

LogicWorks 4 Tutorial

Updated – September 7, 2004 Author: Jose A. Garcia Rivera

Got <http://academic.uprm.edu/rpalomera> and download Logic Works tutorial.

Basic Digital-Gate devices

A logic gate is an elementary building block of a digital circuit. Most logic gates have two inputs and one output. At any given moment, every terminal is in one of the two binary conditions low (0) or high (1), represented by different voltage levels. The logic state of a terminal can, and generally does, change often, as the circuit processes data. In most logic gates, the low state is approximately zero volts (0 V), while the high state is approximately five volts positive (+5 V).

There are seven basic logic gates: AND, OR, XOR, NOT, NAND, NOR, and XNOR.

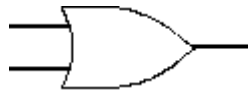
The AND gate is so named because, if 0 is called "false" and 1 is called "true," the gate acts in the same way as the logical "and" operator. The following illustration and table show the circuit symbol and logic combinations for an AND gate. (In the symbol, the input terminals are at left and the output terminal is at right.) The output is "true" when both inputs are "true." Otherwise, the output is "false."



AND gate

| Input 1 | Input 2 | Output |
|---------|---------|--------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

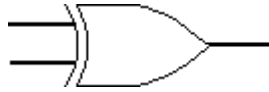
The OR gate gets its name from the fact that it behaves after the fashion of the logical inclusive "or." The output is "true" if either or both of the inputs are "true." If both inputs are "false," then the output is "false."



OR gate

| Input 1 | Input 2 | Output |
|---------|---------|--------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

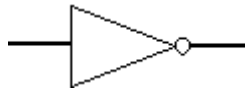
The XOR (exclusive-OR) gate acts in the same way as the logical "either/or." The output is "true" if either, but not both, of the inputs are "true." The output is "false" if both inputs are "false" or if both inputs are "true." Another way of looking at this circuit is to observe that the output is 1 if the inputs are different, but 0 if the inputs are the same.



XOR gate

| Input 1 | Input 2 | Output |
|---------|---------|--------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

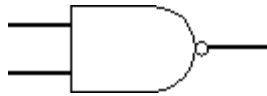
A logical inverter, sometimes called a NOT gate to differentiate it from other types of electronic inverter devices, has only one input. It reverses the logic state.



Inverter or NOT gate

| Input | Output |
|-------|--------|
| 1 | 0 |
| 0 | 1 |

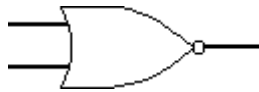
The NAND gate operates as an AND gate followed by a NOT gate. It acts in the manner of the logical operation "and" followed by negation. The output is "false" if both inputs are "true." Otherwise, the output is "true."



NAND gate

| Input 1 | Input 2 | Output |
|---------|---------|--------|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

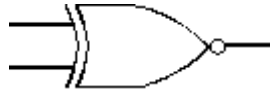
The NOR gate is a combination OR gate followed by an inverter. Its output is "true" if both inputs are "false." Otherwise, the output is "false."



NOR gate

| Input 1 | Input 2 | Output |
|---------|---------|--------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

The XNOR (exclusive-NOR) gate is a combination XOR gate followed by an inverter. Its output is "true" if the inputs are the same, and "false" if the inputs are different.



XNOR gate

| Input 1 | Input 2 | Output |
|---------|---------|--------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Using combinations of logic gates, complex operations can be performed. In theory, there is no limit to the number of gates that can be arrayed together in a single device. But in practice, there is a limit to the number of gates that can be packed into a given physical space. Arrays of logic gates are found in digital integrated circuits (ICs). As IC technology advances, the required physical volume for each individual logic gate decreases and digital devices of the same or smaller size become capable of performing ever-more-complicated operations at ever-increasing speeds.

