

Bidirectional Counter

Tutorial:

1. Write verilog code:

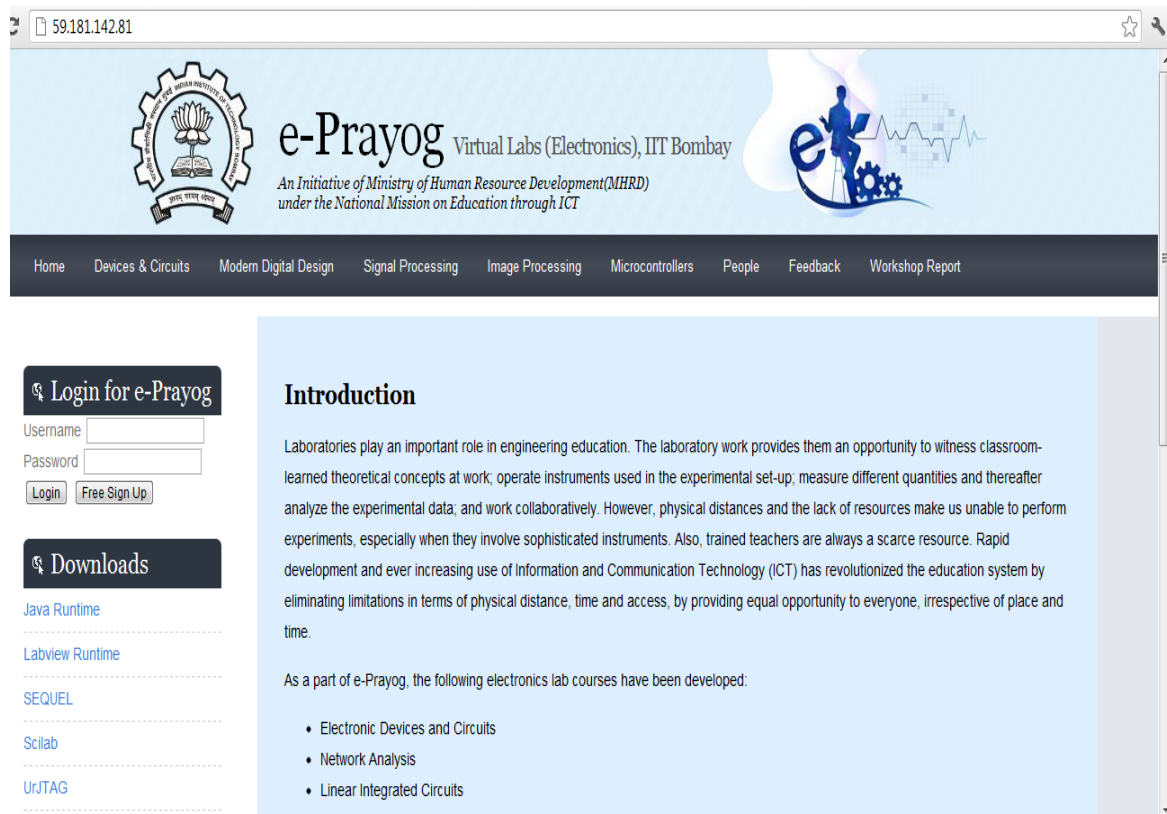
Sample code:

```
module updown_counter(clk, reset, mode, count);
    input clk, reset, mode;
    output [ 7 :0 ] count;
    wire reset, mode;
    reg [ 7 :0] count;

    always @(posedge clk )
    begin
        if(!reset)
            count <= 8'b0100;
        else if (mode == 0)
            count <= count + 1;
        else if (mode == 1)
            count <= count - 1;
    end
endmodule
```

2. Save this code as updown_counter.v and save it in a directory.

3. Visit e-prayog webpage: <http://59.181.142.81/>



The screenshot shows the e-Prayog Virtual Labs (Electronics) IIT Bombay website. The header includes the IIT Bombay logo and the text "e-Prayog Virtual Labs (Electronics), IIT Bombay". Below the header is a navigation bar with links: Home, Devices & Circuits, Modern Digital Design, Signal Processing, Image Processing, Microcontrollers, People, Feedback, and Workshop Report. The main content area is divided into two columns. The left column contains a "Login for e-Prayog" section with fields for Username and Password, and buttons for Login and Free Sign Up. Below this is a "Downloads" section with links to Java Runtime, Labview Runtime, SEQUEL, Scilab, and UrTAG. The right column contains an "Introduction" section with text about the importance of laboratories in engineering education and a list of electronics lab courses developed as part of e-Prayog: Electronic Devices and Circuits, Network Analysis, Linear Integrated Circuits, and Signals and Systems.

4. Login on the page if you are registered or sign-up if not registered.

 **Login for e-Prayog**

Username

jasveer

Password

.....

