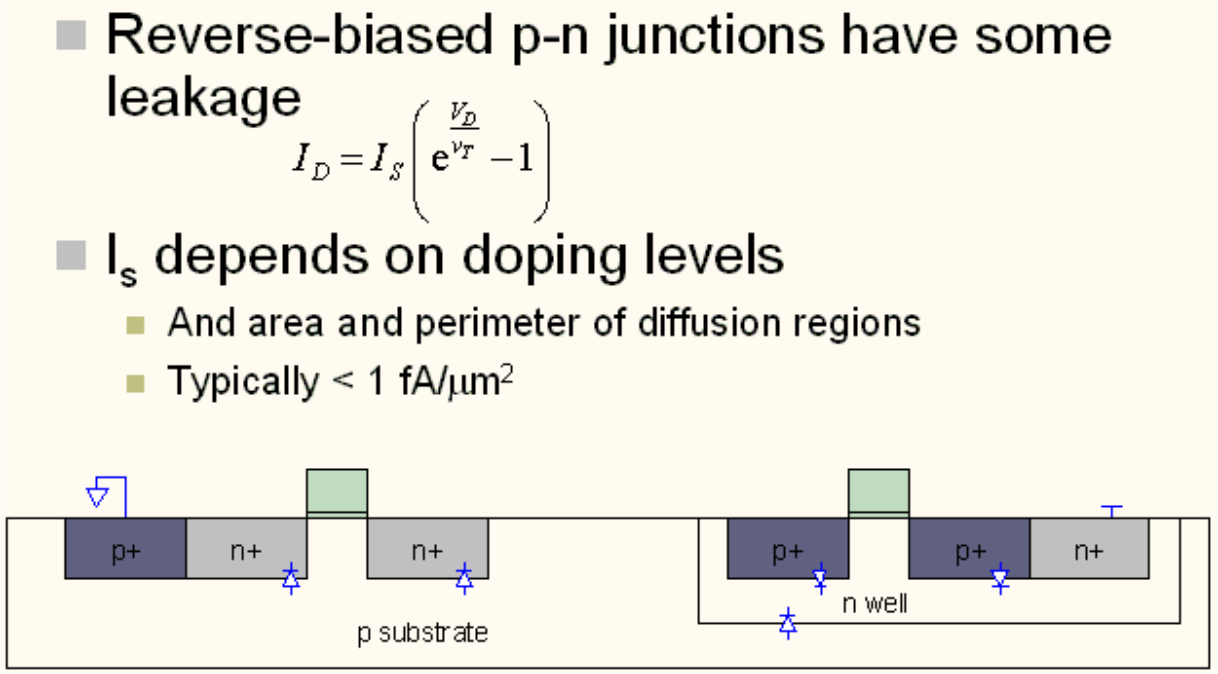
- Carriers may tunnel thorough very thin gate oxides
- Predicted tunneling current (from [Song01])



- Negligible for older processes

-courtesy Khondker

3. reverse biased drain substrate and drain substrate junction band-band tunnelling



15. What are the various yield-losses in the design?

The yield loss in the design is characterized by

1. Functional yield losses, mainly caused by spot defects , especially (shorts & opens)

2. Parametric yield losses, due to process variations.

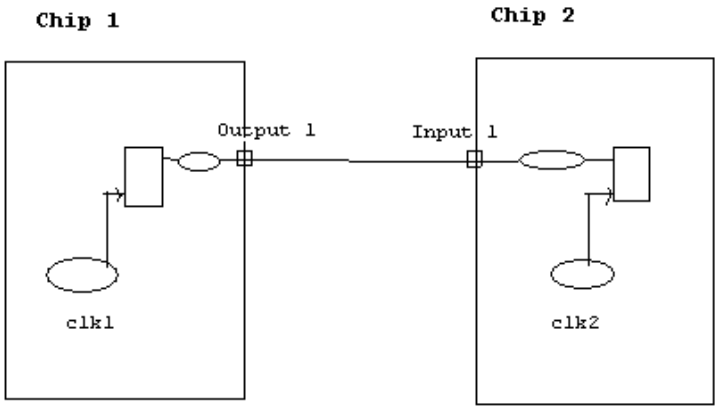
16. what is meant by virtual clock definition and why do i need it?

Virtual clock is mainly used to model the I/O timing specification. Based on what clock the output/input pads are passing the data.

For Further Understanding of the concept.
http://www.vlsichipdesign.com/images/virtual_clock.jpg

attached below..

Concept Behind Virtual-Clocks



Data transfer is from Chip1 ouput1 to Chip2 Input1
we are modelling Chip2. Create a Virtual clock with the frequency of clk2. Model the pad output, w.r.t. VCLK2, and model the output delay = (Output1 + Board delay + Input1 to Flop(chip2).

17. What are the various Variations which impacts timing of the design?

Delay impact of variations

Parameter	Delay Impa
BEOL metal (Metal mistrack, thin/thick wires)	-10% → +25 %
Environmental (Voltage islands, IR drop, temperature)	±15 %
Device fatigue (NBTI, hot electron effects)	±10%
V_t and T_{ox} device family tracking (Can have multiple V_t and T_{ox} device families)	± 5%
Model/hardware uncertainty (Per cell type)	± 5%
N/P mistrack (Fast rise/slow fall, fast fall/slow rise)	±10%
PLL (Jitter, duty cycle, phase error)	±10%

- **Requires 2^{20} timing runs or [-65%,+80%] guard band!**

[Courtesy Chandu

18. What are the various Design constraints used while performing Synthesis for a design?

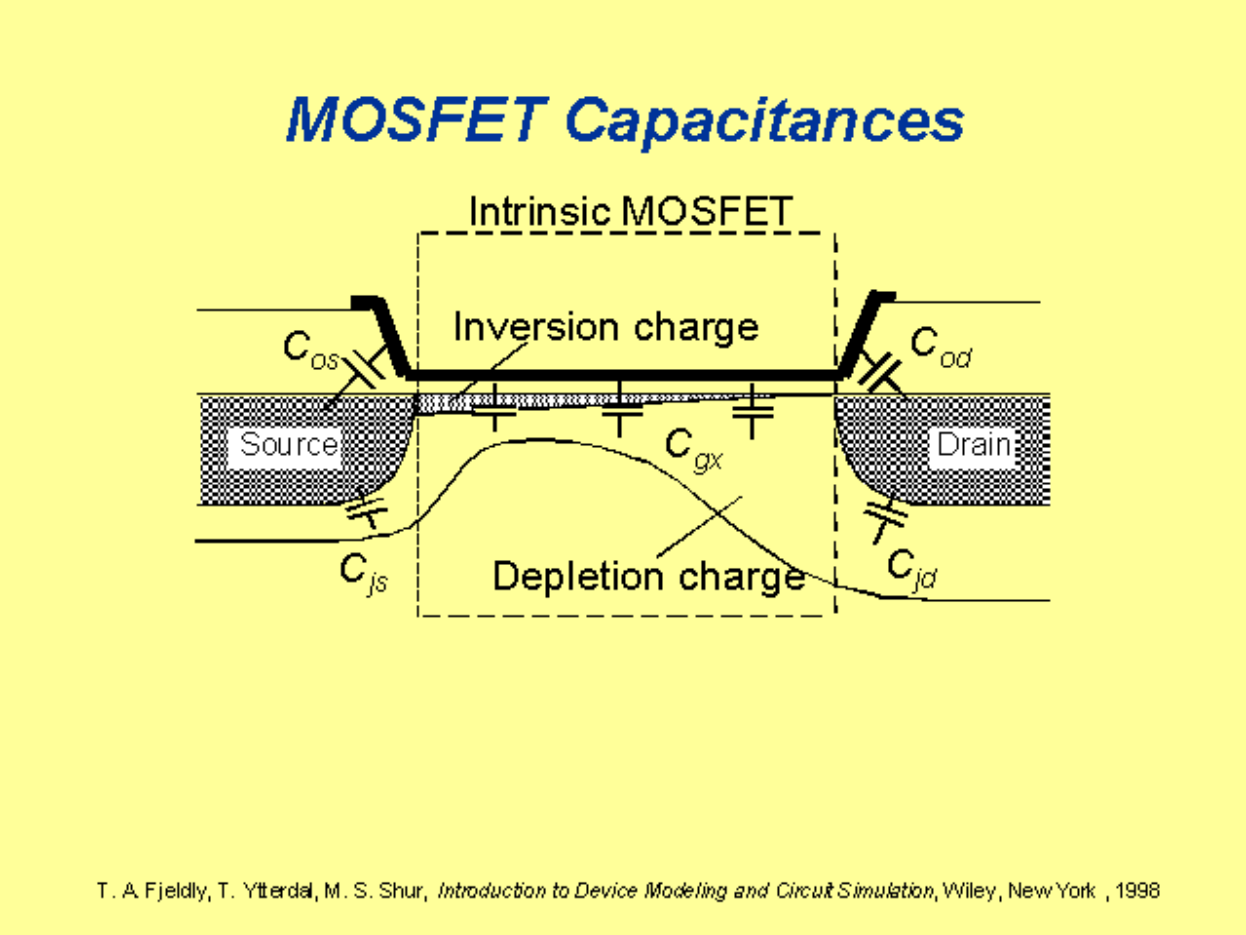
1. Create the clocks (frequency, duty-cycle).
2. Define the transition-time requirements for the input-ports.3
3. Specify the load values for the output ports
4. For the inputs and the output specify the delay values(input delay and ouput delay), which are
already consumed by the neighbour chip.
5. Specify the case-setting (in case of a mux) to report the timing to a specific paths.
6. Specify the false-paths in the design

7. Specify the multi-cycle paths in the design.
8. Specify the clock-uncertainity values(w.r.t jitter and the margin values for setup/hold).

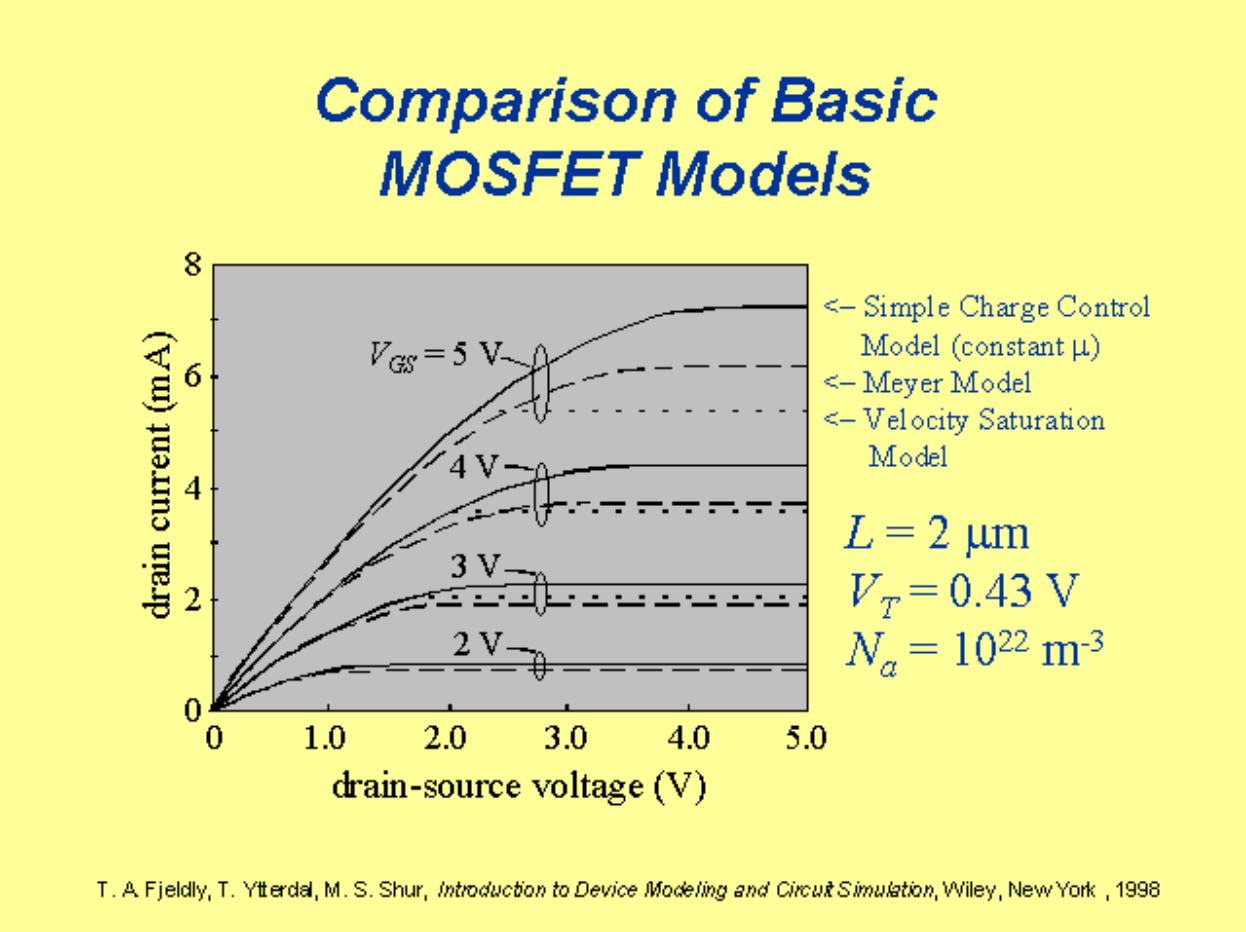
19. Specify few verilog constructs which are not supported by the synthesis tool.

initial, delays, real and time data types, force and release, fork join.

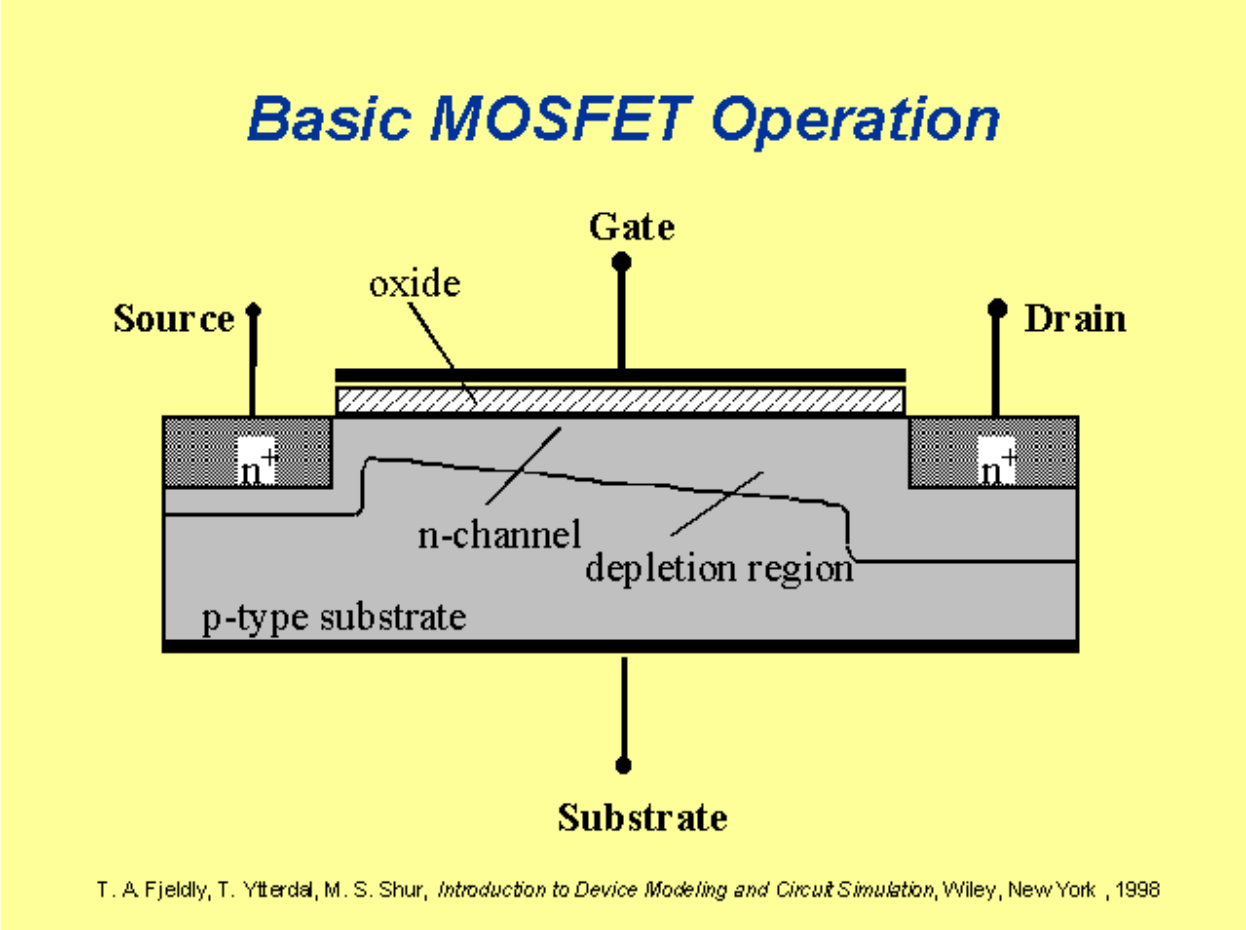
20.what are the various capacitances with an MOSFET?/strong>



21.Vds-Ids curve for an MOSFET, with increasing Vgs.