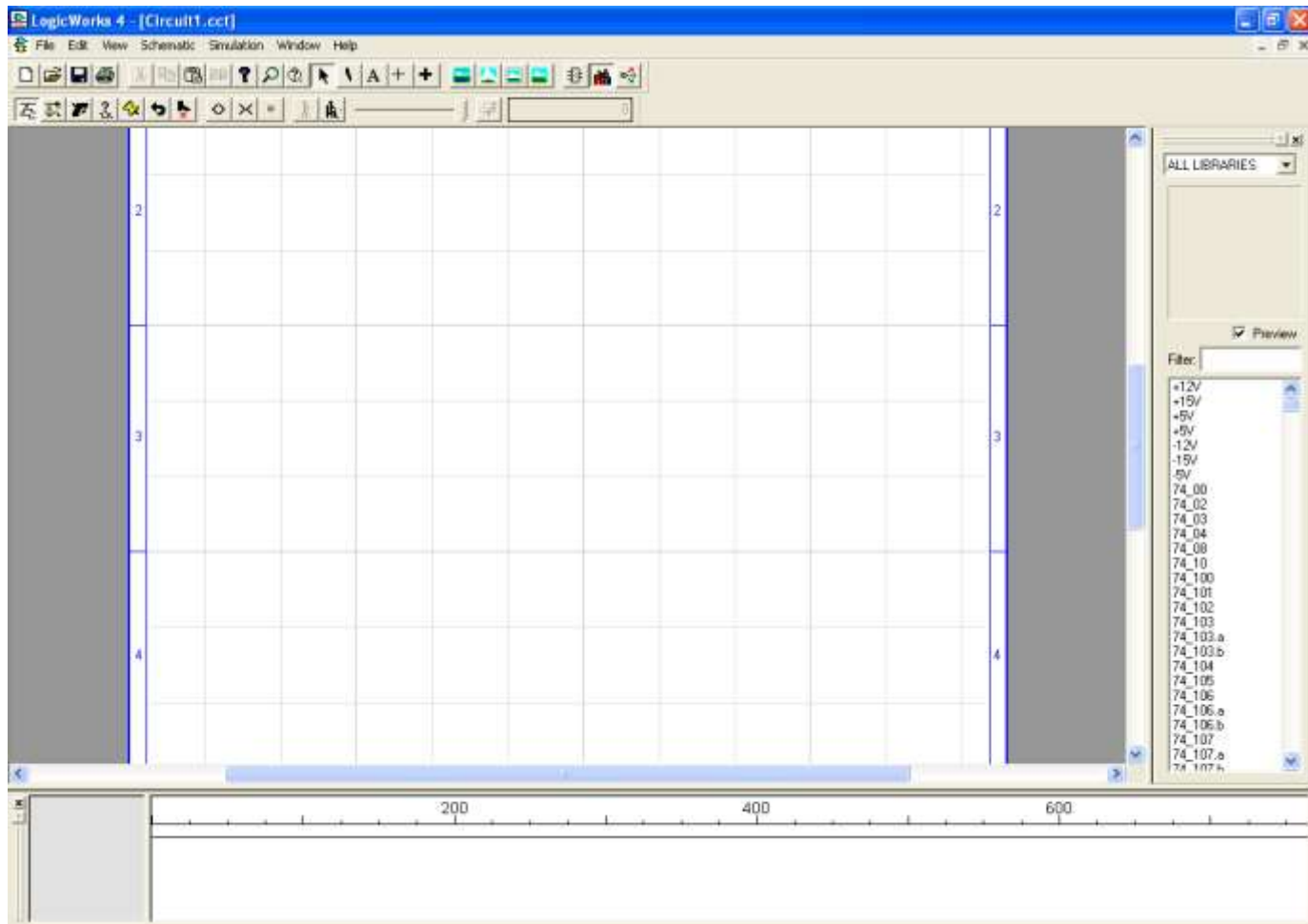
The Five-Minute Schematic and Simulation

In this section, we're going to show how quickly you can create and test a circuit using LogicWorks.

Starting LogicWorks

- Start the LogicWorks program by clicking on:
- Start->Programs->LogicWorks 4->LogicWorks 4

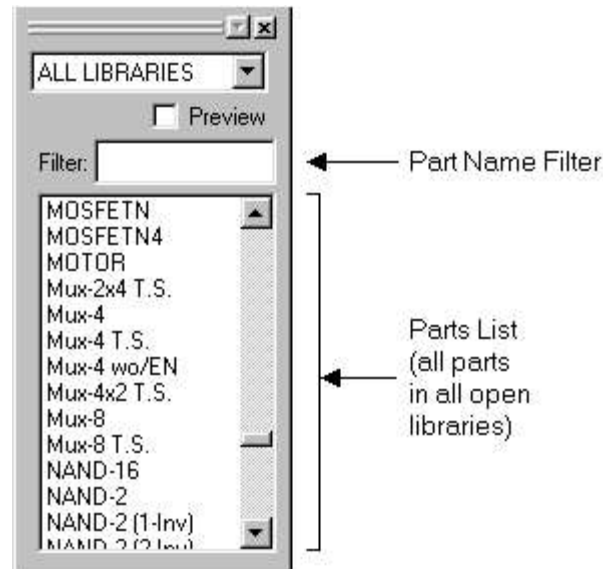
Once the program has started, you will be looking at a screen like this.



The "Design" window is your viewport onto the circuit diagram, which you will manipulate using various drawing tools. The smaller "Timing" window will be used by the program to display a timing diagram of the signals in your circuit. Either of these windows can be moved or resized by the usual methods, to suit your needs.

Placing a Device

The parts palette shows a merged list of all parts in all open libraries. Libraries can be opened and closed manually using the Parts popup menu's Open and Close commands, or any collection of libraries can be opened automatically at startup by placing them in the "Libs" directory.



- Locate the **74_08** part in the parts list and double-click on it.
- Move the cursor back into the circuit window. The cursor on the screen will now be replaced by a moving image of the selected symbol, in this case a Quad AND circuit.

The numbered devices in this library are generic 7400-series types. The labeling and simulation characteristics can be adjusted to match the various 7400 families on the market.

- Position the image somewhere near the centre of the circuit window and click the mouse button. A permanent image of the device will now stay behind in that location and the image will continue to follow your movements.

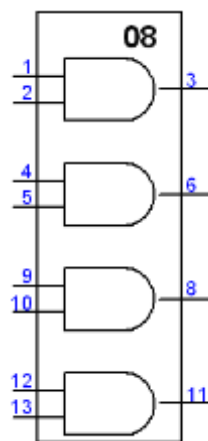


Figure 1 Four-Input AND

More devices of the same type could be created at this point, but in this example we wish to select another symbol.

