VLSI Special Assignment

Two Bit Counter

Hitarth Chudasama
(22BEC019)

Department of Electronics and Communications Engineering
Institute of Technology,
Nirma University,
Ahmedabad

Objective and Motivation

This report attempts to design and implement a 2-bit counter at the transistor level logic with an emphasis on the design of a 4-state counter as dictated by: (00), (01), (10) and (11). It attempts to illustrate how basic logic gates and flip-flops can be utilized at the transistor level, mainly using NMOS and PMOS transistors. Further, the report attempts to analyze the behavior of the counter in a state transition. Accordingly, this report also intents to explore the practical application of such counters in digital systems, for example, timers and control units.

The motivation for this project is to have a greater understanding of the principles of digital design by getting away from the abstract logic into transistor-level implementation. This reduces theory to practical applications in IC design; such is the need that the IC designers require to develop at the system level. It also serves as a step toward gaining expertise in VLSI design, since counters are fundamental in digital electronics. Completing this design accelerates the report to skills in circuit simulation and verification prior to advanced digital systems and other prospects in hardware design.

Introduction

A 2-bit up-down counter is one of the most elementary digital circuits. It counts up and down, and any mode is selectable. As for this assignment, we provide the design and implementation of a 2-bit up-down counter such that Mode 0 is the up-counting mode and Mode 1 the down-counting mode. This counter cycles in a sequence of four different possible states that is represented in binary as 00, 01, 10, and 11, which can either count up or count down depending on the input mode.

The up-down counters are of practical importance because of their wide ranging applications in digital electronics such as frequency divider circuits, digital clocks, and control systems. They are versatile in nature and can be used for applications like reversible counting, position indication in control systems, and bidirectional event counting, as they can switch between up and down counting.

Although synchronous counters have more advantages due to their simpler design and fewer hardware requirements, asynchronous counters or ripple counters have some of them as well. In an asynchronous counter, only the first flip-flop is clock-driven and the rest are driven by the output of the preceding one. This creates a chain-like or "ripple" effect because changes propagate across the flip-flops in sequence rather than simultaneously.

This ripple effect means that asynchronous counters are more frequently simpler to implement, requiring fewer components and less wiring than their synchronous counterparts.

This also makes asynchronous counters relatively cheaper and occupying lesser space on a circuit board. Reduced component count further helps in minimizing power consumption, thus producing asynchronous counters to be energy-efficient, which is particularly useful in low-power or resource-constrained applications.

The Optimized Boolean expression

Let's implement the 2-bit counter by using concepts from the finite state machine. So before we draw up the truth table, we will draw a state diagram based on our requirement. We will use a D flip-flop in the implementation of the truth table.

We consider an asynchronous counter, in which the first flip-flop is driven only the first flip-flop's output will drive some kind of clock, and the others will be driven by that clock.

