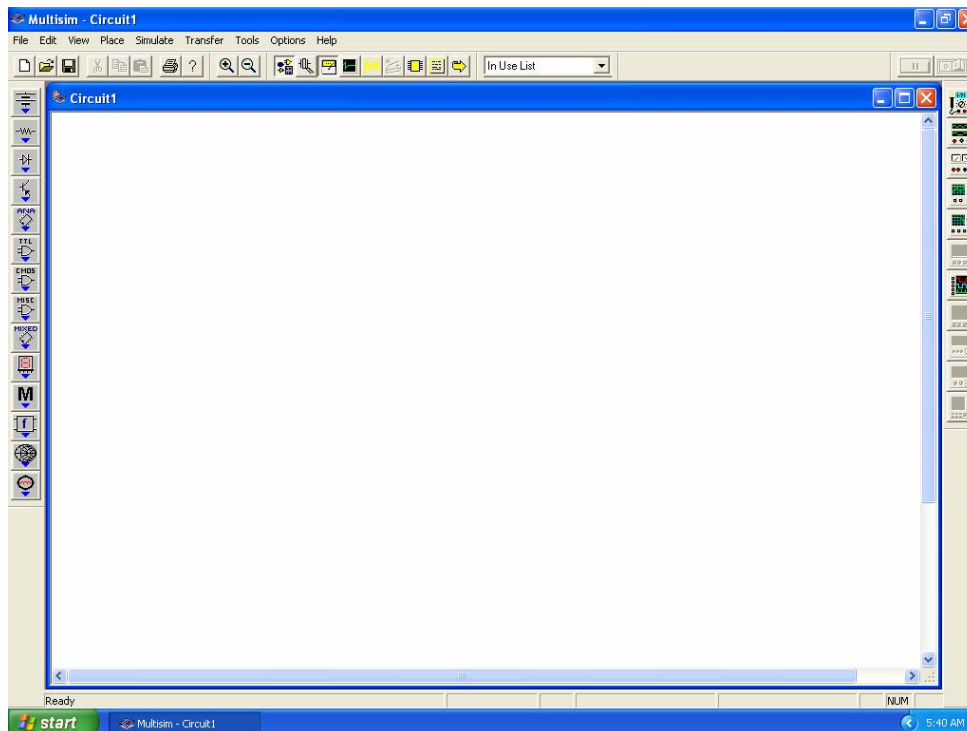


## Multisim Electronics Workbench Tutorial




























Multisim Electronics Workbench is available in several different versions including a professional version, a demo version, a student version, and a textbook version. The professional version is available in the labs on the CECS network and has all of the Workbench features. The demo version may be downloaded from the Workbench website at <http://www.electronicworkbench.com/html/eduproda.html>. The demo version is limited in that the user is unable to save or print results. A more fully featured student suite version (limited to 100 components) is available for home use from Prentice-Hall for about \$80 dollars. Finally, the textbook version is packaged with certain text books including the book currently being used in EE 210/215. It has some limited features such as the user may have only 50 components but this version can read and simulate larger files created on the professional version. This tutorial focuses on the features in the textbook version.

Figure 1 shows the opening screen of Workbench. To use the program you choose a set of components from the *Parts Bin* toolbar located on the left side of the screen and place them on the screen. The components are connected by using the cursor to drag "wires" between the parts. When the circuit is complete, you choose some type of measuring instrument from the *Instruments* toolbar on the right side of the screen. For example, you might connect an oscilloscope to the output of your circuit. This done, you can simulate your circuit and observe the output on the instruments you connected.



**Figure 1**  
The Multisim opening screen

Figure 2 lists the commonly used icons from the toolbars. Note that on the textbook version some icons are grayed out since they are not available on this version (for example there is no spectrum analyzer).

<b>Parts Bin</b>		<b>Instruments</b>	
	Source components. Includes dependent and independent sources and grounds		Multimeter. Amps, volts, or Ohms.
	Basic components. Resistors, capacitors, inductors and variations		Function generator. Sine, square, and triangular waveforms
	Diodes. Includes LED's and bridge networks		Wattmeter.
	Bipolar and MOS transistors		Oscilloscope (2-Channel)
	Analog components. Includes op amps and other amplifiers		Bode plotter. Does both magnitude and phase
	TTL. A wide selection of 74 and 74LS packages		Logic Analyzer
	CMOS. 4xxx and 74HC components	<b>Mode and Simulation</b>	
	Miscellaneous digital including VHDL and Verilog		Component mode button. Click this to add components
	Mixed analog/digital including converters and 555 timer		Component editor to modify component parameters.
	Indicators. voltmeter and ammeter, indicator lamps, and displays		Instruments. Click this to add an instrument.
	Miscellaneous components including a vacuum tube and a motor		Simulate button. Alternatively, push F5 to toggle start/stop and F6 to pause.
	Control system components such as summers, multipliers, etc.		Analysis. Transient, AC, DC, and Fourier included.
	RF components		VHDL and Verilog simulation.
	Electromechanical including a coil, transformer, and a relay.		Report. Click this to generate a report.

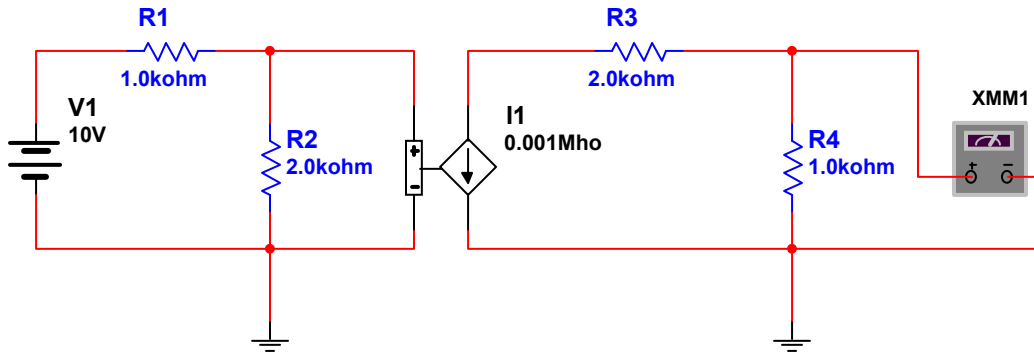
**Figure 2**

The principle icons on the three main tool bars of Workbench.

## Example 1

### A simple resistor network

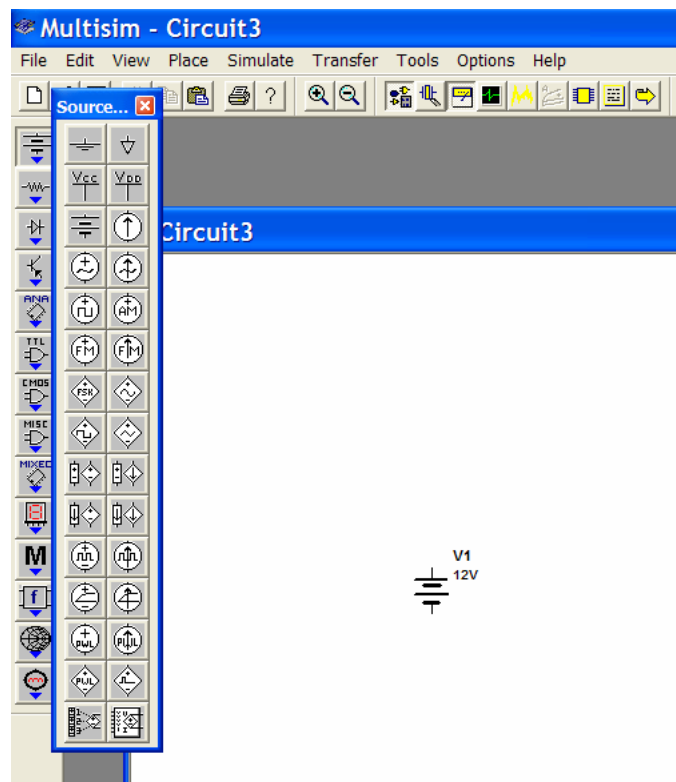
The circuit which we will simulate has a dependent voltage controlled current source, four resistors, a battery, and a meter to measure the output. The finished circuit is shown in Figure E1.



**Figure E1-1**

A resistor network with a dependent current source.

To begin click on the Source Components icon on the *Parts Bin* tool bar. This produces a selection of parts including various sources. Select a battery source from the menu by clicking on it. Click once more on the blank screen to place the battery. Figure E2 shows the result.



