Table of Contents

Acknowledgement 3
Summary 5
1. Introduction 6
  1.1 ATP/EMTP Description 6
  1.2 Learning to use ATP/EMTP 6
2. How to create a data file for the simulation of electric circuits? 7
  2.1 Using ATPDraw to Create a Data File 7
    2.1.1 Introduction 7
    2.1.2 Conventions 8
3. Your first circuit with ATPDraw 9
  3.1 Starting to build a new circuit 11
    3.1.2 The source 12
    3.1.3 Load and switching device 15
    3.1.4 Grounding and Naming Nodes 19
    3.1.5 Storing the Circuit File 21
    3.1.6 Creating the ATP/EMTP Data File 21
4. Your first circuit without ATPDraw 24
  4.1 Introduction 24
  4.2 Helpful Hints 24
    4.2.1 Template 24
    4.2.2 Conventions 25
  4.3 Procedure to Setup a Data File 26
    4.3.1 BEGIN NEW DATA CASE 26
    4.3.2 FLOATING POINT MISCELLANEOUS DATA CARD 27
    4.3.3 CIRCUIT DATA CARDS 28
    4.3.4 SWITCH DATA CARDS 29
    4.3.5 SWITCH DATA CARDS 29
    4.3.6 BLANK CARDS 30
5. Running the ATP/EMTP Simulation 31
  5.1 Running the Simulation from ATPDraw 31
  5.2 Running the Simulation Directly on ATP/EMTP 31
  5.3 Plotting the results 32
6. References 35

Acknowledgement

The authors whishes to express their gratitude to Dr. Lionel Orama-Exclusa for his help throughout this work. His personals experiences and many technical discussions have helped in the completion of this work. The authors thank Dr. Efrain O'Neill-Carrillo and Dr. Miguel Vélez-Reyes for their faith on us and their support. Last but not least this work was supported by the ERC program of the National Science Foundation under award number EEC-9731677.

Mayagüez, P.R. June 18,2002

This report is based on [1], and has been prepared by Bienvenido Rodriguez-Medina and Marianela Santiago-Luna at the Department of Electrical and Computer Engineering of the University of Puerto Rico-Mayagüez Campus, under a research with the Center for Power Electronics Systems (CPES) a National Science Foundation Engineering Research Center. This work was supported by the ERC program of the National Science Foundation under award number EEC-9731677.

It is very important to mention that most of the information presented in this guide was obtained from three main sources:

- ATP/EMTP Rule Book [1].
- ATPDraw for Windows User's Manual [2].
- Instructions to Create a Data File for Simulation of Electric Circuits using the Electro-Magnetic Transients Program [3].

It is not the intention of the authors of this document to take full credit for the contents of the guide. This guide is just a quick description of the processes described in details by experts in the use of ATP/EMTP.

Prepare by:

Bienvenido Rodríguez-Medina

Marianela Santiago-Luna

Coordinator:

Efrain O'Neill-Carrillo

Summary

The Electromagnetic Transients Program (EMTP) and the Alternative Transients Program (ATP) are the standard computer software used by the electric power industry for simulation of electrical systems transients. The purpose of this work is to introduce the beginner user to ATP/EMTP. This document follows the ATPDraw user's manual [2] and the ATP/EMTP manual (Rule Book) [1] in which all the rules for the use of the program appear.

1. Introduction

1.1 ATP/EMTP Description

ATP/EMTP is considered the most widely used program for digital simulation of transient phenomena of electromagnetic, as well as electromechanical nature in electric power systems. It has been continuously developed through international contributions.

The ATP program calculates variables of interest within electric power systems as functions of time, typically started by some disturbances. Fundamentally, the trapezoidal rule of integration is used to solve the differential equations of system components in the time domain. Non-zero initial conditions can be determined either automatically by a steady state, phasor solution or they can be entered by the user for some components. ATP/EMTP has many models including rotating machines, transformers, surge arresters, transmission lines and cables. With this digital program, complex networks of arbitrary structure can be simulated. Analysis of control systems, power electronics equipment and components with nonlinear characteristics such as arcs and corona are also possible. Symmetric or unsymmetric disturbances are allowed, such as faults, lightning surges, or any kind of switching operations including commutation of valves. Calculation of the frequency response of phasor networks is also supported.

1.2 Learning to use ATP/EMTP

The best way of learning about ATP/EMTP capability and usage is by working with competent, experienced veteran. The ideal learning environments are those universities and power organizations that have been working with ATP/EMTP for a wide range of power system simulations, and have not suffer great changes in personnel. For those less fortunate with a lack of local experts, the best alternative is a national or regional EMTP User Group. In the case of questions or problems always ask to the experts. ATP/EMTP users must adopt the habit of asking questions. Since they may be using this program until the day they retire, and everyday a more complex simulation may be waiting for them.

It is really important to mention that this quick guide was written with the intention of introduce the new user to ATP/EMTP program, and have not the intention of creating experts. For a more in-depth or specialize knowledge of the simulation capabilities of ATP/EMTP we encourage the user to use the ATP/EMTP manual called the Rule Book.

2. How to create a data file for the simulation of electric circuits?

There are two methods for the creation of a data file to simulate electric circuits in ATP/EMTP. The first method is by using ATPDraw a graphical, mouse-driven preprocessor to the ATP/EMTP. It assists to create and edit the model of the electrical circuit to be simulated. The other method is by creating the data file using a text editor. For this method the user most have a background using text editors and some computer programming knowledge in FORTRAN, this method is most appropriate for experienced users.

2.1 Using ATPDraw to Create a Data File

2.1.1 Introduction

ATPDraw is a graphical, mouse-driven preprocessor to ATP/EMTP. ATPDraw helps creating and editing the model of the electrical circuit the user wants to simulate interactively. In the program the user can construct an electric circuit, by selecting predefined components from an extensive library. The preprocessor then creates the corresponding ATP/EMTP input file, automatically in correct format. ATPDraw administrates circuit nodes name, and the user can give a name only to the most important nodes. Most types of edit facilities like copy/paste, rotate, import/export, group/ungroup, undo and print are available. Other facilities in ATPDraw are a built-in editor for ATP-file editing, support of Windows clipboard for bitmap/metafile, output of Windows Metafile/Bitmap file format or PostScript files. ATPDraw is most valuable to new users of ATP/EMTP and is an excellent tool for educational purposes.

2.1.2 Conventions

The following file conventions are used in this manual:

Circuit file: The files in which ATPDraw stores the information about the constructed circuits. ATPDraw can load a circuit file and display the equivalent graphical picture on the screen. The default extension of the circuit files is `.CIR`.

