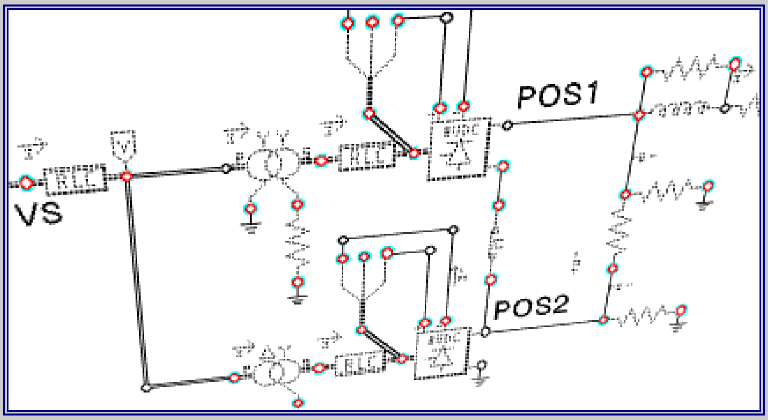
ATPDraw
for Windows 3.1x/95/NT
version 1.0

User's Manual



László Prikler
Hans Kr. Høidalen

Release No. 1.0.1

November 1998

ATPDraw
for Windows
version 1.0

Trondheim, Norway 15th October 1998

PREFACE

ATPDraw is a graphical preprocessor to ATP [3]. This report (TR A4790) contains a complete description of the program (ATPDraw for Windows, version 1.0) and replaces the manual for the DOS version [1]. Data files for old versions of ATPDraw can be converted to the new format, as explained in chapter 2.7. A preprocessor called ATP_LCC.EXE for ATP's Line & Cable Constants is described in chapter 6. New users of ATPDraw are recommended to focus on chapter 2; Installation Manual and chapter 3; Introductory Manual.

ATPDraw is developed by EFI, which now has changed name to SINTEF Energy Research (SEfAS). Program development and writing of this manual have been financed by Bonneville Power Administration, USA.

This report is based on [1], and has been prepared by László Prikler at SYSTRAN Engineering Services Ltd. in Budapest, under a contract with SINTEF Energy Research.

For SINTEF Energy Research

Hans Kr. Hoidalen
project manager

SUMMARY

ATPDraw is a graphical, mouse-driven preprocessor to ATP on the MS-Windows platform.

The ATPDraw program has a standard Windows layout. The user can build a graphical picture of an electric circuit by selecting components from menus. The preprocessor then gives names to unspecified nodes automatically and creates the ATP file in correct format. ATPDraw now supports about 70 standard components and 28 TACS objects. A simplified usage of MODELS is also supported. In addition, the user can create his own circuit objects using the Data Base Module option in ATP. Both single phase and 3-phase circuits can be constructed. Several circuit windows make possible to work on more circuits simultaneously and copy information between the circuits. Most types of edit facilities like copy/paste, rotate, import/export, group/ungroup, as well as printing to paper or metafile/bitmap format are available.

The program is written in Borland Pascal. Two functionally very similar versions of ATPDraw exist: a 32-bit version which is written in Borland Delphi 2.0 and runs only under Windows 95/NT, and a 16-bit version compiled with Borland Delphi 1.0 for Windows 3.1x. The ATPDraw package also includes the ATP_LCC program for Line/Cable constants support and the utility Convert. The latter makes possible the usage of existing circuit files created by the DOS version of ATPDraw under the new environment.

CONTENTS

Page

1. Introduction . . .

. . . .

ATPDraw™
for Windows
1.0

1.1 What is ATPDraw?

ATPDraw™ for Windows is a graphical, mouse-driven preprocessor to the ATP version of the Electromagnetic Transients Program (EMTP). It assists to create and edit the model of the electrical network to be simulated, interactively. In the program the user can construct an electric circuit, by selecting predefined components from an extensive palette. The preprocessor then creates the corresponding ATP input file, automatically in correct format. Circuit node naming is administrated by ATPDraw and the user only needs to give name to "key" nodes. ATPDraw currently supports about 70 standard components and 28 TACS objects. A simplified usage of MODELS is also possible. In addition, the user can create his own circuit objects using the Data Base Module and the $INCLUDE option of ATP. Both single phase and 3-phase circuits can be constructed. Multiple circuit windows are supported to work on several circuits simultaneously and copy information between the circuits. 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 and is an excellent tool for educational purposes. It is to be hoped, however that even experienced users of ATP will find the program useful for documentation of circuits and exchanging data cases with other users. The possibility of building up libraries of circuits and sub-circuits makes ATPDraw a powerful tool in transients analysis of electric power systems. The ATPDraw package also includes the ATP_LCC program for Line/Cable constants support and a utility, that makes possible the usage of existing circuit files created by the previous (DOS/GIGS) versions of the program under the new environment.

The program is written in Borland Pascal. Two functionally very similar versions of ATPDraw exist. A 32-bit version which is written in Borland Delphi 2.0 runs only under Windows 95/NT and a 16-bit version compiled with Borland Delphi 1.0 for Windows 3.x.

ATPDraw™ is a trademark and copyrighted by © 1996-1997 SINTEF Energy Research, Trondheim, Norway. It is programmed and maintained by Dr. Hans Kr. Høidalen. The program was redesigned and converted to Windows by O. G. Dahl, Dahl Data Design, Norway.

The ATPDraw for Windows program is royalty free and can be downloaded free of charge from the ftp server ftp.ee.mtu.edu (user:anonymous, password: your e-mail address). The proprietary rights of the program belong the Bonneville Power Administration, USA, the company who financed the program development and to SINTEF Energy Research (formerly EFI), Norway. The contact person at BPA is Mr. James L. Hall. Any request for obtaining the code should be directed to BPA.

1.2 Short description of ATP

ATP (Alternative Transients Program) is considered to be one of the most widely used software 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 over the past 20 years, coordinated by the Canadian/American EMTP Users Group co-chaired by Drs. W. Scott Meyer and Tsu-huei Liu.

The ATP program calculates variables of interest within electric power networks as functions of time, typically initiated by some disturbances. Basically, 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 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, any kind of switching operations including commutation of valves. Calculation of the frequency response of phasor networks is also supported.

ATP includes at present consists of the following components [3]:

- Uncoupled and coupled linear, lumped elements.
- Transmission lines and cables with distributed and frequency-dependent parameters.
- Elements with nonlinearities: transformers including saturation and hysteresis, surge arresters, arcs.
- Ordinary switches, time-dependent and voltage-dependent switches, statistical switching
- Valves (diodes and thyristors).
- 3-phase synchronous machines, universal machines.
- MODELS and TACS (Transient Analysis of Control Systems).

MODELS in ATP is a general-purpose description language supported by an extensive set of simulation tools for the representation and study of time-variant systems. MODELS allows the description of arbitrary user-defined control and circuit components, providing a simple interface for connecting other programs/models to ATP. As a general-purpose programmable tool, MODELS can be used for processing simulation results either in the frequency domain or in the time domain.

The following supporting routines are available in ATP:

- LINE CONSTANTS, CABLE CONSTANTS and CABLE PARAMETERS for calculation of electrical parameters of overhead lines and cables
- Generation of frequency-dependent line model input data: JMARTI Setup, SEMLYEN Setup and NODA Setup.
- Calculation of model data for transformers (XFORMER and BCTRAN).
- Saturation and hysteresis curve conversion.
- Data Base Modularization

ATP is available for most Intel based PC platforms under DOS, Windows 3.x/95/NT, OS/2, Linux and for other computers, too (e.g., Digital Unix and VMS, Apple Mac's, etc.). The program is in principle royalty free, but requires a license agreement signed by the requester and the Canadian/American EMTP Users Group, or the authorized regional users group representatives. The last chapter of the Installation Manual gives more information about the regional ATP user groups and about the ATP related resources on the Internet.

1.3 History of the ATPDraw development

The first version of a simple graphical preprocessor was developed at the Norwegian Institute of Technology in 1991. The program was intended for educational purposes. Two demo versions of ATPDraw were demonstrated at the European EMTP Users Group fall meetings in Leuven, Belgium in 1991 and 1992. The Bonneville Power Administration (BPA), USA then contacted the developers with the intention to finance further development of the program. The first phase of the project was completed in May 1994 and ATPDraw was made available via the Internet together with its manuals. In the second stage of the project, completed in December 1995, some important limitations of the previous version were overcome. The program became compatible with the DOS Protected Mode Interface and could run more reliably in Windows' DOS box, as well as on network connected PCs. In addition, the program speed was increased since no overlay and disk swapping were required. Thanks to the continuous support of BPA, the program development was going on. A completely new release written in Borland Delphi is now available for the most popular Windows operating systems: a 32-bit version for Windows 95/NT, and a 16-bit version for Windows 3.x. This User's Manual has been prepared for users of that recently issued ATPDraw version 1 for Windows.

1.4 Available components in ATPDraw

ATPDraw supports most of the frequently used components in ATP [3]. The components listed below are single phase as long as nothing else is specified.

Standard components

Linear branches:

- Resistor, Inductor, Capacitor, RLC
- RLC 3-phase, symmetric and non symmetric
- Inductor and capacitor with initial condition

Non-linear branches:

- Current dependent resistor, type 99
- Current dependent inductor, type 98 and type 93
- Time dependent resistor, type 97
- Current dependent, exponential resistor, type 92 (1 and 3 phase). ARRDAT fitting included.
- TACS controlled resistor.

Line models:

- RLC pi-equivalent. 1, 2 and 3-phase
- RL coupled. 2, 3 and 6-phase
- RL symmetric, sequence input. 3 and 6-phase
- Clarke distributed parameter. 1, 2, 3, 6 and 9-phase

Switches:

- Time controlled. 1 and 3-phase
- Voltage controlled
- Diode, type 11
- Valve, type 11
- TACS switch, type 13
- Measuring
- Double TACS switch, type 12

Statistic, independent
Systematic, independent

Sources:

DC, type 11
Ramp, type 12
Two-slope ramp, type 13
AC. 1 and 3 phase, type 14
Double-exponential source, type 15
Heidler source, type 15
TACS source, type 60
Ungrounded DC source, type 11+18
Ungrounded AC source, type 14+18

Machines:

Synchronous machine type 59 with maximum 8 TACS control variables.
Universal machine. Manual and automatic initialization.
Induction machine. Type 3.
DC machine. Type 8.
Synchronous machine. Type 1.

Transformers:

Singe phase ideal. Type 18 source.
Single phase with saturation.
Saturable transformers, 3-phase. Coupling: D/D, Y/Y, D/Y, Y/D, Y/Y/D. Saturation calculation from RMS values included.
Transformer three phase, 3-leg type. Coupling Y/Y. Preprocessing of standard measurement data.

TACS

Transfer functions. G(s) with or without limits.
Sources. DC, AC, PULSE, RAMP, EMTP node voltage.
FORTRAN Statements. 1 -phase. Single line statement.
Devices. Type 50-54 and 58-66. Initial condition.

User specified objects

Users can create new objects in ATPDraw. These objects are written to the ATP file using Data Base Modularization.

Some example cases are included:

- 6-phase thyrisor bridge with control.
- 3-phase J-Marti overhead line.
- transformer modeling using BCTRAN
- hysteretic inductor

Overhead line (PCH) objects

Line/Cable Constants punch file formats are recognized automatically

- 1-9 phase distributed, constant parameter transmission line
- 1-9 phase equivalent PI-circuits
- 1-9 phase frequency dependent JMarti line model

MODELS

Usage of MODELS is possible in ATPDraw. The user can add his own MODELS "procedures" to the program.

1.5 Contents of this manual

This User's Manual of ATPDraw for Windows contains five parts:

INSTALLATION MANUAL
- How to get the program?
- How to install ATPDraw?
- Hardware requirements
- How to configure your system?
- How to convert existing circuit files?
- How to communicate with other users and program developers?

INTRODUCTORY MANUAL
- How to create a circuit in ATPDraw?
- Operating windows
- Your first circuit
- Three phase circuits

REFERENCE MANUAL
- Reference of all menus and ATPDraw objects
- Main menu
- Component selection menu
- Map window
- ATPDraw.ini settings

ADVANCED MANUAL
- Other examples
- How to create new circuit objects in ATPDraw?
- How to use MODELS and $INCLUDE in ATPDraw?
- Creating libraries of circuits and user specified libraries
- Application examples:
  - Line energization studies
  - Single-phase to ground fault and fault tripping transients
  - Transformer energization, inrush currents
  - Shunt capacitor bank switching
  - HVDC station, rectifier/converter modeling
  - Lightning studies, arrester modeling
  - Electric arc simulation using MODELS

LINE/CABLE CONSTANTS MANUAL
- How to get the program?
- How to install ATP_LCC?
- How to create circuits with line/cable data?
- Line/cable constant application examples

1.6 Manual conventions

The following typographical conventions are used in this manual:

Italic: Menus in ATPDraw
E.g.: Select *Edit* | *Rotate* : Select *Rotate* command in the pop-up menu *Edit*
`Courier 9 - 10` : Data files
E.g.: Listing of ATP files, MODELS code, etc.
`Courier 11 - 12` : Data code and file names
E.g.: Give the file the name `HVDC_6.LIB` and store it in the `\USP` directory. The `\USP` directory is a directory under the main directory of ATPDraw.
`Courier 12` : Commands on the DOS prompt
E.g.: `C:\TMP>`**`setup`**: Type the command **`setup`** at `C:\TMP>`.

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 circuit file is a binary file in the Windows version, that can not be edited by text-processors. The default extension of the circuit files is `.CIR`. These files are stored in the `\CIR` sub-directory, but this is completely user selectable.

ATP file: This is the file produced by ATPDraw and can be used for a subsequent simulation as input to ATP. The `.ATP` files are located in the `\ATP` sub- directory and they 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.

Include file: This is the punch file produced by ATP after processing a Data Base Module file. The file must be given a name with extension `.LIB` and stored in the `\USP` directory. A DBM punch file can be included in the ATP file with $Include and is used by all user specified objects in ATPDraw.

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 by specifying new support files. The support files should have a name with extension `.SUP`. The full path is included in the data structure, so a user can store the files anywhere he likes. After installation, the support files of standard components are stored in the `\SUP` directory, the user specified objects in the `\USP` directory, the TACS objects in the `\TAC` directory and the MODELS objects in the `\MOD` directory.

Model file: All MODELS objects have a model file which is a text file containing the actual Model description. This file is included in the `.ATP` file without any modification by ATPDraw. The model file must have the extension `.MOD` and be stored in the `\MOD` sub-directory.

2. Installation Manual . . .

. . . .

ATPDraw™
for Windows
1.0

2.1 How to get the program?

The ATPDraw program can be found anonymously on the ftp sever at `ftp.ee.mtu.edu` (IP: 141.219.23.120) in the directory `/pub/atp/gui/atpdraw`
`Username:` **`anonymous`**
`Password:` your full E-mail address

and on the European mirror server at `ftp.rrzn.uni-hannover.de` (IP: 130.75.2.2) in the directory `/pub/mirror/atp/gui/atpdraw`
`Username:` **`anonymous`**
`Password:` your full E-mail address

For inexperienced Internet users, a short guide about the application of these tools and a summary of ATP related Internet resources are given in section 2.10 of the Installation Manual.

There are two subdirectories under the ATPDraw directory of the above servers: `/ad_win32` for the 32-bit version and `/ad_win16` for the 16-bit version running under Windows 95/NT and Windows 3.x, respectively. Both directories contain a self extracting archive (`ad_win32.exe` and `ad_win16.exe`), a short installation guide and the latest patch file (if any). When you successfully downloaded the distribution kit, follow the installation steps shown below. The installation slightly depends on the version of Windows you are using.

2.2 Program installation under Windows 95/NT

To install ATPDraw under Windows 95/NT perform the following operations:

1) Create a temporary directory.
2) Copy `ad_win32.exe` into that temporary directory.
3) Change current directory to the newly created temporary directory.
4) Run `ad_win32.exe` to unpack the ATPDraw for Windows 95/NT installation kit.
5) Then run the program **`setup.exe`**

The installation process is assisted by an Install Shield Wizard. The user will be requested for the directory or a folder name. As the final step of installation, the Wizard registers ATPDraw for Windows in the Windows' registry. `.CIR` file type is also associated with ATPDraw, so when you click on such a file (e.g. in Windows Explorer) ATPDraw will open it. Uninstall option is also added automatically and it is available either via the *Start menu* | *Settings* | *Control Panel* | *Add-Remove programs* icon of Windows or by clicking the *Uninstall* icon in the ATPDraw folder.

If patch files were also provided on the ftp site, download them and copy their contents into the appropriate ATPDraw directory. When you install ATPDraw to update an older version of the program, some user modified files will be overwritten. To prevent this, you need to make a backup of the files you have changed (circuit files, icon files, support file, library files).

After completing the installation and updating the `atpdraw.exe` with the latest patch file, the ATPDraw directory structure is as follows when typing **`dir`** in the DOS window:

```
   Volume in drive C is ATP_WORK
   Volume Serial Number is 1509-18CF
   Directory of C:\atpdraw

  .              <DIR>        98.02.18   8.28 .
  ..             <DIR>        98.02.18   8.28 ..
  CIR            <DIR>        98.02.18   8.28 Cir
  ATP            <DIR>        98.02.18   8.28 ATP
  SUP            <DIR>        98.02.18   8.28 Sup
  MOD            <DIR>        98.02.18   8.28 Mod
  TAC            <DIR>        98.02.18   8.28 Tac
  USP            <DIR>        98.02.18   8.28 Usp
  ATPDRAW   HLP        75 049  97.06.10  17.08 ATPDraw.hlp
  ATPDRAW   EXE       786 432  98.08.25  12.37 ATPDraw.exe
  ATPDRAW   CNT         2 084  97.06.07  18.08 ATPDraw.CNT
  5 file(s)          863 565 bytes
            8 dir(s)     714 047 488 bytes free
```

2.3 Program installation under Windows 3.x

To install ATPDraw under Windows 3.x, perform the following operations:

1) Create a directory for ATPDraw.
   `C:\>`**`md atpdraw`**
2) Copy `ad_win16.exe` to ATPDraw directory.
   `C:\>`**`copy ad_win16.exe \atpdraw`**
3) Change current directory to the newly created ATPDraw directory.
   `C:\>`**`cd atpdraw`**
4) Run **`ad_win16.exe`** to unpack and install ATPDraw for Windows.
   `C:\atpdraw>`**`ad_win16 -d`**

NOTE: Make sure you specify the -d option to `ad_win16.exe`; otherwise the required subdirectories will not be created! If patch files were also provided on the ftp site, download and unzip them and copy the contents to the appropriate ATPDraw directory.

Having the ATPDraw installed, you can create a program group in Windows as follows:

1) Select *File* | *New* | *Program* group.
2) Specify 'ATPDraw for Windows' as title and `atpdraw.grp` as group file.
3) Select *Ok*.
4) Select *File* |*New* |*Program* item.
5) Specify 'ATPDraw' as Program title, the name and path to `atpdraw.exe` as Command parameter (e.g. **`C:\atpdraw\atpdraw.exe`**) and the ATPDraw directory as Working directory (e.g. **`C:\atpdraw`**).
6) Select *Ok*.
7) To add an icon to the ATPDraw help file, repeat steps 4 to 6 and specify 'ATPDraw Help' as Program title and the path of the help file as Command
   (e.g. **`C:\atpdraw\atpdraw.hlp`**)

Completing the previous installation steps and updating the `atpdraw.exe` with the latest patch file, the ATPDraw directory structure will be as follows when typing **`dir`** :

```
 Volume in drive C is ATP_WORK
 Volume Serial Number is 1509-18CF
 Directory of C:\atpdraw

.              <DIR>        98.02.18   8.41 .
..             <DIR>        98.02.18   8.41 ..
SUP            <DIR>        98.02.18   8.41 SUP
CIR            <DIR>        98.02.18   8.41 CIR
MOD            <DIR>        98.02.18   8.41 MOD
ATP            <DIR>        98.02.18   8.41 ATP
TAC            <DIR>        98.02.18   8.41 TAC
USP            <DIR>        98.02.18   8.41 USP
ATPDRAW  HLP         79 519  97.06.10  17.09 ATPDRAW.HLP
ATPDRAW  EXE        933 632  97.10.19  22.28 ATPDRAW.EXE
         2 file(s)      1 013 151 bytes
         8 dir(s)     713 981 952 bytes free
```

2.4 Hardware requirements under Windows 95/NT

- IBM compatible PC (486 DX4-100 or faster)
- MS-Windows 95, NT 3.51 or later
- XGA screen (1024x768 resolution is suggested)
- 16 MB memory available (32 MB under NT)
- 10 MB free hard disk space

2.5 Hardware requirements under Windows 3.x

- IBM compatible PC (486 DX4-100 or faster)
- MS-Windows 3.x or later
- VGA screen
- 8 MB memory available
- 10 MB free hard disk space

2.6 Configuring ATPDraw

An initialization file (`Atpdraw.ini`) contains customizable program options. Generally, default settings meet most of the user's requirements. When it is required, the `Atpdraw.ini` can either be modified via *Tools* | *Options* menu of the program, or by using a text editor. Default values and supported options are described in the Reference part of this manual.

2.6.1 ATPDraw command line options

On the command line one or more circuit files can be specified to be opened when the program starts. In the example below, the circuit files `my1st.cir` and `my2nd.cir` will be loaded at startup and displayed in separate circuit windows. These files must exist in the directories indicated.

```
C:\ATPDRAW>atpdraw c:\atpdraw\cir\my1st.cir c:\cir\my2nd.cir
```

In Windows 95 or NT 4.0 you can use this property to create shortcuts to the circuit files on your desktop. For instance: Click the right mouse button on an empty space of the desktop and select *New | Shortcut*, then browse and select `ATPDraw.exe`. Specify the 'Target:' properties of the new shortcut as the full path of the program including the name of the circuit file (e.g. **`c:\atpdraw\atpdraw.exe mycir.cir`**), and the 'Start in:' parameter as the circuit file directory (e.g. **`c:\atpdraw\cir`**). The same approach can be used under Windows 3.1 to create a document icon in a program group. From the *File | New | Program item* dialog box, specify the same parameter values as under Window 95/NT, so that when you double-click on the desktop icon, ATPDraw launches and the circuit files are loaded into the circuit window.

2.7 Converting existing .CIR files

A small utility called `CONVERT` is prepared to transform the old DOS version (ATPDraw 3.x) circuit and support files to the new format. The program `CONVERT.EXE` is written in Borland Delphi 1.0, so it equally runs under Windows 3.1 and 95/NT. It is wise to make a backup of old files before attempting to convert your existing data files.

2.7.1 Installation of the CONVERT utility

The autorun archive `CONV.EXE` can be downloaded from the `/gui/atpdraw/convert` directory of the FTP server. `CONV.EXE` contains the program `CONVERT.EXE` which converts the old `.CIR` and `.SUP` files to the new ATPDraw for Windows format. To install it:

1. Copy `CONV.EXE` to the desired CONVERT directory:
2. Change current dir to this directory
3. Run `CONV.EXE`
4. Delete `CONV.EXE`

2.7.2 How to use CONVERT?

Before running the utility the user is requested to specify:

1) The Windows directory: `WinDir` (where ATPDraw for Windows is installed)
2) The DOS directory: `DOSDir` (where ATPDraw for DOS is installed)
3) Which file types to convert.

The following files will be converted, depending on the selected options in the *Convert* checkbox:

Circuit: *(This is recommended to convert)*
After conversion, the ATPDraw for Windows will be able to read the old circuit files. Only `.CIR` files stored in the `DosDir\CIR` directory will be converted.

User specified objects: *(This is recommended to convert)*
Enables ATPDraw for Windows to read your old user specified objects. Only .SUP files stored in the `DosDir\USP` directory will be converted. User created `.LIB` files must be copied to the `WinDir\USP` directory, manually.

Models: *(This is recommended to convert)*
Converts your old model components. Only `.SUP` files stored in the `DosDir\MOD` directory will be converted. `.MOD` files must be copied to the `Windir\MOD` directory manually.

Standard components: *(Not recommended to convert, new standard components are in distribution)*
Conversion of standard components should be selected only when the user has changed the default `.SUP`, `.ICO` or `.HLP` files substantially. Only `.SUP` files stored in the `DosDir\SUP` and `DosDir\TAC` directories will be converted.

Be aware that ATPDraw for Windows store the .SUP, the .ICO and the .HLP files in a single file called .SUP. The TACS object FORTRAN_3 is no longer supported. This object must be replaced with 3 objects of type FORTRAN_1. If a FORTRAN_3 is found in a .CIR file during conversion, it is replaced by a single FORTRAN_1 object with the C-phase FORTRAN string as parameter.

2.7.3 Converting very old (version 2.x) circuit files

If one still has valuable old circuit files created by ATPDraw for DOS version 2.x, it is needed to convert these files to become compatible with ATPDraw for DOS version 3.x before running the `CONVERT.EXE`.

A program called `CIR2-3.EXE` is also located on the FTP server `/pub/atp/gui/atpdraw`. This utility converts the old `.CIR` files to the format used by ATPDraw for DOS version 3.x. The extension of the new format will be: `.CI3`.

1) Make a backup copy of the old .CIR files created by ATPDraw version 2.x
(e.g. type in the `\ATPDRAW\CIR` directory: **`COPY *.CIR *.CI2`**)
2) Copy `CIR2-3.EXE` to the `\ATPDRAW\CIR` directory.
3) Run `CIR2-3.EXE`.
4) Delete the old `.CIR` files. (No risk, you have already got backup copies in the `.CI2` files!)
5) Rename the new files to `.CIR` extension (e.g **`REN *.CI3 *.CIR`**)

Now you can run the Windows conversion utility `CONVERT` as specified in the previous section. In some cases the Windows `CONVERT` generates an error message (and the conversion stops) at files having been converted from the old (version 2.x for DOS) format. To overcome this limitation, load the file first into ATPDraw for DOS 3.2x, then save it unmodified using the same name before attempting to convert it to ATPDraw for Windows format.

2.8 How to get help?

ATPDraw offers a standard Windows help file system. This file provides help on all windows and menus in ATPDraw and assists in building up a circuit. Several links between help pages and a relatively large index register for searching text or phrases are also available. A *Help* button is attached to all circuit objects, which shows a brief overview of the meaning of each parameter. Modification and extension of these help files with users' own remarks are also possible using the built in *Help Editor* in the *Tools* menu.

2.8.1 Help via the Internet

Licensed users of the ATP may subscribe to an e-mail users group, where users share news, provide help to each other, and have better access to the program developers. This server is free and is very easy to subscribe to. For more details, you are advised to read section 2.10 of this manual.

2.8.2 Help from the author of ATPDraw

The author of the program is also available for questions from ATPDraw users, but cannot be made responsible for any problem that they may experince. The address of the author of ATPDraw is :

Dr. Hans Kr. Høidalen
SINTEF Energy Research
7034 Trondheim - NORWAY
http://www.energy.sintef.no
E-mail: hans.hoidalen@energy.sintef.no
Fax: + 47 73597250
Phone: + 47 73597200

The ATPDraw Web page is maintained at address:

`http://www.ee.mtu.edu/atp/atpdraw/ATPDraw.html`

2.9 Running ATP and other utilities from ATPDraw *(in version 1.2 and above)*

The new *Edit batch jobs* feature of ATPDraw supports to add users' own external commands (e.g. *Run ATP / Run LCC / Run TPPlot / Run PCPlot / Run PlotXY / Run PL42mat etc.*) to the existing options under the *ATP* main menu. Using this feature you are requested to specify the name of the batch or executable file and the name of the file which is sent as parameter (e.g. ATP.BAT <current .atp file> or plotxy.exe <current .pl4 file>). You can save these settings into the ATPDraw.ini file, as well, using the *Tools* | *Save Options*. This new feature makes it possible to use ATPDraw as the center of a powerful simulation environment, i.e. from which other components (pre- and post-processors, supporting programs and utilities) of the ATP package are launched.

2.9.1 How to run ATP directly from ATPDraw?

First you must have ATP installed and configured properly. To ensure the smooth interaction between ATPDraw and ATP specify the `ATPDIR` *(Salford ATP)* and/or `WATDIR` *(Watcom ATP)* environmental variable settings in your `AUTOEXEC.BAT` (i.e. SET ATPDIR=Drive:\Path\Salford ATP_program_directory and/or SET WATDIR= Drive:\Path\Watcom ATP_program_directory). Then create a *Run ATP* command using the *ATP* | *Edit batch jobs* submenu by selecting the "current ATP" as *Parameter* and `Drive:\Path\ATP.BAT` as the *Bath file:* name. It is always the user's responsibility to provide the internal commands of that batch files in the correct format.

a) If you are using Watcom ATP, the suggested single line command of `ATP. BAT` is:

`%watdir%tpbigw.exe  /PARAMS %1` *{or* **`/PARAMS disk %1 * -r`***}*

b) If you are using Salford ATP, the `ATP.BAT` file should be as follows:

`run77.exe %atpdir%tpbig.exe /PARAMS %1` *{or* **`/PARAMS disk %1 * -r`** *}*

Under Win95/NT it is wise to set the `ATP.BAT` *Properties* as shown next:

Program:
Cmd line: C:\ATPDRAW\ATP.BAT
Batch file: DBOS_ON.BAT *{leave empty if Watcom ATP is used}*
Run: Normal Window
Close on exit: checked
Font:
Both font types (7x12)
Screen:
Usage: Window

In case *b)* a single line batch file `DBOS_ON.BAT` must be executed before running the TPBIG.EXE. Assuming DBOS ver 3.5 is installed in the directory \dbos3_5 on drive C, the corresponding command must be:

`C:\dbos3_5\dbos /page 800000` *{800000 limits DBOS to grab all extended memory}*

Under Windows 3.1x the commands in the `ATP.BAT` and `DBOS_ON.BAT` can be merged into a single file called `ATP.BAT`:

```
dbos /page 400000
run77 %atpdir%tpbig.exe /PARAMS disk %1 * -r
```

2.9.2 How to execute TPPLOT from ATPDraw?

Because TPPLOT is also a DBOS application, settings are very similar to those specified in *case b)* of the previous section.

When you create a *Run TPPLOT* command using the *ATP | Edit batch jobs* submenu, select "None" as *Parameter* and `Drive:\Path\TPPLOT.BAT` as the *Bath file:* name. `TPPLOT.BAT` must be a single line file again:

```
run77 %atpdir%tpp.exe
```

Properties settings are almost identical with that specified in the previous section, except the Screen usage: It must be set to ***Full-screen***!

Program:
Cmd line: C:\ATPDRAW\TPPLOT.BAT
Batch file: DBOS_ON.BAT
Run: Normal Window
Close on exit: checked
Font:
Both font types (7x12)
Screen:
Usage: Full-screen

The corresponding `DBOS_ON.BAT` command, which is executed before running TPPLOT, is the same as before:

C:\dbos3_5\dbos /page 800000 *{800000 limits DBOS to grab all extended memory}*

Moreover, it is suggested to modify the settings at the end of the `TPPLOT.BEG` file as shown below:

```
C     @LAB2   {comment it out or copy the LAB2 file into your ATPDraw directory}
FILE ATP\*.PL4    {add these two commands to the end of the file}
CHOICE
```

2.9.3 How to execute PCPLOT, PlotXY and LCC from ATPDraw?

First create the corresponding *Run PCPlot, Run PlotXY and Run LCC* user specified commands under the *ATP | Edit batch jobs.* Then click on the "Current PL4" radio button to specify the name of the current pl4 file. which is sent as *Parameter* for the post-processors (PCPlot, PCPlot for Windows, PlotXY, PL42MAT) and "None" for LCC. Finally, *Browse* and select the name of the executable disk file of the corresponding application. When you click the *Update* button, the name of the new batch job will be added to the existing *ATP* menu items.

2.10 ATP related Internet resources

2.10.1 Electronic mail

Electronic mail is the most known feature of the Internet. By this way, anyone who has an account on a computer connected to the Internet can send messages to other users. For ATP users this service provides an easy, efficient and very fast way of communication with other users all over the World, including program developers, regional user group representatives, or the author of ATPDraw.

The ATP-EMTP listserver is an E-mail remailer program, which rebroadcasts incoming messages to all members having been subscribed to the list. It was set up in 1991, by Prof. Bruce Mork at the North Dakota State University, USA. Six years later, the list had more than 600 subscribers. The list is actually maintained by its members, who add themselves, delete themselves and correct their own listing parameters (name, affiliation, country). Anyone who got an ATP license has right to subscribe to the ATP-EMTP Listserver. For subscription, one has to send a single line email to the following address:

```
listserv@listserv.nodak.edu
```

The subject of the message can be left blank. The body of the text is shown next:

```
SUB ATP-EMTP Your Full Name, Institution, Country
```

Name, affiliation and country fields are limited to 39 characters total. Some minutes later the list server program replies with a confirmation letter to the address found in the *From:* filed of the requester's message. It means, that subscription is allowed and accepted only if one subscribes from his own account. After subscription, any messages you send to the address:

```
ATP-EMTP@listserv.nodak.edu
```

will be forwarded to others subscribed to this list. The listserver was set up to support information exchange between ATP users, to provide a facility to help each other, and to have better access to program developers. The list is unmoderated, but a set of common sense rules has to be applied.

The traffic of the ATP-EMTP mailing list is automatically archived by the LISTSERV program on monthly basis. Past mailings are logged into separate disk files, LOGyymm, where "yy" and "mm" are the year and month, respectively. These files are available via the GET command of the listserver. If one sends a message to the address: `listserv@listserv.nodak.edu` containing a single line in the body, e.g.: `GET ATP-EMTP LOG 9703,` after a while, the listserver responds with a message and the requested file (mails delivered during March 1997) will be sent to the requester's e-mail account.

The most important listserver commands are:

SIGNOFF ATP-EMTP	-	Cancel the subscription
REView ATP-EMTP	-	Return e-mail addresses of all subscribers
SET ATP-EMTP options	-	Update your subscription options
INDex ATP-EMTP	-	Send the list of available archive files

2.10.2 The ATP FTP servers

FTP is the method by which users can download or upload files from machines connected to the Internet. The anonymous FTP (or aFTP) permits a user to log-on to a remote computer and access files others have stored in a public directory. If the files are not publicly available, a password is required to have access to files owned by someone else.

By using this service, ATP licensed users can download ATP related files from a remote computer to their personal computer or workstation. Many ATP-related material excepting the source code and the ATP program itself is available over the Internet via FTP. A limited part of this information is accessible via aFTP (ATP license forms, newsletters, upcoming conference/seminar information and the ATPDraw program). To access these files one has to login as "`anonymous`" and to give his personal e-mail address as the password. The majority of ATP related directories are password protected, e.g. the latest revisions of ATP Rule Book pages, utility programs, sample data cases, etc. If you do not know that password, contact your regional user group representative (the person who signed your ATP license agreement). The address of the master FTP site at the Michigan Technological University:

Anonymous ATP- FTP site: `ftp://ftp.ee.mtu.edu/pub/atp`
Secure ATP-FTP directories: `ftp://atp@ftp.ee.mtu.edu/pub/atp`

ATP materials are archived in `.ZIP` files and kept in several subdirectories as shown below:

`/bnchmark`	Benchmark files for testing ATP capabilities
`/canam`	The Can/Am EMTP Users group newsletters
`/conf`	Announcements of EMTP related conferences
`/course`	Announcements for past and future ATP courses&seminars
`/gui/atpdraw`	Graphical preprocessor interface to ATP
`/gui/atpgen`	Graphical preprocessor interface to ATP
`/gui/show`	MS Windows postprocessor (old, unsupported)

`/gui/plotxy`	New postprocessor based on Windows GUI
`/dcase`	Sample data case files, to demonstrate ATP features
`/license/xxxxx`	ATP licensing form for the xxxxx region
`/models/tutor`	MODELS primer by G. Furst and manual by L. Dube
`/models/appl`	Sample MODELS-intensive ATP applications.
`/ruleb`	Latest updated ATP Rule book chapters
`/ruleb/lecruleb`	LEC rulebook chapters, in original Lotus Manuscript format
`/ruleb/wpnofigs`	LEC rulebook chapters, in WordPerfect 5.2, no figures.
`/ruleb/wpwifigs`	LEC rulebook chapters, in WordPerfect 5.2, with figures
`/ruleb/updated`	Updated rulebook chapters
`/theobook`	Latest updated EMTP Theory book chapters
`/util`	Utility programs useful to the ATP user.