- Press the spacebar to return to Point mode. Notice that you can click and drag the device that you placed to any desired new position.
- Move again to the Parts Palette, open the DemoLib.clf library and double-click on Binary Switch. Once you move outside of the Parts Palette, the cursor will immediately change to match the new symbol.

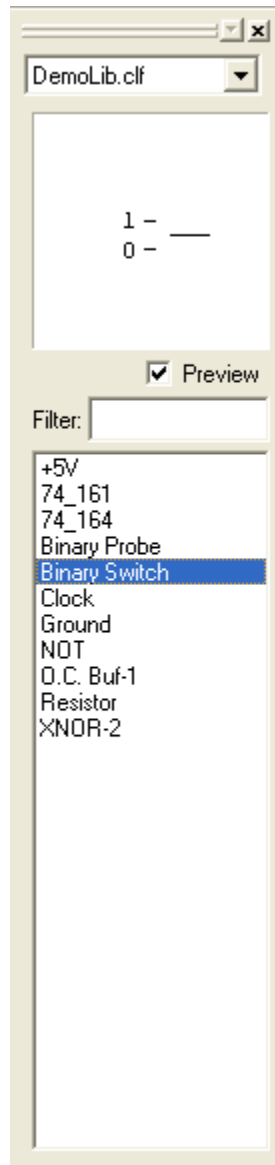


Figure 2 Parts and Library Explorer

The Binary Switch, and the devices in the Simulation Gates.clf, Simulation IO.clf, and Simulation Logic.clf libraries, are called "primitive" types because they have built-in simulation models in LogicWorks. Other devices, such as those in the 7400devs.clf library are called "subcircuit" types because their simulation models

are made up of primitives. If LogicWorks is being used only for schematic entry, it is also possible to make symbols with no simulation function.

- Place 8 of these binary switches in the design windows.

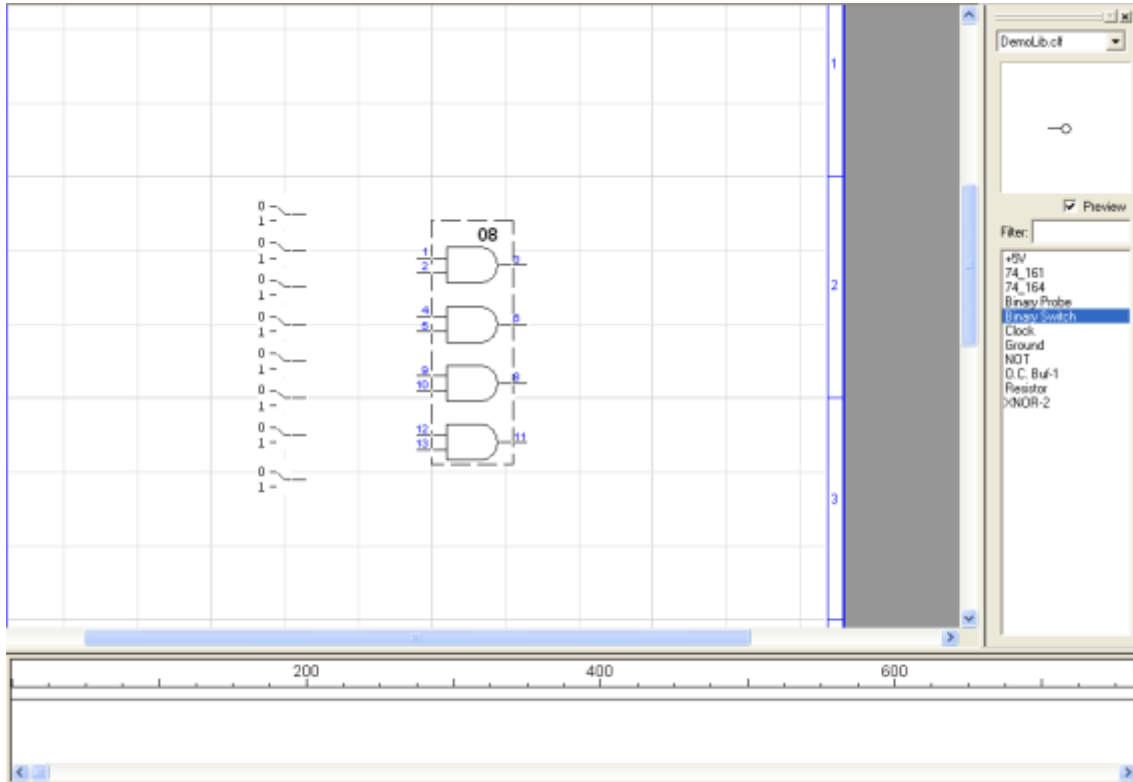


Figure 3 QUAD-Input AND

In order to move a switch, you must first select it by holding the [SHIFT] key while clicking on it. This is necessary because the switch has a special response to a normal mouse click.

- Press the spacebar to return to Point mode.

Whenever you place devices or signal lines so that they touch, you will notice that the signal lines flash briefly. This indicates that a logical connection has been made. You do not need to explicitly request a connection.

Moving a Device

- Point at the AND gate and click and drag to the right. While you hold the mouse button you can drag the device to any desired new position. Note that any signal lines attached to the device are adjusted continuously to maintain connection.

Drawing Signal Connections

- Attach a connection to the output of the gate by positioning the pointer near the endpoint of the pin and dragging away to the upper left.