ATP file: This is the file produced by ATPDraw and can be used for a subsequent simulation as input to ATP. The `.ATP` files can be edited with any text-processors, including ATPDraw's own *Text Editor* in the *Tools* menu. It is advised, however only for experts to manipulate this file manually.

Support file: All types of ATPDraw objects have a support file. This binary file specifies the data and nodes for an object with the icon and help information included. The support file can be edited inside ATPDraw via *Objects | Edit…* menu. The graphical representation of objects on the screen is editable via ATPDraw's built-in icon editor. New objects can be created specifying new support files. The support files should have a name with extension `.SUP`.

3. Your first circuit with ATPDraw

In this part of the guide we give some of the basic information on how to get started with ATPDraw. The guide will show you how to build a circuit step by step, starting from scratch.

ATPDraw has a standard Windows user interface. Fig. 2.1[1] shows the main window of ATPDraw containing two open circuit windows. ATPDraw supports multiple documents and offers the user to work on several circuits simultaneously along with the facility to copy information between the circuits. The size of the circuit window is much larger than the actual screen, as is indicated by the scroll bars of each circuit window.

[1] Figure 3.1 was taken from the ATPDraw for Windows 3.1x/95/NT User's Manual [1].

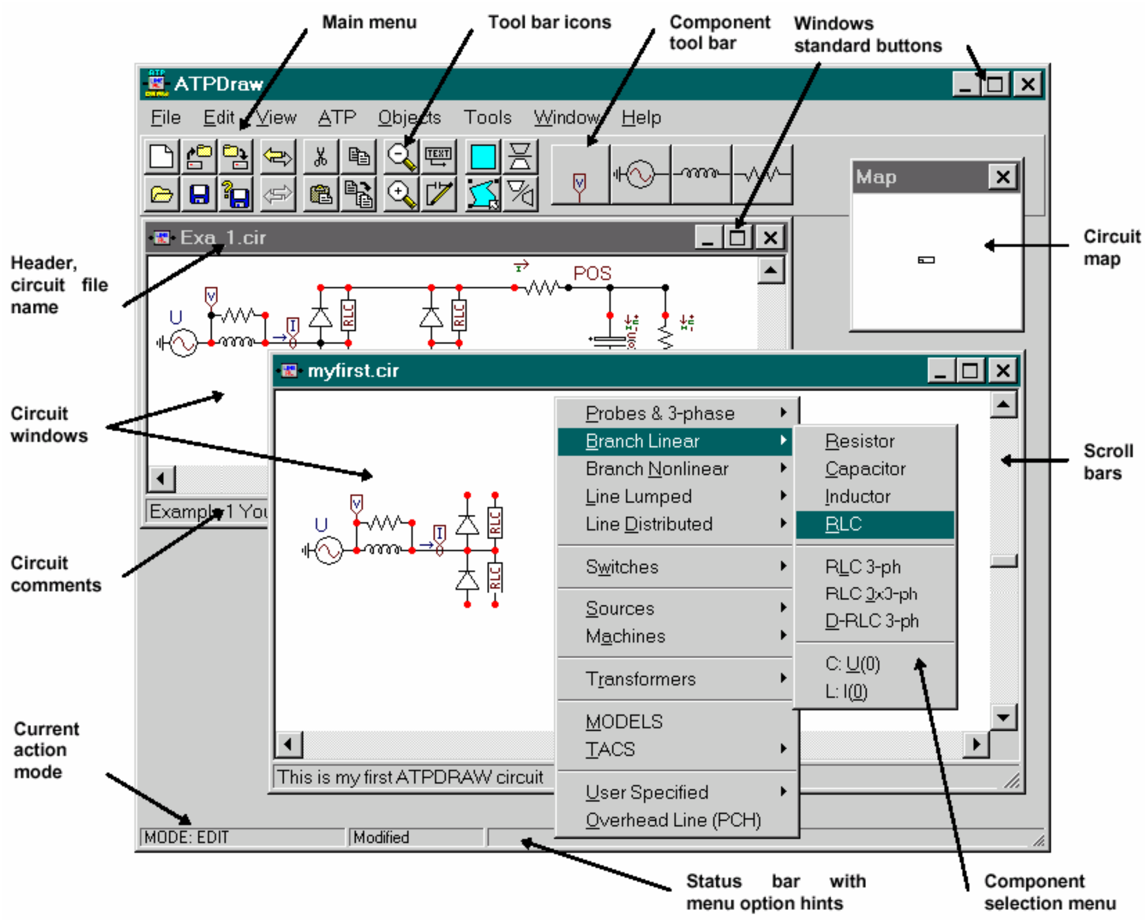

Fig. 3.1 Main window. Multiple Circuit windows and the floating Selection menu.

Now we are going to describes how to use ATPDraw step by step with an example, composing the circuit file of a single phase RLC circuit (see Figure 3.2) containing a time switch. Reading this tutorial carefully, you will learn to use the most important ATPDraw functions, such as:

- How to select and assemble components?
- How to perform edit operations and give data to components?
- How to give node names, draw connections and specify grounding?
- How to create the ATP input file and perform the simulation?

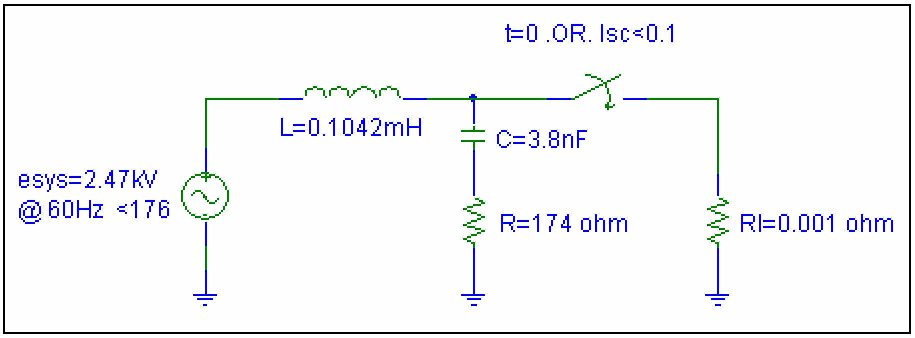

Fig. 3.2a Circuit used for the example[2].

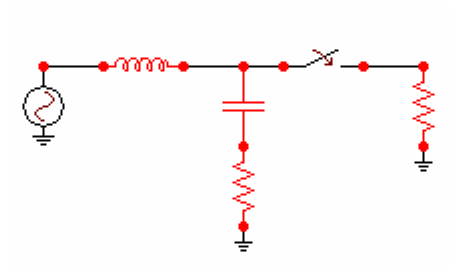

Fig. 3.2b Circuit used for the example drawn in ATPDraw.

