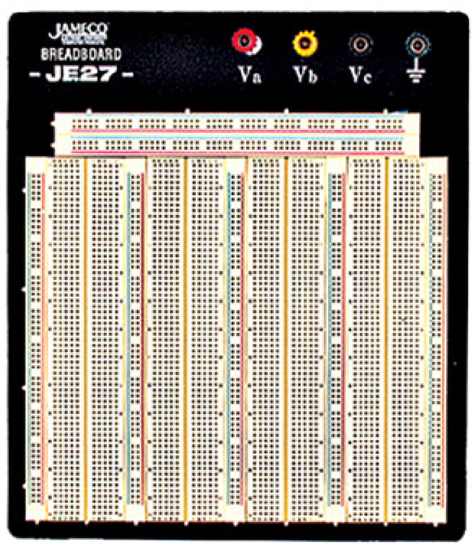
Lab 4 Logic Breadboard

CMPT 328 Computer Architecture



Scott Overholser

This lab was written for the Westminster College CMPT 328 Computer Architecture course Spring 2015 semester.

Last Modified: Fri Apr 3 19:53:23 2015 -0600

Table of Contents

1 Introduction

The LaunchPad will be used in this lab as a controller for circuits built on a breadboard. The program written for the LaunchPad implements some high-level programming constructs in assembler as we will see in *Section 6.4 Programming* in *Digital Design and Computer Architecture*.

The Lab 4 Breadboard project will be used for this lab.

2 Lab Objective

Work with combinational and sequential logic circuits on a breadboard using the LaunchPad as a controller.

3 Notes

The LaunchPad will be connected to the breadboard to act as a controller for the logic circuits you will build. The connections must be made carefully. The LaunchPad can be damaged. There is little risk of this happening but do be aware.

The breadboard is contains four independent modules arranged side by side. Each module has it's own power and ground. For the purpose of this lab they will be numbered left to right as module 1, module 2, module 3, and module 4. There is also power and ground at the top. This is where the power adapter will be connected. Power to each module individually will tap into the top rail.

- Lab 4 Breadboard project
- TM4C123G LaunchPad Evaluation Board User's Guide
- 74LS75 4-bit D Latch datasheet
- 74LS83 4-bit Full Binary Adder datasheet.
- 74LS04 Hex Inverter datasheet.
- 74LS08 Quad 2-Input AND Gate datasheet.
- 74LS32 Quad 2-Input OR Gate datasheet.

4 Data Bus

In this lab we will use one or more 4-bit data buses constructed with red LEDs to provide a visual indication of the value to which each bit is set. The bus is simply a collection of 4 common conductors to which other devices, including the LaunchPad will be connected. Figure 4.1 is a schematic that shows how to connect the LED to the bus serially through a 330 Ohm resistor to ground.

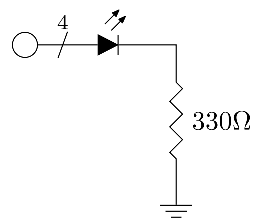


Figure 4.1: 4-Bit Data Bus LED Schematic.

The 330 Ohm resistor was selected specifically because it will allow sufficient current flow through the LED to make it's brightness satisfactory. A 1K Ohm resistor would have limited the current too much and resulted in a dimmer LED.

The "4" next to the LED symbol in the Figure 4.1 schematic indicates that this is a 4-bit bus and that there will be four sets of these LED/resistor circuits—one for each bit in the 4–bit bus.

Install the data bus on the far lower right with each data line positioned on rows as follows:

Each bit on the bus should be arranged on the breadboard in a systematic way so we can all have a standard layout. In module 4 furthest to the right, install each line of the 4-bit data bus according to the schematic as follows:

- Bit 0 positioned on row 60
- Bit 1 positioned on row 50
- Bit 2 positioned on row 40
- Bit 3 positioned on row 30

With your breadboard, construct your bus by installing the red LED and resistor so that the LED anode—the long leg—connects to the bus, the LED cathode connects to one lead of the resistor and the other lead of the resistor connects to ground.

The LED cathode will necessarily need to connect to the resistor on the adjacent row in the module. The resistor will then connect to ground. For example, Bit 3 LED anode will connect on row 30 while Bit 3 LED cathode will connect on row 29. The resistor will connect from row 29 to ground.

Using a short jumper wire, bridge the center gap in the breadboard section to add additional connection points for each bus data line.

Repeat this process and build a second 4-bit bus in module 1 using the same row configuration. The first bus will be referred to as bus 1 and the second, bus 2.

5 Enable Lines

In a similar manner to the 4-bit data bus construction, use green LEDs to create two 1-bit data lines EN1 and EN2. Each of these will be in the same module 4 as the data bus already built.