22. Basic Operation of an MOSFET.



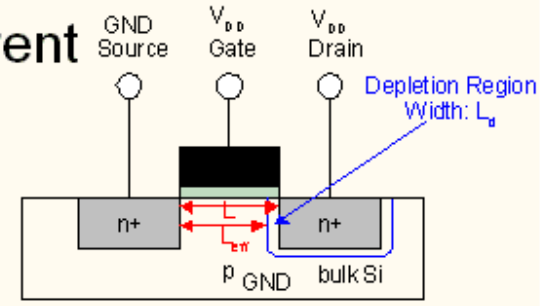
23. What is Channel length Modulation?

Reverse-biased p-n junctions form a *depletion region*

- Region between n and p with no carriers
- Width of depletion L_d region grows with reverse bias
- $L_{eff} = L - L_d$

Shorter L_{eff} gives more current

- I_{ds} increases with V_{ds}
- Even in saturation



-courtesy Khondker

24. what is body effect?

Increase in Vt(threshold voltage) , due to increase in Vs(voltage at source), is called as

body effect.

Body Effect Model

$$V_t = V_{t0} + \gamma\left(\sqrt{\phi_s + V_{sb}} - \sqrt{\phi_s}\right)$$

- ϕ_s = *surface potential* at threshold

$$\phi_s = 2v_T \ln\frac{N_A}{n_i}$$

  - Depends on doping level N_A
  - And intrinsic carrier concentration n_i

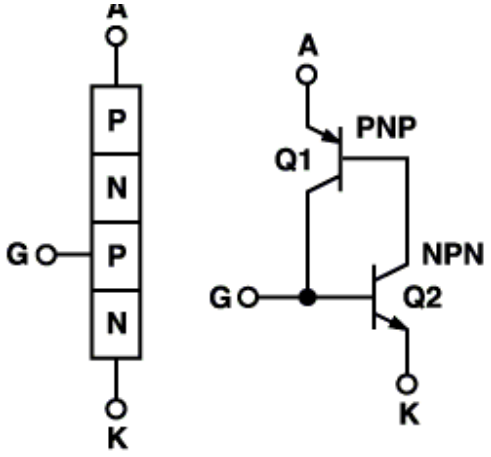
- γ = *body effect coefficient*

$$\gamma = \frac{t_{ox}}{\varepsilon_{ox}}\sqrt{2q\varepsilon_{si}N_A} = \frac{\sqrt{2q\varepsilon_{si}N_A}}{C_{ox}}$$

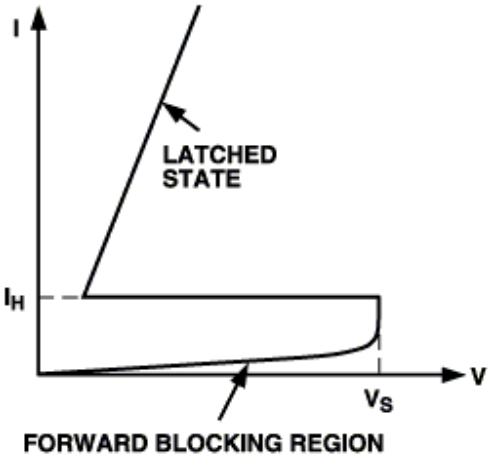
25. What is latch up in CMOS design and ways to prevent it?

To best understand the concept behind the latch up, we need to understand the concept behind SCR(Silicon Controlled Rectifiers), and how to model the basic transistor in an SCR structure and on what conditions SCR structures are created in the CMOS design process and its effects and what are the ways used to prevent it in the design-phase. An SCR is an acronym for Silicon Controlled Rectifier. It works similar to a typical diode, but is controlled similar to a bipolar transistor as far as connections go. Connection points are Anode [A], Cathode [K], and Gate [G]. The SCR is made up of two "P-N" junctions with a "Gate" attachment between them. The gate is connected between the two P-N junctions with a current waiting in the forward bias direction [+ to -] and the voltage is

above 1-volt. A momentary pulse to the gate will cause the SCR to conduct and current will flow across the device until the value changes.



a) Transistor equivalent of an SCR.



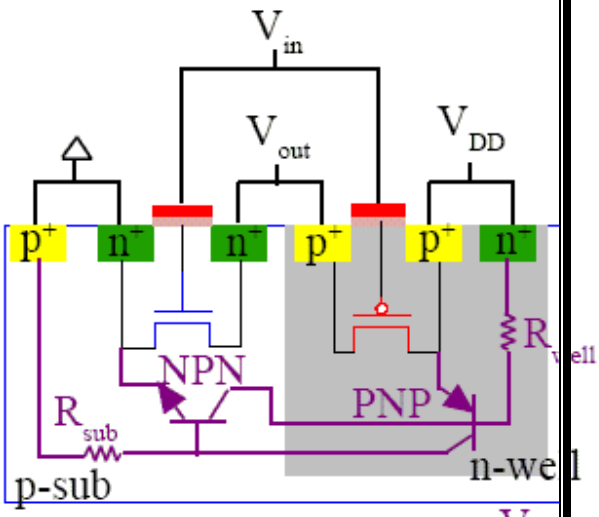
EC &EI

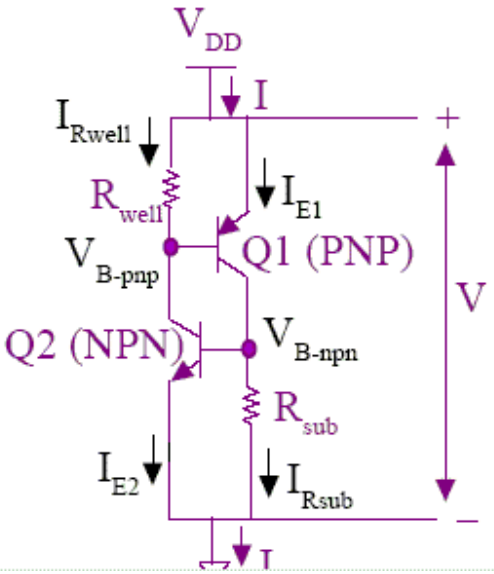
VLSI

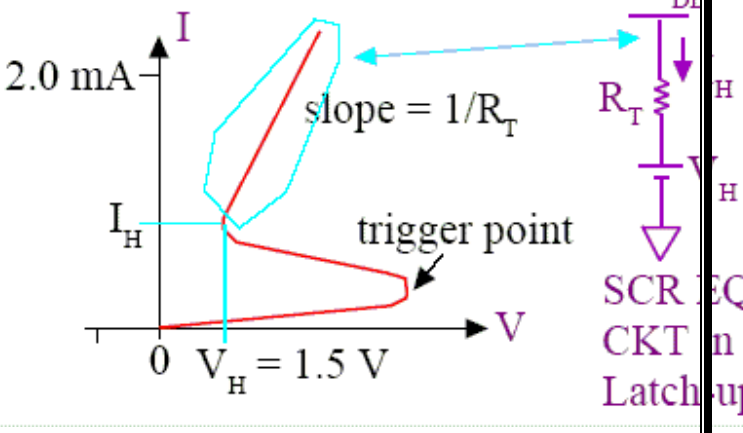
LATCHUP IN CMOS CIRCUITS

latch-up susceptability proportional to

$$\frac{1}{NSUB} \times \frac{1}{(nMOS - to - pMOS\,spacing)^2}$$

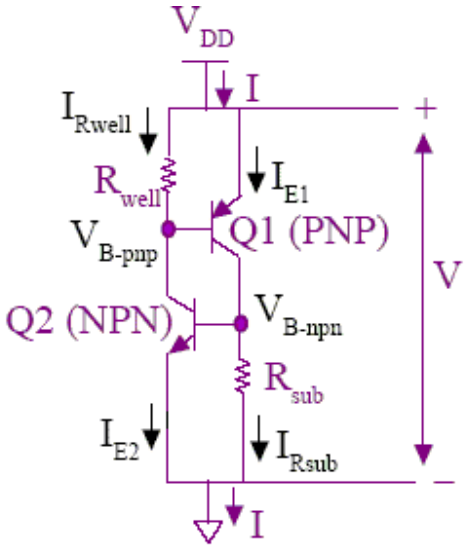


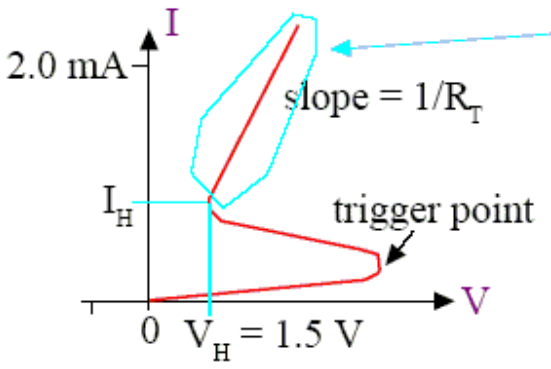


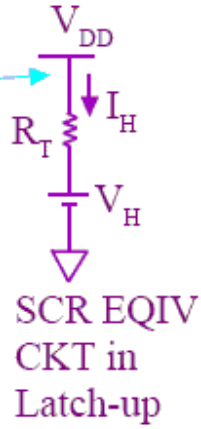


EC &EI

VLSI







$$I_H = \frac{I_{Rsub}\alpha_{pnp} + I_{Rwell}\alpha_{npn}}{\alpha_{pnp} + \alpha_{npn} - 1}$$

At the on-set of latch-up

$I > I_H = (V_{DD} - V_H)/R_T$

LATCH-UP CONDITION

$$\alpha_{npn} + \alpha_{pnp} \geq 1 + \left(\frac{\frac{R_T}{R_{well}}\alpha_{pnp} + \frac{R_T}{R_{sub}}\alpha_{npn}}{\frac{V_{DD}}{V_{BE-sat}} - 2} \right)$$

LATCH-UP PREVENTION

make large

$$\alpha_{npn} + \alpha_{pnp} \geq 1 + \left(\frac{\frac{R_T}{R_{well}}\alpha_{pnp} + \frac{R_T}{R_{sub}}\alpha_{npn}}{\frac{V_{DD}}{V_{BE-sat}} - 2} \right)$$

make small

LATCHUP OCCURS IF SATISFIED-

TO PREVENT LATCH-UP: Insure Latchup Condition is Violated!

Reduce α_{npn}, α_{pnp}; Reduce R_{sub}, R_{well}

A. Use a latchup resistant process.

Latchup Prevention LAYOUT Guidelines:

B. Use p^+ guard rings connected to GND around nMOS transistors and n^+quard rings connected to V_{DD} around the pMOS transistors to reduce R_{well} and R_{sub} and to weaken BJTs.

C. Place sub, well contacts close to the nMOS, pMOS source connections to supply rails (i.e. GND for nMOS, V_{DD} for pMOS) to reduce R_{well} and R_{sub}.

CONSERVATIVE RULE: One sub contact per source connection to a supply, or GND.

LESS CONSERVATIVE: One sub contact per 5-10 transistors.

D. Layout nMOS, pMOS transistors close to GND, VDD rails, respectively and maintain spac between nMOS, pMOS transistors.

25. What are the various design changes you do to meet design power targets?

- Design with Multi-VDD designs, Areas which requires high performance, goes with high VDD and areas which needs low-performance are working with low Vdd's, by creating Voltage-islands and making sure that appropriate level-shifters are placed in the cross-voltage domains
- Designing with Multi-Vt's(threshold voltages), areas which require high performance, goes with low Vt, but takes lot of leakage current, and areas which require low performance with high Vt cells, which has low leakage numbers, by incorporating this design process, we can reduce the leakage power.

 As in the design , clocks consume more amount of power, placing optimal clock-gating cells, in the design and controlling them by the module enable's gives a lot of power-savings.
 As clock-tree's always switch making sure that most number of clock-buffers are after the clock-gating cells, this reduces the switching there by power-reduction.
 Incorporating Dynamic Voltage & Frequency scaling (DVFS) concepts based on the application , there by reducing the systems voltage and frequency numbers when the application does not require to meet the performance targets.
 Ensure the design with IR-Drop analysis and ground-bounce analysis, is with-in the design specification requirement.

 Place power-switches, so that the leakage power can be reduced. related information.

26. what is meant by Library Characterizing: ***"Chip designing is all about Modeling the silicon",*** and how well we characterize the silicon, is all the game. So initially let us assume our process technology is say "32nm", for example: Now we need to develop a test-chip, having modules (digital & analog), and study our silicon timings. Now the toughest job is to generate library views(formats specific to each tool understandable formats).There is a bit of timing in accuracy possible in the views across the formats.

27. what is meant by wireload model:

In the synthesis tool, in order to model the wires we use a concept called as "Wireload models", Now the question is what is wireload models: Wireload models are statistical based on models with respect to fanout. say for a particular technology based on our previous chip experience we have a rough estimate we know if a wire goes for "n" number of fanin then we estimate its delay as say "x" delay units. So a model file is created with the fanout numbers and corresponding estimated delay values. This file is used while performing Synthesis to estimate the delay for Wires, and to estimate the delay for cells, technology specific library model files will be available

28. what are the measures in the Design taken for Meeting Signal-integrity targets

As more and more devices are getting packed, results in more congested areas, and coupling capactiances dominating the wire-capacitance, creates SI violations. Let's see now by what are all the measures we can reduce/solve it.

- As clock-tree runs across the whole chip, optimizing the design for SI, is essential route the clock with double-pitch and triple spacing.
- In-case of SI violation, spacing the signal nets reduces cross-talk impacts.
- Shield the nets with power-nets for high frequency signal nets to prevent from SI.
- Enable SI aware routing , so that the tool takes care for SI
- Ensure SI enabled STA runs, and guarantee the design meeting the SI requirements
- Route signals on different layers orthogonal to each other
- Minimize the parallel run-length wires, by inserting buffers.

29. what are the measures taken in the Design achieving better Yield

Better yield could be achieved by reducing the possibility of manufacturability flaws. Guaranting the circuit performance, by reducing parametric yield, with process variations playing a major role is a big-challenge.

- Create more powerful stringent runset files with pessimistic spacing/short rules.
- Check for the areas where the design is prone to lithographic issues, like sharp cuts and try to re-route it.
- For via-reliability issues, use redundant vias, to reduce the chances for via-breakage.
- In order to design for yield-enhancement , design systems, which could have optimal redundancy, like repairable memories.
- Optimal placing of de-coupling capacitances, reduces the power-surges.
- Doubling the width of the non-critical nets, clock-nets can increase the yield parameter.
- Ensure that the poly-orientation are maintained.

30. what are the measures or precautions to be taken in the Design when the chip has both analog and digital portions

Designing for Optimal integration of Analog and Digital

- As today's IC has analog components also inbuilt , some design practices are required for optimal integration.
- Ensure in the floorplanning stage that the analog block and the digital block are not siting close-by, to reduce the noise.
- Ensure that there exists seperate ground for digital and analog ground to reduce the noise.
- Place appropriate guard-rings around the analog-macro's.
- Incorporating in-built DAC-ADC converters, allows us to test the analog portion using digital testers in an analog loop-back fashion.
- Perform techniques like clock-dithering for the digital portion.

31. what are the steps incorporated for Engineering Change Order[ECO]

As more and more complex the IC design is , and with lot of first time application , is more prone to

last minute changes, there should be provision in the design-flow to accomodate the functional and timing bugs. The step to perform this called as Engineering change order(ECO).

- Ensure that the design has spare functional gates well distributed across the layout.
- Ensure that the selection the spare gates, has many flavours of gates and universal gates, so that any functionality could be achieved.

32. what are the steps performed to achieve Lithography friendly Design

Designing for Manufacturability requires validating the design full-filling lithography rules

- Checking the layout confirming the design rules (spacing,trace-width,shorts).
- Check for the less-congested areas and increasing the spacing of the nets.

