An Intelligent Circuit Analysis Module to analyze student queries in the Universal Virtual Laboratory

Annapoorna Mahalingam, Brian P.Butz, and Michael Duarte
Intelligent Systems Application center, Electrical and Computer Engineering Department,
Temple University, Philadelphia PA, 19122

***Abstract* - This paper presents an intelligent circuit analysis module that is capable of analyzing electrical circuits and determining equivalence. We discuss the module in context of a National Science Foundation funded project - Universal Virtual laboratory (UVL). UVL is a virtual electrical engineering laboratory for able and disabled individuals to construct, simulate and understand the characteristics of basic electrical circuits. This paper discusses the development of an intelligent circuit analysis module called the Circuit Recognizer (CR). The CR is a program that has knowledge of circuit theory concepts, and is capable of using this knowledge to provide assistance to a student while he/she is performing the experiments in UVL. While verifying circuit configuration it identifies errors and provides explanations that guide the student to a better understanding of the experiment. The ultimate objective of the CR is to evaluate the effectiveness of the teaching strategies employed within the UVL as well as to enhance the learning abilities of students.**

Index Terms – Circuit analysis, intelligent tutoring system, and virtual laboratory

INTRODUCTION

According to the Center for Disease Control [1] there are 13.8 million individuals who have physical mobility limitations. In the field of science and engineering, there are approximately 195,000 persons with motor disabilities employed in the United States [2]. Also, approximately 31,300 students with motor disabilities were registered in science and engineering programs in 1995 [3].

The Universal Virtual Laboratory (UVL) [4] is an environment that emulates a real electrical engineering laboratory, offering the user (a disabled or physically able individual) a way to learn the different aspects of instrumentation and circuitry. UVL achieves this by integrating several existing Windows® applications within the laboratory architecture. Several virtual laboratory instruments are available to the student including a power-supply, a function generator, an oscilloscope, a spectrum analyzer and a digital multimeter. The instruments are connected to the holes on the breadboard when the user clicks on the instrument or verbally enters the coordinates of the hole where the component is to be placed. The components in the UVL include resistors, capacitors, inductors, potentiometers, diodes, transistors, switches and jumper wires. The coordinates are the letters (vertical) and numbers (horizontal) seen on the breadboard (see Figure 1).

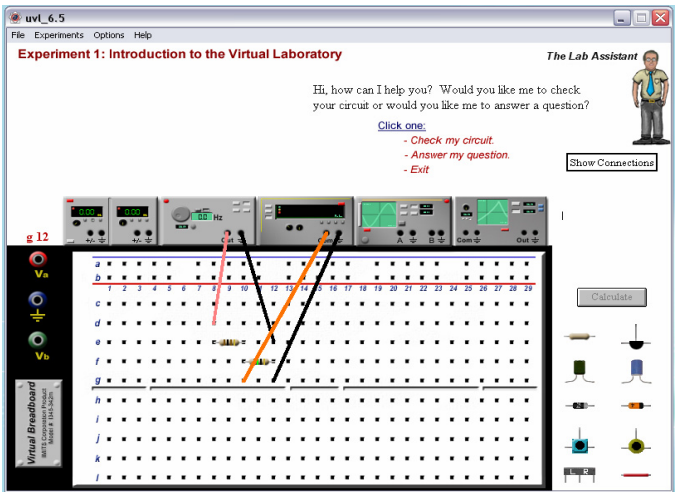

FIGURE 1
THE UNIVERSAL VIRTUAL LABORATORY

The user enters the letter and number corresponding to coordinate of the hole on the breadboard where he/she wishes to place a wire or component. After the circuit has been built, the user clicks on the "Calculate" button to simulate the circuit behavior. The calculate button activates the circuit analysis program 'PSpice' which validates the user built circuit. If the circuit is connected properly (as per PSpice requirements), the output devices (multimeter/oscilloscope/spectrum analyzer) display the result.