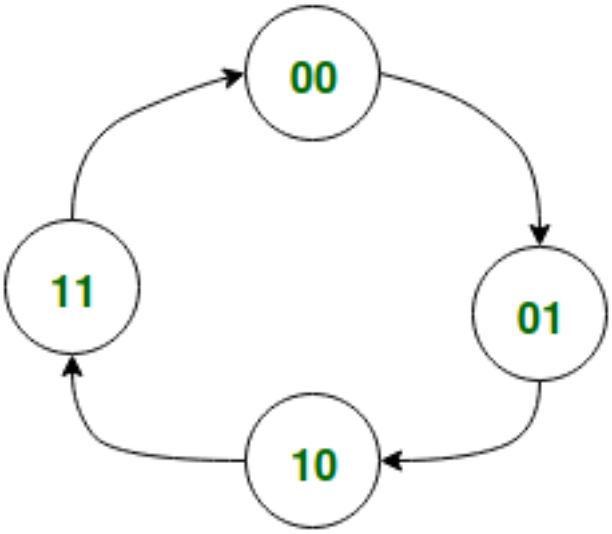


FIG1. THE STATE DIAGRAM OF UP COUNTER

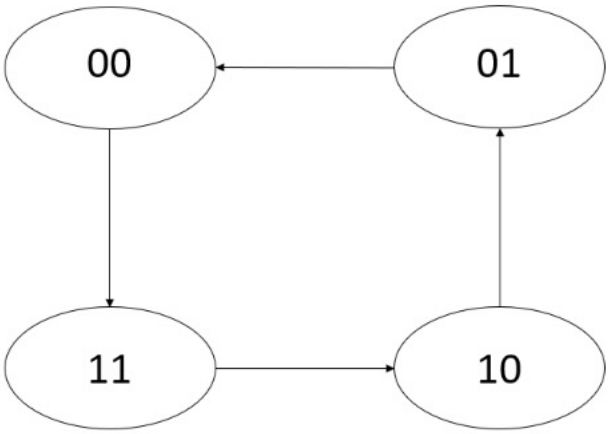


FIG2. THE STATE DIAGRAM OF DOWN COUNTER

Let's examine the circuit diagram and timing diagram for the counter. In an asynchronous counter, the clock input is applied to the Q1 D flip-flop, while the other flip-flops are driven by either the inverted Q1 output for an up-counter or the Q1 output itself for a down-counter. We will use distinct methodologies to implement each type of counter.

