
# LCD Calculator

**ENEE 245: Digital Circuits and Systems Laboratory**
**Lab 12**

## Objectives

The objectives of this laboratory are the following:

- Learn how to build an LCD driver and interact with the LCD output device.
- Build a calculator FSM.
- Interface with an external keyboard

This lab involves the design and implementation of a calculator using the LCD display on the FPGA board. A user will input values and operation codes via a keyboard interfaced with the FPGA, and the calculator will perform addition, subtraction, ANDing, and ORing of the input values, depending on the operation code.

## Keyboard Controller

On the course website is a tarsal containing a keyboard controller. There is one known bug to this keyboard controller—the bug and its solution are further down in this email. There are 5 files in the compressed directory:

| | |
|---|---|
| 1. clkdiv.v | Provides clk dividing to a 25MHz clock |
| 2. keyboard_ctrl.v | Keyboard Driver to gather inputs from PS2 |
| 3. keyboard_top.v | Converts Keyboard inputs to ASCII and displays on LCD |
| 4. lcd_driver.v | LCD Driver to display characters to a streaming output |
| 5. Nexys2_500General.ucf | UCF File with pin ports |

This code should work out of the box. To test, add all files to a new project and generate a programming file. When you first boot up the code, the LCD screen should be blank. Pressing any NUMPAD key should display the corresponding character on the LCD screen. Only the top row is active, but an easy change can activate any characters on the bottom row (This also allows students to display the answer on the bottom line if they want to).

**Note:** there is one known bug that occurs with Xilinx!!!

If you receive the error below:

> Pl_Uap_Flow1FitterRuleFastFeedbacks: bad index to sec_nodes array

Do the following to fix it:

> Project -> Design Goals & Strategies
>
> Change “Timing Goal” to “Timing Performance”

## Calculator

Use the code given to you to set up a connection between the numeric keypad and the LCD driver. Without making any changes to the attached code, you should be able to see whatever you type on the keypad printed out on the LCD screen.

Implement a simple calculator that accepts two input variables and performs one of four possible operations on them:

| | |
|---|---|
| + | add |
| - | subtract |
| / | AND |
| | anything else: OR |

Print the results to the LCD screen (either on the top row or below …).

**Note:** this will require some trial and error on the actual FPGA boards, as not all of this will work in simulation.

## Pre-Lab Preparation

1. Look through the LCD and keyboard code given to you and familiarize yourself with it, to minimize the time it will take for you to get it working in the lab.
2. Develop Verilog codes for the calculator.

## In-Lab Procedure

Ask the TA questions regarding any procedures about which you are uncertain.

1. Show your code and circuit to your TA.
2. Demonstrate the operation of your calculator.

## Post-Lab Report

Write up your code, schematics, and lab procedures.

1. Suggest additional features that can make your calculator more user-friendly.
2. What digital logic elements are generally used to represent the “states” in the finite state machine? Suggest an alternative way of implementing state information.