THE CIRCUIT RECOGNIZER

Students may work on a number of experiments and projects within the UVL. Each designed experiment includes instructions to perform the experiment as well as the necessary circuit schematics. The student is given the freedom to build the circuit in any way that he/she wishes, adhering however to the "essence" of the schematic provided in the laboratory experiment. Therefore, when a student breadboards a circuit in UVL, the circuit may be connected in a manner physically different from the connections shown in the circuit schematic; and yet the circuits might be equivalent. Allowing the student the freedom to deviate from the schematic may result in students requiring assistance as they proceed with the experiment. This necessitates the creation of a module that will

assist students if they encountered a problem while performing the experiment. The Circuit Recognizer (CR) is a program developed for UVL that has knowledge of circuit theory concepts, and is capable of using this knowledge to assist the student and evaluate the student's work. The objective of the CR is to verify if the circuit that is created by the student is a valid representation of the laboratory manual schematic.

If a student wishes to confirm that the circuit created on the breadboard is equivalent to the schematic in the manual, they can seek assistance from the CR by hitting the *"Check my circuit"* button provided in the UVL interface. For the CR to analyze the student's circuit, it needs to know what laboratory manual schematic the student is using. After this information is received from the student, the CR compares the student's circuit with the appropriate laboratory manual schematic. All analysis performed by the CR is recorded in a file that is accessed and analyzed by Macromedia's Authorware , an application program for creating multimedia so that the information may be passed on to the user.

Thus the Circuit Recognizer's purpose is to increase the effectiveness of UVL as an intelligent tutoring system. It is an intelligent circuit analysis module incorporated in UVL to analyze student work. In the process of analyzing student work it identifies errors and provides explanations that guide the student to the correct answers.

IMPLEMENTATION

The Circuit Recognizer uses an expert model - student model comparison approach to circuit analysis. The laboratory manual schematic serves as the expert model, which represents the way in which we would like the student to assemble a circuit. The student's approach to circuit construction constitutes the student model. The two models are compared and the resultant disparity is used to identify equivalence or errors for each user. Based on this comparison of the student and expert models, the CR evaluates the student's performance and offers suggestions for improvement.

The objective of the CR is to function as a circuit configuration comparator, and determine if the student created circuit on the breadboard is completely topologically equivalent to the experimental schematic. Electrical circuit networks are said to be *completely topologically equivalent* if the circuits contain the same set of components in type and value, and the manner in which the components are connected results in the same circuit solution (branch currents and branch voltages). The CR uses the following rationale to determine circuit equivalence. If the components in the student's circuit are identical in value and type as the laboratory manual schematic, AND the topologies of the two circuits are equivalent; the CR concludes that the circuits are equivalent.

If there are any missing or additional components other than those indicated in the laboratory schematic, the CR records them in the result file and displays it to the user. The CR discontinues any further analysis until only the required components are present. Once the user has removed/added the components indicated by the CR, the topologies are analyzed. If the topologies of the two circuits are the same the CR identifies the student's circuit to be a valid representation of the laboratory manual schematic. If the topologies are different then the CR identifies the differences between the circuits and communicates that to the user through the result file. If there are any differences the CR determines that the student created circuit is not completely topologically equivalent to the experimental schematic.

The development of the circuit recognizer is broadly divided in two major portions. The first part involves the development of the component check module and the other is the development of the topology check module. The following sections describe the various modules in the CR and their role in determining circuit equivalence. A flowchart of the sequence of operations involved in the circuit analysis and determination of circuit equivalence is shown in Figure 2.

