ELEX 2117 : Digital Techniques 2
2021 Fall Term

Sequential Logic Design with Verilog

Introduction

In this lab you will display your BCIT ID, four digits at a time, on the 4-digit, 7-segment LED display.

Your circuit should display the last four digits of *your* BCIT ID when keypad button [1] is pressed and the first four digits when it is not. For example, if your BCIT ID is **A00123456** the display should show and 3456 when [1] is pressed and 0012 otherwise.

You will use the same components as in the previous lab, connected the same way.

Multiplexed LED Display

Review the schematic of the LED display in the previous lab. Note that the four digits on the LED display share the same segment (anode) connections. The seven segments on each digit have a common cathode connection. You can therefore only display one digit at a time. But cycling through the digits faster than the eye can perceive[1] makes it appear that all digits are displayed simultaneously.

Design

Your design will consist of two parts.

One enables the digits sequentially, using a register whose value continuously cycles through the required values. For example, if you have a clock signal `clock` (see below):

```
logic digit, digit_next ;
assign digit_next = ... ; // some function of digit
always @(posedge clock) digit = digit_next ;
```

Another must set the segments (`a` through `g`) to the correct values. These will depend on the enabled digit and whether [1] is pressed or not. This part will be similar to the previous lab.

Component Connections

The FPGA board, keypad and LED display should be connected as in the previous lab.

[1] The flicker fusion threshold.

Procedure

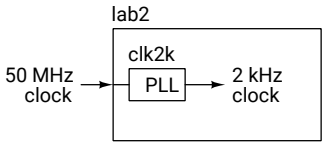
Follow the general procedure in the Software Installation and Use document on the course website to create a project, compile it and configure the FPGA.

Import Pin Assignments

You can import the pin assignments from the previous lab if you are not changing them (this is recommended). Select **Assignments** / **Import Assignments...** and select your `lab1.qsf` file. Click on **Categories...**, un-check everything except **Pin and Location Assignments** and click on **OK**. Check that the pin assignments are correct. You may need to add the weak pull-up resistor assignments on the `col*` inputs.[2]

2 kHz Clock Generator

Digital designs for use by others is sometimes called IP (Intellectual Property). FPGA manufacturers supply IP to access on-chip features. In this lab you will use the **ALT_PLL** IP block to generate a 2 kHz internal clock from the external 50 MHz clock on the FPGA board:



Select **Tools** / **IP Catalog**. In the IP Catalog window select **Library** / **Basic Functions** / **Clocks; PLLs and Resets** / **PLL**. Right-click on **ALTPLL** and select **+ Add Component**. Enter a meaningful file name ending in `.v` (e.g. `clk2k.v`) and choose the Verilog file type.

This will bring up a multi-page multi-tab "MegaWizard Plug-In Manager". You need only change a few things. Under **Parameter Settings** / **General/Modes** select an input frequency for `inclk0` of 50 MHz. Click **Next** and under **Inputs/Lock** uncheck all Optional Inputs and Outputs. Click Next

[2] These don't seem to be imported.

until you get to **Output Clocks** / **clk c0**. Set the output frequency to 0.002 MHz (2 kHz). Click **Finish**.

Quartus will offer to add a **.qip** file to your project. Click on **Yes**. This will add a Verilog file named **clk2k.v** defining a 2 kHz clock generator module named **clk2k** that you can use ("instantiate") in your project with the following statement:

```
logic clock ;
clk2k c0 ( clk50, clock ) ;
```

Where **clk50** is the 50 MHz clock input that you must define as an input to your **lab2** module and **clock** is the 2 kHz clock signal output by the PLL clock generator.

Connect the keypad and LED to the FPGA board. Compile your design and program the FPGA. Test your design and fix any errors.

Submissions

Pre-Lab Report

Submit the following to the appropriate Assignment folder on the course website:

1. A PDF document containing:
    - A block diagram of your design.
    - A listing of your Verilog code.

Post-Lab Report

Submit the following to the appropriate Assignment folder on the course website:

1. A PDF document containing:
    - A listing of your Verilog code.
    - A screen capture of your compilation report.
2. If you were not able to demonstrate your solution to the lab instructor during your scheduled lab period, submit a video showing the keypad and the LED display as you push the 1 key.

Follow the *Report and Video Guidelines* and *Coding Guidelines* documents on the course website.