**Birla Institute of Technology and Science, Pilani**
**II semester 2016-2017**

MEL G641- CAD for IC Design (Midsem Test) -- Closed Book

Date: 11.03.2017 MM: 30 Duration: 90 Min.

---

Note: *Attempt all parts of particular question at one place.*

**Q1:** Answer the following questions in brief (write only important points).

(A) List the typical design blocks used for implementation of hardware and software partitions in a system co-design. Also comment on the type of interfaces used. **(2M)**

(B) Draw the flow chart for low VLSI design flow which indicate abstraction, Unified Power Format, indicate the timing and power verification, functional verification and equivalence checking. **(2M)**

(C) Explain how the adjacent cell logic in a row are isolated in the gate array design? **(2M)**

(D) Differential between timing analysis of pre clock tree synthesis and post clock tree synthesis. **(1M)**

**Q2**: For the circuit shown in figure 1, find the test vectors to determine (i) SA0 fault at node 'I' (ii) SA1 fault at node 'I' (iii) Bridging fault between 'H' and 'I' such that both node are at logic '1' ? **(3 M)**

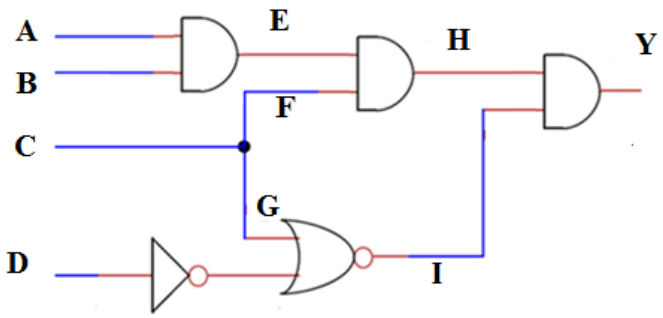


**Figure 1**

**Q3:** The circuit shown in figure 2, shows the network of combinational logic (CL) and D-flip flops. The minimum logic contamination delay ($T_{logic, cd}$) and maximum logic propagation delay ($T_{p, logic}$) are indicated in the figure. Assuming the all positive edge triggered flip flops with $t_{clk-q}$=60 ps, $t_{setup}$=40 ps and $t_{hold}$=50 ps then calculate maximum clock frequency for sequential circuit to operate correctly and check for any hold violations for following cases (i) Clock is ideal i.e. no skew and no jitter. (ii) The difference in clock arrival between any two flip flop is ±55 ps. **(4 M)**

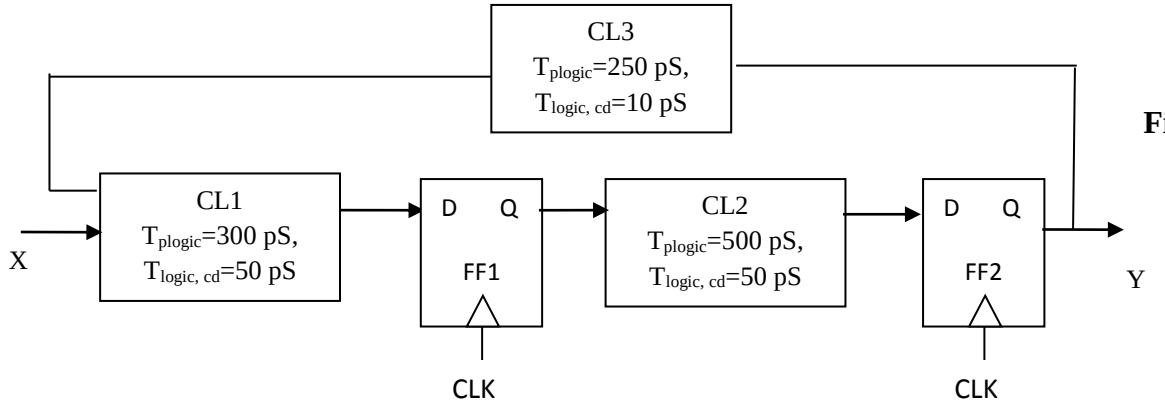


**Figure 2**

**Q4**: (A) Identify if following can be used as a VHDL or Verilog identifiers, justify your answer.

(i) $help (ii) valid op (iii) _std5_qlogic (iv) \last minute\ **(2 M)**

(B) Write a VHDL code for a D latch (positive level sensitive, without set or rest pin) using block statement in dataflow style of modeling. **(3 M)**