33. what does synthesis mean

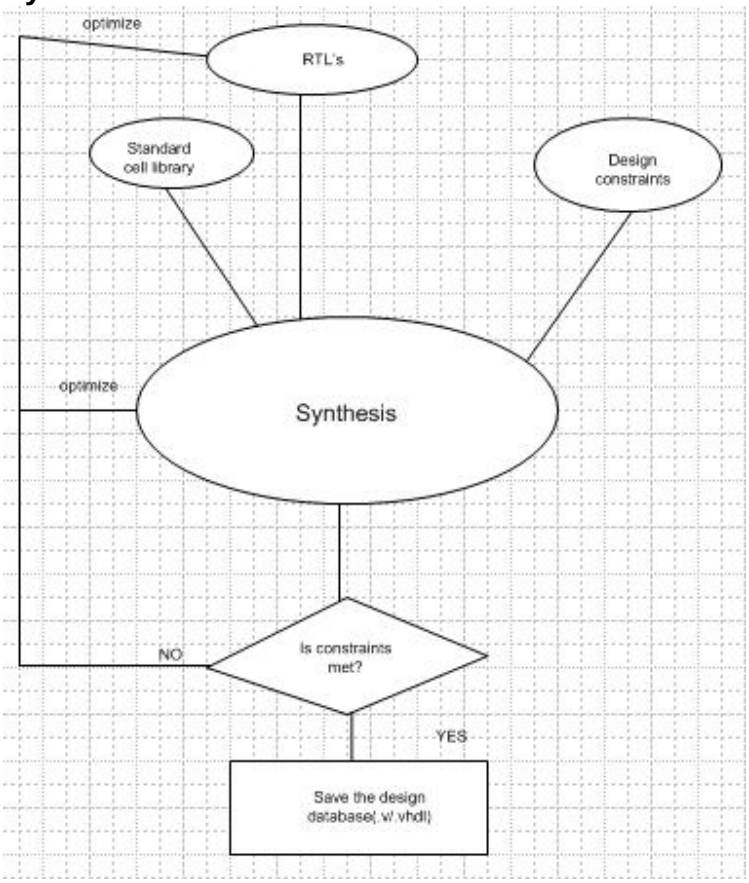
Synthesis is a step of mapping the RTL files (verilog format or vhdl format) to convert it to the technology specific cells..

34. what are the pre-requisties to perform synthesis

1. RTL files
2. Synopsys constraints file, Design constraints file, explaining the priorities of cost functions like area/timing/power
3. Technology specific library files.

35. Explain the Synthesis flow

Synthesis Reference flow



36. What are the various ways to reduce Clock Insertion Delay in the Design

1. Number of Clock sinks
2. Balancing two different clock frequencies
3. Placement of clock sinks.
4. Placement of Clock gating cells
5. Clock tree buffers/inverters drive strength's
6. Clock Transition
7. placement of Clockgating cells and the clock sinks
8. Combinationals cells in the path of clocks (say clock dividers, muxes, clockgates) ...

37. what are the various functional verification methodologies

- TLM(Transaction Level Modelling)
- Linting
- RTL Simulation (Enivronment involving : stimulus generators, monitors, response checkers, transactors)
- Gate level Simulation
- Mixed-signal simulations
- Regression

38. What does formal verification mean?

Formal verification uses Mathematical techniquest by prooving the design through assertions or properties. Correctness of the design can be achieved through assertions with out the necessity for simulations. The methods of formal verification are

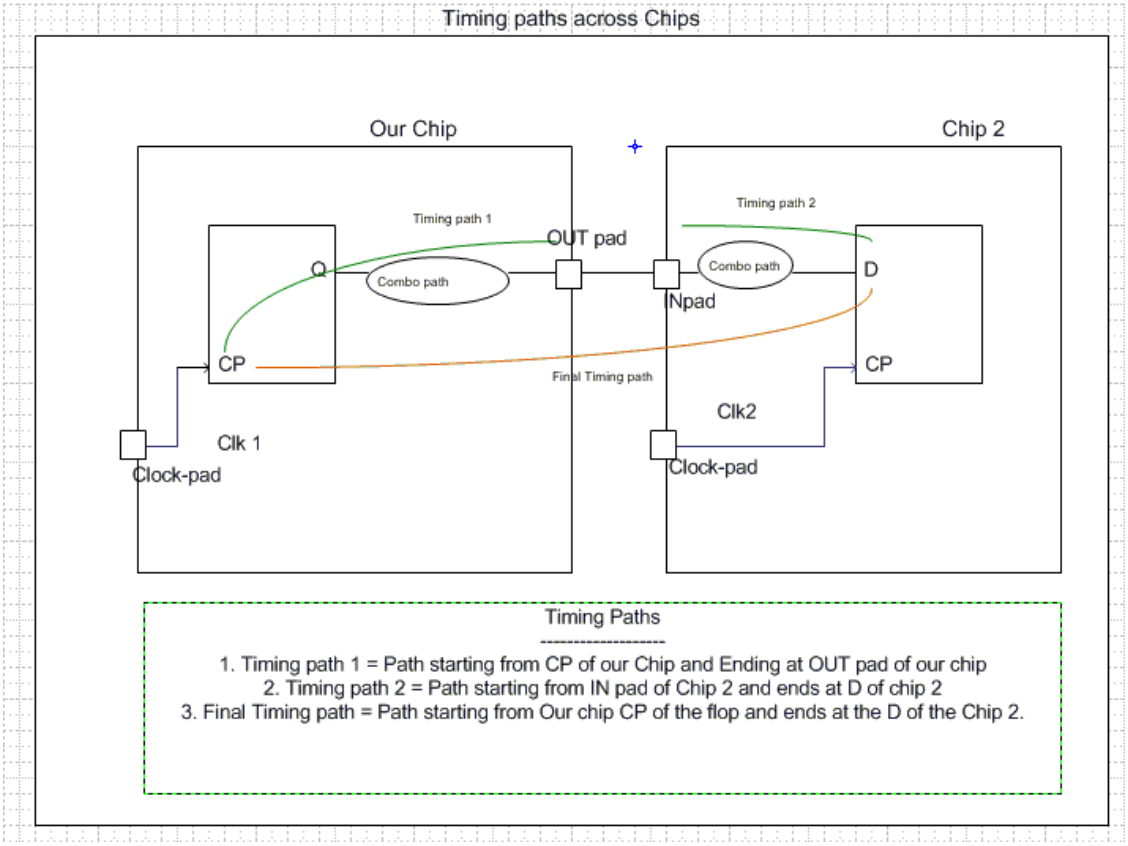
1. Equivalence checking In this method of checking the designs are compared based on mathematical equations and compared whether they are equal or not .

- Original RTL vs Modified RTL
- RTL vs Netlist
- Golden Netlist vs Modified/Edited Netlist
- Synthesis Netlist vs Place and route Netlist

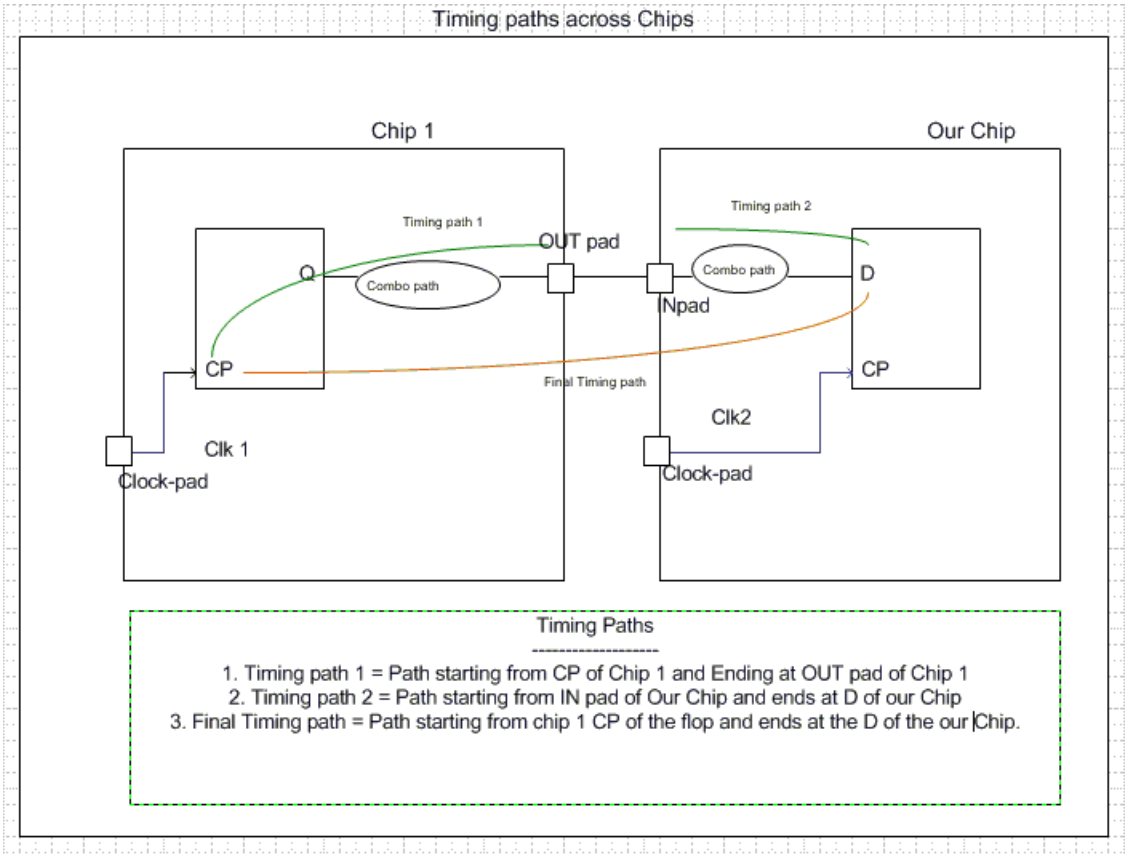
  Remember : Formal verification doesnt check for functionality of the RTL code. It will be only checking the equivalence.

2. Model checking Property specification languages like PSL or SVA, are formally analyzed to see if they are always true for a design. This can exhaustively prove if a property is correct, but does tend to suffer from state-space explosion: the time to analyse a design is directly propotional to the amount of states.

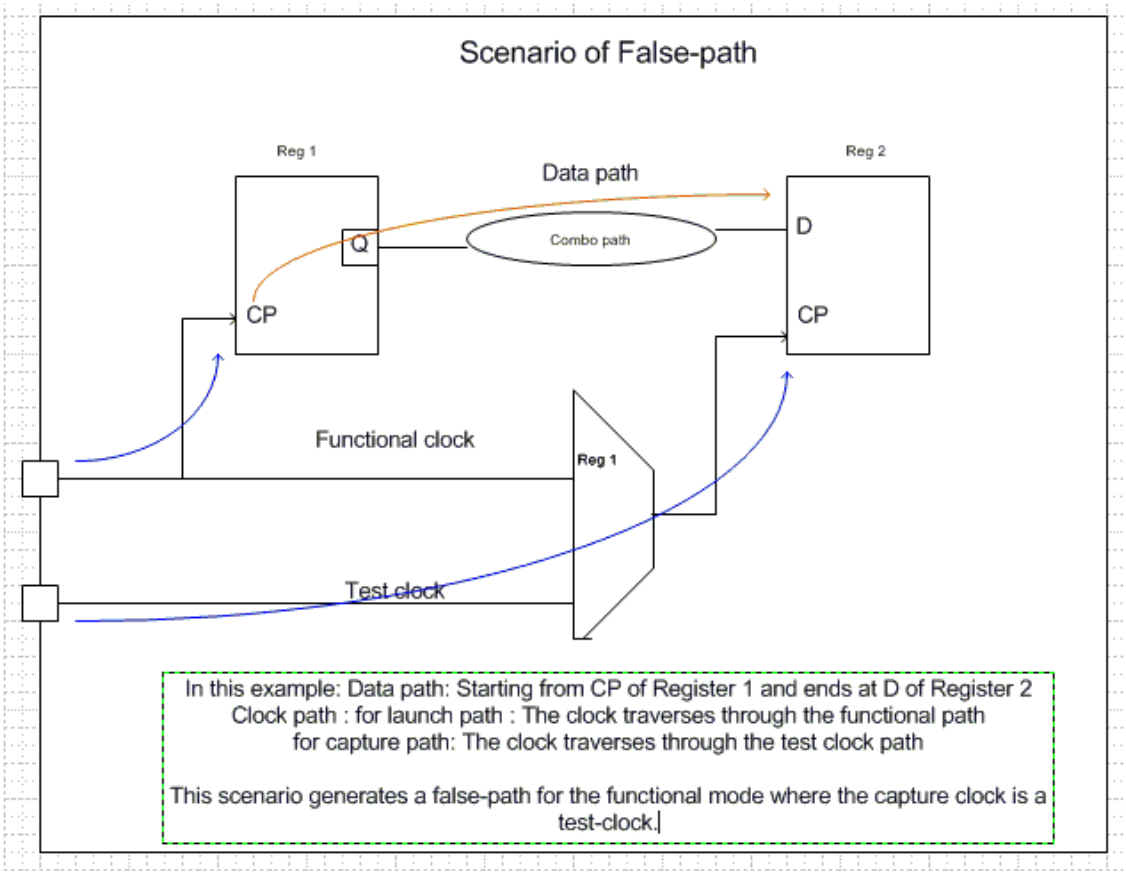
39. How will you time the output paths?



38. How will you time the input paths?



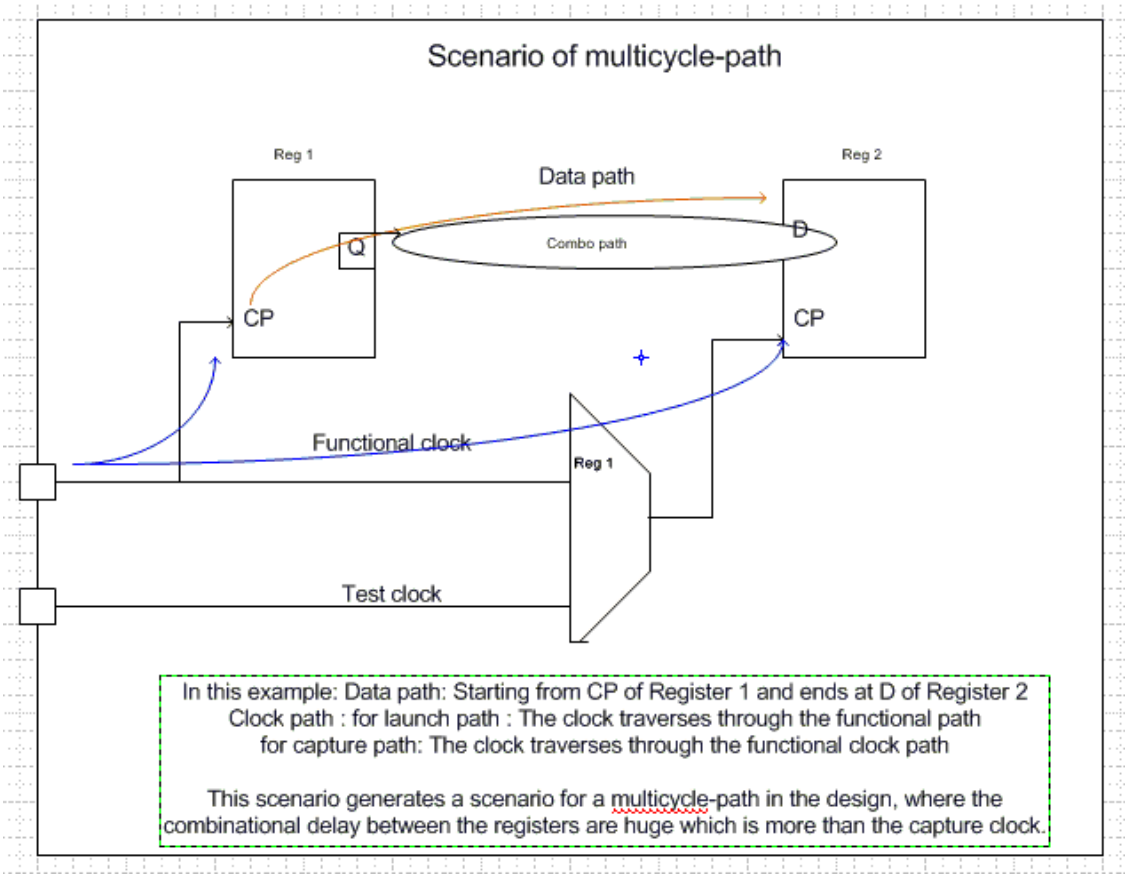
39. what is false path mean in STA and in what scenarios falsepath can come?

