Assignment 1

Submitted By: Sanpreet Singh Gill & Yen-Chun Chen
gillsan@oregonstate.edu
chenyenc@oregonstate.edu

D-flip flop

Introduction

In this testbench we verify the functionality of both synchronous set/reset (SSR) and asynchronous set/reset (ASR) D flip-flops. The testbench module begins by declaring all the logical inputs and outputs to the D flip-flop module and then instantiates an instance of synchronous D flip-flop module and an instance of asynchronous D flip flop module. The named port connection to both the instances are similar except the output port which helps us separate the output waveform of the two modules.

Input Signals

- D: Data input
- CE: Chip enable
- SR: Asynchronous reset
- CLK: Clock
- SRINIT: Initial value for asynchronous reset

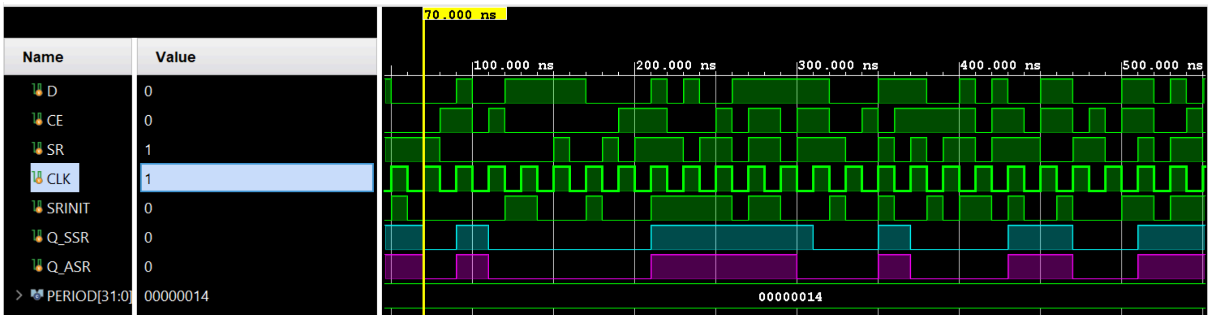
Output Signals

- Q_SSR: Output of SSR D flip-flop
- Q_ASR: Output of ASR D flip-flop

Procedure

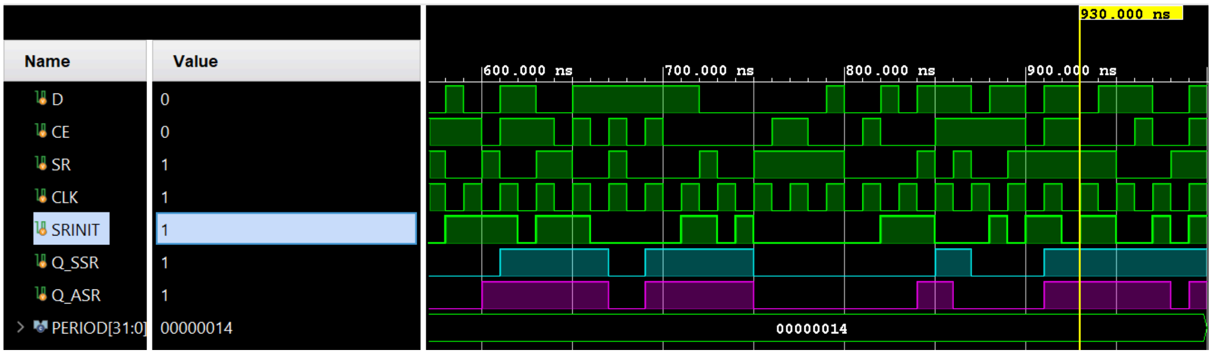
We first generate a clock of **period 20 ns**. To do blackbox testing on this module and to provide every possible combination of inputs we declared a 'random object' class to hold all the input parameters which we later randomised and feed it to the modules under test. Finally, we monitored and displayed the output values of both the flip flops.