Login

Free Sign Up


 **Login for e-Prayog**

Welcome jasveer


[Sign Out](#)

5. Now visit the webpage of Remote Triggered FPGA based Automation system:
<http://59.181.142.81/fpga/automation.php>


59.181.142.81/fpga/automation.php



e-Prayog Virtual Labs (Electronics), IIT Bombay
An Initiative of Ministry of Human Resource Development (MHRD)
under the National Mission on Education through ICT




HomeDevices & CircuitsModern Digital DesignSignal ProcessingImage ProcessingMicrocontrollersPeopleFeedbackWorkshop Report

 **Login for e-Prayog**

Welcome jasveer

[Sign Out](#)

 **Downloads**

[Java Runtime](#)

[Labview Runtime](#)

[SEQUEL](#)

[Scilab](#)




[UrJTAG](#)

[Altera Quartus Web Edition](#)

[Apple QuickTime Player](#)

[Go Back](#)

Automated FPGA System


Video Tutorials

Step 1: Upload Design Under Test Verilog File

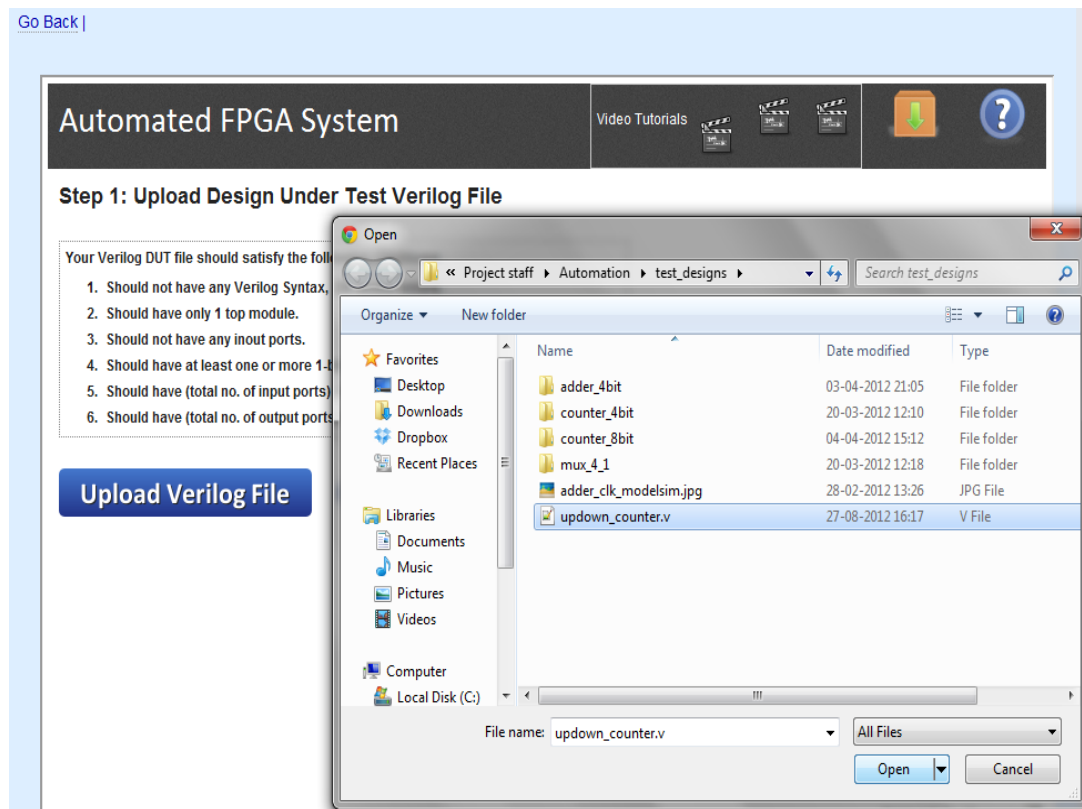
Your Verilog DUT file should satisfy the following conditions:

1. Should not have any Verilog Syntax, OR Netlist errors.
2. Should have only 1 top module.
3. Should not have any inout ports.
4. Should have at least one or more 1-bit input ports for the clock pin.
5. Should have (total no. of input ports) ≤ 17
6. Should have (total no. of output ports) < 8