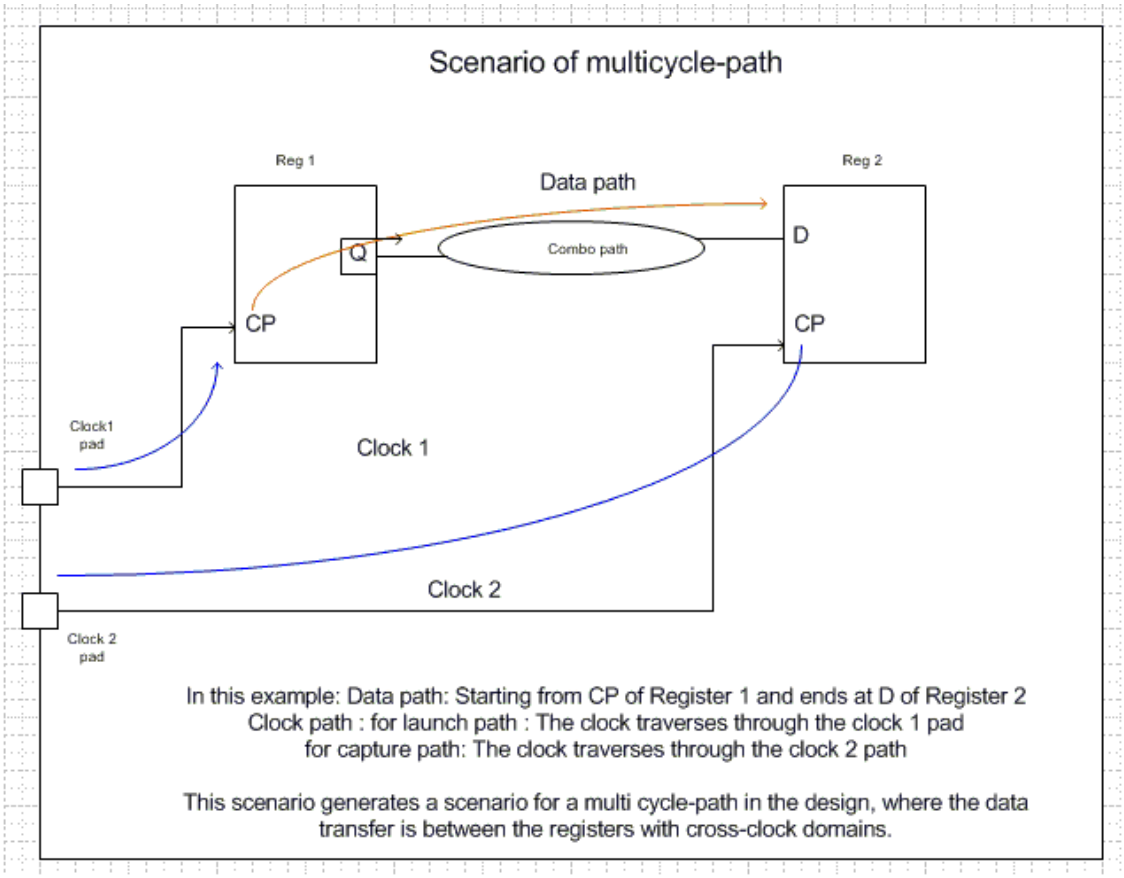


40. what does Multicycle path mean in STA and in what scenarios multicycle paths can come?

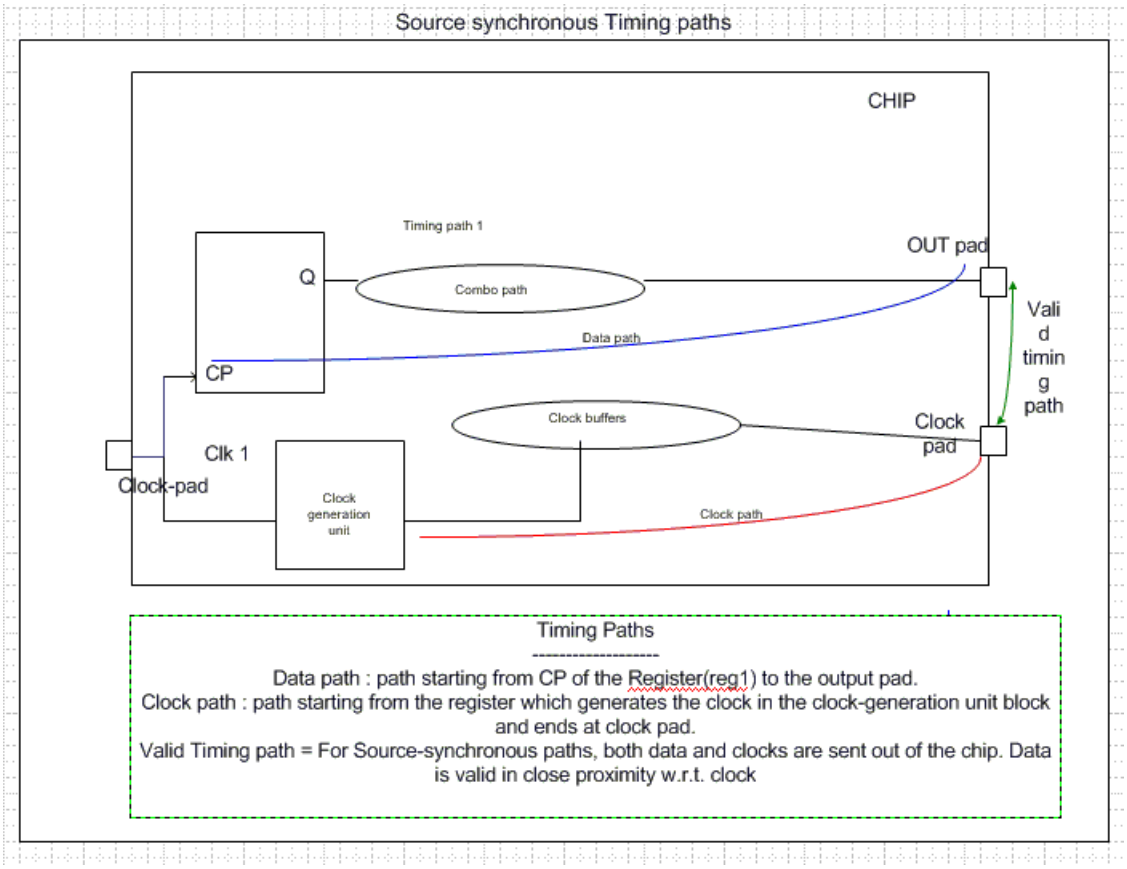
EC &EI

VLSI


Scenario of multicycle-path
Reg 1
Data path
Reg 2
Q
Combo path
D
CP
CP
Functional clock
Reg 1
Test clock
In this example: Data path: Starting from CP of Register 1 and ends at D of Register 2
Clock path : for launch path : The clock traverses through the functional path
for capture path: The clock traverses through the functional clock path
This scenario generates a scenario for a multicycle-path in the design, where the combinational delay between the registers are huge which is more than the capture clock.



41. what are source synchronous paths in STA?



42. assume you have defined latency specified by user both in Master clock and in the Generated clock in STA, how the tool will behave any idea?

If we have defined only Master latency and Generated clock with latency numbers, and the clocks are set to propagated mode after clock-tree, then the Static Timing Analysis Tool, will honour the Generated clock source and Generated clock network latency numbers only and the master clock source and master clock network latencies are ignored.

43. Assume there is a specific requirement to preserve the logic during synthesis, how will do it.

If there is a requirement that some logic needs to be preserved then we can use a command called set_dont_touch or set_dont_design (complete module) and convey the message to the tool not to optimize or smash the logic.

44. We have multiple instances in RTL(Register Transfer Language), do you do anything special during synthesis stage?

While writing RTL(Register Transfer language),say in verilog or in VHDL language, we dont write the same module functionality again and again, we use a concept called as instantiation, where in as per the language, the instanciation of a module will behave like the parent module in terms of functionality, where during synthesis stage we need the full code so that the synthesis tool can study the logic , structure and map it to the library cells, so we use a command in synthesis , called as "UNIQUIFY" which will replace the instantiations with the real logic, because once we are in a synthesis stages we have to visualize as real cells and no more modelling just for functionality alone, we need to visualize in-terms of physical world as well.

45. what do you call an event and when do you call an assertion?

Assertion based Verification Tools, checks whether a statement holds a defined property or not, whereas, Event based Simulators, checks whether there is change in any event, say for every edge of a clock whether there is some activity in a signal or not, in case of an asynchronous designs, checks whether a signal is enabled or not.

1) Explain about setup time and hold time, what will happen if there is setup time and hold tine violation, how to overcome this?

Set up time is the amount of time before the clock edge that the input signal needs to be stable to guarantee it is accepted properly on the clock edge.
Hold time is the amount of time after the clock edge that same input signal has to be held before changing it to make sure it is sensed properly at the clock edge.
Whenever there are setup and hold time violations in any flip-flop, it enters a state where its output is unpredictable: this state is known as metastable state (quasi stable state); at the end of metastable state, the flip-flop settles down to either '1' or '0'. This whole process is known as metastability

46) What is skew, what are problems associated with it and how to minimize it?

In circuit design, clock skew is a phenomenon in synchronous circuits in which the clock signal (sent from the clock circuit) arrives at different components at different times.
This is typically due to two causes. The first is a material flaw, which causes a signal to travel faster or slower than expected. The second is distance: if the signal has to travel the entire length of a circuit, it will likely (depending on the circuit's size) arrive at different parts of the circuit at different times. Clock skew can cause harm in two ways. Suppose that a logic path travels through combinational logic from a source flip-flop to a destination flip-flop. If the destination flip-flop receives the clock tick later than the source flip-flop, and if the logic path delay is short enough, then the data signal might arrive at the destination flip-flop before the clock tick, destroying there the previous data that should have been clocked through. This is called a hold violation because the previous data is not held long enough at the destination flip-flop to be properly clocked through. If the destination flip-flop receives the clock tick earlier than the source flip-flop, then the data signal has that much less time to reach the destination flip-flop before the next clock tick. If it fails to do so, a setup violation occurs, so-called because the new data was not set up and stable before the next clock tick arrived. A hold violation is more serious than a setup violation because it cannot be fixed by increasing the clock period.
Clock skew, if done right, can also benefit a circuit. It can be intentionally introduced to decrease the clock period at which the circuit will operate correctly, and/or to increase the setup or hold safety margins. The optimal set of clock delays is determined by a linear program, in which a setup and a hold constraint appears for each logic path. In this linear program, zero clock skew is merely a feasible point.

Clock skew can be minimized by proper routing of clock signal (clock distribution tree) or putting variable delay buffer so that all clock inputs arrive at the same time

47) What is slack?

'Slack' is the amount of time you have that is measured from when an event 'actually happens' and when it 'must happen'.. The term 'actually happens' can also be taken as being a predicted time for when the event will 'actually happen'. When something 'must happen' can also be called a 'deadline' so another definition of slack would be the time from when something 'actually happens' (call this Tact) until the deadline (call this Tdead).
Slack = Tdead - Tact.
Negative slack implies that the 'actually happen' time is later than the 'deadline' time...in other words it's too late and a timing violation....you have a timing problem that needs some attention.

48) What is glitch? What causes it (explain with waveform)? How to overcome it?

The following figure shows a gated clock. The gated clock's corresponding timing diagram shows that this implementation can lead to clock glitches, which can cause the flip-flop to clock at the wrong time.

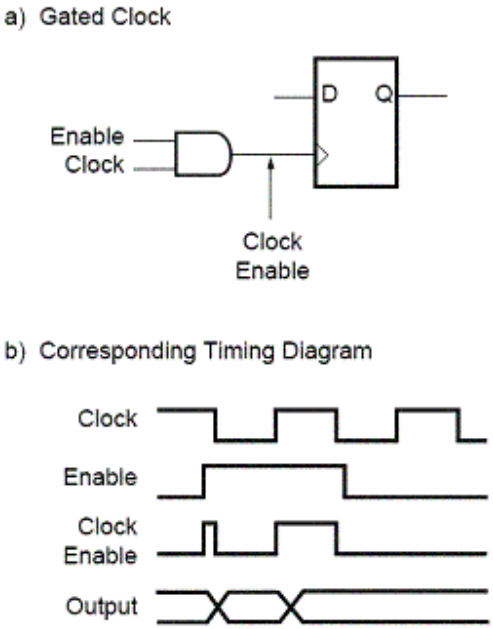
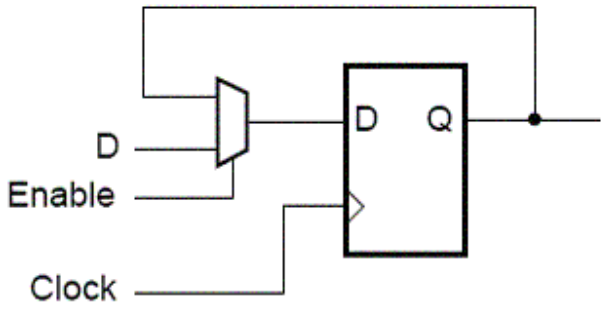


Figure 2-12: **Gated Clock**

The following figure shows a synchronous alternative to the gated clock using a data path. The flip-flop is clocked at every clock cycle and the data path is controlled by an enable. When the enable is Low, the multiplexer feeds the output of the register back on itself. When the enable is High, new data is fed to the flip-flop and the register changes its state

a) Using a Feedback Path



b) Corresponding Timing Diagram

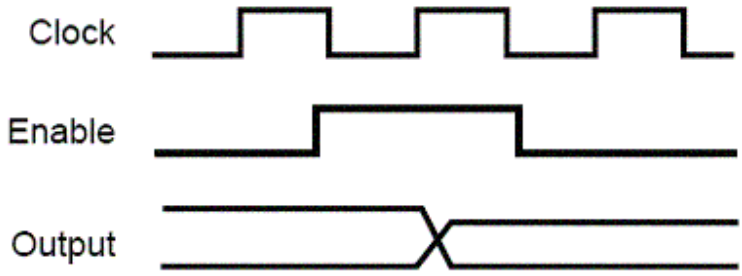


Figure 2-13: **Synchronous Design Using Data Feedback**

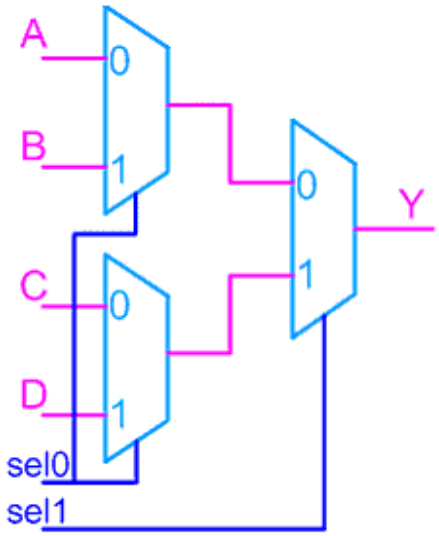
49) Given only two xor gates one must function as buffer and another as inverter?

Tie one of xor gates input to 1 it will act as inverter.
Tie one of xor gates input to 0 it will act as buffer.

50) What is difference between latch and flipflop?

The main difference between latch and FF is that latches are level sensitive while FF are edge sensitive. They both require the use of clock signal and are used in sequential logic. For a latch, the output tracks the input when the clock signal is high, so as long as the clock is logic 1, the output can change if the input also changes. FF on the other hand, will store the input only when there is a rising/falling edge of the clock.

51) Build a 4:1 mux using only 2:1 mux?



52)Difference between heap and stack?

The Stack is more or less responsible for keeping track of what's executing in our code (or what's been "called"). The Heap is more or less responsible for keeping track of our objects (our data, well... most of it - we'll get to that later.).

Think of the Stack as a series of boxes stacked one on top of the next. We keep track of what's going on in our application by stacking another box on top every time we call a method (called a Frame). We can only use what's in the top box on the stack. When we're done with the top box (the method is done executing) we throw it away and proceed to use the stuff in the previous box on the top of the stack. The Heap is similar except that its purpose is to hold information (not keep track of execution most of the time) so anything in our Heap can be accessed at any time. With the Heap, there are no constraints as to what can be accessed like in the stack. The Heap is like the heap of clean laundry on our bed that we have not taken the time to put away yet - we can grab what we need quickly. The Stack is like the stack of shoe boxes in the closet where we have to take off the top one to get to the one underneath it.

53) Difference between mealy and moore state machine?

A) Mealy and Moore models are the basic models of state machines. A state machine which uses only Entry Actions, so that its output depends on the state, is called a Moore model. A state machine which uses only Input Actions, so that the output depends on the state and also on inputs, is called a Mealy model. The models selected will influence a design but there are no general indications as to which model is better. Choice of a model depends on the application, execution means (for instance, hardware systems are usually best realized as Moore models) and personal preferences of a designer or programmer

B) Mealy machine has outputs that depend on the state and input (thus, the FSM has the output written on edges)
Moore machine has outputs that depend on state only (thus, the FSM has the output written in the state itself.

Adv and Disadv
In Mealy as the output variable is a function both input and state, changes of state of the state variables will be delayed with respect to changes of signal level in the input variables, there are possibilities of glitches appearing in the output variables. Moore overcomes glitches as output dependent on only states and not the input signal level.
All of the concepts can be applied to Moore-model state machines because any Moore state machine can be implemented as a Mealy state machine, although the converse is not true.
Moore machine: the outputs are properties of states themselves... which means

that you get the output after the machine reaches a particular state, or to get some output your machine has to be taken to a state which provides you the output.The outputs are held until you go to some other state Mealy machine: Mealy machines give you outputs instantly, that is immediately upon receiving input, but the output is not held after that clock cycle.

54) Difference between onehot and binary encoding?