Simulation and Output Waveform



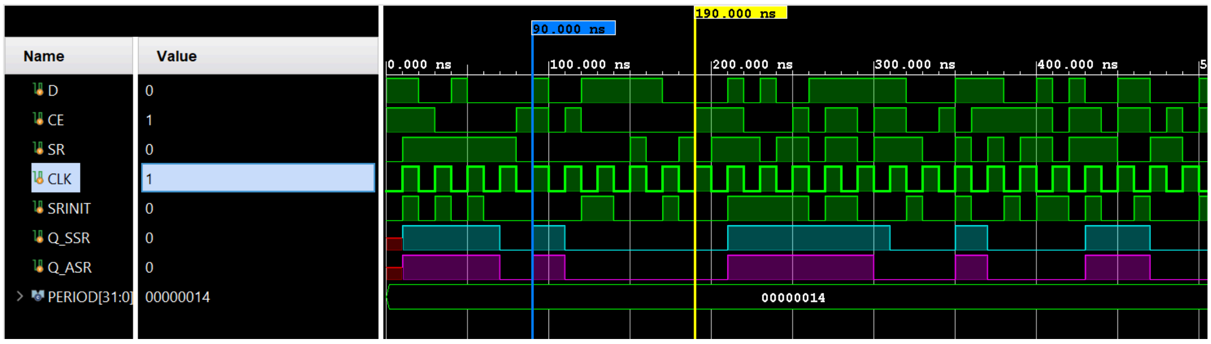
Name	Value
D	0
CE	0
SR	1
CLK	1
SRINIT	0
Q_SSR	0
Q_ASR	0
PERIOD[31:0]	00000014

Figure 1



Name	Value
D	0
CE	0
SR	1
CLK	1
SRINIT	1
Q_SSR	1
Q_ASR	1
PERIOD[31:0]	00000014

Figure 2



Name	Value
D	0
CE	1
SR	0
CLK	1
SRINIT	0
Q_SSR	0
Q_ASR	0
PERIOD[31:0]	00000014

Figure 3

At 70 ns (Figure 1) the SR is asserted with SRINIT's value of 0 and the output of both the flip flops is 0 as well. Conversely, at 930 ns (Figure 2) the SRINIT value is 1 and the output also goes high. At 90 ns and 190 ns (Figure 3), we can see the normal operation of flip-flops.

D-Latch

Introduction

The testbench simulates the behaviour of D latches under various input conditions, including random input values for data (D), chip enable (CE). The testbench module begins by declaring all the logical inputs and outputs to the D flip-flop module and then instantiates an instance of synchronous D latch module and an instance of asynchronous D latch module. However, there is no difference between the two modules. The named port connection to both the instances are similar except the output port which helps us separate the output waveform of the two modules.

Input Signals

- D: Data input
- CE: Chip enable
- SR: Asynchronous reset
- SRINIT: Initial value for asynchronous reset

Output Signals

- Q_SSR: Output of SSR D latch
- Q_ASR: Output of ASR D latch

Procedure

To do blackbox testing on this module and to provide every possible combination of inputs we declared a ‘random object’ class to hold all the input parameters which we later randomised and feed it to the modules under test. Finally, we monitored and displayed the output values of both the latches.

Simulation and Output Waveform

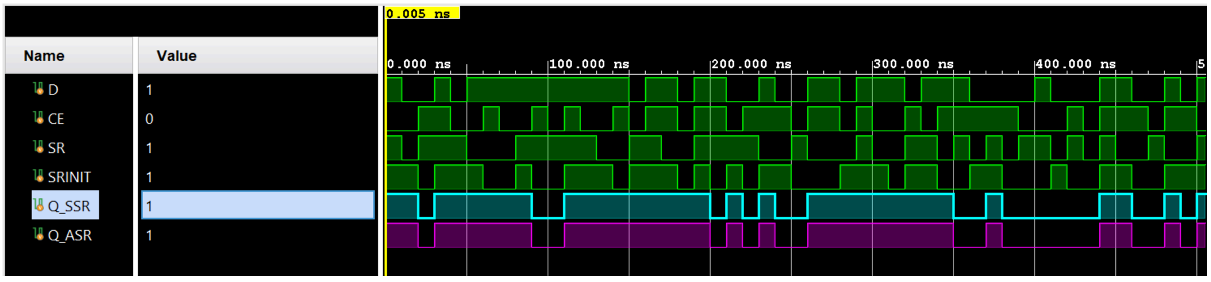


Figure 4

Figure 4 shows the normal operation of the D latch.

nBit Counter

Introduction

The testbench, named nBitCounter_tb, is designed to verify the functionality of both synchronous set/reset (SSR) and asynchronous set/reset (ASR) versions of the n-bit counter module. The testbench module begins by declaring all the logical inputs and outputs to the nBit Counter module and then instantiates an instance of synchronous nBit Counter and an instance of asynchronous nBit Counter. The named port connection to both the instances are similar except the output port which helps us separate the output waveform of the two modules.