The master ATP FTP site is mirrored by some other servers, too. For example in Europe you can find the mirror site at address:

`ftp.rrzn.uni-hannover.de/pub/mirror/atp (IP: 130.75.2.2)`

In Japan the password protected ATP directories are accessible at address:

`http://atp.pwr.eng.osaka-u.ac.jp/~support`

while the publicly available files are at:

`ftp://atp.pwr.eng.osaka-u.ac.jp/pub/atp`

2.10.3 World-Wide Web

Several Web servers have been established mainly for supporting the ATP users in the region. However, originating from the nature of the Web, these services are not limited to the regional users.

The ATP related Web servers can be reached by the following URLs:

USA/Canada:	`http://www.ee.mtu.edu/atp`
Argentina:	`http://iitree.ing.unlp.edu.ar/estudios/caue`
Brasil:	`http://www.furnas.gov.br/atp`
Europe:	`http://www.vmt.bme.hu/eeug`
Japan:	`http://www02.so-net.ne.jp/~m_kan/index-e.htm`

The ATPDraw Web page is available at:

`http://www.ee.mtu.edu/atp/atpdraw/ATPDraw.html`

The ATP bulletin board operating in Japan is a new initiative. The advantage of the bulletin board over the e-mail listserver is that tracking and searching the mail history is more comfortable. The ATP WWWBOARD is located at address:

`http://www.arienter.com/atpwww/`

3. Introductory Manual . . .

. . . .

ATPDraw™
for Windows
1.0

This part of the user’s manual gives the basic information on how to get started with ATPDraw. The Introductory Manual starts with the explanation of how to operate windows and mouse in ATPDraw. The manual shows how to build a circuit step by step, starting from scratch. Then special considerations concerning three phase circuits are outlined.

3.1 Operating Windows

ATPDraw has a standard Windows user interface. This chapter explains some of the basic functionalities of the *Main menu* and the *Component selection menu*, and two important windows: the *Main window* and the *Component dialog box*.

3.2 The Main window

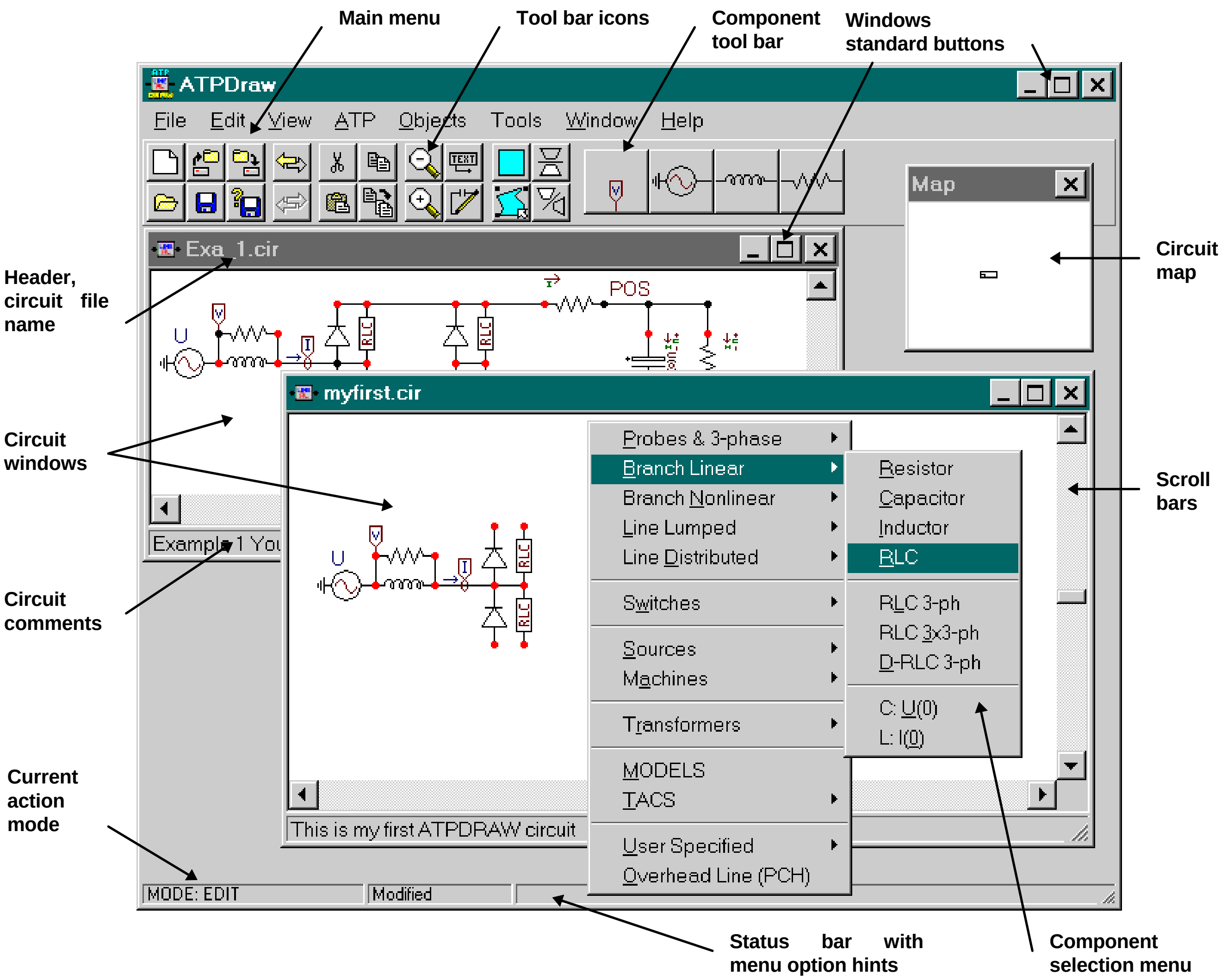


Fig. 3.1 - The Main window. Multiple Circuit windows and the floating Selection menu.

The ATPDraw for Windows program has a functionality similar to the DOS version [1]. The *Component selection menu* is hidden, however, but appears immediately when you click the right mouse in the open area of the *Circuit window*. Fig. 3.1 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 it is indicated by the scroll bars of each circuit window. The *Main window* consists of the following parts:

Header + Frame:
As a standard Windows element, it contains the system menu on the left side, a header text and minimize, maximize, exit buttons on the right side. The main window is resizeable.

System menu:	Contains possible window actions: Close, Resize, Restore, Move, Minimize, Maximize or Resize and Next. The last one exists only if multiple circuit windows are open.
Header text:	The header text is the program name in case of the main window and the current circuit file name in case of the circuit window(s). To move a window, click in the header text field, hold down and drag.
Minimize button:	A click on this button will iconize the main window.
Maximize button:	A click on this button will maximize the window. The maximize button will then be replaced with a resize button. One more click on this button will bring the window back to its previous size.
Corners:	Click on the corner, hold down and drag to resize the window.

Main menu:
The main menu provides access to all the functions offered by ATPDraw. The menu items are explained in detail in the reference part of this manual:

File:	Load and save circuit files, start a new one, import/export circuit files, create postscript and metafile/bitmap files, print the current circuit and exit.
Edit:	Circuit editing:copy/paste/delete/duplicate/flip/rotate, select, move label, copy graphics to clipboard and undo/redo etc.
View:	Tool bar, status bar and comment line on/off, zoom, refresh and view options.
ATP:	Create node names, make ATP file, edit ATP file, ATP file settings (miscellaneous cards and file formats, file sorting etc.), running batch jobs.
Objects:	Edit support files (default values, min/max limits, icon and help file), create new files for MODELS and User Specified Objects.
Tools:	Icon editor, help file editor, text editor, setting of various program options.
Window:	Arranging of the circuit windows. Map window.
Help:	About box and Windows help file system.

Circuit window:
The circuit is built up in this window. The circuit window is the container of circuit objects. From the file menu you can load circuit objects from disk or simply create an empty window to start building a new circuit. Circuit objects include standard ATP components, user specified elements, MODELS and TACS components, connections and relations. To move around in the circuit, you can use the window scrollbars, or drag the view rectangle of the *Map window* to another position.

Component selection menu:
This menu is hidden initially and pops-up only after a right mouse button click in an empty space of the *Circuit window*. In this menu all circuit objects can be selected. After selecting an object in one of the fields or pop-up menus, the object is drawn in the circuit window in marked and moveable mode.

Circuit comments:
A comment line below the circuit window shows a user defined circuit comment text.

MAP window:
This window gives a bird's eye view of the entire circuit. The size of a circuit is 5000x5000 pixels (screen points); much larger than your screen would normally support. Consequently, the *Circuit window* displays only a small portion of the circuit. The actual circuit window is represented by a rectangle in the *Map window*.

Press and hold down the left mouse button in the map rectangle to move around in the map. When you release the mouse button, the circuit window displays the part of the circuit defined by the new rectangle size and position. The map window is a stay-on-top window, meaning that it will always be displayed on the top of other windows. You can show or hide the map selecting the *Map Window* option in the *Window* menu, or pressing the *M* character,

Status bar - Action mode field:
The current action mode of the active circuit window is displayed in the status bar at the bottom of the main window, when the *Status Bar* option is activated in the *View* menu. ATPDraw can be in various action modes. The normal mode of operation is *MODE : EDIT* , in which new objects are selected and data are given to objects. Drawing connections brings ATPDraw into *CONN.END* mode and so on. ATPDraw’s possible action modes are:

EDIT	The normal mode.
CONN.END	After a click on a node, the action mode turns into *CONN.END* indicating that the program is waiting for a left mouse click to set the end-point of a new connection. To cancel drawing a connection, click the right mouse button or press the ESC key to return to *MODE : EDIT*.
MOVE LABEL	Indicates a text label move. Clicking the left mouse button on a text label, then holding it down and dragging it enables you to move the label to a new position. If the text label is overlapped by a component icon, the text label can be moved using *Move Label* in the *Edit* menu. Then the action mode turns into *MOVE LABEL*. Releasing the mouse at the new text label location, the action mode returns to *MODE : EDIT*.
GROUP	Indicates region selection. Double clicking the left mouse button in an empty space of the active circuit window enables you to draw a polygon shaped region. To end the selection, click the right mouse button. Any objects within the selected region are marked then for selection. To cancel region selection, press the *Esc* key.
INFO.START	Indicates the start of a relation when *TACS | Draw relation* is activated in the selection menu. Clicking the left mouse button on a component node or on the end-point of another relation will initiate the drawing of a new relation. Relations are used to visualize information flow into FORTRAN statements and are drawn as blue connections, but do not influence the connections of components. To cancel, click the right mouse button or press the *Esc* key.
INFO.END	Indicates the end of a relation. The program is waiting for a left mouse button click to set the end-point of the new relation. To cancel drawing relation, click the right mouse button or press the *Esc* key.

Status bar - Modified and Hints field:
The middle field of the status bar is used to display the *Modified* state of the active circuit. As soon as you alter the circuit (moving a label, deleting a connection, inserting a new component, etc.), the text 'Modified' appears, indicating that the circuit must be saved before exit. The field will be empty when you save the circuit or undo all modifications. The rightmost field of the status bar displays the menu option hints.

3.3 The Component dialog box

After selecting a component in the *Component selection menu* the new circuit object appears in the middle of the circuit window enclosed by a rectangle. Click on it with the left mouse button to move, or the right button to rotate, finally click in the open space to unselect and place the object. The object input window appears when you click the right mouse button (or double click with the left button) on a circuit object. Assuming you have clicked on the icon of an RLC element, a dialog box shown in Fig. 3.2 appears.

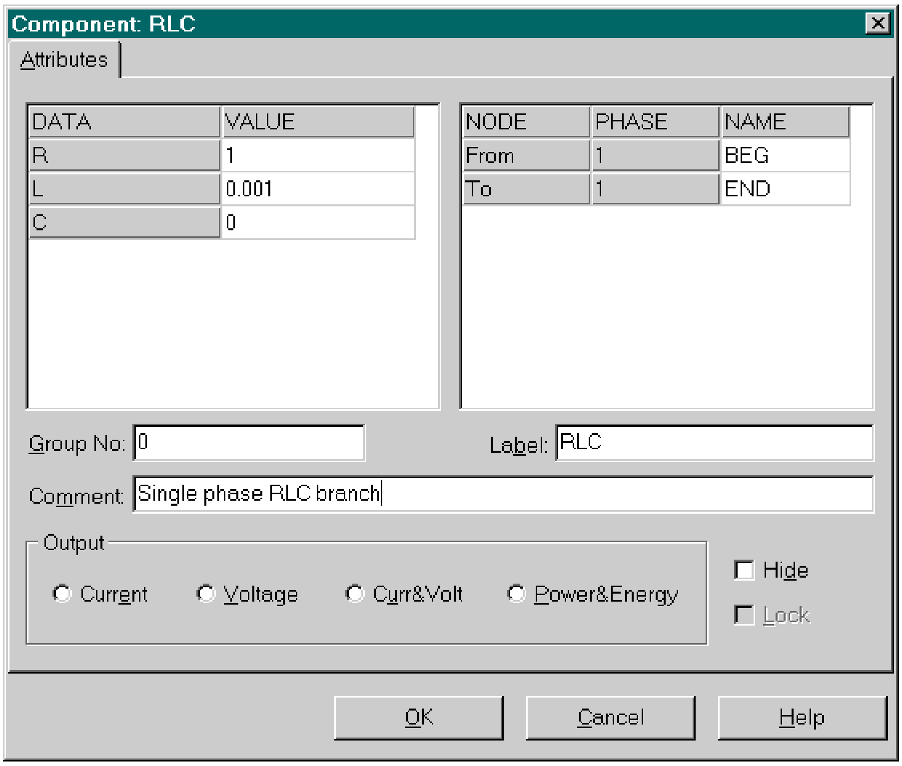


Fig. 3.2 - The Component dialog box.

The *Component* dialog box has the same layout for all circuit objects. In this window the user must specify the required component data. The number of DATA and NODES menu fields are the only difference between input windows for standard objects. The nonlinear branch components have a *Characteristic* page too, in addition to the normal *Attributes* page, where the nonlinear characteristics and some include file options can be specified.

Carriage return, Tab or the mouse can be used to move the cursor between input fields. The arrow keys can be used to move the cursor inside of a menu field. When the cursor is moved out to the right side of the field the menu content is scrolled.

Numerical values in the data input windows can be specified as real or integer, with an optional exponential integer, identified by 'E' or 'e'. An "Illegal numeric specification" message is produced, when the characters are non-numeric. Legal formats are: 3.23e4, 323E+02, 32300, 32.3e+3 etc.

Input texts in the node input menus can be specified with any characters (remember that characters like * - + / $ etc. should not be used in ATP node names, also avoid space and lower case letters). The user does not need to give names to nodes, in general. The name of the nodes without special interest are recommended to be left unspecified. ATPDraw will then give a unique name to those nodes. The node dot of these nodes are displayed in red color in the circuit window.

Below the data input column there is a *Group No* input field. This is an integer field where an optional group number can be specified to the object, which could be used as a sorting criteria (the lowest group number will be written first into the ATP file).

Below the node input column there is a *Label* input text field. The content of this field is written on the screen and also into the circuit file. The label text is movable. The component dialog box has a *Comment* input text field. If you specify a text in this field, it will be written to the ATP file as a comment (i.e. as the first line of the object's data).

The radio buttons of the *Output* group specify the branch output requests. If the *Hide* box is selected, the object becomes hidden (which means that it is not written to the ATP file) and its icon becomes light gray in the circuit window. The *Lock* option is not yet implemented in the present version of the program.

The *OK* button will close the dialog box and the object's data and its properties are updated in the data structure. The red drawing color which indicates that no data is given to the object will be turned off. When you click on the *Cancel* button, the window will be closed without updating the object's data. The *Help* button invokes the *Help Viewer* showing the help text of the object. For obtaining further help press the *F1* key.

3.4 Operating the mouse

This chapter contains a summary of the various actions taken dependent on mouse operations. The left mouse button is generally used for selecting objects or connecting nodes; the right mouse button is used for specification of object or node properties.

Left simple click:
- On object:
  - Selects object (also connection).
  - If the *Shift* key is pressed, the object is added to the current group.
- On object node:
  - Begins to draw a connection.
  - Moves the mouse to the end node, left click to place, right to cancel.
- In the open area of the circuit window:
  - Unselects object.

Right simple click:
- On object node:

Opens the node dialog box.
On unselected object:
Opens the component dialog box.
On unselected object, when you hold down the *Shift* key:
Opens the circuit window shortcut menu.
On selected object(s):
Rotates object(s).
In the open area of the circuit window:
Cancels connection made.
Left click and hold:
On object:
Moves object(s).
On node:
Resizes connection (it is often necessary to select connection first).
In the open area of the circuit window:
Draws a rectangle for group selection.
Objects inside the rectangle become a group when the mouse button is released.
Left double click:
On object node:
Performs the *Node dialog box*.
On unselected object:
Performs the *Component dialog box*.
On selected object:
Performs a *Group Number* specification window.
In the open area of the circuit window:
Starts the group selection facility. Click left to create corners in an enclosing polygon, click right to close. Objects inside the polygon become a group.

3.5 Edit operations

ATPDraw offers the most common edit operations like copy, paste, duplicate, rotate and delete. The edit options operate on a single object or on a group of objects. Objects must be selected before any edit operations can be performed. Selected objects can also be exported to a disk file and any circuit files can be imported into another circuit.

Tool	Shortcut key	Equivalent in menus
Copy	Ctrl+C	*Edit* | *Copy*
Paste	Ctrl+V	*Edit* | *Paste*
Duplicate	Ctrl+D	*Edit* | *Duplicate*
Rotate	Ctrl+R	*Edit* | *Rotate* (or right click)
Flip	Ctrl+F	*Edit* | *Flip*
Group	Ctrl+G	*Edit* | *Select group* (or left double click in open space)
All	Ctrl+A	*Edit* | *Select All*
Label	Ctrl+L	*Edit* | *Move Label*
UNDO	Alt + BkSp	*Edit* | *Undo*
REDO	Shft+Alt+BkSp	*Edit* | *Redo*
Zoom In/Out	+ / -	*View* | *Zoom In / Out*
Zoom window	Z	*View* | *Zoom*

3.6 Overview of the operation of ATPDraw

From the *Component selection menu* you select components to insert into the circuit. This menu pops up when you click the right mouse button in an empty area of the circuit window. To select and move an object, simply press and hold down the left mouse button on the object while you move the mouse. Release the button and click in an empty area to unselect and confirm its new position. The object is then moved to the nearest grid point (known as gridsnapping). If two or more components overlap as a consequence of a move operation, you are given a warning message and can choose to proceed or cancel the operation.

Selecting a group of objects for moving can be done in three ways: If you hold down the *Shift* key while you left-click on an object, you add it to the selected objects group. Pressing and holding down the left mouse button in an empty area enables you to drag a rectangular outline around the objects you want to select. And finally, if you double-click the left mouse button in an empty area, you can define a polygon shaped region by repeatedly clicking the left mouse button in the circuit window. To close the region, click the right mouse button on the last polygon point you want to set. Objects that are defined to fall within the indicated region or rectangle are added to the selected objects group. For components this means that the center point of a component icon must lie within the defined region or rectangle. For connections and relations the region or rectangle must surround both end-points. To move the selected group of objects, press and hold down the left mouse button inside the group while you move the mouse. You unselect and confirm the new position by clicking in an empty area. Any overlapping components will produce a warning.

To move objects outside the visible part of the circuit, use the window scrollbars or the view rectangle in the map window. Any selected objects or group will follow the window to its new position.

Objects or group can be rotated by clicking the right mouse button inside the selected object or group. Other object manipulation functions, such as undo/redo and clipboard options can be found in the *Edit* menu. Additionally, the most frequently used object manipulation functions can be accessed by holding down the *Shift* key while clicking with the right mouse button on an object or on a selected group of objects. This will display and activate the circuit window shortcut menu.

Components and component nodes can be opened for editing. If you right-click or double-click an unselected component or node, either the *Node data*, *Component* or *Open Probe* dialog box will appear, allowing you to change component or node attributes and characteristics.

If you select a single component and press the *Crtl+F1* key combination, the component specific help is displayed. If you double-click on a selected group of objects, the *Open Group* dialog box will appear, allowing you to change attributes common to all components in that group, such as group number and hide and lock state. Default component attributes are stored in support files. Access to create and customize support files is provided by the *Objects menu*.

Components are connected if their nodes overlap or if a connection is drawn between the nodes. To draw a connection between nodes, click on a node with the left mouse button. A line is drawn between that node and the mouse cursor. Click the left mouse button again to place the connection (clicking the right button cancels the operation). The gridsnap facility helps overlapping the nodes. Connected nodes are given the same name by the *Make Names* and *Make File* options in the ATP menu. Nodes can be attached along a connection as well as at connection end-points. A connection should not unintentionally cross other nodes (what you see is what you get). A warning for node naming appears during the ATP file creation if a connection exists between nodes of different names, or if the same name has been given to unconnected nodes. Connections can be selected as any other objects. To resize a connection, click on its end-point with the left mouse button, hold down and drag. If several connections share the same node, the desired connection to resize must be selected first. Selected connection nodes are marked with squares at both ends of the selection rectangle.

Relations are used to visualize information flow into Fortran statements and are drawn as blue connections, but have no influence on components connectivity. Relations are drawn in the same way as drawing a short circuit connection between nodes, except that you have to select the *TACS | Draw relation* option in the component selection menu to start the relation drawing. You can then draw multiple relations until you click the right mouse button or press *Esc* key.

3.7 Your first circuit *(Exa_1.cir)*

This chapter describes how to use ATPDraw step by step. As an example, composing the circuit file of a single phase rectifire bridge (see Fig. 3.3) is presented. Reading this tutorial carefully, you will be proficient in the use of 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?

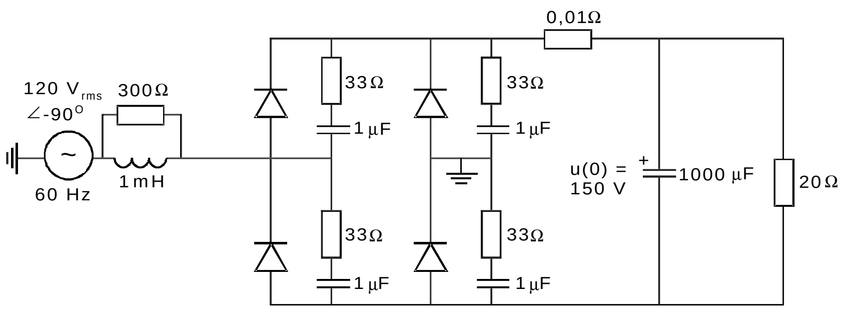


Fig. 3.3/a - Single phase rectifier bridge.

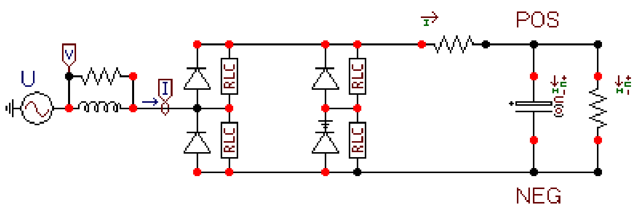


Fig. 3.3/b - Your first circuit (`EXA_1.CIR`).

The circuit is a single phase rectifier bridge, supplied by a 120 V_{rms}, 60 Hz source. The source inductance is 1 mH in parallel with a damping resistor of 300 Ω. The snubber circuits across the rectifying diodes have a resistance of 33 Ω and a capacitance of 1 µF. The smoothing capacitor is 1000 µF and the load resistor is 20 Ω. The example has been taken from [2], exercise 1.

The units given in Fig. 3.3/a are based on settings of Xopt and Copt equal to zero as will be explained later.

The circuit in Fig. 3.3/b has been chosen since its construction involves the most commonly used edit operations.

3.7.1 Building the circuit

Most parts of the building process will be demonstrated in this chapter, along with the explanation of correcting possible drawing errors. 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*.

3.7.1.1 Starting to create a new circuit

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.

3.7.1.2 Source

First an AC source is selected from the floating *Component selection menu*, which appears with a right mouse click on open area of the circuit window. Fig. 3.4 shows how to select a single phase sinusoidal voltage source (*Sources | AC type 14*) using the mouse.

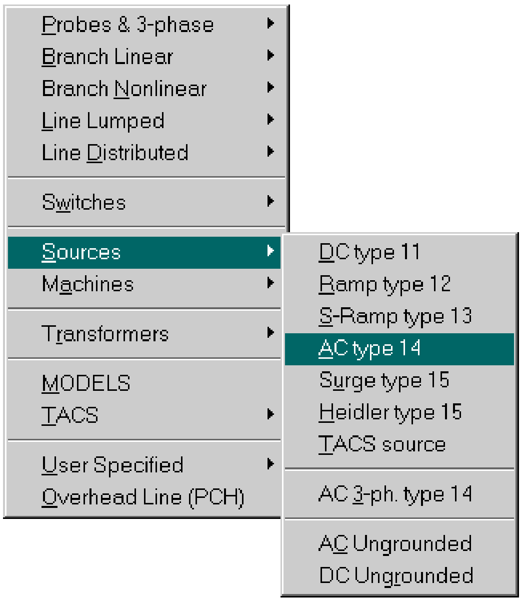
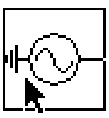


Fig. 3.4 - Selecting an AC source.

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 button**, 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.

Next select the source inductance as shown in Fig. 3.5:

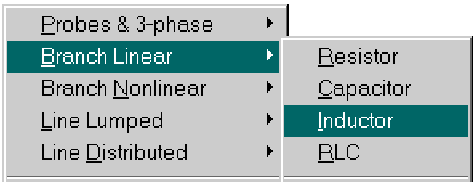


Fig. 3.5 - Selecting an inductor.

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. 3.6:

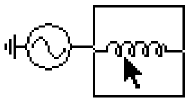


Fig. 3.6

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 component position is rounded to the nearest 10^{th} pixel.

The inductor in Fig. 3.6 should be placed so that the node of the inductor touches the source. Objects having overlapping node dots will automatically be connected.

The next figure shows a situation where the inductor component has been misplaced. In this situation the objects are disconnected. To correct this, a connection could be drawn between the objects as will be explained later. In this example you are supposed to place the inductor so that its left node overlaps the AC source node. To move the inductor, follow the instructions given in Fig. 3.7.

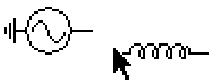


Fig. 3.7 - Error! Click on the object with the left mouse button, hold down and drag it to the proper position, then click on white space.

If you have placed the inductor in the correct position, you can select the damping resistance. After you have clicked in the *Resistor* field of the component selection menu, a resistor icon 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. 3.8. Click on open space with the left mouse button to place it at the position shown in Fig. 3.8.

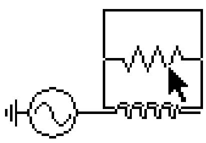


Fig. 3.8

This resistor is supposed to be parallel with the inductor and two connections to ensure this will be drawn later. The resistor in Fig. 3.8 would also be recognized as in parallel with the inductor, if it had been placed in a position completely overlapping the inductor. This tricky way is not recommended however, since the readability of the drawing is strongly reduced.

We want to measure the source current flowing into the diode bridge. To be able to do so, you can add a measuring switch. A special multi-phase current probe is available for such measurements in ATPDraw. When using this object, you are requested to specify the number of phases and in which phases the current should be measured. Select the probe as shown in Fig. 3.9.

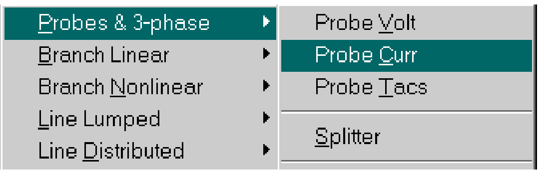
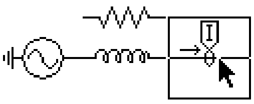


Fig. 3.9 - Selecting current measuring probe.

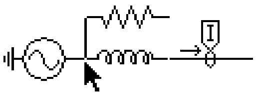


After you have clicked in the *Probe Curr*. field, the selected probe appears in the circuit window enclosed by a rectangle. Click on it with the left mouse button, hold down and drag it to a position shown in the figure, then place it.

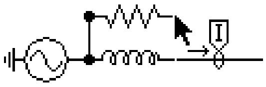
At this stage of the building process it is time to draw some connections in the circuit diagram. To draw a connection you just click the left mouse button on a node, release the button and move the mouse. The cursor style now changes to a pointing hand and a line is drawn between the starting position and the current mouse position (the action mode now is *MODE : CONN.END* indicating that the program is waiting for the end point of the connection). Click with the left mouse button again to place the connection or click with the right button to cancel the starting point.

Drawing the two connections required to parallel connecting the source inductance and the damping resistor is shown below.

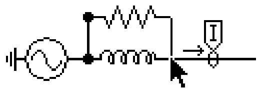
Click left
Release + move



Click left



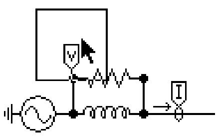
Click left
Release + move



Click left

The connections are always drawn with node dots if the *Node dots* is **On** in the *View | Options* menu.

The last object we want to introduce in the source part of the circuit is a voltage measuring probe which results in an output request for the node voltage in the ATP file. The voltage sensor can be selected via the *Probes & 3-phase | Probe Volt* in the component selection menu (see Fig. 3.9). The probe is drawn in the middle part of the circuit window in marked and moveable mode.



Use the left mouse button to drag and place the object as shown on the figure to the left.

When you place an object by clicking on open area of the circuit window, you will sometimes receive a warning message as shown in Fig. 3.10.

This message appears if a center of one of the permanent objects is inside the enclosing polygon of a marked object (or more general; a group of objects). This is to prevent a disaster if the user presses the left mouse button unintentionally, while moving the object. Normally it is OK to click on *Yes* in this situation. If you click on *No*, the object is not placed but continues to be moveable.

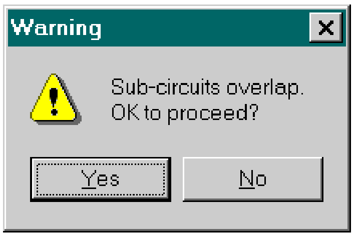


Fig. 3.10 - Overlap warning.

The circuit objects used in the circuit so far were drawn 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 rectifier. To do so: simple click with the **right mouse button** (or double click with the left button) on an object. When you click with the right button on the AC source icon, a window shown in Fig. 3.11 appears.

Fig. 3.11 shows the window after the values for the circuit in Fig. 3.3/a have been specified. The names of the numerical data menus are strongly related to the names used in the ATP Rule Book [3].

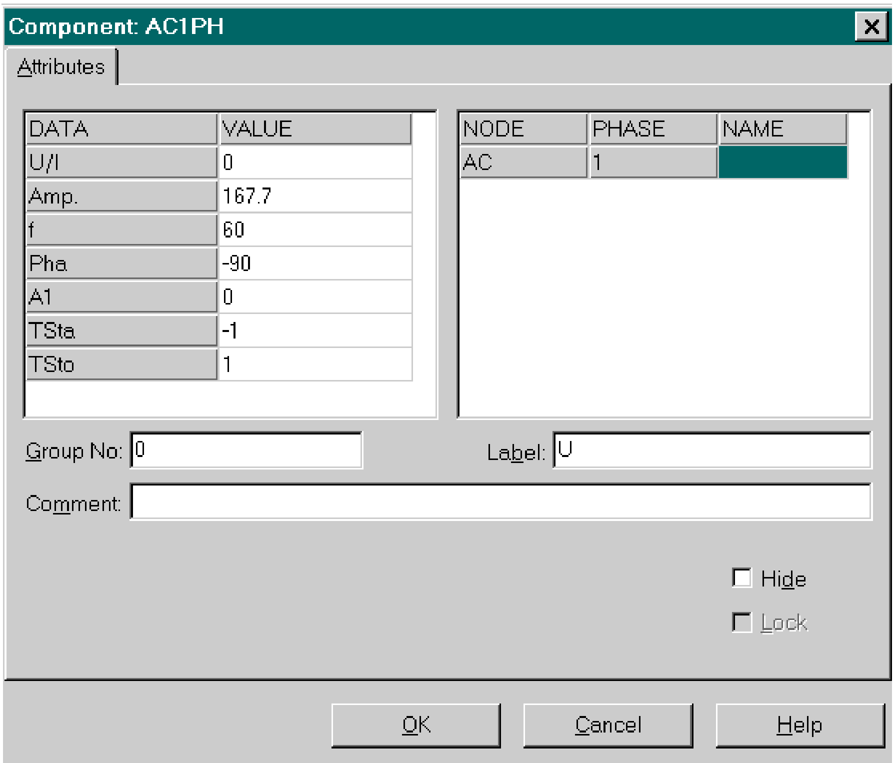


Fig. 3.11 - AC1PH input window.

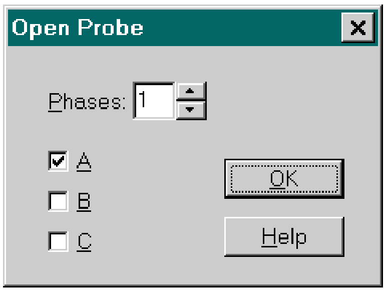
The AC source has 7 input data and one node. The data correspond to the required ATP data. Click on *HELP* to load a help file. This file explains the meaning of each input data and node.

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. 3.11. 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.

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

The probe objects have different input window than other objects.

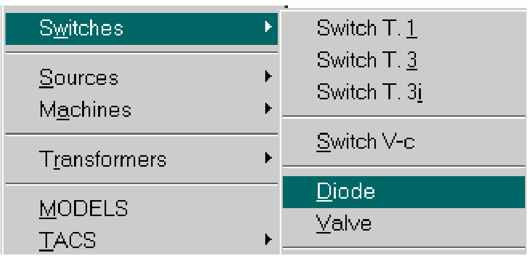


To open the voltage or current probe input window, click on its icon with the right mouse button. In this window you can select the number of phases of the probe and which phases to monitor. In this single phase example default values (no. of phases=1, monitored phase=A) of both voltage and current probes should be selected, as shown in Fig. 3.12

Fig. 3.12 - Open probe dialog box.

3.7.1.3 Diode bridge

In this process you will learn to use some editing options like rotate, group, duplicate and paste. Since the diode bridge consists of four equal branches, you do not need to build all of them from scratch. First you select a diode from the selection menu as shown in Fig. 3.13. After you have clicked on *Diode* the diode appears in the circuit window in marked, moveable mode and enclosed by a rectangle.



The diode has to be rotated, so click the right mouse button or select *Edit* in the main menu and click on *Rotate*. The diode is now rotated 90 deg. counter clock-wise. Click on the diode with the left mouse button, hold down and drag to the position shown in Fig. 3.14.

Fig. 3.13 - Selecting a diode.

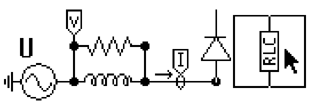


Fig. 3.14

Click with the left mouse button on empty area to place the diode. Remember the grid snap facility and the overlap warning.

Next you must select the snubber circuit across the diode. In this example the snubber circuit is a resistor and a capacitor in series. Select an RLC object from the component selection menu (Fig. 3.5).

Click on the selected RLC branch with the right mouse button to rotate, then click with the left button, hold down and drag the RLC branch to be in parallel with the diode. Click on the left mouse button to place.