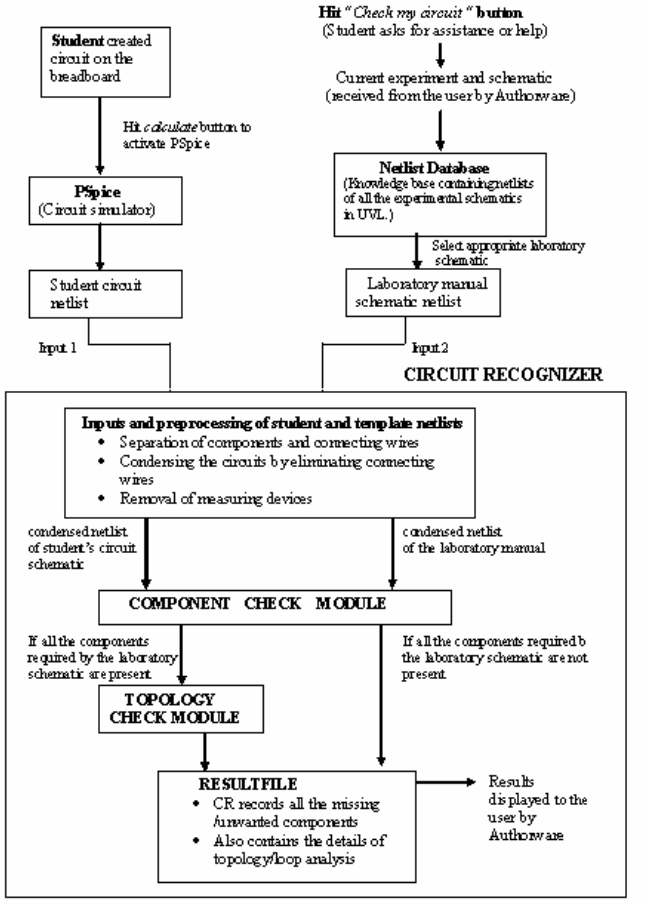

FIGURE2
SEQUENCE OF OPERATIONS PERFORMED BY THE CIRCUIT RECOGNIZER

INPUTS AND PREPROCESSING

Circuits are created, analyzed and their results displayed in UVL by interacting with Authorware, LabVIEW (a trademark of National Instruments) and PSpice application software

integrated within it. Once the layout of the circuit on the breadboard is completed, the student may hit the "calculate" button provided in the UVL user interface to simulate the circuit. Each time the student clicks on the "calculate" button, PSpice is activated in the background. PSpice is told by Authorware how each circuit element is connected within the circuit, which allows PSpice to generate a file called the netlist. The netlist contains information about each component (type and values), connecting wires and measuring devices used in the circuit along with the nodes to which they are connected. This circuit netlist is one input to the circuit recognizer. In order to analyze the equivalence of two circuits, the circuit recognizer also requires the PSpice generated netlist of the circuit shown as a schematic in the laboratory manual.

Once both PSpice netlists have been obtained the CR begins its analysis. The first step involves preliminary analysis and preprocessing. The circuits are condensed by removing all the connecting wires and measuring instruments that have been connected to the circuit. The circuit recognizer begins by seperating the circuit components from the connecting wires in each circuit. Information about the circuit components and the connecting wires are stored in separate files Preprocessing routines examine the wire file and eliminate the wires from the circuit one at a time. The wire removal routine equates the two nodes across which a wire is connected, thereby reducing the wire to a single node/point. If a wire is connected in parallel across a component, the component is removed from the circuit as it amounts to a *short circuit*. The CR maintains a log of the components that are eliminated and the wires that shorted them out. If the elimination of wires results in an isolated node (called a hanging node) then the circuit is detected as *open*. The nodes that are hanging are noted. In case of an *open circuit* or s*hort circuit* the circuit recognizer stops analyzing the circuit and communicates the error to the user.

Once the connecting wires are eliminated from the circuit, the measuring devices are removed. Separate modules determine if a measuring device is connected in series or parallel with a component. If a device is used in parallel across a component it is simply removed from the circuit. If the device is used in series with a component, the device is first removed. A wire is then used to connect the nodes to which the device was connected. Subsequently the wire is removed by wire elimination routines. From the resultant condensed circuits (netlists) the nodes are identified. The circuits are now ready to be analyzed for equivalence.