Input Signals

- CE: Chip enable
- SR: Reset
- CLK: Clock
- SRINIT: Initial value for the counter
- BIT_SIZE: Size of the counter in bits

Output Signals

- Q_SSR: Output of SSR nBit Counter
- Q_ASR: Output of ASR nBit Counter

Procedure

To do whitebox testing on this module and to cover every possible path described through source code we made the test combinations for every input with SRINIT. Finally, we monitored and displayed the output values of both the Counters.

Simulation and Output Waveform

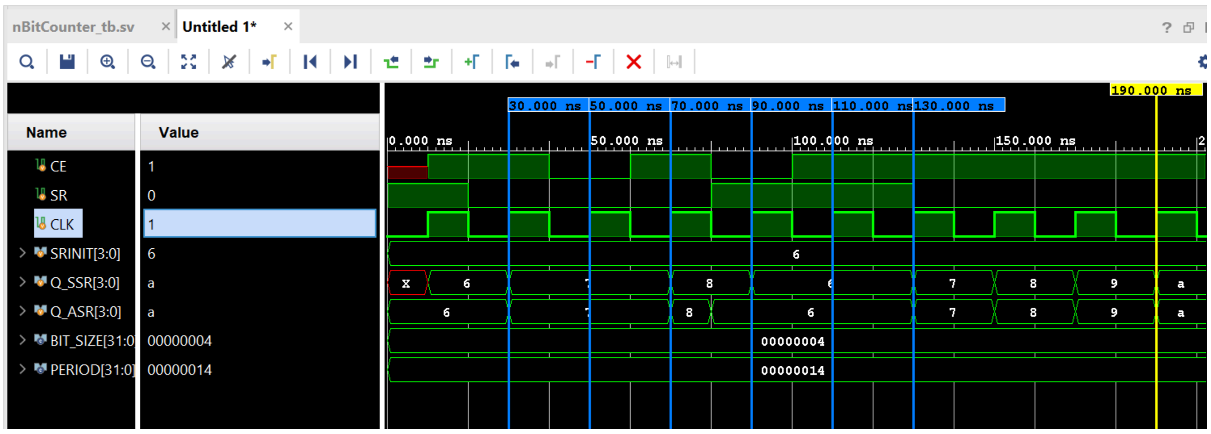


Figure 5

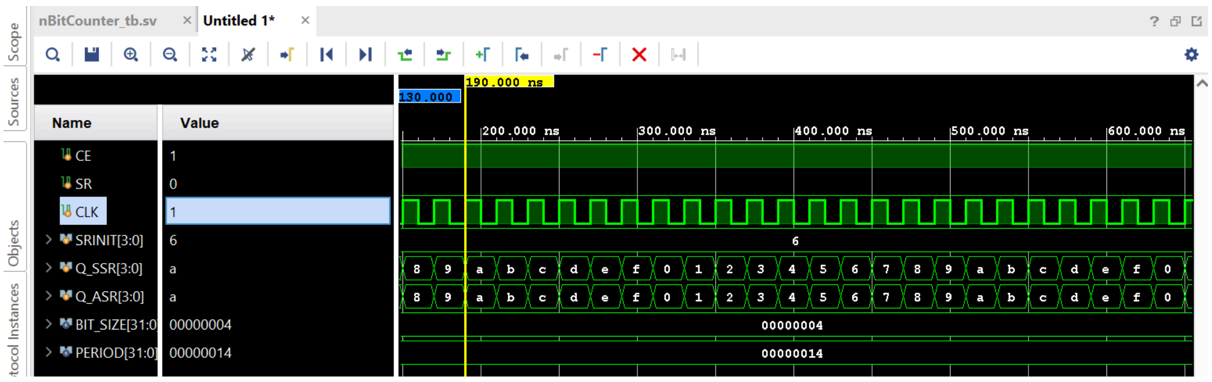


Figure 6

From figure 5 at **30 ns** CE=1 and SR=0, at **50 ns** CE=0 and SR=0, at **90 ns** CE=0 and SR=1 and at **110 ns** CE=1 and SR=1. From Figure 6, we can see the normal operation of the counter. The value of SRINIT is 6 and thus the output at every SR=1 is 6. The counter is **parameterized** but in this case it is set as a 4 bit counter and counts till 15 (decimal) or F (binary) and then restarts from 0.

