



Beni Suef Technological University
Egyptian-Korean Faculty of Technological Industry and Energy

Electric Circuits (Practice Manual)

Editor: Dr. Wael Zakaria Tawfik

Eng. Sammar Nady

Contents

Module 1: BASIC ELECTRICAL ENGINEERING PRINCIPLES

1. (Lab #1) Introduction to the Simulation Software “PSpice”
2. (Lab #2) Digital Multimeter for Resistance Measurement
3. (Lab #3) Function Generator and Oscilloscope
4. (Lab #4) Digital Multimeter for DC Voltage and DC Current Measurement

Module 2: FUNDAMENTALS OF ELECTRIC CIRCUITS

1. (Lab #5) PSpice Analysis of DC Circuits
2. (Lab #6) Ohm’s Law, Series and Parallel Connection
3. (Lab #7) Study & Verification of Maximum Power Transfer

Module 3: ELECTRICAL PRINCIPLES AND TECHNOLOGY

1. (Lab #8) Superposition
2. (Lab #9) Mesh vs. Nodal Analysis
3. (Lab #10) Thévenin’s and Norton's Theorems (PSpice & its manual)

Module 4: ADVANCED CIRCUITS

1. (Lab #11) First Order Circuits: RC and RL Transient Responses
2. (Lab #12) Second Order RLC Circuits
3. (Lab #13) Diode Curves
4. (Lab #14) Zener Diode Circuits
5. (Lab #15) PSpice Simulation of AC Input Circuits.

Module 1

BASIC ELECTRICAL ENGINEERING PRINCIPLES

Objectives

- Familiarize to PSpice tool
- Training on Measurement of resistance values using DMM
- Familiarize to main laboratory instruments

Tasks

- Introduction to simulation software “PSpice”
- Digital multimeter for resistance measurement
- Function generator and oscilloscope
- Digital multimeter for dc voltage and dc current measurement

1. (Lab #1) Introduction to Simulation Software “PSpice”

Objective:

A primary purpose of this lab is for you to become familiar with the use of PSpice and to learn how to use it to assist you in the analysis of circuits. The software is already installed in the computer of every station. This is just an introduction to PSpice. It is your responsibility to learn its use in a more detailed way since you will be using it along the course of this semester and in the future.

Introduction:

SPICE is a powerful general purpose analog and mixed-mode circuit simulator that is used to verify circuit designs and to predict the circuit behavior. This is of particular importance for integrated circuits. It was for this reason that SPICE was originally developed at the Electronics Research Laboratory of the University of California, Berkeley (1975), as its name implies: Simulation Program for Integrated Circuits Emphasis. PSpice is a PC version of SPICE (which is currently available from ORCAD Corp. of Cadence Design Systems, Inc.).

Computer-aided simulation is common practice in industry and is a very useful tool. SPICE is useful way of verifying your lab test results and experimenting with changes to your own circuit designs. It is also widely used in industry for simulating designs prior to production. Internal numerical accuracy of programs such as SPICE is very high with errors that seldom exceeding 1%.

Transistor circuit analysis is burdensome as the number of transistors increases beyond more than a few. Consequently, SPICE is used to test and simulate complex transistor circuits. There are several versions of the SPICE software now available. Aim Spice and PSpice are two versions. PSpice is a graphical simulator, whereas Aim Spice is text based. All SPICE programs are based on the core SPICE programming.

While PSpice makes extensive use of part libraries, Aim Spice uses text entries. Circuits may contain passive components such as resistors, capacitors, and inductors, and active devices such as transistors and diodes as well as independent voltage and current sources. To write code describing a circuit, nodes must be defined in the code. With nodes clearly defined, various elements are then connected between nodes to specified values. SPICE allows the user to perform various analysis of the circuit such as nonlinear dc, large-signal time domain (transient), small-signal frequency domain, nonlinear transient, and linear ac analyses. The dc and transient analysis capabilities are of greatest interest for digital circuit studies. In addition to performing the differing analysis types, SPICE also generates graphical outputs for which the various nodes and inputs can be graphed individually or together. SPICE software

is based on the same logic core in which the code is either manually generated as with Aim Spice or converted from a graphical representation by the software as PSpice does. A netlist file is manually written when using Aim Spice, whereas PSpice generates the netlist file containing the circuit elements and their interconnections for you based on the graphical representation.

Despite the accuracy of computer simulation, hand analysis is still necessary. SPICE simulation is a tool to enhance circuit analysis not replaces hand computations. For instance, hand calculations are the best method for developing appropriate simulation time intervals or rise times for a given circuit.

A curve tracer is a special type instrument similar to an oscilloscope designed to display voltage-current characteristics of three terminal devices such as transistors. The graphical display of an oscilloscope enables a user to easily view and identify the operating regions for a specific transistor and see how quickly the transistor saturates.

The PSpice Light version has the following limitations: circuits have a maximum of 64 nodes, 10 transistors and 2 operational amplifiers. SPICE can do several types of circuit analysis. Here are the most important ones:

- Non-linear DC analysis: calculates the DC transfer curve
- Non-linear transient and Fourier analysis calculates the voltage and current as a function of time when a large signal is applied; Fourier analysis gives the frequency spectrum
- Linear AC Analysis: calculates the output as a function of frequency. A bode plot is generated
- Noise analysis
- Parametric analysis
- Monte Carlo Analysis

In addition, PSpice has analog and digital libraries of standard components (such as NAND, NOR, flip-flops, MUXs, FPGA, PLDs and many more digital components). This makes it a useful tool for a wide range of analog and digital applications.

All analysis can be done at different temperatures. The default temperature is 300K. The circuit can contain the following components:

Independent and dependent voltage and current sources

- Resistors
- Capacitors
- Inductors
- Mutual inductors
- Transmission lines

- Operational amplifiers
- Switches
- Diodes
- Bipolar transistors
- MOS transistors
- JFET
- MESFET
- Digital gates

The values of elements can be specified using scaling factors (upper or lower case) as follows:

[Table 1-1] Units and its values

T or Tera (= 1E12);	U or Micro (= E-6);
G or Giga (= E9);	N or Nano (= E-9);
MEG or Mega (= E6);	P or Pico (= E-12)
K or Kilo (= E3);	F of Femto (= E-15)
M or Milli (= E-3);	

Types of Simulations:

PSpice have four main types of simulation:

1. Bias Point analysis
2. DC Analysis
3. AC Analysis
4. Time domain (transient)

Bias Point analysis

Saves detailed bias point information to the simulation output file. The information reported to the output file includes the following:

- A list of all the analog node voltages
- A list of all the digital node voltages
- The currents through all the voltage sources, and their total power
- A list of the small-signal parameters for all the devices. You can also view the result of this simulation by enabling the bias point display for voltage or current which are specified in (Fig 1-1).