Common classifications used to describe the state encoding of an FSM are Binary (or highly encoded) and One hot.
A binary-encoded FSM design only requires as many flip-flops as are needed to uniquely encode the number of states in the state machine. The actual number of flip-flops required is equal to the ceiling of the log-base-2 of the number of states in the FSM.
A onehot FSM design requires a flip-flop for each state in the design and only one flip-flop (the flip-flop representing the current or "hot" state) is set at a time in a one hot FSM design. For a state machine with 9- 16 states, a binary FSM only requires 4 flip-flops while a onehot FSM requires a flip-flop for each state in the design
FPGA vendors frequently recommend using a onehot state encoding style because flip-flops are plentiful in an FPGA and the combinational logic required to implement a onehot FSM design is typically smaller than most binary encoding styles. Since FPGA performance is typically related to the combinational logic size of the FPGA design, onehot FSMs typically run faster than a binary encoded FSM with larger combinational logic blocks

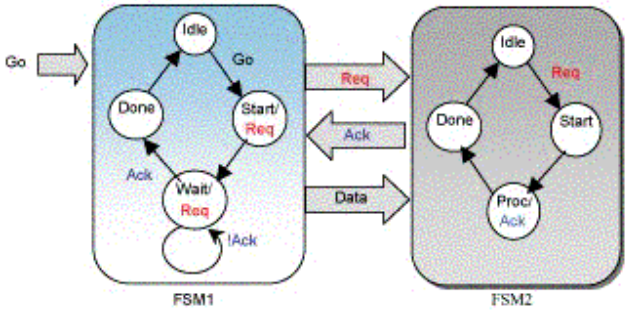
55) What are different ways to synchronize between two clock domains?

Clock Domain Crossing. . .

The following section explains clock domain interfacing

One of the biggest challenges of system-on-chip (SOC) designs is that different blocks operate on independent clocks. Integrating these blocks via the processor bus, memory ports, peripheral busses, and other interfaces can be troublesome because unpredictable behavior can result when the asynchronous interfaces are not properly synchronized

A very common and robust method for synchronizing multiple data signals is a handshake technique as shown in diagram below This is popular because the handshake technique can easily manage changes in clock frequencies, while minimizing latency at the crossing. However, handshake logic is significantly more complex than standard synchronization structures.



FSM1(Transmitter) asserts the req (request) signal, asking the receiver to accept the data on the data bus. FSM2(Receiver) generally a slow module asserts the ack (acknowledge) signal, signifying that it has accepted the data.

it has loop holes: when system Receiver samples the systems Transmitter req line and Transmitter samples system Receiver ack line, they have done it with respect to their internal clock, so there will be setup and hold time violation. To avoid this we go for double or triple stage synchronizers, which increase the MTBF and thus are immune to metastability to a good extent. The figure below shows how this is done.

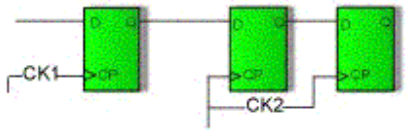


Figure 1a — Single-bit metastability sync

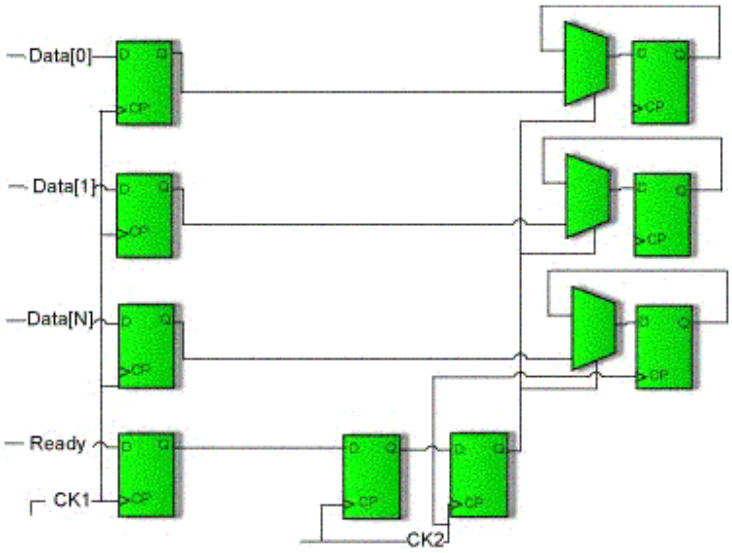


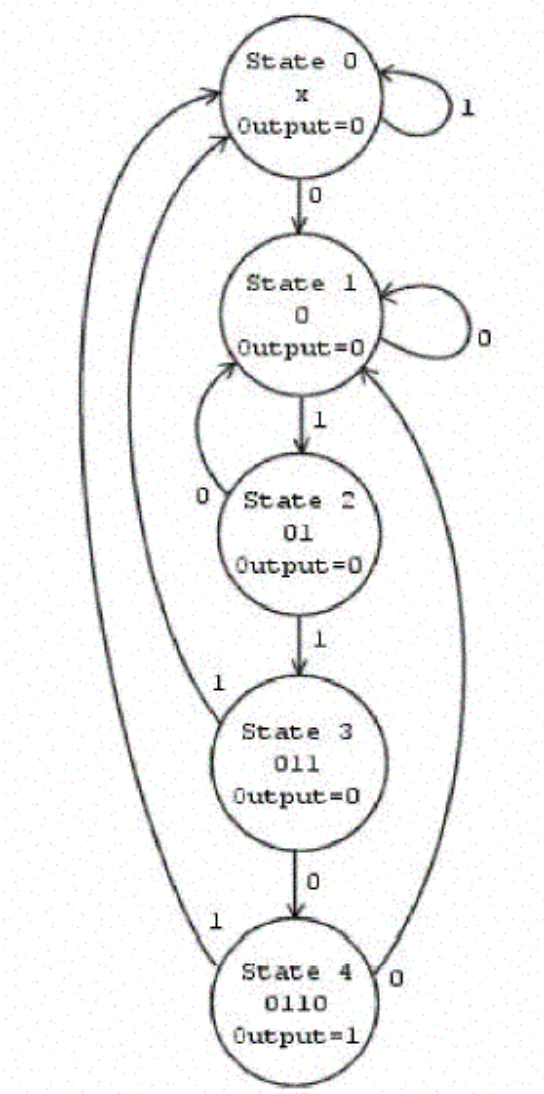
Figure 1b — Multi-bit sync

56) How to calculate maximum operating frequency?

57) How to find out longest path?

You can find answer to this in timing.ppt of presentations section on this site

58) Draw the state diagram to output a "1" for one cycle if the sequence "0110" shows up (the leading 0s cannot be used in more than one sequence)?



59) How to achieve 180 degree exact phase shift?

Never tell using inverter
a) dcm's an inbuilt resource in most of fpga can be configured to get 180 degree phase shift.
b) Bufgds that is differential signaling buffers which are also inbuilt resource of

most of FPGA can be used.

60) What is significance of ras and cas in SDRAM?

SDRAM receives its address command in two address words.
It uses a multiplex scheme to save input pins. The first address word is latched into the DRAM chip with the row address strobe (RAS).
Following the RAS command is the column address strobe (CAS) for latching the second address word.
Shortly after the RAS and CAS strobes, the stored data is valid for reading.

61) Tell some of applications of buffer?

a)They are used to introduce small delays
b)They are used to eliminate cross talk caused due to inter electrode capacitance due to close routing.
c)They are used to support high fanout,eg:bufg

62) Implement an AND gate using mux?

This is the basic question that many interviewers ask. for and gate, give one input as select line,incase if u r giving b as select line, connect one input to logic '0' and other input to a.

63) What will happen if contents of register are shifter left, right?

It is well known that in left shift all bits will be shifted left and LSB will be appended with 0 and in right shift all bits will be shifted right and MSB will be appended with 0 this is a straightforward answer

What is expected is in a left shift value gets Multiplied by 2 eg:consider 0000_1110=14 a left shift will make it 0001_110=28, it the same fashion right shift will Divide the value by 2.

64)Given the following FIFO and rules, how deep does the FIFO need to be to prevent underflow or overflow?

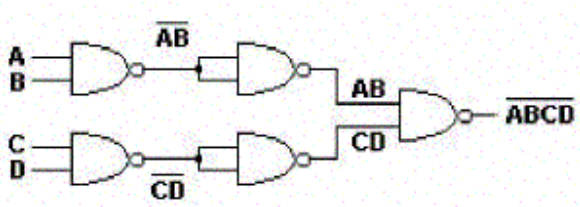
RULES:

1) frequency(clk_A) = frequency(clk_B) / 4
2) period(en_B) = period(clk_A) * 100
3) duty_cycle(en_B) = 25%

Assume clk_B = 100MHz (10ns)
From (1), clk_A = 25MHz (40ns)
From (2), period(en_B) = 40ns * 400 = 4000ns, but we only output for 1000ns,due to (3), so 3000ns of the enable we are doing no output work.
Therefore, FIFO size = 3000ns/40ns = 75 entries.

65) Design a four-input NAND gate using only two-input NAND gates.

A:Basically, you can tie the inputs of a NAND gate together to get an inverter, so...



66)Difference between Synchronous and Asynchronous reset.?

Synchronous reset logic will synthesize to smaller flip-flops, particularly if the reset is gated with the logic generating the d-input. But in such a case, the combinational logic gate count grows, so the overall gate count savings may not be that significant.
The clock works as a filter for small reset glitches; however, if these glitches occur near the active clock edge, the Flip-flop could go metastable. In some designs, the reset must be generated by a set of internal conditions. A

synchronous reset is recommended for these types of designs because it will filter the logic equation glitches between clock.

Disadvantages of synchronous reset:
Problem with synchronous resets is that the synthesis tool cannot easily distinguish the reset signal from any other data signal.
Synchronous resets may need a pulse stretcher to guarantee a reset pulse width wide enough to ensure reset is present during an active edge of the clock[if you have a gated clock to save power, the clock may be disabled coincident with the assertion of reset. Only an asynchronous reset will work in this situation, as the reset might be removed prior to the resumption of the clock.
Designs that are pushing the limit for data path timing, can not afford to have added gates and additional net delays in the data path due to logic inserted to handle synchronous resets.
Asynchronous reset :
The biggest problem with asynchronous resets is the reset release, also called reset removal. Using an asynchronous reset, the designer is guaranteed not to have the reset added to the data path. Another advantage favoring asynchronous resets is that the circuit can be reset with or without a clock present.
Disadvantages of asynchronous reset: ensure that the release of the reset can occur within one clock period. if the release of the reset occurred on or near a clock edge such that the flip-flops went metastable.

67) Why are most interrupts active low?

This answers why most signals are active low
If you consider the transistor level of a module, active low means the capacitor in the output terminal gets charged or discharged based on low to high and high to low transition respectively. when it goes from high to low it depends on the pull down resistor that pulls it down and it is relatively easy for the output capacitance to discharge rather than charging. hence people prefer using active low signals.

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

(a) short the 2 inputs of the nand gate and apply the single input to it.
(b) Connect the output to one of the input and the other to the input signal.

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

set up time: - the amount of time the data should be stable before the application of the clock signal, where as the hold time is the amount of time the data should be stable after the application of the clock. Setup time signifies maximum delay constraints; hold time is for minimum delay constraints. Setup time is critical for establishing the maximum clock frequency.

70) Differences between D-Latch and D flip-flop?

D-latch is level sensitive where as flip-flop is edge sensitive. Flip-flops are made up of latches.

71) What is a multiplexer?

Is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line. (2n =>n).

72)How can you convert an SR Flip-flop to a JK Flip-flop?

By giving the feed back we can convert, i.e !Q=>S and Q=>R.Hence the S and R inputs will act as J and K respectively.

73)How can you convert the JK Flip-flop to a D Flip-flop?

By connecting the J input to the K through the inverter.

74)What is Race-around problem?How can you rectify it?

The clock pulse that remains in the 1 state while both J and K are equal to 1 will cause the output to complement again and repeat complementing until the pulse goes back to 0, this is called the race around problem.To avoid this undesirable operation, the clock pulse must have a time duration that is shorter than the propagation delay time of the F-F, this is restrictive so the alternative is master-slave or edge-triggered construction.

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

XOR each bits of A with B (for e.g. A[0] xor B[0]) and so on.the o/p of 8 xor gates are then given as i/p to an 8-i/p nor gate. if o/p is 1 then A=B.

76)7 bit ring counter's initial state is 0100010. After how many clock cycles will it return to the initial state?

6 cycles

77) Convert D-FF into divide by 2. (not latch) What is the max clock frequency the circuit can handle, given the following information?

T_setup= 6nS T_hold = 2nS T_propagation = 10nS

Circuit: Connect Qbar to D and apply the clk at clk of DFF and take the O/P at Q. It gives freq/2. Max. Freq of operation: 1/ (propagation delay+setup time) = 1/16ns = 62.5 MHz

78)Guys this is the basic question asked most frequently. Design all the basic gates(NOT,AND,OR,NAND,NOR,XOR,XNOR) using 2:1 Multiplexer?

Using 2:1 Mux, (2 inputs, 1 output and a select line)
(a) NOT
Give the input at the select line and connect I0 to 1 & I1 to 0. So if A is 1, we will get I1 that is 0 at the O/P.
(b) AND
Give input A at the select line and 0 to I0 and B to I1. O/p is A & B
(c) OR
Give input A at the select line and 1 to I1 and B to I0. O/p will be A | B
(d) NAND
AND + NOT implementations together
(e) NOR
OR + NOT implementations together
(f) XOR
A at the select line B at I0 and ~B at I1. ~B can be obtained from (a) (g) XNOR
A at the select line B at I1 and ~B at I0

79)N number of XNOR gates are connected in series such that the N inputs (A0,A1,A2.....) are given in the following way: A0 & A1 to first XNOR gate and A2 & O/P of First XNOR to second XNOR gate and so on..... Nth XNOR gates output is final output. How does this circuit work? Explain in detail?

If N=Odd, the circuit acts as even parity detector, ie the output will 1 if there are even number of 1's in the N input...This could also be called as odd parity generator since with this additional 1 as output the total number of 1's will be ODD.
If N=Even, just the opposite, it will be Odd parity detector or Even Parity Generator.

80)An assembly line has 3 fail safe sensors and one emergency shutdown switch.The line should keep moving unless any of the following conditions arise:
(i) If the emergency switch is pressed
(ii) If the senor1 and sensor2 are activated at the same time.
(iii) If sensor 2 and sensor3 are activated at the same time.
(iv) If all the sensors are activated at the same time
Suppose a combinational circuit for above case is to be implemented only with NAND Gates. How many minimum number of 2 input NAND gates are required?

No of 2-input NAND Gates required = 6 You can try the whole implementation.

81)Design a circuit that calculates the square of a number? It should not use any multiplier circuits. It should use Multiplexers and other logic?

This is interesting....
1^2=0+1=1
2^2=1+3=4
3^2=4+5=9
4^2=9+7=16
5^2=16+9=25
and so on
See a pattern yet?To get the next square, all you have to do is add the next odd number to the previous square that you found.See how 1,3,5,7 and finally 9 are added.Wouldn't this be a possible solution to your question since it only will use a counter,multiplexer and a couple of adders?It seems it would take n clock cycles to calculate square of n.

82) How will you implement a Full subtractor from a Full adder?

all the bits of subtrahend should be connected to the xor gate. Other input to the xor being one.The input carry bit to the full adder should be made 1. Then the full

adder works like a full subtractor

83)A very good interview question... What is difference between setup and hold time. The interviewer was looking for one specific reason , and its really a good answer too..The hint is hold time doesn't depend on clock, why is it so...?

Setup violations are related to two edges of clock, i mean you can vary the clock frequency to correct setup violation. But for hold time, you are only concerned with one edge and does not basically depend on clock frequency.

84)In a 3-bit Johnson's counter what are the unused states?

2(power n)-2n is the one used to find the unused states in johnson counter.
So for a 3-bit counter it is 8-6=2.Unused states=2. the two unused states are 010 and 101

85)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 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.

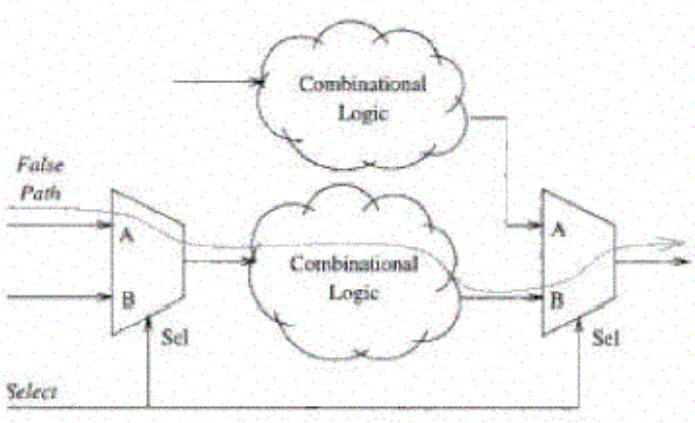
86) What is an LFSR .List a few of its industry applications.?

LFSR is a linear feedback shift register where the input bit is driven by a linear function of the overall shift register value. coming to industrial applications, as far as I know, it is used for encryption and decryption and in BIST(built-in-self-test)

based applications..

87)what is false path?how it determine in ckt? what the effect of false path in ckt?