The idea is further to copy the diode and the RLC 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 RLC or diode icon with the right mouse button activates the component dialog box to give data to objects.

Again an explanation of the input parameters is given in a help file. Press the *HELP* button to see this help file. The numerical values of the diode are all zero, meaning that the diode is ideal and is open during the steady state. The RLC branch in Fig. 3.14 has been given a resistance of 33 Ω and a capacitance of 1 μF.

You have now given data to the diode and the RLC branch, and instead of repeating the drawing and data giving process four times you can use the copy facility. First you have to select a group of components. This can be done by selecting *Edit | Select group* field in the main menu or with a double click with the left mouse button on an empty space of the *Circuit window*. Then cursor style changes to a pointing hand and the action mode is *EDIT : GROUP*. The process is then to click with the left mouse button to create a corner in a fence and to click the right button to enclose the fence (polygon). All components having their center inside the fence are included in the group.

Alternative way of group selection is to draw 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.

You can follow the procedure shown in Fig. 3.15.

Fig. 3.15 - Drawing a polygon: First double click on white space, click the left mouse button at each corner of the polygon, then click the right button to enclose the polygon.

The group created in Fig. 3.15 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 fencing polygon is now a rectangle. The pasted group is moveable, so you can click on it with the **left mouse button**, hold down and drag to a desired position. Click the left mouse button on open space to put the group in the position shown in Fig. 3.16.

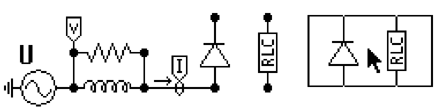


Fig. 3.16 - Move a group.

Only the enclosing polygon is drawn during a move operation. The objects are drawn when the mouse button is released. If you misplaced the group you can mark it again with *Edit | Select group*. *Undo* and *Redo* facilities are also available vie the main menu *Edit.*

You can now paste an other copy of the diode/RLC group into the circuit. Since the duplicate facility has already copied the group to the clipboard, you can just select the *Paste* option from the *Edit* menu by using the mouse or pressing *Ctrl+V*, or selecting the *Paste* icon from the *Toolbar*. The pasted group is drawn on top of the original one enclosed by a rectangle. Click on this group with the left mouse button, hold down and drag it to a position shown in Fig. 3.17:

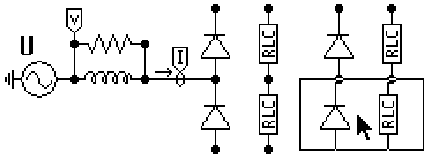


Fig. 3.17

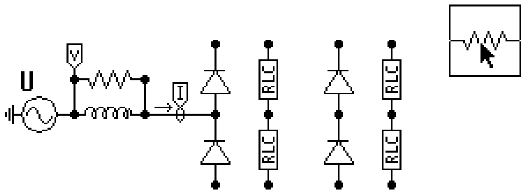


Fig. 3.18

As part of the connection between the rectifier bridge and the load a small resistor is included in Fig. 3.3/a. The resistor is included to demonstrate the option of using a small resistor for current measurement purposes.

Select a resistor in the component selection menu, click on the resistor with the left mouse button, hold down and drag it to a desired position as shown in Fig. 3.18. You must place the resistor precisely, because the next step is to connect the top nodes of the diode bridge with the resistor.

But first, give data to this resistor opening the component dialog box by a right-click on the resistor. Specify data values: RES= 0.01 Ω and *Current* checked under *Output* to get current output in the subsequent ATP run. Having closed the component dialog box a small $\vec{I}$ symbol appears on the top-left side of the resistor indicating the current output request.

Now you can start to connect the diode bridge and the resistor together. The procedure is to first click with the left mouse button on a starting node, as shown in Fig. 3.19. The cursor style now changes to a pointing hand and the action mode is *MODE : CONN.END.* Then release the mouse button and move the mouse (a rubber band is drawn from the starting point to the current cursor position). To place a connection, click on the left mouse button again. Click on the right button or press *Esc* to cancel the connection operation.

The connection drawn in Fig. 3.19 picks up intermediate nodes, so all the five nodes will be connected together. In this way, ATPDraw suits the requirement: "What you see is what you get" and the amount of required connections are significantly reduced.

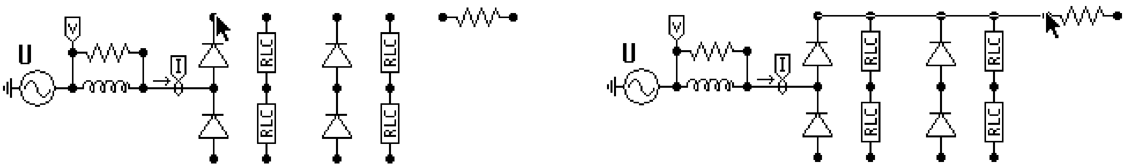


Fig. 3.19 - Click left button. Release + move, then click left button to place the connection.

If you made a mistake in the connection drawing process, you can correct the error easily, because connections are editable (copy/move/rotate) as any other objects. If you would like to correct/modify a misplaced connection click on it with the left mouse button. After this selection the connection is enclosed by a rectangle and two squares replace node dots at the end of the line. To move the connection, click on an internal point of it using the left mouse button, then hold down and move, and release the mouse at the correct position. To reposition a connection, click on the node squares with the left button and stretch the connection as illustrated in Fig. 3.20:

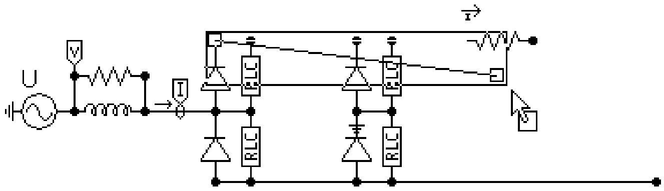


Fig. 3.20 - Edit connection. Click any point of the line then click node squares and stretch.

3.7.1.4 Load

The last part of this example circuit is the load consisting of a smoothing capacitor with initial condition and a load resistor. First you can select the capacitor as shown in Fig. 3.21:

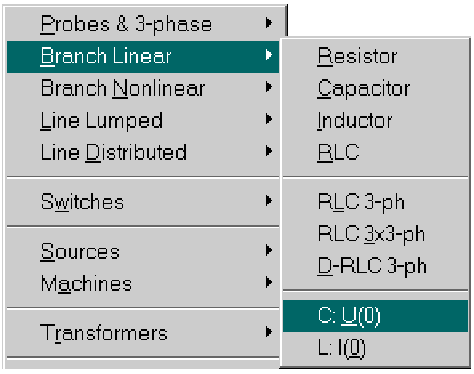
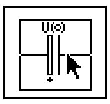


Fig. 3.21 - Select capacitor with initial condition.



After this selection the capacitor appears in the middle of the circuit window in moveable mode enclosed by a rectangle. Click on the capacitor with the left mouse button, hold down and drag to a desired position, then click the right mouse button 3 times (or press *Ctrl+R*) to orient the capacitor as shown in Fig. 3.22. Finally click on open space to place the capacitor.

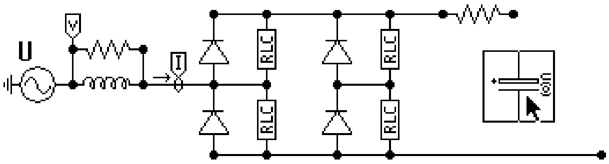


Fig. 3.22 - Placing a capacitor with initial conditions.

Next select the load resistor in the component selection menu *Branch linear + Resistor*. The resistor is drawn in moveable mode in the circuit window. Select *Edit + Rotate* to rotate the resistor. Click on it with the right mouse button to rotate, then click with the left mouse button, hold down and drag it to a desired position and place as shown in Fig. 3.23.

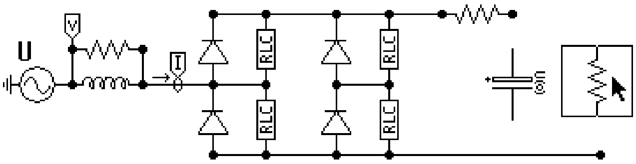


Fig. 3.23 - Place load resistor.

The time has come to connect the load to the rest of the diode bridge. The process has been explained before. Click on the component nodes you wish to connect with the left mouse button, sequentially. A left mouse click on open area while in *MODE: CONN.END* generates a new node dot, that can be used as the starting point of any new connections. This way creating a circuit having only perpendicular connections (recommended for complex circuits, to improve the circuit readability) is a relatively simple task, as shown in Fig. 3.24.

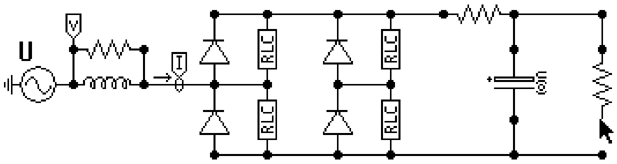


Fig. 3.24 - Your first circuit is almost ready!

After you have finished connecting the source side and the load side of the circuit, you can specify the load data. Click with the right mouse button on the capacitor and specify the parameters shown in Fig. 3.25:

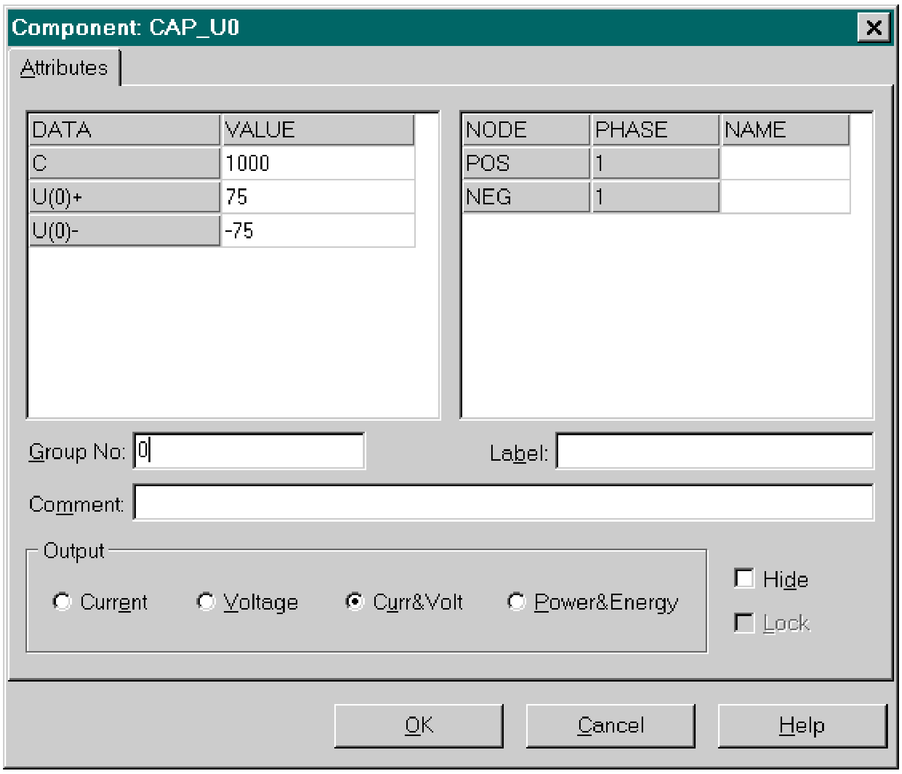


Fig. 3.25 - Capacitor data with initial condition.

The capacitance is 1000 μF (if Copt=0 in *ATP* | *Setting* | *Simulation*). The positive node has an initial voltage of 75 V and the negative -75 V. Both branch current and voltage will be calculated, so the *Curr&Volt* radio button is selected in the *Output* field. Once all the entries in the component dialog box are completed, select the *OK* button to close the window and update the object values or click *Help* to obtain an on-line help. To cancel the description/modification of the element, select the *Cancel* button. When you close the dialog box, a small $\overset{+\,U\,-}{\rightarrow\,I}$ symbol appears on the top-left side of the capacitor, indicating the branch voltage and the current output requests.

Next click with the right mouse button on the load resistor to get the input window and specify the load resistance to be equal to 20 Ω. Both branch current and voltage will be calculated so the small $^{+U-}_{\rightarrow I}$ symbol appears again on the top-left side of the resistor after leaving the dialog box.

3.7.1.5 Node names and grounding

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. 3.26 - Fig. 3.29 show how to give data to four different nodes.

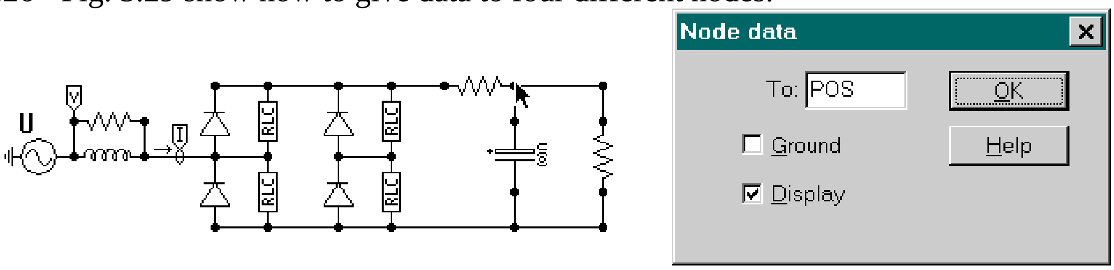


Fig. 3.26 - Click on a node with the right mouse button and specify a name in the dialog box.

When you exit the window in Fig. 3.26 by clicking *OK* , the circuit is updated as shown in Fig. 3.27. All node names are forced to be left adjusted, and as a general rule in the ATP 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.

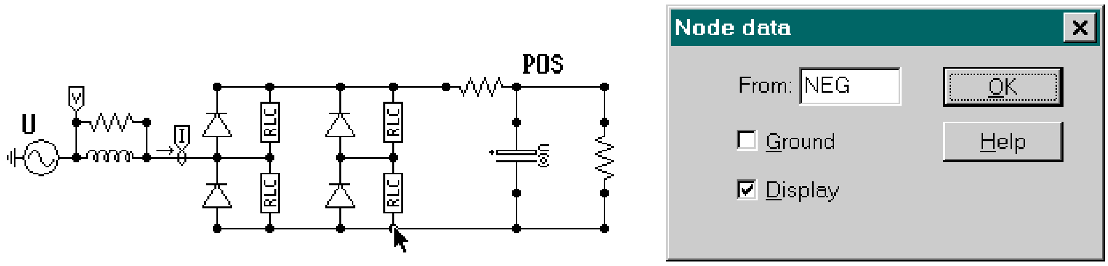


Fig. 3.27 - Click on a node with the right mouse button and specify a name in the node data window. The name ‘NEG’ will be assigned to all nodes visually connected.

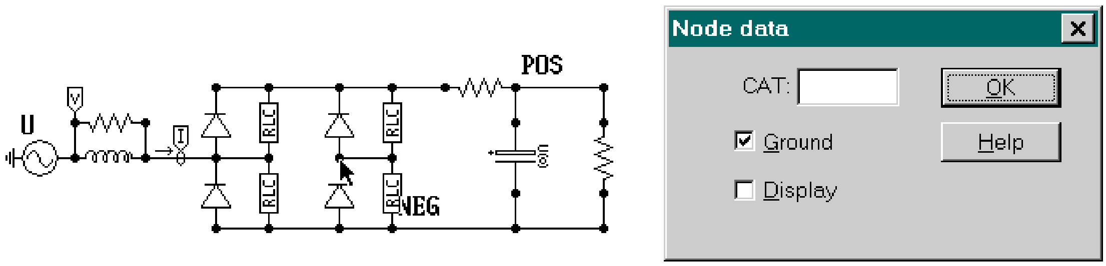


Fig. 3.28 - Click on a node with the right mouse button and check the *Ground* box indicating that the node is connected with the ground reference plane of the circuit.

The ground symbol is drawn at the selected node when you exit the window as Fig. 3.29 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 (i.e. from the program) are distinguished by red color from the user specified node names, as shown in Fig. 3.29.

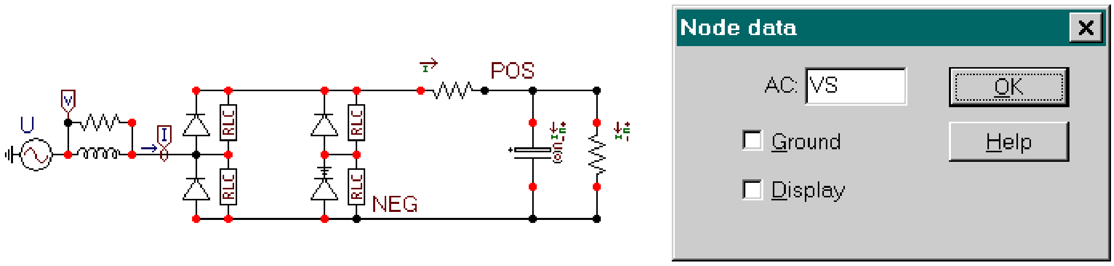


Fig. 3.29 - Click on the voltage source node with the right mouse button and specify the source node name.

3.7.2 Storing the circuit file on disk

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.

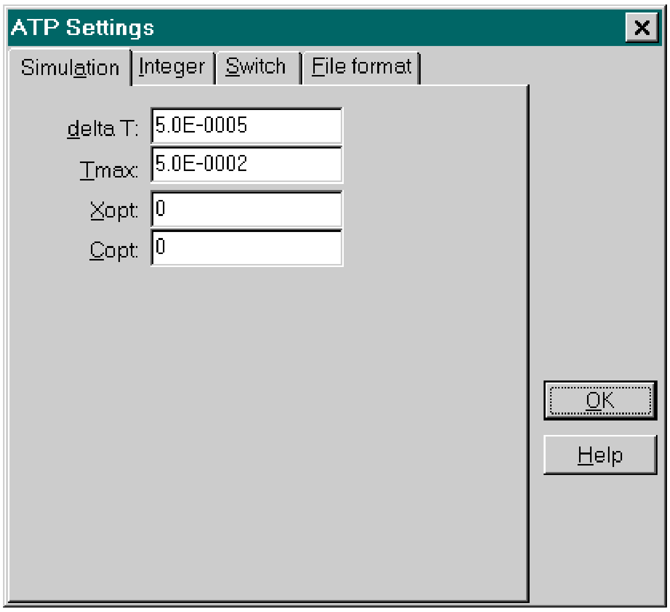
When the circuit once was saved, the name of the disk file appears in the header field of the circuit window. Then if you hit *Ctrl+S* or press the *Save circuit* icon in the Toolbar, the circuit file is updated immediately on the disk. The *File + Save As* option or the *Save As* icon from the Toolbar allows the user to save the circuit currently in use under a name other than that already allocated to this circuit.

3.7.3 Creating ATP file

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

Before you create the ATP file, you must specify some miscellaneous parameters (i.e. parameters, that are printed to Misc. Data card(s) of the ATP input file). 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. 3.30 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.



Select:

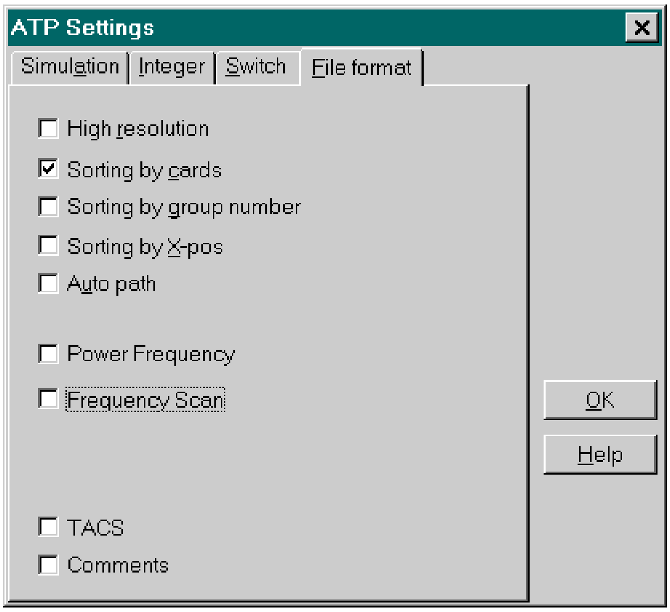
- Time step ΔT in sec.
- End time of simulation Tmax in sec.
- Xopt=0: Inductance in mH.
- Copt=0: Capacitance in μF.

Press *Help* to get more information or *OK* to close the dialog box.

The simulation settings are stored in the circuit file, so you should save the file after changing these settings.

Fig. 3.30 Simulation 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*.



Under the *File format* page the user can select precision mode and the ATP-file sorting criteria. The main characteristic of the simulation (time domain or frequency scan) can also be set on this page. If you select the *File format* page, the window shown in Fig. 3.31 appears:

Select:

☑ Sorting by cards: First /BRANCH, then /SWITCH and then /SOURCE.

All other check boxes are unselected

Fig. 3.31 - The file format menu.

To create an ATP 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 appears, where you can specify the name and path of the ATP 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 (`EXA_1.ATP`) you just have created will look as follows:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  Sat 3. May - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
C Miscellaneous Data Card ....
C  dT  >< Tmax >< Xopt >< Copt >
  .00005     .05
     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
  VC     XX0002               33.           1.                                 0
         XX0002               33.           1.                                 0
  NEG    VC                   33.           1.                                 0
  NEG                         33.           1.                                 0
  XX0002POS                  .01                                               1
  POS    NEG                             1000.                                 3
  POS    NEG                  20.                                              3
  VS     XX0025                        1.                                      0
  VS     XX0025              300.                                              0
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
11VC     XX0002                                                                0
11       XX0002                                                                0
11NEG    VC                                                                    0
11NEG                                                                          0
  XX0025VC                                                 MEASURING           1
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
14VS      0      167.7       60.       -90.                        -1.        1.
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
 2POS     7.500000E+0001
 2NEG    -7.500000E+0001
 3POS    NEG                    1.500000E+0002
  VS
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

You can edit this file or just display it by selecting the *ATP | Edit file* menu.

3.7.4 Running ATP simulation *(in version 1.2 and above)*

ATPDraw allows to execute 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.*

Section 2.9.1 of the Installation Manual describes in detail how to create a *Run ATP* command with a functionality to perform an ATP simulation. Assuming `ATP.BAT` is the name of the batch file which invokes `TPBIG.EXE`, and the "current ATP" option has been selected as *Parameter*, ATPDraw executes the following command when you click *Run ATP*:

```
ATP.BAT <Drive:\path\current_atp_file>
```

3.8 Three phase circuits *(Exa_2.cir)*

Both single phase circuits and three phase circuits are available in ATPDraw. For all three phase objects the number of phases are indicated in the selection menu. The three-phase objects also have a 3D layout in the circuit window and the icon consists of thick lines from the nodes into the object symbol.

All three-phase nodes have only 5 characters available in the input windows. ATPDraw adds the extension *A*, *B* and *C* at the end of the node name. By default, the phase sequence is *ABC*; the first data card uses *A*, the second *B* and the last *C*. The only way to change the phase sequence is to use the available transposition objects (`TRANSP1 - TRANSP4`) selectable under *Probes & 3-phase* in the component selection menu.

The current phase sequence is displayed in the bottom of the node input window after an *ATP | Make File* or *ATP | Make Names* selection has been made. The following example illustrates the usage of three-phase objects:

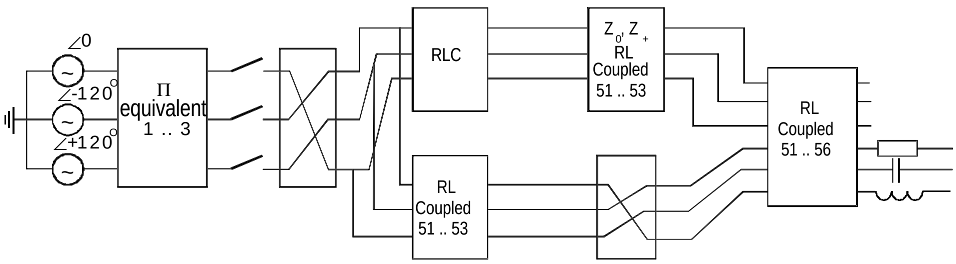


Fig. 3.32/a - Illustrative three-phase circuit.

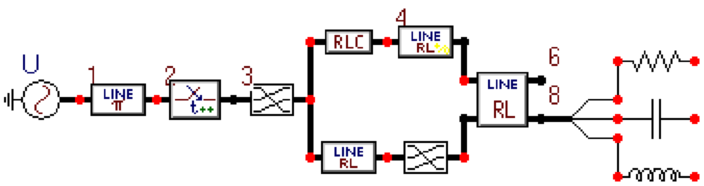
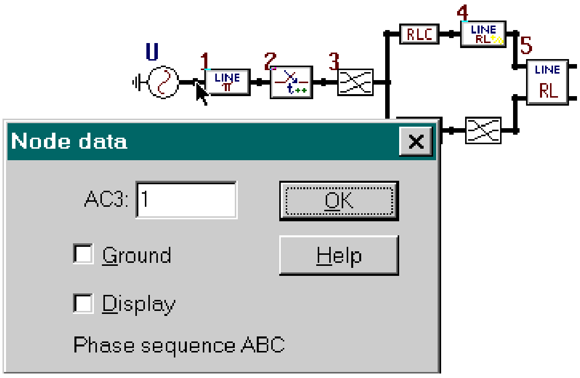


Fig. 3.32/b - Equivalent ATPDraw circuit (`EXA_2.CIR`).

The circuit shown in Fig. 3.32 was built up in the same way as your first circuit. You can note that connections between the three phase nodes appear to be thick. The circuit contains 2 special objects, an already mentioned transposition object (in this case from *ABC* to *BCA*) and a splitter object which splits three phase nodes into three single phase nodes.

Names 1 - 8 have been given to nodes in the circuit. By selecting *Make Names* under *ATP* in the main menu, ATPDraw examines the circuit and creates unique node names.

If you click with the right mouse button on nodes after selecting *ATP | Make Names* you are able to see the phase sequence in the bottom of the node input window as shown in Fig. 3.33/a - Fig. 3.34/b. As shown in Fig. 3.35, single phase nodes do not have a phase sequence, but the single phase side of a splitter object has one as shown in Fig. 3.34/b.



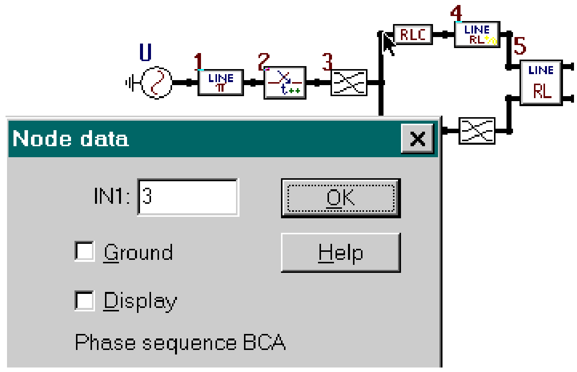
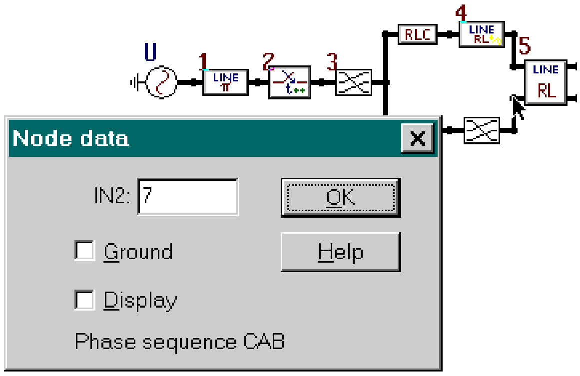


Fig. 3.33/a - Click right button on node '1'   Fig. 3.33/b - Click right button on node '3'.

The node names in Fig. 3.33/a are *1A*, *1B* and *1C*, all left adjusted. The node names in Fig. 3.33/b are *3B*, *3C* and *3A*, all left adjusted. ATPDraw gives the phase sequence *ABC* to the sub-network left of the first transposition object.



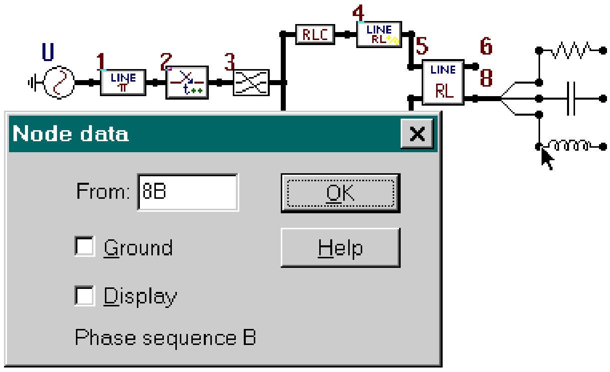


Fig. 3.34/a - Click right button on node '7'   Fig. 3.34/b - Click right on single phase node '8B'.

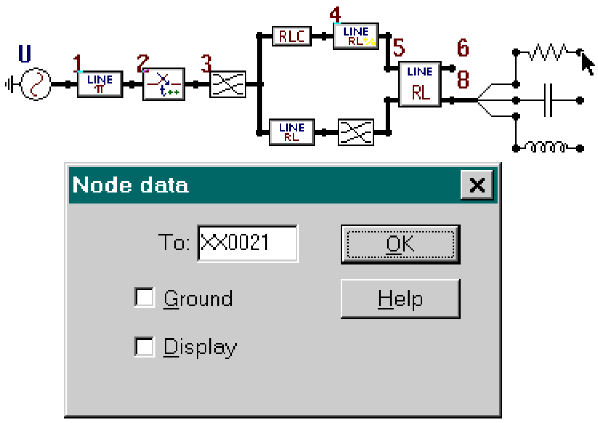


Fig. 3.35 - Click right button on single phase right node of the resistor.

No name is specified by the user for the node in Fig. 3.35. ATPDraw thus gives the node a name starting with XX followed by a unique number. This node is a single phase node with no phase sequence. 3-phase nodes with no user specified names are given a name starting with X, followed by a four digit number and ending with the phase sequence letters *A*, *B* and *C*.

Some special restrictions apply to the splitter object (found under *Probes & 3-phase* in the component selection menu):

- Connecting splitter objects together directly on the 3-phase side is illegal.
- It is not permitted to connect splitters together with connections on the single phase side.
- If the name *NODEA* is given to what you know is phase *A* on the single phase side, ATPDraw does not accept this and adds its own *A* at the end, creating the node name *NODEAA*. The general rule is that ATPDraw takes care of the phase sequence alone!
- The best solution is to specify a node name on the 3-phase side only.

The ATP data file of the circuit created by ATPDraw from Fig. 3.32/b is shown below:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  Mon 4.May, 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
C Miscellaneous Data Card ....
C  dT  >< Tmax >< Xopt >< Copt >
 .000001    .001
     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
1 1A    2A                   10. .0001   1.
2 1B    2B                   10. .0001   1.   10. .0001    1.
3 1C    2C                   10. .0001   1.   10. .0001    1.   10. .0001    1.
  3B    4B                    1.  .001                                         0
  3C    4C                    1.  .001                                         0
  3A    4A                    1.  .001                                         0
514B    5B                    2.        1.5
524C    5C                    1.         1.
534A    5A
513B    7B                   10.         1.
523C    7C                               1.   10.          1.
533A    7A                               1.                1.   10.          1.
515B    6B                   10.         1.
525C    6C                               1.   10.          1.
535A    6A                               1.                1.   10.          1.
547C    8C                               1.                1.                1.
                             10.       1.
557A    8A                               1.                1.                1.
                                         1.   10.          1.
567B    8B                               1.                1.                1.
                                         1.                1.   10.          1.
  8C    XX0021             1000.                                               0
  8A    XX0023                           1.                                    0
  8B    XX0025                     10.                                         0
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><    Ie   ><Vf/CLOP ><  type  >
  2A    3A            -1.      .001                                            0
  2B    3B            -1.      .001                                            0
  2C    3C            -1.      .001                                            0
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
141A     0  150000.       60.                                     -1.       1.
141B     0  150000.       60.     -120.                           -1.       1.
141C     0  150000.       60.      120.                           -1.       1.
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

4. Reference Manual . . .

. . . .

ATPDraw™
for Windows
1.0

This part of the manual outlines all menu items and program options, and gives an overview of the supported ATP objects, TACS components and MODELS features.

ATPDraw has a standard Windows user interface. The *Main window* of the program is shown in Fig. 3.1 . The *Main menu*, the *Circuit window* and the *Component selection menu* are the most important items of that window. Elements of the *Main menu* and supported ATP components in the *Component selection menu* will be referenced in this part of the manual.

4.1 Main window

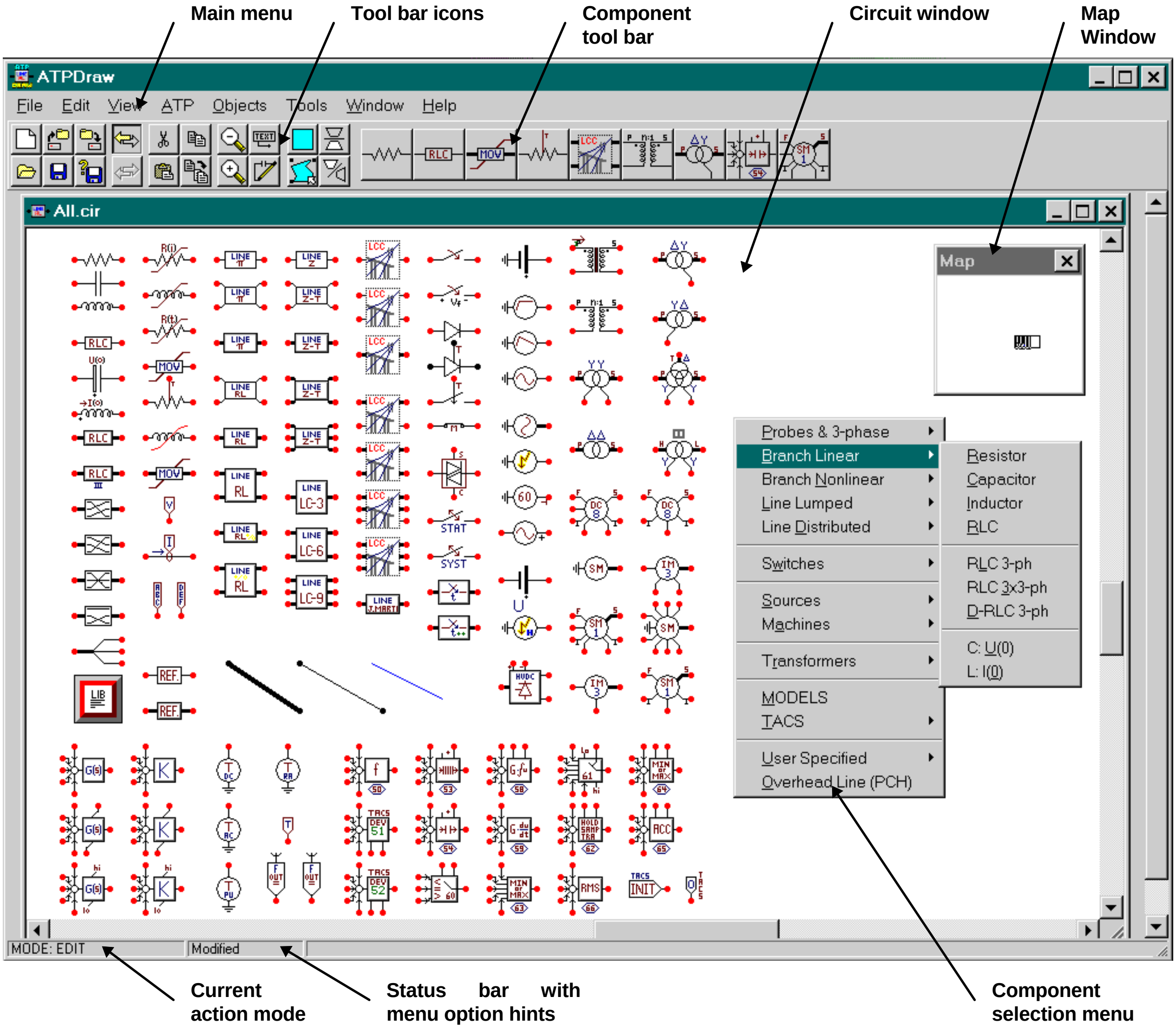


Fig. 4.1 - Main window of ATPDraw for Windows.