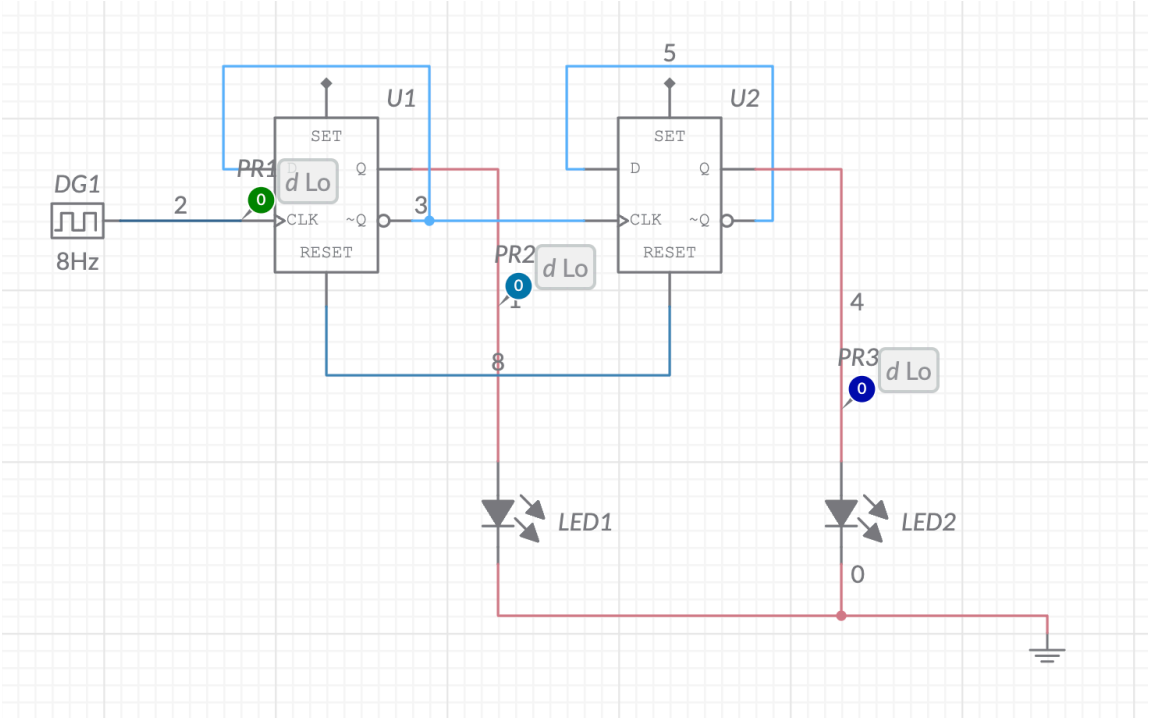


FIG3. CIRCUIT DIAGRAM OF 2-BIT UP COUNTER

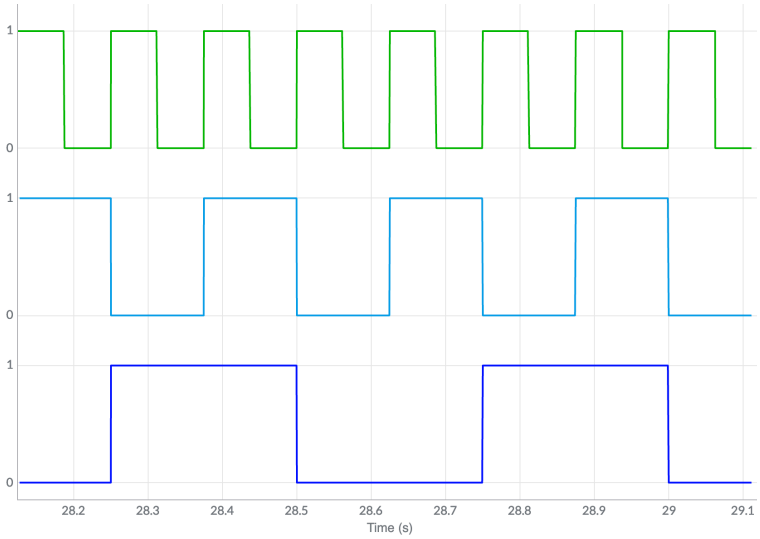


FIG4. OUTPUT OF 2-BIT DOWN COUNTER

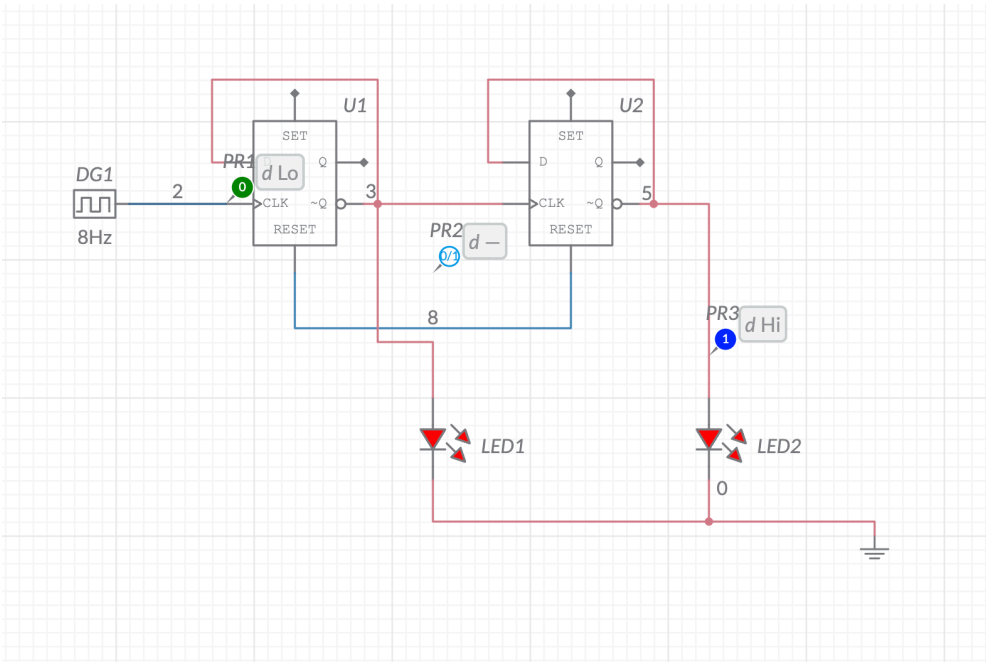