DC analysis

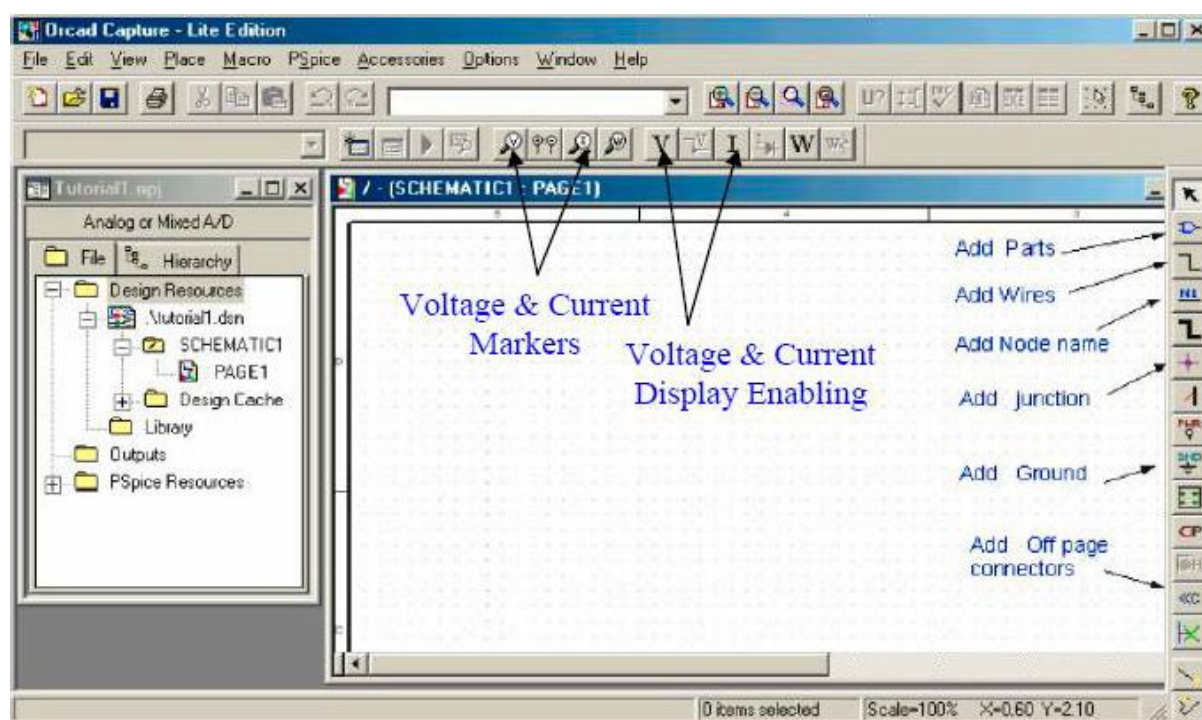
Performs a DC sweep. The DC sweep analysis calculates the circuit's bias point over a range of values. The results can be displayed on a graphical representation using the Voltage or Current Markers which are specified in (Fig 1-1). In the graphical representation the X-axis is the voltage or current source that is specified in the simulation profile.

AC analysis

Calculates the small-signal frequency response of the circuit (linearized around the bias point) over a range of frequencies. The results can be displayed on a graphical representation using the Voltage or Current Markers which are specified in (Fig 1-1).

Transient analysis

Calculates the behavior of the circuit over time. The results can be displayed on a graphical representation using the Voltage or Current Markers which are specified in (Fig 1-1).



[Figure 1-1] Graphical User Interface of Capture Software

In PSpice the program we run in order to draw circuit schematics is called CAPTURE. The program that will let us run simulations and see graphic results is called PSpice. You can run simulation from the program where your schematic is. There are a lot of things we can do with PSpice, but the most important things for you to learn are

- Design and draw circuits
- Simulate circuits
- Analyze simulation results (Probe for older versions)

For this course you will not need the full capacity of CAPTURE. The devices that we will use are resistors, inductors, capacitors and various independent/dependent sources. It is good to know that CAPTURE has extensive symbol libraries and includes a fully integrated symbol editor for creating your own symbols or modifying existing symbols.

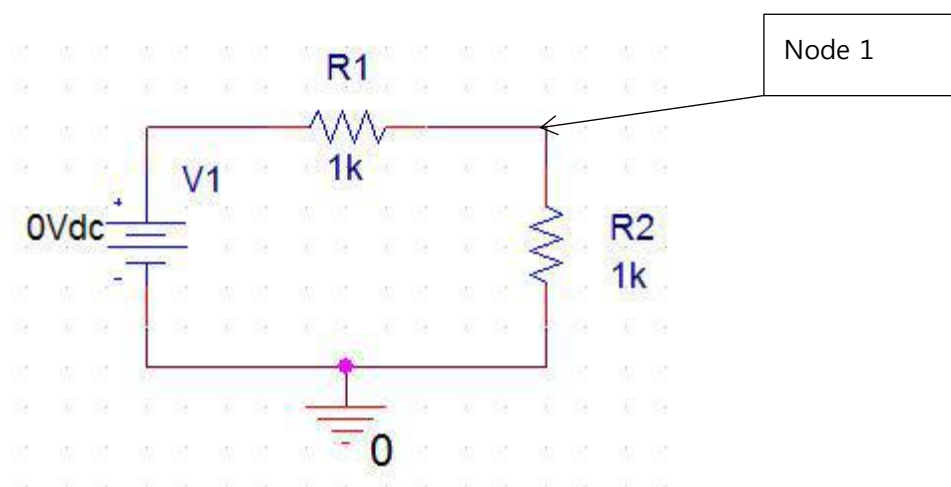
The main tasks in CAPTURE are

- Creating and editing designs
- Creating and editing symbols
- Creating and editing hierarchical designs
- Preparing your design for simulation

Tasks for the (Lab #1):

Familiarizing with the PSpice by the following Step by Step Procedure

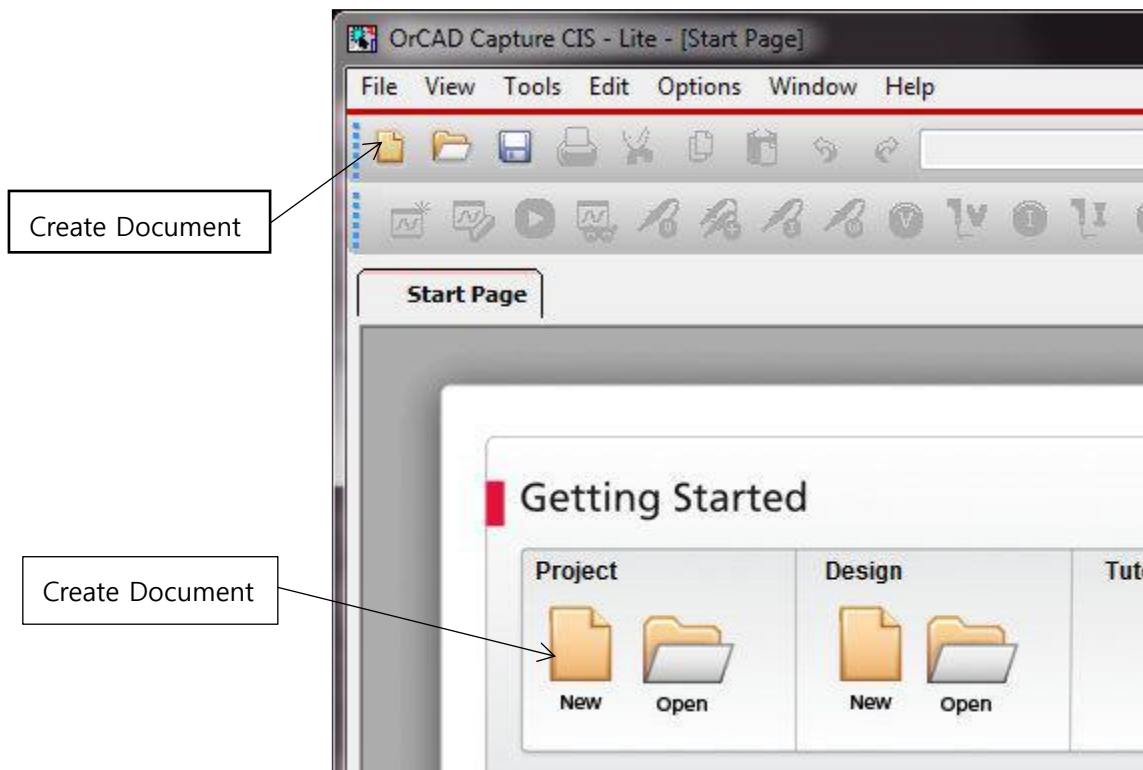
In this part we will create a simple DC circuit shown in (Fig 1-2) just to let you know how to start working with PSpice. Your goal is to find the current value in the resistor labeled R2 and the voltage at node 1 as we change the voltage V1 from -10V to +10V and obtain IV characteristics of R1. We will add two resistors, one DC voltage source, a ground and wires connecting components.