COMPONENT CHECK MODULE

After preprocessing, the CR proceeds to analyze the circuit in the following two steps. The component check module verifies if all the components required by the laboratory manual schematic are present in the circuit created by the student. The component check module begins by parsing the netlists of the student created circuit and template circuits (the circuit netlist resulting from the circuit schematic shown in the laboratory experiment) and grouping components based on their type. The various types of components in UVL are resistors, capacitors, inductors, potentiometers and voltage sources. Components of a particular type are gathered and grouped in arrays. For example, all the resistors in the template netlist are placed in an array called the template_resistor_array and all the resistors used by the student are placed in an array called the student_resistor_array. Similarly the capacitors in student's and template netlists are placed in the student_capacitor_array and template_capacitor_array respectively. Once this is done for all the types of components present in both the netlists, each array is sorted numerically by their component values using a selection-sorting algorithm. The selection-sorting algorithm is a universally accepted method used to sort data. It makes repeated passes through the array (vector) from top to bottom, and after each pass the largest value in the vector is swapped with the element at the bottom. After each swap the bottom of the vector is moved up a position now known as the false bottom. The procedure continues till the false bottom moves right to the top or end of the vector. Corresponding component arrays of the student and template netlists are then compared for order and individual values. If any disparity is detected, the missing or additional component along with its type is noted down in the result file to be communicated to the student when the analysis is complete.

Certain experiments in UVL require students to change the values of some components and study the corresponding changes in circuit behavior. Such experiments have a tabular column associated with the experimental schematic to document the changes in the circuit behavior as the component values change. For example, Experiment 3 on Ohms Law requires the student to change the values of circuit voltage and note the corresponding changes in circuit current keeping the resistance constant. For such circuits, the component check module allows the student to use the range of component values indicated in the tabular column. If the student uses values of components other than those specified in the schematic or tabular column the CR identifies that as an error.

If the component check module is successful in determining that the student assembled circuit contains all the components required by the experimental schematic, the circuit recognizer proceeds to perform a topological analysis. If however the student's circuit has any missing components or additional unwanted components (components other than those indicated by the laboratory schematic), the circuit recognizer discontinues any further analysis. The errors are noted and communicated to the student. The student may then rectify the errors using the explanations provided and then call the CR once again (by hitting the "check my circuit" button) to perform a check.

TOPOLOGICAL EQUIVALENCE MODULE

The CR defines equivalence based on circuit parameters such as branch voltages and branch currents. Completely topologically equivalent circuits, though physically different

have the same circuit current through and same voltage drop across corresponding components or branches. The branch currents and voltages can be obtained through PSpice, and then compared to those resulting from an analysis of the template netlist to determine circuit equivalence. Rather than just measuring the voltage and current through the components, the CR relies on a more comprehensive approach to topological analysis. The approach used extracts more information from the circuit than just the circuit parameters. This gives us a better insight into the level of understanding of the subject matter, and the areas of weakness of the student. This information can then be used to modify the teaching strategy for each student and thereby tutor the student more efficiently.

A circuit is viewed as a *graph* [5] consisting of nodes and branches. The nodes of the graph are connected by edges (circuit elements in our case). One of the basic laws in circuit theory Kirchoff's Voltage Law (KVL), states that the net voltage drop across various components in a circuit loop is zero. (A path that can be traced through a circuit starting from a node and ending at the same node without passing through a component more than once is called a circuit loop.) It is known that the corresponding branch currents and voltages should be identical in completely topologically equivalent circuits. From this, it follows that the loop currents and voltages should also be the same. If the fundamental loops (an independent circuit loop formed by a unique path through a chosen tree T along with each link of the cotree Tc of tree T) for both circuits under consideration are identified and the nature and orientation of components in corresponding fundamental loops are determined to be the same, then both circuits will have identical loop characteristics and thereby yield identical circuit solutions. In essence, the results of loop analysis for equivalent circuits should be the same.

