Solution Description Assignment 1:

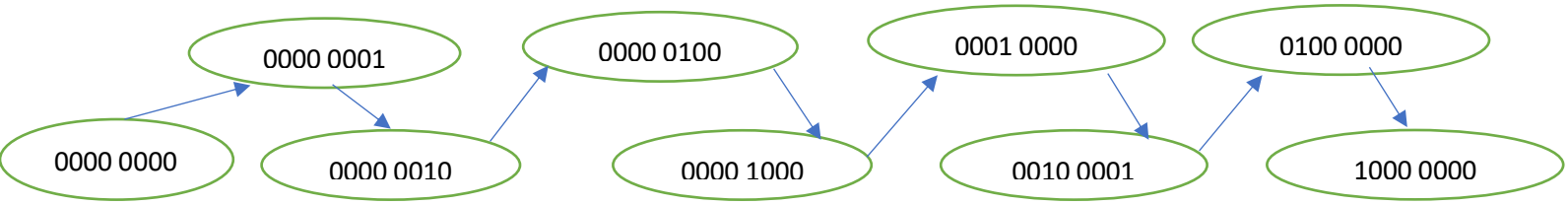
Q1. Design a 3:8 decoder using Verilog and test the design with all possible inputs. Count the number of 1-input, 2-inputs, …, n-inputs gate used for the design, which indicates a good approximation of the area requirement. Count the number of transitions (0→1 or 1→0) for each component's output of the design, for a given input, which indicates a good approximation of the power consumption.

Solution:

1.1 Number of Gates:

Gate	Fan-in	Number of gates
AND	3*1	8
NOT	1*1	3

1.2 Number of transitions: There are 2-bit transition from one state to another:



Total number of transitions=2*8=16 transitions.

Q2. Design a 32-bit 3x8 Multiplexer circuit using Verilog and test the design with all possible select inputs. Count the number of 1-input, 2-inputs, …, n-inputs gate used for the design, which indicates a good approximation of the area requirement. Count the number of transitions (0➐1 or 1➐0) for each component's output of the design, for a given input, which indicates a good approximation of the power consumption.

Solution:

2.1 Number of Gates:

Gate	Fan-in	Number of gates
AND	35*1	8
NOT	1*1	3
OR	256*1	1

2.2 Transition of Inputs (Based on the test bed input)

0000 0000 0000 0000 0000 0000 0000 0000 **->** 0000 0000 0000 0000 0000 0000 0000 0001 **(1 Bit)->**

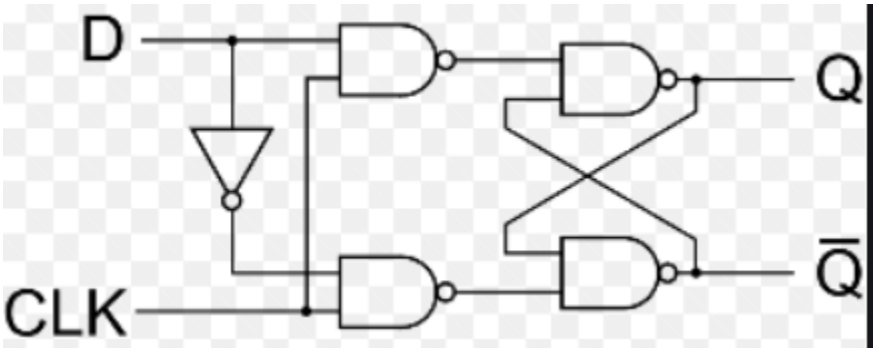
0000 0000 0000 0000 0000 0000 0000 0010 **(2 Bit)->** 0000 0000 0000 0000 0000 0000 0000 0011 **(1)->**

0000 0000 0000 0000 0000 0000 0000 0100**(3) ->** 0000 0000 0000 0000 0000 0000 0000 0101**(1) ->**

0000 0000 0000 0000 0000 0000 0000 0110**(2) ->** 0000 0000 0000 0000 0000 0000 0000 0111 **(1)**

Total number of transitions= 11 Bits

Q3. Design a 32-bit register using the D-flipflop (f/f).
Solution:
3.1
In this question we are calling 32 times D- flip flop module. One D Flipflop contains 5 not and 4 and gates. Each D – Flipflop module call is storing the 1-bit information into Q of size 32 bit.