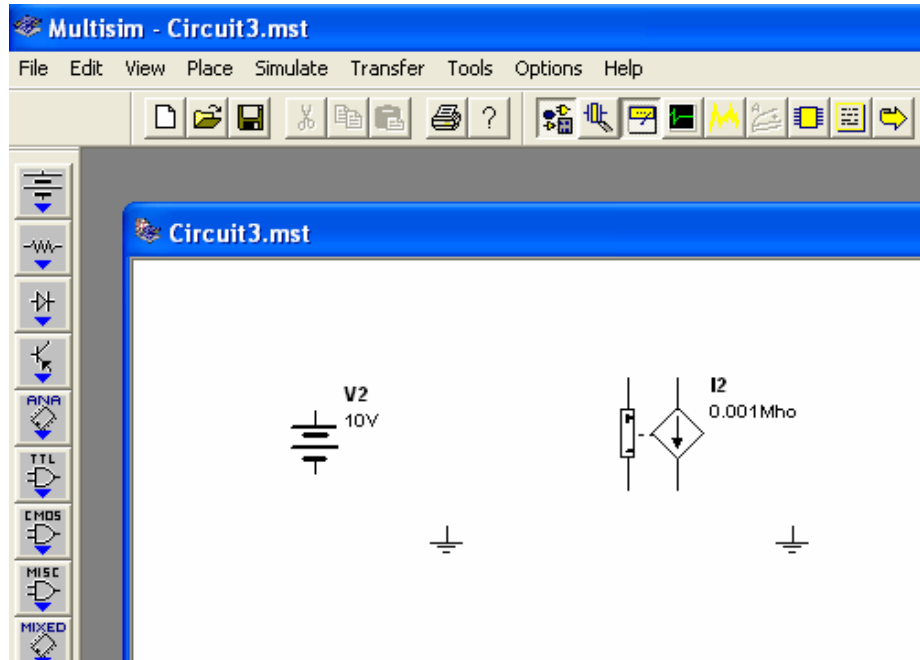
**Figure E1-2**

Select a battery voltage source and place it on the blank screen.

After placing the battery on the screen, double click on it to get a menu to set the battery values. Change the battery value to 10 volts.

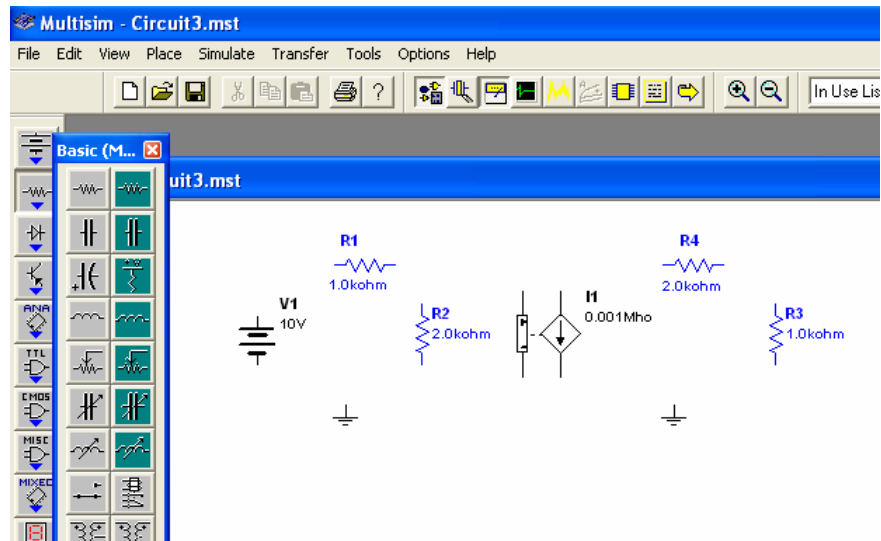
With the battery in place the next item to chose will be the voltage dependent current source. Choose this from the Source Components in the *Parts Bin*. A voltage dependent current source has a value in mhos or amps/volt. Set this value to 0.001 mhos. The circuit also has a ground on either side of the dependent source. Select a ground symbol from the Source Components and place them on the diagram. Your circuit should look something like that shown in Figure E3.

There are four resistors to be added to the circuit and we are done with the sources menu. You can close the source menu and open the Basic Components menu on the *Parts Bin* tool bar. Note that there are two types of resistors in the parts bin. One type is a "real world" resistor that has a value that you can purchase. The second type is a "virtual" resistor that is idealized and can have any value you choose to assign to it. Choose a 1K $\Omega$  resistor and place it on the circuit diagram between the battery and dependent source. Similary choose a 2K $\Omega$  resistor and place it immediately to the left of the dependent source. You can rotate a resistor by right clicking on it and choosing one of the rotation options. Choose another 1K $\Omega$  and 2K $\Omega$  resistor and place them to the right of the dependent source. Rotate the resistors as needed so that your circuit looks like that shown in Figure E4.



**Figure E1-3**

A battery and a voltage dependent current source have been placed.



**Figure E1-4**

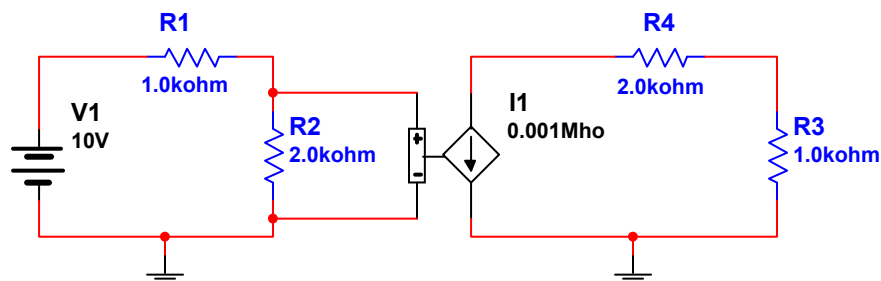
The circuit with all of the components in place. Note that resistors can be rotated by right clicking on them and choosing one of the rotation options.

With the components in place you are ready to connect the wires. Workbench has no wire mode. You connect wires by moving the cursor to a component wire end, click on it, go to a second component and click on its wire end. Workbench will then fill in a wire between those two components. If you get a wire wrong, simply click on the wire and push delete to get rid of it.

It may be tempting to place components close enough together so that their terminals touch and thereby avoid having to connect them together with a wire. But this doesn't work in Workbench. *Components must be connected together by wires.*

On occasion you may want to connect the end of one wire to a component. You cannot begin a new wire from an old one unless you place a junction on the wire first. You can do this from the *Place* menu at the top of the screen.

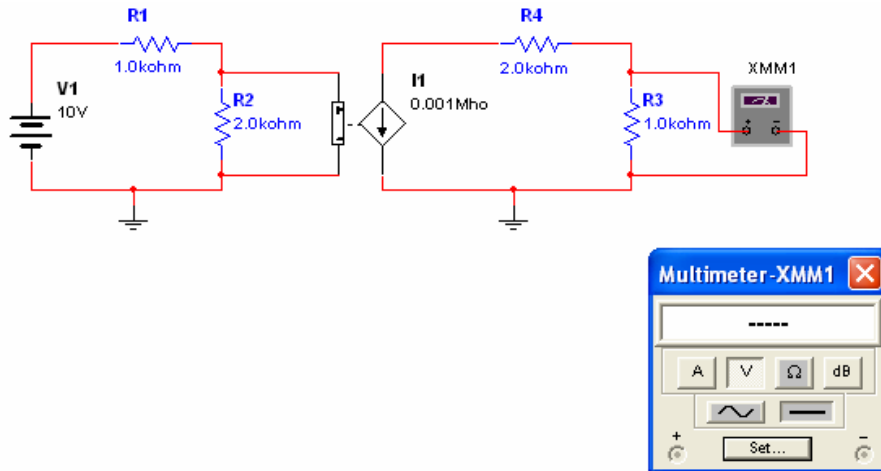
Sometimes Workbench does not choose a very good path between two points and the wire makes the circuit appear cluttered or unnecessarily complicated or the wire path passes through text. You can choose the path for any wire by clicking on intermediate points on the screen to form corners or junctions. You do this by first clicking on a component wire end, clicking on one or more intermediate points, and finally clicking on the component wire end of the second component. The wire path will then go from the first component through each intermediate point to the second component. After connecting the wires in the example your circuit should look like that shown in Figure E5.



**Figure E1-5**

The complete circuit with all of the components and wires in place.

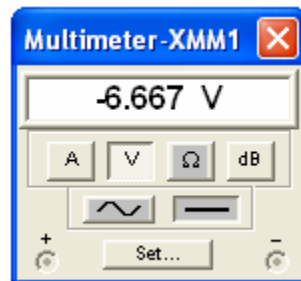
The circuit in this example is ready for simulation except that we have no way to see the output. The simulator in Workbench outputs only to instruments which you connect to the circuit diagram. For this dc example we will connect a voltmeter across R3 which we will take as the output. To connect the voltmeter, select the multimeter from the *Instruments* toolbar on the right side of the screen. Place the multimeter near the R3 resistor and connect the plus terminal to the top of R3 and the minus terminal to the bottom of R3 as shown in Figure E6. Double click on the multimeter and verify that it is set up as a dc voltmeter. Let the multimeter set up screen open so that you can view the results of the simulation.



**Figure E1-6**

Connect a multimeter across R3. The voltage across R3 is taken as the output voltage. Double click on the multimeter to open the meter's setup and view screen.

You can start the simulation by pushing the F5 function key, by pressing the on/off switch at the top right corner of the Workbench screen, or by selecting run from the Simulation menu at the top of the screen. The result of the simulation will be shown on the multimeter.



**Figure E7**

The multimeter showing the results of the simulation.