FIG5. CIRCUIT DIAGRAM OF 2-BIT DOWN COUNTER

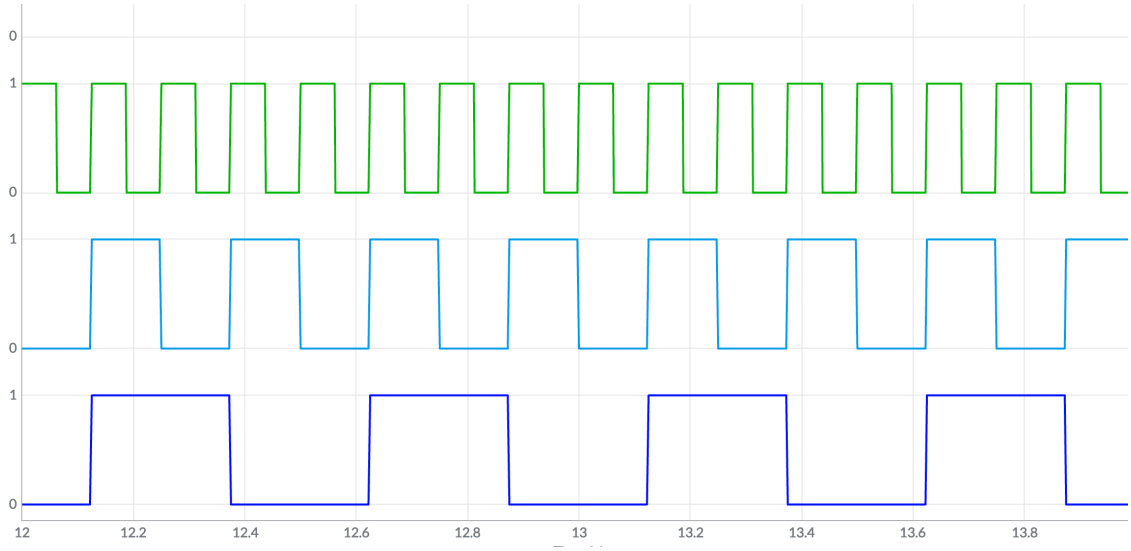


FIG6. OUTPUT OF 2-BIT DOWN COUNTER

Optimized gate-level circuit diagram

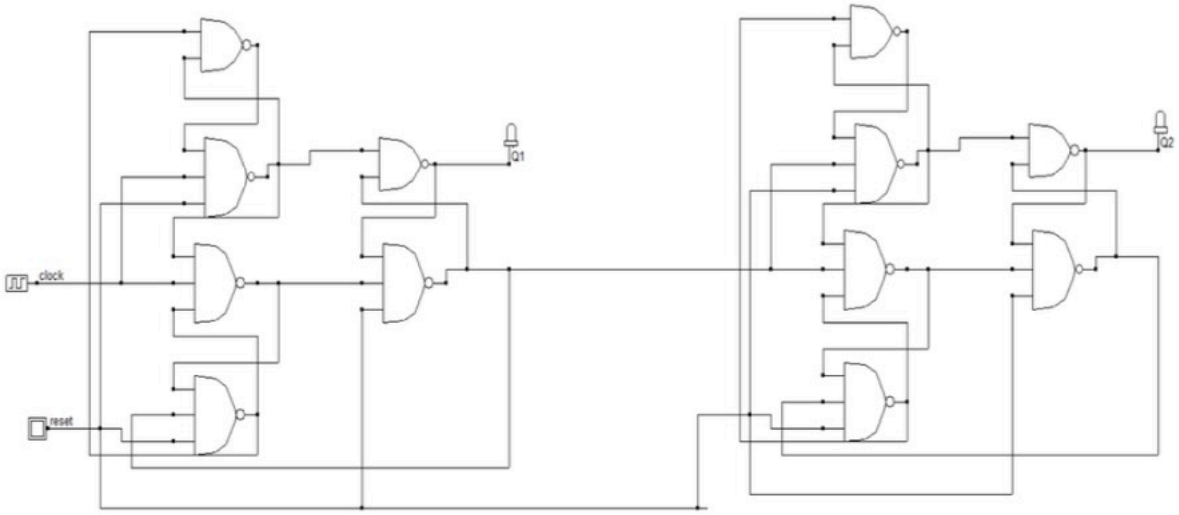


FIG7. CIRCUIT DIAGRAM 2-BIT UP COUNTER