**Q5:** Draw the waveform for signal 'A', X and Z when the following VHDL code is executed. **(2 M)**

```
library work;
entity abc is
port (Z: out bit);
end abc;
architecture Arch of abc is
signal A,X: bit;
begin
```

```
A<='0', '1' after 5 ns,'0' after 8 ns, '1' after 10 ns,'0' after 25 ns, '1' after 28 ns,'0' after 30 ns,'1' after 45 ns,'0' after 48 ns;
X<=A after 10 ns;
Z<= A and X after 5 ns;
end Arch;
```

**Q6:** A entity PQR has three architectures namely ABC1, ABC2 and ABC3. If entity PQR is used as a sub-module to design a top module having entity name LMN, which has two architecture namely XYZ1 and XYZ2. Now write a configuration which binds architecture XYZ2 to entity LMN, it also binds architecture ABC2 to the all the instances of entity PQR within the entity LNM and its associated architecture. (Assume default binding rules wherever applicable). **(2 M)**

**Q7:** Show all values of X1, X2, X3, X4, X5 up to 40 time units, when the following verilog code is executed. **(3 M)**

```
module LMN_2016 (X1, X2, X3, X4, X5);
output X1, X2, X3, X4, X5;
reg X1, X2, X3, X4, X5;
integer y=2;
initial begin
X1 = 0; X2 = 0; X3 = 0; X4 = 0; X5 = 0;
end
```

```
always
begin
#(y) X1=1'b1; #(y+2) X1=1'b0; #(y+5) X1=1'b1;
end
always #10 X2 = ~X2;
always @(X2) #6 forever #2 X3 = ~X3;
always wait(X1|X2) #4 X4 = ~X4;
always @(X1,X2) #4 X5 = ~X5;
endmodule
```

**Q8:** Write a behavioral Verilog HDL code for a 2-bit pipelined ripple adder as shown figure 3, which must take input every clock cycle and generates the output after 2 clock cycles at positive edge only. **(4 M)**

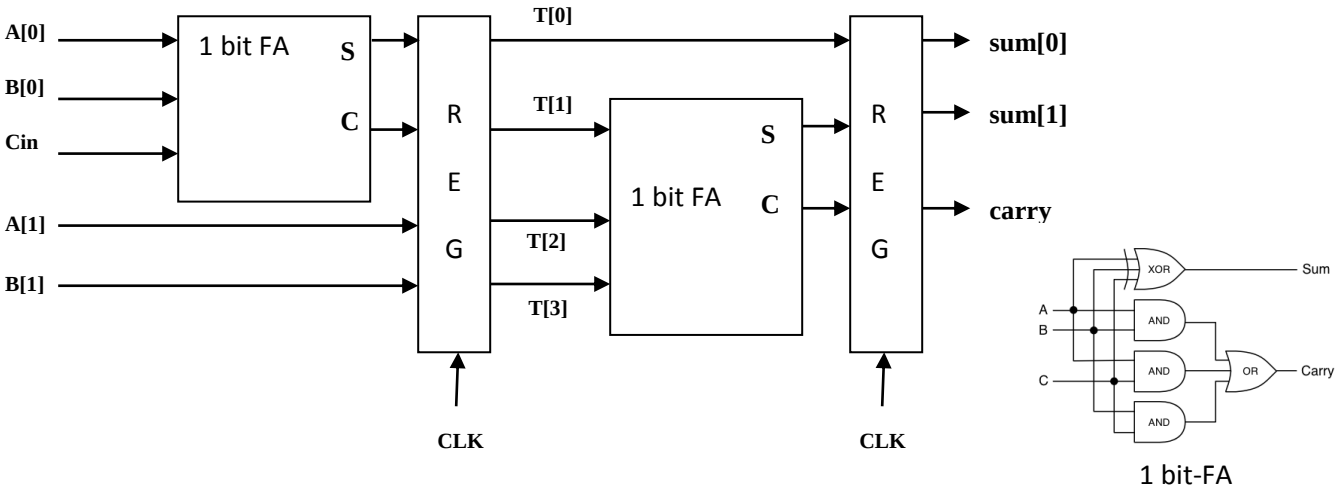


**Figure 3**

---*********---