nBit Register

Introduction

The testbench, named nBitRegister_tb, is designed to verify the functionality of both synchronous set/reset (SSR) and asynchronous set/reset (ASR) versions of the n-bit register module. The n bit register's width can be set during instantiation using parameters. The testbench module begins by declaring all the logical inputs and outputs to the nBit register module and then instantiates an instance of synchronous nBit Register and an instance of asynchronous nBit Register. The named port connection to both the instances are similar except the output port which helps us separate the output waveform of the two modules.

Input Signals

- CE: Chip enable
- SR: Reset
- CLK: Clock
- DIN: Data input
- SRINIT: Initial value for the register
- BIT_SIZE: Size of the register in bits

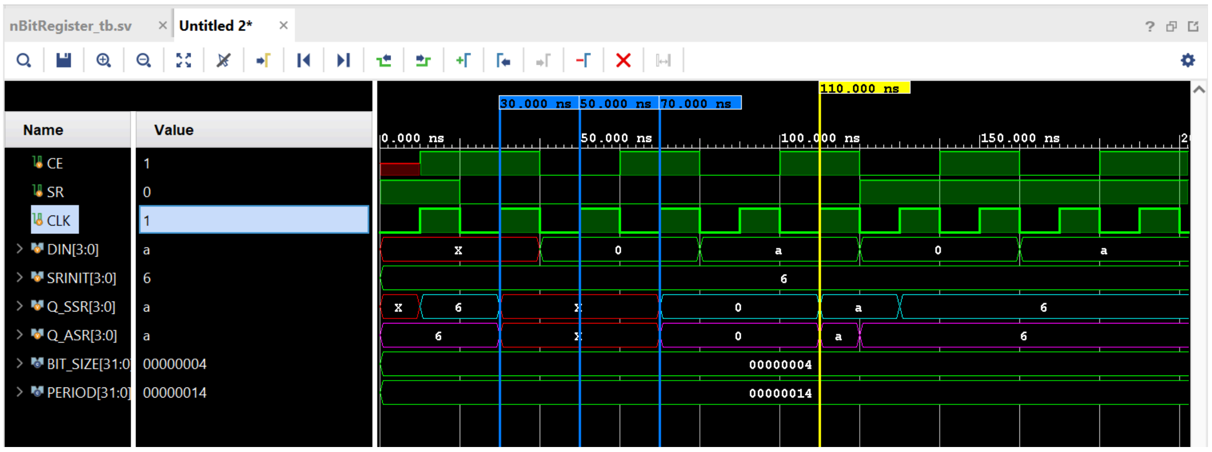
Output Signals

- Q_SSR: Output of SSR nBit Register
- Q_ASR: Output of ASR nBit Register

Procedure

To do whitebox testing on this module and to cover every possible path described through source code we made the test combinations for every input with SRINIT. Finally, we monitored and displayed the output values of both the registers. The simulation results indicate successful verification of the n-bit register module under all test scenarios. The output behaviour of both the SSR and ASR architectures matches the expected results for each test case. The register module demonstrates correct functionality in terms of storing data, responding to set and reset signals, and maintaining output consistency with input.

Simulation and Output Waveform



Name	Value
CE	1
SR	0
CLK	1
DIN[3:0]	a
SRINIT[3:0]	6
Q_SSR[3:0]	a
Q_ASR[3:0]	a
BIT_SIZE[31:0]	00000004
PERIOD[31:0]	00000014

Figure 7

The test cases are: Reset low, enable low, data input = 0

Reset low, enable high, data input = 0

Reset low, enable low, data input = 6 (binary: 0110)

Reset low, enable high, data input = 6 (binary: 0110)

Reset high, enable low, data input = 0

Reset high, enable high, data input = 0

Reset high, enable low, data input = 10 (binary: 1010)

Reset high, enable high, data input = 10 (binary: 1010)

Miscellaneous test: Toggle reset, enable high, data input = 10, and compare output with input.