[Figure 1-2] Basic DC Circuit

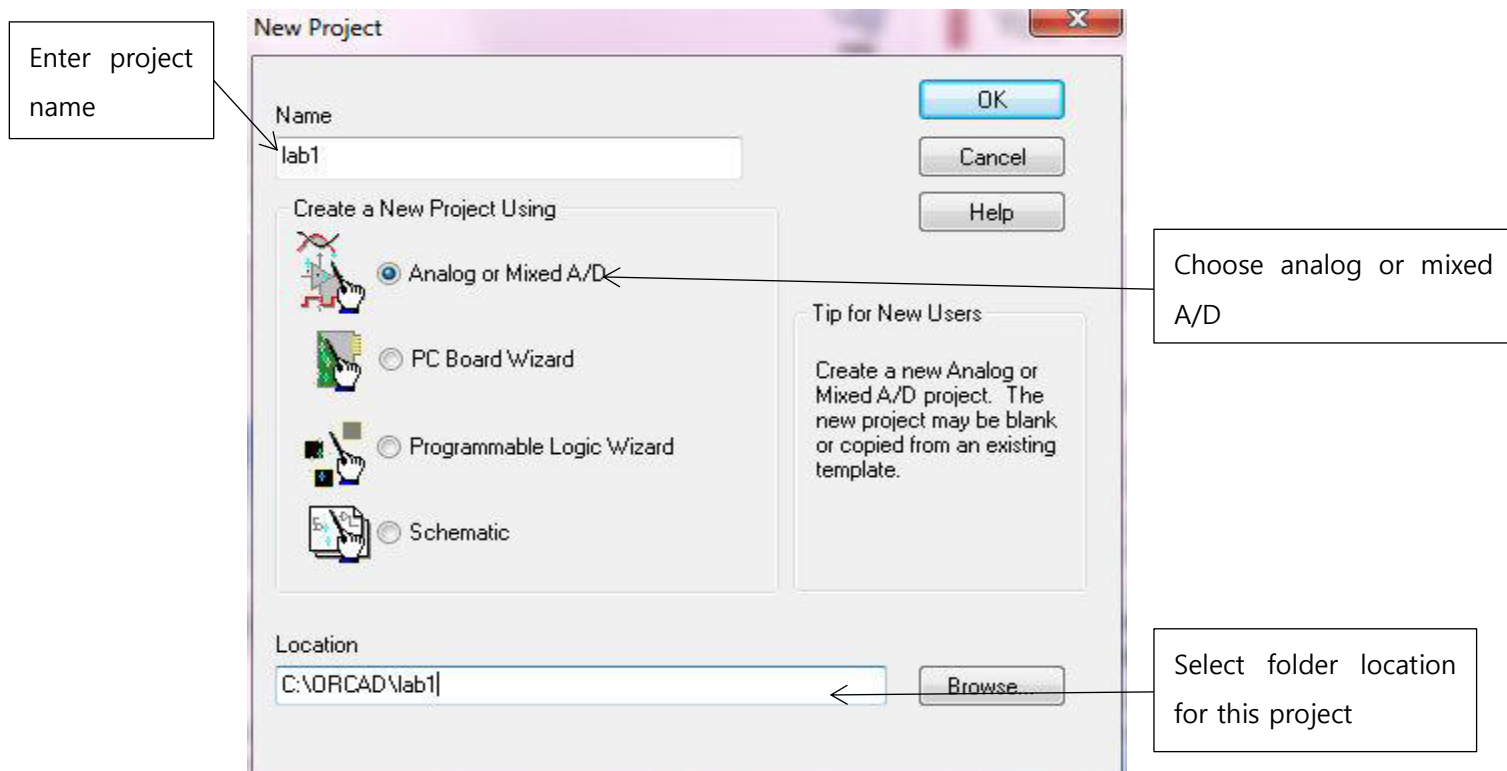
Proceed as follows to obtain the answer using PSpice.

1. Run the CAPTURE program.
2. Create a new project: Click the “Create document” icon on the toolbar or follow [File] → [New] → [Project] as shown in (Fig 1-3)



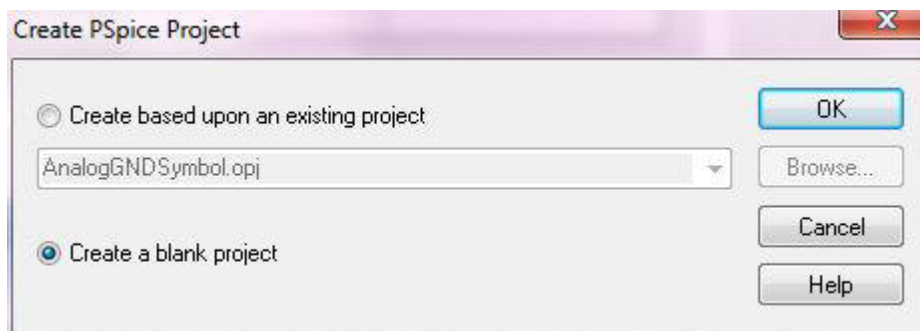
[Figure 1-3] Creating New Project

3. Enter a project name and a folder location to save your work and choose “Analog or Mixed A/D.” Then click “OK” as shown in (Fig 1-4).