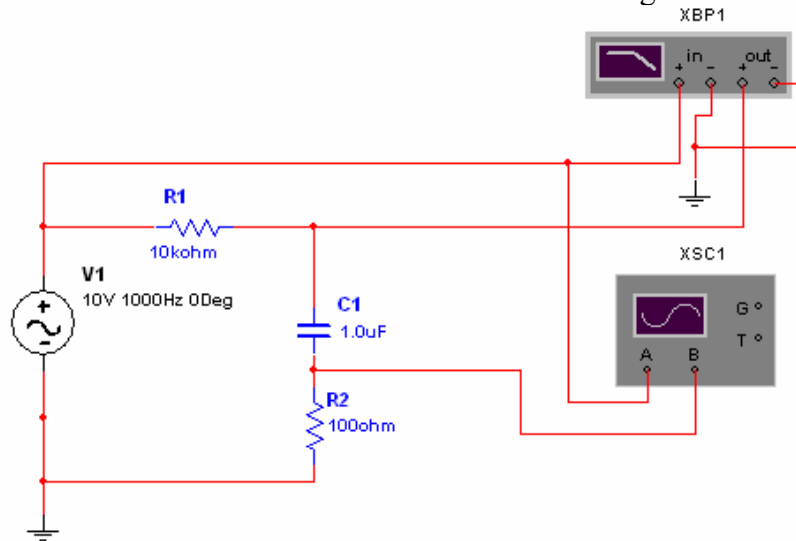
The Multimeter also allow can be used as an ammeter or an ohmmeter and, by clicking on the Set... button you can manually set the impedance of the meters to conform to real world instruments.

Workbench also has a voltmeter and an ammeter in the *Indicators* section of the *Parts Bin*.

## Example 2

### An RC network using the oscilloscope and Bode plotter

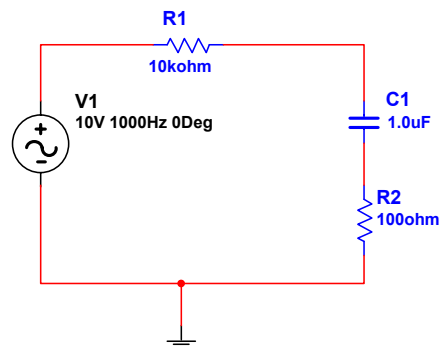
In this example we use the oscilloscope and the Bode plotter in an RC circuit that has an AC source. The circuit which we will construct is shown in Figure E2-1.



**Figure E2-1**

The complete circuit showing the oscilloscope and Bode plotter.

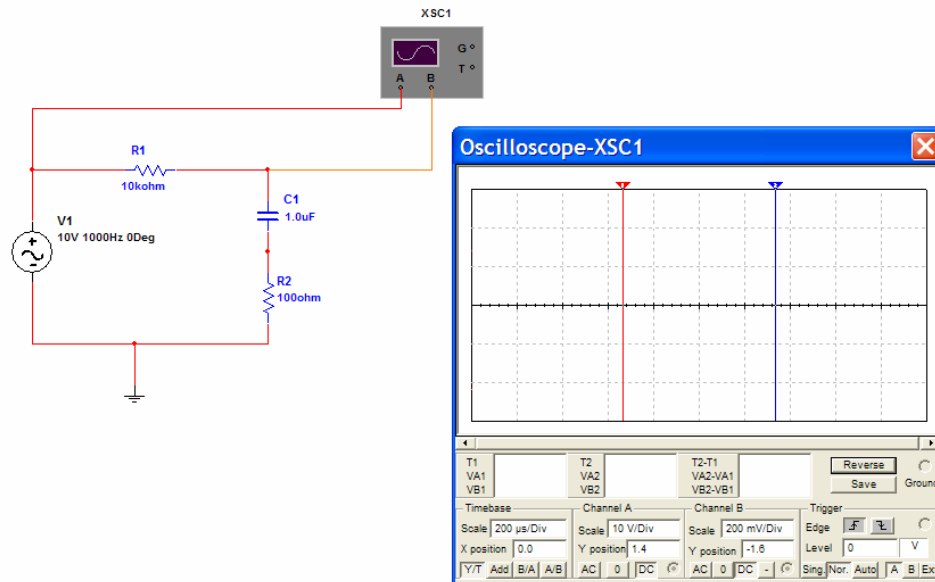
Begin by placing a 10K $\Omega$  resistor, a 1.0 $\mu$ F capacitor, a 100 $\Omega$  resistor, an AC voltage source, and an analog ground as shown in Figure E2-2. Use the cursor to draw in the connecting wires. If you are unfamiliar with placing components and connecting wires refer to Example 1.



**Figure E2-2**

The basic RC circuit with all components in place. Note that you will have to change the default value of the AC voltage source to 10 Volts by double clicking on the source and setting the value on the pop-up menu.

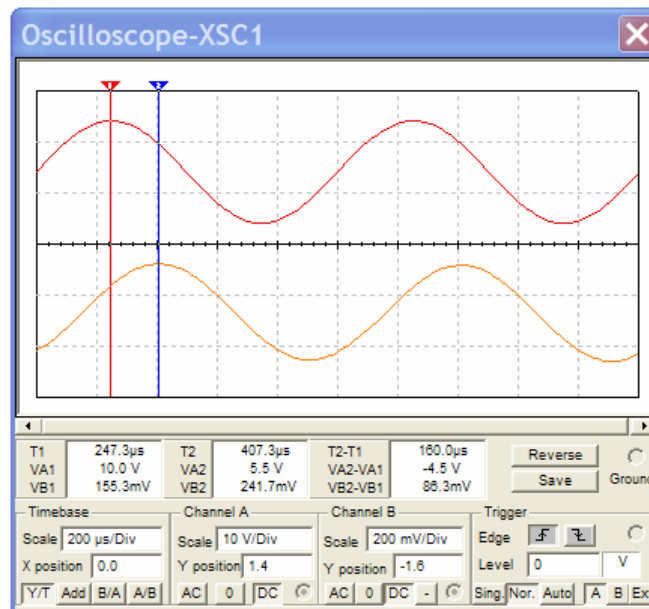
For this circuit we will take the input to be the 10 volt AC source and the output to be the voltage across the series combination of C1 and R2. Select the oscilloscope from the *Instruments* toolbar and connect the A channel to the top of V1 and the B channel to the top of C1. It's not necessary to connect the oscilloscope ground. Your circuit should look like that shown in Figure E2-3. Double click on the oscilloscope to open the oscilloscope control window and set the values as shown in the figure.



**Figure E2-3**

The circuit with the oscilloscope connected. The time base is set to 200µsec/div. Set channel A to 10 V/Div and channel B to 200 mV/Div. Set the y-position of channel A to +1.4 volts and the y-position of channel B to -1.6 volts.

Push the F5 function key to run the simulation (or push the rocker switch at the top right of the Workbench screen). The oscilloscope display will look like that shown in Figure E2-4.



**Figure E2-4**

The oscilloscope showing the simulation.

The oscilloscope display has two arrows in red and blue near the top of the display. Use the cursor to move the red arrow to a peak of the channel A display and move the blue arrow to the peak in the channel B display. The display will then show the difference in time as T2-T1 about equal to 160 µsec. Note that the input voltage source is at 1KHz so the phase shift between the two is  $(160/1000) \times 360^\circ = 57.6^\circ$ .

Note that the oscilloscope has a Save button. This allows you to save the data collected and displayed by the scope into an ascii file. The file can then be read by other program such as Matlab or Excel for reports or graphs. When you click on Save, you are



prompted to enter a file name which has the extension .scp by default. For this example the first few lines of such a file look like that shown in Figure E2-5 when read by a program like Notepad.

```
Oscilloscope data for

Time base: 0.000200 seconds per division
Time offset: 0.000000 seconds
Channel A sensitivity : 10.000000 volts per division
Channel A offset: 14.000000 volts
Channel B sensitivity : 0.200000 volts per division
Channel B offset: -0.320000 volts
Channel A connected: yes
Channel B connected: yes

Column 1  Time (S)
Column 2  Channel_A Voltage (V)
Column 3  Channel_B Voltage (V)

Time                Channel_A      Channel_B
-----
0.000000000000e+000  0.0000e+000  -5.0931e-002
7.294321988385e-006  4.5819e-001  -4.6255e-002
1.729432198838e-005  1.0845e+000  -3.9247e-002
2.729432198838e-005  1.7066e+000  -3.1671e-002
3.729432198838e-005  2.3219e+000  -2.3557e-002
4.729432198838e-005  2.9281e+000  -1.4938e-002
5.729432198838e-005  3.5227e+000  -5.8467e-003
...
```

**Figure E2-5**

This is the first part of a file containing the oscilloscope data. The data is in ascii format and may be read by other program such as Excel.