3.1 Starting to build a new circuit

Most parts of the building process will be demonstrated in this part. The normal mode of operation is *MODE : EDIT*. You must always be in this mode to be able to select and specify data to objects. To return to EDIT from other modes, press *Esc*. Selecting the *New* command in the *File menu* or pressing the new (empty) page symbol in the *Component Toolbar*, a new circuit window will be created.

3.1.2 The source

The first step is to place the AC source, which is selected from the floating *Component selection men*u, which appears with a right mouse click on open area of the circuit window. Figure 3.3 shows how to select a single-phase sinusoidal voltage source *(Sources | AC type 1*4) using the mouse. After you have clicked in the *AC type 14* field, the selected source appears in the circuit window enclosed by a rectangle. Click on it with the **left mouse butto**n, hold down and drag it to a desired position. Then click with the left mouse button in open space to place it. The AC object is redrawn in red color as an indication that no data have been given to the object.

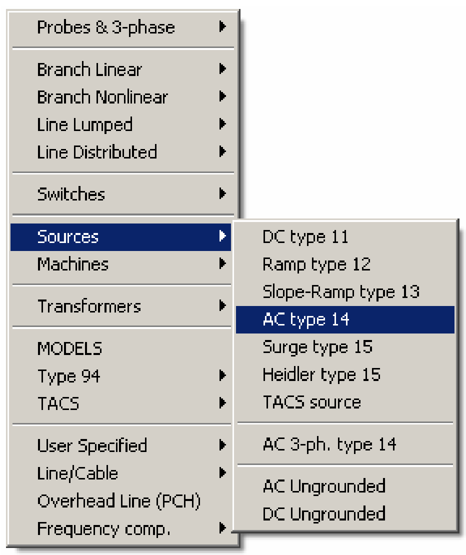

Fig. 3.3 Selecting a single phase AC source.

[2] The circuit shown in Fig. 3.2a was provided by Prof. Lionel Orama it was presented in [2].

Next select the source inductance as shown in Fig. 3.4:

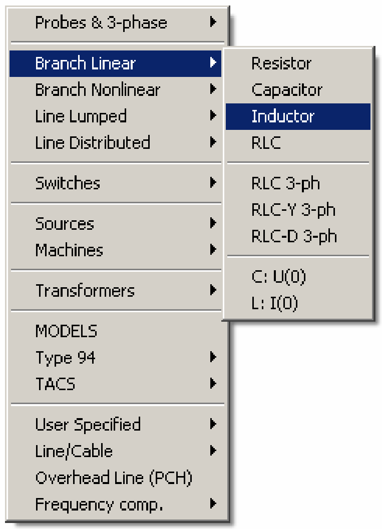

Fig. 3.4 Selecting the source inductance.

After you have clicked in the *Inductor* field, the selected inductor appears in the circuit window enclosed by a rectangle. Click on it with the left mouse button, hold down and drag it to a position shown in Fig. 3.5:

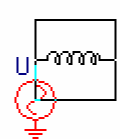

Fig. 3.5

Click on the white space with the left mouse button to place the inductor (the enclosing rectangle disappears). A grid snap facility helps you to place the inductor in the correct position. The inductor in Fig. 3.5 should be placed so that the node of the inductor touches the source. Objects having overlapping node dots will automatically be connected.

The circuit objects used in the circuit so far are in red color. This tells you that no data have been given to these objects. You can give data to objects at any time during the building process. We will now give data to the objects in the source part of the network. **To do so**: simple click with the **right mouse button** (or double click with the left button) in an object. When you click with the right button on the AC source icon, a window shown in Fig. 3.6 appears. Fig. 3.6 shows the window after the values for the circuit in Fig. 3.2a have been specified. The names of the numerical data menus are strongly related to the names used in the ATP/EMTP Rule Book [1].

The AC source has 7 input data and one node. The data correspond to the required ATP/EMTP data. Click on *HELP* to load a help file. This file explains the meaning of each input data and node.

U/I = 0 results in voltage source with default label U.

U/I = -1 results in a current source with label I.

Specify the data as shown in Fig. 3.6. The node names should normally not be specified in this window. Click *OK* to close the window and update the object values. Click on *Cancel* to just quit the window.

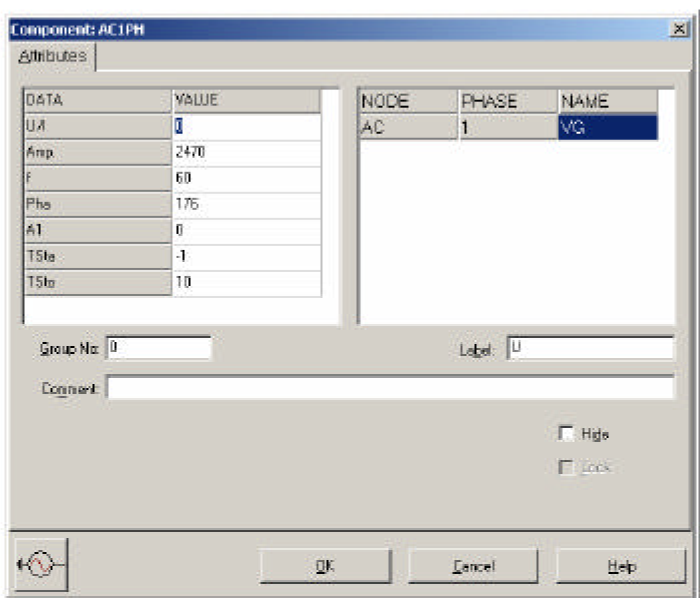

Fig. 3.6 AC1PH input window.

After you have given data to the AC source and closed the window note how the object layout changes when you exit the window. Repeat the procedure explained above to give data to the inductor by calling the *Component* dialog box of the objects. **To do so**: click with the right mouse button on the inductor icon.

3.1.3 Load and switching device

In this process you will learn to use some editing options like rotate, group, duplicate and paste. Since those are useful tools that give you the opportunity of not needing to build all of your posterior designs from scratch.

First you select a capacitor from the selection menu as shown in Fig. 3.7. After you have clicked on *Capacitor* the capacitor appears in the circuit window in marked, moveable mode enclosed by a rectangle.