If you are unfamiliar with the use of ATPDraw, read the Introductory Manual or the Advanced Manual to learn how to create a circuit and new objects in ATPDraw. The Introductory Manual starts with the explanation of operating windows and the mouse in ATPDraw, and shows how to build up a circuit and how to create an ATP file to be used as input for a subsequent transient simulation.

4.2 Main menu

4.2.1 File

This field contains actions for input/output of ATPDraw circuits. Pressing the *File* item will result in a popup menu shown in Fig. 4.2.

Fig. 4.2 - File menu.

4.2.1.1 New

A selection of this field will open a new empty *Circuit window*. ATPDraw supports to work on several circuits simultaneously and copy information between the circuits. The number of simultaneous open windows are limited only by the available MS-Windows resources. The circuit window is much larger than the actual screen, as it is indicated by the scroll bars of each circuit window.

4.2.1.2 Open

This field performs a Windows' standard Open dialog box shown in Fig. 4.3 where the user can select a circuit file and load it into the ATPDraw. Short key: *Ctrl+O*.

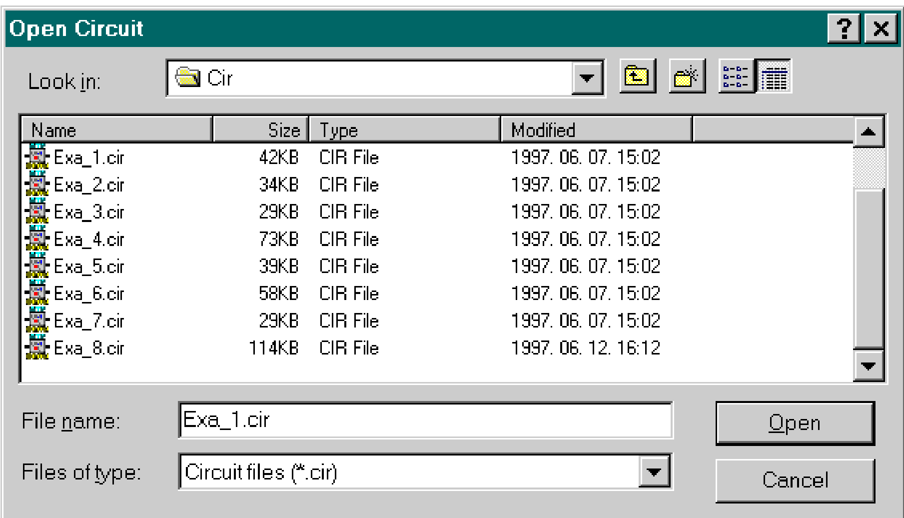


Fig. 4.3 - Open file window (Win95 style).

This Open/Save dialog box is used for several different selections in the main menu. An alternative MS-Windows 3.1 style is also supported as shown in Fig. 4.4. There is a check box in the *Tools* | *Options* menu to switch between supported alternatives.

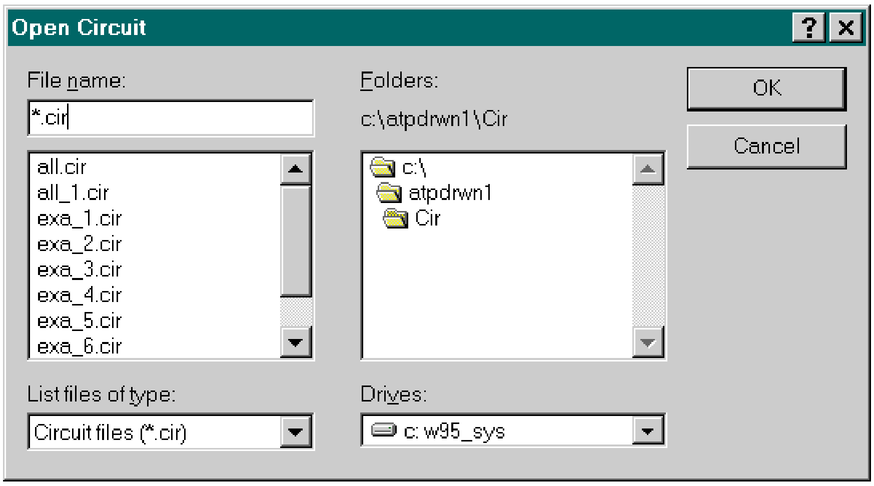


Fig. 4.4 - Open file window (Win 3.1 style).

The existing circuit files in the `\CIR` subfolder of ATPDraw are shown below the *File name:* field. A specific file can be selected either by typing the name directly, or by a left mouse click in the file list. Clicking *OK* will perform the selection made and the file is loaded into a new circuit window. Clicking on *Cancel* will simply close the window.

4.2.1.3 Save

Stores the circuit that is currently active to a disk file. If the name `Noname.cir` is shown in the circuit window a *Save As* dialog box will be performed, where the user must specify the current circuit file name. Short key: *Ctrl+S*.

4.2.1.4 Save As

Always performs the *Save As* dialog box (like the one shown in Fig. 4.3 or in Fig. 4.4) where the user must specify a file name for the current circuit. This command allows the user to save the circuit in the current window under a name other than that is already used.

4.2.1.5 Save All

Stores all open circuits in its own circuit files. If an open circuit still has not got a name (`Noname.cir`), it will be requested in *Save As* dialog boxes successively.

4.2.1.6 Close

The current circuit window will be closed. If any changes to the current circuit have not been saved yet, the user will be queried as shown in Fig. 4.5 to confirm before the circuit is closed.

4.2.1.7 Close All

Close all circuit windows. If a circuit has been modified since the last Save operation, a message will be displayed prompting the user for confirmation.

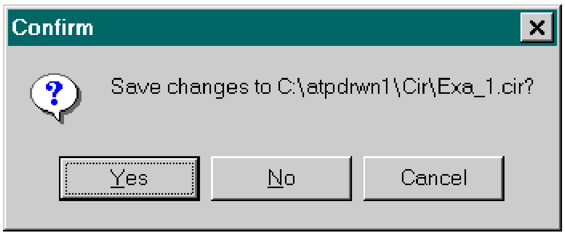


Fig. 4.5 - Confirmation is needed to close an unsaved circuit.

4.2.1.8 Import

Selecting this field results in the same Open dialog box as shown in Fig. 4.3. The user must specify a circuit file to load into the current circuit window. The imported circuit is pasted into the existing one and appears in the circuit window as a group in marked moveable mode.

4.2.1.9 Export

Same as *Save As*, but only the marked, moveable part of the circuit is written to a circuit file.

4.2.1.10 Save Metafile *(Windows 95/NT only)*

Writes the selected objects of the active circuit to a disk file in Windows Metafile (`.wmf`) format. If no objects are selected, the entire circuit window contents is written to disk. This way even large circuits can be saved graphically without loss of resolution seen on the screen when the *Zoom* option is used to fit the circuit to the screen size. Metafiles created by this command can be imported as picture into other Windows applications having filter available for this format.

4.2.1.11 Save Bitmap *(Windows 3.x only)*

Writes the selected objects of the active circuit into a Windows bitmap file (`.bmp`). If no objects are selected, the entire circuit window is written to disk. This 16-bit version of ATPDraw supports only the bitmap representation of the circuit pictures, which results in poorer resolution.

4.2.1.12 Save Postscript

Writes the objects in the active circuit window to a PostScript (`.ps`) or Encapsulated PostScript (`.eps`) file. A PostScript dialog box (Fig. 4.6) appears which enables you to specify a name and path for the PostScript file to generate, the PostScript file format (Standard:`.ps` or Encapsulated:`.eps`), and the output orientation (Portrait or Landscape). The PostScript file created by ATPDraw is independent of the current circuit window position and the screen resolution. It is possible to print out much larger circuits than the actual circuit window. The output file can later be sent to a PostScript printer or loaded into other applications (i.e. Ghostscript/Ghostview or word processors supporting the `.eps` picture format).

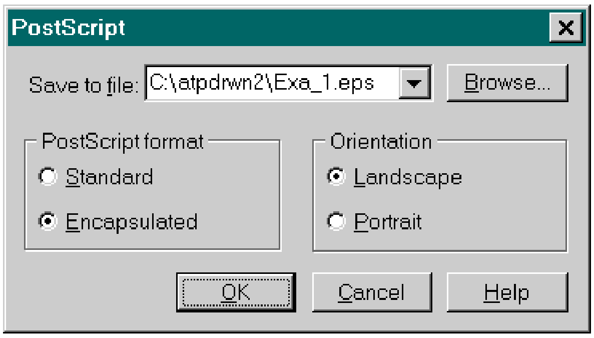


Fig. 4.6 - PostScript dialog box.

4.2.1.13 Print

Sends the contents of the active circuit window to the default printer. Short key: *Ctrl+P*. This command is executed as a standard Windows Print dialog box and allows the user to select the current printer, as well as to set up parameters for the printer.

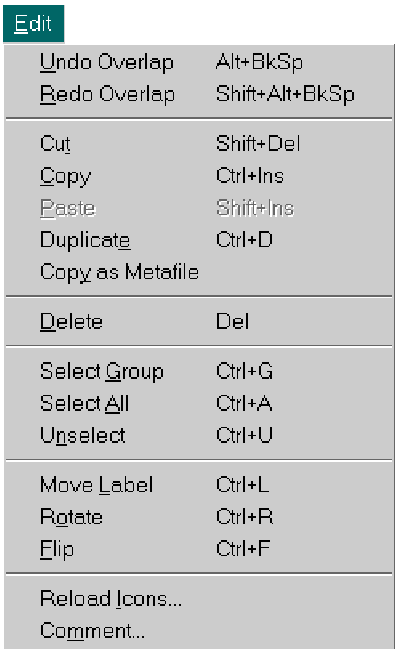
4.2.1.14 Print Setup

Enables to define default printer characteristics. This command will open a standard Windows Print Setup dialog box.

4.2.1.15 Exit

This command closes all open circuit windows of ATPDraw. User will be asked to save any modified circuits before the application is terminated.

4.2.2 Edit



This menu contains the various edit facilities of circuit objects in ATPDraw. The *Edit* popup menu is shown in Fig. 4.7.

Objects or group of objects have to be selected before any edit operations can be performed on them. If the user clicks on an object with the left mouse button in the circuit window, it will be rounded by a rectangular frame indicating the object is selected.

Fig. 4.7 - Edit menu.

4.2.2.1 Undo/Redo

Undo cancels the last edit operation, *Redo* cancels the last undo command. Short key for Undo/Redo: *Alt+Backspace* and *Shift+Alt+Backspace*.

The number of operations that can be undone/redone depends on the Undo/redo Buffers property setting on the *Preferences* page of the *Tools | Options* menu. Default value is 10. Almost all object manipulation functions (object create, delete, move, rotate, etc.) can be undone (or redone). Changes made to the circuit data in the component dialog box are also supported by the Undo/redo functions. These functions also update the circuit's *Modified* state in the status bar to indicate that the circuit has been modified. During an undo operation, the modified state is reset its previous value. If you undo the very first edit operation, the *Modified* text in the status bar will disappear. Any operations undone can be redone. Since only a limited number of buffers are allocated, you are never guaranteed to undo all modifications.

4.2.2.2 Cut

Copies the selected objects to the Windows clipboard and deletes them from the circuit window. The objects can later be pasted into the same or other circuit windows, or even other instances of ATPDraw. Short key: *Ctrl+X* or *Shift+Del*.

4.2.2.3 Copy

Copies the selected objects to the clipboard. Short key: *Ctrl+C* or *Ctrl+Ins*. A single marked object or a group of objects can be copied to the clipboard. This command unselects the selected objects.

4.2.2.4 Paste

The contents of the clipboard are pasted into the current circuit when this menu item is selected. Short key: *Ctrl+V* or *Shift+Ins*. The pasted objects appear in the current window in marked moveable mode.

4.2.2.5 Duplicate

Copies the selected object or a group of objects to the clipboard and then duplicates them in the current circuit window. Duplicated objects appear in the current window in marked moveable mode. Short key: *Ctrl+D*.

4.2.2.6 Copy as Metafile *(Windows 95/NT only)*

Copies the selected objects to the clipboard as a Windows Metafile object. This way even large circuits can be saved graphically without loss of resolution seen on the screen when the *Zoom* option is used. *Copy as Metafile* command supports the Windows' standard metafile format, so the product of this operation can be pasted into other applications which also support this file format specification.

4.2.2.7 Copy as Bitmap *(Windows 3.x only)*

Copies the selected objects to the clipboard as a Windows Bitmap object. The 16-bit version of ATPDraw supports only the bitmap representation of a circuit which results in poorer resolution when copying them to the clipboard.

4.2.2.8 Delete

Removes selected objects from the circuit window. Short key: *Del*.

4.2.2.9 Select Group

This option lets the user select a range of objects by specifying a polygon shaped region in the circuit window. A selected group of objects can be moved then and edited like a single object. Short key: *Ctrl+G*.

By selecting this menu item the mouse cursor turns into a pointing hand, the cursor is locked in and moved to the middle of the circuit window. The action mode also changes to *MODE:GROUP* in the status bar. A click with the left mouse button will create a corner in a polygon (click and release the button and a rubber band line is drawn between the starting point and the current mouse cursor). A click with the right mouse button will close the polygon. All components having their centre inside the polygon and all connections having both endpoints inside the polygon will be included in the group.

The selected group can be the subject of most editing operations: Move: (click left button, hold down and drag. Only the polygon is drawn while moving), Rotate/Copy/ Duplicate/Delete or Export in the File menu. To unselect a group, just click with the left mouse button outside the polygon in an empty area of the circuit window.

You can also enter this mode by double-clicking the left mouse button in an empty area of the circuit window. To close the polygon and unlock the mouse, press the right mouse button.

4.2.2.10 Select All

Selects all objects in the current circuit window. Short key: *Ctrl+A*.

4.2.2.11 Unselect

Cancels the selection of objects. Short key: *Ctrl+A*. Note that this operation may cause selected objects to be moved to the nearest grid point (also called gridsnapping). You can also unselect by clicking the left mouse button in an empty area of the circuit window.

4.2.2.12 Move Label

All circuit objects have a label specified in the object input window. These labels are moveable and written on the screen in blue (component label) or in red color (node name). By selecting the *Move Label* menu item, the mouse cursor turnes to a pointing hand and moves to the middle of the circuit window. The action mode indicator in the status bar becomes *MODE: MOVE LABEL*. The user can now click on a label, hold the left mouse button down and drag the label to a desired position. The operation ends when you finish moving the label and release the mouse button. Short key: *Ctrl+L*.

In many cases the component labels and node names are not overlapping the component icons. In such cases they can be moved by a simple left click on the label, then by holding the left mouse button down and dragging. So the *Move Label* command is most frequently used when labels are located close to or behind the component icons and can not be selected otherwise.

4.2.2.13 Rotate

A marked object or a group of objects are rotated 90 degrees counter clock-wise when this field is selected. This operation can also be performed by clicking the right mouse button inside the selected region. Short key: *Ctrl + R.*

4.2.2.14 Flip

Flips the selected objects by rotating them 180 degrees. Short key: *Ctrl + F*.

4.2.2.15 Reload icons

The command reads and displays component icons from their respective support files. This function is useful when the support file icons have been redesigned and the user wants the changes to be reflected in the circuit window.

4.2.2.16 Comment

Opens a comment dialog box, where three lines of text can be written as comments to the active circuit, as shown in Fig. 4.8. This command also enables you to change the circuit comment if it already exists.

This three comment lines are written to the header section of the ATP file created by ATPDraw if the *Comments* option is selected in the *ATP | Settings* menu. To display the circuit comment at the bottom of the circuit window, select the *Comment Line* option in the *View menu*.

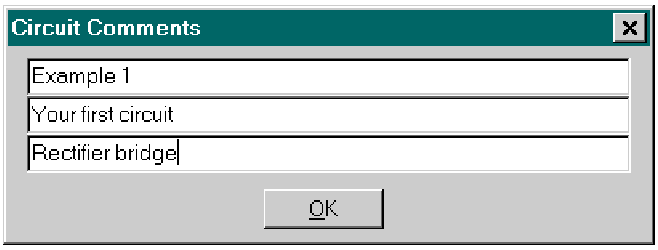
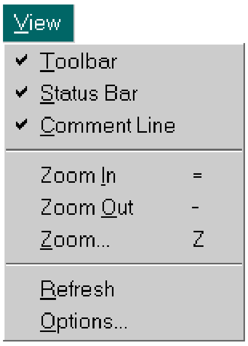


Fig. 4.8 - Circuit comment dialog box.

4.2.3 View



In the *View menu* you can control the visibility of several main and circuit window components and you can select how the circuit is drawn in the current window. The popup menu appearing after a click in the *View* main menu field is shown in Fig. 4.9.

Fig. 4.9 - View menu.

4.2.3.1 Toolbar

Shows or hides the toolbar at the top of the main window. The toolbar contains speed buttons for the most frequently used menu options. The following list describes available speed buttons.

 Opens an empty circuit window.

 Loads a circuit file into a new window.

 Saves the objects in the active circuit window to disk.

 Saves the objects in the active circuit window to a specified disk file.

 Inserts a circuit from file into the active circuit window.

 Saves the selected objects of the active circuit to a disk file.

 Cancels the last edit operation.

 Cancels the last undo command.

 Copies the selected objects to the Clipboard and deletes them from the circuit window. The objects can later be pasted into the same or other circuit windows.

 Copies the selected objects to the Clipboard.

 Inserts the objects in the Clipboard into the circuit window.

 Copies the selected objects to the Clipboard and then inserts them into the circuit.

 Enables you to select and move a component or node text label. The mouse cursor turns to a pointing hand.

 Redraws all objects in the active circuit window.

 Selects all objects in the active circuit window.

 Enables you to select a group of objects by specifying a polygon shaped region in the active circuit window. The mouse cursor style will turn to a pointing hand to indicate this mode of operation. To close the region and unlock the mouse, press the right mouse button.

 Enlarges the objects by increasing the current zoom factor by 20 percent.

 Diminishes the objects by reducing the current zoom factor by 20 percent.

 Rotates the selected objects 90 degrees counter-clockwise. This operation can also be performed by clicking the right mouse button inside the selected region.

 Flips the selected objects by rotating 180 degrees.

To the right of the speed buttons the nine most recently used component icons are displayed. Selecting one of these shortcut icons inserts a new component into the active circuit window. The leftmost icon represents always the last inserted component.

4.2.3.2 Status bar

Shows or hides the status bar at the bottom of the main window. The status bar at the bottom of the main window displays status information about the active circuit window. The mode field on the left hand side shows which mode of operation is active at present. Possible modes are:

EDIT	Normal mode. Indicates no special type of operation.
CONN.END	Indicates the end of a connection. The program is waiting for a left mouse button click to set the end-point of a new connection. To cancel drawing a connection, click the right mouse button or press the *Esc* key.
MOVE LABEL	Indicates a text label move. Clicking the left mouse button on a text label, then holding down and dragging enables you to move the label to a position of your choice. To cancel moving a label, click the right mouse button or press the *Esc* key.
GROUP	Indicates region selection. Double clicking the left mouse button in an empty area of the active circuit window enables you to draw a polygon shaped region. To end the selection click the right mouse button. Any object within the region becomes a member of the selected group. To cancel region selection, press the *Esc* key.
INFO.START	Indicates the start of a relation when *TACS | Draw relation* is activated in the component selection menu. Clicking the left mouse button on a component node or on the end-point of another relation will initiate the drawing of a new relation. Relations are used to visualize information flow into Fortran statements and are drawn as blue connections, but do not influence the connectivity of components.
INFO.END	Indicates the end of a relation. The program is waiting for a left mouse button click to set the end-point of the new relation. To cancel drawing a relation, click the right mouse button or press the *Esc* key.

The field to the right of the mode field displays the modified status of the active circuit. As soon as you alter the circuit (moving a label, deleting a connection, inserting a new component, etc.), the text *Modified* will appear indicate that the circuit needs saving before quitting ATPDraw. The field will be empty when you save the circuit or undo all modifications. Note that the number of available undo buffers are limited (default value is 10, but can be increased from the *Preferences* page of the *Tools* | *Options* menu to maximum 100). In the default case, if more than 10 modifications are done, the field will indicate a modified status until you save the circuit.

The rightmost field of the status bar displays the menu option hints.

4.2.3.3 Comment line

Shows or hides the comment line at the bottom of the active circuit window.

4.2.3.4 Zoom In

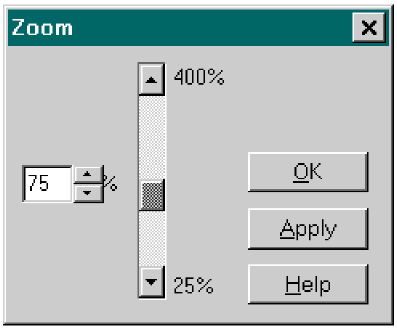
Enlarges the objects in the active circuit window by increasing the current zoom factor by 20 percent. Short key: + *(plus sign on the numeric keypad or +/= key).*

4.2.3.5 Zoom Out

Diminishes the objects in the active circuit window by reducing the current zoom factor by 20 percent. Short key: - *(minus sign on the numeric or the standard - key).*

4.2.3.6 Zoom

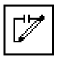
Selecting this field brings up the *Zoom* dialog box shown in Fig. 4.10. Short key: *Z.* In the Zoom dialog you can specify the zoom factor of the active circuit window. The actual zoom factor is given in the input field at left in percent. The normal view has a zoom factor of 100 percent. To zoom in on circuit objects, increase the zoom factor. To zoom out and view a larger portion of the circuit, decrease the factor. Upper and lower limits are 400 and 25 percent, respectively.



To accept current zoom factor setting and return from the Zoom dialog, select the *OK* button. To set a new zoom factor and view the result without returning, select the *Apply* button.

Fig. 4.10 - The Zoom dialog box.

4.2.3.7 Refresh

Redraws all objects in the active circuit window. No short key is assigned, but this command can also be activated via the Toolbar icon: 

4.2.3.8 Options

Selecting this field brings up the *View Options* dialog box shown in Fig. 4.11, where the user can specify object visibility options.

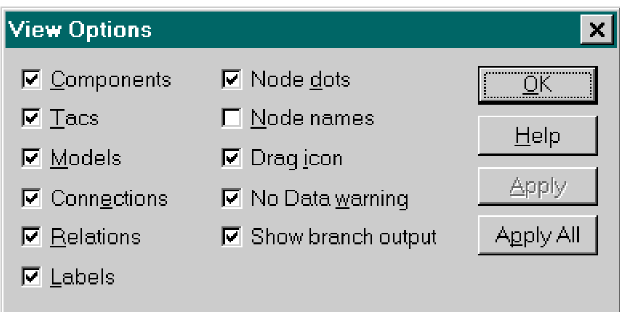


Fig. 4.11 - View Options dialog box.

The View Options dialog can be used to control the visibility of the objects in the active circuit window. By default, all objects except node names are visible. The following description explains each option in normal checked (☑) state:

Components	All standard and user specified components are displayed.
Tacs	All TACS components are displayed.
Models	All MODELS components are displayed.
Connections	All connections (short circuits) are displayed.
Relations	All relations (information arrows) are displayed.
Labels	Component labels are written on the screen.
Node dots	Node and connection end-points are displayed as filled circles.
Node names	Node names are written on screen (overrides the Display attribute of nodes). This option is useful after a *Make Names* selection in the *ATP* menu.
Drag icon	The complete icon is drawn during single component moves (in the unchecked state, only an inverted rectangular outline is drawn).
No Data warning	Components and node dots are drawn with a red color until the component or node has not received data. No such checking is done when this option is unselected.
Show branch output	Branch output request is included in the *Component* dialog box of standard components such as linear branches, non-linear elements, switches and transformers. When you select this option, small symbols appear on the top-left side of the objects' icon indicating the branch output request.

To accept the current view options and return from the View Options dialog, select the *OK* button. To set and view the results of the new options without returning, selected the *Apply* button. If you want the current settings to be applied to all current and future circuit windows, select the *Apply All* button before you exit the dialog box.

4.2.4 ATP

The ATP menu allows you to generate names for unnamed nodes, to generate or edit the ATP file, and specify ATP settings for the circuit. Version 1.2 of the program also supports editing your batch jobs here. The jobs you specify, are listed immediately below the *Edit batch jobs* command. The menu appearing after clicking on the *ATP* menu field is shown in Fig. 4.12.

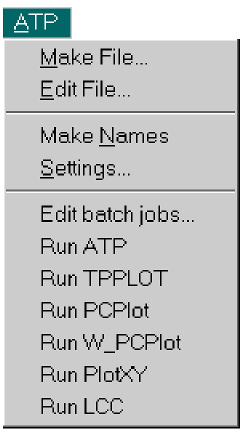


Fig. 4.12 - The ATP menu.

4.2.4.1 Make File

Generates an ATP input file for the active circuit window. The user will be asked to confirm the name of the file. Default filename is the name of the current circuit file, with `.atp` extension. ATPDraw first calls the make name procedure (even if the *Make Name* has been previously selected), then generates an ATP input file on the form specified in the *Settings* menu.

4.2.4.2 Edit File

This selection calls a built-in text editor which enables you to contemplate or edit the ATP file. When *Edit File* option is selected a file having the same name as the active circuit file with extension `.atp` is searched for, and is automatically opened as shown in Fig. 4.13.

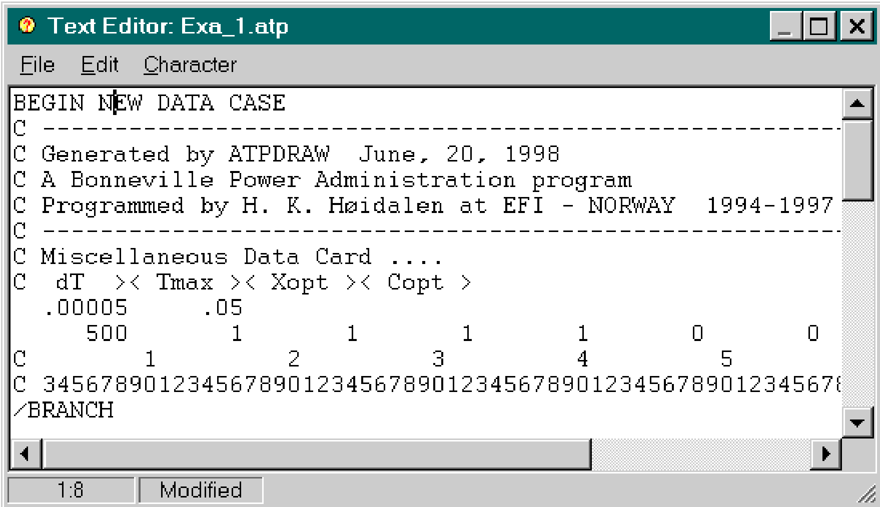


Fig. 4.13 - The main window of the built in text editor.

The status bar at the bottom of the window displays the current line and column position of the text cursor and the buffer modified status. Basic text editing facilities (Open/Save, Print, Copy/Paste, Find&Replace) are supported, the default text font can be changed by selecting the *Font* option in the *Character menu*. The text buffer of this editor is limited to maximum 32kB in size, however the user can specify his own favorite text editor (wordpad.exe, write.exe, notepad.exe) on the *Preferences* page of the *Tools* | *Options* dialog box.

Text Editor option in the *Tools* menu provides an alternative way of invoking this editor. In that case the text buffer will initially be empty.

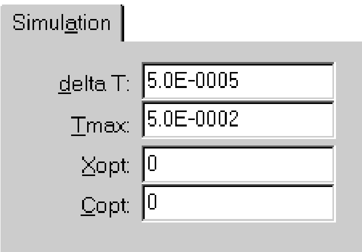
4.2.4.3 Make Names

When this field is selected ATPDraw examines the current circuit and gives unique names to all nodes in the active circuit window. Connected or overlapped nodes are given the same name. User will be asked to confirm this operation. While ATPDraw establishes the node names a **`Generating node names`** message is displayed in the middle of the current circuit window. After the *Make Names* field has been selected, the node name and phase sequence attributes in the *Component* dialog box and in the *Node data* window will be updated.

4.2.4.4 Settings

In the *ATP Settings* dialog box several options for the active circuit window can be specified. Settings on this dialog are used when ATPDraw generates the ATP input file. The dialog box has four pages, such as the miscellaneous data card settings (*Simulation*, *Integer*, *Switch)* and the *File format* settings.

Simulation settings



delta T	Time step of the simulation in seconds.
Tmax	End time of the simulation in seconds.
Xopt	Inductances in [mH] if zero; otherwise, inductances in [Ohm] with Xopt as frequency.
Copt	Capacitances in [mF] if zero; otherwise, capacitances in [Ohm] with Copt as frequency.

Fig. 4.14 - Simulation settings.

Integer settings

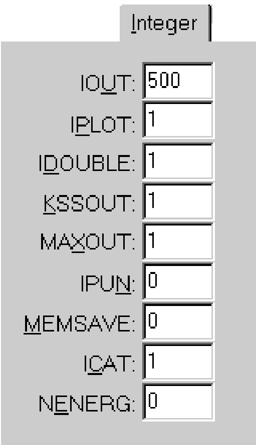


Fig. 4.15 - Integer settings.

IOUT	Frequency of LUNIT6 output within the time-step loop. For example, a value of 3 means that every third time step will be printed.
IPLOT	Frequency of saving solution points to the PL4 output file. For example, a value of 2 means that every second time step will be written to the PL4 file.
IDOUBLE	If 1, table of connectivity written in the LUNIT6 output file. If 0 (zero), no such table written.
KSSOUT	Controls steady state printout to the LUNIT6 output file. Possible values are: 0: No printout. 1: Print complete steady state solution. Branch flows, switch flows and source injection. 2: Print switch flows and source injection. 3: Print switch flows, source injection and branch flows requested in column 80 punches.
MAXOUT	If 1, extrema printed at the end of the LUNIT6 output file. If 0 (zero), no such printout.
IPUN	Flag for requesting an additional card for controlling the IOUT frequency. If IPUN equals -1 an additional card follows (not implemented). If IPUN equals 0 (zero), no such card follows.
MEMSAVE	Controls the dumping of EMTP memory to disk at the end of simulation if START AGAIN request is specified. A value of 1 indicates memory saving (START AGAIN). Zero implies no memory dumping.
ICAT	Controls saving of raw plot data points that is written to the I/O channel LUNIT4. 0: No saving. 1: Save points, but ignore any batch-mode plot cards present. 2: Save points and handle batch-mode plot cards.
NENERG	Number of simulations. A value of 0 (zero) means single, deterministic simulation; otherwise, statistic switch study (NENERG > 0) or systematic switch study (NENERG < 0).

Switch settings

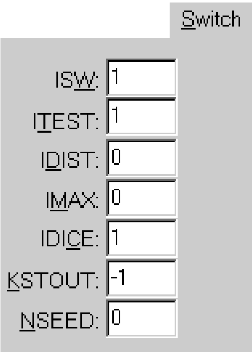


Fig. 4.16 - Switch settings.

ISW	If 1, printout of all variable switch closing/opening time to LUNIT6. If 0 (zero), no printout.
ITEST	Extra random delay using DEGMIN, DEGMAX and STATFR in STARTUP. Possible values are: 0: Extra random delay for all switches. 1: No random delay. 2: Extra random time delay added to all closing switches. 3: Extra random time delay added to all opening switches.
IDIST	Select probability distribution of switch. 0(zero) means Gaussian distribution and a value of 1 uniform distribution.
IMAX	If 1, printout of extrema to LUNIT6 for every energization. If 0 (zero), no such printout.
IDICE	Use of standard random generator. A value of 0 (zero) implies computer-dependent random generator and a value of 1 standard random generator.
KSTOUT	Extra printed (LUNIT6) output for each energization. Output of time-step loop and variable extrema (MAXOUT>0). If 0 (zero), extra printed output. If -1, no such output.
NSEED	Repeatable MonteCarlo simulations. Possible values are: 0: Every simulation on the same data case will be different. 1: Same result each time the data case is run on the same computer.

The ATP file can be arranged according to four different types of file formats. The File format settings page contains four buttons for setting the ATP input file data format and one button for controlling auto path generation. The structure of the `.atp` file if options are selected (☑), will be defined as follows:

File format

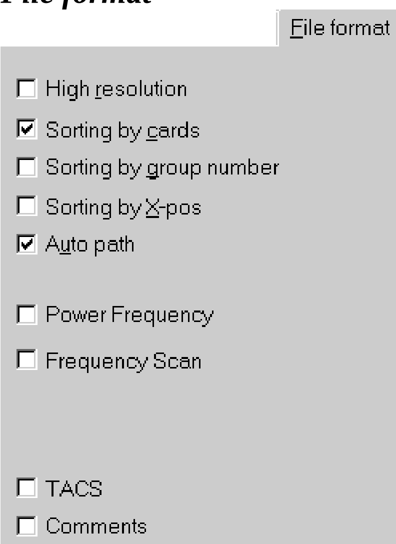


Fig. 4.17 - File format settings.

High resolution	Usage of $Vintage, 1 (if possible), otherwise no $Vintage cards written to the file.
Sorting by cards	Data file written with BRANCH cards first, followed by SWITCH cards and the SOURCE cards.
Sorting by group	The group number given to each object determines the sequence of cards. The lowest group number comes first.
Sorting by X-pos	The leftmost object in the circuit window is written first.

Any combination of the three different sorting mechanisms above can be specified.

Auto path	Library files are supposed to have extension *.lib and* located in the USP folder. Writes $Prefix and $Suffix to the ATP file.

Each library file specification is verified to meet these requirements. If the path of a library file specifies a different folder or the extension is not `.LIB`, an error dialog is displayed during ATP file generation, enabling you to correct the erroneous specification by stripping off path and extension, continue the operation using an unresolvable ATP include reference, or cancel the entire ATP file generating process.

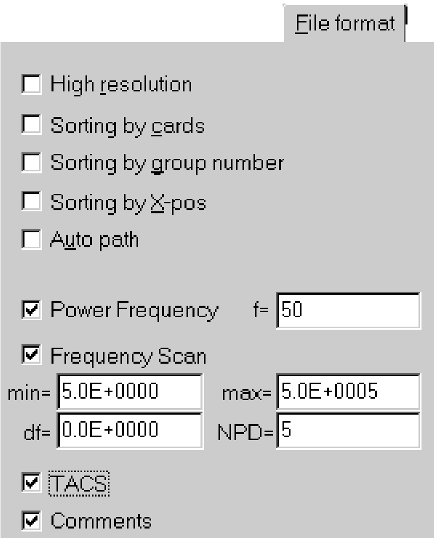
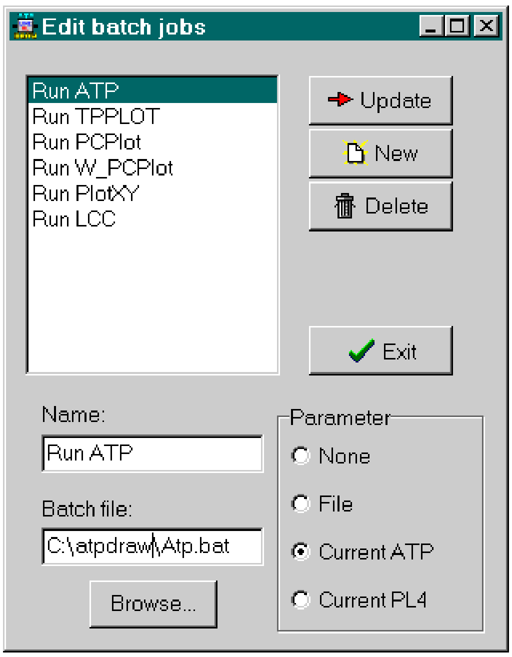


Fig. 4.18 - File format settings.

```
Power frequency   The file is written with
                  POWER FREQUENCY request word.
                  The value of the system
                  frequency can be set when the
                  option is selected.
Frequency scan    Data file written with request
                  word for frequency domain
                  simulation.
            min = starting frequency of the
                  simulation
            max = final frequency point
             df = frequency step (if NPD=0
                  linear scale is used)
            NPD = request for logarithmic
                  scaling with the given number
                  of points per decade.
TACS              Will force the TACS HYBRID
                  and BLANK TACS cards to be
                  written to the ATP file.
Comments          Circuit comments are written
                   to the ATP file
```

4.2.4.5 Edit batch jobs *(in version 1.2 and above)*

This feature makes possible to execute your own external commands directly from ATPDraw. The new menu items specified here are inserted into the *ATP* menu, so it will change dynamically. After clicking on the *New* button of the dialog box shown in Fig. 4.19, you are requested to specify:



- the *Name* of the job,
- the name and path of a *Batch file* or executable file,
- and the name of the file which is sent as *Parameter*.

