Step By Step Learning Robei

II. Counter Design

Robei LLC

1. Objective

Counter is widely used in digital design. It can be used in counting clock signal, frequency divider and sequence of signals generation. In this lab, we will use Robei and verilog to design a four bits counter.

2. Requirements

Design a counter to count every rising edge if clock, and output the value directly. A four bits counter can count range from 0 to F, which means it can count max value is 16. The waveform of your counter should be similar to fig.1.

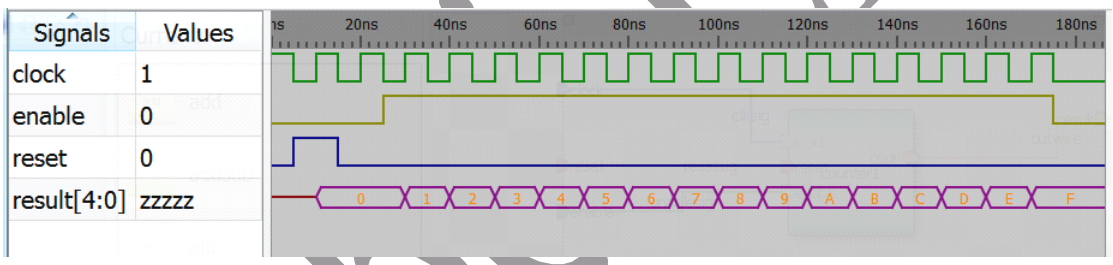


Fig1. Four bits Counter waveform

3. Procedure

3.1 Module Design

1) Create a new module. Click on icon  in toolbar, or select drop-down menu “New” from menu “File”, a dialog will pop up shows in fig.2. You can set the properties of your design in this dialog.

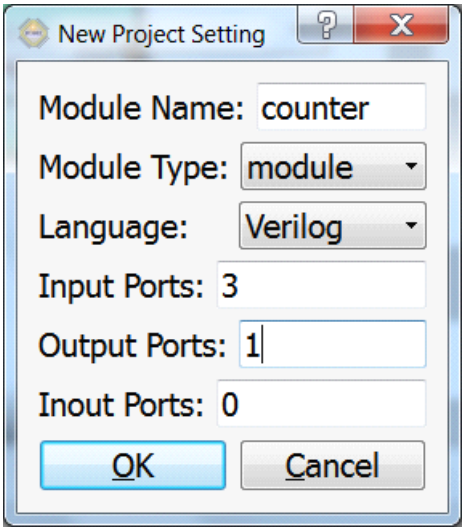


Fig 2. Create a new module

Press “OK” when you finish, Robei will generate a new module with name “counter” as shown in fig.3.

