

Combinational Logic Design with Verilog

Introduction

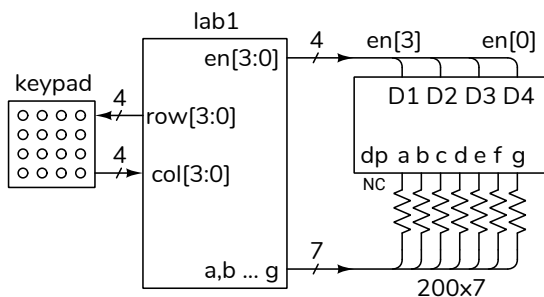
In this lab you will connect a 4x4 matrix keypad and a 4-digit, 7-segment LED display to the CPLD board. You will write a Verilog module to display one of the last three digits of your student number on the rightmost LED digit depending on which keypad button is pressed.

You will need the components used in the previous lab plus the following:

- solderless breadboard (from previous course)
- 11 M-F (“Dupont” or “Berg”) pin-header jumpers (from previous course)
- 4 × 4 matrix keypad
- seven (7) 200 Ω resistors
- an FJ5463AH 4-digit, 7-segment common-cathode LED display

Circuit Description

The following diagram shows how the keypad and LED are connected to the CPLD and the recommended signal names:



Your circuit should display one of 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, when button 2 is pressed the rightmost LED should display 5, and when button 3 is pressed the rightmost LED should display 6. Anything else should produce a blank display.

Component Connections

CPLD Board

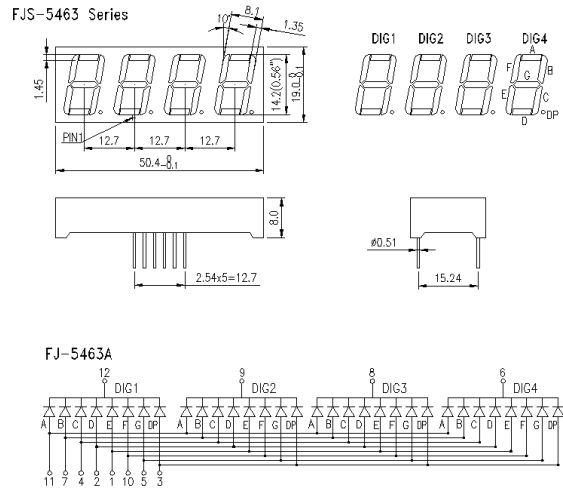
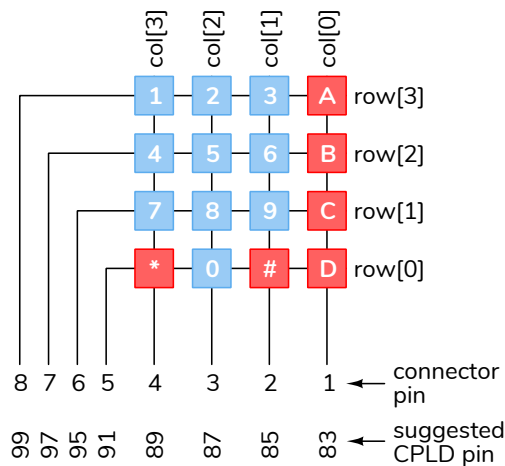
The CPLD has 100 pins. Seventy-six (76) of these are available on four 20-pin headers and will be used to connect components such as the LED display and keypad. The CPLD pin numbers are marked on the PCB. The remaining 6 header pins provide ground, 3.3 V, and 5 V supplies.

The CPLD’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.*

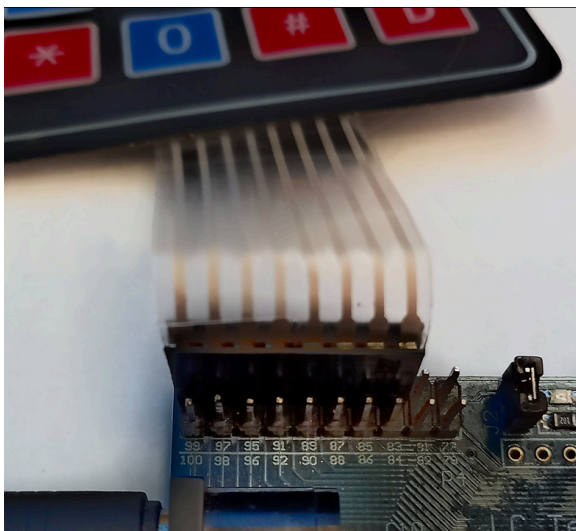
4x4 Matrix Membrane Keypad

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

The diagram below shows the suggested HDL signal names and the suggested CPLD 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:



To detect button presses on matrix keypads the rows are connected to outputs and the columns to inputs. In this lab the top row will be set low and the others high. The column inputs will be pulled high by an internal (to the CPLD) pull-up resistor. Pressing a button along the top row will connect the low logic output on the top row to one of the column inputs which will drive it low. For example, `col1` will be `4'b1111` when no button is pushed and will be `4'b1110` when button A is pressed.