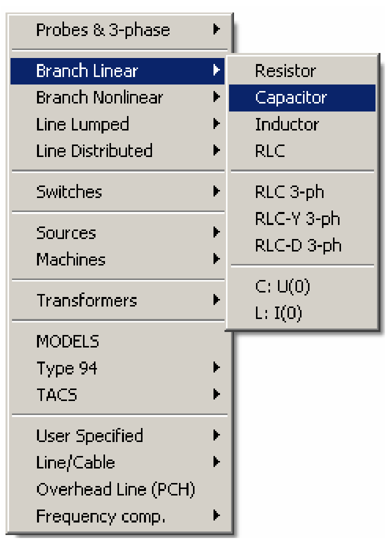

Fig. 3.7 Selecting the capacitor

The capacitor has to be rotated, so click the right mouse button or select *Edit* in the main menu and click on *Rotate.* The capacitor is now rotated 90 degrees counter clock-wise. Click on the capacitor with the left mouse button, hold down and drag to the position shown in Fig. 3.8. Click with the left mouse button on empty area to place the capacitor.

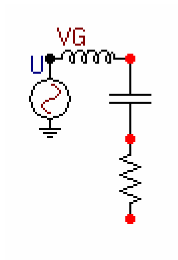

Fig. 3.8

Next you must select the resistor. Repeat the procedure explained above for placing the capacitor, and place the resistor. An RLC object could replace this RC branch. Selecting an RLC object could be done from the component selection menu (Fig. 3.7). The idea now is to present the copy tool with the RC branch, but before doing so, it is wise to give data to them (since the data are kept when copied). A simple click on the resistor or capacitor icon with the right mouse button activates the component dialog box to give data to objects. The RC branch in Fig. 3.8 has been given a resistance of 174 Ω and a capacitance of 3.8 nF.

You have now given data to the RC branch, and for example if you need more than one RC branch instead of repeating the drawing and data giving process many times you can use the copy tool. First you have to select the group of components. This can be done by drawing a rectangle around the objects by a left mouse click and hold at the upper-left corner of the desired rectangle, and moving thereafter to the lower-right corner. Objects inside the rectangle become a group when the mouse button is released. The group created can be copied/rotated etc. like a single object. Now we want to duplicate this group. So enter the main menu *Edit* field and choose *Duplicate* or press the *Ctrl+D* shortcut key. The selected group is copied to the clipboard and pasted in the same operation. The old group is redrawn in normal mode and the copy is drawn in the top of the original. The pasted group is moveable, so you can click on it with the **left mouse button**, hold down and drag to a desired position.

Now back to our circuit. The next step in our circuit is to place our circuit breaker or switching device. Switches have some parameters to help you simulate different disturbances. In this example we are going to use an ideal time controlled switch.

First you select the time-controlled switch from the selection menu as shown in Fig. 3.9. After you have clicked on *Switch time control* the switch appears in the circuit window in marked, moveable mode enclosed by a rectangle.

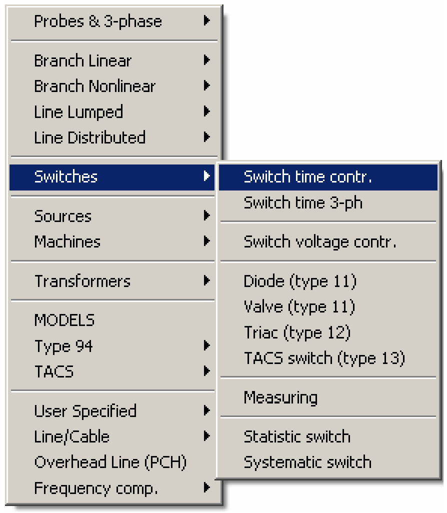

Fig. 3.7 Selecting the switching device

Click on the switch with the left mouse button, hold down and drag to the position shown in Fig. 3.10. Click with the left mouse button on empty area to place the switch.

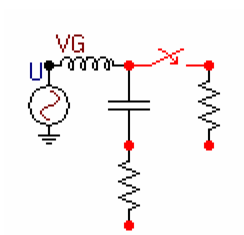

Fig. 3.10

Repeat the procedure explained above for placing the resistor, and place the resistor. A simple click on the resistor icon with the right mouse button activates the component dialog box to give data to resistor. The resistor in Fig. 3.10 has been given a resistance of 0.001Ω.

We will now give data to the time-controlled switch of the network. **To do so**: simple click with the **right mouse button** in the object. When you click with the right button on the icon, a window shown in Fig. 3.11 appears. Fig. 3.11 shows the window after the values for the switch in Fig. 3.2a have been specified.

The time-controlled switch has 3 input data and 2 nodes. The data correspond to the required ATP/EMTP data. Click on *HELP* to load a help file. This file explains the meaning of each input data and node. Specify the data as shown in Fig. 3.11. Click *OK* to close the window and update the object values.

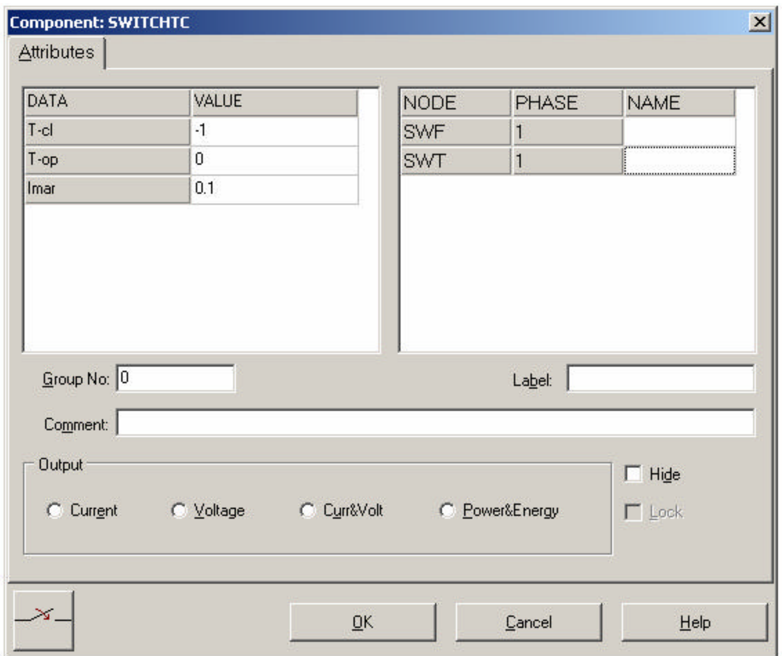

Fig. 3.11 SWITCHTC input window.

3.1.4 Grounding and Naming Nodes

The final step of building this circuit is to give data to nodes (node names and grounding). All nodes will automatically receive names from ATPDraw, so the user should normally only give names to nodes of special interest. It is generally advisable to let the node naming process be the last step in building up a circuit. This is to avoid undesirable multiple node names (which is corrected by ATPDraw automatically, but results in irritating warning messages).

To give data to a node, you simply have to click on this node once with the right mouse button. Fig. 3.12 - Fig. 15 show how to give data to four different nodes.