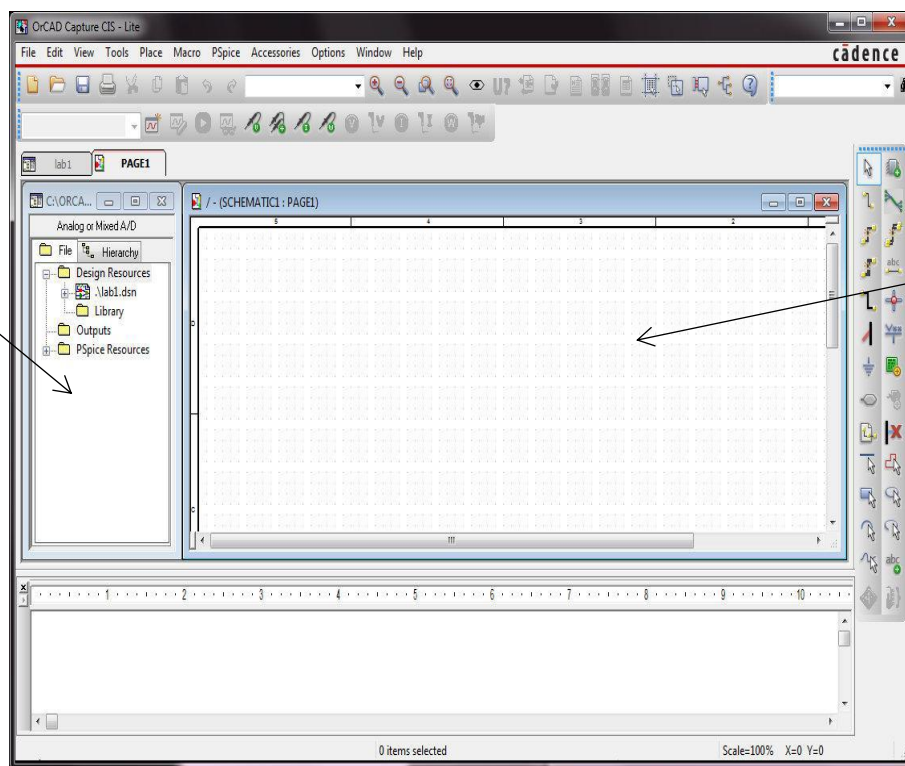
[Figure 1-4] Initiating New Project

4. Select “Create a blank project” and click “OK.”



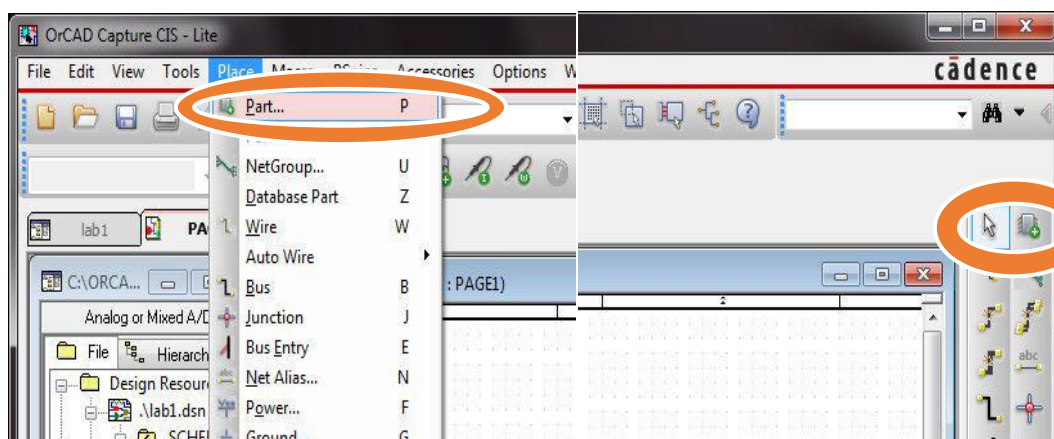
[Figure 1-5] Create Blank Project

5. You will see two main windows at this point: Project Manager window and Schematic window. You will draw the circuit in “Schematic window”. First you are going to make sure that all the libraries are added to your system. Click on Schematic window to activate it.



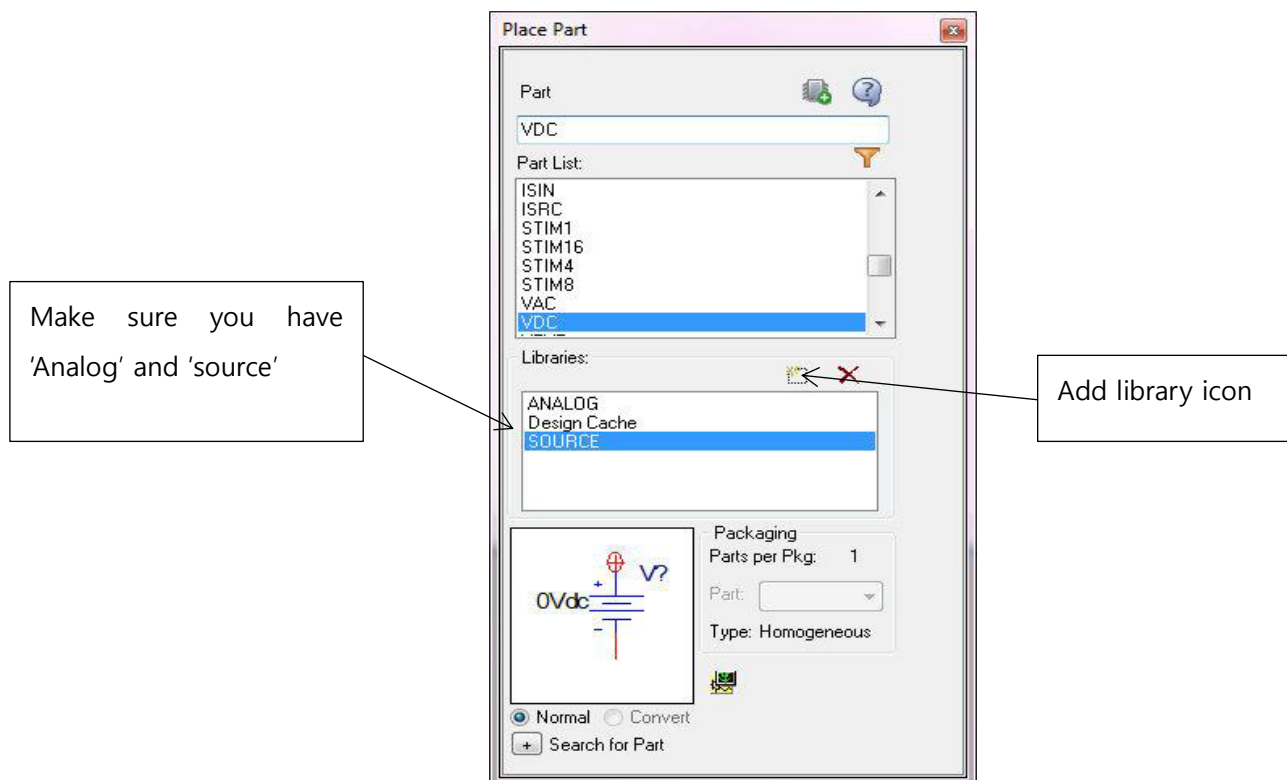
[Figure 1-6] Main Windows on The Software

6. Select “Place” → “Part...” or click on “Place part” icon in “Draw” toolbar or ribbon.



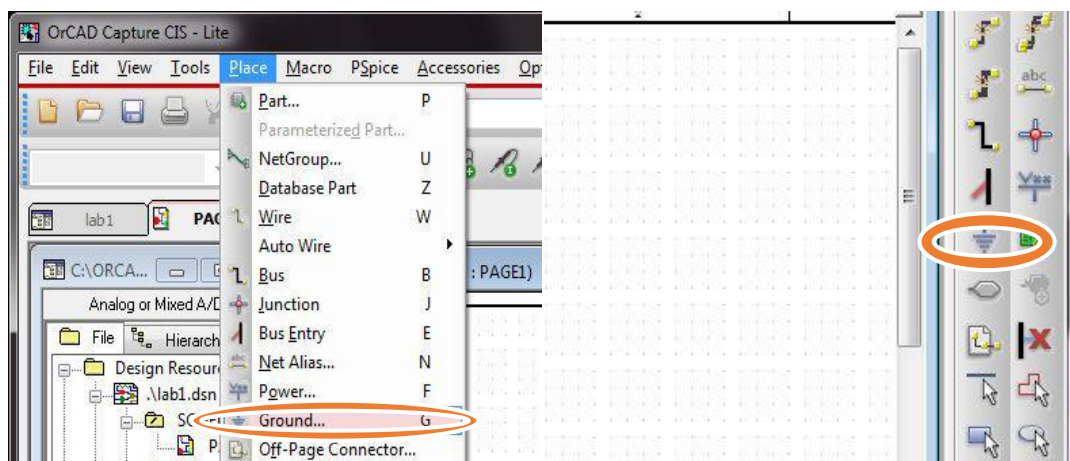
[Figure 1-7] Begin to Place Items and Draw Circuits

7. “Place Part” window will pop up somewhere in the main window. Make sure you see “ANALOG” and “SOURCE” in Libraries window. If not click on “Add library” icon, find “analog.olb” and “source.olb” under ../OrCAD_16.5_xxx/tools/capture/library/pspice/ and open the files to add them to the libraries to your system. If you cannot find any of those files, consult with your teaching assistant.