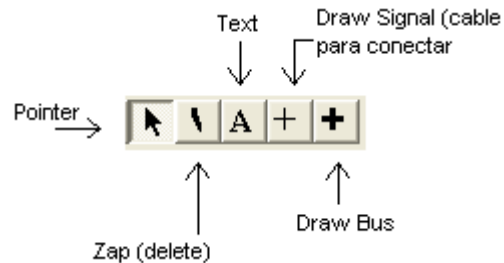


Figure 4 Design tools menu

- Notice that two lines at right angles will follow your mouse movements to connect the starting and ending points.

While moving the mouse, try pressing the [CTRL] and/or [TAB] keys and note the different line-routing methods available. Click mouse once to anchor the signal line.

Making a connection using a wire

- Connect the binary switches to the inputs of the AND gates as shown in figure 5.

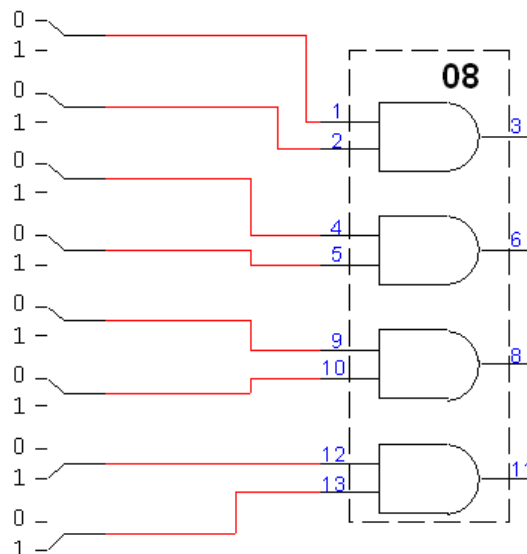


Figure 5 Connection of binary switches to AND gates

- To verify the output the **Binary probe** part is used. It is found in the DemoLib.cif

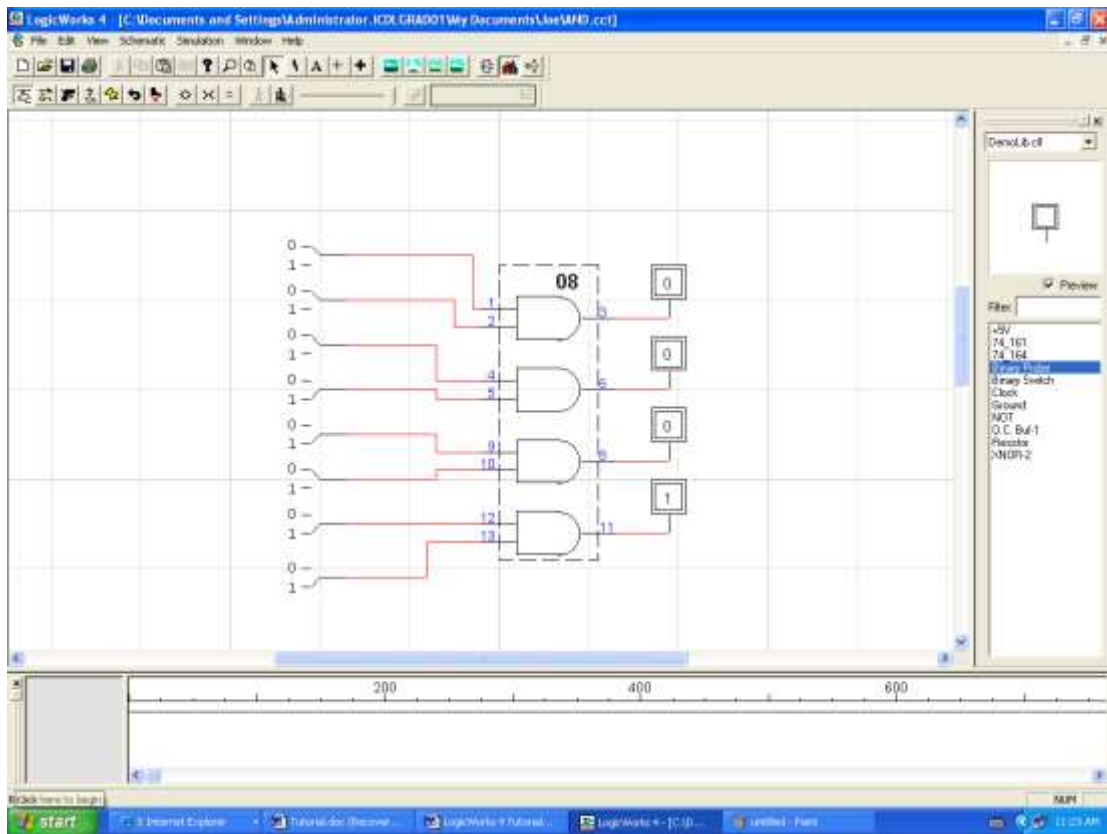


Figure 6 Probes added to verify output

Example 2 Use of tri-state gate

The tri-state gate is shown in figure 7.

