Combinational Logic Design with Verilog

Revision 2: Only top keypad row to be set low.

Introduction

In this lab you will connect a 4x4 matrix keypad and a 4-digit, 7-segment LED display to the FPGA board and write a Verilog module to display the last four digits of your student number on one of the digits of the LED display.

You will need the following from your ELEX 1117 parts kits:

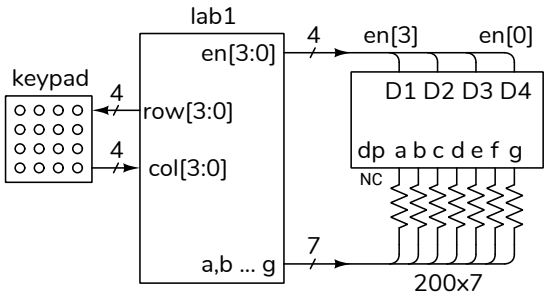
- solderless breadboard
- 11 M-F ("Dupont" or "Berg") pin-header jumpers
- "Byte-Blaster" JTAG interface

and the following from your ELEX 2117 parts kit:

- EP4CE6E22C8N FPGA board, and micro-USB power connector,
- an FJ5463AH 4-digit 7-segment common-cathode LED display
- seven (7) 200 Ω resistors
- 4 × 4 matrix keypad

Circuit Description

The following diagram shows how the keypad and LED are connected to the FPGA and the suggested signal names:



Your circuit should display the last three digits of *your* BCIT ID in the rightmost digit position when the keypad buttons 1, 2 and 3 are pressed. For example, if your BCIT ID is A00123456 then when button 1 is pressed the rightmost LED should display 4 and when button 3 is pressed it should display 6. Anything else should produce a blank display.

"Active-low" means an input or output is at a low logic level when it's true. We will consider a button to be "true" when pressed and a segment "true" when lit.

Component Connections

FPGA Board

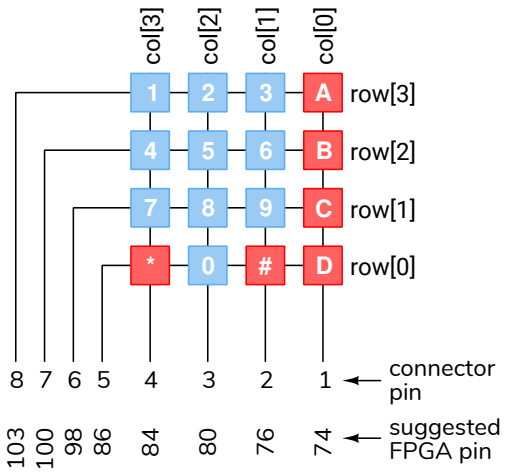
The FPGA has 144 pins. Eighty-two (82) of these are available on two 48-pin headers and will be used to connect components such as the LED display and keypad. The FPGA pin numbers are marked on the PCB. The remaining 14 header pins provide ground, 3.3 V, and 5 V supplies.

The FPGA's I/O pins use 3.3V logic levels. To avoid damaging the board, *never connect your circuits to an external power supply or use the on-board 5V supply*.

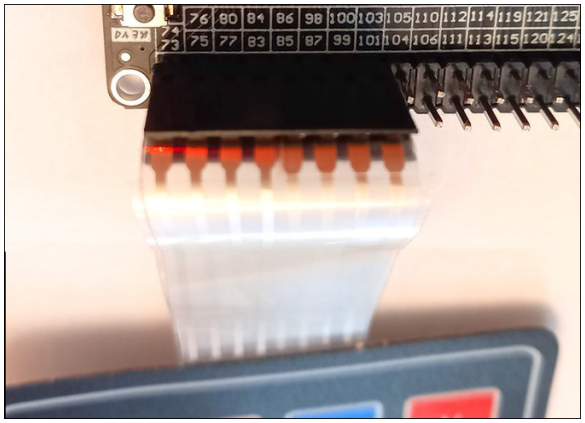
4x4 Matrix Membrane Keypad

This inexpensive keypad consists of two sheets of plastic, one with horizontal traces and the other with vertical traces printed in conductive ink. These sheets are separated by a spacer with openings at the button positions. Pressing the membrane in one of these locations connects the corresponding row and column.