3.1 Number of Gates:

Gate	Fan-in	Number of gates
NAND	2*1	4*32
NOT	1*1	1*32

3.2 Transition:

In this scenario transition will depends on the input, as D Flipflop will return the same output as input.

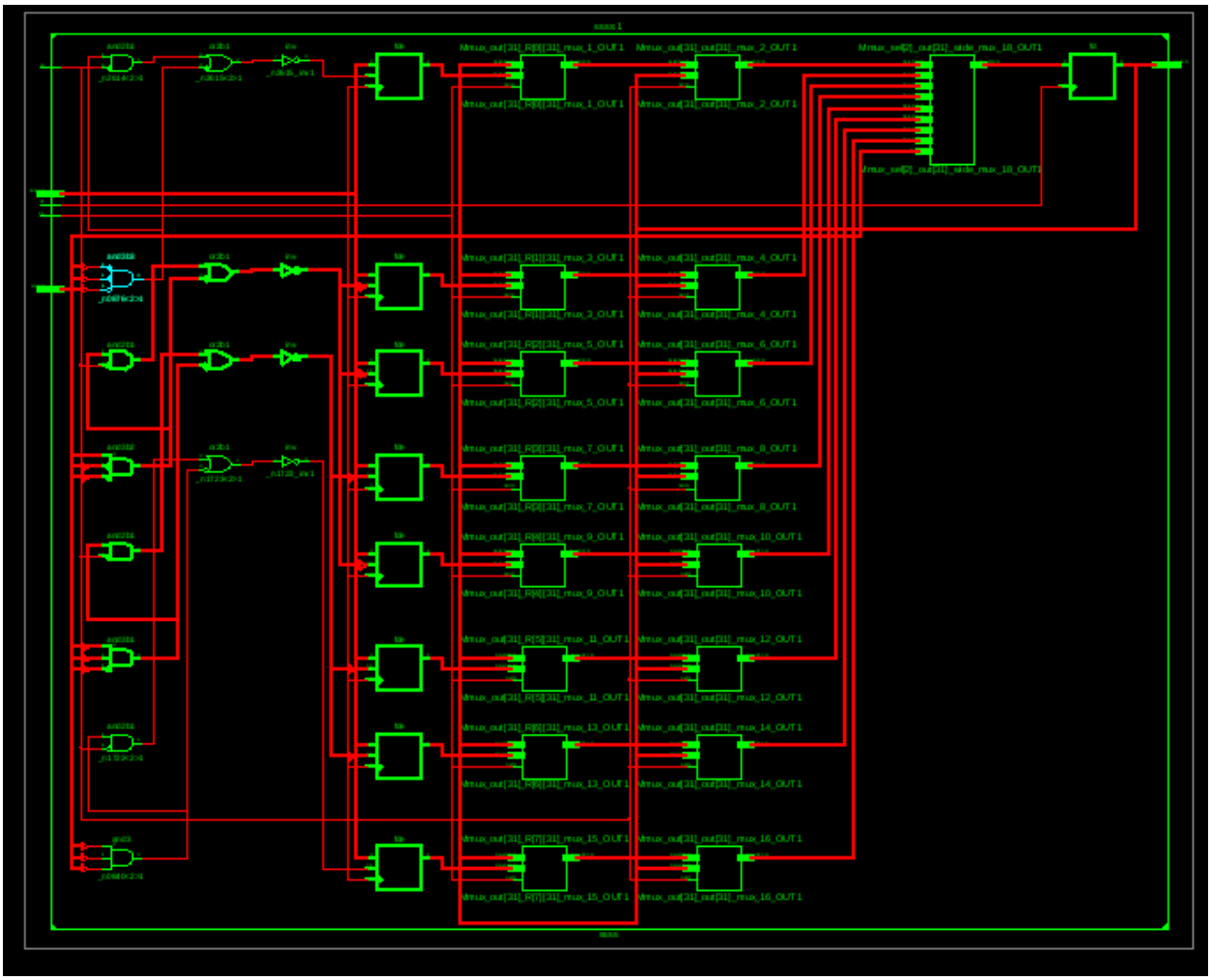
i/p→output
1010_1100_1010_0110_1010_1100_1010_0110→1010_1100_1010_0110_1010_1100_1010_0110 (No change in bits from input to output)