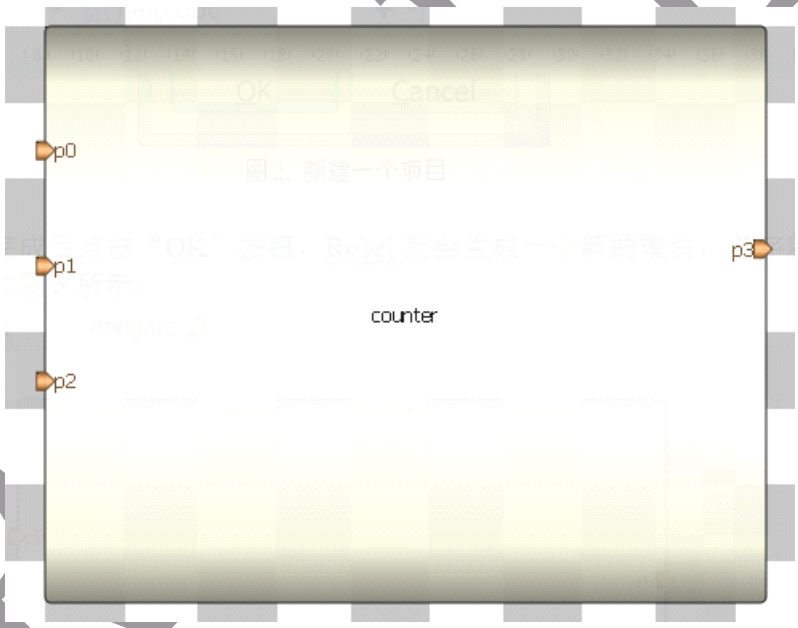
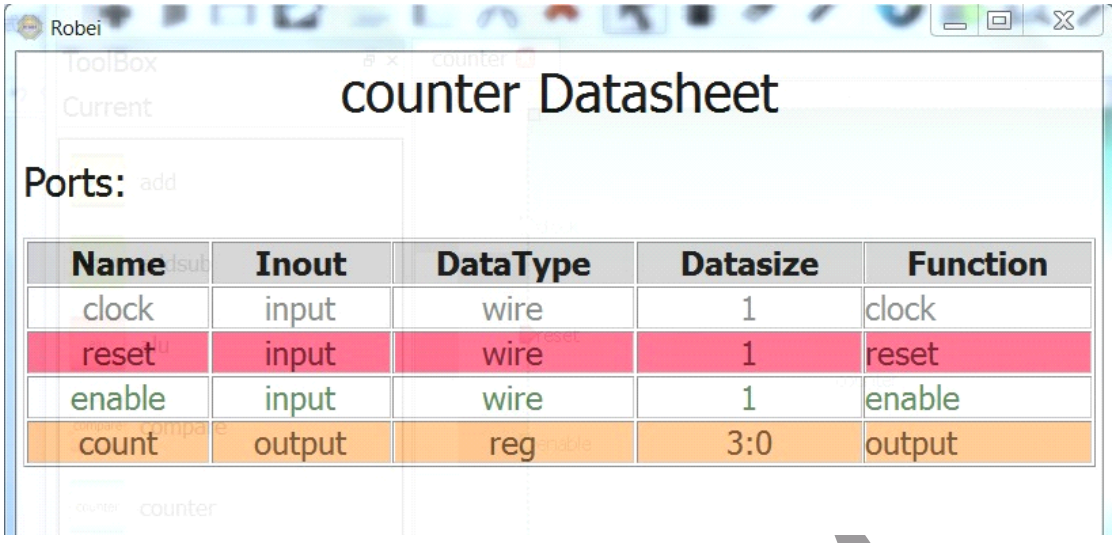


Fig. 3. Counter interface

2) Modification. Select “p0”, all the properties of this port will show up in property editor. You can modify the color property of each port to distinguish with each other. Please changes the properties of ports following the table in fig.4. When you press “F1” after selecting the current module, a datasheet like fig.4 will show up. After the modification, interface will changed to fig. 5.



Name	Inout	DataType	Datasize	Function
clock	input	wire	1	clock
reset	input	wire	1	reset
enable	input	wire	1	enable
count	output	reg	3:0	output

Fig. 4. Datasheet of conter

Fig. 5. Interface after modification

3) Code algorithm. Click on the “Code” tab under module, code editor will show up for you to typing algorithm code for this module (shows in fig.6).