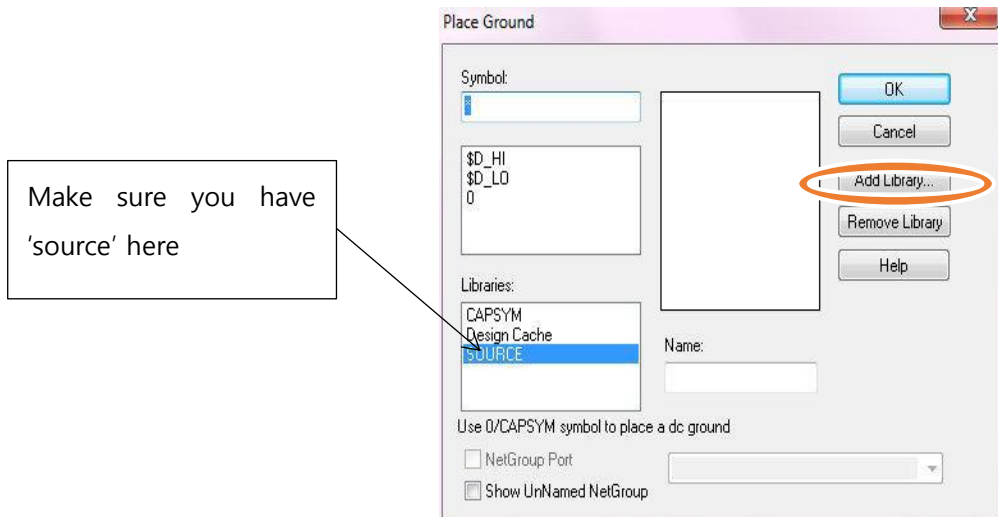
[Figure 1-8] Ensure Important Libraries

8. Select “Place”→“Ground...” or click on “Place ground” icon in “Draw” toolbar or ribbon.



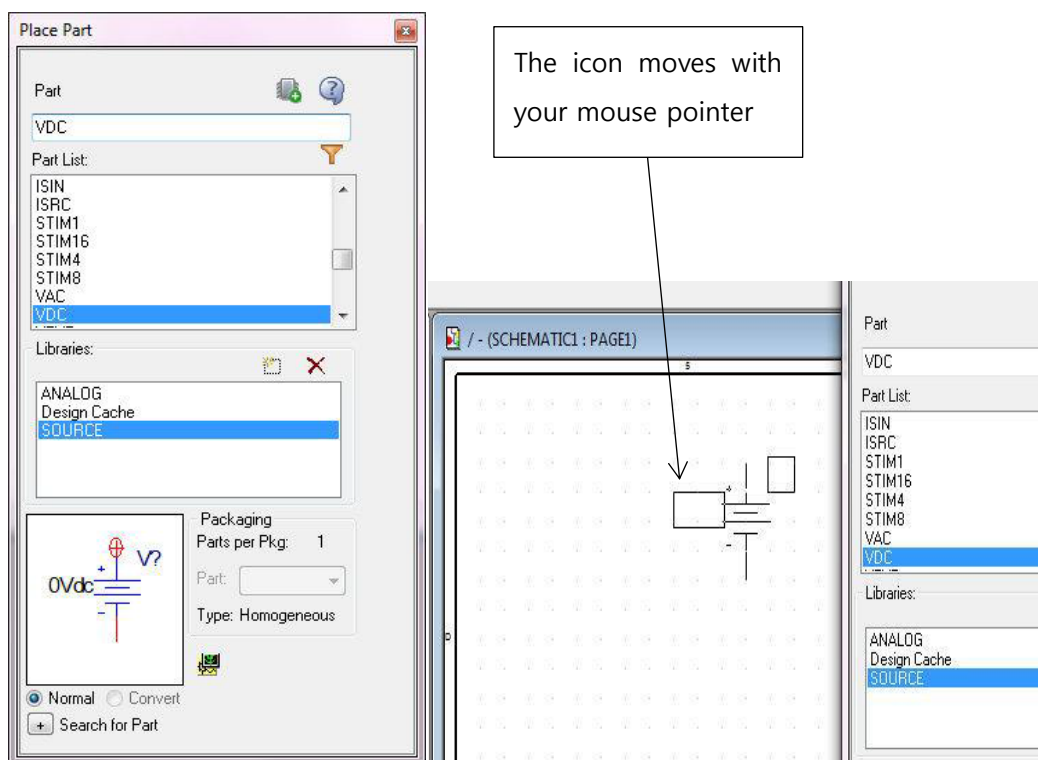
[Figure 1-9] Define Ground Item

9. “Place Ground” window will pop up somewhere in the main window. Make sure you see “SOURCE” in Libraries window. If not, find “source.olb” in the same location shown in step 3). Now we are ready to draw circuits for simulations.



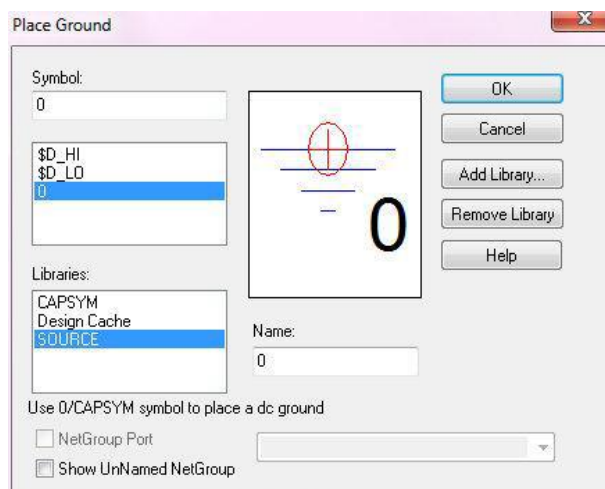
[Figure 1-10] Check Source Item

- Place DC voltage source: Open “Place Part” again. Click on “SOURCE” in Libraries window, find and double click on “VDC” in Part List. Then you will see a floating ”VDC” icon on “Schematic window”. Click once somewhere in the window and then the part will be placed. Then right click somewhere in the window then select “End Mode” or simply use ESC to terminate the mode.



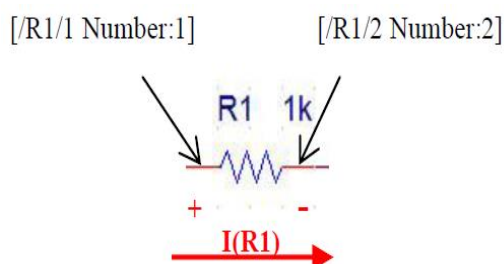
[Figure 1-11] Choose the Voltage Source

- Place a ground: Open “Place Ground” window (see step 7). Click on “SOURCE” in Libraries window, find and double click on “0” in Part List. Place it in “Schematic window” in the same manner as you did for “VDC”.