By timing all the paths in the circuit the timing analyzer can determine all the critical paths in the circuit. However, the circuit may have false paths, which are the paths in the circuit which are never exercised during normal circuit operation for any set of inputs.
An example of a false path is shown in figure below. The path going from the input A of the first MUX through the combinational logic out through the B input of the second MUS is a false path. This path can never be activated since if the A input of the first MUX is activated, then Sel line will also select the A input of the second MUX.
STA (Static Timing Analysis) tools are able to identify simple false paths; however they are not able to identify all the false paths and sometimes report false paths as critical paths. Removal of false paths makes circuit testable and its timing performance predictable (sometimes faster)

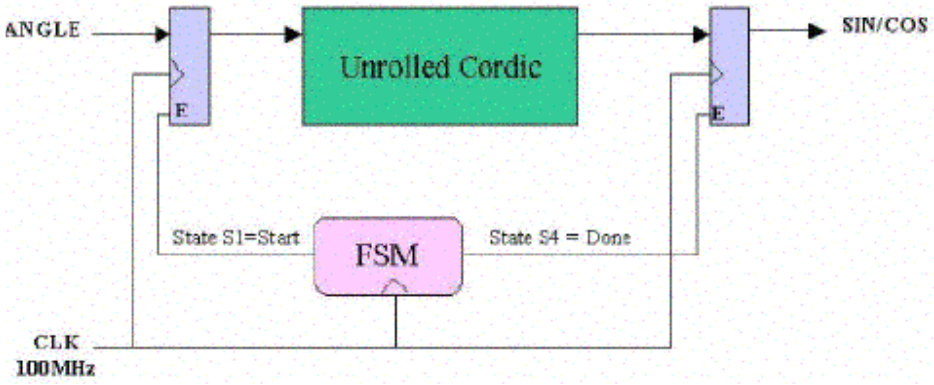


88)Consider two similar processors, one with a clock skew of 100ps and other with a clock skew of 50ps. Which one is likely to have more power? Why?

Clock skew of 50ps is more likely to have clock power. This is because it is likely that low-skew processor has better designed clock tree with more powerful and number of buffers and overheads to make skew better.

89)What are multi-cycle paths?

Multi-cycle paths are paths between registers that take more than one clock cycle to become stable.
For ex. Analyzing the design shown in fig below shows that the output SIN/COS requires 4 clock-cycles after the input ANGLE is latched in. This means that the combinatorial block (the Unrolled Cordic) can take up to 4 clock periods (25MHz) to propagate its result. Place and Route tools are capable of fixing multi-cycle paths problem.



90)You have two counters counting upto 16, built from negedge DFF , First circuit is synchronous and second is "ripple" (cascading), Which circuit has a less propagation delay? Why?

The synchronous counter will have lesser delay as the input to each flop is readily available before the clock edge. Whereas the cascade counter will take long time as the output of one flop is used as clock to the other. So the delay will be propagating. For Eg: 16 state counter = 4 bit counter = 4 Flip flops Let 10ns be the delay of each flop The worst case delay of ripple counter = 10 * 4 = 40ns The delay of synchronous counter = 10ns only.(Delay of 1 flop)

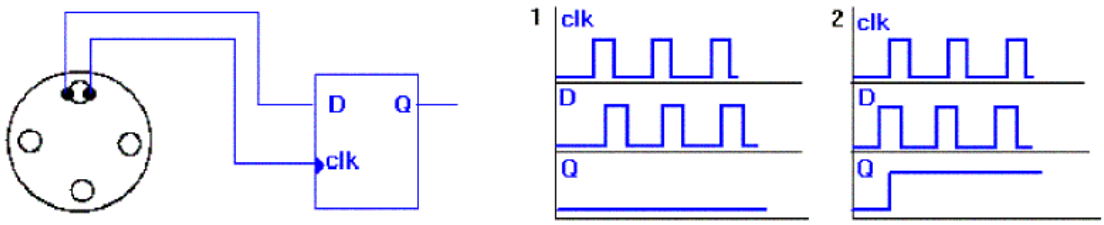
91) what is difference between RAM and FIFO?

FIFO does not have address lines
Ram is used for storage purpose where as fifo is used for synchronization purpose i.e. when two peripherals are working in different clock domains then we

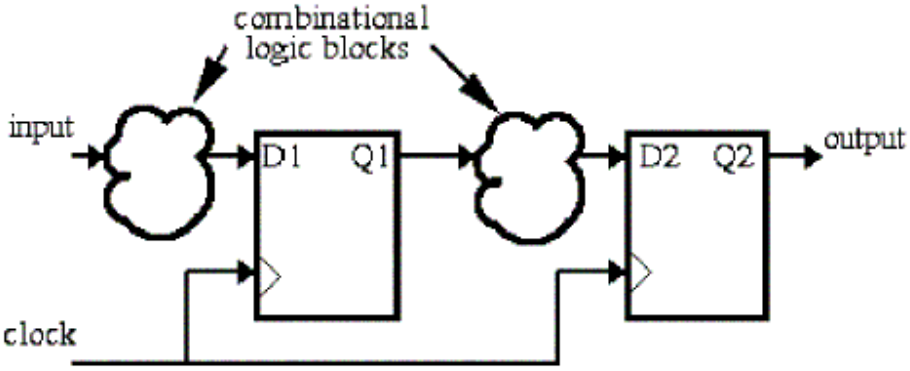
will go for fifo.

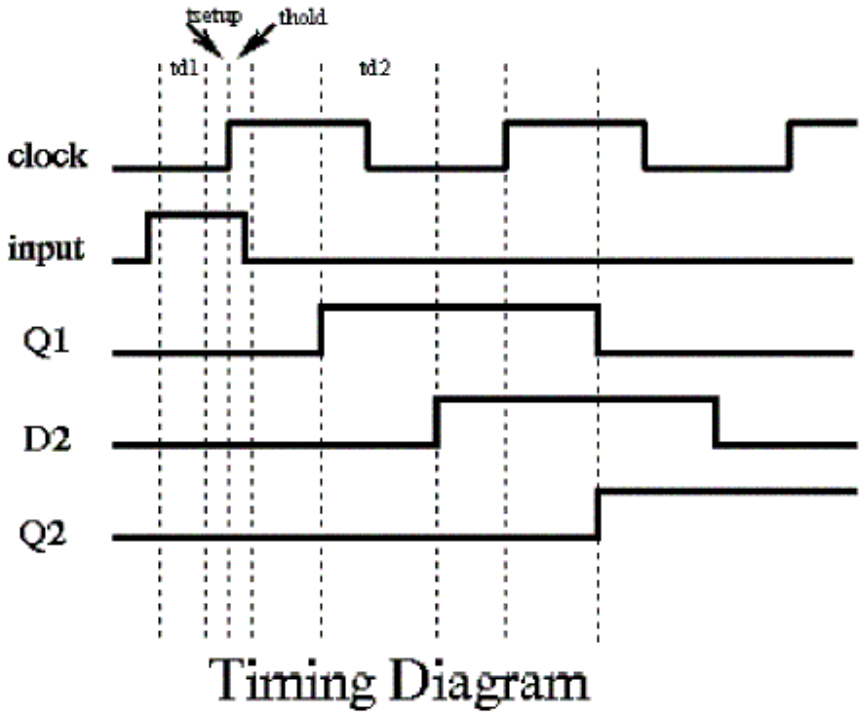
92)The circle can rotate clockwise and back. Use minimum hardware to build a circuit to indicate the direction of rotating.?

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.



93) Draw timing diagrams for following circuit.?





Timing Diagram

94)Implement the following circuits:
(a) 3 input NAND gate using min no of 2 input NAND Gates
(b) 3 input NOR gate using min no of 2 inpur NOR Gates
(c) 3 input XNOR gate using min no of 2 inpur XNOR Gates
Assuming 3 inputs A,B,C?

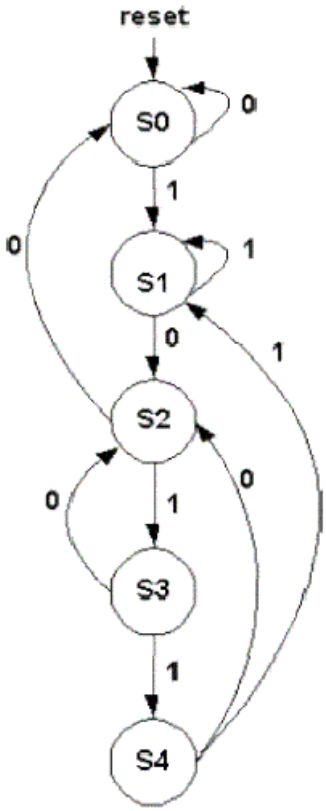
3 input NAND:
Connect :
a) A and B to the first NAND gate
b) Output of first Nand gate is given to the two inputs of the second NAND gate (this basically realizes the inverter functionality)
c) Output of second NAND gate is given to the input of the third NAND gate, whose other input is C
((A NAND B) NAND (A NAND B)) NAND C Thus, can be implemented using '3' 2-input NAND gates. I guess this is the minimum number of gates that need to be used.
3 input NOR:

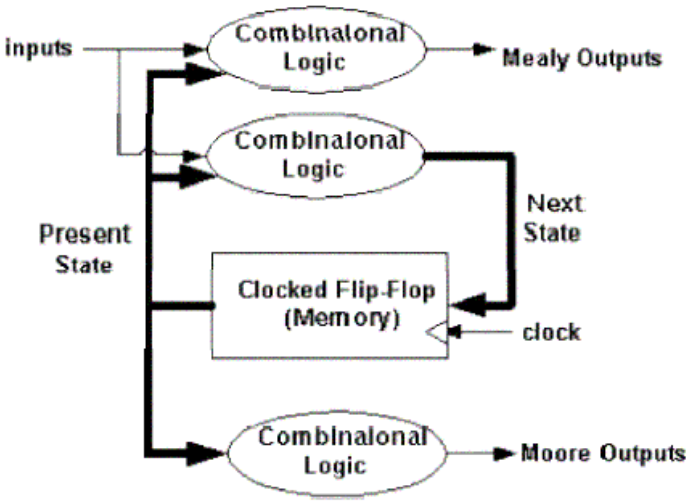
Same as above just interchange NAND with NOR ((A NOR B) NOR (A NOR B)) NOR C
3 input XNOR:
Same as above except the inputs for the second XNOR gate, Output of the first XNOR gate is one of the inputs and connect the second input to ground or logical '0'
((A XNOR B) XNOR 0)) XNOR C

95) Is it possible to reduce clock skew to zero? Explain your answer ?

Even though there are clock layout strategies (H-tree) that can in theory reduce clock skew to zero by having the same path length from each flip-flop from the pll, process variations in R and C across the chip will cause clock skew as well as a pure H-Tree scheme is not practical (consumes too much area).

96)Design a FSM (Finite State Machine) to detect a sequence 10110?

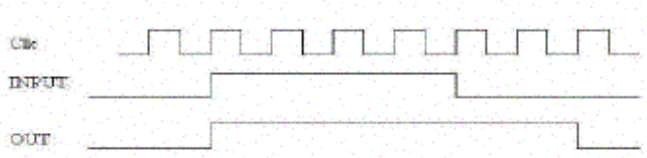


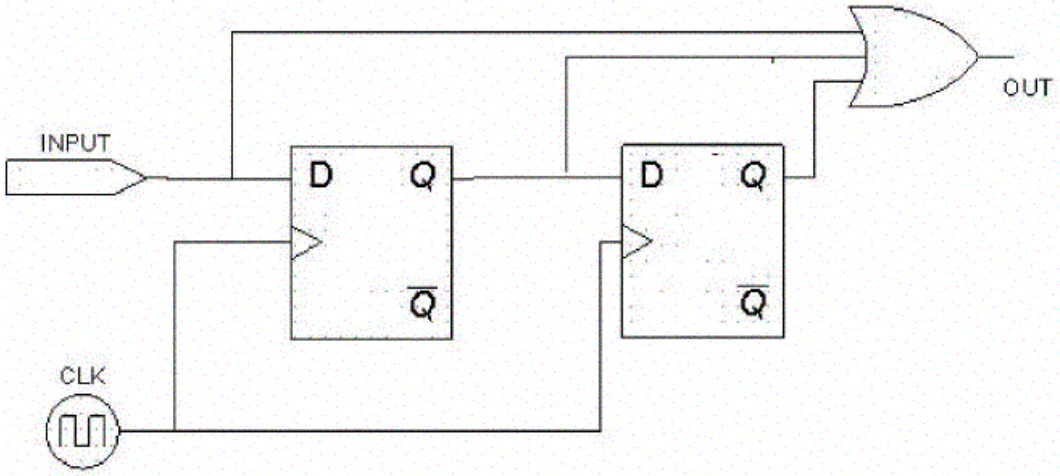


97)Convert D-FF into divide by 2. (not latch)? What is the max clock frequency of the circuit , given the following information?
T_setup= 6nS
T_hold = 2nS
T_propagation = 10nS

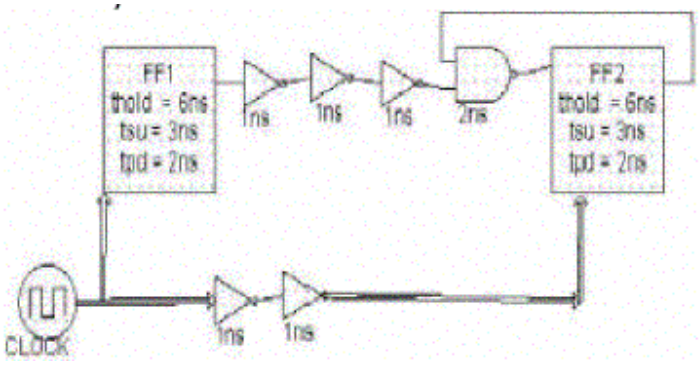
Circuit:
Connect Qbar to D and apply the clk at clk of DFF and take the O/P at Q. It gives freq/2.
Max. Freq of operation:
1/ (propagation delay+setup time) = 1/16ns = 62.5 MHz

98)Give the circuit to extend the falling edge of the input by 2 clock pulses?The waveforms are shown in the following figure.



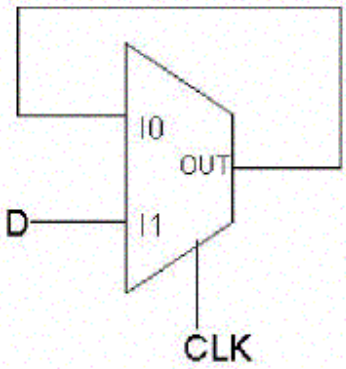


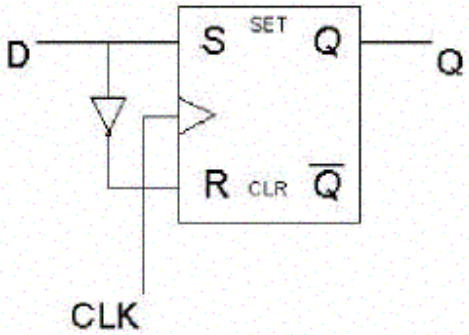
99) For the Circuit Shown below, What is the Maximum Frequency of Operation?Are there any hold time violations for FF2? If yes, how do you modify the circuit to avoid them?



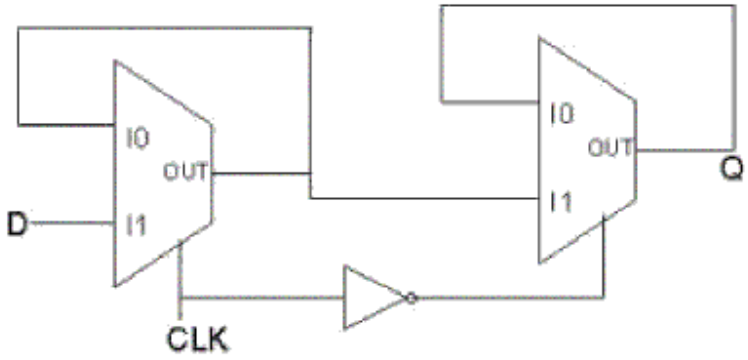
The minumum time period = 3+2+(1+1+1) = 8ns Maximum Frequency = 1/8n= 125MHz.
And there is a hold time violation in the circuit,because of feedback, if you observe, tcq2+AND gate delay is less than thold2,To avoid this we need to use even number of inverters(buffers). Here we need to use 2 inverters each with a delay of 1ns. then the hold time value exactly meets.

100)Design a D-latch using (a) using 2:1 Mux (b) from S-R Latch ?





101)How to implement a Master Slave flip flop using a 2 to 1 mux?

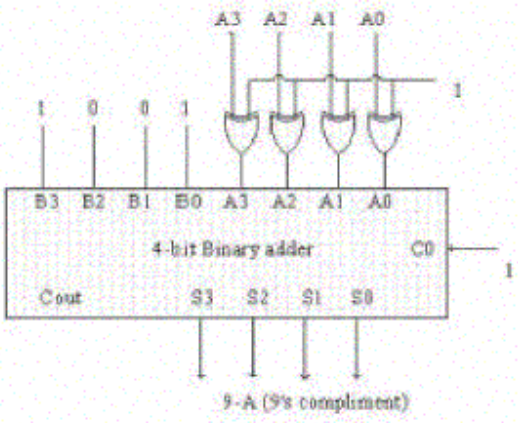


102)how many 2 input xor's are needed to inplement 16 input parity generator ?

It is always n-1 Where n is number of inputs.So 16 input parity generator will require 15 two input xor's .