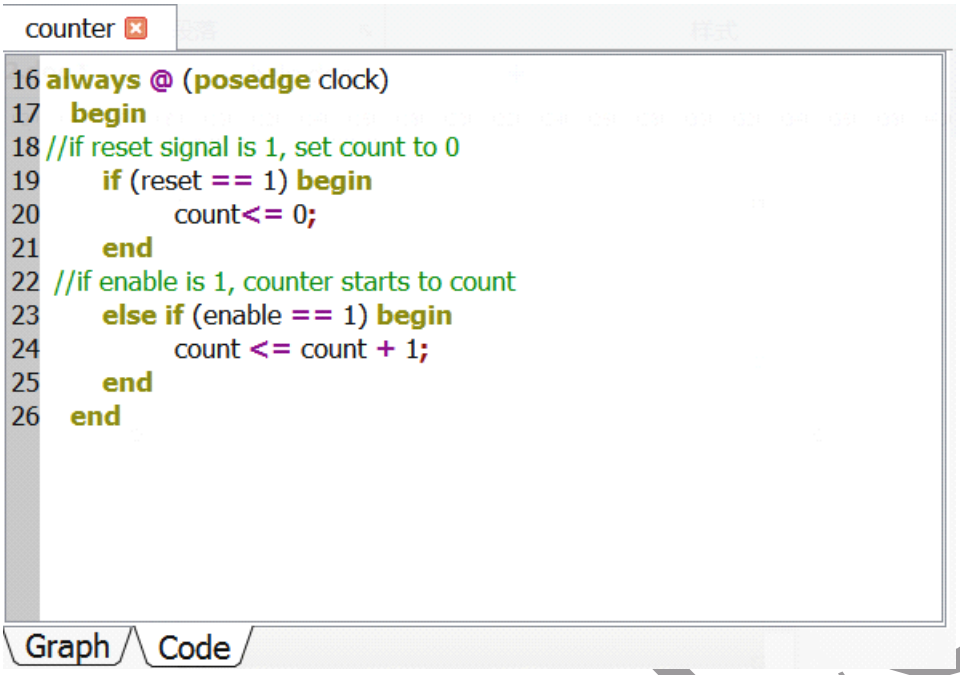


Fig. 6. Code input

Type in the following verilog code:

```
always @ (posedge clock)      //learn how to use always and sensitive signal.
   begin
     //learn how to write if else in Verilog
      if (reset == 1) begin
          count<= 0;
      end
     //if enable is 1, counter starts to count
     else if (enable == 1) begin
          count <= count + 1;
      end
   end
```

4) Save. Press icon  on Toolbar, or click on drop-down menu “Saveas” in menu “File”, save this model in an empty folder. Note: this empty folder can not have white space in path, and can only use English characters.

5) Run. Press on icon  or click on drop-down menu “Run” in menu “Build”.

In this process, Robei will generate complete Verilog code and check. If there is any error, it will show error message in Output window. Please follow the instruction to correct any errors. After that, congratulation, you have complete the counter design.

3.2 Testbench Design

1) Create a new design. Press on icon  in toolbar, in the pop up dialog, fill in blank with the following parameters in fig. 7.

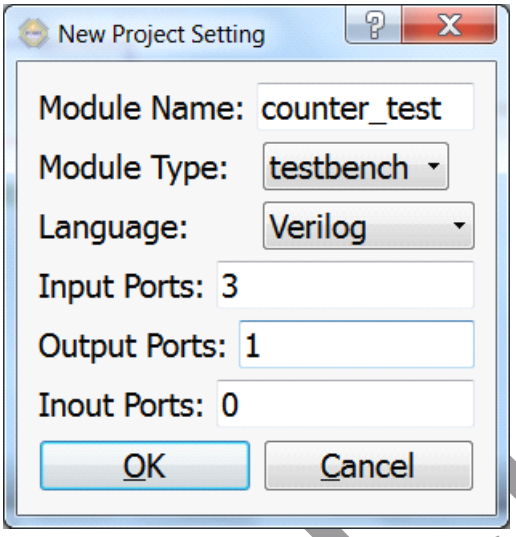


Fig7. Create a new testbench

2) Modify color of ports. Select each port, modify color from property editor. Please make sure the properties of each port are same as shown in fig.8.

Name	Inout	DataType	Datasize
clock	input	wire	1
reset	input	wire	1
enable	input	wire	1
result	output	wire	3:0

Fig. 8 Ports' properties

3) Save as testbench. Click on icon  in Toolbar, save testbench to the same location as "counter" .

4) Add model. Under category "Current" in Toolbox, there will be one model show up as fig.9. Click on this model and click on the testbench design, Robei will add this model in automatic.