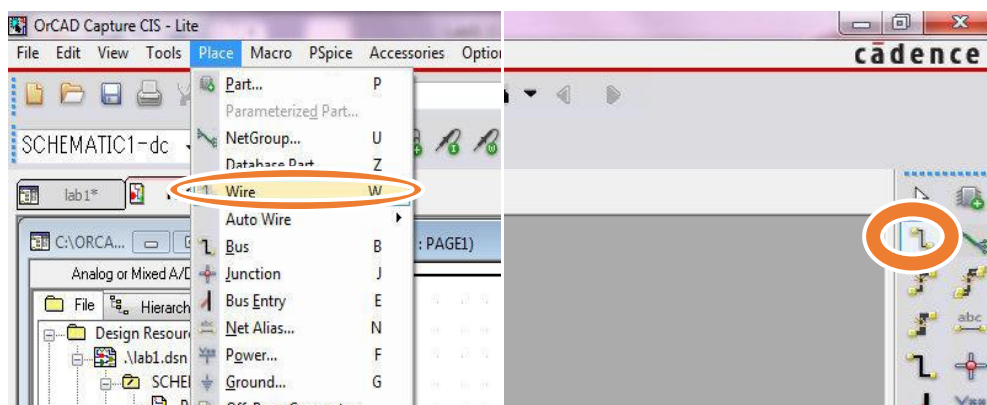
[Figure 1-12] Placing Ground Item

- Place resistors: Open “Place Part” again. Simple resistor “R” is in the library “ANALOG” Place the part in the same manner as you did for “VDC”. After placing the resistors, hover the mouse pointer on the left and right end of the resistors. You will see one end [/R1/1 Number:1] and the other end [/R1/2 Number:2]. PSpice recognizes the current through R1, I(R1) as shown below. I(R1) is the current entering R1 at node R1:1 (or R1/1) and leaving its node R1:2 (or R1/2). You will see the importance of the polarity of elements in PSpice simulations.



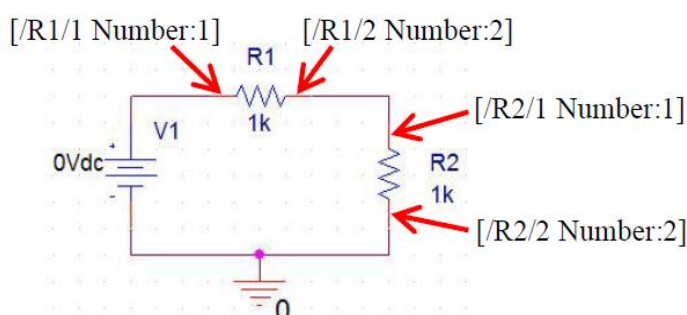
[Figure 1-13] Define Resistance Items

- Place wires: Select “Place” → “Wire...” or click on “Place wire” icon in “Draw” toolbar or ribbon. Then connect the circuits components.



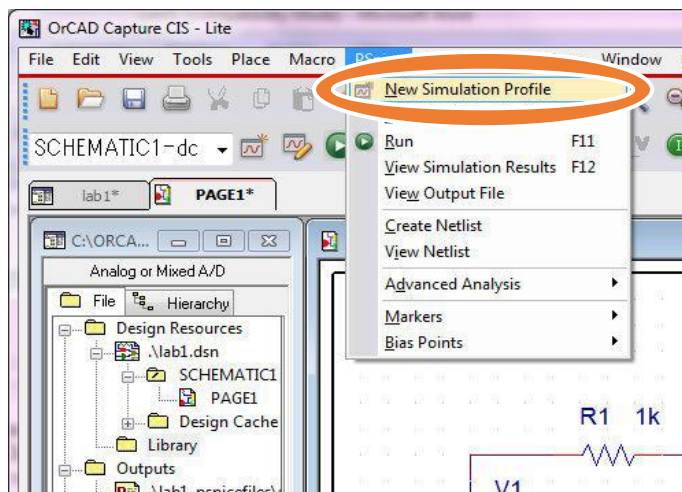
[Figure 1-14] Choose Wire to Connect Components

14. Using the wires connect the circuit components like shown below. Make sure the polarity of elements is in the same manner. Click on R1, R2 or V1 → It turns magenta → Right click on it → Bring up the menu besides your mouse pointer and you can modify the component. For now you are going to use “Mirror Horizontally”, “Mirror Vertically”, “Mirror Both” and “Rotate” to change it to the required direction in the circuit. Make sure the polarity of elements by repeating step 11).



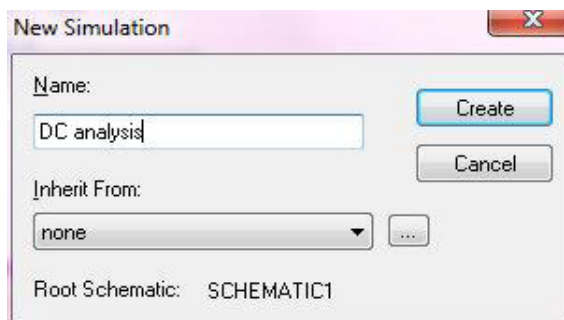
[Figure 1-15] Connect Whole Circuit

15. Now we are ready to run the simulation: Select “PSpice” → “New Simulation Profile”. “New Simulation” window will pop up.



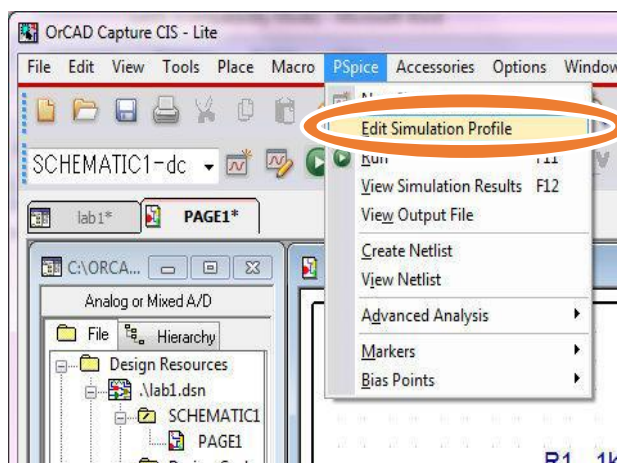
[Figure 1-16] Simulate the Project

16. “New Simulation” window will pop up. Give a name for the simulation and click on “Create”



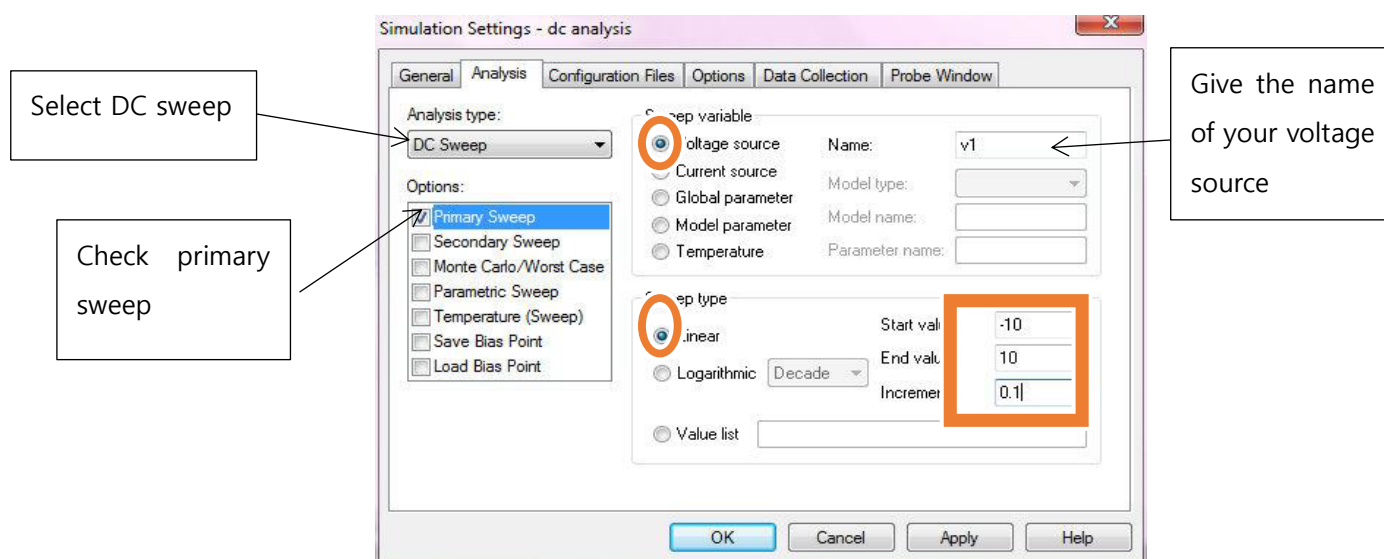
[Figure 1-17] Specify Simulation Technique