The file can be read into Excel directly and the column data can be plotted using the Excel plot features. To read the program into Matlab you must first strip off the header data so that the file consists only of the time base, Channel A, and Channel B data. You can then load and plot the three columns of data into a single Matlab variable with a MatLab statement such as

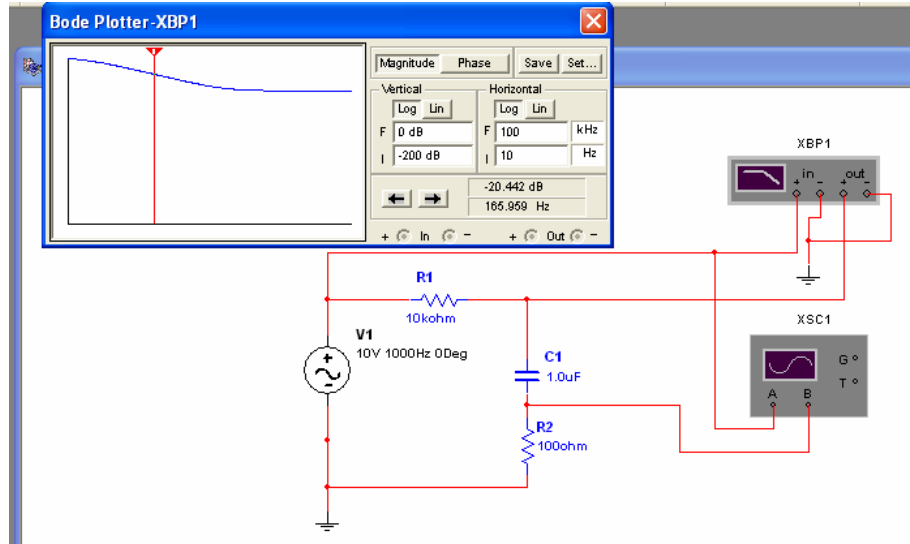
```
s = load('RCBode.txt'); %RCBode.txt is the amended scope file name.
plot(s);
```

Note that when plotting data in either Matlab or Excel the scaling that is done by the oscilloscope is lost.

The Bode plotter is an instrument that allows you to look at the frequency response of your circuit. Select the Bode plotter from the *Instruments* tool bar on the right side of the screen. The Bode plotter has both input and output terminals and has the additional requirement that your circuit must have at least one AC source as a circuit element. For our example connect the input terminals of the Bode plotter directly across the AC source and the output terminals across the series combination of C1 and R2. This is shown in Figure E2-1 where a ground is used as a common connection.

In use, the Bode plotter disables the AC source so that it has no real effect on the Bode plot. The plotter applies a series of sinusoids to the input and monitors the sinusoids on the output to determine the gain and the phase shift produced by the circuit.

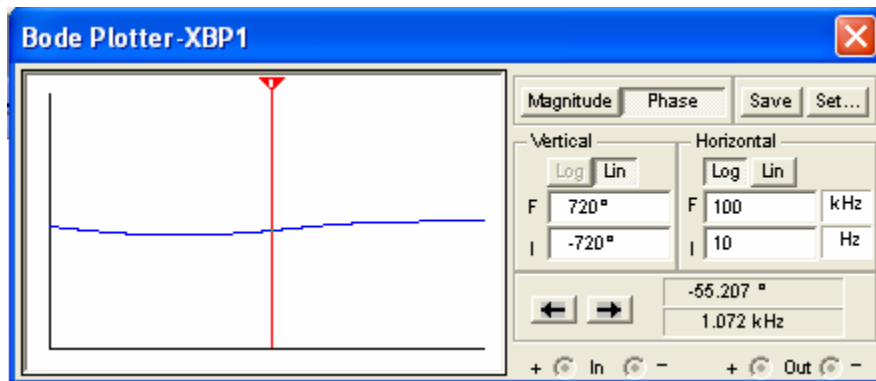
Double click on the Bode plotter to get the control and output screen for the instrument. By tradition, a Bode plot is logarithmic plot of gain magnitude and phase vs. frequency. On Workbench you can alter the vertical and horizontal scales to be either linear or logarithmic. For this example we will leave the traditional log plot in place. Change the horizontal frequency scale so that the starting frequency (in window marked I) is at 10Hz and the ending frequency (in the window marked F) is at 100KHz. Run the simulation to get the Bode plot similar to that shown in Figure E2-6 below.



**Figure E2-6**

The Bode plot simulation showing the magnitude function. Note that you can move the red arrow horizontally and read out values in the window. The window above shows that the gain is at -20.442DB at a frequency of 165.059Hz.

Click on the Phase button to see the phase curve. If you move the horizontal arrow to 1000Hz you will see the display shown in Figure E2-7.



**Figure E2-7**

The phase curve with the marker set to about 1KHz. The phase is  $-55.207^\circ$  which agrees with the phase measurement taken on the oscilloscope.

Like the oscilloscope, the Bode plotter also has a Save option which allows you to save the data taken by the plotter as an ascii file. The first part of the Bode data file is shown in Figure E2-8. You can read the data file directly into Excel and make a prettier plot or you can strip off the header information and read the data into Matlab.

```

Bode data for

column 1   Frequency (Hz)
column 2   Gain (dB)
column 3   Gain (Linear)
column 4   Phase (Deg)
trace name: Bode Result

Frequency      Gain (dB)      Gain      Phase
-----
1.00000e+001  -1.46954e+000  8.44351e-001  -3.20393e+001
1.02329e+001  -1.52790e+000  8.38697e-001  -3.26306e+001
1.04713e+001  -1.58818e+000  8.32897e-001  -3.32275e+001
1.07152e+001  -1.65042e+000  8.26950e-001  -3.38295e+001
1.09648e+001  -1.71465e+000  8.20857e-001  -3.44365e+001
1.12202e+001  -1.78090e+000  8.14620e-001  -3.50483e+001
1.14815e+001  -1.84921e+000  8.08239e-001  -3.56645e+001
1.17490e+001  -1.91960e+000  8.01715e-001  -3.62850e+001
1.20226e+001  -1.99211e+000  7.95050e-001  -3.69093e+001
1.23027e+001  -2.06676e+000  7.88246e-001  -3.75374e+001
...

```

**Figure E2-8**

The first portion of the text file created by the Bode plotter. Click on the Save button to create this file.

To get the Matlab plot, open the Bode plot data file in a word processor such as Word or Notepad, strip off all of the information above the raw data appearing in the 4 columns, and save the resulting file in a txt format. In Matlab you can plot the magnitude data using the following commands:

```

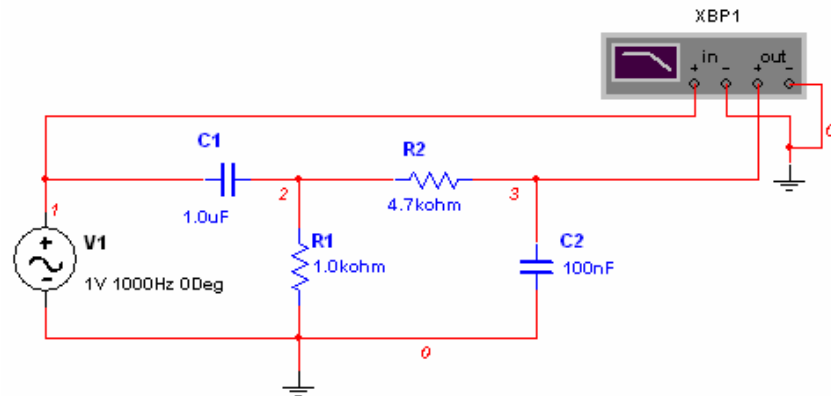
s = load('BodeData.txt');
semilogx(s(:,1),s(:,2));

```

### Example 3

#### Transient and AC Analysis

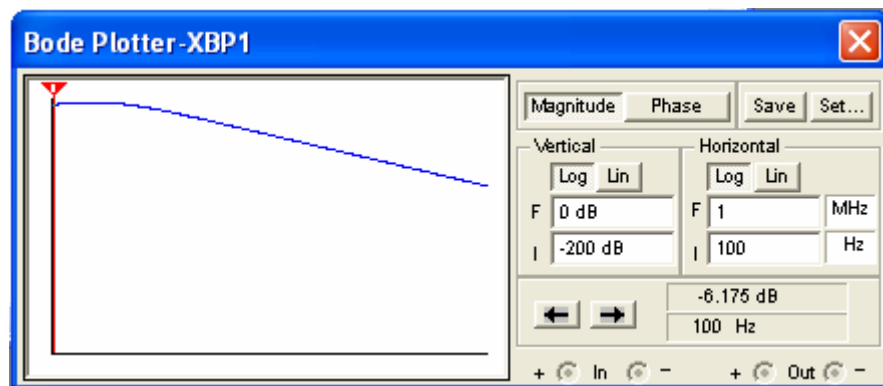
For this example we will look at the analysis options under the simulate menu. Begin by constructing the circuit shown in Figure E3-1.



**Figure E3-1**

An RC circuit with a Bode plotter and an AC source for use in doing the analysis options.

Note that in this circuit we have changed the AC voltage source from the default 10 volts to 1 volt. You might be tempted to use the function generator in place of the AC voltage source but the function generator will not work with AC Analysis. Workbench requires at least one AC voltage source in the circuit. We have also numbered the nodes since it is necessary to know the node numbers for the analysis. You can number the nodes in your circuit by choosing *Options* → *Preferences* and checking the *Show node names* box. The output node is number 3.



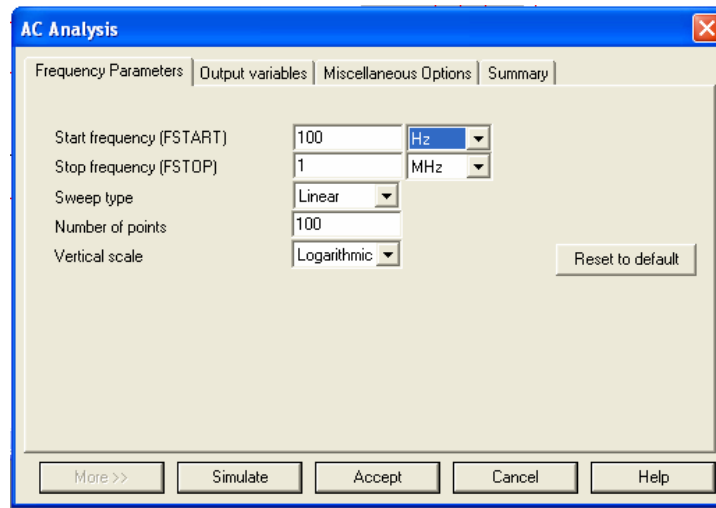
**Figure E3-2**

The Bode plotter set up and result.