4-digit, 7-segment Multiplexed LED Display

The datasheet¹ for the multiplexed, common-cathode, 4-digit, 7-segment LED display included in your parts kit is shown below:

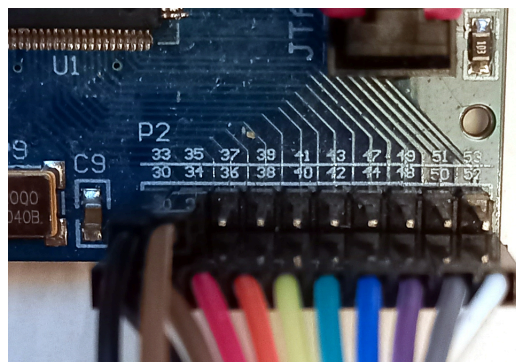
¹Inexpensive components – \$0.40 in this case – often don't come with great documentation.

To turn on an LED segment the pin for that segment (A-G) must be set high and the desired digit enable (DIG1 through DIG4) must be set low.

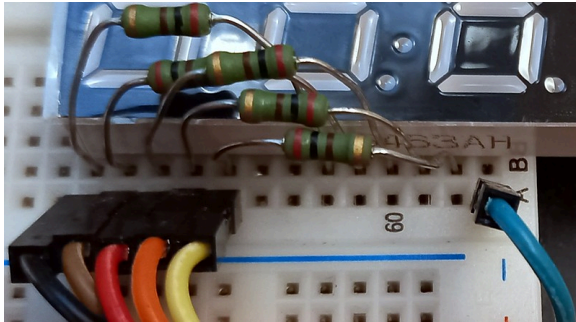
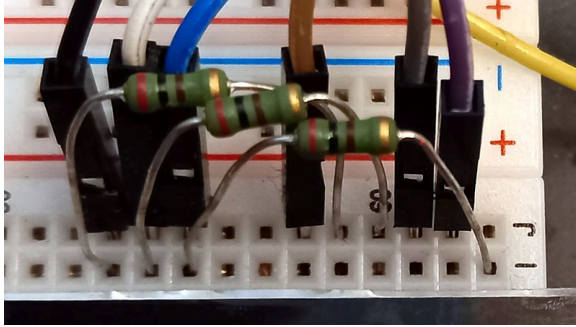
The table below shows one way to connect the CPLD to the display. Note the use of colour coding.

LED Pin	wire colour	segment	CPLD pin
1	black	e	30
2	brown	d	34
3	red	dp	36
4	orange	c	38
5	yellow	g	40
6	green	en[0]	42
7	blue	b	44
8	violet	en[1]	48
9	gray	en[2]	50
10	white	f	52
11	black	a	33
12	brown	en[3]	35

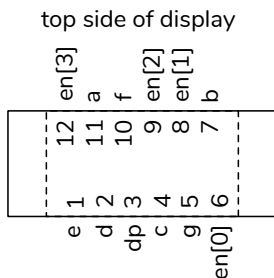
The photo below shows the connections to the CPLD pins:



and the photos below shows the connections to the LED on the breadboard.



The display pinout is:



The active-low² 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 CPLD’s maximum current specification. The four common cathodes are connected directly.

²“Active-low” means an input or output is at a low logic level when it’s true. A button would be considered “true” when pressed and an LED segment “true” when lit.

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 CPLD. Connect the CPLD 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 (bus) name (`col`). 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.

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

To	Assignment Name	Value
out a	Location	PIN_33
out b	Location	PIN_44
out c	Location	PIN_38
in clk50	Location	PIN_12
in col	Weak Pull-Up Resistor	On
in col[0]	Location	PIN_83
in col[1]	Location	PIN_85
in col[2]	Location	PIN_87
in col[3]	Location	PIN_89
out d	Location	PIN_34
out dp	Location	PIN_36
out e	Location	PIN_30
out en[0]	Location	PIN_42
out en[1]	Location	PIN_48
out en[2]	Location	PIN_50
out en[3]	Location	PIN_35
out f	Location	PIN_52
out g	Location	PIN_40
out row[0]	Location	PIN_91
out row[1]	Location	PIN_95
out row[2]	Location	PIN_97
out row[3]	Location	PIN_99

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 bit³:

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. Build and test a little bit at a time. For example, you could start by checking that you can detect keypad button presses by using a column input to turn the on-board LED on and off. Then you could check that you can display one digit of your ID.

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. Label all signals. Use multiplexers for conditional operators and Verilog expressions for other logic. Use Verilog syntax for numeric literals.

³From Wikipedia.

- A listing of your Verilog code.
- A screen capture of your compilation report, for example:

Flow Status	Successful - Wed Jan 12 21:59:04 2022
Quartus Prime Version	20.1.1 Build 720 11/11/2020 Patches 1.02i SJ Lite Edition
Revision Name	cpldtest
Top-level Entity Name	cpldtest
Family	MAX II
Device	EPM240T100C5
Timing Models	Final
Total logic elements	82 / 240 (34 %)
Total pins	22 / 80 (28 %)
Total virtual pins	0
UFM blocks	0 / 1 (0 %)

2. If you were not able to demonstrate your solution to the lab instructor during your lab period, submit a video showing the keypad and the LED display as you push the four three keypad buttons on the top row from left to right and the buttons **5**, **9** and **D**. 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.