ASIC-FPGA Chip Design

Electrical Engineering Department

Sharif University of Technology

Assignment #4, 2018

1- Consider the following code describing a combinational logic:

```
module DUT (A, B, C, S, out);
input A, B, C;
input [2:0] S;
output reg out;
always @(*)
begin
   if (S[0])
       out = A;
else if (S[1])
          out = B;
else if (S[2])
          out = C;
end
endmodule
```

a. Draw the synthesized circuit out of the above code. What is the possible problem with this design?
b. Proposed two solutions to resolve this problem along with their synthesized circuits.
c. Implement the modified design in part (b) using tri-state buffers without using multiplexers. Show your architecture and RTL code.

2- **I:** In the following partial code, *count* becomes 1 at time 10. Write values of *a* and *b* between time 0 and 40. Assume a=0 and b=0 as their initial value.

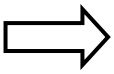
```
always begin
wait (count)
#10 a = a + 1;
end

always begin
@ (count)
#10 b = b + 1;
end
```

Time	Count	a	b
00	0		
10	1		
20	1		
30	1		
40	1		

II: Rewrite the following code fragment using blocking statements but without using fork:

```
fork begin
#5 a = b;
#12 c = d;
end
```



3- Consider the following variables in a system:

I. A : a signed number with (16,7)
II. B : an unsigned number with (12,5)
III. C : a signed number with (6,1),

where (n,m) represents a 2's complement n-bit number with m bits for the fractional part. Write a Verilog code that implements the following function

$$D = A + B.C,$$

where D is a signed number with the highest possible resolution.

4- Consider the following recursive filter:

$$Y(n) = X(n) + aY(n-1)$$

a) Write a Verilog code to implement this filter. (A complete module that is synthesizable)
b) Show its corresponding architecture.
c) What is the critical path of the above architecture?
d) Come up with an architecture that implements the above filter with twice as much throughput.
e) Can we decrease the critical path with the simple pipelining technique? Why?
f) The pipelining technique can be applied to the above architecture with a minor modification to the above architecture. Show how to do it on the architecture to reduce the critical path and show the final pipelined design. (Hint: write the above equation for a few consecutive iterations).

5- Design an architecture that sorts four 10-bit inputs. For each of the following flavors find

I. The architecture
II. Critical path of the design
III. Number of clock cycles it takes to perform sorting
IV. Throughput of the design
V. Number of comparisons

a. A zero-latency architecture
b. A high-throughput architecture that outputs a sorted list every clock cycle
c. Write the RTL code of the design in part b