Double click on the Bode plotter and set the frequency range to run from 100Hz to 1MHz. (Note that in the Bode plot set up mHz stands for milli-Hertz and MHz stands for MegaHertz.) Run the simulation by pushing F5 and you will get the Bode plot shown in Figure E3-2.

#### AC Analysis

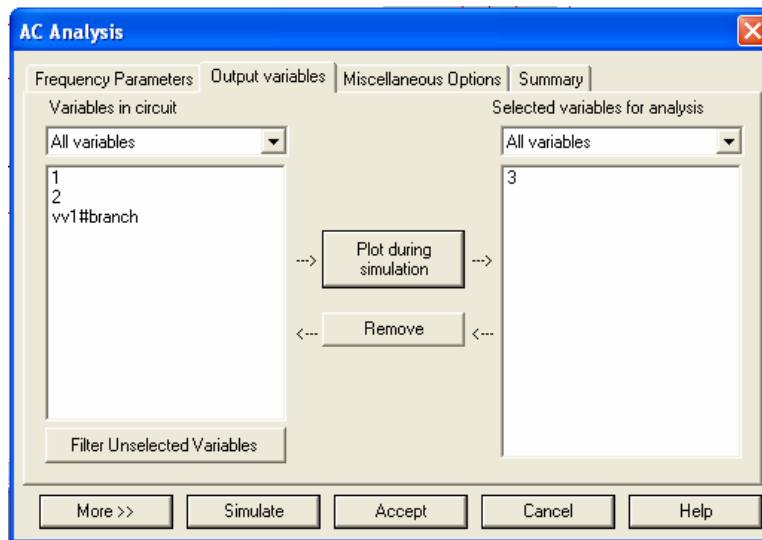
The first analysis that we will run on this circuit will be an AC analysis. From the menu select *Simulate* → *Analysis* → *AC Analysis*. This will give you a menu screen similar to that shown in Figure E3-3. Choose the starting and ending frequency range to run from 100Hz to 1MHz and set the sweep type to linear with 100 points and a logarithmic vertical scale.



**Figure E3-3**

Choose *Simulate*→*Analysis*→*AC Analysis* and set up the frequency parameters as shown.

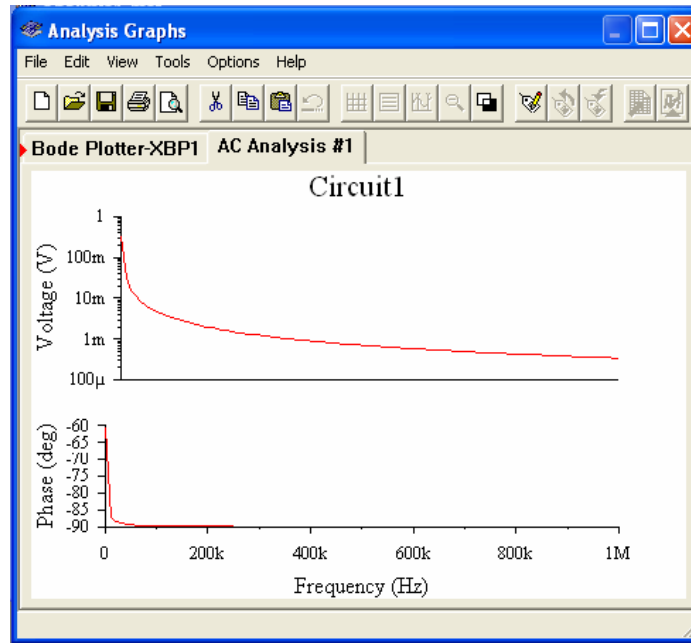
Next choose the Output variables tab from the top of the menu, select node 3 from the left side of the page and click on the arrow to move it to the right side.



**Figure E3-4**

The output variables screen for AC Analysis. Select node from the left and click on the "Plot during simulation" button to move it to the right.

Click on Simulate to complete the AC Analysis simulation. The results will appear in a new window called "Analysis Graphs" as shown in Figure E3-5. Note that the Bode plot which we did before also appears in this window under its own tab. The AC analysis and the Bode plot both show the same information but they are plotted with different scales. The Bode plot has a logarithmic frequency scale and we chose a linear frequency scale for the AC analysis.



**Figure E3-5**

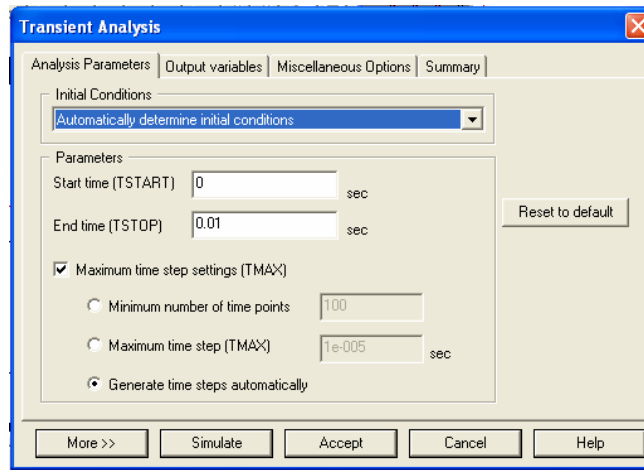
The AC Analysis. Note that this graph provides the same information as does the Bode plotter but the scales are different.

### Transient Analysis

In some books the term "Transient Analysis" refers to the step and impulse response of a given circuit. For Workbench the term simply means the analysis of the circuit beginning at time 0 and running for a user specified time. The input is not limited to a step or impulse.

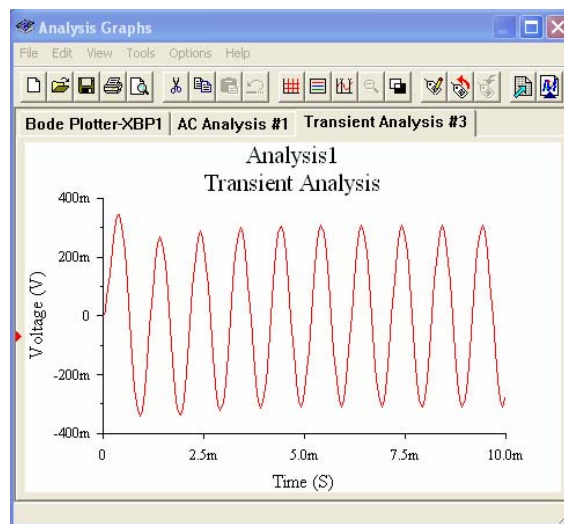
To do a transient analysis choose *Simulate*→*Analysis*→*Transient* from the menu. This will produce an options screen similar to that shown in Figure E3-6. Change the end time (TSTOP) to 0.01 seconds. Click on the Output variables tab and verify that node 3 is the output variable. Click on the simulate button to get the results of the simulation. This is shown in Figure E3-7. Since we have an AC source as the input the transient analysis shows the sinusoidal start up at the output. The transient start up rapidly dies away within a couple of cycles and settles to the steady state response of the circuit.

To get a step response we would need to change the input voltage source to a Pulse Voltage Source and set the frequency and duty cycle such that the circuit settles out during each cycle. For this particular example, delete the AC Voltage source by clicking on it one time and pushing delete. Select a Pulse Voltage Source from the Parts Bin and connect it in place of the AC source. Double click on the pulse source and set the voltage pulse to go from 0 to 1 volt, set the delay time to 0, the rise and fall times to 1 nsec, and pulse width to 10 msec with a period of 20 msec. Choose simulate to see the results shown in Figure E3-8.



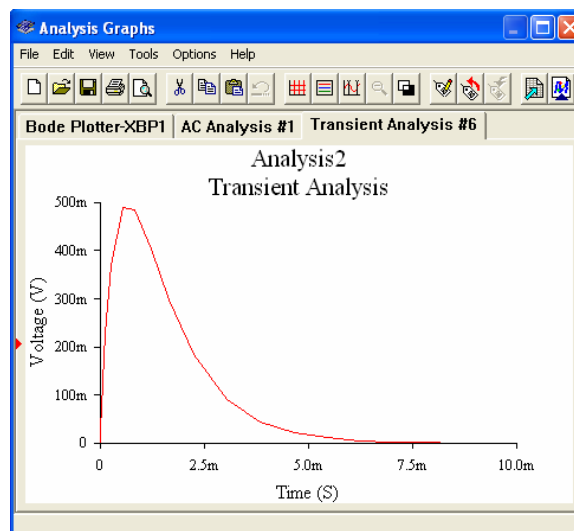
**Figure E3-6**

The Transient Analysis menu options. Change the value of TSTOP to 0.01 seconds.



**Figure E3-7**

The transient analysis results showing the sinusoidal start up at time 0.