The CR begins topological analysis by obtaining information about the fundamental loops in each of the circuit. The *fundamental loop matrices* (that contain information about all the independent circuit loops and the components in each loop) are obtained for each circuit. Next the fundamental loop matrices are compared to determine if the circuits are topologically the same.

The fundamental loop matrix compactly represents the KVL equations associated with a circuit. From the CR point of view, if the fundamental loop matrices of the schematic and the student's circuit are the same; it would mean that the set of KVL equations generated by these matrices would also be the same. If b equals the number of branches and n equals the number of nodes, then (b-n+1) equations are needed to solve the circuit. Once the (b-n+1) independent KVL equations are solved the circuit solutions would be identical, thus making the circuits' equivalent. The following sections describe the various processes involved in the generation and analysis of the fundamental loop matrices. The CR obtains the fundamental loop matrices by performing mathematical operations on the basic incidence matrix (BIM).

BASIC INCIDENCE MATRIX

An electric circuit consisting of Nb branches and Nn nodes can be represented by a node branch matrix of order Nb * Nn (called the Augmented Matrix) where each row of the matrix represents a node and each column represents an edge or branch. The matrix consists of 1, -1 or 0 according to the convention,

+1 if the reference current of the branch is leaving the node.
-1 if the reference current of the branch is entering the node.
0 if the reference current does not interact with the node i.e. if the branch is not connected to the node.

If one of the dependant rows (representing the reference node) is deleted from the Augmented Matrix, the matrix so formed is called the Basic Incidence Matr*ix* (BIM). The BIM is a matrix of rank (Nn-1), which represents the node component incidence pattern of a network.

FUNDAMENTAL LOOP MATRICES

For a directed graph Gd, with b branches and ln oriented loops the loop matrix Ba is defined as an (ln * b) matrix where

$$Ba = [\ bij\ ] \qquad (1)$$

and

bij = 1 if branch j is in the loop i and their directions agree
bij = -1 if branch j is in the loop i and their directions oppose
bij = 0 if branch j is not in the loop i .

KVL states that the algebraic sum of the voltages around any loop of a lumped network is zero at all times. If the branch voltages are represented by a (b*1) column vector v(t) such that the rows of v are in the same branch order as the columns of Ba then KVL, when applied to all the loops, may be expressed as

$$Ba*v=0 \qquad (2)$$

This generates a set of ln equations. However, in network analysis all these ln equations are not required, but only the ones that are independent. This leads us to a more useful submatrix of Ba that consists of the maximum number of independent rows of Ba called the *fundamental loop matrix* denoted by Bb. It can be shown that for a directed graph Gd with n nodes and b branches there are (b-n+1) loops, i.e. Bb has (b-n+1) rows [5]. Thus the (b-n+1) independent KVL equations may be expressed as

$$Bb * v = 0 \qquad (3)$$

A systematic method of constructing the fundamental loop matrix is through the aid of a tree T. A network tree is a sub graph connecting all the network nodes but having no closed paths. Each link (an element when added to a network tree forms a closed path) of a co-tree Tc along with the unique path through the tree T forms a loop, which is the fundamental loop for that link with respect to the chosen tree T. For a directed

graph Gd there are (b-n+1) links and hence (b-n+1) fundamental loops. The rows of the fundamental loop matrix represent these (b-n+1) fundamental loops. The concept of a tree is instrumental in the formulation of the fundamental loop matrix and consequently the independent KVL equations.

FORMULATION OF THE FUNDAMENTAL LOOP MATRICES

The tree T that is used to generate Matrix Bb must be chosen with some preference as to the order of different types of network elements included in the tree. The CR uses the following order of preferred element types: independent voltage sources, controlled voltage sources, capacitances, resistances and inductances. In other words this particular tree T should contain all the voltage sources as tree branches, all the current sources as links, as many capacitors as tree branches as possible and as many inductors as links as possible. Our task is now twofold.

