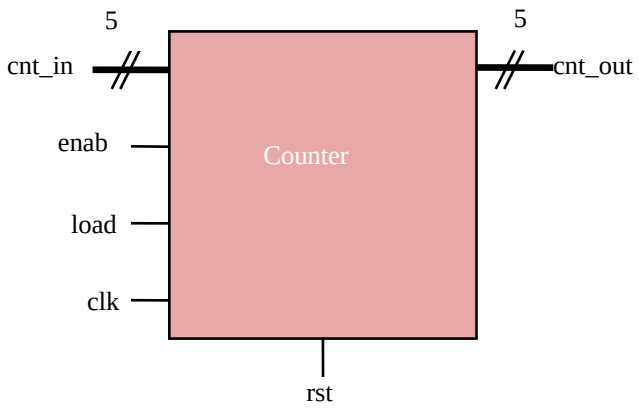
Module 9: Understanding the Simulation Cycle

Lab 9-1 Modeling a Generic Counter

Objective: To use blocking and nonblocking assignments to describe a counter.

The VeriRISC CPU contains a program counter and a phase counter. One generic counter definition can serve both purposes.

In this lab, you create a simple register design as per the specification and verify it using the provided testbench. Please make sure to use the same variable name as the ones shown in block diagrams.



Specifications

- The counter is clocked on the rising edge of `clk`.
- `rst` is active high.
- `cnt_in` and `cnt_out` are both 5-bit signals.
- If `rst` is high, output will become *zero*.
- If `load` is high, the counter is loaded from the input `cnt_in`.
- Otherwise, if `enab` is high, `cnt_out` is incremented, and `cnt_out` is unchanged.

Designing a Generic Counter

1. Change to the *lab10-cntr* directory and examine the following files.

counter_test.v	Counter test

2. Use your favorite editor to create the *counter.v* file and to describe the counter module named as "*counter* ".

3. Parameterize the counter data input and output width so that the instantiating module can specify the width of each instance. Assign a default value to the parameter.

4. Describe the counter behavior in separate combinational and sequential procedures.

Verifying the Counter Design

1. Using the provided test module, check your counter design using the following command with Xcelium™.

```
xrun counter.v counter_test.v (Batch Mode)
```

or

```
xrun counter.v counter_test.v -gui -access +rwc ( GUI Mode)
```

2. You might find it easier to list all the files and simulation options in a text file and pass the file into the simulator using the `-f xrun` option.

```
xrun -f filelist.txt -access rwc
```

You should see the following results.

```
At time 20 rst=0 load=1 enab=1 cnt_in=10101 cnt_out=10101
At time 30 rst=0 load=1 enab=1 cnt_in=01010 cnt_out=01010
At time 40 rst=0 load=1 enab=1 cnt_in=11111 cnt_out=11111
At time 50 rst=1 load=1 enab=1 cnt_in=11111 cnt_out=00000
At time 60 rst=0 load=1 enab=1 cnt_in=11111 cnt_out=11111
At time 70 rst=0 load=0 enab=1 cnt_in=11111 cnt_out=00000
TEST PASSED
```

3. Correct your counter description as needed.