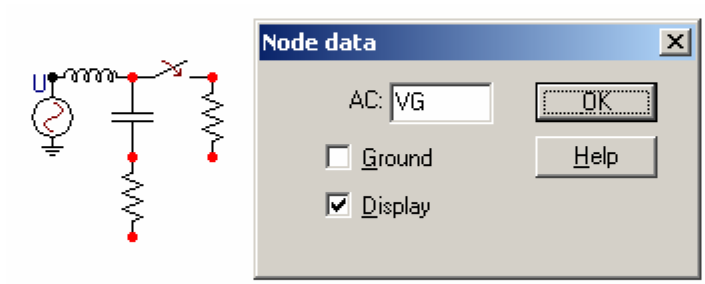

Fig. 3.12 Click on a node with the right mouse button and specify a name in the dialog box.

When you exit the window in Fig. 3.12 by clicking *OK*, the circuit is updated as shown in Fig. 3.13. All node names are forced to be left adjusted, and as a general rule in the ATP/EMTP simulation, capital letters should be used. ATPDraw does accept lower case characters in the node data window; however this "feature" should be avoided, in particular if the node is connected with electric sources.

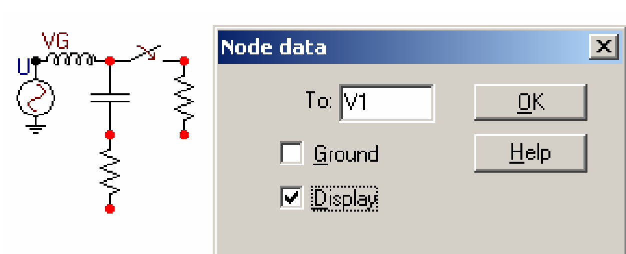

Fig. 3.13 Click on the node with the right mouse button and specify a name in the node data window.

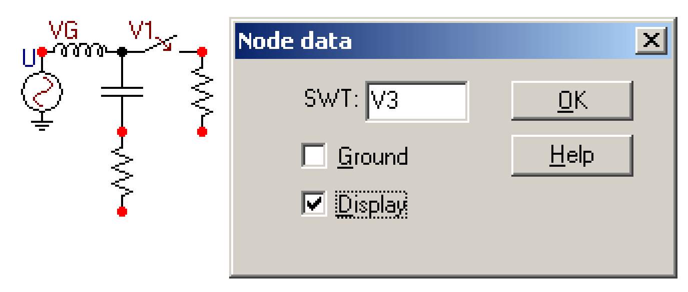

Fig. 3.14 Click on the node with the right mouse button and specify a name in the node data window.

The ground symbol is drawn at the selected node when you exit the window as Fig. 3.16 shows. The nodes not given a name by the user will automatically be given a name by ATPDraw, starting with XX followed by a four digit number. Nodes got the name this way are distinguished by red color from the user specified node names, as shown in Fig. 3.16.

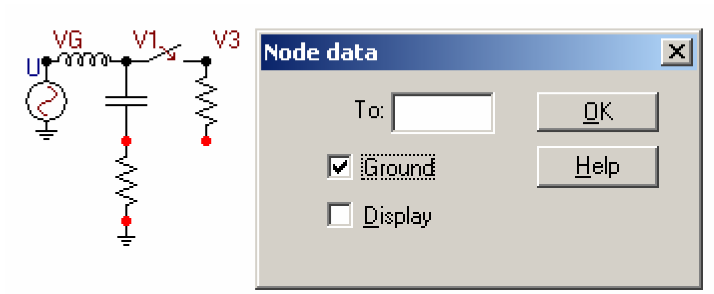

Fig. 3.15 Click on a node with the right mouse button and check the *Ground* box indicating that the node is connected with the ground reference plane of the circuit.

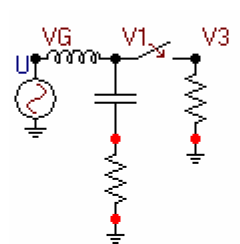

Fig. 3.16 All red nodes will be name by ATPDraw.

3.1.5 Storing the Circuit File

You can store the circuit in a disk file whenever you like during the building process. This is done in the main menu with *File | Save* (or *Ctrl+S*). If the current circuit is a new one, which has not been previously saved, a *Save As* dialog box appears where you can specify the circuit name. Two different styles of the *Save As* dialog boxes are available, depending on the *Open/Save dialog* setting in the *Tools | Options | General* menu: a Windows 95 standard dialog box and a Windows 3.1 style. The default extension is .*CIR* in both cases and it is automatically added to the file name you have specified. When the circuit once was saved, the name of the disk file appears in the header field of the circuit window. Then if you hit *Ctrl+S* or press the *Save circuit* icon in the Toolbar, the circuit file is updated immediately on the disk. The *File + Save As* option or the *Save As* icon from the Toolbar allows the user to save the circuit currently in use under a name other than that already allocated to this circuit.

3.1.6 Creating the ATP/EMTP Data File

The ATP/EMTP file is the file required by ATP/EMTP to simulate a circuit. The ATP/EMTP file is created by selecting *Make File* command in the *ATP* main menu.

Before you create the ATP/EMTP file, you must specify some miscellaneous parameters. The default values of these parameters are given in the `ATPDraw.ini` file. Changing these default values can either be done in the *Settings | Simulation* sub-menu under the *ATP* main menu for the current circuit, or under the *Tools | Options | View/ATP | Edit settings* for all new circuits created henceforth. Fig. 3.17 shows an example of the ATP's 1st miscellaneous data card settings (specifying time step, time scale of the simulation etc.). This window appears if you select the *Simulation* tab of the *ATP | Settings* menu.

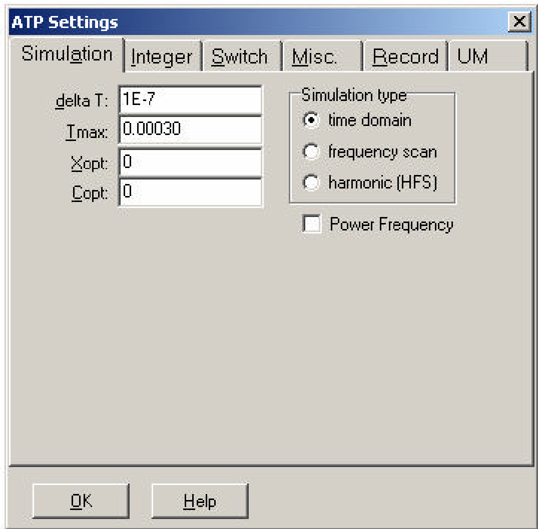

Fig. 3.17 Simulation settings.