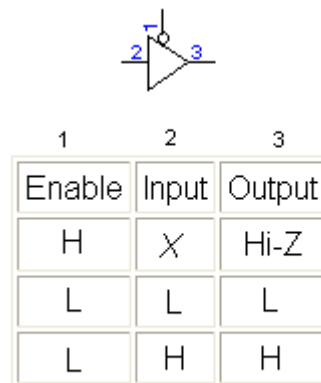


Figure 7 Tri-state and truth table

- Place the tri-state gate in the design window. Search in the explorer parts for the **74_425.a**
- In the same way with the AND gate, add an input switch and output probe as shown in figure 8.

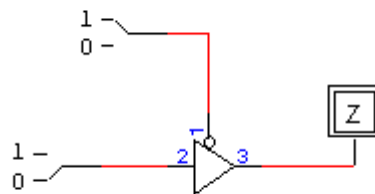


Figure 8 Tri-state with input switch

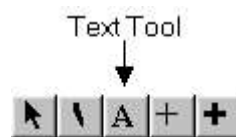
Adding a clock to the tri-state gate

Clock Generator Device

- Select a Clock device from the demolib.clf library and place it on the diagram as shown in figure 9.
- Press the spacebar to return to Point mode.

Naming a Signal

- Click on the text icon in the Tool Palette. The cursor will then change to a pencil shape, which will be used to select the item we want to name.



The text cursor is used to name devices and signals, to apply pin numbers to device pins, and to add free text notations to the diagram.

- Position the tip of the pencil anywhere along the length of the line running from the clock device, and press *and hold* the mouse button. The cursor will change to an I-beam shape.

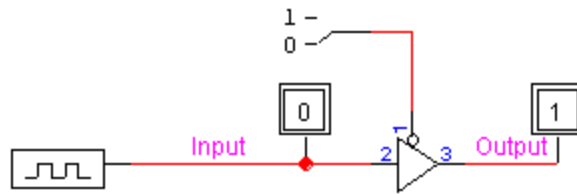


Figure 9 Tri-state with clock input

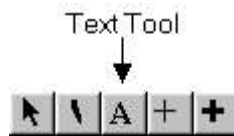
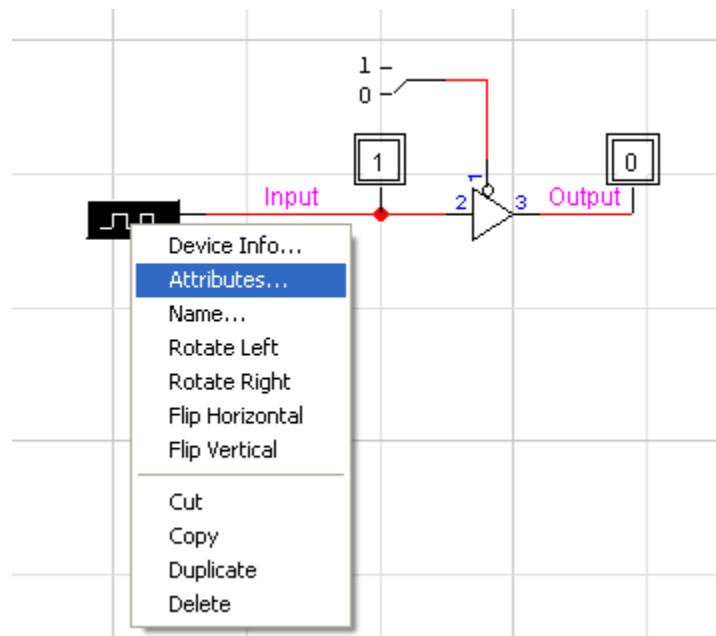
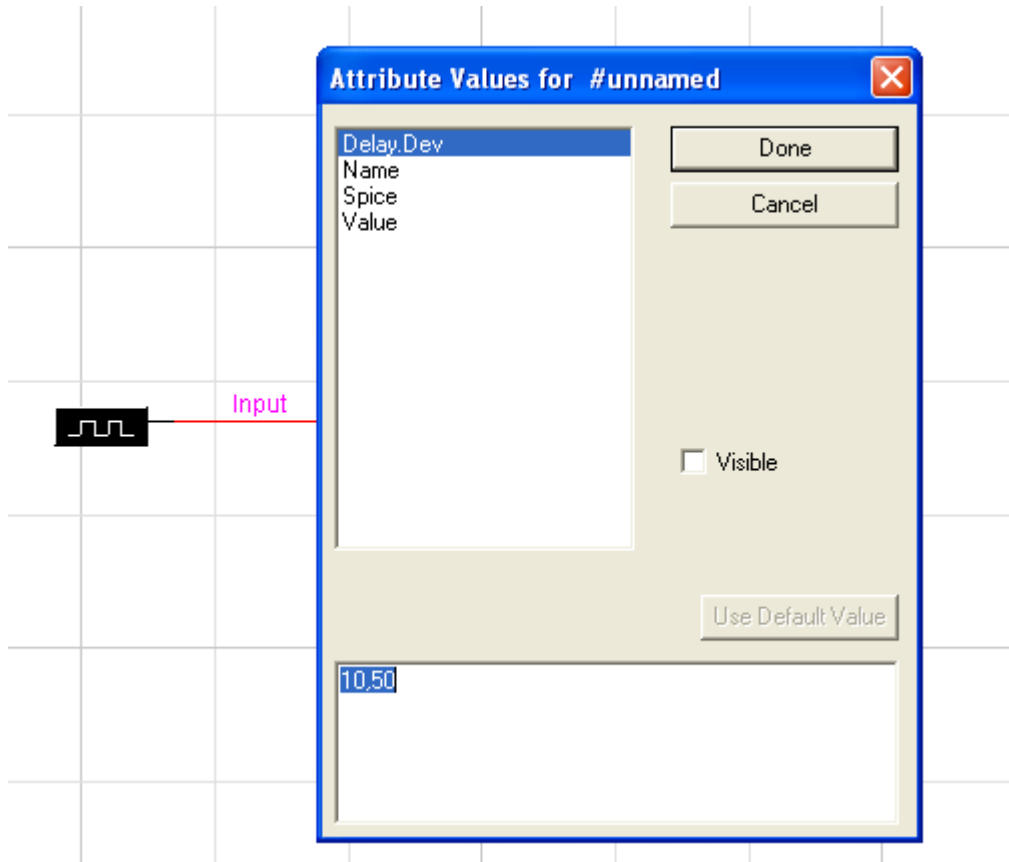


