Verilog notes

Tommy Bui

01-29-2023

Contents

1 Verilog Tutorial

Reference to Chipverify

1.1 Lore

In the early days of integrated circuits, engineers had to physically draw transistors & their netlist on paper. As circuits became more complex and larger in scale, this process eventually tedious. Languages such as VHDL & Verilog were developed to simply the process of describing the functionality of IC and let tools convert the behavior into hardware using combinational & sequential logic.

1.2 How is Verilog useful

Verilog creates a level of abstraction that hides the details of its implementation & technology.

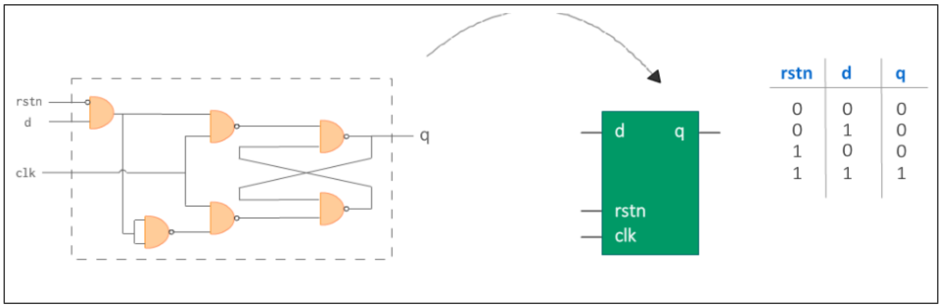
E.g. the design of a D flip-flop requires the knowledge of the transistor layout in order to achieve an edge-triggered FF, rise/setup time, fall/clk-Q times to latch value onto flop, etc.

2 Introduction to Verilog

Source

A digital element such as a FF can be represented using combinational gates such as NAND or NOR gates. The functionality of a FF is determined based on the layout of such gates. How the gates have to be connected is usually determined using K-maps

Below is an schemetic of a Data flip-flop & its corresponding truth table. The output, q is asserted only when rstn & d are both set.



rstn	d	q
0	0	0
0	1	0
1	0	0
1	1	1

Figure 1: D flip flop