17. Select “PSpice” → “Edit Simulation Profile”.



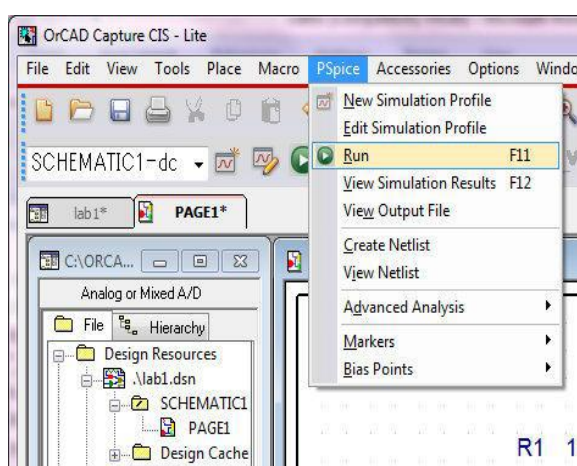
[Figure 1-18] Edit Simulation Profile

- You will see the new window “Simulation Settings - your simulation name”. Under “Analysis” tab, change “Analysis type” to “DC sweep” and make sure “Primary Sweep” is checked in “Options”. Select “Voltage source” in “Sweep variable” and give the exact name of your voltage source under “Name:”. In our case “V1”. Select “Linear” for “Sweep type” and make “Start value” -10, “End value” 10 and “Increment” 0.1. Then click “OK” to exit the window. The setting you made here means that simulator changes the voltage of V1 from -10V to 10V with the increment of 0.1V (-10V, -9.9V, -9.8V, ..., 10V) and simulate the voltages and currents of the circuit each time.



[Figure 1-19] Determine Simulation Settings

- Now you can run the simulation: “PSpice” → “Run”.



[Figure 1-20] Run Your Project

2. (Lab #2) Digital Multimeter for Resistance Measurement

Objective:

A primary purpose of this lab is for you to become familiar with the measurement of voltage, current, and resistance using the multimeters provided in the lab. Proficiency creating electrical circuits using resistors, wires, and power supplies. Another objective is to learn resistors color code

Introduction:

General Remarks All voltmeters, ammeters, and ohmmeters make measurements by means of a two-terminal connection to a circuit or circuit element under test. All voltmeters, ammeters, and ohmmeters have at least two jacks into which only two test probes are inserted. If there are more than two jacks on the face of a particular meter, there is usually an indication of some sort, next to each jack, to inform the user which jacks to use to measure a particular quantity.

Many times, a voltmeter, an ammeter, and an ohmmeter are combined into one package called a multimeter. Multimeters come in two flavors: analog and digital. Analog meters have a display that consists of a needle which points to a number scale. These meters have function and range controls which allow the user to select what kind of meter (voltmeter, ammeter, or ohmmeter) the multimeter will be and what range of values the meter will read. Digital multimeters typically have a LCD or LED display and in many cases are auto-ranging. That is, the meter will automatically select the most appropriate range for making the measurement. In this lab either an analog or a digital meter can be used with little difference in operation.

Voltmeter

A voltmeter measures electrical potential between its terminals. Voltmeters are always placed in parallel with the circuit or circuit element where the voltage measurement is desired. Since the voltage across two or more parallel elements is the same, the voltage measured by the meter will be the same as the element to which the meter is connected. When using a non-auto-ranging meter, select the highest possible range and reduce the range as necessary until the desired level of accuracy is reached. Always start with a range higher than the expected value to prevent damage to the meter.

Ammeter

An ammeter measures the current that flows between its terminals. An ammeter is always placed in series with the circuit or circuit element where the current flow is of interest. Since the current in each element of a series circuit is the same, the current flow through the

meter will be the same as the current flow to the element of interest. Never connect an ammeter in parallel unless you intend to measure the short circuit current of a circuit or circuit element and you have made sure that destructive current levels won't be reached. When using a non-auto-ranging meter, select the highest possible range and reduce the range as necessary until the desired level of accuracy is reached. Always start with a range higher than the expected value to prevent damage to the meter.

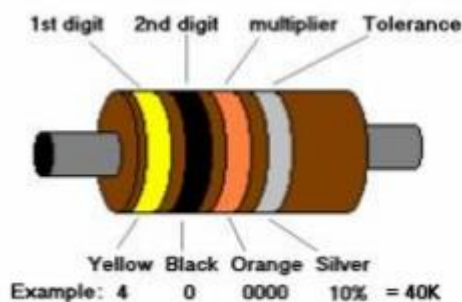
Ohmmeter

An ohmmeter measures the electrical resistance between its terminals. An ohmmeter is connected to the circuit or circuit element of interest after the element of interest has been isolated from the rest of the circuit. The element of interest has to be isolated from the rest of the circuit so that its resistance value isn't obscured by the resistance values of the other circuit components connected to the element of interest. Never connect an ohmmeter to an energized circuit or the meter could be destroyed. There is an additional caveat for an analog ohmmeter; it has to be zeroed every time the resistance range is changed. To zero the analog ohmmeter, touch its probes together (the needle will deflect to approximately full scale, use the "zeroing" knob to adjust the needle to read zero. The analog meter can now be used in this particular range.

Tasks for the (Lab #2): Reading Resistance & Measuring by DMM

1) READING RESISTOR COLOR CODES

The values of most of the resistors in your lab kit are indicated by color bands. These color bands are painted on the body of each resistor and are decoded in the following manner: Look at the resistor. There will be a group of color bands clumped together at one end and a lone band at the other. Orient the resistor so the lone band points to the right and the clumped bands point to the left. The resistor color code is typically using 4 color bands as shown in fig (2-1). The resistor can be read left to right using the guidelines listed below.



[Figure 2-1] Simple Resistor

1. The first color band indicates the first digit in the numerical value of the resistance.
2. The second color band gives the second digit in the numerical value of the resistance. Any subsequent bands before the last band in the “clump” give subsequent digits in the numerical resistance value.
3. The last color band in the “clump” gives the number of zeros that follow the first digits.
4. The “lone” color band on the right end gives the tolerance.

Note: Tolerance is the amount by which the actual resistance can be different from the color-coded value and it is given in percent. For example, a 1000 Ω resistor with a +10% tolerance can have an actual resistance value between 900 Ω and 1100 Ω.

The color codes for digits and tolerance are given in Table (2-1).

[Table 2-1] Units and its values

Color	Digit	Multiplier	Tolerance
Black	0	1	
Brown	1	10	±1%
Red	2	100	±2%
Orange	3	1,000	
Yellow	4	10,000	
Green	5	100,000	±0.5%
Blue	6	1,000,000	±0.25%
Violet	7	10,000,000	±0.1%
Gray	8		
White	9		
Gold		0.1	±5%
Silver		0.01	±10%
None			±20%

It is important to note that the physical size of the resistor indicates its power dissipation rating, not its ohmic value.

2) MEASURING RESISTOR BY DMM

Measurement of resistors with a DMM is a very straightforward process. Simply set the DMM to the resistance function and choose the first scale that is higher than the expected value. Clip the leads to the resistor and record the resulting value.