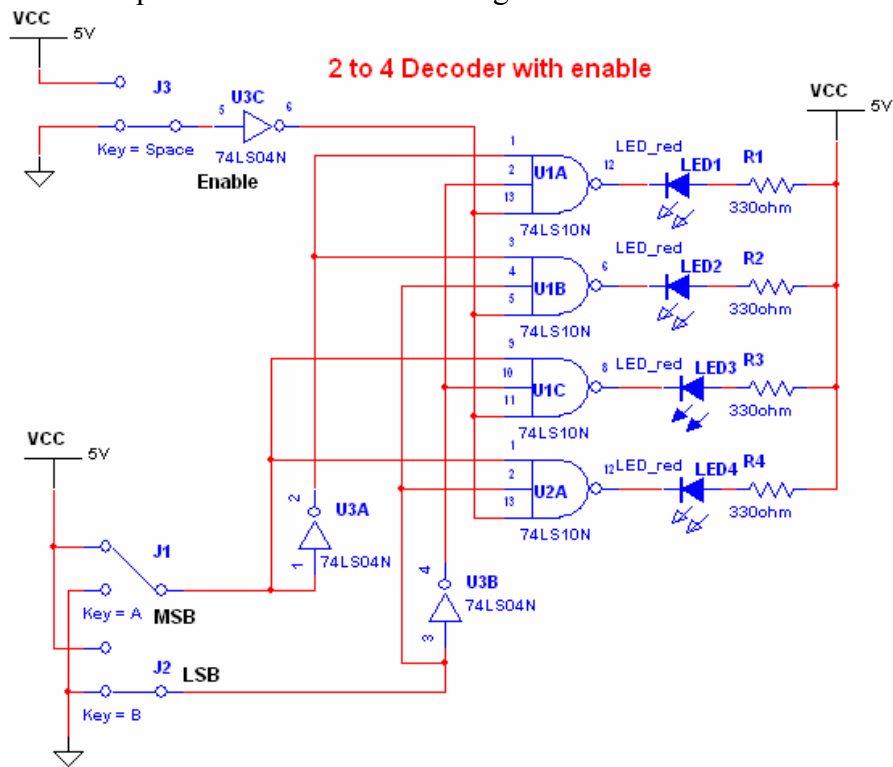
**Figure E3-8**

The step response of the circuit.

## Example 4

### Combinational Logic – A 2 to 4 decoder with enable

For this example we will simulate a 2-line to 4-line decoder with mechanical switches as input devices and LED's as out put displays. The decoder logic will be made from 74LS TTL parts. The completed circuit is shown in Figure E4-1.

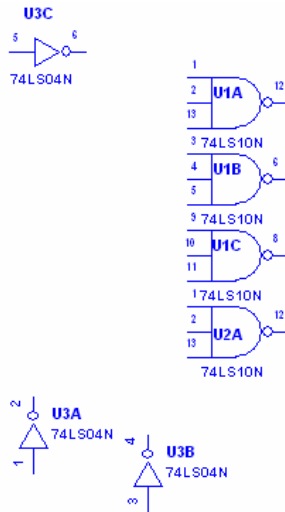


**Figure E4-1**

A 2-line to 4-line decoder with an enable. The red LED's indicate the state of the output.

Begin by placing the TTL parts on the diagram first. Select the TTL icon from the Parts Bin and choose the 74LS series. The 3-input nand gates are of type 74LS10 and there are three gates per package so you will need two packages. For the inverters use the 74LS04 part which has 6 gates per package and will need only three gates. Note that you can rearrange the text that goes with each package to allow for a neater configuration. In Figure E4-1 I have moved the UXX package designations onto the gates themselves. The pin numbers shown would be used if you were actually building this circuit but since this is just a simulation example the pin numbers for an individual gate are of no importance. Be sure to right click on the two bottom most inverters and rotate each counter-clockwise. Your circuit should look something like that shown in Figure E4-2.

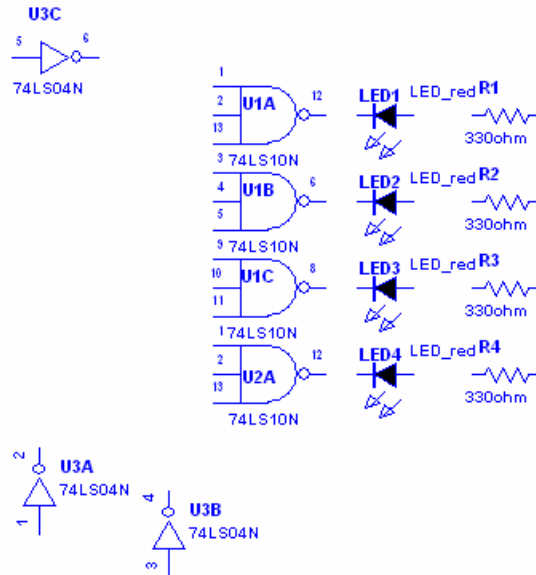




**Figure E4-2**

Place the 3-input together using two packages of 74LS10 type. Use one package of 74LS04 type for the inverters. Right click on the bottom most inverters and rotate them counter-clockwise.

Next we place the LED's on the board. These are on the Diodes menu in the Parts Bin. You have a choice of several colors. I have rotated the LEDs clockwise and moved their names around to tighten the configuration. After placing the first LED you can copy it and paste it for the others or you can select another LED from the *in use list* at the top center of the Workbench page. Add in four 330Ω resistors as shown in Figure E4-3.

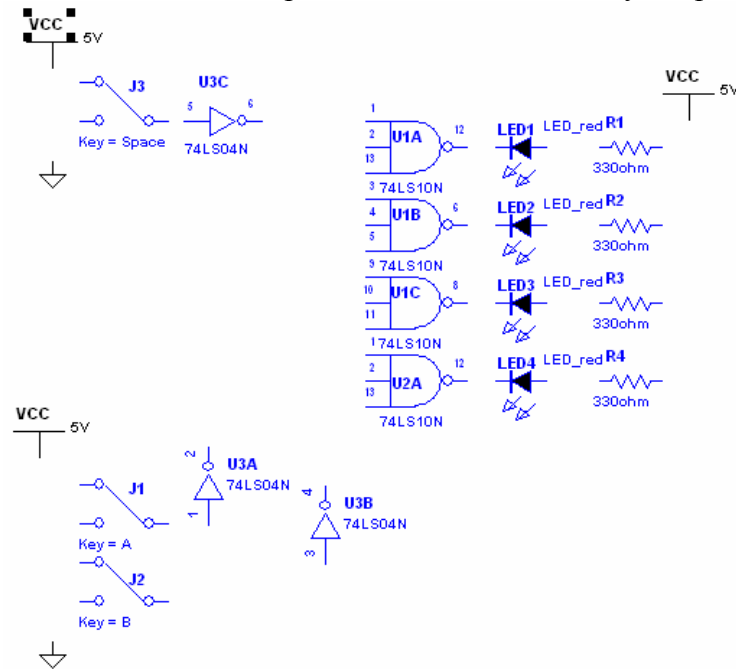


**Figure E4-3**

Place four LEDs and four 330Ω resistors on the board as show.

Finally we need to add the switches and the power supplies. Workbench has switches in two places. The switches we will use are located on the Basic menu and they are single pole double throw (SPDT) switches. (There are some momentary contact switches on the Electromechanical menu also.) Add three of the switches to the circuit and flip each horizontally as shown in Figure E4-4. Add in the digital power supply in three places for convenience.

Notice that each switch has a corresponding "key" parameter. This refers to the key on the keyboard which activates the switch during simulation. Double click on any switch and you get a menu that allows you to change this key. Change the two bottom switches to have keys of A and B and set the top most switch to have a key of space.



**Figure E4-4**

The circuit with all of the components in place. Note that the key parameter of the three switches has been set to space, A, and B.

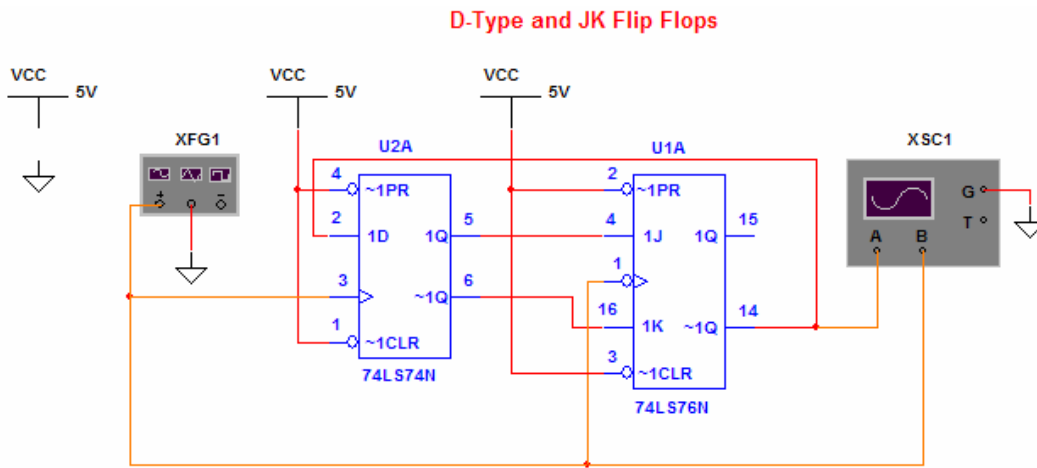
Connect the wires on the circuit so that your final circuit looks like that shown in Figure E4-1. Test your completed circuit by running the simulation (push F5). If all is correct, you should be able to activate the switches by pushing the space bar, the A, or the B keys on the keyboard. To get the LEDs to light the top switch will need to be in the ground position (active low) to enable the circuit. With the top switch down you will be able to light any LED by varying the two A and B switches to one of four different configurations.