Previous o/p-> next o/p transition:
1010 1100 1010 0110 1010 1100 1010 0110-→ 1010 1110 1010 0111 1010 1110 1010 0111**(4 bit)→**
1010 1110 1010 0111 1010 1110 1010 0111->1110 1110 1010 0110 1110 1110 1010 0110**(4 bit)**
Total 8 bit changes for given testbench.

D Flip flop always work on triggered clock, so we have adjusted the input in the test bench to accommodate the impact of triggered clock.

Q 4. Design a register file which contains the 8 number of registers, with each of width of 32 bits. The circuit must have load enable (LE), output enable (OE) inputs and data input. It also contains an output terminal. A register has to be identified, uniquely, for its operations: write/load or read.

Solution:
RTL Schematic:



From a schematic diagram number of gates used are:

Gate	Fan-in	Number of gates
AND	2*1->4, 3*1->4	8
NOT	1*1	4
OR	2*1	4

MUX	15*1	8
MUX	16*1	8
MUX	18*1	1

2.2 Transition:
From input to output
1010 1100 1010 0110 1010 1100 1010 0110->1010 1100 1010 0110 1010 1100 1111 0110(2 Bit Bit)

1010 1100 1010 0110 1010 1100 1111 0110->1010 1100 1010 0110 1010 1101 1011 0110(2 Bit)

1010 1100 1010 0110 1010 1101 1011 0110->1010 1100 1010 0110 1011 1101 1011 0110(1 bit)
1010 1100 1010 0110 1010 1100 1010 0110->1011 1100 1010 0110 1011 1101 1011 0110(4 bit)
1010 1100 1010 0110 1010 1101 1011 0110->1010 1100 1010 0110 1010 1100 1111 0110(2 Bit)

10 Bit changes

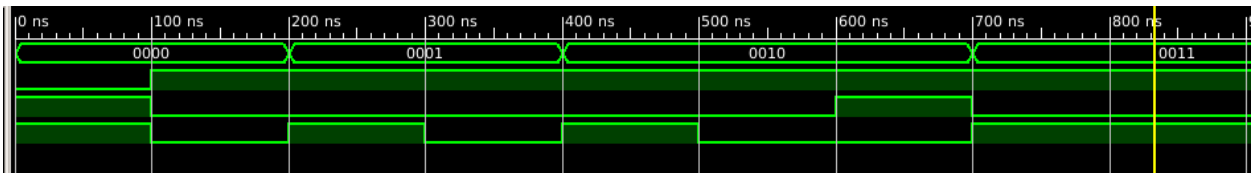
Q 5. Design a 4-bit counter using Verilog and the test the design properly:

Number of Gates as per code are given below

Gate	Fan-in	Number of gates
AND	2*1	4
NOT	1*1	1

Transition:

0000->0001→0010->0011->0011 (4 bit changes for given test bench)



Q6. Design a 32-bit ALU. The ALU contains two input registers: Accumulator and Data. The ALU can perform these operations: Addition of two numbers, Compliment the content of accumulator and Logical AND of two inputs.
Sol:

ALU has been designed using case condition so finding out exact number of gates are not feasible. Number of Gates as per code are given below:

Gate	Fan-in	Number of gates
AND	2*1	1
NOT	1*1	1

Transition(Based on the testbench)
1. For complement-> 32 bit change
2. For AND-> 2 Bit Change