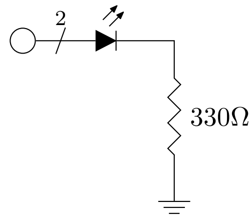


Figure 5.1: 2-Bit Enable Lines LED Schematic.

- EN1 (mode green) positioned on row 10.
- EN2 (mode blue) positioned on row 20.

6 Connect The LaunchPad

Using jumper wires of sufficient length, connect the LaunchPad pins to the breadboard circuit.

See the TM4C123G LaunchPad Evaluation Board User's Guide schematic on page twenty for jumper block maps.

- J2 pin 1 is ground. Connect this to the ground rail on the breadboard.
- J2 pins 8, 9, and 10 correspond to GPIO port A pins 4, 3, and 2 respectively. connect J2 pins 8, 9, and 10 to breadboard data bus bits 2, 1, and 0 respectively.
- J1 pin 8 corrsponds to GPIO port A pin 5. Connect J1 pin 8 to breadboard data bus bit 3.
- J1 pins 9 and 10 correspond to GPIO port A pins 6 and 7. Connect J1 pin 9 to EN1. Connect J1 pin 10 to EN2.

7 The Finished Product

Rather than a paint-by-numbers approach indicating row and column connection points for each bus, here is a photograph of the finished product.

In Figure 7.1, the LaunchPad is shown on the breadbboard. That is not a necessary arrangement. It was done just for the illustration.

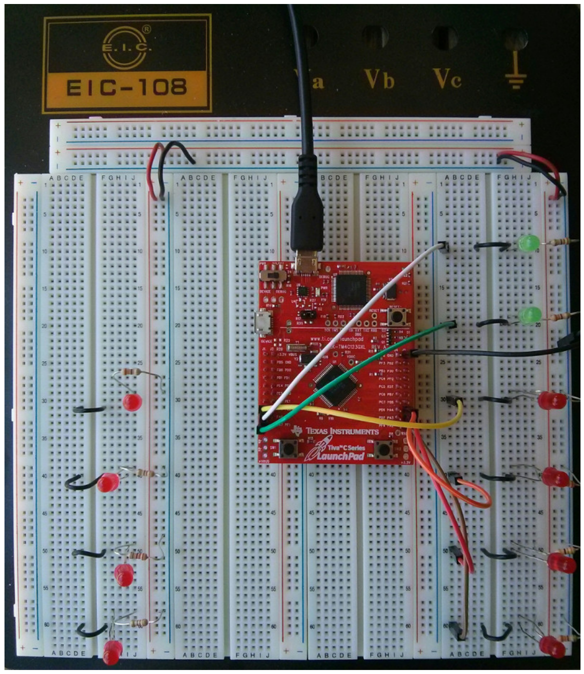


Figure 7.1: The breadboard with 2 4-bit buses, EN1, and EN2.

8 Experiments

Download the Lab 4 Breadboard project from Canvas. Extract the project to a location of your choice.

Connect your LaunchPad to your development environment. Start the IAR Systems Embedded Workbench and open the Lab 4 workspace. Compile the project then download and debug.

You should now be able to run the program on your LaunchPad without running the IAR Systems Embedded Workbench.

Keep the LaunchPad connected to your USB port for power.

8.1 4-bit D Latch

Refer to the 74LS75 4-bit D Latch datasheet. Find a 74LS75 and install it on the breadboard in module 2 at row 40 across the middle line in columns E–F.

- Connect D_n to the corresponding bits of bus 1 on the breadboard.
- Connect Q_n to the corresponding bits of bus 2 on the breadboard.
- Connect E_{0-1} and E_{2-3} to EN1 on the breadboard.
- Connect V_{CC} and GND on the 74LS75.

Load the bus 1 data onto the 74LS75 latch by pressing the SW2 button to activate EN1. You may need to long press SW2 to select EN1 (green) versus EN2 (blue.)

Experiment with loading data onto bus 1 by fast pressing SW1 and then activating EN1 by pressing SW2. Repeat with different values. Each time, you should see that the LEDs on bus 2 will update to match the LEDs on bus 1 when EN1 is activated.

8.2 4-bit Full Binary Adder

In this section you will continue to use the 4-bit D latch used in the previous section. Refer to the 74LS83 4-bit Full Binary Adder datasheet. Find a 74LS83 and install it on the breadboard in module 2 at row 50 across the middle line in colums E–F.