The simulation settings are stored in the circuit file, so you should save the file after changing these settings. The first integer miscellaneous data card is changed under the *ATP* | *Settings* | *Integer* page, and the statistic/systematic switch control card is specified under the *ATP* | *Settings* | *Switch settings*.

To create an ATP/EMTP file you must select the *Make File* in the *ATP* menu. This selection will start a procedure, which examines your circuit and gives node names to circuit nodes. Then a standard Windows' *Save As* file window appear, where you can specify the name and path of the ATP/EMTP file. The same name as the circuit file with extension .ATP is suggested. You can load an old circuit whenever you like (select *File* | *Open*) and create the corresponding ATP file (select *ATP* | *Make File*).

The ATP file (QUICK_1.ATP) you just have created will look as follows:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  June, wednesday 19, 2002
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at SEfAS - NORWAY  1994-98
C --------------------------------------------------------
C Miscellaneous Data Card ....
C  dT  >< Tmax >< Xopt >< Copt >
1.000E-7   .0003
     500       1       1       1       1       0       0       1       0
C        1         2         3         4         5         6         7         8
C 345678901234567890123456789012345678901234567890123456789012345678901234567890
/BRANCH
C < n 1>< n 2><ref1><ref2>< R  >< L  >< C  >
C < n 1>< n 2><ref1><ref2>< R  >< A  >< B  ><Leng><><>0
  VG    V1                    .1042                                            3
  XX0004V1                          .0038                                      3
  XX0004                   174.                                                3
  V3                       .001                                                3
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
  V1    V3           -1.                .1                                     3
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
14VG     0   2470.       60.       176.                               -1.       10.
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

Fig. 3.18 Resulting ATP data file

You can edit this file or just display it by selecting the *ATP | Edit file* menu. In the next section we will teach you how to create the same data file using a text editor. But this method is recommended only for expert users, or in the case of specialize simulation, which require entering the data file.

For now the important thing here is that the file you just created is the input to run ATP/EMTP to make the simulation of your circuit. However, running the simulation is worth of a new set of instructions. Those instructions are given in section 5.

4. Your first circuit without ATPDraw

4.1 Introduction

The procedure in this text is made for the beginner user of ATP/EMTP, which is going to work with the data file by its self for the first time. In these instructions we will assume that you have the following background:

A. DOS basics
B. Use a text editor;
   1. Create files.
   2. Open files.
   3. Save files.
C. Run programs.
D. Some computer programming knowledge in FORTRAN.
E. Basic electrical circuit understanding.

4.2 Helpful Hints

For better understanding, we will make reference to the data file example of the previous sections during the instructions. Also, we placed the related segment of the data file before the explanation of each set of fields.

4.2.1 Template

The **template** is a file already stored in the computer editor in which all the variables' fields are commented. We use the template whenever we want to create a new data file. With a template it is very easy to create a new data file by changing the values in the fields.

4.2.2 Conventions

If you look at the example (QUICK_1.ATP), you will see many lines beginning with a letter C, those are comment lines. Comment lines have different purposes in the template. For example:

```
C Miscellaneous Data Card ....
C  dT  >< Tmax >< Xopt >< Copt >
1.000E-7   .0003
     500       1       1       1       1       0       0       1       0
```

The first comment line is the title of the set of fields, the second comment line is the identification of each field column. We recommend you type all the comment lines while typing the example so you can save a file with the template.

This section will take you through a chronological implementation of the example. Also, the section describes all the necessary variables and parameters that the data file needs so that ATP/EMTP will understand and compile your file. If you follow the instructions while writing your data file, you should have a document that looks exactly like the one presented in section 3.1.6. In addition, we use the following conventions:

A. **System voltage** on the circuit, esys, is the node voltage, **VG**, on the data file.
B. The voltage of interest is **V1,** the voltage experienced by the switch.
C. The current of interest is **Isc**, the short circuit current.
D. You can use any six letters name for any circuit parameter, node voltages and branch currents, etc.
E. ATP/EMTP is very sensitive to the **column position.** For example, if a field starts at column 3 and end at column 8 you should keep your entry in the column range 3 to 8. Otherwise, ATP/EMTP will not run or will give wrong results.
F. **Comments** are place by typing a **C** at the first column followed by a blank space and then your comments.
G. EMTP will read your file only if the name has extension **.dat**, i.e., when saving the file do it as **name.dat**. Remember to save your file regularly, so that you do not lose your work if there is a problem with the computer or the power supply.

4.3 Procedure to Setup a Data File

Load your favorite editor from the computer to start the procedure. We recommend the use of capital letters for the whole data file. In capital letters any version of the ATP/EMTP will recognize your files, for example, UNIX versions are case sensitive.

4.3.1 BEGIN NEW DATA CASE

This is the first line of your file and should start at the first column. ATP/EMTP looks for this line to start the compilation of your program.

Notes:

i) We recommend that all **numbers** have their **last digit** aligned with the **last column** of the field, for those versions of ATP/EMTP that are extremely sensitive to column position.

ii) We recommend that all **names have** their **first letter** aligned with the **first column** of the field, for the same reason as note (*i*).

There are several special instructions of a specific type that follows the BEGIN NEW DATA CASE request. They are called **MISCELLANEOUS DATA CARDS.** In this beginner's guide, we will show you the use of the two that are most important. When you become an advance user you have to refer to the Rule Book (RB) for more advance instructions.

4.3.2 FLOATING POINT MISCELLANEOUS DATA CARD

This instruction gives the program the parameters of the simulation. Lets look at the file lines, to see the parameters and explain them.

```
C Miscellaneous Data Card ....
C  dT  >< Tmax >< Xopt >< Copt >
1.000E-7   .0003
```

a. **dt** is the time step of the integration. In the example, the natural period of the transient oscillations will be $T = 2 * p * \sqrt{LC}$ in seconds (~3.95E-6 sec). You can start by dividing *T* by 10 and that is the **biggest value of dt** recommended. This value is different for different circuits.

b. **tmax** is the maximum time of simulation. The value, again, is dependent on the case that you are simulating. You could start with two or three orders of magnitude bigger than the period computed above (100**T* or 1000**T*).

c. **xopt and copt** are options for the input of inductance and capacitance.

Type:

0. for inductances in milihenries and capacitances in microfarads

1. for inductance in ohms (reactance) and capacitance in ohms (reactance)

Important: All real numbers should be typed with the decimal point (for example 1. instead of 1).

