100 Days of RTL

DAY 4

Switch Level Modelling

Achutha Aswin Naick

Introduction

Switch-level modelling in Verilog represents circuits at the transistor level using predefined switch primitives such as NMOS and PMOS. This technique closely mirrors actual transistor-based hardware implementation. In this project, a simple switch circuit was implemented using switch-level modelling in Verilog. A corresponding testbench was developed to simulate the circuit and verify its correctness.

Implementation Details

The switch module, switch, consists of a single input (in) and a single output (out). The design uses supply0 (gnd) and supply1 (vdd) to represent logic low (0) and logic high (1), respectively. The circuit is defined using the built-in Verilog switch primitives nmos and pmos, which simulate the behaviour of NMOS and PMOS transistors:

- nmos(out, gnd, in): Connects out to ground (gnd) when in is high (1).
- pmos(out, vdd, in): Connects out to vdd when in is low (0).

Code Explanation

The implementation consists of two Verilog modules: switch (design module) and switch_tb (testbench module).

Below is a breakdown of their components and functionality.

Switch Module (switch)

The switch module defines a simple transistor-based switch circuit using NMOS and PMOS transistors.

- **Input and Output Declaration:**
    - input in: Defines the input signal.
    - output out: Defines the output signal.
    - The out signal is of type wire since it is directly connected to the transistor gates.

- **Supply Nets:**
    - supply0 gnd; assigns a constant logic 0 to gnd.
    - supply1 vdd; assigns a constant logic 1 to vdd.
- **NMOS and PMOS Primitives:**
    - nmos(out, gnd, in); represents an NMOS transistor:
        - When in = 1, out is connected to gnd (0).
        - When in = 0, the NMOS transistor is off.
    - pmos(out, vdd, in); represents a PMOS transistor:
        - When in = 0, out is connected to vdd (1).
        - When in = 1, the PMOS transistor is off.

This combination results in an inverter-like behaviour.

Testbench Module (switch_tb)

The switch_tb module is used to verify the functionality of the switch module.

- **Input Declaration:**
    - reg in; defines in as a reg type, allowing it to be assigned values in the testbench.
- **Output Declaration:**
    - wire out; defines out as a wire, since it is driven by the DUT (Device Under Test).
- **Instantiation of DUT:**
    - switch DUT(in, out); creates an instance of the switch module, connecting its input and output to the testbench signals.
- **Simulation Execution:**
    - initial in = 1'b0; sets the initial value of in to 0.

  - initial forever #10 in = ~in; toggles in every 10 time units, generating a clock-like behaviour.
  - initial #100 $finish; stops the simulation after 100 time units.

Simulation and Results

The testbench applies alternating values (0 and 1) to the input at 10-time unit intervals. The expected behaviour of the switch model is observed:

- When in = 0, the PMOS transistor connects out to vdd (logic 1), resulting in, out = 1.
- When in = 1, the NMOS transistor connects out to gnd (logic 0), resulting in, out = 0.
- This confirms that the switch model behaves as expected, effectively acting as an inverter.

Conclusion

This project successfully demonstrates switch-level modelling in Verilog using NMOS and PMOS transistors. The implementation correctly simulates the behaviour of a simple switch circuit, showcasing how transistors operate at the fundamental level.

Future Enhancements

Future improvements could include:

- Implementing more complex transistor-level circuits, such as pass transistors or transmission gates.
- Simulating real-world effects such as threshold voltage and leakage currents.
- Extending the design to model digital logic gates using transistor-level modelling.