UNIVERSITY of GUELPH

Department of Computing and Information Science

Digital Systems
(CIS*3120)

Laboratory Exercise 1

Purpose

This laboratory exercise introduces you to the process of building digital-logic circuits using ***LogicWorks***. Make sure that you understand all of the steps detailed below and ask the Teaching Assistant (TA) if you have any questions.

The Circuit

In this first lab we will be building and simulating a 2-input multiplexor (or mux). A 2-input multiplexor is a simple component of digital circuitry that passes (i.e., multiplexes) one of two input signals, IN0 or IN1, onto the output line, OUT. A third input signal, SEL, selects whether IN0 or IN1 is passed.

Specifically, the mux takes three input signals: IN0, IN1, and SEL. The mux generates one output signal: OUT. OUT has the value of IN0 when SEL is low and the value of IN1 when SEL is high.

The mux's functionality can be described by the following Boolean equation:

$$\text{OUT} = (\overline{SEL} \bullet IN0) + (SEL \bullet IN1) + (IN0 \bullet IN1)$$

Though this is not the simplest way to implement a mux, it will be illustrative for our tour of LogicWorks. Later on in the course, you will learn how to derive equations.

We are now ready to build the actual circuitry of the above equation. The following section will guide you through the design of this simple equation using LogicWorks.

Your Assignment

- Startup LogicWorks. Five windows should have appeared. The **LogicWorks** window is where all of the application's main functions can be selected, such as opening a new schematic. The **Palette** contains shortcuts for many of the common circuit editing commands, such as drawing a signal (i.e. a wire). The **Parts** window contains libraries of pre-built components that come with the software package. Later on, you will add your own

parts as well as your own libraries. The **Design** window is where the actual circuit design is built. The final window is the **Timing** window, where you can see the results of simulations of your design.

- Your first step should be to create your own library. Go to the **Parts** window and press the right mouse button. Click on the **New** option, name your new library "3120" by replacing the "*" in the **File Name** field of the popped up window. Make sure that you have a directory on your local disk where you can store all of your designs. A library is automatically opened when created, from now on, however, you will need to open your library each time you start up LogicWorks. You can list all the libraries by clicking the left button to the right of the current library name (i.e. 3120.clf). You can select a library from the list by clicking on it.

- We can now move to creating the actual design. The first thing you should do is add the input pins to your design. To do this, switch to the **Connect** library and double-click on **Port In**. When you move to the **Design** window, the symbol for a port should appear. If the symbol is not facing the way you would like it to , you can change its orientation by using the arrow keys before placing the symbol on the schematic. To place the symbol on the schematic, click the left button. Place three ports on the left side of the window. When done, you can press the spacebar to indicate to the program that you are done placing ports.

- You should assign your ports to our three inputs, SEL, IN0, and IN1. To name an input port, simply select and click the right mouse button. Choose the **Name** option. Name your ports SEL, IN0, and IN1 respectively. Also be sure to click the **Visible** option, which indicates that the names should be displayed on the schematic.

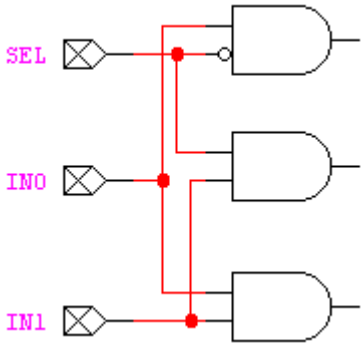
- Now save your design. Click on the **Design** window and, from the **File** menu, choose **Save Design As**. Call your design "tutorial". Your design should now look like this:

- We can now add the necessary gates. From the equation we can tell that we will first need to put three AND gates on our design. To do this, select the **Primgate** library. Select the AND-2 gate by double clicking. The "-2" indicates a 2-input AND gate.

  Place two of the gates towards the bottom of your design. Again, you may have to change the orientation of the gate using the arrow keys before placing it. Use the space bar to indicate that you are done placing library components. We will use a different AND gate for

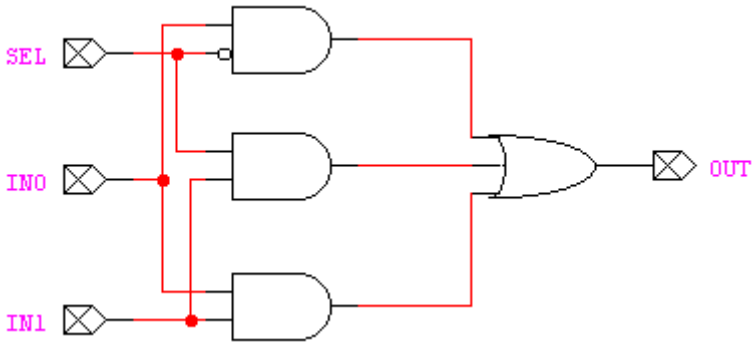
the top one. Notice that one of its inputs, SEL, is inverted. Instead of adding an inverter, we can simply use the **AND-2 (1-INV)** gate which indicates a 2-input AND gate with one inverted input. Place this gate at the top of your design.