Figure 10 Text tool to label a wire

- Type the name "CLK" on the keyboard, then press the [ENTER] key or click the mouse button once.
- Return to Point mode by clicking the arrow icon in the Tool Palette. Note that the name can be dragged to any desired position.
- Click once on the Binary Switch to change it to the logical 1 state.
- Change the clock attributes





The Timing Window

- To run a simulation, the simulation menu provides the tools to aid in the simulation process as shown in figure 11.

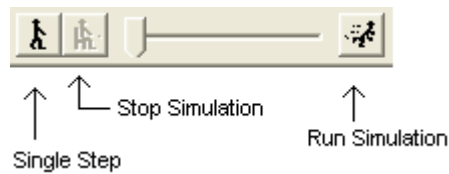


Figure 11 Simulation tools icons

You will immediately see the Timing window come to life with the displayed values on the CLK line. By default, any named signal is shown automatically in the Timing window.

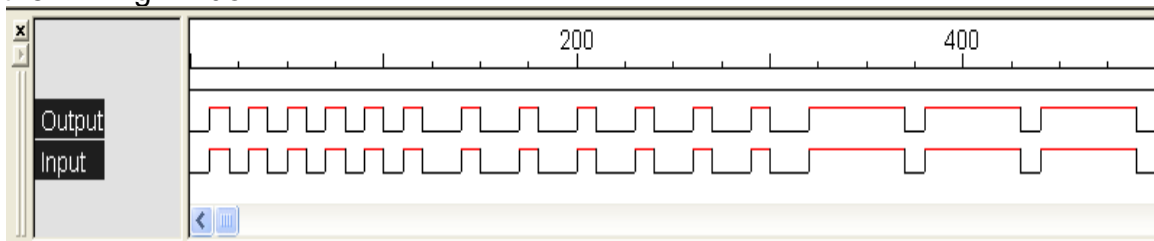


Figure 12 Timing diagram

Example 3

- Use the design window to simulate the schematic in figure 13

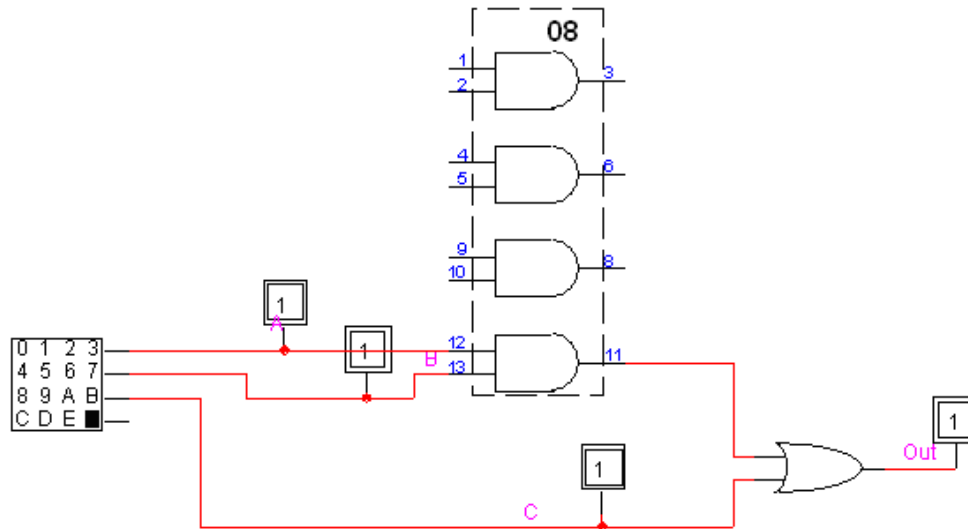
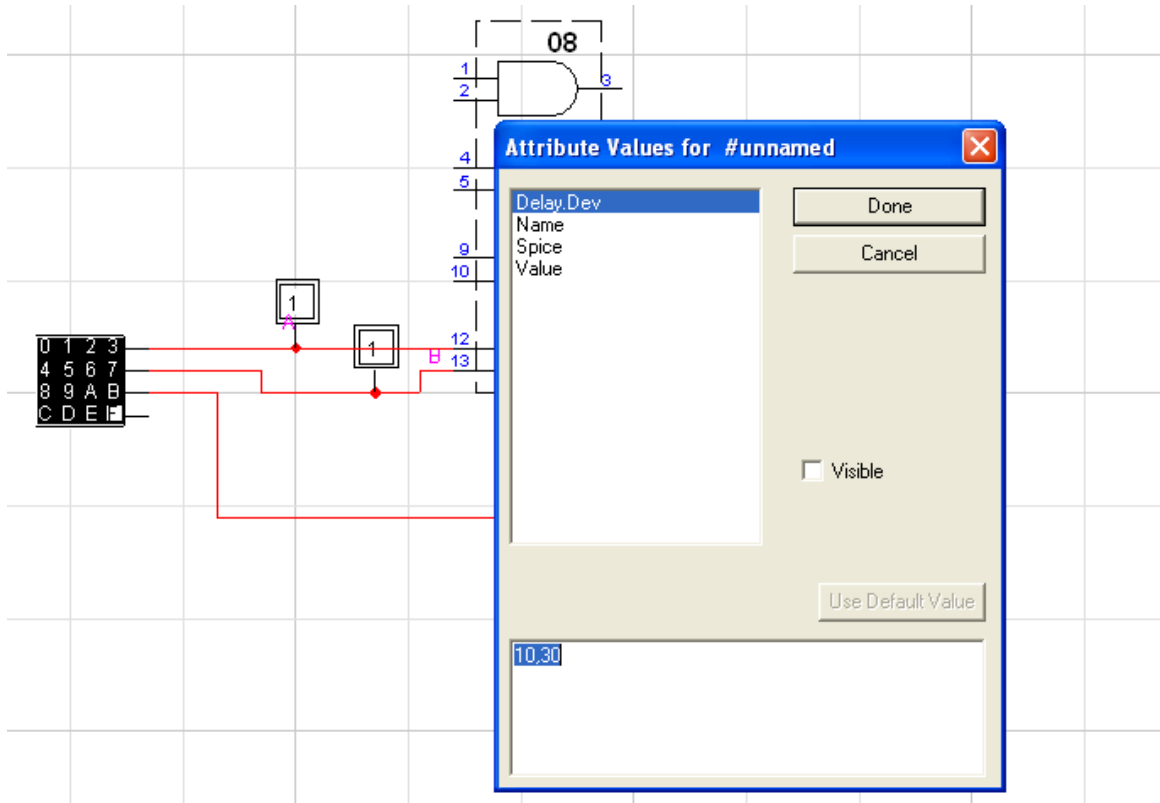
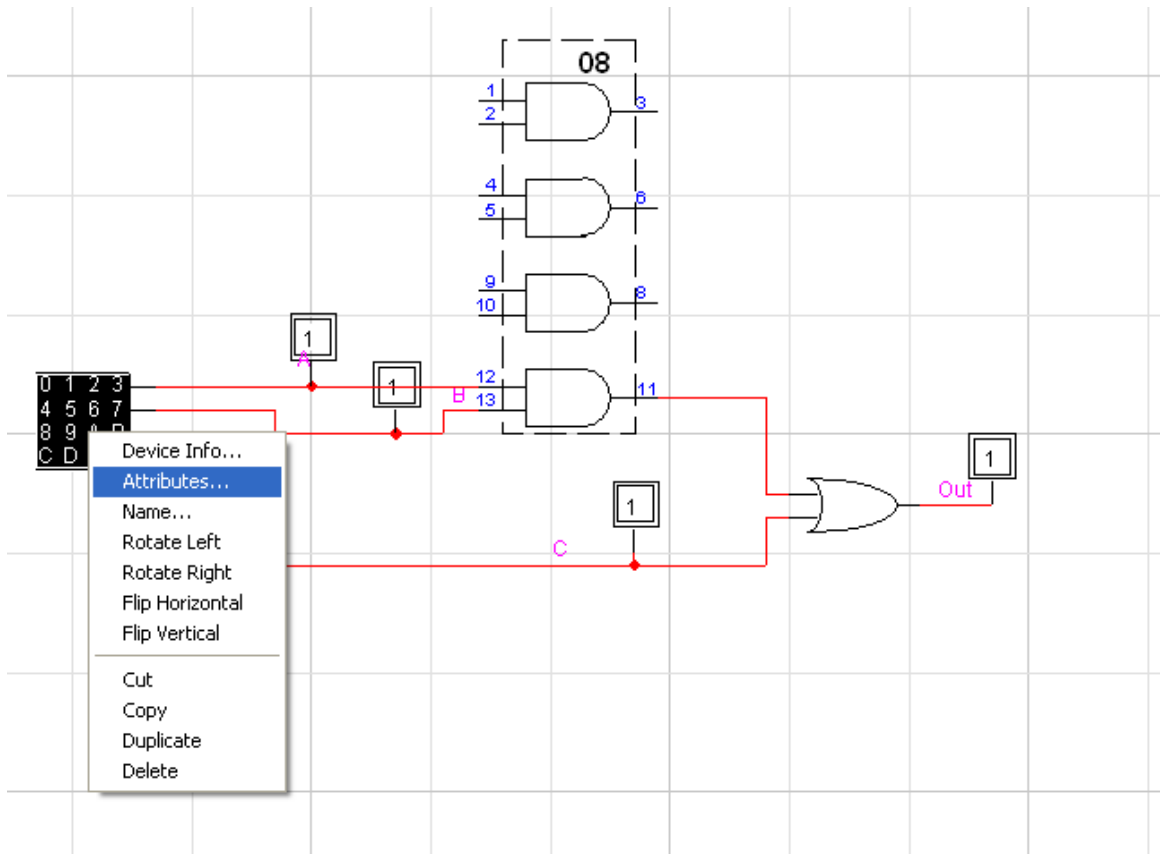


Figure 13 Example 3 schematic

- The input clock on this example is created using the Hex-Keyboard in the explorer library.
- Change the attributes of the hex-keyboard



- The timing diagram results in figure 14.