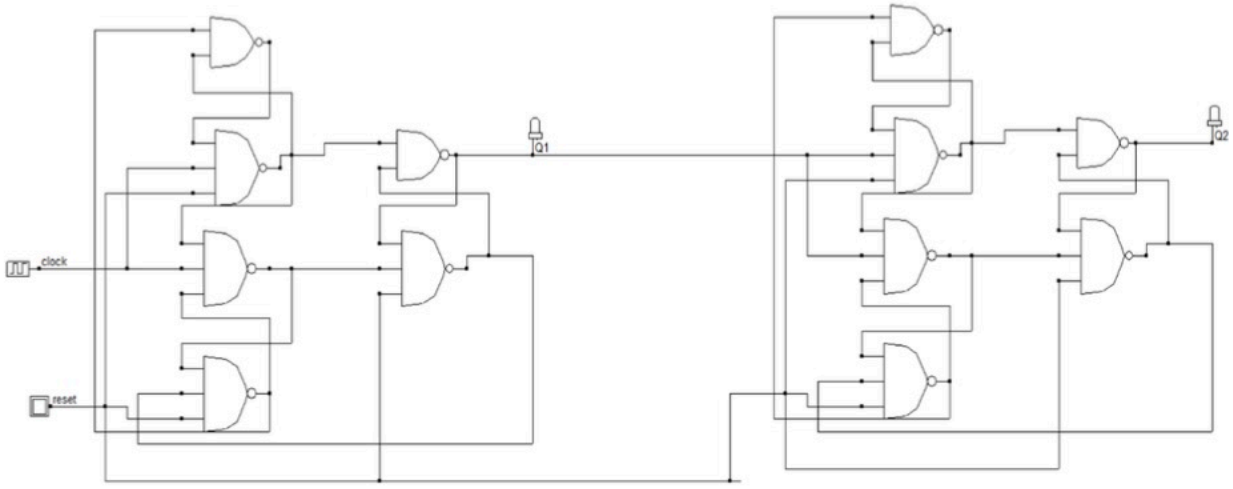


FIG8. CIRCUIT DIAGRAM 2-BIT DOWN COUNTER

Transistor level CMOS circuit for down counter

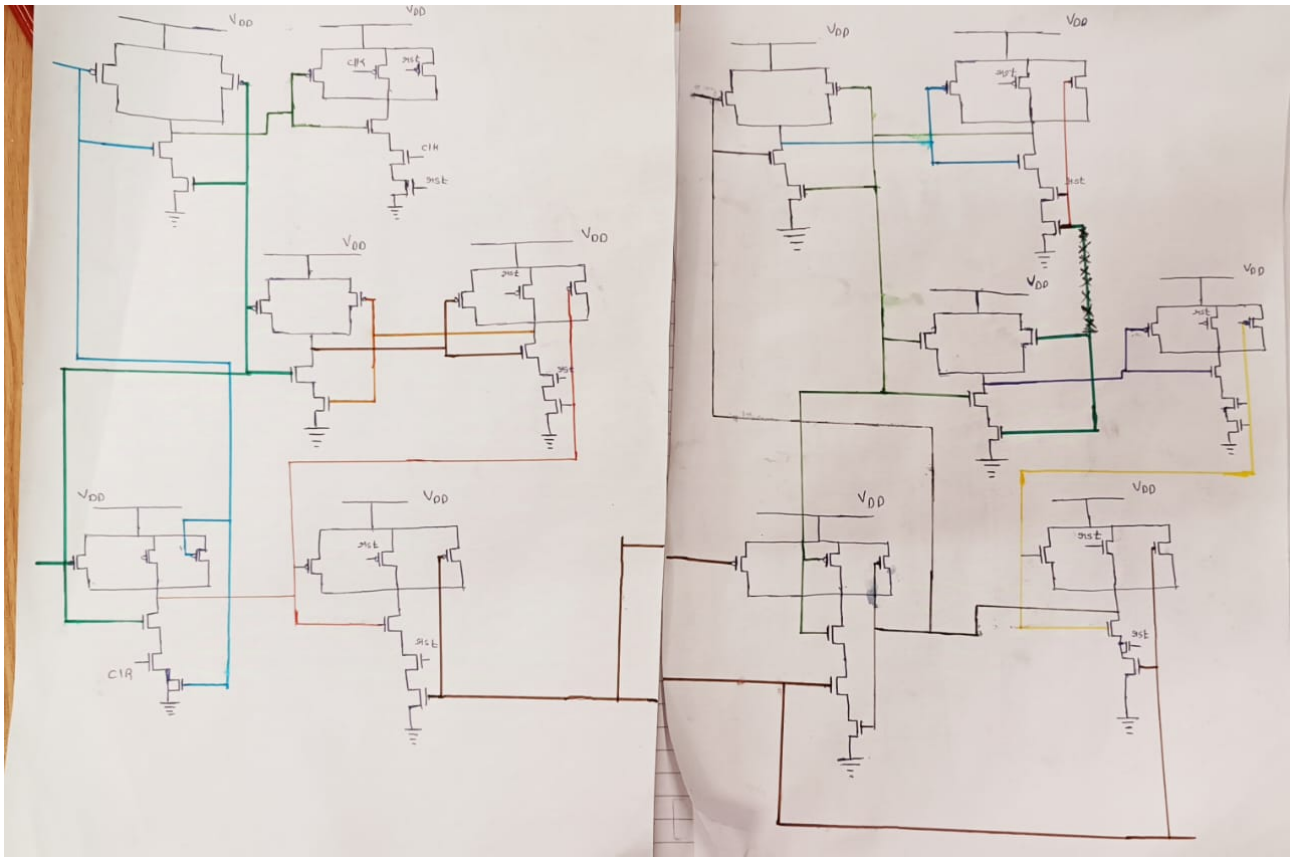


FIG9. TRANSISTOR LEVEL CIRCUIT (DOWN COUNTER)

