Digital Design (CS F215) **18-09-2024**

Lab 5: Decoder and Latches

Lecturer: Harikrishnan N B *Student:*

READ THE FOLLOWING CAREFULLY:

Honour Code for Students: I shall be honest in my efforts and will make my parents proud. Write the oath and sign it on the assignment.
For the lab, you need a dedicated fresh notebook. Keep this notebook dedicated for solving the questions pertaining to lab.

There will be design questions in the lab, the students are supposed to theoretically show the correctness of their design function. Following that, the students are encouraged to use iverilog software to implement the same.

The lab evaluation consists of two parts.

1) Step 1 (Theory): The students are supposed to solve the design question in their dedicated lab notebook. They are supposed to get it checked and validated by TA's/ instructors. (**Deadline for theory evaluation: 10:15 AM**)

2) Step 2 (Lab): Once Step 1 is completed, they can use iverilog to implement their design. Once you have completed the design, inform the TA's/ instructors and get it verified. (**Deadline for lab file submission in local quanta: 10:40 AM**)

3) Step 3: Upload your solution to quanta.bits-goa.ac.in (local quanta). Create a zip file CampusID.zip which contains all the files.

All questions must be solved using the modeling mentioned in the question. Only solutions implemented using the mentioned modeling will be accepted for evaluation.

Latches

1. **Theory(4 Marks):** Design a hypothetical SR latch with Enable using NAND gates. The latch works similarly to an SR latch, with the input S being replaced by S' and input R being replaced by R'. Write the characteristic table for the same. The inputs in the characteristic table are Enable, S and R. The outputs are Q and Qbar. Draw the digital logic diagram in the lab note. For the forbidden state, the output should be Q = 1 and Qbar = 1.

2. **Lab(4 Marks):** Complete the module `SR_latch.v` in Verilog using a behavioural model to implement the hypothetical latch. The module takes two inputs `S` and `R`, an `enable` signal, and has outputs `Q` and `Qbar`. Complete the corresponding testbench (`testbench_SR_latch.v`). Verify the correctness of the module for the test cases provided in Table 5.1(outputs are Q and Q bar).

Table 5.1: Testcases for the hypothetical SR latch.

Time	enable	S	R
0	1	1	0
5	1	1	1
10	0	0	1
15	1	0	1
20	1	1	1
25	0	1	0
30	1	0	0

Decoders

1. **Theory(2 Marks):** Design a 4-to-16-line decoder with Enable. Let the input be denoted as $I = I_3 I_2 I_1 I_0$ and the output be denoted as $O = O_{15} O_{14} \ldots O_1 O_0$. Draw the block diagram for this decoder.

2. **Lab(6 Marks):** Complete the module `decoder_4to16.v` in Verilog using a behavioural model to implement a 4-to-16-line decoder. The default value of the output should be zero. The module takes `A[3:0]` and `enable` as inputs and gives `out[15:0]` as the output. Complete the corresponding testbench (`testbench_decoder.v`). Verify the correctness of the module for the testcases provided in Table 5.2. Testcases should be in the order given below.

Table 5.2: Testcases for the 4-to-16-line decoder.

Time	enable	A[3:0]
0	0	0000
5	1	0000
10	1	0010
15	1	0100
20	1	0111
25	0	1001
30	1	1001
35	1	1110
40	0	1010

3. **Theory(6 marks):** Suppose you are given a 5-bit binary number. You accept the number if it is a palindrome, else reject it. Design a circuit using 4-to-16 line decoders that takes in a 5-bit number `in[4:0]` as input and gives the output as 1 if the number is a palindrome or else gives the output as 0. The output is denoted by `out`.

4. **Lab(6 marks):** Complete the module `palindrome.v` in Verilog using a structural model to implement the above circuit. The module takes `in[4:0]` as the input and gives `out` as the output. Complete the corresponding testbench (`testbench_palindrome.v`). Verify the correctness of the module for the testcases provided in below Table 5.3. Testcases should be in the order given below.

Table 5.3: Testcases for the palindrome.

Time	**in[4:0]**
0	00000
5	01100
10	00100
15	01010
20	01110
25	11001
30	10101
35	11011
40	11111
45	10010
50	10110
55	01101

Instructions for Lab 5

NOTE: Complete all the modules using the modeling mentioned in the question.
NOTE: Make sure to **not modify the monitor line in testbench**, and keep the **module name** and **input/output variable names unchanged**.
NOTE: Make sure you **don't** change the order of the test cases.

1. Complete the `SR_latch.v` module and `testbench_SR.v`.

   How to test:

   ```
   python3 autograder.py
   ```

2. Complete the `decoder_4to16.v` module and `testbench_decoder.v`.

   How to test:

   ```
   python3 autograder.py
   ```

3. Complete the `palindrome.v` module using the `decoder_4to16.v` module.
   Also complete `testbench_palindrome.v`

   How to test:

   ```
   python3 autograder.py
   ```