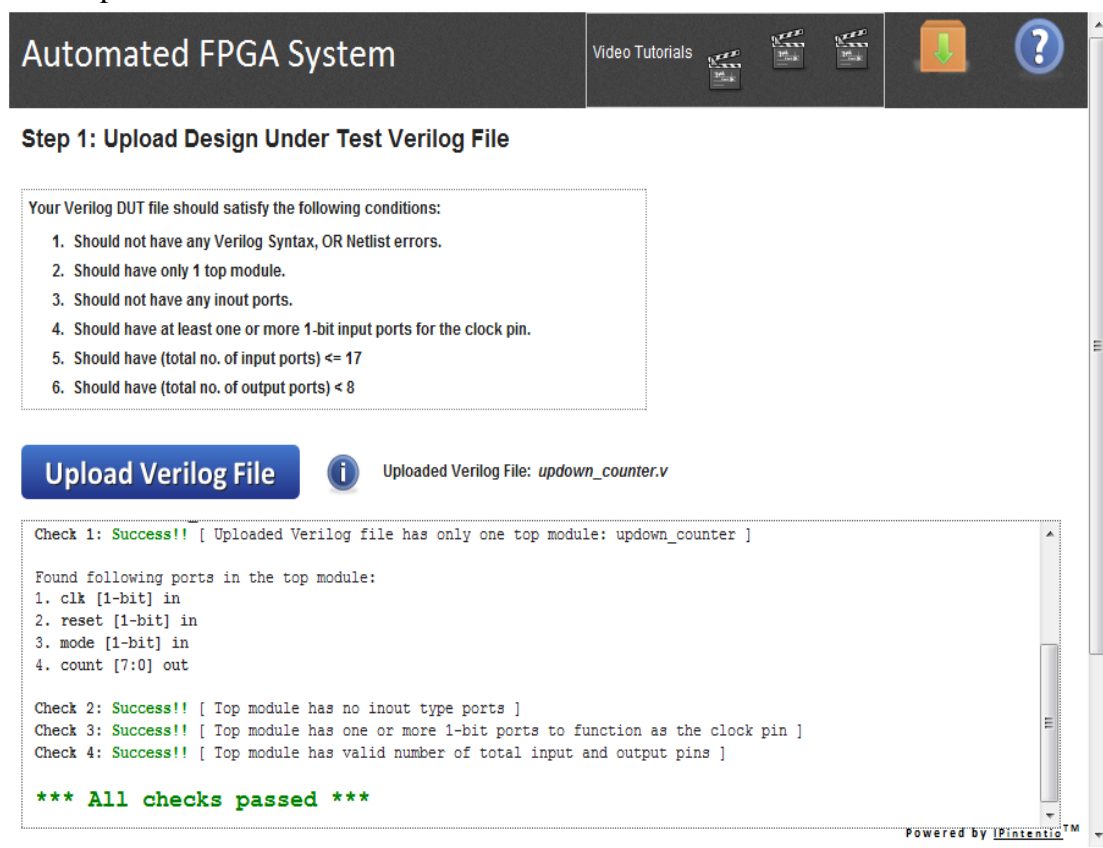
Upload Verilog File



6. Click on upload verilog file button and upload your verilog file “updown_counter.v”





Now click on open:



7. After the browser specifies “All checks passed”, select the clock pin as “clk”.


Automated FPGA System

Video Tutorials



5. Should have (total no. of input ports) ≤ 17
6. Should have (total no. of output ports) < 8

Upload Verilog File

 Uploaded Verilog File: *updown_counter.v*

Check 1: **Success!!** [Uploaded Verilog file has only one top module: updown_counter]


Found following ports in the top module:
1. clk [1-bit] in
2. reset [1-bit] in
3. mode [1-bit] in
4. count [7:0] out

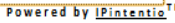
Check 2: **Success!!** [Top module has no inout type ports]
Check 3: **Success!!** [Top module has one or more 1-bit ports to function as the clock pin]
Check 4: **Success!!** [Top module has valid number of total input and output pins]

***** All checks passed *****

Step 2: Select the clock pin

☒ clk ☐ reset ☐ mode



Set clock pin

Powered by 


8. Give the test inputs for clock instances where the input is changing.

Automated FPGA System

Video Tutorials



☒ clk ☐ reset ☐ mode

Set clock pin


 Selected Clock Pin: *clk*


Step 3: Provide input vectors & Execute Design

Clock Cycle	reset	mode
0	0	0
5	1	
11		0
16	0	
4		
5		
6		

Add Row

Delete Row

Set Input Vectors & Execute

Powered by 

9. Click on “Set input vectors and Execute” - this will compile the design, assign pins according Altera DE2-70 board, synthesize it and execute it on the board with the given test inputs to generate a vcd file output. (this will take about some time from 1-3 mins depending on the design complexity).

Automated FPGA System
Video Tutorials

Set clock pin ⓘ Selected Clock Pin: *clk*

Step 3: Provide input vectors & Execute Design

Clock Cycle	reset	mode
0	0	0
5	1	
11		0
16	0	
4		
5		
6		

Add Row Delete Row

ⓘ ⚙ Working..

Powered by iPintento™

The log will appear and the output vcd file will be available for download:

Automated FPGA System
Video Tutorials

4		
5		
6		

Add Row Delete Row

Set Input Vectors & Execute ⓘ Download Value Change Dump (VCD) file for the execution [here.](#) ⓘ

FPGA Execution Output

```

Info: *****
Info: Running Quartus II Programmer
Info: Version 11.0 Build 208 07/03/2011 Service Pack 1 SJ Web Edition
Info: Copyright (C) 1991-2011 Altera Corporation. All rights reserved.
Info: Your use of Altera Corporation's design tools, logic functions
Info: and other software and tools, and its AMPP partner logic
Info: functions, and any output files from any of the foregoing
Info: (including device programming or simulation files), and any
Info: associated documentation or information are expressly subject
Info: to the terms and conditions of the Altera Program License
Info: Subscription Agreement, Altera MegaCore Function License
Info: Agreement, or other applicable license agreement, including,
Info: without limitation, that your use is for the sole purpose of
Info: programming logic devices manufactured by Altera and sold by
  
```

Powered by iPintento™