The diagram below shows the suggested HDL signal names and the suggested FPGA pin numbers used to connect the keypad. You may use different pins; for example, if some of your IO pins are damaged.



The photo below shows how the keypad is connected:

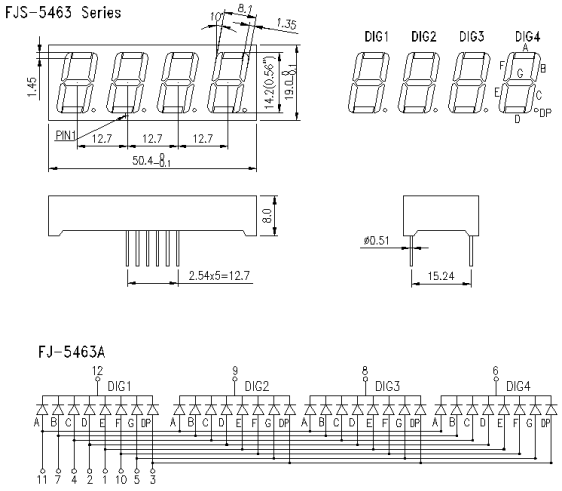


In this lab the keypad will be used as four pushbutton switches by configuring the rows as outputs with the top one set low and using the columns as inputs. These inputs will be pulled high by an internal (to the FPGA) pull-up resistor. Pressing a button along the top row then drives the corresponding column input low. For example, `col` will be `4'b1111` when no button is pushed and `4'b1110` buttons `A` is pressed.

4-digit, 7-segment Multiplexed LED Display

The datasheet[1] for the multiplexed, common-cathode, 4-digit, 7-segment LED display included in your parts kit is shown below:

[1]Inexpensive components – $0.40 in this case – often don't have great documentation.

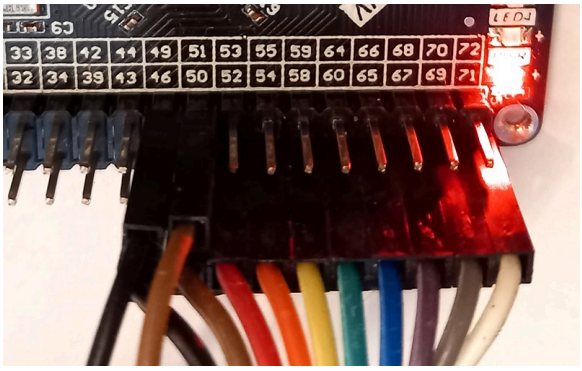


To turn on an LED segment the corresponding segment must be set high and the desired digit must be set low.

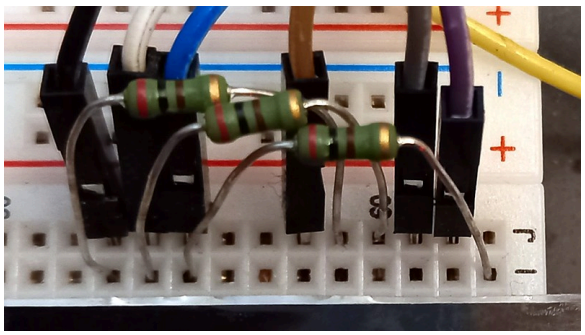
The table below shows one way to connect the FPGA to the display. Note the use of colour coding.

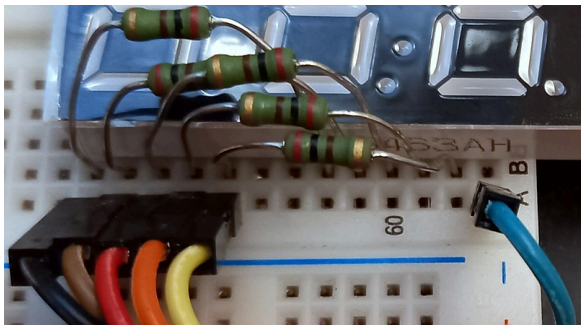
LED Pin	wire colour	segment	FPGA pin
1	black	`e`	46
2	brown	`d`	50
3	red	`dp`	52
4	orange	`c`	54
5	yellow	`g`	58
6	green	`en[0]`	60
7	blue	`b`	65
8	violet	`en[1]`	67
9	gray	`en[2]`	69
10	white	`f`	71
11	black	`a`	49
12	brown	`en[3]`	51

The photos below shows how the LED segments are connected to the FPGA pins:

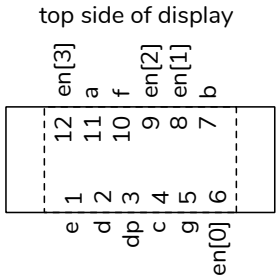


The photo below shows the connections to the LED on the breadboard.