Procedure

1. Given the nominal values and tolerance in table (2-2), determine and record the corresponding color code bands.
2. Given the color code from table (2-3), determine and record the nominal value, tolerance and the minimum and maximum acceptable values.
3. Obtain the resistor equal to the first value listed in table (2-4), determine the minimum and maximum acceptable values based on the nominal value and tolerance. Record these values in table (2-4). Determine the deviation percentage of this component and record it in table (2-4). The deviation percentage may be found via:

$$\text{Deviation} = 100 * (\text{measured} - \text{nominal}) / \text{nominal}.$$
4. Repeat step 3 for the remaining resistors in table (2-4).

[Table 2-2] Nominal value of resistors Vs color bands

Value	Band 1	Band 2	Band 3	Band 4
27@10%				
56@10%				
180@5%				
390@10%				
680@5%				
1.5k@20%				
3.6k@10%				
7.5k@5%				

10k@5%				
47k@10%				
820k@10%				
2.2M@20%				

[Table 2-3] Reading color code resistors

Colors	Nominal	Tolerance	Minimum	Maximum
Red-Red-Black-Silver				
Blue-Gray-Black-Gold				
Brown-Green-Brown-Gold				
Orange-Orange-Brown-Gold				
Green- Blue-Brown- Gold				
Brown-Red-Red-Silver				
Red-Violet-Red- Silver				

Gray-Red-Red-Gold				
Brown-Black-Orange-Gold				
Orange-Orange-Orange-Silver				
Blue-Gray-Yellow-None				
Green-Black-Green-Silver				

[Table 2-4] Nominal value of resistors Vs the Deviation Value

Value	Minimum	Maximum	Measured	Deviation
22@10%				
68@5%				
150@5%				
330@10%				
560@5%				
1.2k@5%				
2.7k@10%				

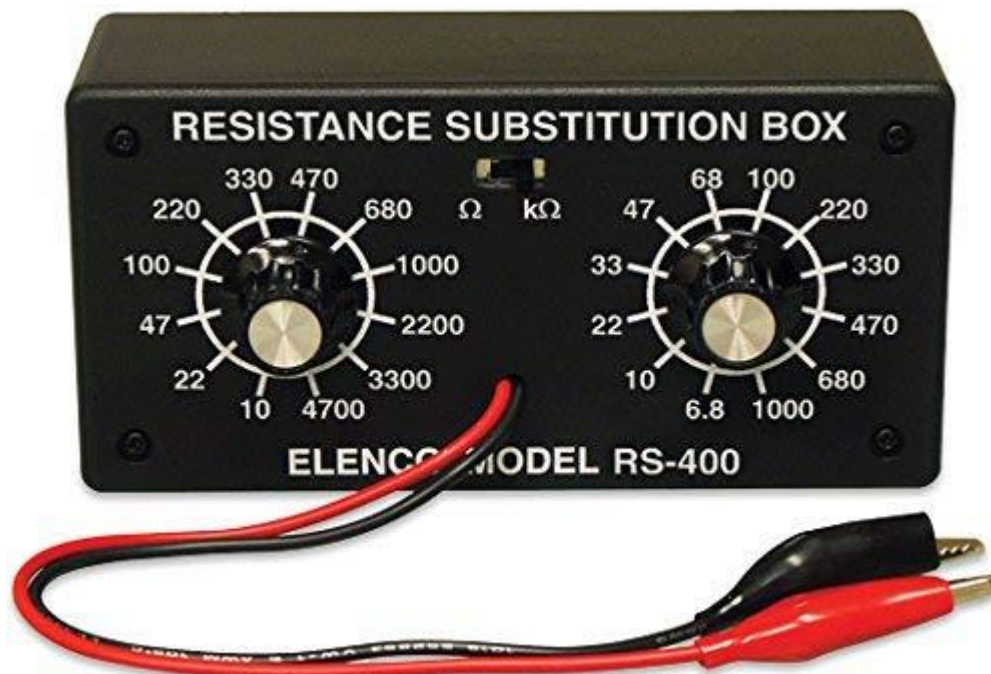
8.2k@5%				
10k@5%				
33k@10%				
680k@10%				
5M@20%				

- Obtain about 15 resistors of the 15KΩ value, measure the resistance of each resistor using the DMM range that will give the maximum number of significant figures. Draw a distribution graph with the vertical axis being the number of resistors within 1% range and the horizontal axis being the percent deviation from the nominal resistance value. Make your table for all 15-resistor same as table (2-5)

[Table 2-5] Resistance measurement

	Coded Resistance	Tolerance Value	Measured Resistance	Within Tolerance (Y/N)
1				
2				
3				
4				

6. Obtain a Resistance Substitution Box as seen in fig(2-2) and use a DMM to measure the resistance values for all the resistance settings to the maximum number of significant figures. Compare the actual values to the nominal values in terms of the percent difference.



[Figure 2-2] Resistance Substitution Box

7. Connect short lead or jumper wire using your red and black wire between the DMM input terminals and read the resistance, using the lowest resistance range to obtain the maximum number of significant figures. Now connect long lead wire to the DMM and repeat the measurement. Calculate the approximate values of resistance per meter of the lead wire. Use the DMM NULL for an easier measurement.
8. Variable Resistors: Obtain a 5000 Ω variable resistor (potentiometer or rheostat).

The potentiometer can be used as a voltage divider to obtain a manually adjustable output voltage at the slider (wiper) from a fixed input voltage applied across the two ends of the potentiometer as seen in figure (2-3). This is their most common use



[Figure 2-3] Potentiometer

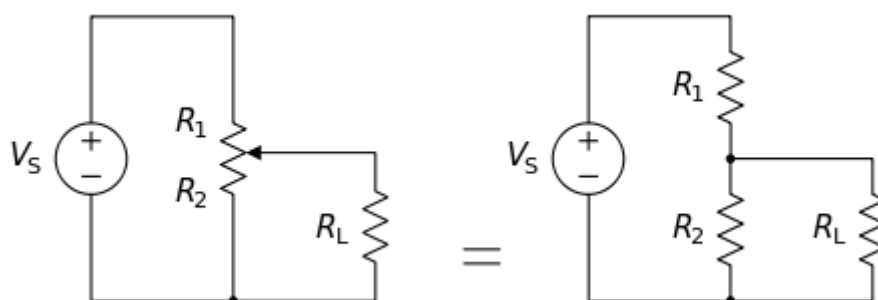
The voltage across R_L can be calculated by:

$$V_L = \frac{R_2 R_L}{R_1 R_L + R_2 R_L + R_1 R_2} \cdot V_S \quad (2-1)$$

If R_L is large compared to the other resistances (like the input to an operational amplifier), the output voltage can be approximated by the simpler equation:

$$V_L = \frac{R_2}{R_1 + R_2} \cdot V_S \quad (2-2)$$

(dividing throughout by R_L and cancelling terms with R_L as dominator.)



[Figure 2-4] Connecting potentiometer

- a. Measure the total (i.e. end-to-end) resistance.
 - b. Measure the resistance between the “wiper arm” and one end with the knob in the fully clockwise position, and then with the knob in the fully counter-clockwise position, then repeat this for the resistance between the wiper arm and the other end of the resistor.
9. In some cases, especially when using the DMM to measure resistance value associated with various electronic devices, it is important to know the voltage

produced by the DMM when measuring resistance. Connect the DMM to a resistance substitution box and for every DMM resistance range adjust the resistance to give a reading as close to the full- scale resistance reading as possible. Connect a second DMM set to read voltage across (in parallel with) the first DMM and reading the test voltage of the first DMM.

3) QUESTION TO BE REPLIED

Q1: What is the largest deviation in Table (2-4)? Would it ever be possible to find a value that is outside the stated tolerance? Why or why not?

Q2: if steps 3 and 4 were to be repeated with another batch resistors, would the final two columns be identical to the original table (2-4)? Why or why not?

Q3: Do the measured values in table (2-4) represent the exact values of the resistors tested? Why or why not?