- Connect the A_n inputs of the adder to the Q_n outputs of the 74LS75 4-bit latch used in the previous section. This means you will disconnect the 74LS75 Q_n outputs from the LEDs in bus 2.
- Connect the B_n inputs of the adder to the corresponding bits of bus 1.
- Connect the sum outputs to the corresponding bits of the bus 2 LEDs.
- Connect the adder carry in C_0 to ground. It floats high and represents an incorrect carry in if this is not done.

Repeat the process of fast pressing SW1 on the LaunchPad to select a 4-bit value on bus 1. When satisfied, press SW2—when EN1 is selected—to latch the value. Remember the value you latched. Then fast press SW1 again to write a new value to bus 1. Inspect the LEDs on bus 2 and confirm that the addition is correct.

Now disconnect C_0 so the carry in is active. Repeat the previous steps and confirm that the carry is being added to the result on bus 2.

8.3 Theorems of Boolean Algebra

For this section, start the IAR Systems Embedded Workbench and open the lab 4 program workspace.

On line 69 of `main.s`, `count_limit` may be changed to 4. This will restrict the data output to bus 1 to a 2-bit value 0–3.

Consider an alternate approach. Rather than changing `count_limit`, leave it as is and utilize EN2 for the second 2-bits of the 4-bit latch. This will permit you to independently latch the 2 most significant bits and the 2 least significant bits on bus 1 using EN1 and EN2.

Using the alternate approach will enable you to build and operate both logic circuits at the same time.

If you choose the first approach, recompile the program, then download and debug. This is sufficient to flash the new program to the LaunchPad which will now operate without need of the IDE.

Reference the datasheets for the 74LS04 Hex Inverter, 74LS08 Quad 2-Input AND Gate, and 74LS32 Quad 2-Input OR Gate.

8.3.1 DeMorgan's Theorem

Using the appropriate components, build the circuits shown in Figure 8.1 and Figure 8.2.

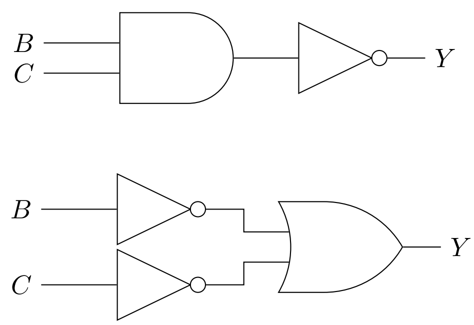


Figure 8.1: NAND Equivalent Circuits.

$$\overline{B\,C} = \overline{B} + \overline{C}$$

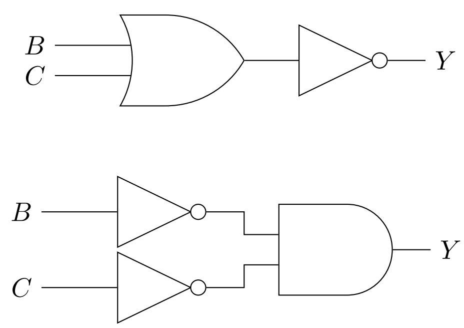


Figure 8.2: NOR Equivalent Circuits.

$$\overline{B + C} = \overline{B}\,\overline{C}$$

For each circuit in turn, connect bus 1 bit 0 to input B and bus 1 bit 1 to input C. Connect output Y to bus 2 bit 0.

Using the LaunchPad as a controller, cycle through the 2-bit range and verify the output. You should be making notes on this output for your lab report.

8.3.2 Covering Theorem

Build a simple logic gate circuit that demonstrates the covering theorem and it's dual shown in Figure 8.3 and Figure 8.4.

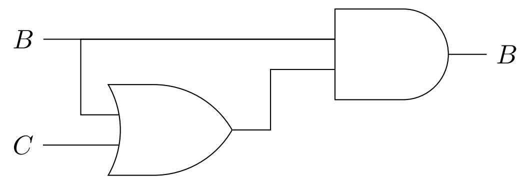


Figure 8.3: Covering Theorem Circuit.

$$B \bullet (B + C) = B$$

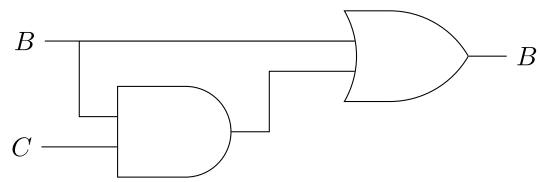


Figure 8.4: Covering Dual Theorem Circuit.

$$B + (B \bullet C) = B$$

For each circuit in turn, connect bus 1 bit 0 to input B and bus 1 bit 1 to input C. Connect output Y to bus 2 bit 0.

Using the LaunchPad as a controller, cycle through the 2-bit range and verify the output. You should be making notes on this output for your lab report.

9 Lab Report

Write a lab report summarizing your observations.