Lab 3: Clock Divider and LED Controller

Submission Due Dates:
Source Code: 2020/10/13 18:30
Report: 2020/10/18 23:59

Objective

Getting familiar with the clock divider and LED control on the FPGA demo board.

Action Items

1 `Module: clock_divider` (10%)

Write a Verilog module for the clock divider that divides the frequency of the input clock by 2^{25} to get the output clock. Here is the template you should use:

```
module clock_divider (clk, clk_div);
    parameter n = 25;
    input clk;
    output clk_div;

    // add your design here

endmodule
```

2 `lab3_1.v` (30%)

Write a Verilog module of the LED Controller which is synchronous with the positive clock edges. The clock frequency is obtained by dividing the frequency of Basys3's on-board clock (i.e., at 100MHz) by 2^{25}. Also, download and run your LED Controller on the FPGA board. You should use the clock divider you design in Action Item 1.

Here are the inputs and outputs constraints:

- ✓ If **rst** == **1**: the left-most LED (LD15) is ON, and others are OFF.
  - The reset is positive-edge triggered.
- ✓ If **en** == **0**: hold the LEDs unchanged.
- ✓ If **en** == **1**:
  - If **dir** == **1**: the LED will begin to shift (to turn on and off one by one) from the right to the left synchronized to the clock.
  - If **dir** == **0**: the LED will begin to shift (to turn on and off one by one) from the left to the right synchronized to the clock.

IO Connection:

clk	connected to W5
rst	connected to W16
en	connected to V17
dir	connected to V16
led	connected to LD15-LD0

You should use the following template for your design and name the file as "lab3_1.v".

```
module lab3_1(clk, rst, en, dir, led);
    input clk;
    input rst;
    input en;
    input dir;
    output [15:0] led;

    // add your design here

endmodule
```

Demo: https://www.youtube.com/watch?v=_gJu3U_iTjw

3 lab3_2.v (30%)

Write a Verilog module of the LED Controller which is synchronous with the clock whose frequency is obtained by dividing the frequency of Basys3's on-board 100MHz clock by either 2^{23} or 2^{25}.

There are two LED runners, called Mr. 1 and Mr. 3.

- ✓ Mr. 1 and Mr. 3 race on the 16 LEDs of the FPGA Demo board.
- ✓ We represent Mr. 1 by one single LED.
- ✓ We represent Mr. 3 by three consecutive LEDs. (His position is determined by the middle LED.)
- ✓ If **rst** == **1** (positive-edge-triggered reset)
  - ■ Mr. 1's position is at LD15
  - ■ Mr. 3's position is at LD1
  - ■ E.g., the initial state: (●: LED on, ○: LED off)

    Mr. 1: (LD15) ●○○○○○○○○○○○○○○○ (LD0)

    Mr. 3: (LD15) ○○○○○○○○○○○○○●●● (LD0)
- ✓ Mr. 1 always shifts from left to right; Mr. 3 always shifts from right to left.

- ✓ If **en** == **0**: Mr. 1 and Mr. 3 hold at their current positions.
- ✓ If **en** == **1**:
  - If **speed** == **0**: **Mr. 3** runs at the clock rate of (100 MHz / 2^{25}); **Mr. 1** runs at the clock rate of (100 MHz / 2^{23}), respectively.
  - If **speed** == **1**: **Mr. 3** runs at the clock rate of (100 MHz / 2^{23}); **Mr. 1** runs at the clock rate of (100 MHz / 2^{25}), respectively.

IO Connection:

clk	connected to W5
rst	connected to W16
en	connected to V17
speed	connected to V16
led	connected to LD15-LD0

You should use the following template for your design and name the file as "lab3_2.v".

```
module lab3_2 (
  input clk,
  input rst,
  input en,
  input speed,
  output [15:0] led
  );

  // add your design here

endmodule
```

Demo: https://www.youtube.com/watch?v=Whw_Nqs1el0

4 lab3_3.v (30%)

Write a Verilog module of the LED Controller which is synchronous at the clock rates of either (100MHz / 2^{23}) or (100MHz / 2^{25}), dividing from Basys3's 100MHz clock.