Complete the project by adding some text to your circuit. Add text by selecting *Place*→*Place Text* from the menu. Click the cursor at the location on the screen where you want the text and enter the text by typing it in. After entering the text you can change its color by right clicking on it and selecting a color from the menu. You can also change the text font by selecting *Options*→*Preferences*→*Font* and selecting a font from the menu.

## Example 5

### Sequential Logic – D-Type and JK Flip Flops

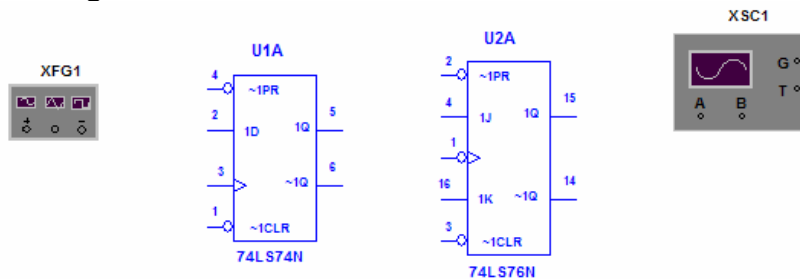
For this example we will simulate a circuit containing a D-Type and a JK flip-flop and make use of the function generator as a clock driver. The final circuit is shown in Figure E5-1.



**Figure E5-1**

A D-Type and JK flip-flop.

Begin by adding the two flip-flops to the diagram. These are both on the TTL XXLS series menu in the Parts Bin. For both the 74LS74 and the 74LS76 there are two independent flip-flops per package so you choose just one section from each package. After placing the flip-flops, choose a function generator and an oscilloscope and place them as shown in Figure E5-2.

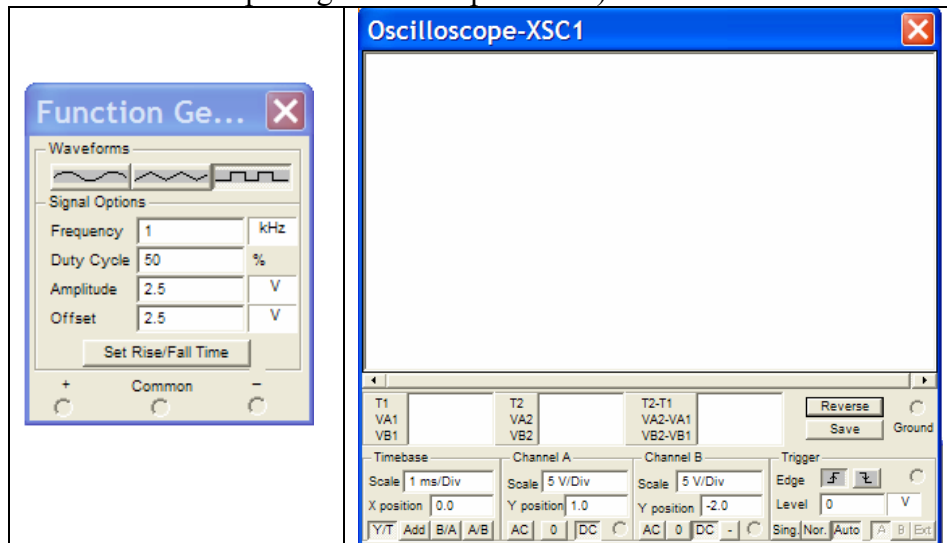


**Figure E5-2**

In this diagram we have chosen one section from the 74LS74 and one section from the 74LS76 flip flop packages. These are in the TTL Parts Bin. The function generator and the oscilloscope are chosen from the Instruments menu on the right.

This circuit is all digital. We will add three 5 volt Vcc power sources and three digital grounds to the circuit from the Sources menu in the Parts Bin. Connect the wiring to the diagram as shown in Figure E5-1. Be sure to connect the clear and preset terminals of the flip-flops to +5. Note that the center terminal of the function generator is grounded and the negative terminal is not used. The function generator is specified as having an amplitude, a frequency, a duty cycle, and an offset. The output wave is assumed to be centered vertically around 0 so that, for a square wave, an amplitude setting of 2.5 volts and offset setting of 2.5 volts produces a 5 volt square wave that goes from 0 volts to 5 volts. The negative terminal of the function generator puts out a signal of the opposite polarity and is not used in this simulation. Set up the function generator and the

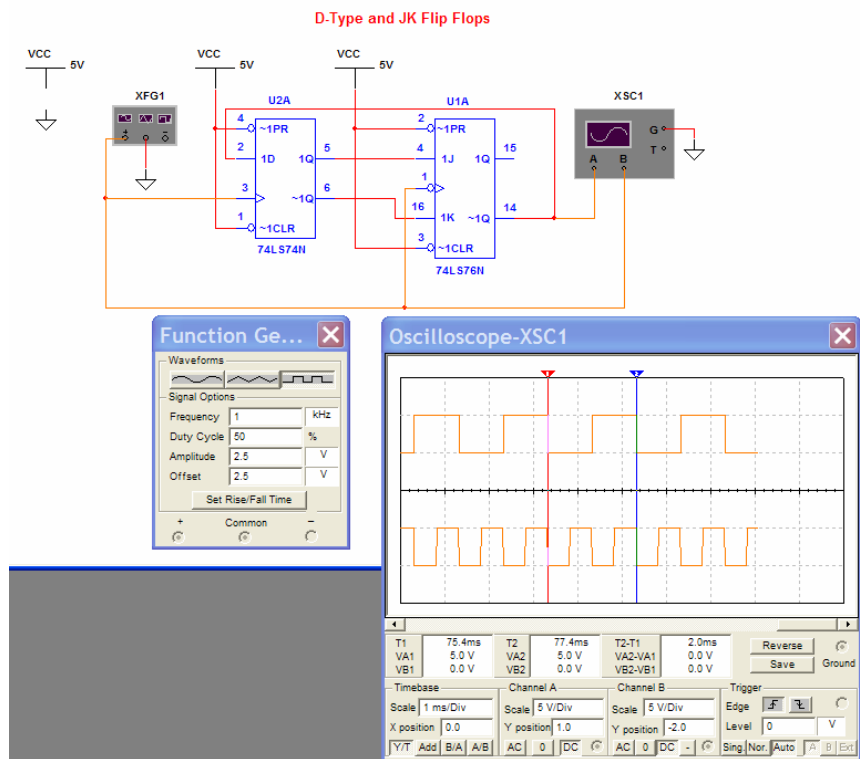
oscilloscope to look like those shown in Figure E5-3. (Double click on the function generator or the oscilloscope to get the set up screens.)



**Figure E5-3**

Set up the function generator and the oscilloscope as shown above.

Click on the simulate button in the top right corner or push function key F5. The simulation shows the JK flip-flop running at one half of the clock frequency.



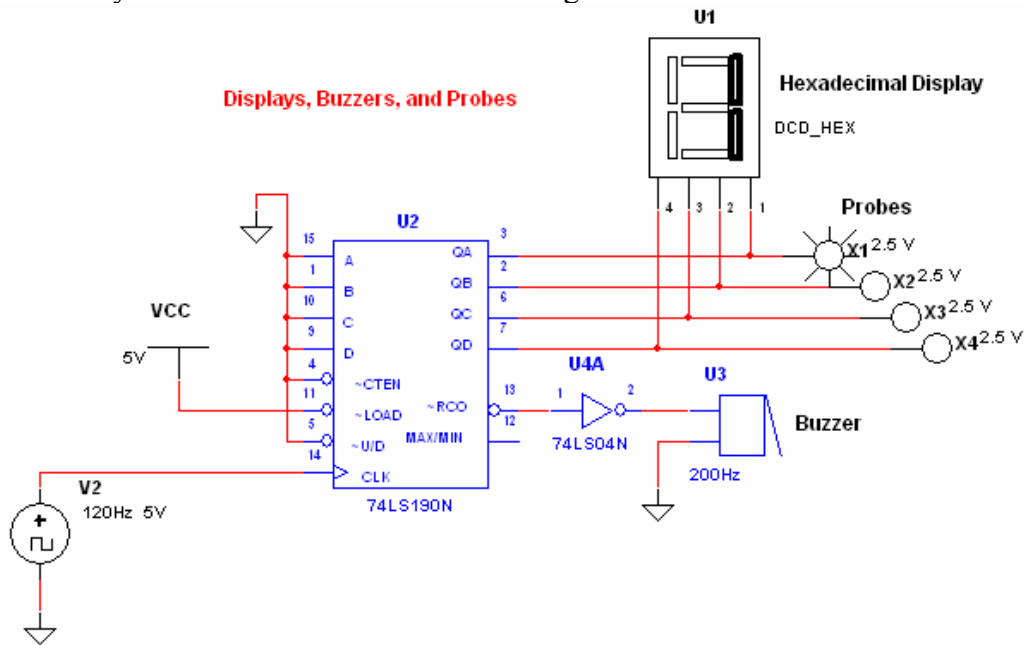
**Figure E5-4**

A simulation of the flip-flop circuit.