The display pinout is:



The active-low digit-enable outputs are named **en[3]** through **en[0]** in order from left to right and the active-high segment-enable outputs are **a** through **g** (see the diagram above). For this lab only the rightmost digit is used so (**en[0]**) needs to be set low and the other digit enables high.

The segments are connected through 200 Ω resistors to avoid exceeding the FPGA's maximum current specification. The four common cathodes are connected directly.

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

As an example, the following truth table shows the values of the **col** input, the displayed LED digit and the values of the segment a through g for an ID of A00123456:

button	col	display	**a...g**
none	4'b1111		7'b000_0000 (7'h00)
1	4'b0111	4	7'b011_0011 (7'h33)
2	4'b1011	5	7'b101_1011 (7'h5b)
3	4'b1101	6	7'b101_1111 (7'h5f)

Internal Pull-Up Resistors

When you assign signals to pins you'll also need to configure internal pull-up resistors on the four **col** input pins. Open the Assignment Editor (**Assignments > Assignment Editor**). Double-click on «new». in the **To** column and enter the pin name (**col[0]**). Select **Weak Pull-Up Resistor** from the drop-down menu in the **Assignment Name** column. Select **On** from the drop-down menu in the **Value** column. Repeat for the other **col** inputs[2].

If you used the pin assignments above you should end up with the following:

out a	Location	PIN_49
out b	Location	PIN_65
out c	Location	PIN_54
in clk50	Location	PIN_23
* col*	Weak Pull-Up Resistor	On
in col[0]	Location	PIN_74
in col[1]	Location	PIN_76
in col[2]	Location	PIN_80
in col[3]	Location	PIN_84
out d	Location	PIN_50
out e	Location	PIN_46
out en[0]	Location	PIN_60
out en[1]	Location	PIN_67
out en[2]	Location	PIN_69
out en[3]	Location	PIN_51
out f	Location	PIN_71
out g	Location	PIN_58
out row[0]	Location	PIN_86
out row[1]	Location	PIN_98
out row[2]	Location	PIN_100
out row[3]	Location	PIN_103

[2]You can also use the "wild-card" assignment **col*** to select all the column signals.

Hints

1. You can use the Verilog concatenation operator (`{,}`) on the left-hand side of an assignment. For example:

```
...
   assign {a,b,c,d,e,f,g}
     = col == 4'b1011 ? 7'h5b :
...
```

2. To save you time, here are the active-high seven-segment values (`a` to `g`) for digits 0 to 9 in order from most- to least-significant bit[3]:

```
0  7'h7e
1  7'h30
2  7'h6d
3  7'h79
4  7'h33
5  7'h5b
6  7'h5f
7  7'h70
8  7'h7f
9  7'h7b
```

3. We'll be using the same display in later labs. Leave the LED connected if you have enough space on your breadboard.

4. You may want to use ModelSim to check your code for syntax errors – it compiles much faster.

5. Use SignalTap to check that the inputs and outputs have the values you expect. You will need to add a clock input signal. A 50 MHz clock signal is available on pin 23 as in the previous lab.

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.

[3] From Wikipedia.

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, for example:

Flow Status	Successful - Wed Sep 15 02:26:20 2021
Quartus Prime Version	20.1.1 Build 720 11/11/2020 SJ Lite Edition
Revision Name	lab1
Top-level Entity Name	lab1
Family	Cyclone IV E
Device	EP4CE6E22C8
Timing Models	Final
Total logic elements	762 / 6,272 (12 %)
Total registers	590
Total pins	20 / 92 (22 %)
Total virtual pins	0
Total memory bits	2,432 / 276,480 (< 1 %)
Embedded Multiplier 9-bit elements	0 / 30 (0 %)
Total PLLs	0 / 2 (0 %)

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 three keypad buttons from left to right. The rightmost LED should display the last three digits of *your* BCIT ID.

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