103)Design a circuit for finding the 9's compliment of a BCD number using 4-bit binary adder and some external logic gates?

9's compliment is nothing but subracting the given no from 9.So using a 4 bit binary adder we can just subract the given binary no from 1001(i.e. 9).Here we can use the 2's compliment method addition.



104) what is Difference between writeback and write through cache?

A caching method in which modifications to data in the cache aren't copied to the cache source until absolutely necessary. Write-back caching is available on many microprocessors , including all Intel processors since the 80486. With these microprocessors, data modifications to data stored in the L1 cache aren't copied to main memory until absolutely necessary. In contrast, a write-through cache performs all write operations in parallel -- data is written to main memory and the L1 cache simultaneously. Write-back caching yields somewhat better performance than write-through caching because it reduces the number of write operations to main memory. With this performance improvement comes a slight risk that data may be lost if the system crashes.
A write-back cache is also called a copy-back cache.

105)Difference between Synchronous,Asynchronous & Isynchronous communication?

Sending data encoded into your signal requires that the sender and receiver are both using the same enconding/decoding method, and know where to look in the signal to find data. Asynchronous systems do not send separate information to indicate the encoding or clocking information. The receiver must decide the clocking of the signal on it's own. This means that the receiver must decide where to look in the signal stream to find ones and zeroes, and decide for itself where each individual bit stops and starts. This information is not in the data in the signal sent from transmitting unit.

Synchronous systems negotiate the connection at the data-link level before communication begins. Basic synchronous systems will synchronize two clocks before transmission, and reset their numeric counters for errors etc. More advanced systems may negotiate things like error correction and compression.

Time-dependent. it refers to processes where data must be delivered within certain time constraints. For example, Multimedia stream require an isochronous transport mechanism to ensure that data is delivered as fast as it is displayed and to ensure that the audio is synchronized with the video.

106) What are different ways Multiply & Divide?

Binary Division by Repeated Subtraction

- Set quotient to zero

- Repeat while dividend is greater than or equal to divisor

Subtract divisor from dividend
Add 1 to quotient

- End of repeat block
- quotient is correct, dividend is remainder
- STOP

Binary Division by Shift and Subtract
Basically the reverse of the mutliply by shift and add.

- **Set** quotient to 0
- Align leftmost digits in dividend and divisor
- **Repeat**

If that portion of the dividend above the divisor is greater than or equal to the divisor
Then subtract divisor from that portion of the dividend and
Concatentate 1 to the right hand end of the quotient
Else concatentate 0 to the right hand end of the quotient
Shift the divisor one place right

- **Until** dividend is less than the divisor
- quotient is correct, dividend is remainder
- **STOP**

Binary Multiply - Repeated Shift and Add
Repeated shift and add - starting with a result of 0, shift the second multiplicand to correspond with each 1 in the first multiplicand and add to the result. Shifting each position left is equivalent to multiplying by 2, just as in decimal representation a shift left is equivalent to multiplying by 10.

- Set result to 0
- Repeat

Shift 2nd multiplicand left until rightmost digit is lined up with leftmost 1 in first multiplicand
Add 2nd multiplicand in that position to result
Remove that 1 from 1st multiplicand

- Until 1st multiplicand is zero
- Result is correct
- STOP

107)What is a SoC (System On Chip), ASIC, "full custom chip", and an FPGA?

There are no precise definitions. Here is my sense of it all. First, 15 years ago, people were unclear on exactly what VLSI meant. Was it 50000 gates? 100000 gates? was is just anything bigger than LSI? My professor simply told me that; VLSI is a level of complexity and integration in a chip that demands Electronic Design Automation tools in order to succeed. In other words, big enough that manually drawing lots of little blue, red and green lines is too much for a human to reasonably do. I think that, likewise, SoC is that level of integration onto a chip that demands more expertise beyond traditional skills of electronics. In other words, pulling off a SoC demands Hardware, Software, and Systems Engineering talent. So, trivially, SoCs aggressively combine HW/SW on a single chip. Maybe more pragmatically, SoC just means that ASIC and Software folks are learning a little bit more about each other's techniques and tools than they did before. Two other interpretations of SoC are 1) a chip that integrates various IP (Intellectual Property) blocks on it and is thus highly centered with issues like Reuse, and 2) a chip integrating multiple classes of electronic circuitry such as Digital CMOS, mixed-signal digital and analog (e.g. sensors, modulators, A/Ds), DRAM memory, high voltage power, etc.

ASIC stands for "Application Specific Integrated Circuit". A chip designed for a specific application. Usually, I think people associate ASICs with the Standard Cell design methodology. Standard Cell design and the typical "ASIC flow" usually means that designers are using Hardware Description Languages, Synthesis and a library of primitive cells (e.g. libraries containing AND, NAND, OR, NOR, NOT, FLIP-FLOP, LATCH, ADDER, BUFFER, PAD cells that are wired together (real libraries are not this simple, but you get the idea..). Design usually is NOT done at a transistor level. There is a high reliance on automated tools because the assumption is that the chip is being made for a SPECIFIC APPLICATION where time is of the essence. But, the chip is manufactured from scratch in that no pre-made circuitry is being programmed or reused. ASIC designer may, or may not, even be aware of the locations of various pieces of circuitry on the chip since the tools do much of the construction, placement and wiring of all the little pieces.

Full Custom, in contrast to ASIC (or Standard Cell), means that every geometric feature going onto the chip being designed (think of those pretty chip pictures we have all seen) is controlled, more or less, by the human design. Automated tools

are certainly used to wire up different parts of the circuit and maybe even manipulate (repeat, rotate, etc.) sections of the chip. But, the human designer is actively engaged with the physical features of the circuitry. Higher human crafting and less reliance on standard cells takes more time and implies higher NRE costs, but lowers RE costs for standard parts like memories, processors, uarts, etc.

FPGAs, or Field Programmable Gate Arrays are completely designed chips that designers load a programming pattern into to achieve a specific digital function. A bit pattern (almost like a software program) is loaded into the already manufactured device which essentially interconnects lots of available gates to meet the designers purposes. FPGAs are sometimes thought of as a “Sea of Gates” where the designer specifies how they are connected. FPGA designers often use many of the same tools that ASIC designers use, even though the FPGA is inherently more flexible. All these things can be intermixed in hybrid sorts of ways. For example, FPGAs are now available that have microprocessor embedded within them which were designed in a full custom manner, all of which now demands “SoC” types of HW/SW integration skills from the designer.

108)What is "Scan" ?

Scan Insertion and ATPG helps test ASICs (e.g. chips) during manufacture. If you know what JTAG boundary scan is, then Scan is the same idea except that it is done inside the chip instead of on the entire board. Scan tests for defects in the chip's circuitry after it is manufactured (e.g. Scan does not help you test whether your Design *functions* as intended). ASIC designers usually implement the scan themselves and occurs just after synthesis. ATPG (Automated Test Pattern Generation) refers to the creation of "Test Vectors" that the Scan circuitry enables to be introduced into the chip. Here's a brief summary:

- Scan Insertion is done by a tool and results in all (or most) of your design's flip-flops to be replaced by special "Scan Flip-flops". Scan flops have additional inputs/outputs that allow them to be configured into a "chain" (e.g. a big shift register) when the chip is put into a test mode.

- The Scan flip-flops are connected up into a chain (perhaps multiple chains)

- The ATPG tool, which knows about the scan chain you've created, generates a series of test vectors.
- The ATPG test vectors include both "Stimulus" and "Expected" bit patterns. These bit vectors are shifted into the chip on the scan chains, and the chips reaction to the stimulus is shifted back out again.
- The ATE (Automated Test Equipment) at the chip factory can put the chip into the scan test mode, and apply the test vectors. If any vectors do not match, then the chip is defective and it is thrown away.
- Scan/ATPG tools will strive to maximize the "coverage" of the ATPG vectors. In other words, given some measure of the total number of nodes in the chip that could be faulty (shorted, grounded, "stuck at 1", "stuck at 0"), what percentage of them can be detected with the ATPG vectors? Scan is a good technology and can achive high coverage in the 90% range.
- Scan testing does not solve all test problems. Scan testing typically does not test memories (no flip-flops!), needs a gate-level netlist to work with, and can take a long time to run on the ATE.
- FPGA designers may be unfamiliar with scan since FPGA testing has already been done by the FPGA manufacturer. ASIC designers do not have this luxury and must handle all the manufacturing test details themselves.

109**. what is Body effect ?**

The threshold voltage of a MOSFET is affected by the voltage which is applied to the back contact. The voltage difference between the source and the bulk, VBS changes the width of the depletion layer and therefore also the voltage across the oxide due to the change of the charge in the depletion region. This results in a difference in threshold voltage which equals the difference in charge in the depletion region divided by the oxide capacitance, yielding:.

$$\Delta V_T = \frac{\sqrt{2\varepsilon_s q N_a}}{C_{OX}} \left(\sqrt{(2\phi_F + V_{SB})} - \sqrt{2\phi_F} \right)$$

110.What are stansdard Cell's?

In semiconductor design, standard cell methodology is a method of designing Application Specific Integrated Circuits (ASICs) with mostly digital-logic features. Standard cell methodology is an example of design abstraction, whereby a low-level VLSI-layout is encapsulated into an abstract logic representation (such as a NAND gate). Cell-based methodology (the general class that standard-cell belongs to) makes it possible for one designer to focus on the high-level (logical function) aspect of digital-design, while another designer focused on the implementation (physical) aspect. Along with semiconductor manufacturing advances, standard cell methodology was responsible for allowing designers to scale ASICs from comparatively simple single-function ICs (of several thousand gates), to complex multi-million gate devices (SoC).

111.What are Design Rule Check (DRC) and Layout Vs Schematic (LVS) ?

Design Rule Check (DRC) and Layout Vs Schematic (LVS) are verification processes. Reliable device fabrication at modern deep submicrometre (0.13 µm and below) requires strict observance of transistor spacing, metal layer thickness, and power density rules. DRC exhaustively compares the physical netlist against a set of "foundry design rules" (from the foundry operator), then flags any observed violations. LVS is a process that confirms that the layout has the same structure as the associated schematic; this is typically the final step in the layout process. The LVS tool takes as an input a schematic diagram and the extracted view from a layout. It then generates a netlist from each one and compares them. Nodes, ports, and device sizing are all compared. If they are the same, LVS passes and the designer can continue. Note: LVS tends to consider transistor fingers to be the same as an extra-wide transistor. For example, 4 transistors in parallel (each 1 um wide), a 4-finger 1 um transistor, and a 4 um transistor are all seen as the same by the LVS tool. Functionality of .lib files will be taken from spice models and added as an attribute to the .lib file.

112.What is Antenna effect ?

The antenna effect, more formally plasma induced gate oxide damage, is an

efffect that can potentially cause yield and reliability problems during the manufacture of MOS integrated circuits[1][2][3][4][5]. Fabs normally supply antenna rules, which are rules that must be obeyed to avoid this problem. A violation of such rules is called an antenna violation. The word antenna is somewhat of a misnomer in this context—the problem is really the collection of charge, not the normal meaning of antenna, which is a device for converting electromagnetic fields to/from electrical currents. Occasionally the phrase antenna effect is used this context[6] but this is less common since there are many effects[7] and the phrase does not make clear which is meant.

113.What are steps involved in Semiconductor device fabrication ?

This is a list of processing techniques that are employed numerous times in a modern electronic device and do not necessarily imply a specific order.

Wafer processing
Wet cleans
Photolithography
Ion implantation (in which dopants are embedded in the wafer creating regions of increased (or decreased) conductivity)
Dry etching
Wet etching
Plasma ashing
Thermal treatments
Rapid thermal anneal
Furnace anneals
Thermal oxidation
Chemical vapor deposition (CVD)
Physical vapor deposition (PVD)
Molecular beam epitaxy (MBE)
Electrochemical Deposition (ECD). See Electroplating
Chemical-mechanical planarization (CMP)
Wafer testing (where the electrical performance is verified)
Wafer backgrinding (to reduce the thickness of the wafer so the resulting chip can be put into a thin device like a smartcard or PCMCIA card.)
Die preparation
Wafer mounting
Die cutting
IC packaging

Die attachment
IC Bonding
Wire bonding
Flip chip
Tab bonding
IC encapsulation
Baking
Plating
Lasermarking
Trim and form
IC testing

114.What is Clock distribution network ?

In a synchronous digital system, the clock signal is used to define a time reference for the movement of data within that system. The clock distribution network distributes the clock signal(s) from a common point to all the elements that need it. Since this function is vital to the operation of a synchronous system, much attention has been given to the characteristics of these clock signals and the electrical networks used in their distribution. Clock signals are often regarded as simple control signals; however, these signals have some very special characteristics and attributes.
Clock signals are typically loaded with the greatest fanout, travel over the greatest distances, and operate at the highest speeds of any signal, either control or data, within the entire synchronous system. Since the data signals are provided with a temporal reference by the clock signals, the clock waveforms must be particularly clean and sharp. Furthermore, these clock signals are particularly affected by technology scaling (see Moore's law), in that long global interconnect lines become significantly more resistive as line dimensions are decreased. This increased line resistance is one of the primary reasons for the increasing significance of clock distribution on synchronous performance. Finally, the control of any differences and uncertainty in the arrival times of the clock signals can severely limit the maximum performance of the entire system and create catastrophic race conditions in which an incorrect data signal may latch within a register. The clock distribution network often takes a significant fraction of the power consumed by a chip. Furthermore, significant power can be wasted in transitions within blocks, even when their output is not needed. These observations have lead to a power saving technique called clock gating, which

involves adding logic gates to the clock distribution tree, so portions of the tree can be turned off when not needed.

115.What is Clock Gating ?

Clock gating is one of the power-saving techniques used on many synchronous circuits including the Pentium 4 processor. To save power, clock gating refers to adding additional logic to a circuit to prune the clock tree, thus disabling portions of the circuitry where flip flops do not change state. Although asynchronous circuits by definition do not have a "clock", the term "perfect clock gating" is used to illustrate how various clock gating techniques are simply approximations of the data-dependent behavior exhibited by asynchronous circuitry, and that as the granularity on which you gate the clock of a synchronous circuit approaches zero, the power consumption of that circuit approaches that of an asynchronous circuit.

116**.What is Netlist ?**

Netlists are connectivity information and provide nothing more than instances, nets, and perhaps some attributes. If they express much more than this, they are usually considered to be a hardware description language such as Verilog, VHDL, or any one of several specific languages designed for input to simulators. Most netlists either contain or refer to descriptions of the parts or devices used. Each time a part is used in a netlist, this is called an "instance." Thus, each instance has a "master", or "definition". These definitions will usually list the connections that can be made to that kind of device, and some basic properties of that device. These connection points are called "ports" or "pins", among several other names.
An "instance" could be anything from a vacuum cleaner, microwave oven, or light bulb, to a resistor, capacitor, or integrated circuit chip.
Instances have "ports". In the case of a vacuum cleaner, these ports would be the three metal prongs in the plug. Each port has a name, and in continuing the vacuum cleaner example, they might be "Neutral", "Live" and "Ground". Usually, each instance will have a unique name, so that if you have two instances of vacuum cleaners, one might be "vac1" and the other "vac2". Besides their names, they might otherwise be identical.
Nets are the "wires" that connect things together in the circuit. There may or may not be any special attributes associated with the nets in a design, depending on

the particular language the netlist is written in, and that language's features. Instance based netlists usually provide a list of the instances used in a design. Along with each instance, either an ordered list of net names are provided, or a list of pairs provided, of an instance port name, along with the net name to which that port is connected. In this kind of description, the list of nets can be gathered from the connection lists, and there is no place to associate particular attributes with the nets themselves. SPICE is perhaps the most famous of instance-based netlists.
Net-based netlists usually describe all the instances and their attributes, then describe each net, and say which port they are connected on each instance. This allows for attributes to be associated with nets. EDIF is probably the most famous of the net-based netlists.

117.What Physical timing closure ?

Physical timing closure is the process by which an FPGA or a VLSI design with a physical representation is modified to meet its timing requirements. Most of the modifications are handled by EDA tools based on directives given by a designer. The term is also sometimes used as a characteristic, which is ascribed to an EDA tool, when it provides most of the features required in this process. Physical timing closure became more important with submicrometre technologies, as more and more steps of the design flow had to be made timing-aware. Previously only logic synthesis had to satisfy timing requirements. With present deep submicrometre technologies it is unthinkable to perform any of the design steps of placement, clock-tree synthesis and routing without timing constraints. Logic synthesis with these technologies is becoming less important. It is still required, as it provides the initial netlist of gates for the placement step, but the timing requirements do not need to be strictly satisfied any more. When a physical representation of the circuit is available, the modifications required to achieve timing closure are carried out by using more accurate estimations of the delays.

118.What Physical verification ?

Physical verification of the design, involves DRC(Design rule check), LVS(Layout versus schematic) Check, XOR Checks, ERC (Electrical Rule Check) and Antenna Checks.