The last option can be selected by radio buttons in the *Parameter* field. If the *File* button is *on* state, ATPDraw performs an open dialog box, where the user can select a file name, to be sent as parameter when executing the batch file. Probably the most frequently used selection here is the *Current ATP* or *Current PL4* to pass the actual file name to the batch job.

When you completed editing the batch job settings, click on the *Update* button and the new commands are inserted into the *ATP* menu.

Fig. 4.19 - Specifying your own batch jobs

As any other program options, the previous settings can be saved to the `ATPDraw.ini` file by using the *Tools | Save Options* command or by selecting the "Save options on exit" check box on the *General* page of the *Tools | Options* menu.

This feature can be used for many different purposes in ATP simulation: e.g. running ATP within ATPDraw; processing simulation results by TPPLOT, PCPlot or PlotXY; launching supporting programs, like LCC for Line/Cable constant support, or any other data assembler. It must be noted however, that it is always the user's responsibility to provide the internal commands of that batch files in the correct format. Inexperienced users are requested to read chapter 2.9 of the Installation Manual, which describes how to create these batch files for different applications.

4.2.5 Objects

In this menu the user can customize the component support files and create new one. Circuit objects in ATPDraw can be divided in 4 categories :

1. Standard components
2. User specified components ($Include)
3. MODELS components
4. TACS components

Each object has a support file containing all information about data and nodes, the graphical representation of the object (icon) and the associated help file. Each circuit object has a name internally in ATPDraw, equal to the name of the support file. The support files have extension `.SUP` and are stored in a directory dependent on the object type. The full path to the support files are included in the data structure of the circuit files, so the user can store the `.SUP` files anywhere he likes, but the usage of the directory structure given below is recommended to ensure compatibility with other users:

Object type	Support file (`.SUP`)	MODELS file (`.MOD`)	Include file (`.LIB`)
Standard Components	`\SUP`	-	-
User specified components	`\USP`	-	`\USP`
MODELS components	`\MOD`	`\MOD`	-
TACS components	`\TAC`	-	-

The objects support files can be edited in the *Objects | Edit Component* menu. The user can create new Models and User Specified components as explained in the Advanced Manual.

4.2.5.1 New component

The *New component* menu item is reserved for program developers. To add a new component, the source code needs to be modified, so this feature is not available for the users.

4.2.5.2 Edit Component

The standard component support files in the `\SUP` folder can be customized here. Selecting the *Edit Component* field will first perform an open file dialog box shown in Fig. 4.20, where the user can specify which support file to edit. After the user has specified a file name (normally by marking a file in the window and clicking *Open*), a dialog box shown in Fig. 4.21 appears.

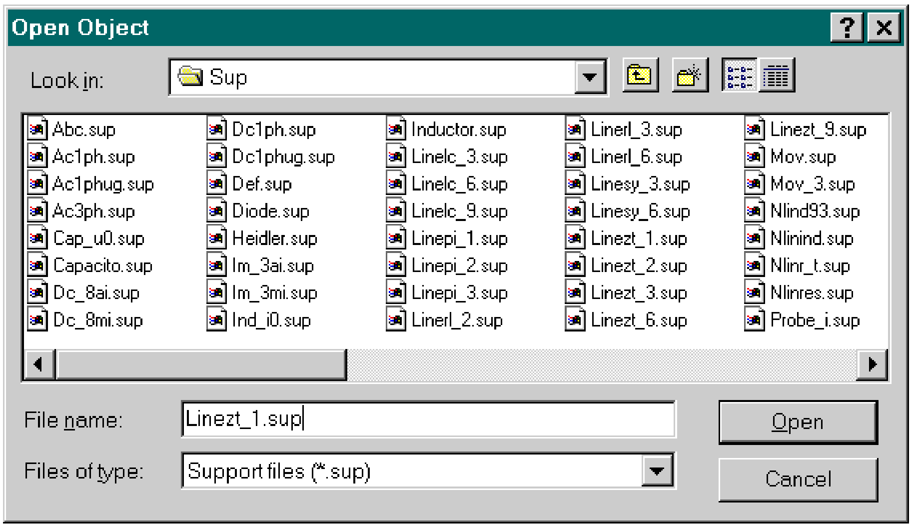


Fig. 4.20 - Specify the support file to edit

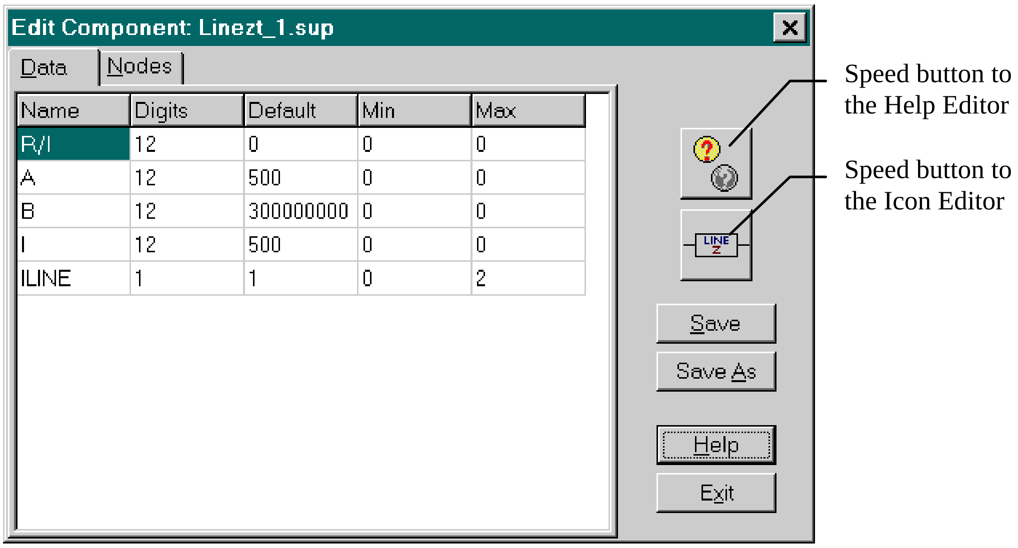


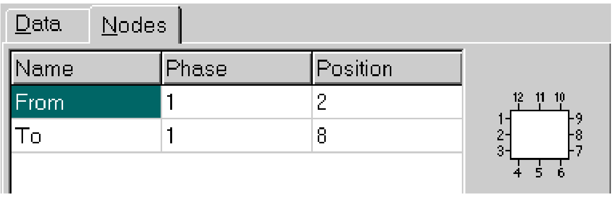
Fig. 4.21 - Control page of object data

On the *Data* page of the *Edit Component* dialog box, five control variables of each data parameters of the support file (one row for each object data parameter) can be specified. The following table describes available parameter options:

Name	The name of the parameter. Used to identify the parameter in the *Component* dialog box. This name often reflects the name used in the ATP Rule Book.
Digits	Maximum number of digits allowed in the Component dialog box. This field is more important for user specified components in $INCLUDE statement.
Default	Initial value of the parameter.
Min/Max	Minimum/Maximum value allowed.

An error message will appear in the Component dialog box if a parameter value is out of range. To cancel range checking, set Min=Max (e.g. set both equal to zero).

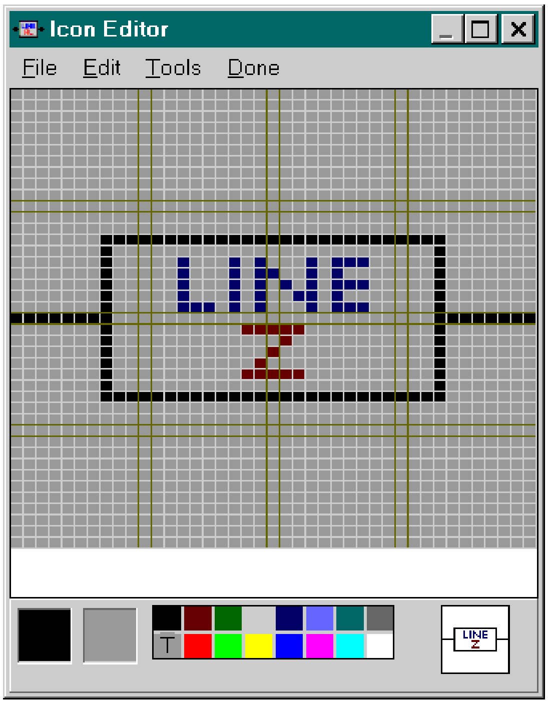
Clicking the *Node* tab on the *Edit Component* dialog box, the node attributes of the support file (one row for each component node) can be specified. Available node options are:



Name	The name of the node. Used to identify the node in the *Open Node* and *Component* dialog boxes.
Phase	Number of phases (1 or 3)
Position	Node position on the icon border.

Fig. 4.22 - Node control page

If Phase is set to 3 the node will be a 3-phase node with a 5 character name. ATPDraw adds the extensions *A*, *B* and *C* to the node name automatically. The figure on the right hand side of the page shows the possible node positions. The position should correspond to the icon drawing.



Each circuit object has an icon which represents the object on the screen. A speed button on the right hand side of the *Edit Component* dialog box invokes the built in pixel editor where icons can be edited. An icon occupies 41x41 pixels on the screen.

Clicking with the left mouse button will draw the current color selected from a 16 color palette at the bottom. Clicking the right button will draw with the gray background color. Olive colored lines indicate the possible node positions on the icon border.

Menu field items of the *Icon Editor* are described in the *Tools* | *Icon Editor* section of this manual.

Fig. 4.23 - Icon Editor

Each standard component has a pre-defined help file, which can be edited by a built in *Help Editor* accessible via speed button of the *Edit Component* dialog. In the help editor the user can write his optional help file for the objects.

Available functions and menu field items of the *Help Editor* are described in the 4.2.6.2 section of this manual.

When the user completed all modifications on the component data and on the icon and help files, the new support file can be saved to disk using *Save* (existing support file will be overwritten) or *Save As* (new file will be created) buttons. ATPDraw performs data input checking before saving the modified file to disk. In case of mistake an Error dialog box, such as on Fig. 4.24 appears.

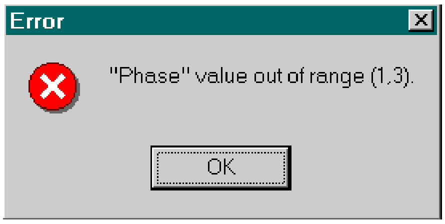
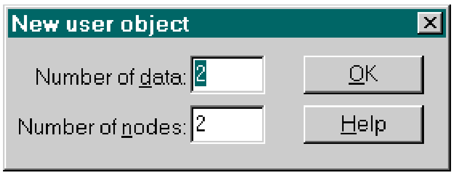


Fig. 4.24 - Error dialog

4.2.5.3 New User Specified

The user can also create new user specified objects which can be used in ATP thanks to $Include and Data Base Modularization. This menu item enables you to customize data and node values, icon and help text of a user specified component. Selecting this field will first display the object size window shown in Fig. 4.25.



The Number of data is specified in the range 0 to 32, and the Number of nodes in the range 0 to 12.

Fig. 4.25 - Specify size of object

When the user determined the size of the new object based on the Data Base Module (DBM) file header and clicks *OK*, a notebook-style dialog box as the one shown in Fig. 4.21 appears. On the *Nodes* and *Data* pages, the control parameters for each object data can be specified. The names of the data and the nodes do not have to be equal to the ones used in the DBM punch file. If the number of phases is specified to 3, the user has two choices:

- to let ATPDraw write one five-character node name in the $Include statement and let the extensions *A*, *B* and *C* be added inside the DBM punch file, or
- to let ATPDraw write three six-character node names in the $Include statement.

Checking the *Old 3-phase* button in the *Component dialog* box enables the first choice (not recommended)

Each user specified components can also have an icon which represents the object on the screen and an optional on-line help which describes the meaning of parameters. These properties can be edited using the built in *Help* and *Icon Editor* exactly the same way as it was described in the previous session.

Finally, after clicking the *Save* or *Save As* buttons the new support file will be saved to disk. Support files for user specified components are normally located in the `\USP` folder. The file name can be specified in a standard file save dialog box. The object must have the extension `.SUP`.

4.2.5.4 Edit User Specified

The user specified objects can be edited in the same way as the standard components. The menus are identical as shown in Fig. 4.20 - Fig. 4.24. The Digits field in the data control page (Fig. 4.21) specifies the number of characters used for the data parameter in the $Include statement.

4.2.5.5 New Model

The user can also create new MODELS component as explained in the Advanced Manual. Selecting the *New Model* field will first perform the object size dialog shown in Fig. 4.25. The number of nodes is now the sum of inputs and outputs of Model. The maximum number of nodes is 12 and the maximum number of data passed into a Model is 32. Parameters on the Data control page are identical as shown in Fig. 4.21. Settings on the Node control page of the New Model dialog box are given on the next Figure.

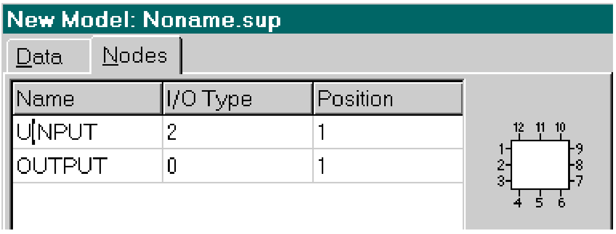


Fig. 4.26 -Node control page for a Model

```
Name        The name of the node. Used
            to identify the node in
            the Open Node and
            Component dialog boxes.
I/O Type    Type of input/output to
            MODELS component node:
            0: Output node.
            1: Current input node.
            2: Voltage input node.
            3: Switch status input
            4: Machine variable input
Position    Node position on the icon
            border.
```

The procedure is further the same as for new user specified objects. Support files for model components are normally located in the \MOD folder. In addition to a support file and an icon definition, each Model component also needs a text file created outside ATPDraw or using the built in *Text Editor* in the *Tools* menu. This text file contains the actual Model description. By default you can store these files also in the \MOD folder.

The names on the Models *Data* and *Nodes* pages must be equal to the names used in the actual Model file. The I/O Type parameter can be changed later in the Model node input window (right click on the object's node). All model nodes are assumed to be single phase.

4.2.5.6 Edit Model

A model object can be edited like any other circuit object. If the user chooses *Edit Model*, the well known *Open Object dialog* appears where a model support file can be selected. Then a data control window enables the user to customize data and node values, icon and help text of a model component.

4.2.5.7 Edit TACS

A TACS object can be edited like a Model object. *Edit TACS* menu item enables the user to customize a standard TACS component in the \TAC folder.

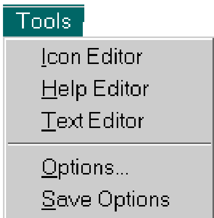
A TACS object will use exactly the same control pages as a Model object. Settings on the Node and Data control pages are similar to that of the Models objects, except I/O Type field which is a code for the type of TACS node:

```
I/O Type    Type of input/output to TACS component node
            0:    Output node
            1:    Positive input
            2:    Negative input
            3:    Disconnected input. A node will also be taken as
```

disconnected if it has no visual connections in the circuit drawing and has no user specified node name.

This I/O Type can be changed in TACS node input windows (right click on a TACS node). All TACS nodes are assumed to be single phase.

4.2.6 Tools

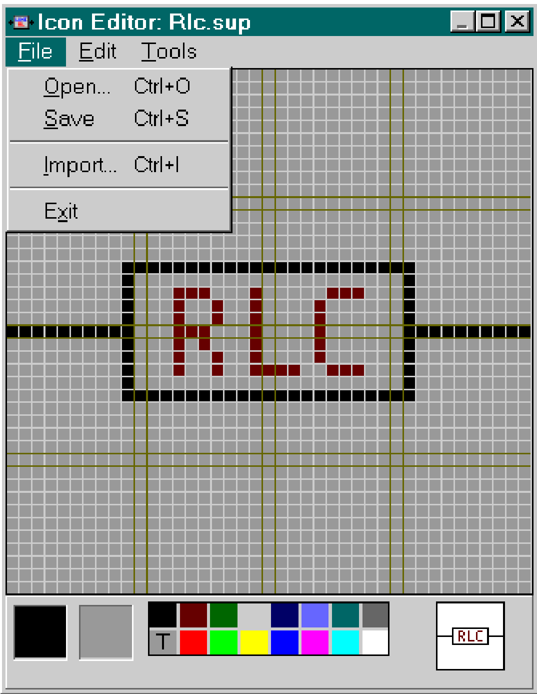


Items in the *Tools* menu enable you to edit component icons or help, view or edit text files, and save or customize several program options. Fig. 4.27 shows the available commands in the *Tools* menu.

Fig. 4.27 - Tools menu

4.2.6.1 Icon Editor

Brings up an icon editor shown in Fig. 4.28 where the user can edit the components' icon. It can be invoked either from the *Edit Component* dialog box or by selecting the *Icon Editor* option in the *Tools* menu. In either case you can create or edit the component icon of a support file.



Depending on how the editor was invoked, the file menu provides different options. When opened from the *Objects* menu (*Edit Component / Edit User-Specified / Edit Model / Edit TACS)*, you can only import icons from other support files or cancel the edit operation and close the editor window. In this case the *Done* option in the main menu is used to accept and store the modified icon as shown on Fig. 4.23.

When opened from the *Tools* menu, additional options to load from and save to disk appear in the *File* menu.

Fig. 4.28 - Icon Editor menus

At the bottom of the editor window there is a color palette with two boxes indicating the current color selection, and a real-size image of the icon being edited. In the color palette, the color marked with a capital letter *T* is the transparent color. To select a color from the palette, click either the left or the right mouse button in one of the color boxes. The selected color will be assigned to the mouse button you clicked until you use the same mouse button to select another color. Two boxes on the left hand side of the palette indicate the current color selection. The leftmost box displays the color currently assigned to the left mouse button. The one to the right displays the color assigned to the right mouse button. It is common to think of these two boxes as indicators of current foreground (leftmost) and background (rightmost) color selection.

The foreground color is normally used to draw with, and the background color to erase any mistakes made during the drawing. It is therefore convenient to assign the transparent color (indicated by *T*) to the right mouse button, and desired drawing color to the left button. Mistakes can then easily be corrected by alternating left/right mouse button clicks.
Drawing in the icon buffer can be accomplished by clicking the left or right mouse button. The olive colored vertical and horizontal lines indicate the icon node positions. These are the same positions as indicated on the *Node* page of the *Edit Component* dialog box.

The icon editor has a *File* menu, an *Edit* menu and a *Tools* menu. In addition, a *Done* option appears to the right of the *Tools* menu if the editor is opened from the *Edit Component* dialog box. Selecting *Done*, changes made to the icon will be accepted. Available menu options are described below:

```
File options
     Open          Loads the icon of a support file into the icon buffer.
     Save          Stores the contents of the icon buffer to disk.
     Import        Reads the icon of a support file and inserts it into the
                   icon buffer.
     Exit/Cancel   Closes the icon editor window. If the option displays Exit
                   and the icon buffer has been modified, you are given a
                   chance to save the icon before closing. If a Done option is
                   available from the main menu, this option displays Cancel,
                   and the window is closed without any warning with respect to
                   loss of modified data.
Edit options
     Undo          Cancels the last edit operation.
     Redo          Cancels the undo command.
     Cut           Copies a bitmap version of the icon to the Clipboard and
                   clears the icon buffer. This bitmap can be pasted into other
                   applications (e.g. pbrush.exe).
     Copy          Places a bitmap version of the icon in the Clipboard.
     Paste         Inserts the bitmap in the Clipboard into the icon buffer. If
                   colors are different from those used in the original bitmap,
                   it is because the icon editor calculates which color in its
                   own color palette provides the nearest match to any bitmap
                   color.
     Delete        Clears the icon buffer.

Tools options
     Pen           Selects the pen drawing tool, enabling you to draw single
                   icon pixels, or lines or shapes by pressing and holding down
                   the left or right mouse button while you move the mouse.
     Fill          Selects the flood fill tool. Fills any shape with the
                   current color.
```

4.2.6.2 Help Editor

Displays a help editor where the current help text assigned to components can be modified. The *Help Editor* or viewer is actually the same window as the built-in *Text Editor*, but with different menu options and capabilities.

When used as a viewer of component help, no editing is allowed and the *File* menu provides printing options only. Consequently, the *Find & Replace* option of the *Edit* menu is not available.

The *Word Wrap* option of the *Character* menu can be used to toggle the insertion of extra line breaks at the right margin, so that the text fits in the window.

To edit the help file you must either select the *Help Editor* in the *Tools* menu or the *Help Editor* speed button in the *Edit Component/Edit User-Specified/Edit Model/Edit TACS* dialog boxes. In the latter case(s) a *Done* option appears in the main menu and the *File* menu provides printing options and a *Cancel* choice. By selecting *Done* you accept any changes made to the help text.
When the editor is invoked from the *Tools* menu, the *File* menu will contain an *Open* and a *Save* option. The text buffer is initially empty, so one has to select *File* | *Open* in order to load the help text of a support file.

The default help editor/view font can be changed by selecting the *Font* option in the *Character* menu. A detailed description of available options can be found in the next chapter.

4.2.6.3 Text Editor

Calls a text editor where user can create or edit standard text files. By default the built-in text editor (see Fig. 4.13) is displayed, but you can specify any text editor programs to use on the *Preferences* page of the *Tools* | *Options* dialog box.

To open the editor, select the *Text Editor* option in the *Tools* menu or the *Edit File* option in the *ATP* menu. In the latter case, a file with extension `.atp` and the same name as the name of the active circuit file is searched for and automatically loaded if it is found. When invoked from the *Tools* menu, the text buffer will initially be empty.

The main menu of the help editor/viewer or built-in text editor contains a *File* menu, an *Edit* menu and a *Character* menu. In addition, a *Done* option appears to the right of the *Character* menu if the editor is opened from the *Edit Component* dialog box. By selecting *Done*, changes made to the help text will be accepted.

A description of all the available options are described next:

File options

New	Opens an empty text buffer. *(Built-in text editor only!)*
Open	Loads the help text of a support file or the contents of a text file into the text buffer.
Save	Stores the contents of the text buffer to disk.
Save As	Stores the contents of the text buffer to a specified disk file. *(Built-in text editor only!)*
Print	Sends the contents of the text buffer to the default printer.
Print Setup	Enables you to define default printer characteristics.
Exit/Cancel	Closes the editor or viewer window. If the option displays Exit and the text buffer has been modified, you are given a chance to save the text before closing. If a Done option is available from the main menu, this option displays Cancel, and the window is closed without any warning with respect to loss of modified data.

Edit options

Undo	Cancels the last edit operation.
Cut	Copies selected text to the Clipboard and deletes the text from the buffer.
Copy	Puts a copy of the selected text in the Clipboard.

```
     Paste          Inserts the text in the Clipboard into the text buffer at
                    the current caret position.
     Delete         Deletes any selected text from the text buffer.
     Select All     Selects all the text in the buffer.
     Find           Searches the text buffer for the first occurrence of a
                    specified text string and jumps to and selects any matching
                    text found. This option displays the Windows standard Find
                    dialog box.


     Find Next      Searches for the next occurrence of the text string
                    previously specified in the Find dialog.
     Find&Replace   Searches the text buffer for one or all occurrences of a
                    specified text string and replaces any instance found with a
                    specified replacement string. This option displays the
                    Windows standard Replace dialog box.
Character options
     Word Wrap      Toggles wrapping of text at the right margin so that it fits
                    in the window.
     Font           From the Windows standard Font dialog box you can change the
                    font and text attributes of the text buffer.
```

The status bar at the bottom of the window displays the current line and character position of the text buffer caret and the buffer modified status. This status bar is not visible when viewing the component help.

The text buffer is limited to 32kB in size and will therefore not be suitable for editing large files. If you want to use your favorite text editor program rather than the built-in editor, you can specify the path of your program (wordpad.exe, write.exe, etc.) on the *Preferences* page of the *Tools | Options* dialog box.

4.2.6.4 Options

In the *Tools | Options* menu several user customizable program options for a particular ATPDraw session can be set and saved to the `ATPDraw.ini` file read by all succeeding sessions. During program startup each option is given a default value. Then, the program searches for an ATPDraw.ini file in the current directory, the directory of the `ATPDraw.exe` program, the Windows installation directory and each of the directories specified in the `PATH` environment variable. When an initialization file is found, the search process stops and the file is loaded. Any option values in this file override the default settings.

The *ATPDraw Options* dialog enables you to specify the contents of the initialization file without having to load and edit the file from a text editor. As shown on Fig. 4.29 it has four sub-pages: *General*, *Preferences*, *Directories* and *View/ATP*.

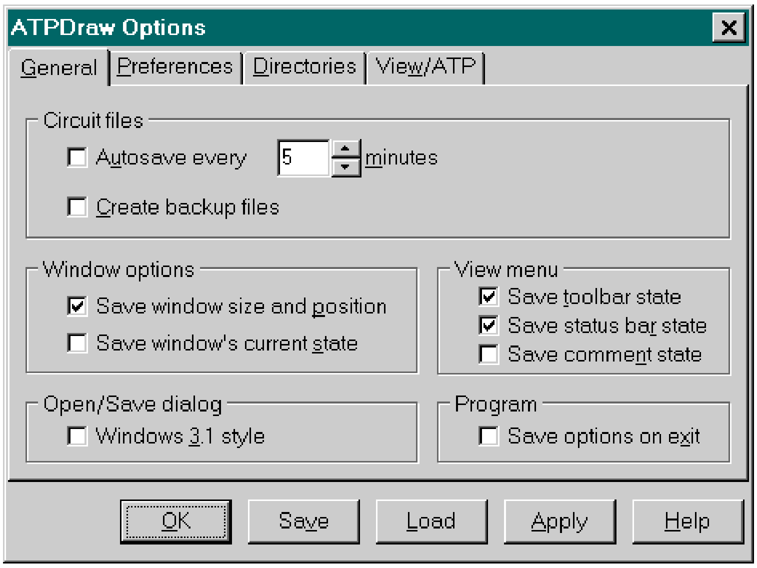


Fig. 4.29 - Customizing program options

General

The *General* tab specifies the circuit file and ATPDraw main window options. The following table lists and describes the available options:

Option	Description
Autosave every ? minutes	Saves all modified circuits to a separate disk file every specified interval of minutes. The file name is the same as the circuit file but with extension '.ci~'. The circuit's modified state does not change as a consequence of an autosave operation.
Create backup files	Changes the extension of the original circuit file to '.~ci' each time the circuit is saved. This option does not apply to autosave operations.
Save window size and position	Records the current size and position of the main window. The next time ATPDraw is started, it will be displayed with the same size and in the same position as the previous instance.
Save window's current state	Records the current main window state (maximized or normalized). The next time ATPDraw is started, it will be displayed in the same state.
Save toolbar state	Records the current view state (visible or hidden) of the main window toolbar, so it can be redisplayed in the same state next time ATPDraw is started.
Save status bar state	Records the current view state (visible or hidden) of the main window status bar, so it can be redisplayed in the same state next time ATPDraw is started.
Save comment state	Records the current view state (visible or hidden) of the circuit window comment line, so it can be redisplayed in the same state next time ATPDraw is started.
Windows 3.1 style	Causes the Open/Save dialogs to be drawn in the Windows 3.1 style.
Save options on exit	Causes program options to be automatically saved to the initialization file when the program is terminated.

Note that the 'save state' options will have no effect unless program options are saved to the initialization file (`ATPDraw.ini`) by the *Save* command at the bottom of the ATPDraw Options dialog, by the 'Save options on exit' option, or by the *Tools* | *Save Options* menu.

At the bottom of the *ATPDraw Options* dialog box there are five menu buttons provided with the following functionality:

Option	Description
OK	Stores current settings into program option variables, updates the screen and closes the dialog box. Changes made will only affect the current session.
Save	Saves the current settings to the ATPDraw.ini file.
Load	Loads settings from the ATPDraw.ini file.
Apply	Same as OK, but does not close the dialog box.
Help	Displays the help topic related to the options on the current page.

Note that if no initialization file exists, ATPDraw creates a new file in its installation directory when you select the *Save* button or the *Save Options* option in the *Tools menu*.

Preferences

On the *Preferences* page the user can specify the size of undo/redo buffers, circuit window color and text editor program.

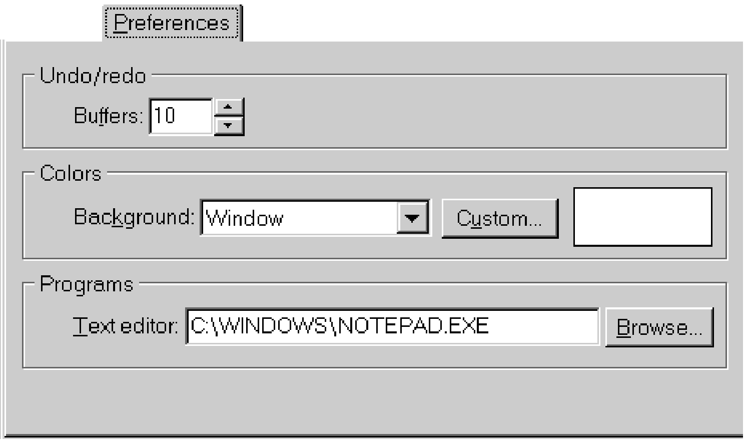


Fig. 4.30 - Customizable program options on the Preferences page

The following table lists and describes available options:

Option	Description
Undo/redo buffers	Specifies the number of undo and redo buffers to allocate for each circuit window. Changing this option does not affect the currently open circuit windows; only new windows will make use the specified value. Almost all object manipulation functions (object create, delete, move, rotate, etc.) can be undone (or redone). Since only a limited number of buffers are allocated, you are never guaranteed to undo all modifications.
Background color	Selects the background color of circuit windows. The color list provides available system colors, but you may customize your own from the Windows standard Color dialog displayed by the Custom button. The current color selection is shown in the box to the right of the Custom button.
Text editor program	Holds the name and path of the text editor program to use for editing ATP files (e.g. notepad.exe or wordpad.exe). If no program is specified (the field is empty), the built-in

	text editor is used. Note that the program specified must accept a filename on the command-line; otherwise, the ATP file will not be automatically loaded by the editor.

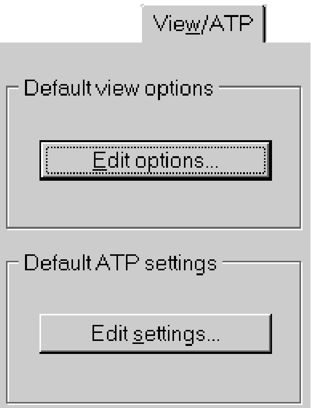
Directories

The following table lists and describes available options on the *Directories* page:

Option	Description
Circuit folder	The directory where you store your circuit files (.cir).
ATP folder	Specifies the directory in which .atp files are created
Support folder	The container of standard component support files (.sup).
Model folder	Directory containing support (.sup) and model (.mod) files for MODELS components.
TACS folder	The container of standard TACS component support files
User spec.folder	Directory containing support (.sup), library (.lib) and punch (.pch) files for user specified components.

View/ATP

Two groups of option settings can be initiated on the *View/ATP* page, as shown on Fig. 4.31. These are the default view and ATP options.



The *Edit options* button opens the *View Options* dialog which enables you to specify view options to apply as default to all new circuit windows. Available options are described in section 4.2.3.8.

The *Edit settings* button calls the ATP settings dialog described in section 4.2.4.4 of this manual.

Note that all circuit windows maintain their own set of view options, and only the circuit windows opened from this time on will use the options specified here. To change the view options of existing circuit windows, select the *View* | *Options* menu.

Fig. 4.31 - Setting default view and ATP options

Note that all circuits have their own settings; stored together with the objects in the circuit file. The settings you specify here will only be used by the new circuit files. To customize ATP settings of an existing circuit select the *Settings…*option in the *ATP* menu.

4.2.6.5 Save Options

Saves program options to the `ATPDraw.ini` file. This file is normally located in the program installation directory and can be used to store default option settings. At the end of the Reference Manual a complete description of ATPDraw's `.INI` file properties are given.

4.2.7 Window

From the *Window menu* the user can rearrange the open circuit windows or select the active circuit window. This menu contains a command for displaying or hiding the *Map window*. Fig. 4.32 shows the available menu options.

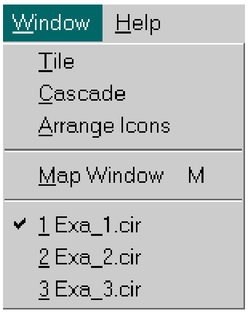


Fig. 4.32 - Supported options on the Window menu

Tile

The *Tile* command arranges the circuit windows horizontally in equal sizes to fit on your screen. To activate a circuit, click the title bar of the window. The active circuit window is marked by a ✓symbol in front of the name of the circuit file.

Cascade

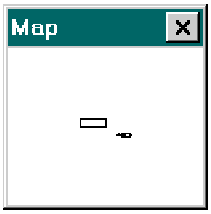
The *Cascade* command lays the circuit windows on each other with the title bars visible. To activate a circuit, click the title bar of the window.

Arrange Icons

The *Arrange Icons* command arranges the icons of the minimized circuit windows so that they are evenly spaced and don't overlap.

4.2.7.1 Map Window

The *Map Window* command displays or hides the map window. The map window is a stay-on-top window, meaning that it will always be displayed on top of all other windows.



This window displays the entire contents of the active circuit. The circuit window itself is represented by the map rectangle and circuit components are drawn as black dots.

Fig. 4.33 - Map window

When you press and hold down the left mouse button in the map rectangle, you can move it around in the map window. When you release the mouse button, the circuit window displays the part of the circuit defined by the new rectangle size and position, and the circuit window scrollbars are repositioned to reflect the updated circuit view.

If any circuit objects are currently selected when you reposition the map rectangle, selected objects will also be moved, and their relative position is retained in the new window. This functionality can be used to move a collection of objects quickly to a relatively large distance.

You can show or hide the map selecting the Map Window option in the *Window menu*, or by pressing the *M* key. In the latter case the user can quickly enable it when it is needed or hide it when it conceals vital circuit window information.

4.2.8 Help

The *Help menu* option pulls down the commands related to the on-line help available to the user of ATPDraw. This menu contains an option for displaying the help topics and ATPDraw copyright and version information.

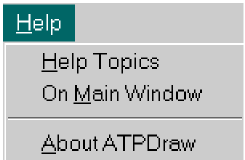


Fig. 4.34 - Help menu

ATPDraw's on-line help is a standard Windows dialog which provides help on all Main menu options of the program and also gives a short introduction how to build up a circuit.