Stick diagram

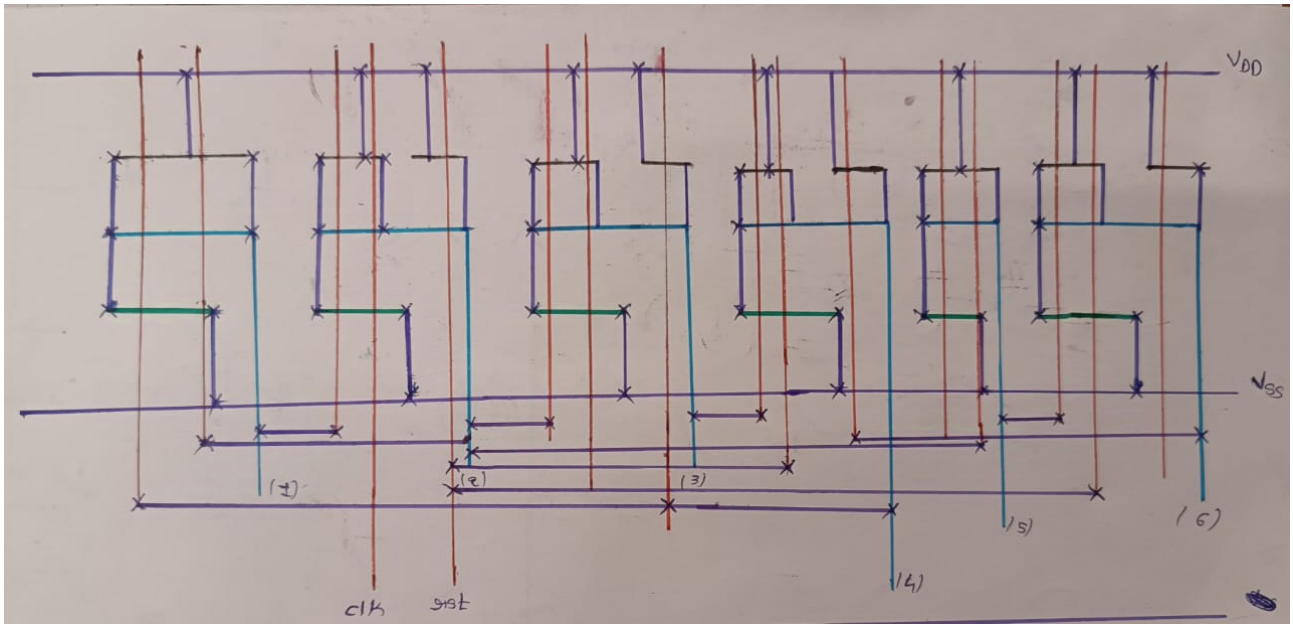
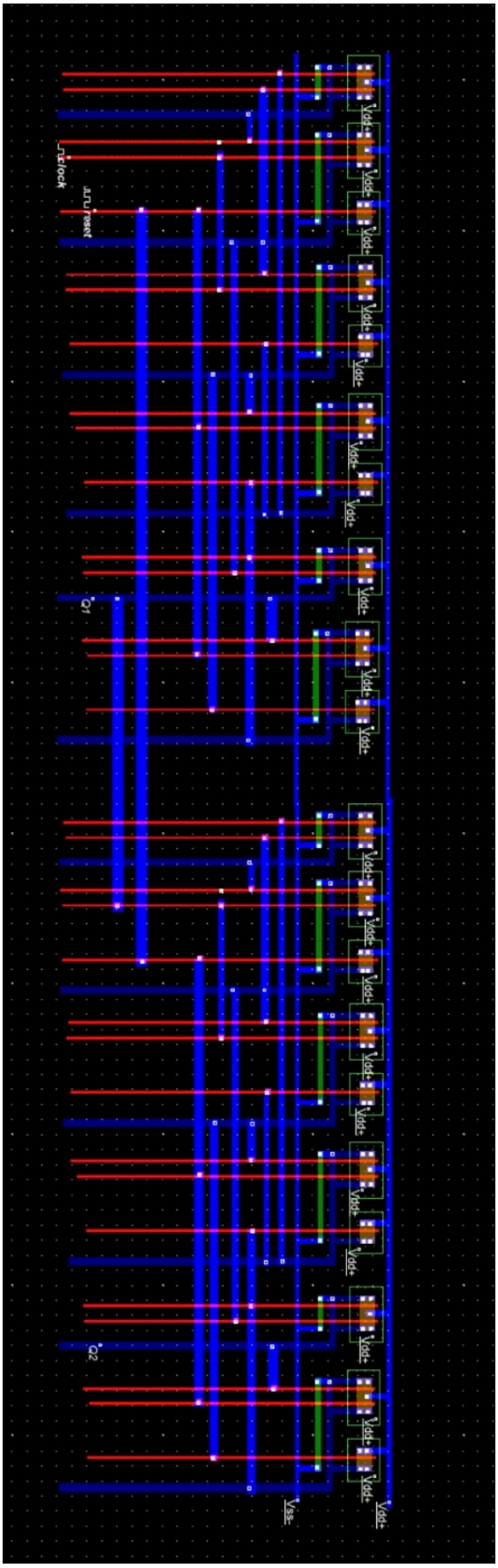