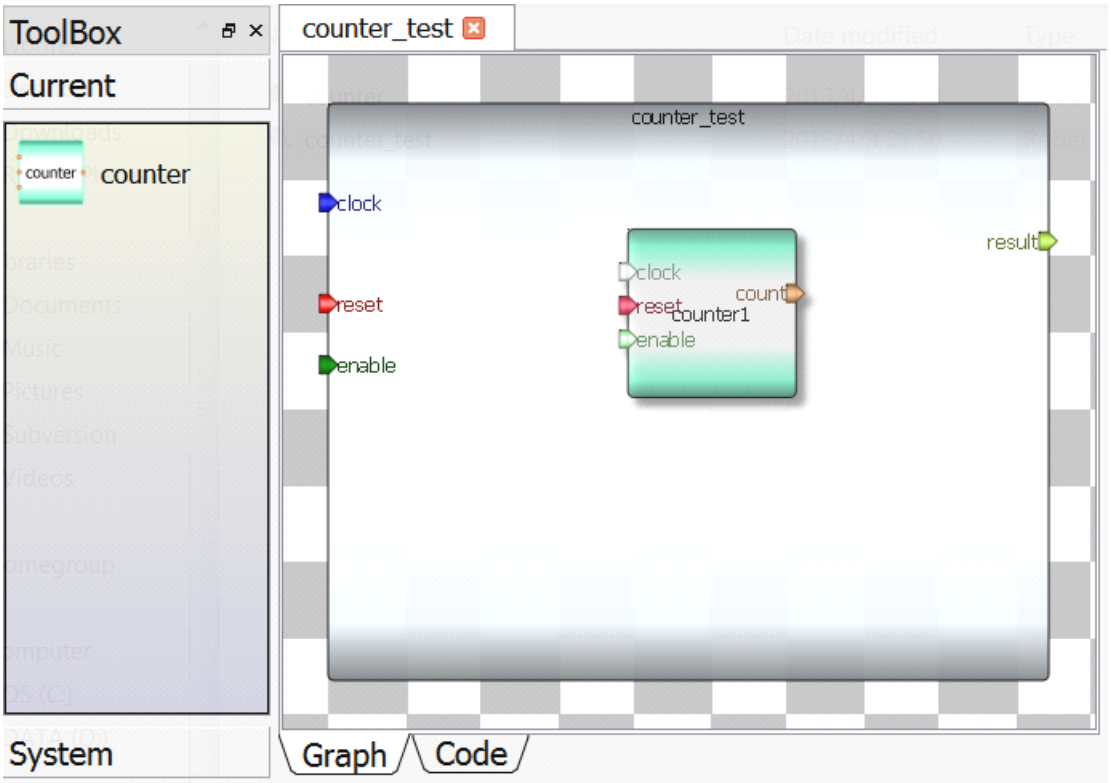


Fig. 9. Add model

5) Connect ports. Click on icon  in Toolbar, or select “Connect” from menu “Tool”. When you connect, remember to check the color of connection, as it will inherent from the first port directly. If you want to quit connection mode, press on icon  .

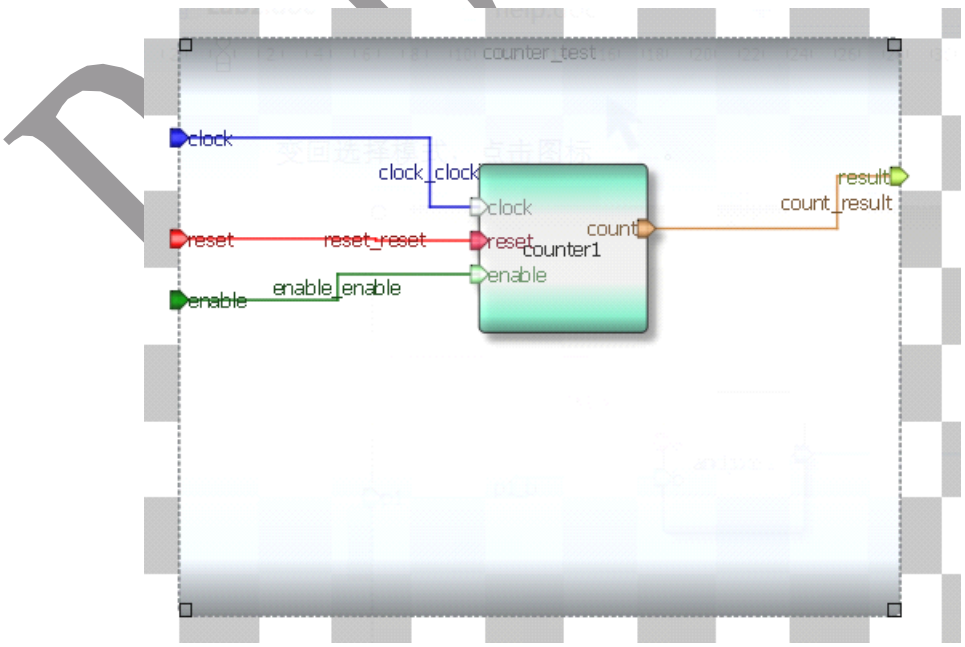


Fig. 10. Connect ports

6) Code stimulate. Press “Code” tab under testbech, type in the following code as stimulate for simulation. Remember to use $finish to end this simulation.

```
initial begin
   clock = 1;
   reset = 0;
   enable = 0;
   #5 reset = 1;
   #10 reset = 0;
   #10 enable = 1;
   #150 enable = 0;
   #5 $finish;
end
always begin              // learn how to generate clock
    #5 clock=~clock;    //every 5 delay, clock invert once.
end
```