4.2.8.1 Help Topics

The *Help Topics* command invokes a standard Windows tabbed help dialog box. Several links and a relatively large index register support the users in searching.

Selecting *Contents* tab you get a lists of available help functions as shown on Fig. 4.35.

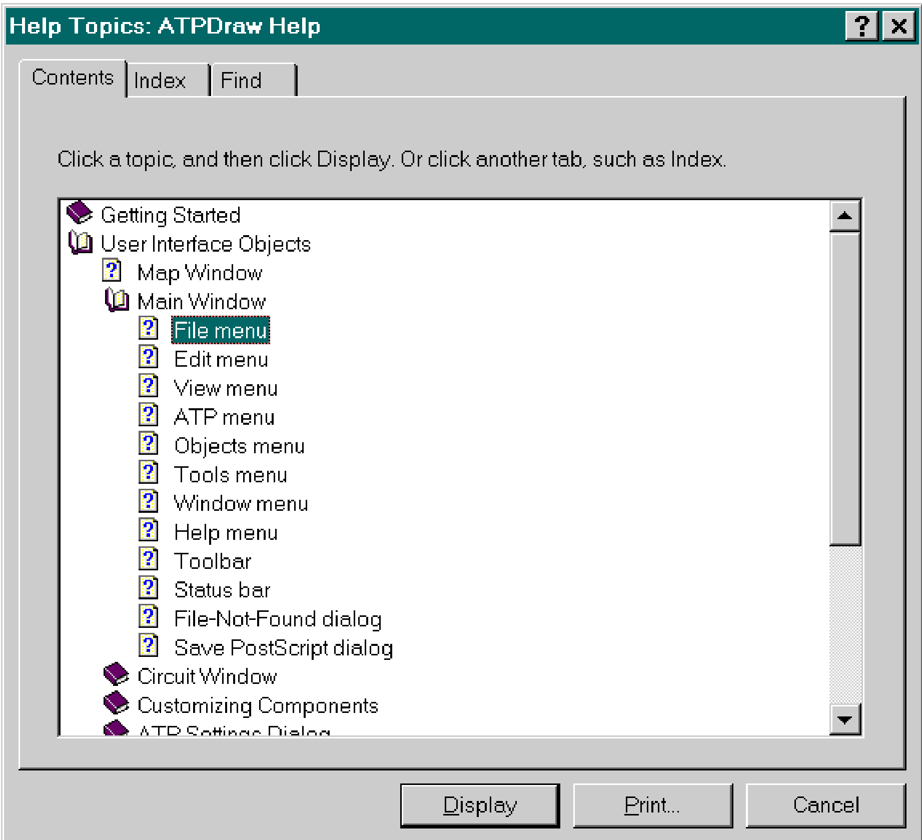
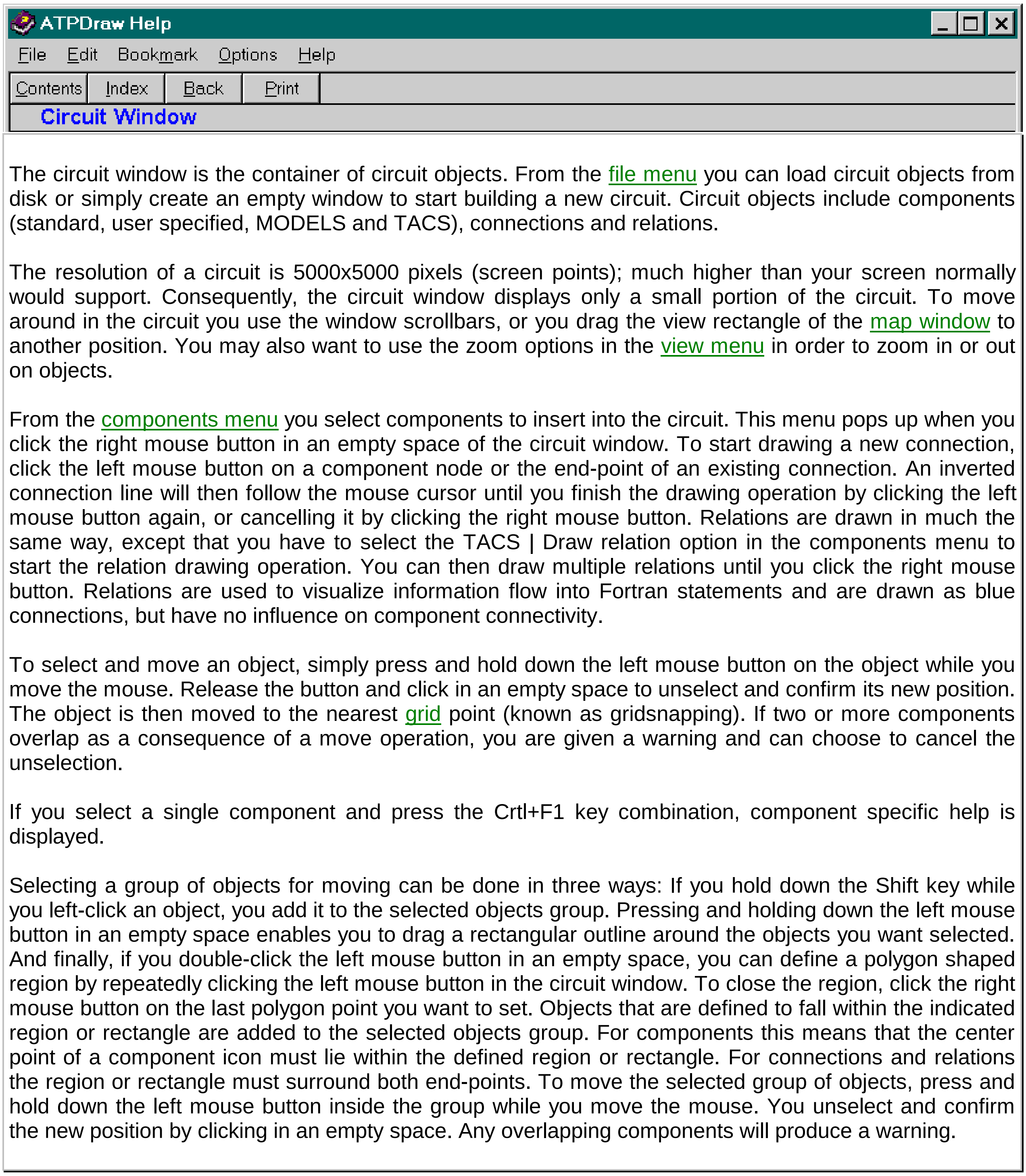


Fig. 4.35 - ATPDraw's on-line help

This page allows you to move through the list and select an entry on which you need help. To display an entry select one from the list by a simple mouse click and press *Display*, or double click on the entry with the mouse.

Index and *Find* tabs are used to obtain help by naming the topics you are looking for. E.g. if you ask for help on topics “Circuit Window” type these two words in the uppermost input field on the *Index* page and press the *Display* button. Then description of the Circuit Window topics will be displayed as shown below including several links marked as green underlined text:



The circuit window is the container of circuit objects. From the file menu you can load circuit objects from disk or simply create an empty window to start building a new circuit. Circuit objects include components (standard, user specified, MODELS and TACS), connections and relations.

The resolution of a circuit is 5000x5000 pixels (screen points); much higher than your screen normally would support. Consequently, the circuit window displays only a small portion of the circuit. To move around in the circuit you use the window scrollbars, or you drag the view rectangle of the map window to another position. You may also want to use the zoom options in the view menu in order to zoom in or out on objects.

From the components menu you select components to insert into the circuit. This menu pops up when you click the right mouse button in an empty space of the circuit window. To start drawing a new connection, click the left mouse button on a component node or the end-point of an existing connection. An inverted connection line will then follow the mouse cursor until you finish the drawing operation by clicking the left mouse button again, or cancelling it by clicking the right mouse button. Relations are drawn in much the same way, except that you have to select the TACS | Draw relation option in the components menu to start the relation drawing operation. You can then draw multiple relations until you click the right mouse button. Relations are used to visualize information flow into Fortran statements and are drawn as blue connections, but have no influence on component connectivity.

To select and move an object, simply press and hold down the left mouse button on the object while you move the mouse. Release the button and click in an empty space to unselect and confirm its new position. The object is then moved to the nearest grid point (known as gridsnapping). If two or more components overlap as a consequence of a move operation, you are given a warning and can choose to cancel the unselection.

If you select a single component and press the Crtl+F1 key combination, component specific help is displayed.

Selecting a group of objects for moving can be done in three ways: If you hold down the Shift key while you left-click an object, you add it to the selected objects group. Pressing and holding down the left mouse button in an empty space enables you to drag a rectangular outline around the objects you want selected. And finally, if you double-click the left mouse button in an empty space, you can define a polygon shaped region by repeatedly clicking the left mouse button in the circuit window. To close the region, click the right mouse button on the last polygon point you want to set. Objects that are defined to fall within the indicated region or rectangle are added to the selected objects group. For components this means that the center point of a component icon must lie within the defined region or rectangle. For connections and relations the region or rectangle must surround both end-points. To move the selected group of objects, press and hold down the left mouse button inside the group while you move the mouse. You unselect and confirm the new position by clicking in an empty space. Any overlapping components will produce a warning.

To move objects outside the visible part of the circuit, use the window scrollbars or the view rectangle in the map window. Any selected object or group will follow the window to its new position.

You rotate objects by clicking the right mouse button inside the selected object or group. Other object manipulation functions, such as undo/redo and clipboard options, are found in the edit menu. However, the most frequently used object manipulation functions can be accessed by holding down the Shift key while clicking with the right mouse button on an object or a selected group of objects. This will display and activate the circuit window shortcut menu.

Components and component nodes can be opened for editing. If you right-click or double-click an unselected component or node, either the Component, Open Probe or Open Node dialog box will show, allowing you to change component or node attributes and characteristics. If you double-click in a selected group of objects, the Open Group dialog box will show, allowing you to change attributes common to all components in that group, such as group number and hide and lock state. Default component attributes are stored in support files. Access to create and customize support files is provided by the objects menu.

Components are connected if their nodes overlap or if a connection is drawn between the nodes. To draw a connection between nodes, click on a node with the left mouse button. A line is drawn between that node and the mouse cursor. Click the left mouse button again to place the connection (clicking the right button cancels the operation). The gridsnap facility helps overlapping the nodes. Nodes connected together are given the same name by the Make Names and Make File options in the ATP menu.

Nodes can be attached along a connection as well as at connection end-points. A connection should not unintentionally cross other nodes (what you see is what you get). A node naming warning appears during the ATP file creation if a connection exists between nodes of different names or if the same name has been given to unconnected nodes. Connections are selectable as any other object. To resize a connection, click on its end-point with the left mouse button, hold down and drag. If several connections share the same node, the desired connection to resize must be selected. Selected connection nodes appear as squares at both ends of the selection rectangle.

4.2.8.2 On Main Window

The option *On Main Window* calls the help window and displays the help information related to the menu options on the Main menu of ATPDraw for Windows.

4.2.8.3 About ATPDraw

Selecting this field the ATPDraw's copyright and version information page will be displayed.

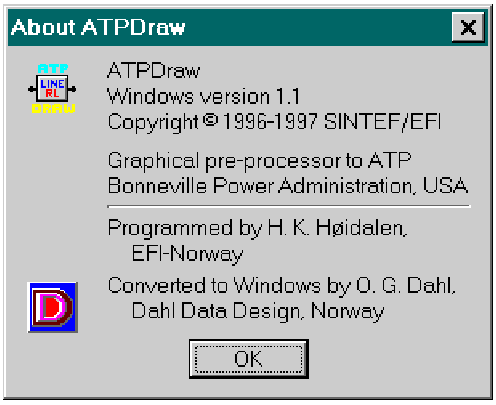
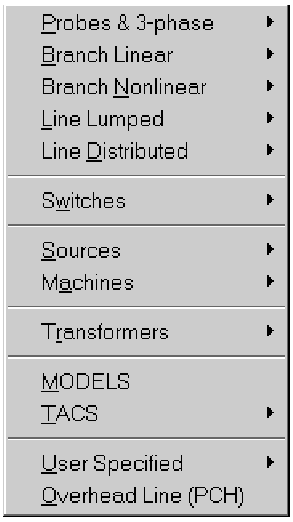


Fig. 4.36 - Information window about the program developers, copyright owner and sponsoring institution, and the program version actually used

4.2.9 Component selection menu

The *Component selection menu* provides options for inserting new components into the circuit window. The menu shown on Fig. 4.37 is normally hidden. To open and activate the menu, click the right mouse button in an empty circuit window area. In this menu all ATPDraw components are collected. After a selection in one of the floating menus, the selected object is drawn in the circuit window in marked moveable mode.



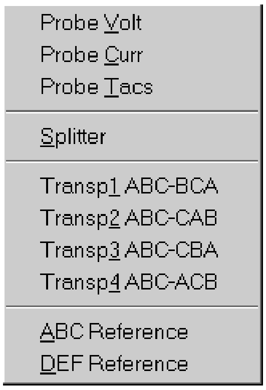
The upper four sections of the menu provide access to the probe, splitter and transposition objects, and to many standard ATP components: linear and nonlinear elements, switches, sources, transformers and machines.

The next section provides access to MODELS and TACS components. In the TACS submenu you will find a *Relation* option which enables you to visualize information flow into Fortran statements.

User specified components and overhead lines described as .PCH files (calculated by the Line/Cable Constant supporting routines of ATP) can be selected in the lowermost section.

Fig. 4.37 - Component selection menu

4.2.9.1 Probes & 3-phase



The menu *Probes & 3-phase* appears when the mouse moves over this item in the *Component selection menu* or when the user hits the *P* character.

Fig. 4.38 - Drawing objects on the Probe & 3-phase menu

Probes are components for monitoring the voltage, branch current or TACS values, and are handled differently than other components you open.

Probe Volt

Selecting this field draws the voltage probe used to specify voltage output request in the ATP file.

All probe objects have an *Open Probe* dialog window where the user can specify the number of phases to connect to and select phases to be monitored.

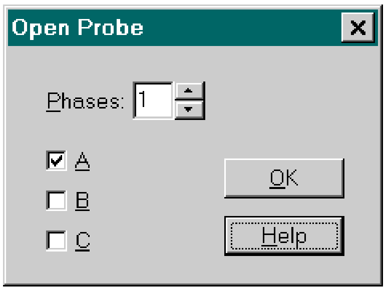
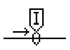


Fig. 4.39 - Open probe dialog box

Probe Curr

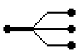


Selecting this field draws the current probe (measuring switch) used to specify current output request in column 80 in the ATP file. The number of phases to connect to and the phases to be monitored are user selectable.

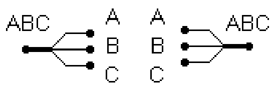
Probe Tacs

Selecting this field draws the Tacs probe used to specify signal output from TACS (type 33) in the ATP file.

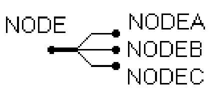
Splitter



The *splitter* object can be used as a transformation between a 3-phase node and three 1-phase nodes. The object has 0 data and 4 nodes. The object can be moved, rotated, selected, deleted, copied and exported as any other standard components.



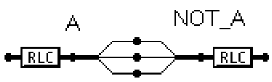
When a splitter is rotated the phase sequence of the single phase side changes as shown left.



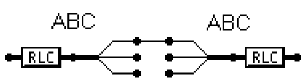
If a name has been given to the splitter’s three phase side, the letters *A*, *B*, *C* are added automatically on the single phase side as demonstrated left.

The following restrictions apply to this object:

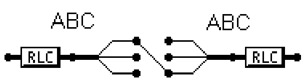
- do not connect splitters together directly on the 3-phase side (use a connection).
- normally do not give node names to the single phase side of the splitters.
- do not connect splitters together on the single phase side (e.g. all three examples shown below are illegal!).



nodes are _not_ connected

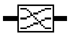


all phases are connected



transposition is not supported this way

Transp1 ABC-BCA



Selecting this field draws a *transposition* object used to change the phase sequence for 3-phase nodes from *ABC* to *BCA*.

Transp2 ABC-CAB

Selecting this field draws a *transposition* object used to change the phase sequence for 3-phase nodes from *ABC* to *CAB*.

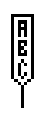
Transp3 ABC-CBA

Selecting this field draws a *transposition* object used to change the phase sequence for 3-phase nodes from *ABC* to *CBA*.

Transp4 ABC-ACB

Selecting this field draws a *transposition* object used to change the phase sequence for 3-phase nodes from *ABC* to *ACB*.

ABC reference



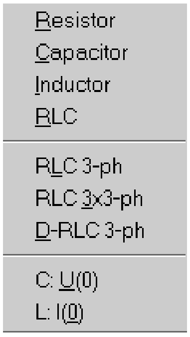
When attached to a 3-phase node in the circuit this node becomes the "master" node with phase sequence *ABC*. The other nodes will adapt this setting.

DEF reference

When attached to a 3-phase node in the circuit this node becomes the "master" node with phase sequence *DEF*. The other nodes will adapt this setting. A combination of *ABC* and *DEF* references is possible for e.g. in 6-phase circuits.

4.2.9.2 Branch Linear

This menu contains linear branch components.

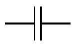
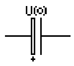
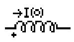


The name and the icon of linear branch objects, as well as a brief description of the components are given next in tabulated form.

Data parameters and node names to all components can be specified on the Attributes tab of the generalized *Component* dialog box (see Fig. 4.41/a) which is displayed when you click on the component's icon with the right mouse button in the circuit window.

Fig. 4.40 - Supported linear branch elements

There is a *Help* button in each component's dialog box that invokes the *Help Viewer* with description of the meaning of parameters and gives a reference to the corresponding ATP Rule-Book chapter. Fig. 4.41/b shows the help information associated with the ordinary RLC branch, as an example.

Selection	Object name	Icon	ATP card	Description
Resistor	`RESISTOR`		BRANCH type 0	Pure resistance in Ω
Capacitor	`CAPACITO`		BRANCH type 0	Pure capacitance in μF if Copt=0.
Inductor	`INDUCTOR`		BRANCH type 0	Pure inductance in mH if Xopt=0.
RLC	`RLC`		BRANCH type 0	R, L and C in series.
RLC 3-ph	`RLC_3`		BRANCH type 0	3-phase R, L and C in series. Symmetric. 3-phase nodes.
RLC 3x3-ph	`RLC_3X3`		BRANCH type 0	3-phase R, L and C in series. Non-symmetric. 3-phase nodes.
C : U(0)	`CAP_U0`		BRANCH + initial condition	Capacitor with initial condition.
L : I(0)	`IND_I0`		BRANCH + initial condition	Inductor with initial condition.

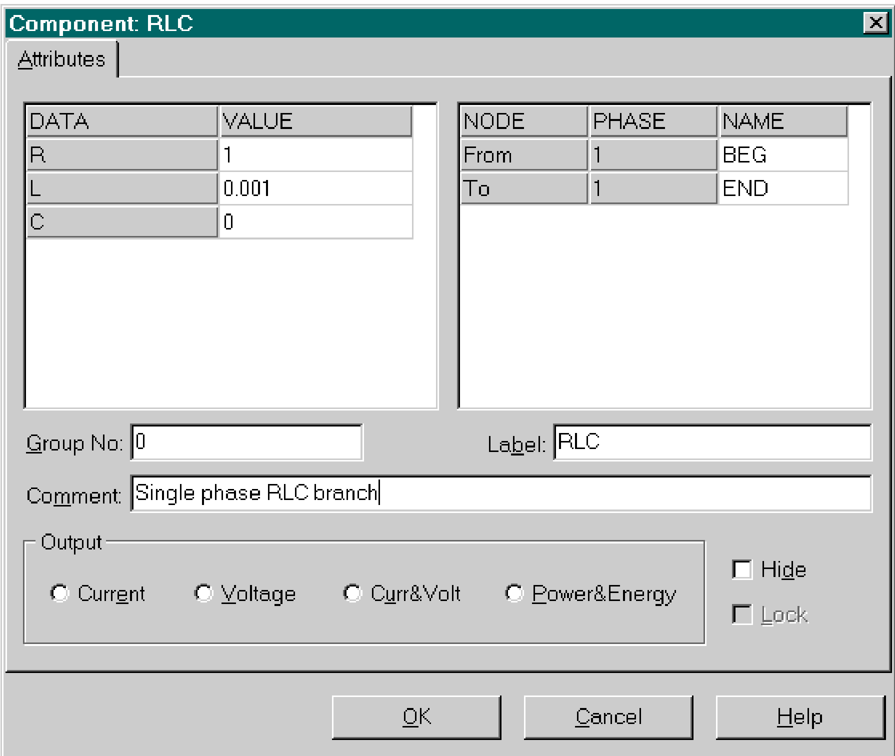


Fig. 4.41/a - Input dialog box for RLC branches

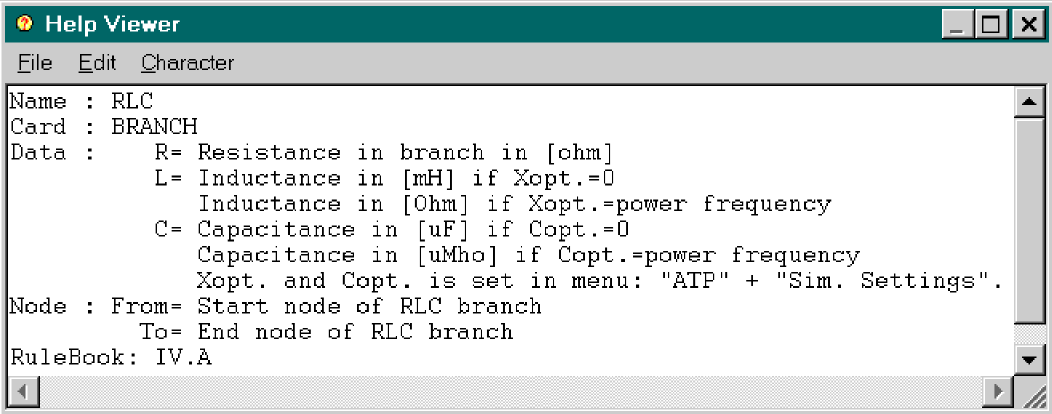


Fig. 4.41/b - Help information associated with the RLC branches

On the *Attributes* tab of the *Component* dialog box the users can specify component data parameters and node names in the top-left and top-right tables, respectively. Many data parameters have a legal range specified. To see what is the legal range of a parameter value, place the input caret in the data field and press the *Ctrl+F1* keys. Illegal values are issuing an error message when you move the caret to another data field, change the active page or select the *OK* button.

All components have a group number, serving as an optional sorting criterion for the ATP file (the components with the lowest group number are written first). You can specify the group number in the *Group No* field.

Component label and comment are set in the *Label* and *Comment* fields. The visibility of the component label is controlled by the *Labels* option in the *View* | *Options* menu.

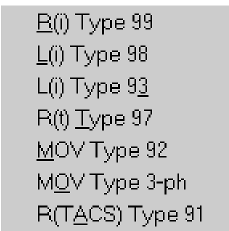
The *Hide* and *Lock* buttons are common to all components. Hidden components are not included in the ATP file and are displayed as light-gray icons. The *Lock* option is **not implemented** in this version, but is provided here for compatibility with future versions. Locked components are meant to have fixed positions and not subject for customization of component data and node names.

Many standard components such as branches, non-linear, switches and transformers contain an *Output* section for setting the branch output request. Possible values are Current, Voltage, Current&Voltage, Power&Energy or none (no button selected).

To obtain component specific help about data parameters, select the Help button.

4.2.9.3 Branch Nonlinear

This menu contains the available nonlinear components.



All the objects except the TACS controlled resistor can also have a nonlinear characteristic. These attributes can be specified by selecting the *Characteristic* tab of the notebook style *Component* dialog boxes as shown in Fig. 4.43.

Fig. 4.42 - Nonlinear branch elements

Selection	Object name	Icon	ATP card	Description
R(i) Type 99	NLINRES		BRANCH type 99	Current dependent resistance
L(i) Type 98	NLININD		BRANCH type 98	Current dependent inductance
L(i) Type 93	NLIND93		BRANCH type 93	Current dependent inductance
R(t) Type 97	NLINR_T		BRANCH type 97	Time dependent resistance
MOV Type 92	MOV		BRANCH type 92	Current dependent resistance on exponential form.
MOV Type 3-ph	MOV_3		BRANCH type 92	3-phase current dependent resistance.
R(TACS) Type 91	TACSRES		BRANCH type 91	TACS / MODELS controlled resistor.

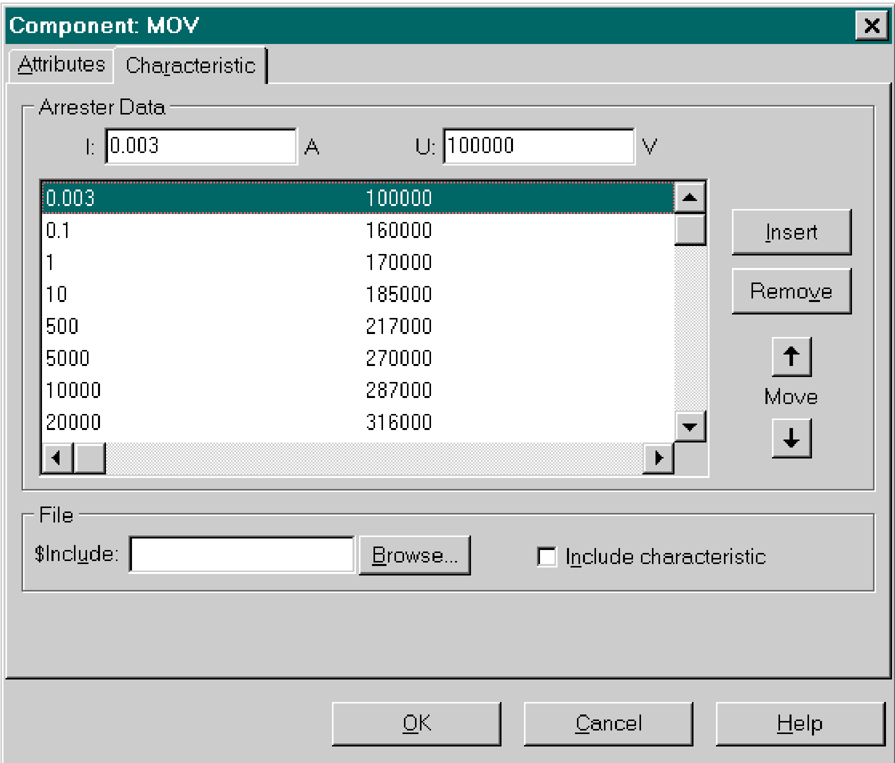


Fig. 4.43 - Input window for nonlinear characteristic

The *Characteristic* page of the component dialog box specifies the input characteristic for a non-linear object. New pairs of input values are set in the data field at the top of the page. When you press the *Enter* key in the rightmost field, or select the *Insert* button, the current input values are transferred to the points table in the position above the currently marked line.

To delete a line from the table, select one or more lines by pressing and holding down the left mouse button while dragging the mouse cursor up or down. Release the mouse button and select the *Remove* button. The ↑ and ↓ arrow buttons can be used to change the position of the currently selected line in the table step-by-step up or down.

If you specify a metal oxide arrester with MOV Type-92 component, ATPDraw accepts the current/voltage characteristic for the arrester and performs an exponential fitting in the logarithmic ln-ln domain and produces the required ATP data format.

The *File* section at the bottom of the page contains an *$Include* field where you can specify the name of a standard text file containing input characteristic values (and the final 9999. card). If the *Include characteristic* button is checked, this file will be referenced in an `$INCLUDE` statement in the ATP file rather than including each of the value pairs from the points table. This file must have extension `.LIB` and be stored in the `\USP` directory, if the *Auto path* check box is selected on the *File format* page of the *ATP Settings* dialog box, because in that case ATPDraw writes `$PREFIX` and `$SUFFIX` to the ATP file. If the file is located somewhere else (i.e. not in the `\USP` directory) use the *Browse* button to specify its path and uncheck the *Auto path* option.

4.2.9.4 Line Lumped

Selecting *Line Lumped* will display a popup menu where three different types of line models can be chosen. All the line models are lumped elements, frequency independent lines.

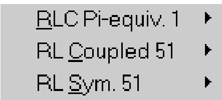
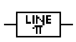
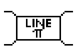


Fig. 4.44 - Frequency independent line models

RLC Pi-equiv. 1

These line models are simple, lumped, non-symmetric pi-equivalents of ATP type 1, 2, 3 etc. Three selections are available:

Selection	Object name	Icon	ATP card	Description
RLC Pi-equiv. 1 +1 ph	`LINEPI_1`		BRANCH type 1	Single phase RLC pi-equivalent
RLC Pi-equiv. 1 + 2 ph	`LINEPI_2`		BRANCH type 1-2	Two phase RLC pi-equivalent Non-symmetric.
RLC Pi-equiv. 1 + 3 ph	`LINEPI_3`		BRANCH type 1-3	Three phase RLC pi-equivalent Non-symmetric. 3-phase nodes.

RL Coupled 51

These line models are simple, lumped, non-symmetric mutually RL coupled components of type 51, 52, 53 etc. The popup menu has three selections:

Selection	Object name	Icon	ATP card	Description
RL Coupled 51 + 2 ph	LINERL_2		BRANCH type 51-52	Two phase RL coupled line model. Non-symmetric.
RL Coupled 51.. + 3 ph	LINERL_3		BRANCH type 51-53	Three phase RL coupled line model. Non-symmetric. 3-phase nodes
RL Coupled 51.. + 6 ph	LINERL_6		BRANCH type 51-56	2x3 phase RL coupled line model. Non-symmetric. 3-phase nodes. Off-diagonal R's set to zero.

RLC Sym. 51

These line models are symmetric with sequence value input. The line models are special applications of the RL coupled line models in ATP. The popup menu has two selections available:

Selection	Object name	Icon	ATP card	Description
RL Sym. 51 + 3 ph	LINESY_3		BRANCH type 51-53	Three phase RL coupled line model with sequence (0, +) input. Symmetric. 3-phase nodes.
RL Sym. 51 + 6 ph	LINESY_6		BRANCH type 51-56	2x3-phase RL coupled line model with sequence (0, +) input. Symmetric. 3-phase nodes.

4.2.9.5 Line Distributed

Selecting *Line Distributed* opens a popup menu where two different types of line models can be selected. All the line models are distributed parameters, frequency independent lines.

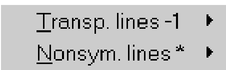


Fig. 4.45 - Distributed transmission line models

Transp. lines -1

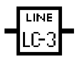
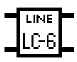
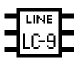
These components can be characterized as symmetric, distributed parameter and lumped resistance models (called as Clarke-type in the ATP Rule-Book). Five different types are supported:

Selection	Object name	Icon	ATP card	Description
Transp. lines -1 + 1 ph	LINEZT_1		BRANCH type -1	Single phase, distributed parameter line, KCLee (Clarke) model.
Transp. lines -1 + 2 ph	LINEZT_2		BRANCH type -1.. -2	Two phase, distributed parameter, transposed line, Clarke model.

Transp. lines -1 + 3 ph	LINEZT_3		BRANCH type -1.. -3	Three phase, distributed parameter, transposed line, Clarke model.
Transp. lines -1 + 6 ph	LINEZT_6		BRANCH type -1.. -6	Six phase, distributed parameter, transposed line, Clarke model. Two parallel 3-phase lines with mutual coupling.
Transp. lines -1 + 9 ph	LINEZT_9		BRANCH type -1.. -9	Nine phase, distributed parameter, transposed line, Clarke model.

Nonsym. lines *

These line models are some sort of "dummy" lines. No data is written to the ATP file, only the node names are specified in between the ***** separators. Parameters of that nonsymmetrical lines are generated outside ATPDraw. The popup menu has three selections available:

Selection	Object name	Icon	ATP card	Description
*Nonsym. lines * + 3 ph*	LINELC_3		BRANCH	Three phase "dummy" line. Node names only. 3-phase nodes.
*Nonsym. lines * + 6 ph*	LINELC_6		BRANCH	Six phase "dummy" line. Node names only. 3-phase nodes.
*Nonsym. lines * + 9 ph*	LINELC_9		BRANCH	Nine phase "dummy" line. Node names only. 3-phase nodes.

4.2.9.6 Switches

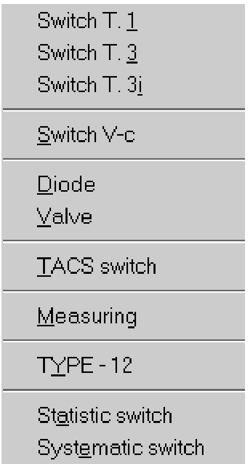
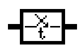
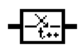
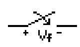
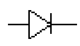
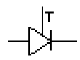
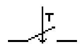
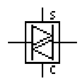


Fig. 4.46 - Supported switch type ATP components

ATPDraw supports most of the switch type elements in ATP, such as ordinary time- or voltage controlled switches, options for modeling diodes, valves and triacs, as well as statistics/systematics switches. The popup menu contains the following items:

Selection	Object name	Icon	ATP card	Description
Switch T.1	SWITCHTC		SWITCH type 0	Time controlled switch.
Switch T.3	SWITCH_3		SWITCH type 0	Three-phase time controlled switch, symmetric. 3-phase nodes.
Switch T.3i	SWIT_3XT		SWITCH type 0	Three-phase time controlled switch, non-symmetric. 3-phase nodes.
Switch V-c	SWITCHVC		SWITCH type 0	Voltage controlled switch.
Diode	DIODE		SWITCH type 11	Diode. Switch type 11. Uncontrolled.
Valve	VALVE		SWITCH type 11	Valve/Thyristor. Switch type 11. TACS/MODELS- controlled.
TACS switch	TACSSWIT		SWITCH type 13	Simple TACS/MODELS controlled switch.
Measuring	SWMEAS		SWITCH type 0	Measuring switch. Current measurements.
Type-12	TRIAC		SWITCH type 12	Double TACS/MODELS controlled switch.
Statistic switch	STATSWIT		SWITCH	Statistic switch. See *ATP + Switch settings*.
Systematic switch	SYSTSWIT		SWITCH	Systematic switch. See *ATP + Switch settings*.

4.2.9.7 Sources

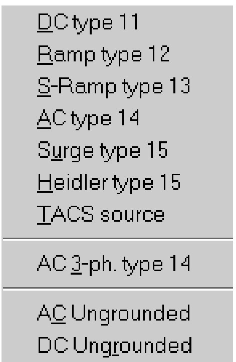
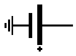
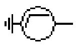
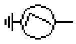
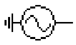
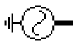
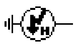
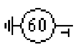
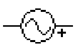
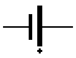


Fig. 4.47 - Electric sources in ATPDraw

The popup menu under *Sources* contains the following items:

Selection	Object name	Icon	ATP card	Description
DC type 11	DC1PH		SOURCE type 11	DC step source. Current or voltage.
Ramp type 12	RAMP		SOURCE type 12	Ramp source. Current or voltage.
S-Ramp type 13	SLOPE_RA		SOURCE type 13	Two-slope ramp source. Current or voltage.
AC type 14	AC1PH		SOURCE type 14	AC source. Current or voltage.
AC 3-ph. type 14	AC3PH		SOURCE type 14	AC source. Current or voltage. 3-phase node.
Surge type 15	SURGE		SOURCE type 15	Two-exponential source. Current or voltage.
Heidler type 15	HEIDLER		SOURCE type 15	Heidler type source. Current or voltage.
TACS source	TACSSOUR		SOURCE type 60	TACS/MODELS controlled source. Current or voltage.
AC ungrounded	AC1PHUG		SOURCE type 14+18	Ungrounded AC source. Voltage.
DC ungrounded	DC1PHUG		SOURCE type 11+18	Ungrounded DC source. Voltage.