- We can now wire the gates to the appropriate inputs. To add wires, select the **Draw Signal** option from the **Edit** menu of the **Schematic** window. As you go over the Schematic window, your cursor should take the shape of a cross. Simply tie together the inputs to the appropriate gates by clicking the left button wherever you wish to make a connection. Double-click to terminate the line. Be sure to tie SEL to the input of the first AND gate with the little bubble in front it; this means that the input will be inverted. Note that though two wires may cross on your screen, they only connect if a bubble appears. Your circuit should now look like the following:



- The only gate that we need to add is the OR gate. We could add two 2-input OR gates. But a simpler way is to add one 3-input OR gate, which can be found in the **Primgate** library. Go ahead and add it to your schematic, you can also tie it to its inputs.

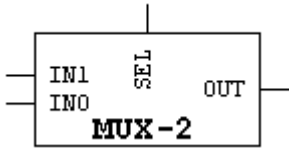
- The only remaining part of the design that needs to be added is the output port. To do this, use the same procedure as the input ports but use the **Port Out** component instead. Place the port on the schematic at the output of the OR gate and name it OUT. Your completed design should look like the following:



- Our circuit is now fully functional. It would not be very practical, however, to keep having to redraw the circuitry when we need a MUX-2. Not only that, but the circuitry takes up a lot of space on our screen. We can use LogicWorks' **DevEditor** to create a symbol of our circuit. To do this, keep your schematic window open, then go to the **Tools** menu of the **LogicWorks** window and select **DevEditor**. A new window should have appeared.

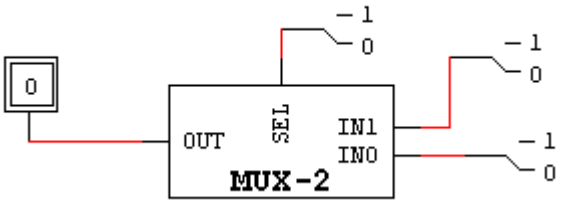
- From the **Options** menu, select the **Subcircuit / Part Type** option. Select the **Create a subcircuit and store the subcircuit with it** option. Choose the tutorial circuit that we have been working on. Click on **Internal** then click **Done**.

- From the **Options** menu, select the **Autocreate Symbol** option. Type in "MUX-2" as the part name. Now click on the **Extract Pin List** option. You should see the names of all your ports appear. LogicWorks assumes that input pins are located on the left or top side of a design, while output pins are located on the right or bottom side of a design. Note that the following notation IN0..INn indicates all the inputs IN0, IN1 through INn. Move the SEL pin to the top by deleting it from the list of **Left Pin Names** and adding it to the list of **Top Pin Names**. Once you have done this, press the **Generate** button. Our symbol should now appear on your screen like this:



- Now that you have created a multiplexor circuit and created a symbol for it, let's save the multiplexor for future use in our 3120 library. Inside **DevEditor**, click on **File** and choose **SavePartAs**. Choose the 3120 library from the list of available libraries.

- You have now completed the design component of your lab. What remains is to simulate your design in order to confirm that it works correctly. Open a new design by clicking on **File** in the **LogicWorks** window and selecting **New Design**. Place your 2-input multiplexor in your design. To simulate our multiplexor, we need a way to drive input signals and observe output signals. LogicWorks uses a **Binary Switch** to drive a signal and a **Binary Probe** to observe a signal. Both of these can be found in the **Primio** library. Attach three switches to your inputs and one binary probe to your output. Your design should look like the following:



- Now test your design by switching the inputs and watching the output. To 'switch' a switch, just click on it. Is your design correct?

  NOTE: To drag a switch in the **Design** window, hold down the Shift key and use the left mouse button.

- A binary probe only displays the circuit's output at one instance in time. To get a more detailed understanding of your circuit's behavior, you can simulate your circuit for some time and display the input and output waveforms in the **Timing** window. Start by assigning names to your input and output signals. Click the right mouse button on the **IN0** signal. The signal will turn yellow and a menu will appear. Select the **Name** option. Name your signal "IN-0" and check **Visible**. The word **IN0** now appears on the left axis of the **Timing** window. Repeat for the **IN1, SEL**, and **OUT** signals.

  NOTE: If the signals do not automatically appear in the **Timing** window, pull down the **Timing** menu in your design window and make sure that **Add Automatically** is enabled.

- To start the simulation, click on **Tools** in the **LogicWorks** window and choose **Simulator**. Now test your circuit by clicking on **Run** and switching your inputs to all eight possible settings. If the waveforms are too small, click on the **Timing** menu in the **Timing** window and choose **Enlarge**. Show the final circuit and timings to the TA.

- You are now done with the first lab. You understand LogicWorks and are ready to design and simulate digital circuits. Be sure to see your lab assistant during office hours if you have any questions about LogicWorks.