Assignment 7

October 30, 2023

Problem 1 : Behavioral 4-to-1 Multiplexer

- Design a 4-to-1 Multiplexer using Behavioral statements.
- Write testbench and simulate your design.

```
module mux4_to_1 (out, i0, i1, i2, i3, s1, s0);
// write your design here
endmodule
```

NOTE : You can re-use the testbench module that you created earlier for the Structural 4-to-1 Multiplexer in Assignment 3, or the Data Flow 4-to-1 Multiplexer in Assignment 6. The simulation result should be the same.

Problem 2 : Behavioral 4-bit Counter

- Design a 4-bit Counter using Behavioral statements.
- The counter contains an asynchronous, active-high reset signal.
- Write testbench and simulate your design.

```
module counter(out, clock, reset);
// write your design here
endmodule
```

Problem 3 : Behavioral Flip-Flops

- Design the following module :
  1. A D flip-flop with an *asynchronous*, active-low reset signal.
  2. A D flip-flop with an *synchronous*, active-low reset signal.
  3. A SR flip-flop with an *synchronous*, active-high reset signal.
  4. A T flip-flop with an *synchronous*, active-low reset signal.
  5. A JK-flip-flop, with no reset signal.
- Write testbench and simulate your design.