4.2.9.8 Machines

Two categories of electrical machines are available in ATPDraw: Synchronous machines and Universal machines. ATPDraw does not support machines in parallel or back to back. All 3-phase machines are Y-connected.

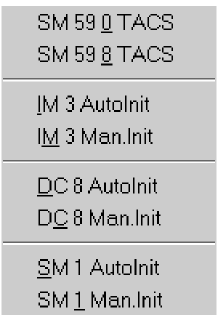


Fig. 4.48 - Supported electric machine alternatives

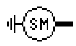
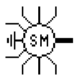
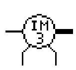
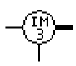
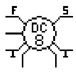
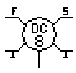
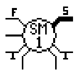
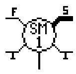
The *Synchronous machine* models in ATPDraw have the following features/limitation:

- With and without TACS control.
- Manufacturers data.
- No saturation.
- No eddy-current or damping coils.
- Single mass.

The *Universal machine* models in ATPDraw have the following features/limitation:

- Manual and automatic initialization.
- SM, IM and DC type supported.
- Raw coil data (internal parameters).
- Saturation.
- One single excitation coil in each axis (d, q).
- Network option for mechanical torque only.
- Single torque source.

The popup menu under *Machines* contains the following items:

Selection	Object name	Icon	ATP card	Description
SM 59 0 TACS	SM59_NC		MACHINE type 59	Synchronous machine. No TACS control. 3-phase armature.
SM 59 8 TACS	SM59_FC		MACHINE type 59	Synchronous machine. Max. 8 TACS control. 3-phase armature.
IM 3 AutoInit	IM_3AI		UM-MACHINE type 3	Induction machine. Automatic initialization. 3-phase armature.
IM 3 Man.Init	IM_3MI		UM-MACHINE type 3	Induction machine. Manual initialization. 3-phase armature.
DC 8 AutoInit	DC_8AI		UM-MACHINE type 8	DC machine. Automatic initialization. 1-phase armature.
DC 8 Man.Init	DC_8MI		UM-MACHINE type 8	DC machine. Manual initialization. 1-phase armature.
SM 1 AutoInit	SM_1AI		UM-MACHINE type 1	Synchronous machine. Automatic initialization. 3-phase armature.
SM 1 Man.Init	SM_1MI		UM-MACHINE type 1	Synchronous machine. Manual initialization. 3-phase armature.

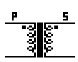
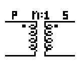
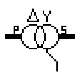
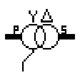
Next, a part from the help file is shown for the Type-3 Universal machines describing the modeling of the external electrical network for torque representation.

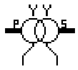
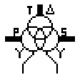
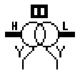
```
Mechanical network:
   * Shaft mass (moment of inertia) <-> Capacitance
     ( 1kg/m2 <-> 1 Farad )
   * Shaft section (spring constant) <-> Inverse inductance.
     ( 1 Nm/rad <-> 1/Henry )
   * Shaft friction (viscous damping) <-> Conductance.
     ( 1 Nm/rad/s <-> 1/ohm )
   * Angular speed <-> Voltage
     (1 rad/s <-> 1 Volt )
   * Torque <-> Current
     (1 Nm <-> 1 Amp )
   * Angle <-> Charge
     (1 rad <-> 1 Coulomb )
                                                    L1
      __       __                    ____ ___oooo________________
     |  |     |  |                  |    |         |    |     |
     |J1| K1  |J2|   --> T         _|_   <        _|_   <     |
   ==|  |=====|  |====      ====>  ___C1 >R1      ___C2 >R2   O I
     |  |     |  |                  |    <         |    <     |
     |__|     |__|                  |    |         |    |     |
      D1       D2

      C1=J1, C2=J2, R1=1/D1, R2=1/D2, L1=1/K1, I=T
```

4.2.9.9 Transformers

Fig. 4.49 - Available transformer models

The popup menu under *Transformers* contains the following items:

Selection	Object name	Icon	ATP card	Description
Sat. 1 phase	TRAFO_S		BRANCH TRANSFORMER	Single phase saturable transformer.
Ide. 1 phase	TRAFO_I		SOURCE type 18	Single phase ideal transformer.
Sat. Δ/Y 3 ph	TRADY_3		BRANCH TRANSFORMER	3x1-phase saturable transformer. D/Y coupled windings. 3-phase node.
Sat. Y/Δ 3 ph	TRAYD_3		BRANCH TRANSFORMER	3x1-phase saturable transformer. Y/D coupled windings. 3-phase node.

Sat. Y/Y 3 ph	TRAYY_3		BRANCH TRANSFORMER	3x1-phase saturable transformer. Y/Y coupled windings. 3-phase node.
Sat. Δ/Δ 3 ph	TRADD_3		BRANCH TRANSFORMER	3x1-phase saturable transformer. D/D coupled windings. 3-phase node.
Sat. Y/Y/Δ 3 ph	TRAYYD_3		BRANCH TRANSFORMER	3x1-phase saturable transformer. 3 win. Y/Y/D coupled windings. 3-phase node.
Sat. Y/Y 3-leg	TRAYYH_3		BRANCH TRANSFORMER THREE PHASE	3-phase saturable transformer. High homopolar reluct. (3-leg). 3-ph node. Preprocessing of manufact. data.

The nonlinear magnetizing characteristic of the saturable transformers can be given in the Characteristic page of the component dialog box. The saturable transformers have an input window like the one shown in Fig. 4.43. In this window data for the saturating magnetizing branch can be specified. The *RMS* flag indicates the form of the non-linear characteristic. If the check box *Include characteristic* is selected on the Attributes page of the component dialog box, the disk file referenced in the *$Include* field will be used. Thus three alternatives for input format are supported:

- Flux/current (λ/i) (if the flag *RMS* in the input window is equal to 0)
- U_{rms}/I_{rms} (if the flag *RMS* in the input window is equal to 1)
- Include parameters with $Include

If *RMS*=1, ATPDraw calculates the flux/current values used in the ATP file, automatically.

4.2.9.10 MODELS

Besides the standard components, the user can create his own models using the MODELS simulation language in ATP. Using this feature requires knowledge about the syntax and general structure of MODELS language. The process normally consists of two steps:

1. To create a new support file (`.SUP`) using the *New Model* command in the *Objects menu*.
2. To create a model file (`.MOD`) containing the actual model description.

The Advanced part of this manual gives more information about the use of Models objects in ATPDraw and about the procedure to create a new one.

Selecting *MODELS* in the component selection menu performs an Open Model dialog box where the user can choose a model file name.

The *Component* dialog box of any Models objects has a new input section *Models* besides the specification of *DATA* values and the name of the *NODES* as shown in Fig. 4.50. This new section has two input fields: *Model file* for locating the model description file and and a *Use As* field for specification of the *model_name* in the `USE model AS model_name` statement of MODELS in ATP [4].

Input and output to MODELS and usage of each Model are handled automatically by ATPDraw. The general structure of MODELS section in an ATP file created by ATPDraw is shown next:

```
MODELS
INPUT
  IX0001 {v(NODE1 )},
  IX0002 {v(NODE2 )},
OUTPUT
  FIRE1 ,
  FIRE2 ,
  FIRE3 ,
  FIRE4 ,
  FIRE5 ,
  FIRE6 ,
$INCLUDE, C:\ATPDRAW\MOD\VALVE_6.mod
USE VALVE_6 AS VALVE_6
INPUT
  UA:= IX0001
  UC:= IX0002
DATA
  ALPHA:= 3.000E+0001
OUTPUT
  FIRE1 :=GATE1
  FIRE2 :=GATE2
  FIRE3 :=GATE3
  FIRE4 :=GATE4
  FIRE5 :=GATE5
  FIRE6 :=GATE6
ENDUSE
ENDMODELS
```

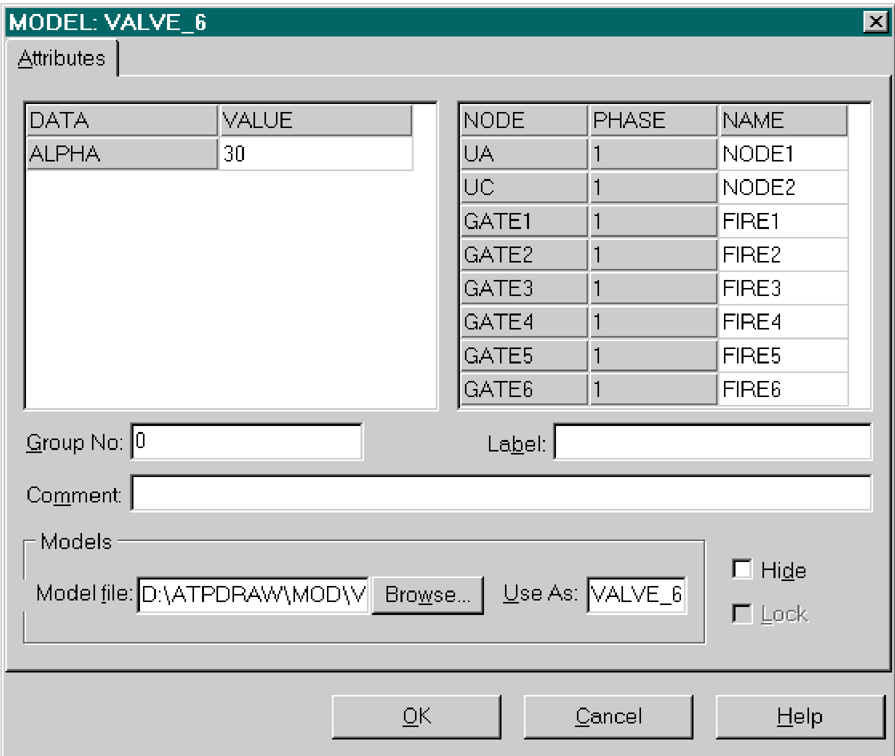


Fig. 4.50 - Model input window

4.2.9.11 TACS

Most of ATP's TACS objects are available in ATPDraw. The exceptions are Devices 55-57 and multi-line Fortran statements ($ continuation). The TACS sub-menu on the component selection menu contains the following items:

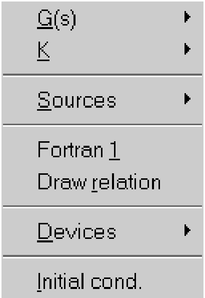
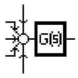
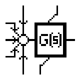
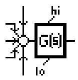
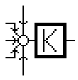
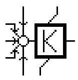
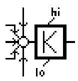


Fig. 4.51 - Supported TACS objects

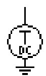
Transfer functions

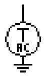
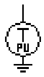
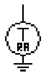
The first item *G(s)* defines a transfer functions in the *s* domain, the next one a constant *K* gain factor. Both can be specified with or without limits:

Selection	Object name	Icon	ATP card	Description
G(s) *+ No limits*	`G(S)_NOL`		TACS	Laplace transfer function. Order 0-9 No limits.
G(s) *+ Named limits*	`G(S)_NAL`		TACS	Laplace transfer function. Order 0-9 Named dynamic limits.
G(s) *+ Fixed limits*	`G(S)_FIL`		TACS	Laplace transfer function. Order 0-9 Fixed dynamic limits.
K *+ No limits*	`K_NOL`		TACS	Transfer function. Order 0 No limits.
K *+ Named limits*	`K_NAL`		TACS	Transfer function. Order 0 Named static limits.
K *+ Fixed limits*	`K_FIL`		TACS	Transfer function. Order 0 Fixed static limits.

Sources

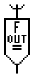
The *Sources* menu contains the following items:

Selection	Object name	Icon	ATP card	Description
DC - 11	`DC_01`		TACS type 11	TACS DC source.

AC - 14	AC_02		TACS type 14	TACS AC source.
Pulse - 23	PULSE_03		TACS type 23	TACS Square pulse train.
Ramp - 24	RAMP_04		TACS type 24	TACS Sawtooth train.
EMTP out	EMTP_OUT		TACS type 90-93	Value from the electrical circuit into TACS. 90 - EMTP node voltage 91 - EMTP switch current 92 - internal variable special EMTP comp. 93 - EMTP switch status.

Fortran 1

Only single phase Fortran statements are supported in ATPDraw for Windows. The object icon is shown below:



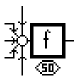
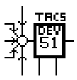
The *OUT=* input field in the *FORTRAN* section at the bottom of the component dialog box can be used for specification of the Fortran expression. This expression is written into the ATP file starting in column 12 in the TACS card. The user can also define the *Type* of the object (88, 98 or 99) in the *DATA* field.

Draw relation

Relations are used to visualize information flow into Fortran statements. These objects are drawn as blue connections, but do not influence the connectivity of components. They are used exactly the same way as connections. When you select *Draw relation*, the mouse cursor will change to a pointing hand and the program is waiting for a left mouse click on a circuit node to set the starting point of the new relation. To cancel drawing a relation, click the right mouse button or press the *Esc* key.

Devices

The *Devices* menu consists of the following items:

Selection	Object name	Icon	ATP card	Description
freq_sensor - 50	DEVICE50		TACS type 88,98 or 99	Frequency sensor
relay switch - 51	DEVICE51		TACS type 88,98 or 99	Relay switch

level switch - 52	DEVICE52		TACS type 88,98 or 99	Level switch
trans_delay - 53	DEVICE53		TACS type 88,98 or 99	Transport delay
pulse_delay - 54	DEVICE54		TACS type 88,98 or 99	Pulse delay
cont_integ - 58	DEVICE58		TACS type 88,98 or 99	Controlled integrator
simple_deriv - 59	DEVICE59		TACS type 88,98 or 99	Simple derivative
input_IF - 60	DEVICE60		TACS type 88,98 or 99	Input-IF component
signal_select - 61	DEVICE61		TACS type 88,98 or 99	Signal selector
sample_track - 62	DEVICE62		TACS type 88,98 or 99	Sample and track
inst_min/max - 63	DEVICE63		TACS type 88,98 or 99	Instantaneous min/max
min/max_track - 64	DEVICE64		TACS type 88,98 or 99	Min/max tracking
acc_count - 65	DEVICE65		TACS type 88,98 or 99	Accumulator and counter
rms meter - 66	DEVICE66		TACS type 88,98 or 99	RMS value of signal

Initial Cond.

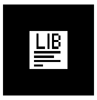
The initial condition of TACS variables could be set by the `INIT_T` object found in the *Initial cond.* menu under *TACS*. The `INIT_T` icon is shown here:

4.2.9.12 User Specified

The *User Specified* menu has four items:

Fig. 4.52 - Supported user specified objects

Library



Selecting the *Library* item will draw the predefined user specified object `LIB`. This object has no data and no nodes. Using this object in a circuit will result just in a $Include statement in the ATP file. The user must keep track of internal node names in the include file. The name and the path of the file can be specified in the *User specified* section at the bottom of the component dialog box.

Ref. 1-ph

Selecting *Ref. 1-ph* will draw the object `LIBREF_1.` This object is a “dummy” object that is not written to the ATP file. Its purpose is just to visualize a connection inside a e.g. `LIB` object. The object is single phase one.

Ref. 3-ph

Selecting *Ref. 3-ph* will draw the object `LIBREF_3.` This object is a “dummy” object that is not written to the ATP file. Its purpose is just to visualize a connection inside a e.g. `LIB` object. The object is three- phase one.

Files

Besides the standard components, the user can create his own User Specified Objects. The usage of this feature requires knowledge about the Data Base Modularization in ATP. The process normally consists of two steps:

1. Creating a new support file (`.SUP`) using the *New User Specified* command in the *Objects menu.*
2. Creating a Data Base Module punch file (`.LIB`) containing the object description.

The Advanced part of this manual gives more information about the user specified objects in ATPDraw and about the procedure to create a new one.

Selecting *Files* in the component selection menu, an *Open Component* dialog appears where the support files of the user specified objects in the `\USP` directory are listed. If you select a `.SUP` file from the list and click *Open*, the icon of the object will be displayed in the middle of the active circuit window. This way any existing components (even the standard components in the `\SUP` directory) can be loaded into the circuit.

4.2.9.13 Overhead line (PCH)

A completely separate program called ATP_LCC has been developed for line/cable constant support. In this program the user can specify the cross section and material data for an overhead line or a cable. Based on this data, the corresponding ATP file is generated ready to be processed by ATP for creation of punched output (PCH file).

In most cases ATPDraw can read these punch files and create a `.LIB` file equivalent with the output of the Data Base Modul routine.

Selecting *Overhead Line (PCH)* the program performs an *Open Punch File* dialog in which the available `.PCH` files are listed. If you select a file from the list and click *Open*, ATPDraw interprets the file (see Fig. 4.53) and generates a `.LIB` file in the `\USP` sub-folder for the recognized file types, exactly the same way as it were created by the Data Base Modularization.

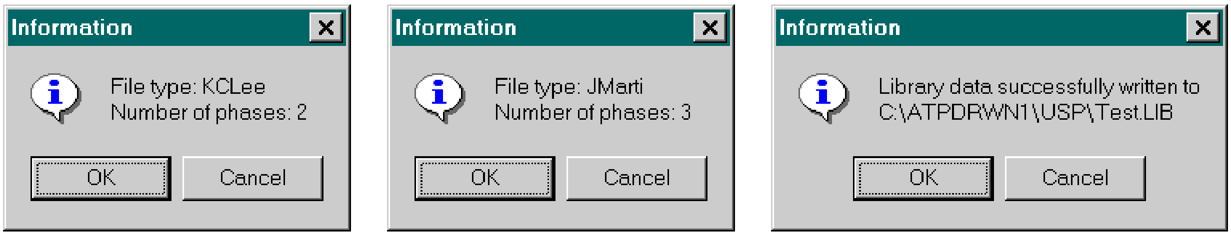
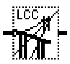


Fig. 4.53 - Interpretation results are communicated in dialog boxes



When the `.LIB` file is written successfully, the icon of the just created new LCC component appears in the middle of the circuit window.

4.2.10 Shortcut menu

The *Shortcut menu* provides access to the most frequently used object manipulation functions. To show and activate the shortcut menu, hold down the *Shift* key while you click the right mouse button on an object or a selected group of objects in the circuit window.

`Open`	`enables component customization by bringing up the Component dialog box of the object.`
`Cut, Copy` `Delete, Dupl.`	`Access to the standard clipboard functions`
`Flip, Rotate.`	`Rotating and flipping the objects' icon`
`Select` `Unselect`	`Select/unselect the object(s) in question.`

Fig. 4.54 - Available options in the Shortcut menu

4.2.11 Settings in the ATPDraw.ini file

The ATPDraw initialization file `ATPDraw.ini` contains user specified program options. On program startup all program option variables are given default values. Then, ATPDraw searches the disk for an initialization file and if one is found, new parameter values are read from the file into the option variables, overriding the default values. The initialization file is searched for in the following order:

1. In the current directory.
2. In the program directory (where `ATPDraw.exe` is located).
3. In Windows installation directory (normally `c:\windows`).
4. In each of the directories specified in the PATH environment variable.

When a file is found the search process terminates and the file is read. If no file is found, the default values apply.

The ATPDraw.ini file is a standard Windows initialization file. Empty lines or lines that starts with a semicolon (;) sign are ignored. A standard initialization file is divided into one or more sections, each section contains one or more parameter = value lines. The ATPDraw.ini file has 7 sub-sections. The following tables list and describe the name and legal value range of available parameters, as well as the default settings. One line is required for each parameter and at least one space is needed between the parameter and any in-line comment string.

[ATPDraw]

Parameter	Range	Description
AutoSave	On|Off	Enables/disables circuit file auto saving.
AutoSaveInterval	1-60	Specifies the autosave interval in number of minutes.
CreateBackupFile	On|Off	Enables/disables the creation of circuit backup files.
SaveWindowSizePos	On|Off	Enables/disables the storing of window size and position.
SaveWindowState	On|Off	Enables/disables the recording of main window current state.
SaveToolbarState	On|Off	Enables/disables the recording of toolbar visibility state.
SaveStatusBarState	On|Off	Enables/disables the recording of status bar visibility state.
SaveCommentLineSt	On|Off	Enables/disables the recording of circuit window comment line visibility state.
Win31DialogStyle	On|Off	Enables/disables Windows 3.1 style open and save dialogs.
SaveOnExit	On|Off	Enables/disables the auto saving of program options on exit.
PolyDots	On|Off	Enables/disables the removal of extra points left on screen by the polygon drawing function used to select groups of objects in the circuit window. Some display adapters seem to invert the connection point of lines incorrectly.
PolyBug	On|Off	Enables/disables the use of an internal PolyLine function. This is a workaround function that corrects a problem which may appear with some display adapters when a polygon is moved outside the left or upper edge of the screen.

Note that the last two parameters cannot be set from the *Tools|Options* dialog box. If you experience problems during polygon drawing operations, try to set one or both parameters to On using a text editor. Parameters accepting the boolean values: On or Off. True, False, 1 or 0 are also accepted.

Default settings:

```
AutoSave=Off
AutoSaveInterval=5
CreateBackupFile=Off
SaveWindowSizePos=On
SaveWindowState=Off
SaveToolbarState=On
SaveStatusBarState=On
SaveCommentLineState=Off
Win31DialogStyle=Off
SaveOnExit=Off
PolyDots=Off
PolyBug=Off
```

[Preferences]

Parameter	Range	Description
UndoBuffers	1-100	The number of undo/redo buffers to allocate for each circuit window.
BackgroundColor		The background color of circuit windows. You can specify one of the system color identifiers or a numeric value to set the red, green and blue color intensity.
TextEditor		Full path of text editor program to use for ATP file editing. If this parameter is an empty string (default), the built-in text editor is used.

Default settings:

```
UndoBuffers=10
BackgroundColor=Window
TextEditor=
```

[Directories]

Parameter	Range	Description
Circuits		The directory where you store your circuit files (.cir).
ATP		Specifies the directory in which ATP files are created (.atp).
SupportFiles		The container of standard component support files (.sup).
UserSpecified		Directory containing support (.sup), library (.lib) and punch (.pch) files for user specified components.
Tacs		The container of standard TACS component support files (.sup).
Models		Directory containing support (.sup) and model (.mod) files for MODELS components.

By default the different directories or folders are expected to be located in the ATPDraw installation directory.

Default settings:

```
Circuits=C:\ATPDRAW\CIR\
ATP=C:\ATPDRAW\ATP\
SupportFiles=C:\ATPDRAW\SUP\
UserSpecified=C:\ATPDRAW\USP\
Tacs=C:\ATPDRAW\TAC\
Models=C:\ATPDRAW\MOD\
```

[View Options]

Parameter	Range	Description
NodeNames		Enables/disables the visibility of node names.
Labels		Enables/disables the visibility of component labels.
Components		Enables/disables the visibility of standard and user specified components.
Models		Enables/disables the visibility of MODELS components.
Tacs		Enables/disables the visibility of standard TACS components.
Connections		Enables/disables the visibility of connection lines.
Relations		Enables/disables the visibility of relation lines.

NodeDots	Enables/disables the visibility of node dots.
DragIcon	Enables/disables complete icon drawing during single component move operations.
NoDataWarning	Enables/disables the visible warning (red color) of components and nodes not opened and given meaningful data.

The parameters accept boolean values only (On, Off, True, False, 1 or 0).

Default settings:

```
NodeNames=Off
Labels=On
Components=On
Models=On
Tacs=On
Connections=On
Relations=On
NodeDots=On
DragIcon=On
NoDataWarning=On
```

[ATP Settings]

Parameter	Type	Description
DeltaT	Real	Time step of simulation in seconds.
Tmax	Real	End time of simulation in seconds.
Xopt	Real	Inductances in [mH] if zero; otherwise, inductances in [Ohm] with Xopt as frequency.
Copt	Real	Capacitances in [mF] if zero; otherwise, capacitances in [Ohm] with Copt as frequency.
SysFreq	Real	System frequency in Hz.
IOut	Integer	Frequency of LUNIT6 output within the time-step loop. For example, a value of 3 means that every third time step will be printed.
IPlot	Integer	Frequency of saving solution points to the PL4 output file. For example, a value of 2 means that every second time step will be written to the PL4 file.
IDouble	Integer	If 1, table of connectivity written in the LUNIT6 output file. If 0 (zero), no such table written.
KssOut	Integer	Controls steady state printout to the LUNIT6 output file. Possible values are: 0: No printout. 1: Print complete steady state solution. Branch flows, switch flows and source injection. 2: Print switch flows and source injection. 3: Print switch flows, source injection and branch flows requested in column 80 punches.
MaxOut	Integer	If 1, extrema printed at the end of the LUNIT6 ouput file. If 0, no such print.

IPun	Integer	Flag for requesting an additional card for controlling the IOUT frequency. If IPUN equals -1, an additional card follows (not implemented). If IPUN equals 0 (zero), no such card follows.
MemSave	Integer	Controls the dumping of EMTP memory to disk at the end of simulation if START AGAIN request is specified. A value of 1 indicates memory saving (START AGAIN). 0 (zero) implies no memory dumping.
ICat	Integer	Controls saving of raw plot data points that is written to the I/O channel LUNIT4. Possible values are: 0: No saving. 1: Save points, but ignore any batch-mode plot cards present. 2: Save points and handle batch-mode plot cards.
Nenerg	Integer	Number of simulations. A value of 0 (zero) means single, deterministic simulation; otherwise, statistic switch study (NENERG > 0) or systematic switch study (NENERG < 0).
ISW	Integer	If 1, printout of all variable switch closing/opening time to LUNIT6. If 0 (zero), no printout.
ITEST	Integer	Extra random delay using DEGMIN, DEGMAX and STATFR in STARTUP. Possible values are: 0: Extra random delay for all switches. 1: No random delay. 2: Extra random time delay added to all closing switches. 3: Extra random time delay added to all opening switches.
IDIST	Integer	Select probability distribution of switch. A value of 0 (zero) means Gaussian distribution and a value of 1 uniform distribution.
IMAX	Integer	If 1, printout of extrema to LUNIT6 for every energization. If 0 (zero), no such printout.
IDICE	Integer	Use of standard random generator. A value of 0 (zero) implies computer-dependent random generator and a value of 1 standard random generator.
KSTOUT	Integer	Extra printed (LUNIT6) output for each energization. Output of time-step loop and variable extrema (MAXOUT>0). If 0 (zero), extra printed output. If -1, no such output.
NSEED	Integer	Repeatable MonteCarlo simulations. Possible values are: 0: Every simulation on the same data case will be different. 1: Same result each time the data case is run on the same computer.
HighResolution	Boolean	Usage of $Vintage 1 (if possible).
SortByCard	Boolean	Data file written with BRANCH cards first, followed by SWITCH cards and the SOURCE cards.
SortByGroup	Boolean	The group number given to each object determines the sequence of cards. The lowest group number comes first.
SortByXpos	Boolean	The leftmost object is written first.

AutoPath	Boolean	Library files are supposed to be located in the USP folder and have the extension .LIB. Each library file specification is verified to meet these requirements. If the path of a library file specifies a different folder or the extension is not .LIB, an error dialog is displayed during ATP file generation, enabling you to correct the erroneous specification by stripping off path and extension, continue the operation using an unresolvable ATP include reference, or cancel the entire ATP file generating process.
BatchJob*x*.name	Text	Name of the user specified batch job. Number *x* specifies the location of the job in the ATP menu. Max. 10 are supported.
BatchJobx.filename	Text	Name of the batch (or .EXE) file executed by ATPDraw when the job name is selected in the ATP menu.
BatchJobx.parameter	Integer	Specifies which file is sent as parameter, before executing the batch job: 0: No file name is sent. 1: File dialog opens where the file name is selected. 2: Name of the current ATP file name is sent. 3: Name of the current PL4 file name is sent.

Both standard decimal notation and E-format notation can be used when specifying real type values. Legal boolean values include On, Off, True, False, 1 and 0.

Default settings :

```
DeltaT=1.0E-0006
Tmax=1.0E-0003
Xopt=0
Copt=0
SysFreq=50
IOut=500
IPlot=1
IDouble=1
KssOut=1
MaxOut=1
IPun=0
MemSave=0
ICat=1
Nenerg=0
ISW=1
ITEST=1
IDIST=0
IMAX=0
IDICE=1
KSTOUT=-1
NSEED=0
HighResolution=Off
SortByCard=On
SortByGroup=Off
SortByXpos=Off
AutoPath=Off
No BatchJob is specified
```

[1024x768]

The initialization file may contain one or more screen resolution sections to record the ATPDraw main window size and position. Such sections are typically named [1024x768], [1280x1024], and so on.

Parameter	Description
Left	Window left edge position.
Top	Window upper edge position.
Width	Window width.
Height	Window height

The default [1024x768] resolution section looks something like this:

```
Left=64
Top=48
Width=896
Height=672
```

[Objects]

Parameter	Description
Toolbar	Shows or hides the main window toolbar at program startup.
StatusBar	Shows or hides the main window status bar at program startup.
CommentLine	Shows or hides the comment line of circuit windows when opened.
WindowState	Specifies ATPDraw startup state: Normal, Maximized or Minimized.

All parameters accept the boolean type values On, Off, True, False, 1 and 0.

Default settings:

```
Toolbar=On
StatusBar=On
CommentLine=Off
WindowState=Normal
```

5. Advanced Manual . . .

. . . .

ATPDraw™
for Windows
1.0

This chapter contains several examples of how to use ATPDraw on real engineering problems. You will not be shown how to create these circuits, but the circuit files `EXA_*.CIR` are part of the ATPDraw distribution kit. To load these example circuits into ATPDraw, use the *File* | *Open* command (or *Ctrl* + *O*) and select the file name in the *Open Circuit* dialog box. The resulting `.ATP` files will be given at the end of each description.

The second part of the manual explains the possibility of creating new circuit objects in ATPDraw. This extension is possible due to the ATP facilities $INCLUDE, DATA BASE MODULARIZATION and MODELS.

5.1 Switching in 500 kV system *(Exa_3.cir)*

This example shows how to perform a switching analysis in a simplified network. The line model used is a Π-equivalent line calculated for 5000 Hz, by LINE CONSTANTS outside of ATPDraw. How to create a more accurate JMarti line model and how to use the ATP_LCC, the new interactive graphical preprocessor for LINE/CABLE CONSTANTS support, are shown in section 5.5 of this manual and in the Line/Cable Manual.

The example circuit is shown on Fig. 5.1/a, the equivalent ATPDraw circuit in Fig. 5.1/b.

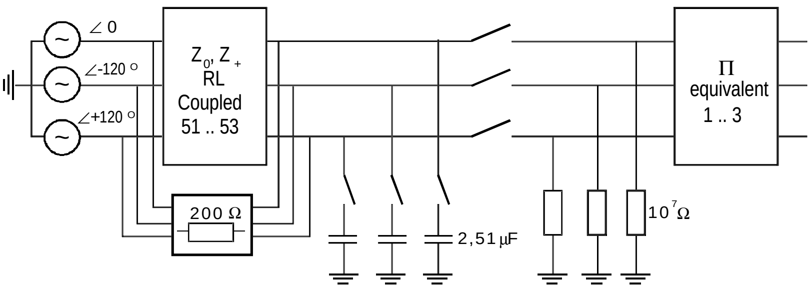


Fig. 5.1/a - Switching example circuit

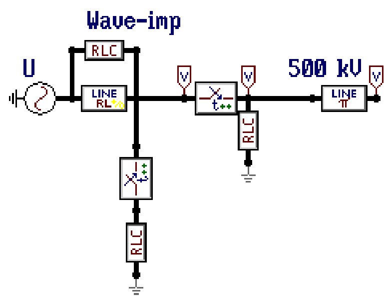


Fig. 5.1/b - Example circuit 3, Line switching (`EXA_3.CIR`)

Source:
The source is a 500 kV three phase AC source. The amplitude of the voltage source (U/I = 0) is

$$\text{Amp} = 500 \cdot \frac{\sqrt{2}}{\sqrt{3}}\ \text{kV}.$$

The input menu for the source is shown in Fig. 5.2:

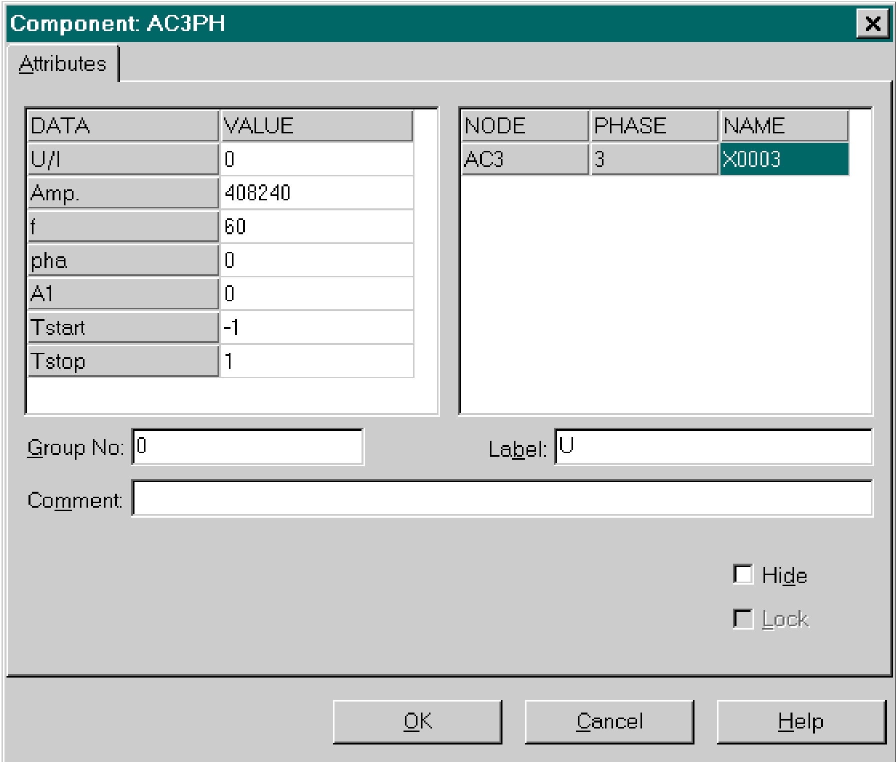


Fig. 5.2 Three phase source input window

Line switch:
The pi-equivalent line is connected to the source with a three-phase switch, having independent closing and opening times in all phases. The switches are initially open and close at:

Phase *A*: 33.33 ms, Phase *B*: 36.10 ms, Phase *C*: 38.80 ms

Capacitor bank:
The capacitor bank is 2.51 μF in all phases. The switches connecting the bank to the network is initially open and close at

Phase *A*: 133.33 ms, Phase *B*: 136.10 ms, Phase *C*: 138.80 ms

Pi-equivalent line:
The data for the Π-equivalent line are calculated by the LINE CONSTANTS supporting routine of ATP. The line is a 500 kV overhead line with 3 phase conductors and 2 ground wires, taken from benchmark `DCN3.DAT`. This calculation has resulted in the following line matrices for the 138 miles line (only lower-triangle part is given). Read the inductances in [mH], the resistances in [Ω] and the capacitances in [μF]:

R

851.06		
773.84	755.62	
818.41	773.84	851.06

L

317.37		
111.6	322.66	
101.83	111.6	313.37

C

2.3224		
-0.3637	2.3684	
-0.2726	-0.3637	2.3224

If you click on the Π-line object with the right mouse button drawn on the rightmost side of Fig. 5.1/b, an object input window appears, where the RLC data can be set, manually.