4.3.3 CIRCUIT DATA CARDS

These are the branches of your circuit. **Series elements** are input in the same line (RC branch). **Parallel elements** are connected in different lines (short circuit branch of low impedance).

```
C 345678901234567890123456789012345678901234567890123456789012345678901234567890
/BRANCH
C < n 1>< n 2><ref1><ref2>< R  >< L  >< C  >
C < n 1>< n 2><ref1><ref2>< R  >< A  >< B  ><Leng><><>0
  VG    V1                        .1042                                         3
  XX0004V1                              .0038                                   3
  XX0004                    174.                                                3
  V3                        .001                                                3
```

a. **n1** and **n2** are the beginning and ending nodes of a branch. In third branch line the beginning node is XX0004 and the ending node has no name because it is the reference node (ground). From the circuit you can see the values of the individual elements of the branches.

Remember: The file needs the capacitance in microfarads, not farads or nanofarads (3.8nF=3.8E-3=0.0038)

b. **ref1 and ref2** can be use if there are branches with the same impedance. Instead of rewriting the values of the elements you can refer the program to a branch above.[3]
c. **R** is the **resistance** branch in ohms.
d. **L** is the value of the **inductance**, in milihenries or ohms.
e. **C** is the value of the **capacitance**, in microfarads or ohms.

Important: If you want information of any specific branch in your output file you must type one of the following integers in **column 80,**

1, for branch current output

2, for branch voltage output

3, for both, branch current and voltage

4, for branch power and energy consumption

0 or blank, there is no output from that branch.

[3] The use of this is totally optional. It is convenient for every large circuits.

4.3.4 SWITCH DATA CARDS

Switches have some parameters to help you simulate different disturbances.

```
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><    Ie    ><Vf/CLOP ><  type  >
  V1    V3          -1.                    .1                      3
```

a. **n1** and **n2** are the beginning and ending nodes of the branch containing the switches.
b. **Tclose** is the time at which the switch will close. In our example the switch will close before the simulation starts (@ t = -1 sec).[4]
c. **Top/Tde**, before this time the switch can not open.
d. **Ie** is the current at which the switch is able to physically open.[5]

Note: You can have the switch open first and then close it. The open/close sequence is determined by the problem that you are solving.

4.3.5 SWITCH DATA CARDS

This line gives the program the information about the energy source of your circuit. In the **first two columns** you type **the source code number**. This code tells the program the type of source of your circuit, in our example, the circuit has a sinusoidal voltage source that is code number 14.[6]

```
/SOURCE
C < n 1><>< Ampl.   >< Freq.   ><Phase/T0><    A1    ><    T1    >< TSTART >< TSTOP  >
14VG      0     2470.        60.       176.                                    -1.        10.
```

a. **n1** is the node at which the source is connected. Notice that at this level you can only have sources connected from one node to ground..
b. **Ampl** is the peak magnitude of the source.
c. **Freq** is the frequency of oscillation of the sinusoidal signal.

[4] In the example, when the simulation starts a short circuit current is flowing through node V3.
[5] At the beginner level this is not an issue, you can type a very small value to avoid a current chopping.

d. **Phase/T0** is the phase of the source signal in seconds or degrees, depending on the parameter **A1**.

e. **A1** indicates that a value of 0 or blank means a phase in degrees. This implicates that a value different from 0 indicates a phase in seconds (**0.=degrees,1.=sec**).

f. **TSTOP** is the maximum simulation time for the source. Usually, the value is bigger than the simulation time.

Notes:

i) We recommend that all **numbers** have their **last digit** aligned with the **last column** of the field, for those versions of ATP/EMTP that are extremely sensitive to column position.

ii) We recommend that all **names have** their **first letter** aligned with the **first column** of the field, for the same reason as note (*i*).

4.3.6 BLANK CARDS

You have at the end of the case these types of cards to let the program knows the end of the data files.

Now that you finished your data file, save it again. Check your file with the example in figure 1 to see if they are exactly the same. This file is your input to run ATP/EMTP to make the simulation of your circuit. However, that is material for another set of instructions, which is given in the next section.

[6] There are many sources available for use within EMTP. Each has a code number and different parameters.

5. Running the ATP/EMTP Simulation

5.1 Running the Simulation from ATPDraw

ATPDraw allows the execution of user specified batch jobs internally. One of the most frequently used batch jobs might be to run your current circuit file through ATP, performing a simulation directly from ATPDraw. This feature is supported via the *ATP | Edit batch jobs* submenu, where you can specify your own menu items, which are added to the existing commands of the *ATP menu*. The Installation Manual of the ATPDraw User's Manual [2] describes in detail how to create a *Run ATP* command to perform an ATP simulation.

5.2 Running the Simulation Directly on ATP/EMTP

To run the simulation on ATP/EMTP the user must save the file with extension .ATP (*QUICK_1.ATP*) into the directory containing the `TPBIG.EXE`. This `TPBIG.EXE` is the executable for ATP/EMTP. The user needs to double click on the ATP/EMTP icon to start the program, and then a DOS type prompt window will appear with a menu just like the one in Fig. 5.1. On this menu screen the user must write the name of the file including the extension (*QUICK_1.ATP*) and press enter.

Fig. 5.1 ATP program window.

The program will read the file information and run the simulation case. If there is a problem in the file the program will stop and display an error message. After the program recognize all the data and run the case it will create a file with the same name but with the extension .pl4 (*QUICK_1.pl4*). This file must be sent, to the directory containing the GTPLOT32.EXE, which is the executable file for GTPLOT graphic generator. This program generates the plots of the output variables of the simulation.

5.3 Plotting the results

To get the plots of the output variables of the simulation of the ATP/EMTP simulation the user must save the file with extension .pl4 (*QUICK_1.pl4*) into the directory containing the GTPLOT32.EXE. This GTPLOT32.EXE is the executable for plot generator program. The user needs to double click on the GTPLOT icon to start the program, and then a DOS type prompt window will appear with a menu just like the one in Fig. 5.2. On this menu screen the user must write the name of the file including the extension (*QUICK_1.pl4*) and press enter.

Fig. 5.2 GTPLOT program window.

The program will read the file information of the simulation case. If there is a problem in the file the program will stop and display an error message. After the program recognizes all the data, the user should write **choice** in the menu to see the variables option as in Fig. 5.3. The user should then call the desired variables by writing **# symbol and the number of the variable** (look on Fig. 5.3 to follow the example). For the purpose of our example the user should write **#5** and **#6** separated by a space (#5 #6) and the press enter.