FIG10. STICK DIAGRAM

In this stick diagram, we've implemented only a single D flip-flop rather than the entire circuit, as doing so makes it easier to understand. In an asynchronous counter, the inverted output from the first flip-flop serves as the clock input for the next flip-flop. Therefore, in the complete circuit, this stick diagram shows that the Q1 inverted output is used as the clock for the second flip-flop.

Layout using micro-wind



Simulation of the layout

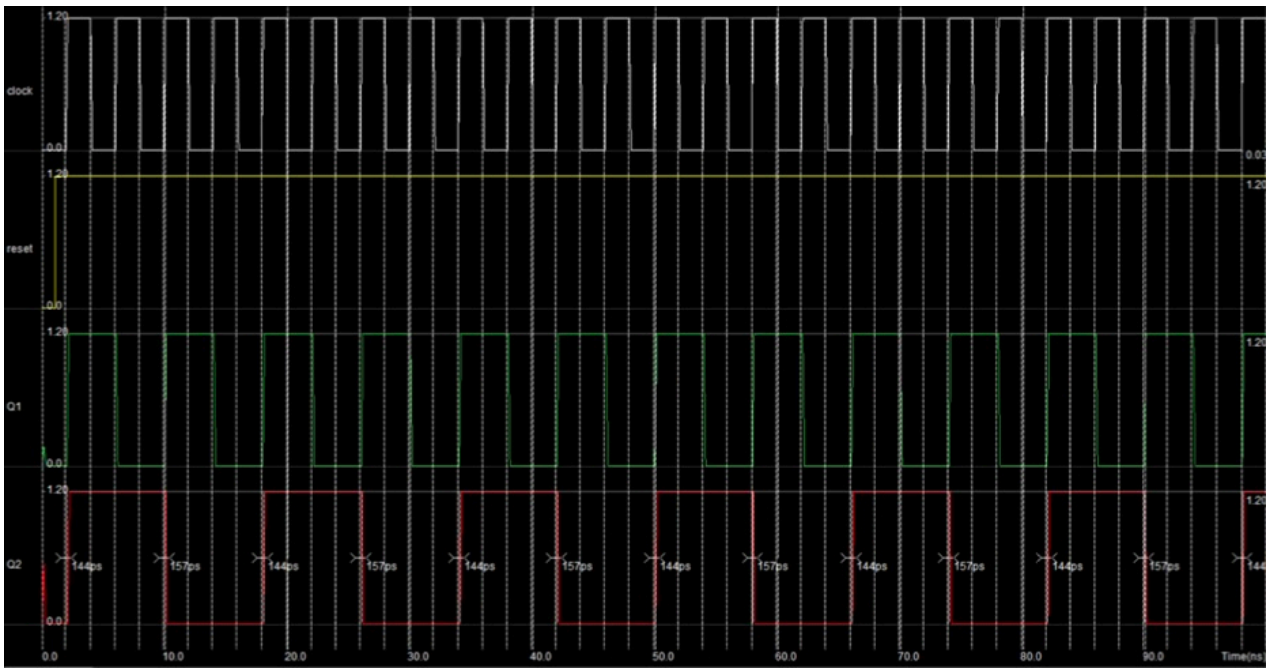


FIG12. OUTPUT OF UP COUNTER

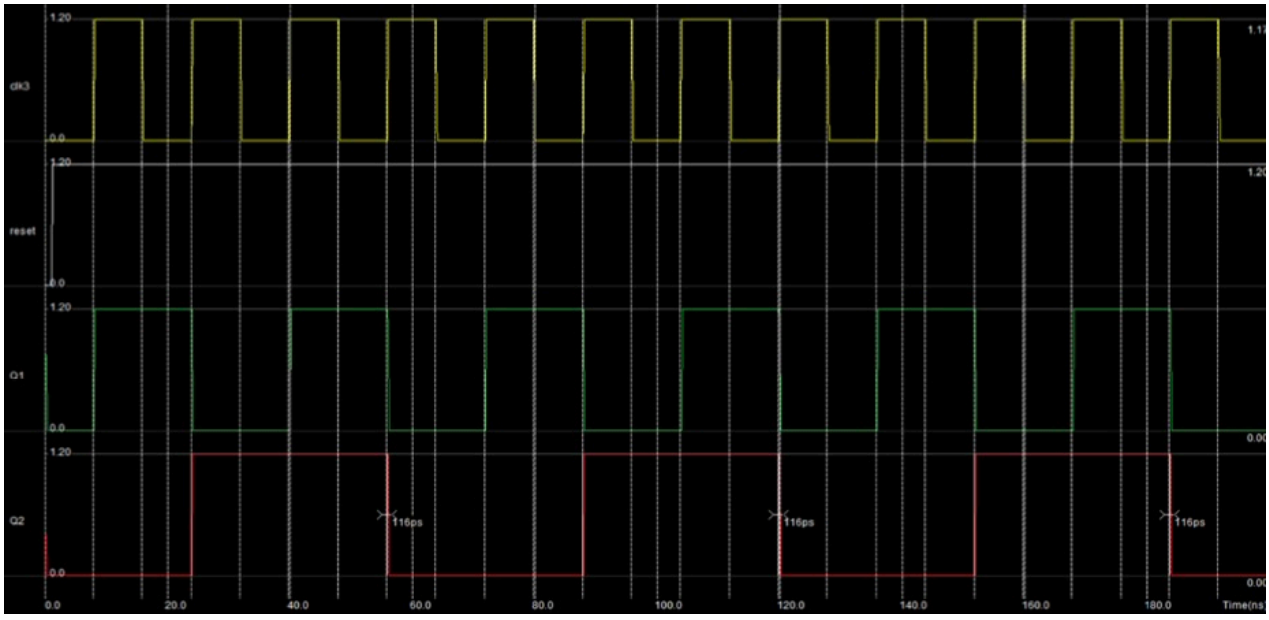


FIG13. OUTPUT OF DOWN COUNTER

The rise time, fall time, propagation delay and other parameter

Technology = 0.12um
Voh = 1.2v
For down counter

Output	Tplh	Tphl
Q1	-	-
Q2	143ps	159ps

PROPAGATION DELAY

Input	Risetime	Falltime
CLK	0.025ps	0.025ps
RST	0.1ps	0.1ps

RISE AND FALL TIME

Technology = 0.18um
Voh = 2V

Output	Tplh	Tphl
Q1	-	-
Q2	354ps	395ps

PROPAGATION DELAY

Technology = 0.35um
Voh = 3.5V

Output	Tplh	Tphl
Q1	-	-
Q2	970ps	1114ps

PROPAGATION DELAY

Technology = 0.12um
Voh = 1.2V
For up counter

Output	Tplh	Tphl
Q1	-	-
Q2	962ps	1227ps

PROPAGATION DELAY

Input	Risetime	Falltime
CLK	0.025ps	0.025ps
RST	0.1ps	0.1ps

RISE AND FALL TIME

Technology = 0.18um
Voh = 2V

Output	Tplh	Tphl
Q1	-	-
Q2	271ps	298ps

PROPAGATION DELAY

Technology = 0.35um
Voh = 3.5V

Output	Tplh	Tphl
Q1	-	-
Q2	710ps	790ps

PROPAGATION DELAY

Conclusion

In summary, I designed and implemented successfully a 2-bit asynchronous counter using D flip-flops and got the expected output. By analysis, several problems with alternatives of configurations of flip-flops had occurred. The first was the SR flip-flop because its net propagation delays are significant when the S and R inputs are connected in a complementary manner. Another is the TG latch and also the master-slave configurations fail to give a good square wave clock, an essential requirement for a counter to work correctly. In addition, these configurations are level-triggered, while our design strictly needed edge-triggered behaviour for an accurate count on clock transitions. Using an edge-triggered NAND3-based flip-flop, these problems were overcome and reliable and precise operation was guaranteed in our presented 2-bit Asynchronouscounter. This approach demonstrated that edge-triggered D flip-flops are utilizable for proper and accurate implementation of asynchronous counters.