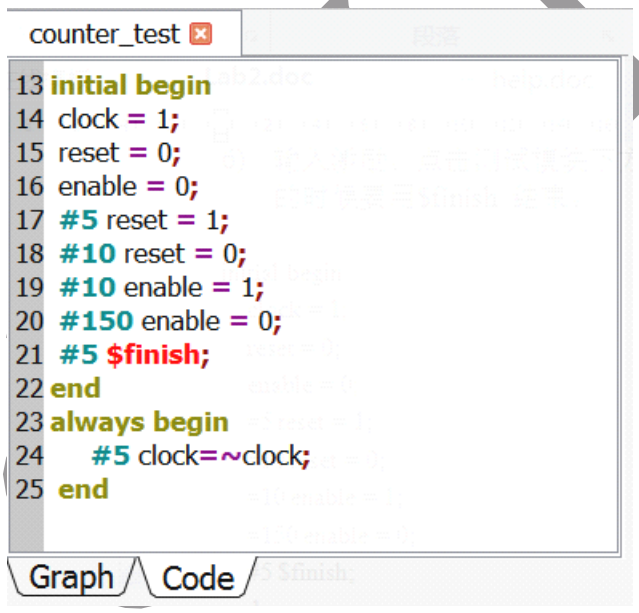


Fig. 11 Stimulation code

7) Run simulation and check the waveform. Click on  , check output information, if there is no error, then click on  . The Waveform window will show up. Click on signals listed in Workspace, add to wave window. Press  on toolbar of this window, it will zoom full of the wave. Analyze the waveform result and compare with your lab requirement.

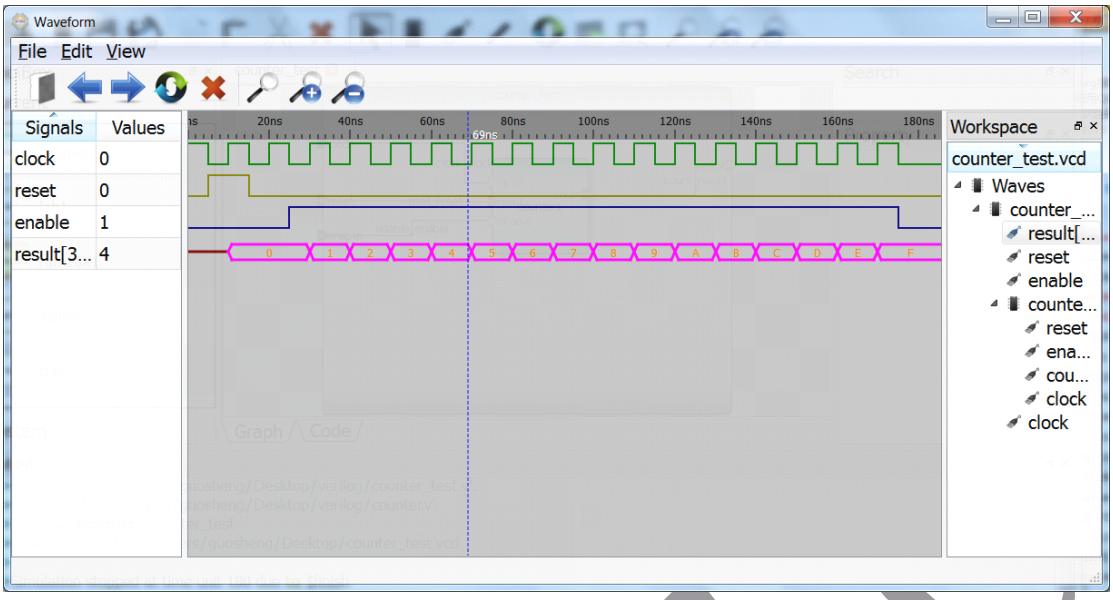


Fig. 12. View the waveform

4. Questions

1) How to use Robei to design a count down counter. For example, every clock cycle, the counter will reduce 1 instead of add 1. Test your counter with a testbench.
2) Use counter to realize frequency division of 2,4 and 16 with duty cycle 50%. Duty cycle is high voltage last time/period.