Here are three LED runners, called Mr. 1A, Mr. 1B and Mr. 3:

- ✓ If **rst** == **1**: set Mr. 1A, Mr. 1B and Mr. 3's positions to LD15, LD0 and LD10, respectively.
  - ➢ E.g., the initial state: (●: LED on, ○: LED off)

    Mr. 1A: (LD15) ●○○○○○○○○○○○○○○○ (LD0)

Mr. 3: (LD15) ○○○○●●●○○○○○○○○○ (LD0)

Mr. 1B: (LD15) ○○○○○○○○○○○○○○○● (LD0)

✓ If **en** == **0**: Mr. 1 and Mr. 3 will all hold at their current positions.

✓ If **en** == **1**:

- **Mr. 1A** will begin to shift from left to right
- **Mr. 1B** will begin to shift from right to left
- **Mr. 3** will begin to shift from left to right

- If **speed** == **0**: **Mr. 3** runs at the clock rate of (100 MHz / 2^{25}); **Mr. 1A** and **Mr. 1B** run at the clock rate of (100 MHz / 2^{23}), respectively.
- If **speed** == **1**: **Mr. 3** runs at the clock rate of (100 MHz / 2^{23}); **Mr. 1A** and **Mr. 1B** run at the clock rate of (100 MHz / 2^{25}), respectively.

✓ When Mr. 1A and Mr. 1B reach the left-most end and right-most end, they will change the running direction.

✓ When Mr. 3 reach the position of LD5 or LD10, Mr. 3 will change his direction.

➢ Mr. 3's moving range is restricted.

Mr. 3: (LD15) ○○○○○●●●○○○○○○○○○(LD0)

✓ When Mr. 1A collides with Mr. 3, Mr. 1A will change its direction, but Mr. 3 will not. Similarly, when Mr. 1B collides with Mr. 3, Mr. 1B will change its direction, but Mr. 3 will not.

➢ Collision Situation 1 (Touch): Mr. 1A (Mr. 1B) touches one end of Mr.3

e.g.

Mr. 1A: (LD15) ○○○  ●○○○○○○○○○○(LD0)

Mr. 3: (LD15) ○○○○○○●●●○○○○○○○(LD0)

Mr. 1B: (LD15) ○○○○○○○○○●○○○(LD0)

➢ Collision Situation 2 (Overlap): Mr. 1A (Mr. 1B) overlaps one end of Mr. 3

e.g.

Mr. 1A: (LD15) ○○○○  ●○○○○○○○○ (LD0)

Mr. 3: (LD15) ○○○○○○●●●○○○○○○ (LD0)

Mr. 1B: (LD15) ○○○○○○○○●  ○○○○ (LD0)

(Hint: At what condition will Collision Situation 2 happen? Given the different clock rates of the two runners.)

➢ Collision Situation 3 (Overlap): Mr. 1A (Mr. 1B) overlaps the middle of Mr. 3.

e.g.

Mr. 1A: (LD15) ○○○○○  ●○○○○○○○○ (LD0)

Mr. 3: (LD15) ○○○○○○●●●○○○○○○○ (LD0)

Mr. 1B: (LD15) ○○○○○○○●  ○○○○○○ (LD0)

IO Connection:

clk	connected to W5
rst	connected to W16
en	connected to V17
speed	connected to V16
led	connected to LD15-LD0

You should use the following template for your design and name the file as "lab3_3.v".

```
module lab3_3 (
  input clk,
  input rst,
  input en,
  input speed,
  output [15:0] led
  );

   // add your design here

endmodule
```

Demo: https://www.youtube.com/watch?v=5zMsRTSvvWo

Attention

- ✓ Please refer to Lab 0 and Lecture 02 for the FPGA implementation flow using Vivado.
- ✓ You may have to change your runtime (ns) in “Simulation Settings” to fit the testbench settings before the simulation.

- ✓ You should hand in three Verilog file, **lab3_1.v, lab3_2.v, lab3_3.v**. If you have multiple modules for an Action Item, you must integrate them into a single Verilog file (e.g., lab3_1 and clock_divider in lab3_1.v; lab3_2 and clock_divider in lab3_2.v, etc.).
  - ■ **Upload each source file directly, DO NOT hand in a compressed ZIP or RAR file!**

- ✓ **DO NOT** copy-and-paste code segments from the PDF materials. Occasionally, it will also paste invisible non-ASCII characters and lead to hard-to-debug syntax errors.

- ✓ You should also hand in your report as lab3_report_StudentID.pdf (i.e., lab3_report_108080001.pdf).

- ✓ You should be able to answer questions of this lab from TA during the demo.

- ✓ You need to generate the bitstream files before the lab demo. Prepare three separate bitstream files for lab3_1, lab3_2, and lab3_3, respectively, to make the demo process smooth.