XOR Check

This step involves comparing two layout databases/GDS by XOR operation of the layout geometries. This check results a database which has all the mismatching geometries in both the layouts. This check is typically run after a metal spin, where in the re-spin database/GDS is compared with the previously taped out database/GDS.
Antenna Check

Antenna checks are used to limit the damage of the thin gate oxide during the manufacturing process due to charge accumulation on the interconnect layers (metal, polysilicon) during certain fabrication steps like Plasma etching, which creates highly ionized matter to etch. The antenna basically is a metal interconnect, i.e., a conductor like polysilicon or metal, that is not electrically connected to silicon or grounded, during the processing steps of the wafer. If the connection to silicon does not exist, charges may build up on the interconnect to the point at which rapid discharge does take place and permanent physical damage results to thin transistor gate oxide. This rapid and destructive phenomenon is known as the antenna effect. The Antenna ratio is defined as the ratio between the physical area of the conductors making up the antenna to the total gate oxide area to which the antenna is electrically connected.
ERC (Electrical rule check)

ERC (Electrical rule check) involves checking a design for all well and substrate areas for proper contacts and spacings thereby ensuring correct power and ground connections. ERC steps can also involve checks for unconnected inputs or shorted outputs.

119.What is Stuck-at fault ?

A Stuck-at fault is a particular fault model used by fault simulators and Automatic test pattern generation (ATPG) tools to mimic a manufacturing defect within an integrated circuit. Individual signals and pins are assumed to be stuck at Logical '1', '0' and 'X'. For example, an output is tied to a logical 1 state during test generation to assure that a manufacturing defect with that type of behavior can be found with a specific test pattern. Likewise the output could be tied to a logical 0 to model the behavior of a defective circuit that cannot switch its output pin.

120.What is Different Logic family ?

Listed here in rough chronological order of introduction along with their usual abbreviations of Logic family
* Diode logic (DL)
* Direct-coupled transistor logic (DCTL)
* Complementary transistor logic (CTL)
* Resistor-transistor logic (RTL)
* Resistor-capacitor transistor logic (RCTL)
* Diode-transistor logic (DTL)
* Emitter coupled logic (ECL) also known as Current-mode logic (CML)
* Transistor-transistor logic (TTL) and variants
* P-type Metal Oxide Semiconductor logic (PMOS)
* N-type Metal Oxide Semiconductor logic (NMOS)
* Complementary Metal-Oxide Semiconductor logic (CMOS)
* Bipolar Complementary Metal-Oxide Semiconductor logic (BiCMOS)
* Integrated Injection Logic (I2L)

121.What is Different Types of IC packaging ?

IC are packaged in many types they are: * BGA1
* BGA2
* Ball grid array
* CPGA
* Ceramic ball grid array
* Cerquad
* DIP-8
* Die attachment
* Dual Flat No Lead
* Dual in-line package
* Flat pack
* Flip chip
* Flip-chip pin grid array
* HVQFN
* LQFP
* Land grid array
* Leadless chip carrier
* Low insertion force
* Micro FCBGA
* Micro Leadframe Package

* MicroLeadFrame
* Mini-Cartridge
* Multi-Chip Module
* OPGA
* PQFP
* Package on package
* Pin grid array
* Plastic leaded chip carrier
* QFN
* QFP
* Quadruple in-line package
* ROM cartridge
* Shrink Small-Outline Package
* Single in-line package
* Small-Outline Integrated Circuit
* Staggered Pin Grid Array
* Surface-mount technology
* TO220
* TO3
* TO92
* TQFP
* TSSOP
* Thin small-outline package
* Through-hole technology
* UICC
* Zig-zag in-line package

122.What is Substrate coupling ?

In an integrated circuit, a signal can couple from one node to another via the substrate. This phenomenon is referred to as substrate coupling or substrate noise coupling.
The push for reduced cost, more compact circuit boards, and added customer features has provided incentives for the inclusion of analog functions on primarily digital MOS integrated circuits (ICs) forming mixed-signal ICs. In these systems, the speed of digital circuits is constantly increasing, chips are becoming more densely packed, interconnect layers are added, and analog resolution is increased. In addition, recent increase in wireless applications and its growing market are introducing a new set of aggressive design goals for realizing mixed-

signal systems. Here, the designer integrates radio frequency (RF) analog and base band digital circuitry on a single chip. The goal is to make single-chip radio frequency integrated circuits (RFICs) on silicon, where all the blocks are fabricated on the same chip. One of the advantages of this integration is low power dissipation for portability due to a reduction in the number of package pins and associated bond wire capacitance. Another reason that an integrated solution offers lower power consumption is that routing high-frequency signals off-chip often requires a 50O impedance match, which can result in higher power dissipation. Other advantages include improved high-frequency performance due to reduced package interconnect parasitics, higher system reliability, smaller package count, smaller package interconnect parasitics, and higher integration of RF components with VLSI-compatible digital circuits. In fact, the single-chip transceiver is now a reality.

123.What is Latchup ?

A latchup is the inadvertent creation of a low-impedance path between the power supply rails of an electronic component, triggering a parasitic structure, which then acts as a short circuit, disrupting proper functioning of the part and possibly even leading to its destruction due to overcurrent. A power cycle is required to correct this situation. The parasitic structure is usually equivalent to a thyristor (or SCR), a PNPN structure which acts as a PNP and an NPN transistor stacked next to each other. During a latchup when one of the transistors is conducting, the other one begins conducting too. They both keep each other in saturation for as long as the structure is forward-biased and some current flows through it - which usually means until a power-down. The SCR parasitic structure is formed as a part of the totem-pole PMOS and NMOS transistor pair on the output drivers of the gates.

124) What is latch up?

Latch-up pertains to a failure mechanism wherein a parasitic thyristor (such as a parasitic silicon controlled rectifier, or SCR) is inadvertently created within a circuit, causing a high amount of current to continuously flow through it once it is accidentally triggered or turned on. Depending on the circuits involved, the

amount of current flow produced by this mechanism can be large enough to result in permanent destruction of the device due to electrical overstress (EOS) .

125)Why is NAND gate preferred over NOR gate for fabrication?

NAND is a better gate for design than NOR because at the transistor level the mobility of electrons is normally three times that of holes compared to NOR and thus the NAND is a faster gate.
Additionally, the gate-leakage in NAND structures is much lower. If you consider t_phl and t_plh delays you will find that it is more symmetric in case of NAND (the delay profile), but for NOR, one delay is much higher than the other(obviously t_plh is higher since the higher resistance p mos's are in series connection which again increases the resistance).

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

The minimum amount of noise that can be allowed on the input stage for which the output will not be effected.

127)Explain sizing of the inverter?

In order to drive the desired load capacitance we have to increase the size (width) of the inverters to get an optimized performance.

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

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

The minimum amount of noise that can be allowed on the input stage for which the output will not be effected.

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

delay increases.

131)What happens to delay if we include a resistance at the output of a

CMOS circuit?

Increases. (RC delay)

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

The delay can be reduced by increasing the power supply but if we do so the heating effect comes because of excessive power, to compensate this we have to increase the die size which is not practical.

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

R = (*l) / A.

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

Power dissipation=CV2f ,from this minimize the load capacitance, dc voltage and the operating frequency.

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

In the serially connected NMOS logic the input capacitance of each gate shares the charge with the load capacitance by which the logical levels drastically mismatched than that of the desired once. To eliminate this load capacitance must be very high compared to the input capacitance of the gates (approximately 10 times).

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

Because it can not drive the output load straight away, so we gradually increase the size to get an optimized performance.

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

Latch-up is a condition in which the parasitic components give rise to the Establishment of low resistance conducting path between VDD and VSS with Disastrous results.

138) Give the expression for CMOS switching power dissipation?

CV2

139) What is Body Effect?

In general multiple MOS devices are made on a common substrate. As a result, the substrate voltage of all devices is normally equal. However while connecting the devices serially this may result in an increase in source-to-substrate voltage as we proceed vertically along the series chain (Vsb1=0, Vsb2 0).Which results Vth2>Vth1.

140) Why is the substrate in NMOS connected to Ground and in PMOS to VDD?

we try to reverse bias not the channel and the substrate but we try to maintain the drain,source junctions reverse biased with respect to the substrate so that we dont loose our current into the substrate.

141) What is the fundamental difference between a MOSFET and BJT ?

In MOSFET, current flow is either due to electrons(n-channel MOS) or due to holes(p-channel MOS) - In BJT, we see current due to both the carriers.. electrons and holes. BJT is a current controlled device and MOSFET is a voltage controlled device.

142)Which transistor has higher gain. BJT or MOS and why?

BJT has higher gain because it has higher transconductance.This is because the current in BJT is exponentially dependent on input where as in MOSFET it is square law.

143)Why do we gradually increase the size of inverters in buffer design when trying to drive a high capacitive load? Why not give the output of a

circuit to one large inverter?

We cannot use a big inverter to drive a large output capacitance because, who will drive the big inverter? The signal that has to drive the output cap will now see a larger gate capacitance of the BIG inverter.So this results in slow raise or fall times .A unit inverter can drive approximately an inverter thats 4 times bigger in size. So say we need to drive a cap of 64 unit inverter then we try to keep the sizing like say 1,4,16,64 so that each inverter sees a same ratio of output to input cap. This is the prime reason behind going for progressive sizing.

144)In CMOS technology, in digital design, why do we design the size of pmos to be higher than the nmos.What determines the size of pmos wrt nmos. Though this is a simple question try to list all the reasons possible?

In PMOS the carriers are holes whose mobility is less[aprrox half] than the electrons, the carriers in NMOS. That means PMOS is slower than an NMOS. In CMOS technology, nmos helps in pulling down the output to ground ann PMOS helps in pulling up the output to Vdd. If the sizes of PMOS and NMOS are the same, then PMOS takes long time to charge up the output node. If we have a larger PMOS than there will be more carriers to charge the node quickly and overcome the slow nature of PMOS . Basically we do all this to get equal rise and fall times for the output node.

145)Why PMOS and NMOS are sized equally in a Transmission Gates?

In Transmission Gate, PMOS and NMOS aid each other rather competing with each other. That's the reason why we need not size them like in CMOS. In CMOS design we have NMOS and PMOS competing which is the reason we try to size them proportional to their mobility.

146)All of us know how an inverter works. What happens when the PMOS and NMOS are interchanged with one another in an inverter?

I have seen similar Qs in some of the discussions. If the source & drain also connected properly...it acts as a buffer. But suppose input is logic 1 O/P will be degraded 1 Similarly degraded 0;

147)A good question on Layouts. Give 5 important Design techniques you would follow when doing a Layout for Digital Circuits?

a)In digital design, decide the height of standard cells you want to layout.It depends upon how big your transistors will be.Have reasonable width for VDD and GND metal paths.Maintaining uniform Height for all the cell is very important since this will help you use place route tool easily and also incase you want to do manual connection of all the blocks it saves on lot of area.
b)Use one metal in one direction only, This does not apply for metal 1. Say you are using metal 2 to do horizontal connections, then use metal 3 for vertical connections, metal4 for horizontal, metal 5 vertical etc...
c)Place as many substrate contact as possible in the empty spaces of the layout.
d)Do not use poly over long distances as it has huge resistances unless you have no other choice.
e)Use fingered transistors as and when you feel necessary.
f)Try maintaining symmetry in your design. Try to get the design in BIT Sliced manner.

148)What is metastability? When/why it will occur?Different ways to avoid this?

Metastable state: A un-known state in between the two logical known states.This will happen if the O/P cap is not allowed to charge/discharge fully to the required logical levels.
One of the cases is: If there is a setup time violation, metastability will occur,To avoid this, a series of FFs is used (normally 2 or 3) which will remove the intermediate states.

149)Let A and 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 and B which one would you place near to the output?

The late coming signals are to be placed closer to the output node ie A should go to the nmos that is closer to the output.

150)Explain zener breakdown and avalanche breakdown?

A thermally generated carrier (part of reverse saturation current) falls down the junction barrier and acquires energy from the applied potential. This carriers collides with a crystal ion and imparts sufficient energy to disrupt a covalent bond.In addition to the original carrier, a new electron-hole pair has been generated. These carriers may also pick up sufficient energy and creates still another electron-hole pair. This cumulative process is called the Avalanche breakdown.
A reverse electric field at the junction causes a strong force to be applied on a bounded electron by the field to tear it out of its covalent bond. The new hole-electron pair which is created increases the reverse current, called zener breakdown.

151)What is Instrumentation Amplifier(IA) and what are all the advantages?

An instrumentation amplifier is a differential op-amp circuit providing high input impedances with ease of gain adjustment by varying a single resistor

152) What is the fundamental difference between a MOSFET and BJT ?

In MOSFET,current flow is either due to electrons(n-channel MOS) or due to holes(p-channel MOS)
- In BJT, we see current due to both the carriers.. electrons and holes. BJT is a current controlled device and MOSFET is a voltage controlled device.

153) What is the basic difference between Analog and Digital Design?

Digital design is distinct from analog design. In analog circuits we deal with physical signals which are continuous in amplitude and time. Ex: biological data, sesimic signals, sensor output, audio, video etc.

Analog design is quite challenging than digital design as analog circuits are sensitive to noise, operating voltages, loading conditions and other conditions which has severe effects on performance. Even process technology poses certain topological limitations on the circuit. Analog designer has to deal with real time continuous signals and even manipulate them effectively even in harsh environment and in brutal operating conditions.
Digital design on the other hand is easier to process and has great immunity to noise. No room for automation in analog design as every application requires a different design. Where as digital design can be automated. Analog circuits

generally deal with instantaneous value of voltage and current(real time). Can take any value within the domain of specifications for the device.consists of passive elements which contribute to the noise(thermal) of the circuit . They are usually more sensitive to external noise more so because for a particular function a analog design
uses lot less transistors providing design challenges over process corners and temperature ranges. deals with a lot of device level physics and the state of the transistor plays a very important role Digital Circuits on the other hand deal with only two logic levels 0 and 1(Is it true that according to quantum mechanics there is a third logic level?) deal with lot more transistors for a particular logic, easier to design complex designs, flexible logic synthesis and greater speed although at the cost of greater power. Less sensitive to noise. design and analysis of such circuits is dependant on the clock. challenge lies in negating the timing and load delays and ensuring there is no set up or hold violation.

154)What is ring oscillator? And derive the freq of operation?

Ring oscillator circuit is a coupled inverter chain with the output being connected to the input as feedback. The number of stages(inverters) is always odd to ensure that there is no single stable state(output value). sometimes one of the stages consists of a logic gate which is used to initialise and control the circuit. The total time period of operation is the product of 2*number of gates and gate(inverter) delay. And frequency of operation will be inverse of time period. Application: used as prototype circuits for modeling and designing new semiconductor processes due to simplicity in design and ease of use. Also forms a part of clock recovery circuit.

155)What are RTL, Gate, Metal and FIB fixes? What is a "sewing kits"?

There are several ways to fix an ASIC-based design. >From easiest to most extreme:

RTL Fix -> Gate Fix -> Metal Fix -> FIB Fix

First, let's review fundementals. A standard-cell ASIC consists of at least 2 dozen manufactured layers/masks. Lower layers conists of materialsmaking up the actual CMOS transistors and gates of the design. The upper 3-6 layers are metal layers used ti connect everything together. ASICs, of course, are not intended to be flexible like an FPGA, however, important "fixes" can be made during the

manufacturing process. The progression of possible fixes in the manufacturing life cycle is as listed above.

An RTL fix means you change the Verilog/VHDL code and you resynthesize. This usually implies a new Plance&Route. RTL fixes would also imply new masks, etc. etc. In other words - start from scratch.

A Gate Fix means that a select number of gates and their interconections may be added or subtracted from the design (e.g. the netlist). This avoids resynthesis. Gate fixes preserve the previous synthesis effort and involve manually editing a gate-level netlist - adding gates, removing gates, etc. Gate level fixes affect ALL layers of the chip and all masks.

A Metal Fix means that only the upper metal interconnect layers are affected. Connections may be broken or made, but new cells may not be added. A Sewing Kit is a means of adding a new gate into the design while only affecting the metal layers. Sewing Kits are typically added into the initial design either at the RTL level or during synthesis by the customer and are part of the netlist. A Metal Fix affects only the top layers of the wafers and does not affect the "base" layers.

Sewing Kits are modules that contain an unused mix of gates, flip-flops or any other cells considered potentially useful for an unforseen metal fix. A Sewing Kit may be specified in RTL by instantiating the literal cells from the vendor library. The cells in the kit are usually connected such that each cell's output is unconnected and the inputs are tied to ground. Clocks and resets may be wired into the larger design's signals, or not.

A FIB Fix (Focussed Ion Beam) Fix is only performed on a completed chip. FIB is a somewhat exotic technology where a particle beam is able to make and break connections on a completed die. FIB fixes are done on individual chips and would only be done as a last resort to repair an otherwise defective prototype chip. Masks are not affected since it is the final chip that is intrusively repaired.

Clearly, these sorts of fixes are tricky and risky. They are available to the ASIC developer, but must be negotiated and coordinated with the foundry. ASIC designers who have been through enough of these fixes appreciate the value of adding test and fault-tolerant design features into the RTL code so that Software Fixes can correct mior silicon problems!