1. To find the set of trees Ti with the above preference of network element types.
2. To find the fundamental loop matrices relative to the chosen tree set Ti.

I. Finding the Tree set Ti

The CR uses the following procedure to pick out the required tree from the circuit using the basic incidence matrix Ba [6].

(i) The columns of the incidence matrix are rearranged so that the branches (columns) are in the following order: independent voltage sources, controlled voltage sources, capacitances, resistances and inductances. This rearrangement of columns of the BIM assures that all the voltage sources and as many capacitors as possible are contained in the desired tree.

(ii) Row operations are now performed on the rearranged basic incidence matrix to reduce it to the form:

$$[I \mid C] * i = 0 \quad (4)$$

$$[-I \mid C] * i = 0 \quad (5)$$

Where

[I] is the $(N_n - 1) * (N_n - 1)$ Identity matrix
[C] is a $[N_n - 1] * [N_b - (N_n - 1)]$ matrix.

(iii) The columns of the [C] Matrix of equation (4) and (5) are then rearranged so that the final form of the columns of the matrix has the following order: dependent voltage sources, independent voltage sources, tree capacitors, tree inductors, tree resistors, link resistors, link capacitors, link inductors, independent current sources, dependent current sources.
The columns of the identity matrix now correspond to the tree of the above-preferred order. This particular tree is called a proper tree of the network. The fundamental loop matrix relative to this tree can be computed using a specific algorithm (described in Chua [6], Pg: 147) shown below.

II. Generation of Loop Matrices relative to Tree set Ti

(i) The BIM is partitioned as

$$A = [At \mid Al] \quad (6)$$

The sub-matrix of A which corresponds to the proper tree branches At is obtained.The inverse of At (Atinv) is then computed.

(ii) The product of matrices Atinv and A is computed and the result is assigned to a new matrix Dn

$$Dn = Atinv * A \quad (7)$$

Where
Dn is a matrix of order (n –1)* b
n represents the number of nodes
b is the number of branches/elements in the circuit.

(iii) Matrix Dn is partitioned

$$Dn = [Dt \mid Dl] \quad (8)$$

Where
Dt is a matrix of order (n-1) whose columns represent the tree branches
Dl is a matrix of order (n-1)*(b-n+1) whose columns correspond to the tree links

(iv) Matrix Dl (The sub-matrix of Dn which corresponds to the links of the tree T), is obtained. We then compute matrix Dt such that

$$Dt = \text{transpose}(Dl) \quad (9)$$

The negative of the transpose matrix Dt is then computed.

(v) Fundamental loop matrix Bb relative to tree T is then obtained by the equation

$$Bb = [-Dt \mid I] \quad (10)$$

Where
Dt is a matrix of order of (b-n+1)* (n-1)
I is an identity matrix of order (b-n+1)* (b-n+1)
Each of the (b-n+1) rows of Bb corresponds to the fundamental loops of the network relative to tree T. Once the fundamental loop matrices of the student's and template circuits are obtained they are compared to determine equivalence.

OBJECTIVE BASED TOPOLOGICAL ANALYSIS

In some circuits in UVL, it may happen that even if the student's circuit is completely topologically equivalent to the template, the student's answers might be incorrect to certain questions. For example Figure 3 a circuit schematic where the objective is to measure the thevenin's resistance across nodes 2 and 5. The student-assembled circuit is illustrated in Figure 4. It can be seen that both circuits are topologically equivalent. But the thevenin's resistance measured across nodes 2 and 5

are different in each case due to the rearrangement of the 10 and 15 ohm resistors in the student's circuit. Even though CR identifies the circuits to be equivalent, the student does not arrive at the correct answer for the thevenin's resistance. In such situations it may be that the objective of the current task is not well understood by the student. In other words, the student is not able to satisfy the objective of the experiment because of how he/she has changed the physical connections of the circuit components. This might lead the student to use the measuring devices incorrectly thereby leading them to the wrong answers. Such circuits, where it is possible for the student to hook up the circuit correctly but yet have errors in the answers, is handled differently by the CR. The CR knows the parts of experiments that require more than just topological analysis. The objective of the experiment as well as the topology of the circuit must both be addressed by the CR. This action constitutes "objective based topological analysis".