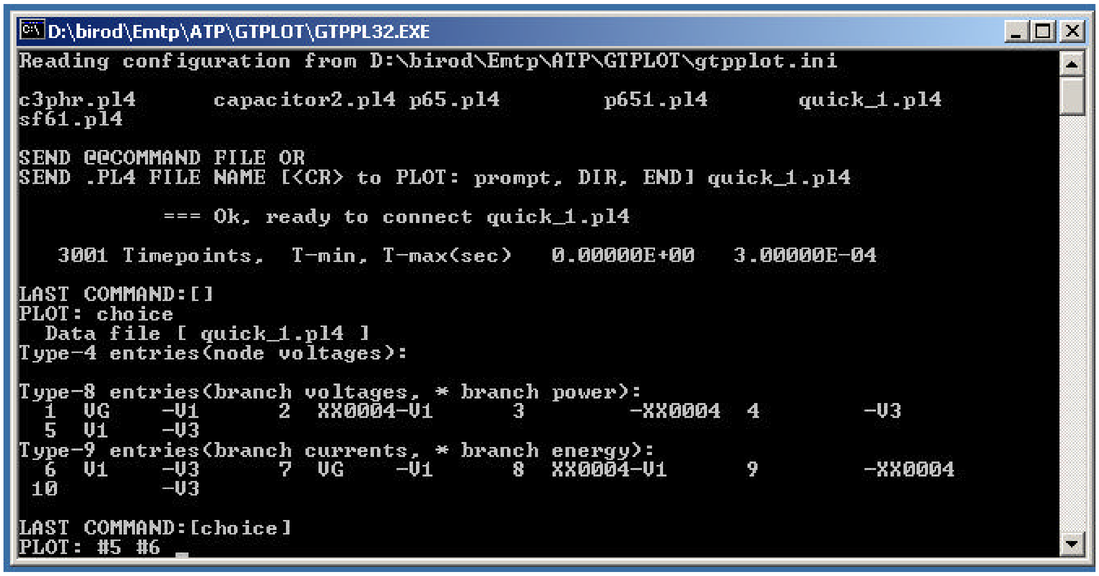

Fig. 5.3 Selecting variables for plotting in GTPLOT.

Now the user has two choices, see the plot results on the screen or save the data for the plots on a MATLAB data file (.mat file). The authors recommend the second choice, because the users will have a wider range of tools when they are working with the data in MATLAB. So, the next step should be to write **matlab** on the menu screen and then press enter (see Fig. 5.4). The final step is to type go and the program will create a file with the extension .mat (*QUICK01.mat*). This file contains all the output variables you need from the simulation case, now the user is free to do whatever he wants with the simulations variable in MATLAB. Fig. 5.5 shows the plots of the two variables choose for the example, so the user can compare their results and look for possible errors.

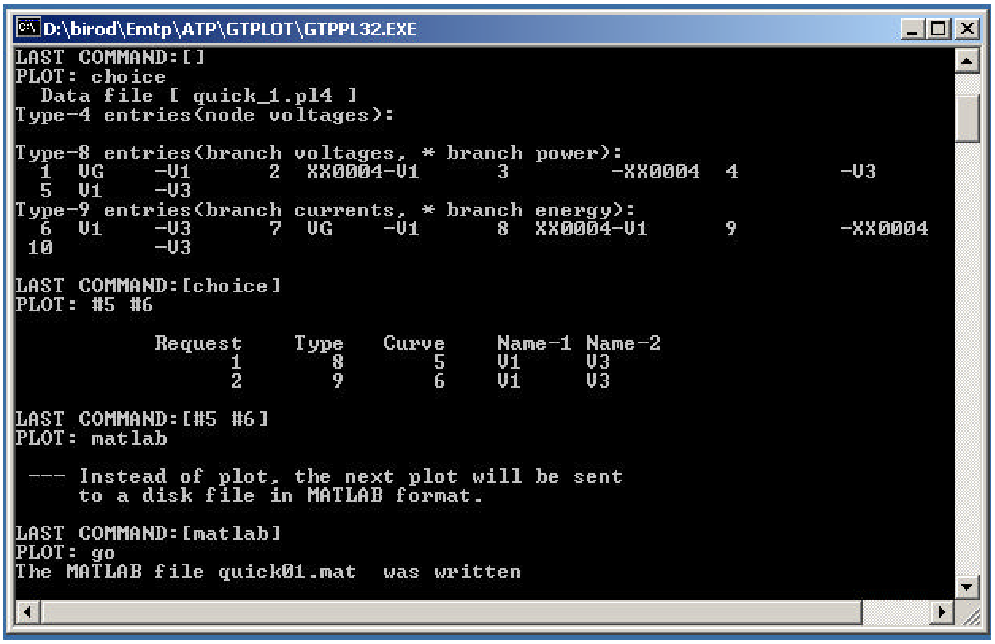

Fig. 5.4 Creating the MATLAB data file.

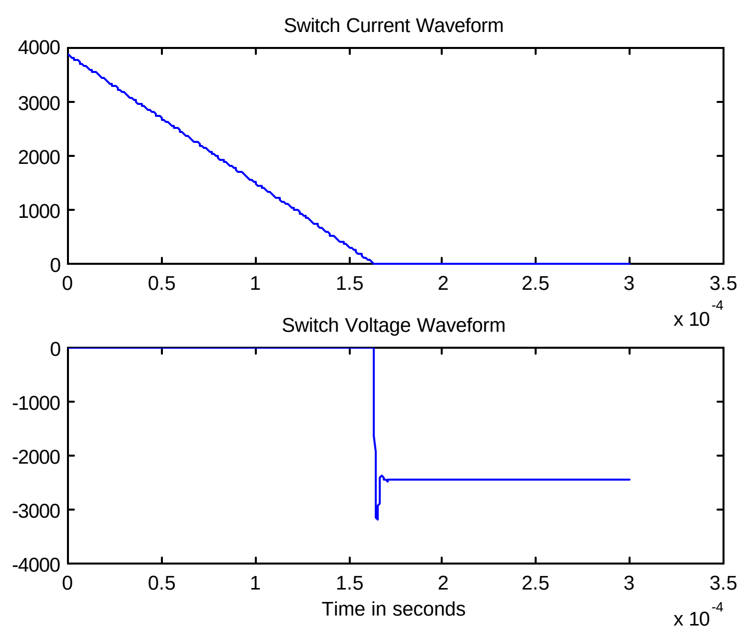

Fig. 5.5 Plots of the selected variables.

To finish the authors want to let the user know that this guide had no intention of create ATP/EMTP experts. The purpose was to introduce you to the basic of ATP/EMTP and we encourage you to look for the ATP/EMTP Rule book and the ATPDraw User's Manual for a more in-depth sight of the program capabilities.

6. References

[1] ATP/EMTP Rule Book, Canadian-American EMTP Users Group

[2] ATPDraw for Windows 3.1 User's Manual; Lázló Prikler, Hans Kr. Høidalen; 1998

[3]"Instructions to Create a Data File for Simulation of Electric Circuits using the Electro-Magnetic Transients Program", Dr. Lionel R. Orama-Exclusa