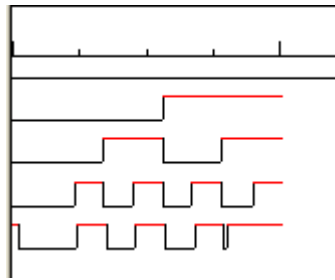



Figure 14 Timing diagram of figure 13

Simulation Controls


Click on the <> and >> buttons and observe that they affect the time scale of the Timing window.



Display resolution can be adjusted from 4 pixels/time unit to 100 time units/pixel. The interpretation of a "time unit" is arbitrary, but it is convenient to think of it as a nanosecond.

- Select the Timing Window item in the View menu. You will notice that the Timing window disappears and the current time indicator in the Simulator palette advances much more quickly.
- Select the Timing Window item again to re-enable the display.
- Click on the Reset () button and notice that the simulation restarts at time 0.

Adjust the speed slider control in the Simulator toolbar and notice that simulation slows.

- Click repeatedly on the Step () button and observe that the simulation proceeds one step at a time.
- Click the Run button in the Simulator toolbar.

NOTE: The Step button advances the simulation to the next time at which there is some circuit activity, not necessarily just one time unit. The size of the step will depend upon the circuit.

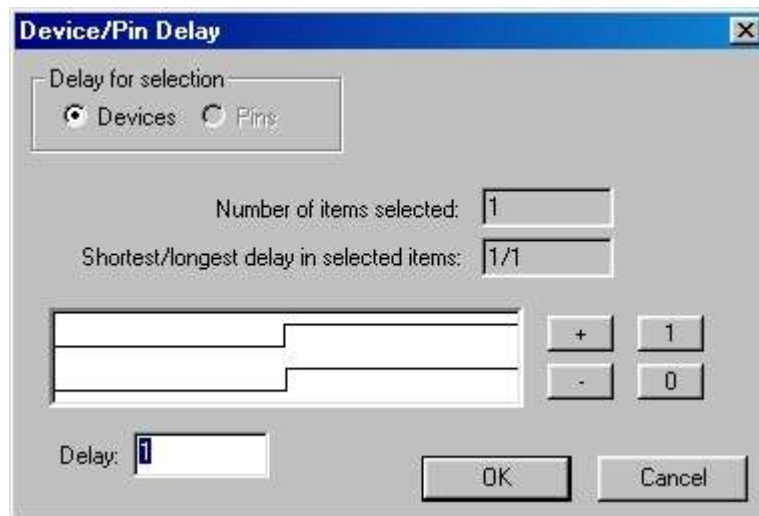
As the simulation progresses, the values on all probes are updated immediately. A similar device, the Hex Display, is also available to show groups of lines in hexadecimal. These simulation devices can be flagged to indicate that they are

not a real part of the finished product and should not be included in any netlists or bills of materials.

Setting Device Parameters

- Click in the window, but away from any circuit objects. This deselects everything.
- Click on the OR gate to select it.

- Select the Simulation Params command in the Simulation menu.
- Click on the "+" button a couple of times to increase the propagation delay in this device.

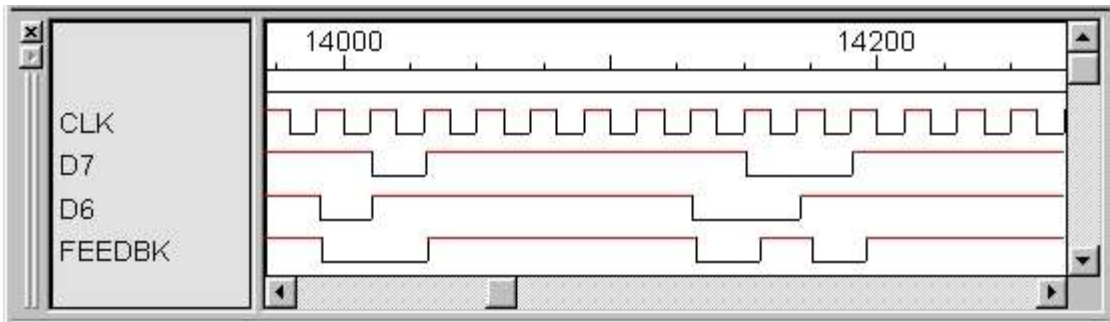


The Simulation Params command is used to view and set delays associated with devices and pins. Pin delays normally default to zero but can be used to fine-tune the delays for different paths through a device.


- Click on the OK button.

Device Delay on the Timing Window

Notice that the altered device delay immediately affects the simulation. You will see an increased delay between the clock reference lines and the changes in the FEEDBK signal.



Saving the Design

- Click the Save button (), and save your circuit so you can continue with it later.

This ends the Five-Minute Schematic and Simulation tutorial section.