If a student is working on an experimental schematic that requires objective based topological analysis, measuring devices that are usually not considered as circuit components are now seen as circuit elements. If they are considered as circuit elements then they have a place in the circuit loop matrix. The nature and position of measuring devices is critical to fulfilling the objective. If the position of the measuring device is not as indicated in the schematic then the fundamental loop matrices for the two circuits will be different and consequently will not match. In the above example the elements in the two fundamental loops in both circuits are not the same because of the rearrangement of the 10 and 15-ohm resistors. This is a clear indication that either the circuit is topologically incorrect or the measuring devices have not been used in a manner so as to complete the objectives of the experiment. The student is then urged to go back and perform necessary corrections to rectify the errors.

ENHANCEMENTS AND FUTURE DIRECTIONS

Although the CR is capable of analyzing circuits for equivalence, it restricts the user from using component values other than those indicated in the manual. A possible future improvement might be to allow the user the freedom of choice of the components to be used while assembling the circuit. The circuit checker mechanism can also be implemented in other advanced electrical engineering laboratories. It can be used in microprocessor and electron devices laboratories so long as the R-L-C models of the laboratory components are available. The output of the CR is currently recorded linearly in a file, which is formatted and displayed to the user. The possibility of presenting the output of circuit analysis in a Socratic manner through a dialogue system to the user is a very interesting research proposition that arises out of this research. This dialogue framework would use the result file (which contains all the details about student errors and explanations to rectify it) as the basis of the interaction with the user.

ACKNOWLEDGMENT

Partial Support for this work was provided by the National Science Foundation's Division of Human Resources Development through grant HRD-0004292.

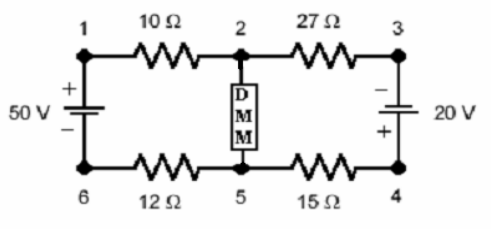

FIGURE3
EXAMPLE LABORATORY MANUAL SCHMATIC

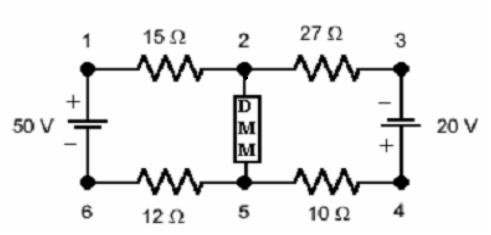

FIGURE4
STUDENT CREATED CIRCUIT ON THE BREADBOARD

REFERENCES

[1] Center for Disease Control, "Summary Health Statistics for the U.S. Population National Health Interview Survey", 2002.

[2] National Science Foundation, "SBIR Phase II Computer Simulation of Science and Technical Laboratory Exercises for Physically-Disabled Students", 29 September 1997 <http://www.fastlane.nsf.gov/servlet/showaward?award=9710548>.

[3] National Center for Education Statistics, "The 1996 National Postsecondary Student Aid Study" *U.S. Department of Education Washington, DC, Oct. 1996. 72-73.*

[4] Duarte M & Butz B.P, "An Intelligent Universal Virtual Laboratory (UVL)", *34th Southeastern Symposium on System Theory, Huntsville, 2002.*

[5] Vago Istvan, *"Graph Theory, Applications to the Calculation of Electrical Networks",* Elsevier 1985.

[6] Leon Chua .O, Pen-Min Lin, *"Computer Sided analysis of electronic circuits: Algorithms and computational techniques"*, Prentice Hall 1975