## Example 6

### Displays, Probes, Buzzers, and Busses

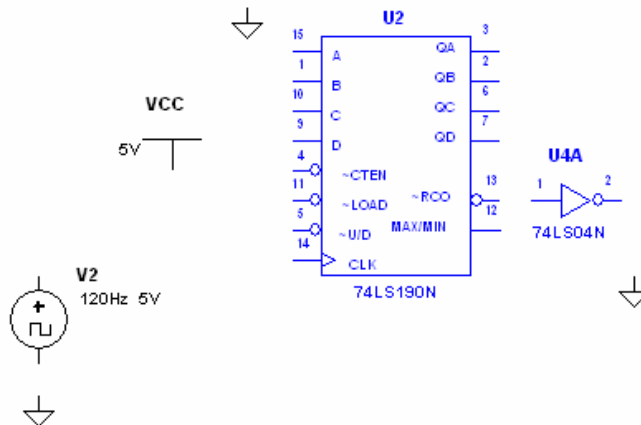
For this example we will simulate a circuit containing several indicators to illustrate how they are used and we will add a bus to the system to simplify the wiring. We will first simulate the system without a bus as shown in Figure E6-1.



**Figure E6-1**

This circuit uses a decade counter to illustrate the use of displays, probes, and buzzers.

For this circuit we need two TTL LS components which are the 74LS190 decade counter and one section of a 74LS04 inverter. Select these items from the Parts Bin and place them on the blank screen. Add a Clock Source from the Sources menu of the parts bin and set its frequency to 120Hz and its voltage to +5volts. You will also need a 5 Volt Vcc source and two digital grounds. Your circuit should look like that shown in Figure E6-2.



**Figure E6-2**

The TTL components and the sources have been added.

Add the three indicators as the next items. Click on the indicators icon on the Parts Bin and select the Hex Display item and select the part named DCD\_HEX. (All of the displays are somewhat idealized. The hex display shows a seven segment hex number that corresponds to the logic level on its four input pins. No power supply is necessary. Likewise the other two displays have no power

connection and light up when the input terminals are activated.) Place the hex display on the screen. Select a Buzzer from the same indicator menu and place it on the screen. Double click on the buzzer to set its voltage trip level. Set this voltage to 3 volts. You can also set a current level but we won't use that feature for this example. Click on the red probe and place four of these on the circuit diagram. Each of these has a trip point for being on or off which defaults to 2.5 volts.

Connect the wires to your circuit and get a final circuit similar to that shown in Figure E6-1.

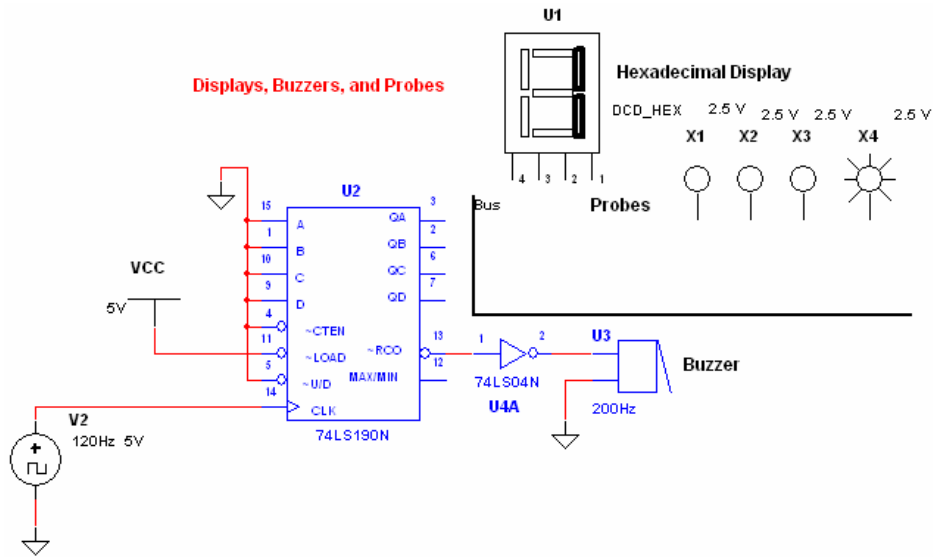
Note that for the hex display the least significant bit is pin 1 and the most significant bit is pin 4. For the 74LS190 counter the least significant bit is pin 3 and the most significant bit is pin 7. You might be tempted to flip the hex display horizontally by right clicking on it and selecting "flip horizontal" since this would place the least significant bit closer to that of the counter's least significant bit and result in less wire crossover. But if you do this with the display, Workbench also flips the display so the numbers displayed will appear as if you were looking at it from behind.

For a real circuit it's unlikely that you will have a circuit like this connected to a 120Hz clock source. If you did things would change so fast that you would be unable to see them. But if you slow the clock down to a reasonable rate such as 10Hz the simulation takes a very long time.

Simulate your finished circuit and verify that it does count upward from 0 to 9, that the probes do indicate the binary state of the output lines, and that the buzzer actually buzzes when the counter rolls over from 9 back to 0.

#### *Adding a Bus*

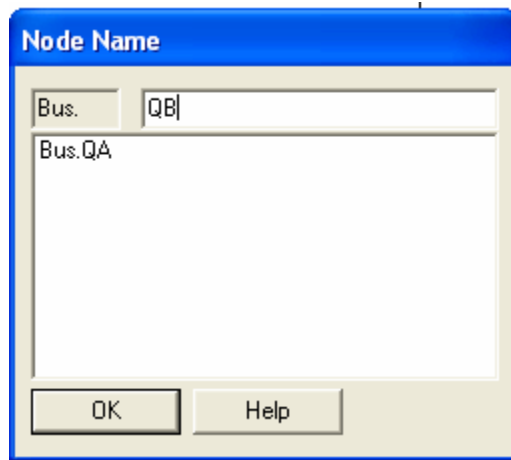
Before you add a bus to the system you will need to rearrange the hex display and the probes and delete the wires which the bus will replace. Rearrange your circuit to look something like that shown in Figure E6-3. To add the bus, click on *Place→Place Bus* from the menu at the top of the screen. Adding the bus is like adding a wire. Click on the starting point and click on first corner and double click at the end point. The bus will be drawn to be a series of lines between your clicks. If you double click on the completed bus you'll see that it has only one parameter which you can change and that's the reference ID. This defaults to the name bus for this example but you can make it anything you like. Note that a bus need not look continuous. You can place another bus anywhere else in the circuit and give it the same reference ID. Workbench then takes it to be the same set of lines. This allows you to draw complicated connections without having to drag the bus all over the diagram.



**Figure E6-3**

The hex display and the probes have been rearranged and the wires connecting them have been replaced with a bus.

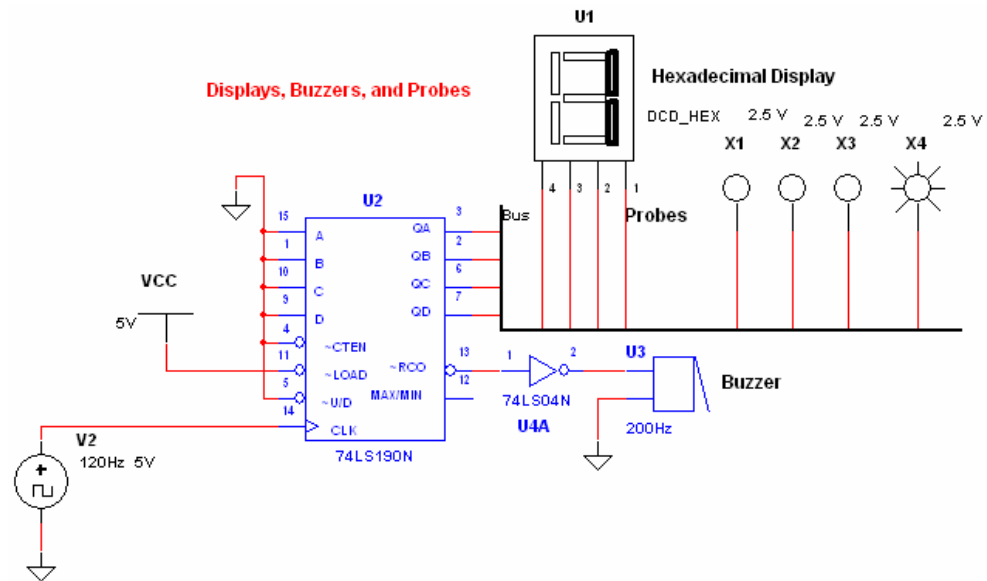
For this example connect the counter to the bus by connecting a wire to one of the counter outputs and connecting the other end to the bus. When you do this you will get a pop up screen that asks you to give the wire a name. In Figure E6-4 the wire is being named QB and one wire QA has already been named.



**Figure E6-4**

This pop up screen appears when you connect a wire to the bus. You can name the bus line or you can select a name already chosen.

For this example our bus will have only four wires and we will name them QA, QB, QC, and QD after the four outputs of the counter. After you connect the counter to the bus similarly connect the hex display and the probes as shown in Figure E6-5. Note that you should be careful to get the bits in the right order. In our case QA is the least significant bit and QD is the most significant bit.



**Figure E6-5**  
The circuit with a bus replacing the wiring connections.

Finally, simulate your circuit to verify that it works as before.