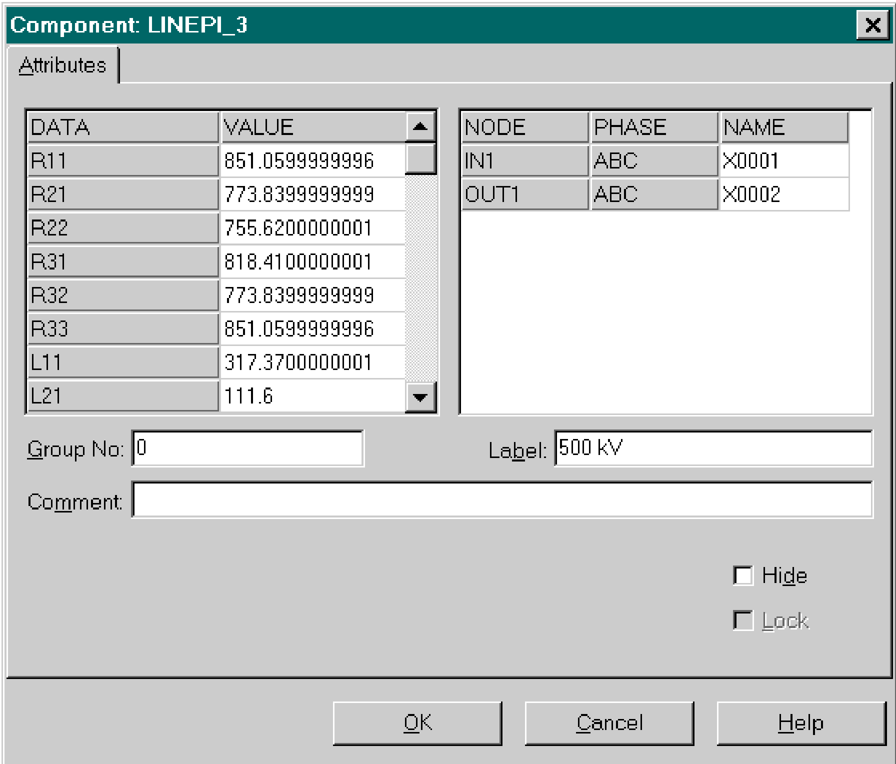


Fig. 5.3 - Three phase pi-equivalent input window

The ATP file created by ATPDraw is shown below:

```
BEGIN NEW DATA case
C ------------------------------------------------
C Generated by ATPDRAW  Mon, 4 May - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C ------------------------------------------------
$PREFIX,C:\ATPDRAW\USP\
$SUFFIX, .LIB
$DUMMY, XYZ000
C Miscellaneous Data Card ....
POWER FREQUENCY                      60.
  .00001      .2
     500       5       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
1 X0001AX0002A            851.06317.372.3224
2 X0001BX0002B            773.84 111.6-0.364755.62322.662.3684
3 X0001CX0002C            818.41101.83-0.273773.84 111.6-0.364851.06317.372.3224
51X0003AX0005A               .55        8.98
52X0003BX0005B              .711      11.857
53X0003CX0005C
  X0003AX0005A              200.                                               0
  X0003BX0005B              200.                                               0
  X0003CX0005C              200.                                               0
```

```
        X0010A                          2.51                                   0
        X0010B                          2.51                                   0
        X0010C                          2.51                                   0
        X0001A            1.00E7                                               0
        X0001B            1.00E7                                               0
        X0001C            1.00E7                                               0
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
  X0005AX0001A    .03333       10.                                             0
  X0005BX0001B     .0361       10.                                             0
  X0005CX0001C     .0388       10.                                             0
  X0010AX0005A    .13333       10.                                             0
  X0010BX0005B     .1361       10.                                             0
  X0010CX0005C     .1388       10.                                             0
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
14X0003A 0   408248.       60.                                    -1.        1.
14X0003B 0   408248.       60.     -120.                          -1.        1.
14X0003C 0   408248.       60.      120.                          -1.        1.
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
  X0001AX0001BX0001CX0002AX0002BX0002CX0005AX0005BX0005C
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA case
BLANK
```

5.2 TACS controlled induction machine *(Exa_4.cir)*

This example shows the usage of a universal machine type 3, manual initialization along with usage of TACS. Also the info arrows, whose purpose is to visualize TACS variables into FORTRAN statements are shown. The info arrows are selected under *TACS | Draw relation* in the component selection menu and they are handled graphically as normal connections. They do not affect the ATP file, however. The example is taken from exercise 46 in [2]. The ATPDraw constructed circuit is shown in Fig. 5.4/b:

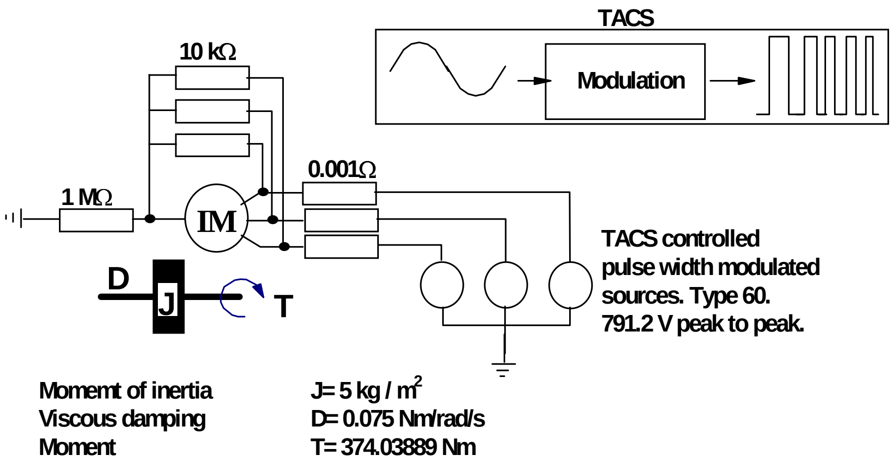


Fig. 5.4/a - Induction machine + TACS

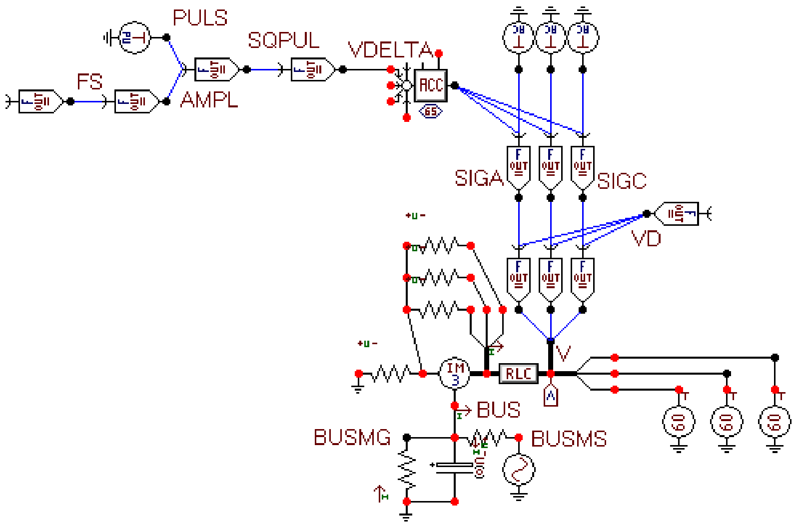


Fig. 5.4/b - ATPDraw scheme of the induction machine example (`EXA_4.CIR`)

The TACS part of the circuit controls three sources, creating a pulse width modulated armature voltage. The TACS objects are listed in the reference part of this manual.

The input window of the TACS object at the end of the TACS chain is shown in Fig. 5.5. This TACS object creates the armature voltage in phase *A* of the 3-phase node *V*.

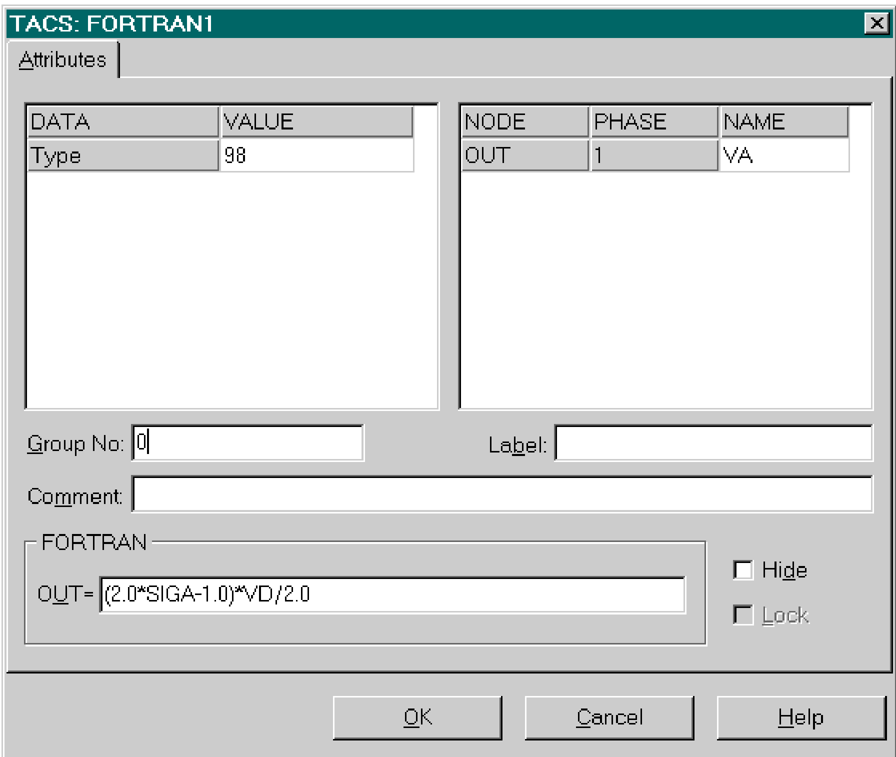


Fig. 5.5 - TACS Fortran input window

In the TACS statement the user must type in the expression(s). Only single phase TACS Fortran objects are supported. The two (blue) info arrows into this TACS object serve as visualization of the *SIGA* (from node *SIGA*) and *VD* signals.

The induction machine was given the data shown in Fig. 5.6:

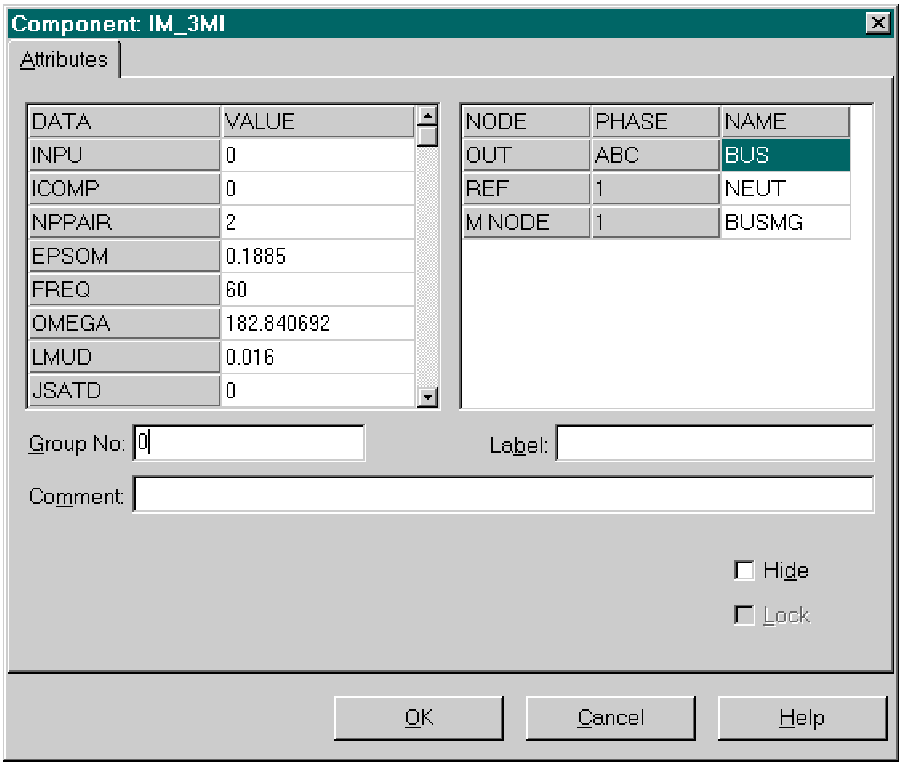


Fig. 5.6 - Induction machine input window

The numerical values in Fig. 5.6 must be specified by the user as in the case for all object input windows. The identity text in front of each attribute strictly follows the input variable in the ATP Rule Book [3].

The ATP file created by ATPDraw is shown below:

```
BEGIN NEW DATA case
C ------------------------------------------------
C Generated by ATPDRAW  Thu, 14 May - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C ------------------------------------------------
$BEGIN PL4 COMMENTS
C Induction motor supplied by a
C pulse width modulated source.
C Test example 1.
$END PL4 COMMENTS
$PREFIX,C:\ATPDRAW\USP\
$SUFFIX, .LIB
$DUMMY, XYZ000
C Miscellaneous Data Card ....
POWER FREQUENCY                     60.
  .00001      .1
     500       1       1       1       1       0       0       1       0
TACS HYBRID
/TACS
98FS      =1000
23PULS          2.      .001     .0005                           .000252
98AMPL    =4.0*FS
98SQPUL   =AMPL*(UNITY-PULS)
98VDELTA  =SQPUL*DELTAT
```

```
98VTRI  65        +VDELTA
14VCONTA         .95       60.      -90.
14VCONTB         .95       60.     -210.
14VCONTC         .95       60.       30.
98VB      =(2.0*SIGB-1.0)*VD/2.0
98VA      =(2.0*SIGA-1.0)*VD/2.0
98SIGA    =VCONTA .GT. VTRI
98VC      =(2.0*SIGC-1.0)*VD/2.0
98SIGB    =VCONTB.GT. VTRI
98SIGC    =VCONTC .GT. VTRI
98VD      =791.2
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
        NEUT              1.00E6                                               2
        BUSMG              13.33                                               1
  BUSMG                                 5.00E6                                 3
  BUSMG BUSMS             1.0E-6                                               1
  NEUT  BUSC              10000.                                               2
  NEUT  BUSB              10000.                                               2
  NEUT  BUSA              10000.                                               2
  BUSA  VA                  .001                                               1
  BUSB  VB                  .001                                               1
  BUSC  VC                  .001                                               1
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
14BUSMS -1-374.03889    .00001
60VC     0
60VB     0
60VA     0
19 UM
 0
BLANK
 3 1 1331BUSMG        2                                     .1885           60.
C Magnetization inductances
    182.840692          .016
    .785398163          .016
C Armature coils
                            BUSA  NEUT        173.55870000005
          .095         .0005BUSB  NEUT        180.54500000004
          .095         .0005BUSC  NEUT        1-154.103399999
C Rotor coils
          .075         .0004                  1169.6725000001
          .075         .0004                  1        19.285
BLANK
BLANK TACS
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
 2BUSMG  1.828407E+0002
 3BUSMG                       1.828407E+0002
  VA    VB    VC
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA case
BLANK
```

5.3 Usage of the Library and library reference objects *(Exa_5.cir)*

This example shows how ATPDraw can be used efficiently by a professional ATP user. The ATP expert simply creates the required ATP file for a power system (which is often already available) outside of ATPDraw, and then builds a simple ATPDraw case, where the power system is included with $Include and a limited number of additional components e.g. switches are added. Node names internally in the $Include file must be the same as those used in the ATPDraw data case (*F1 - F4*, left adjusted in this example!).

The data case can then be sent to a person whose knowledge about ATP is rather limited. This person can run the case, manipulate the switches and establish the ground fault current along a transmission line, eg. for relay setting purposes.

Any user specified objects can be used as a simple $Include library (*Send Param*. button off), but a predefined object is available under the *User Specified* field in the selection menu. Two other special objects are also available; the `LibRef_1` and `LibRef_3`. These objects are not written in the ATP file at all, and their purpose is only to visualize a connectivity in the $Include file.

An example, where these objects (Library (`LIB1.A-B`) and 3-phase reference (`LIBREF_3`)) have been used, is shown in Fig. 5.7.

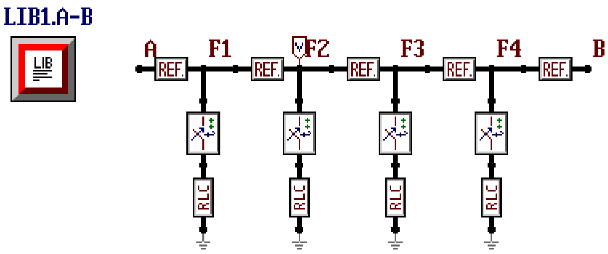


Fig. 5.7 - Usage of a library reference objects (`EXA_5.CIR`)

If you click the right mouse button on the Library object (identified by the LIB1.A-B label) you get the input window to set the element attributes, as shown in Fig. 5.8.

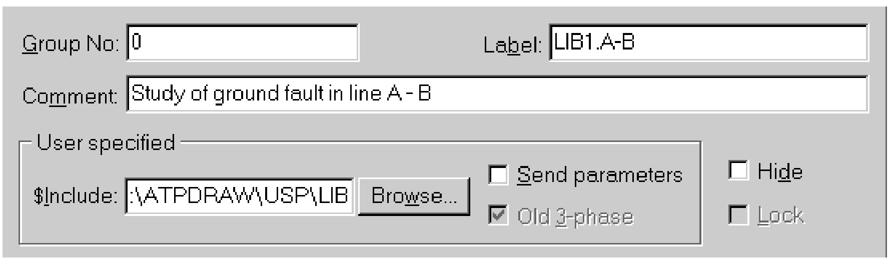


Fig. 5.8 - Library input window.

The most important menu field in this window is the *$Include* which gives the name and path of the file to be included. The *Browse* button allows you to select a file in the file Open dialog box. The *Send Parameters* check box must be off. *Label* and *Comment* are optional fields.

If you click with the right mouse button on one of the 3-phase reference objects, a similar dialog appears with the difference that it has node name fields, too.

The *Label* menu is the only input variable having a meaning in this window, along with the node names. The *$Include* field should be left empty.

The reference objects are not represented in the ATP data file, but serve only as visualization of connectivity.

The ATPDraw generated ATP file is shown below:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  Tue, 30 Jun - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
$DUMMY, XYZ000
C Miscellaneous Data Card ....
C  dT   >< Tmax >< Xopt >< Copt >
 .000001    .001
     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
        X0011A               10.                                               0
        X0011B               10.                                               0
        X0011C               10.                                               0
        X0013A               10.                                               0
        X0013B               10.                                               0
        X0013C               10.                                               0
        X0015A               10.                                               0
        X0015B               10.                                               0
        X0015C               10.                                               0
        X0017A               10.                                               0
        X0017B               10.                                               0
        X0017C               10.                                               0
C Study of ground fault in line A - B
$INCLUDE, C:\ATPDRAW\USP\LIB.LIB
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
  X0011AF1A         .033       2.                                              0
  X0011BF1B           2.       2.                                              0
  X0011CF1C           2.       2.                                              0
  X0013AF2A           2.       2.                                              0
  X0013BF2B           2.       2.                                              0
  X0013CF2C           2.       2.                                              0
  X0015AF3A           2.       2.                                              0
  X0015BF3B           2.       2.                                              0
  X0015CF3C           2.       2.                                              0
  X0017AF4A           2.       2.                                              0
  X0017BF4B           2.       2.                                              0
  X0017CF4C           2.       2.                                              0
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
  F2A   F2B   F2C
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

Auto path option was assumed to be unselected on the *ATP | Settings | File* page, so the full path of the library file is written to the .ATP file.

Otherwise, if this option is selected, the library files are supposed to be located in the `\USP` folder and have the extension `.LIB`. Each library file specification is verified to meet these requirements.

If the path of a library file specifies a different folder or the extension is not `.LIB`, an error dialog is displayed during the ATP file generation process, enabling you to correct the erroneous specification by stripping off path and extension, continue the operation using an unresolvable ATP include reference, or cancel the entire ATP file generating process. Fig. 5.9 shows this error dialog, as an example.

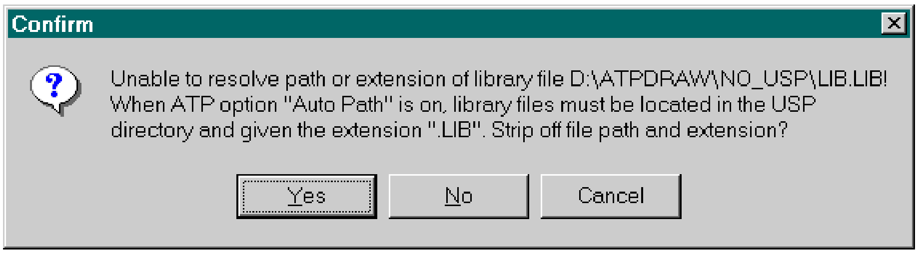


Fig. 5.9 - Error dialog when unresolvable .LIB object specification is detected.

5.4 Modeling an HVDC station

This example shows how to create a 6-pulse controlled thyristor rectifier bridge and make it available in ATPDraw as a single object. How to create the required Data Base Module (DBM) file will be explained along with the necessary actions inside ATPDraw. Finally a practical example (*Exa_6.cir*), where the new object is used to build up a 12-pulse HVDC station including transformers will be shown.

5.4.1 Creating a Data Base Module file

The first step is to create a DBM file which is an ATP file for the specific circuit along with a header explaining variables in the DBM file. The ATP Rule Book [3] chapter XIX-F explains in detail how to create such a file. The DBM punch file can actually be considered as an external procedure to the final ATP file. The process of creating a DBM file is certainly the most difficult part of adding a new object to ATPDraw. Below is shown a DBM file describing a 6-pulse thyristor rectifier bridge (based on exercise 54 in [2]):

```
BEGIN NEW DATA CASE --NOSORT--
DATA BASE MODULE
$ERASE
ARG,U____,POS___,NEG___,REFPOS,REFNEG,ANGLE_,Rsnub_,Csnub_
NUM,ANGLE_,Rsnub_,Csnub_
DUM,PULS1_,PULS2_,PULS3_,PULS4_,PULS5_,PULS6_,MID1__,MID2__,MID3__
DUM,GATE1_,GATE2_,GATE3_,GATE4_,GATE5_,GATE6_,VAC___,RAMP1_,COMP1_
DUM,DCMP1_,DLY60D
/TACS
11DLY60D  .002777778
90REFPOS
90REFNEG
98VAC___  =REFPOS-REFNEG
98RAMP1_58+UNITY                                  120.00   0.0   1.0VAC___
98COMP1_  =(RAMP1_-ANGLE_/180) .AND. UNITY
98DCMP1_54+COMP1_                                       5.0E-3
98PULS1_  = .NOT. DCMP1_ .AND. COMP1_
```

```
98PULS2_54+PULS1_                                                         DLY60D
98PULS3_54+PULS2_                                                         DLY60D
98PULS4_54+PULS3_                                                         DLY60D
98PULS5_54+PULS4_                                                         DLY60D
98PULS6_54+PULS5_                                                         DLY60D
98GATE1_  = PULS1_ .OR. PULS2_
98GATE2_  = PULS2_ .OR. PULS3_
98GATE3_  = PULS3_ .OR. PULS4_
98GATE4_  = PULS4_ .OR. PULS5_
98GATE5_  = PULS5_ .OR. PULS6_
98GATE6_  = PULS6_ .OR. PULS1_
/BRANCH
$VINTAGE,0
  POS___U____A            Rsnub_      Csnub_
  POS___U____BPOS___U____A
  POS___U____CPOS___U____A
  U____ANEG___POS___U____A
  U____BNEG___POS___U____A
  U____CNEG___POS___U____A
/SWITCH
11U____APOS___                                                        GATE1_
11U____BPOS___                                                        GATE3_
11U____CPOS___                                                        GATE5_
11NEG___U____A                                                        GATE4_
11NEG___U____B                                                        GATE6_
11NEG___U____C                                                        GATE2_
BEGIN NEW DATA CASE
C
$PUNCH
BEGIN NEW DATA CASE
BLANK
```

The user must first decide which data and node names to send to the module from the outside, and must set the local variables. A thyristor bridge needs a 3-phase AC input node and two DC output nodes. Also two external voltages whose difference is used as a zero crossing detector can be sent to the DBM file (alternatively the DBM module file could have detected its own AC input). The firing angle is required and the values of the snubber circuits are also passed to the DBM file.

The header of the DBM file shown above starts with the special request card `DATA BASE MODULE` followed by a variable list. The first card is the `ARG` card giving all the input variables to the DBM file. These variables are used in the final ATP files as the parameters of the $INCLUDE expression. The arguments to this procedure are:

`U_____` :	The AC 3-phase node
`POS___`:	The positive DC node
`NEG___`:	The negative DC node
`REFPOS`:	Positive reference node.
`REFNEG`:	Negative reference node.
	`REFPOS-REFNEG` is used as a zero crossing detector.
`ANGLE_`:	The firing angle of the thyristors.
`Rsnub_`:	The resistance in the snubber circuits.
`Csnub_`:	The capacitance in the snubber circuits.

Note the importance of the number of characters used for each parameter. The `U_____` parameter has only 5 characters, because it is a 3-phase node and the extensions *A*, *B* and *C* are added inside the DBM file. Underscore '_ ' has been used to force the variables to occupy all the corresponding positions in the data cards.

The NUM card tells what arguments are numerical. DUM card lists all the dummy or local variables. These are typically internal node names. ATP gives them a unique node name and thus let you use the same DBM several times in a data case avoiding node name conflicts.

The rest of the DBM file describes the rectifier bridge in a normal ATP way.

Sorting by cards must be used in a special way: i.e. /TACS, /BRANCH, /SWITCH etc. cards are required, but no BLANK TACS, BLANK BRANCH etc. indicators are needed.

Running this DBM module file through ATP will produce a .PCH punch file shown below (the comment cards are removed):

```
KARD  3  4  5  6  6  6  7  7  8  8  8  9  9 10 10 10 11 11 11 12 12 12 13 13 13
     14 14 14 15 15 15 16 16 16 17 17 17 18 18 18 19 19 19 20 20 20 21 21 21 24
     24 24 24 25 25 25 25 26 26 26 26 27 27 27 27 28 28 28 28 29 29 29 29 31 31
     31 32 32 32 33 33 33 34 34 34 35 35 35 36 36 36
KARG-20  4  5  4  5-16-16-17  6-17-18-18-19 -1-18-19 -1 -2-20 -2 -3-20 -3 -4-20
     -4 -5-20 -5 -6-20 -1 -2-10 -2 -3-11 -3 -4-12 -4 -5-13 -5 -6-14 -1 -6-15  1
      2  7  8  1  1  2  2  1  1  2  2  1  1  2  3  1  1  2  3  1  1  2  3  1  2
    -10  1  2-12  1  2-14  1  3-13  1  3-15  1  3-11
KBEG  3  3  3 12 19  3 69  3 20 13  3 12  3  3 32 19 12  3 69 12  3 69 12  3 69
     12  3 69 12  3 69 13 25  3 13 25  3 13 25  3 13 25  3 13 25  3 25 13  3  9
      3 27 39  9 21  3 15  9 21  3 15  3 21 15  9  3 21 15  9  3 21 15  9  3  9
     65  3  9 65  3  9 65  9  3 65  9  3 65  9  3 65
KEND  8  8  8 17 24  8 74  8 25 18  8 17  8  8 37 24 17  8 74 17  8 74 17  8 74
     17  8 74 17  8 74 18 30  8 18 30  8 18 30  8 18 30  8 18 30  8 30 18  8 13
      8 32 44 13 25  8 20 13 25  8 20  7 25 20 14  7 25 20 14  7 25 20 14  7 14
     70  7 14 70  7 14 70 13  8 70 13  8 70 13  8 70
KTEX  1  1  1  1  1  1  1  1  0  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
      1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
      1  0  0  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
      1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
$ERASE
/TACS
11DLY60D  .002777778
90REFPOS
90REFNEG
98VAC___  =REFPOS-REFNEG
98RAMP1_58+UNITY                                  120.00    0.0    1.0VAC___
98COMP1_  =(RAMP1_-ANGLE_/180) .AND. UNITY
98DCMP1_54+COMP1_                                       5.0E-3
98PULS1_  = .NOT. DCMP1_ .AND. COMP1_
98PULS2_54+PULS1_                                                   DLY60D
98PULS3_54+PULS2_                                                   DLY60D
98PULS4_54+PULS3_                                                   DLY60D
98PULS5_54+PULS4_                                                   DLY60D
98PULS6_54+PULS5_                                                   DLY60D
98GATE1_  = PULS1_ .OR. PULS2_
98GATE2_  = PULS2_ .OR. PULS3_
98GATE3_  = PULS3_ .OR. PULS4_
98GATE4_  = PULS4_ .OR. PULS5_
98GATE5_  = PULS5_ .OR. PULS6_
98GATE6_  = PULS6_ .OR. PULS1_
/BRANCH
$VINTAGE,0
  POS___U____A            Rsnub_      Csnub_
  POS___U____BPOS___U____A
  POS___U____CPOS___U____A
  U____ANEG___POS___U____A
  U____BNEG___POS___U____A
  U____CNEG___POS___U____A
/SWITCH
11U____APOS___                                                  GATE1_
11U____BPOS___                                                  GATE3_
11U____CPOS___                                                  GATE5_
```

```
11NEG___U____A                                                 GATE4_
11NEG___U____B                                                 GATE6_
11NEG___U____C                                                 GATE2_
$EOF   User-supplied header cards follow.        01-Dec-95  20.11.59
ARG,U____,POS___,NEG___,REFPOS,REFNEG,ANGLE_,Rsnub_,Csnub_
NUM,ANGLE_,Rsnub_,Csnub_
DUM,PULS1_,PULS2_,PULS3_,PULS4_,PULS5_,PULS6_,MID1__,MID2__,MID3__
DUM,GATE1_,GATE2_,GATE3_,GATE4_,GATE5_,GATE6_,VAC___,RAMP1_,COMP1_
DUM,DCMP1_,DLY60D
```

This file is very similar to your DBM file, but with a different header and with the DBM file header given at the bottom instead. This file will be $Included in the ATP file by ATPDraw. You must give this file a name with extension `.LIB` and store it in the `\USP` directory. The name `HVDC_6.LIB` is used here as an example.

5.4.2 Creating a new User Specified ATPDraw object

When the punch file from the DBM file has been created, the next step is to create a *New User Specified* object. So start up ATPDraw and enter the *Objects* field in the main menu. The process of creating a new object consists of two steps: a) Create parameter support and b) Create icon.

5.4.2.1 Creating parameter support

First select the *New User Specified* field in the popup menu. A window shown in Fig. 5.10 appears, where you must specify the number of data and nodes of the new object. The required size of the ATPDraw object is found in the header of the DBM file. The number of arguments on the `NUM` card(s) tells you the *Number of data*, which is 3 in this example. The number of arguments on the `ARG` card(s) tells you the total number of arguments which in this case is 8. The *Number of nodes* thus in this example, is set to 5.

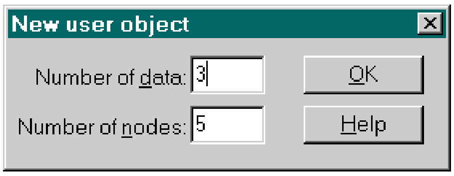


Fig. 5.10 - Size of the new object

When the user has determined the size of the new object based on the Data Base Module (DBM) file header and clicks *OK*, a notebook-style dialog box as the one shown in Fig. 4.20 appears. On the Node and Data tabs you must specify the control parameters for the new object.

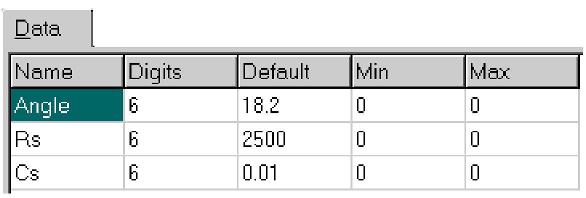


Fig. 5.11 - Data window of the new object

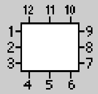
In this window you can specify the names of the data parameters, number of digits (must be <= the space used in the DBM file which is 6 in this case) a default value, and the Min/Max values.

The names of the data do not have to be equal to the ones used in the DBM punch file, but the sequence of the data must be the same as used in the ARG and NUM card(s).

After you have specified the window contents shown above, you just click the *Node* tab and specify the node control parameters as shown in Fig. 5.12.

Nodes

Name	Phase	Position
AC	3	2
POS	1	9
NEG	1	7
Ua	1	12
Uc	1	11



In this window you must specify the Id names of the nodes in the *Name* field, the number of phases (1 or 3) in the *Phase* field and the position on the icon border (1-12) in the *Position* field. Codes for the available node positions are shown in the icon at right in Fig. 5.12.

Fig. 5.12 - Node window of the new object

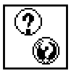
The names of the nodes do not have to be the same as the names used in the DBM punch file, but the sequence of the nodes must be the same as used in the ARG card.

As default, ATPDraw writes all three names of 3-phase nodes in the $Include statement, but optionally only the first 5 basic characters of the 3-phase nodes can be specified. This option requires the *Old 3-phase* option be checked in the *User specified* field of the component window. If this option is selected, ATPDraw writes only five character long node names in the $Include statement and let the extensions *A*, *B* and *C* be added inside the DBM library file. At the end of section 5.4.3 more explanation about the usage of this ATPDraw feature is given.

It must be noted, however that the *Old 3-phase* option may result in conflicts with transposition objects and always requires full five character long node names. To avoid conflicts, it is suggested to use all three names of 3-phase nodes as input when you create new Data Base Module files.

5.4.2.2 Creating icon and help file for the object

Each user specified component must have an icon which represents the object on the screen and an optional on-line help which describes the meaning of the parameters. These properties can be edited using the built in *Help* and *Icon Editors.*

The  button at the right hand side of the *New User Specified* dialog box is used to call the help editor. All the functions and help editor menus are described in the Reference part of this manual, and are not repeated here. Fig. 5.13 shows the help file associated with the user specified 6-phase rectifier bridge. The *Done* button is used to return when you have finished editing the help.

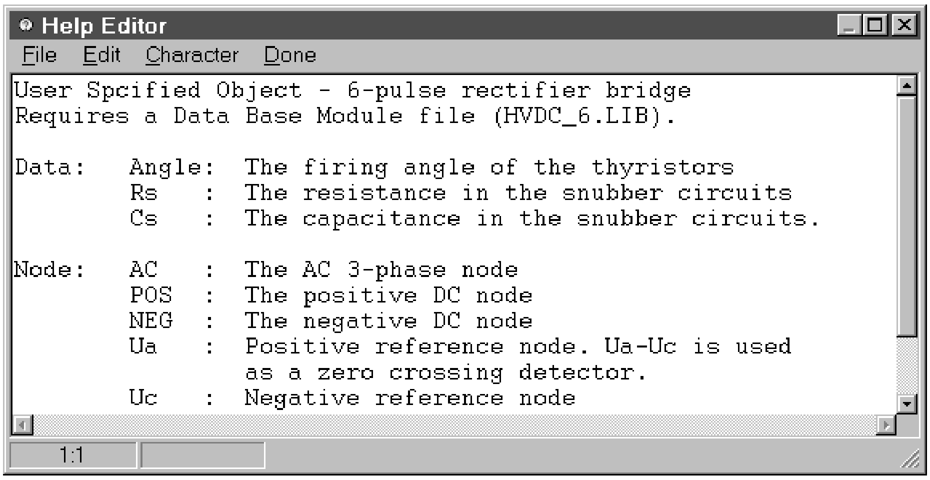
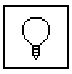


Fig. 5.13 - Help editor

The  icon at the right hand side of the *New User Specified* dialog box is used to call the icon editor. All the functions and menus of the editor are described in the Reference part, so this information is not repeated here.

In the icon editor window you can use your own fantasy. Select pen colors from the palette at the bottom. Click in the grid field with the left mouse button to place the color and the right mouse button to erase. The red lines in the grid indicate the possible node positions on the icon border. Lines should be drawn from the symbol in the middle and out to the node positions you have chosen in the window in Fig. 5.12. The completed icon of the 6-pulse rectifier bridge is shown in Fig. 5.14. Click on the *Done* button when you have finished.

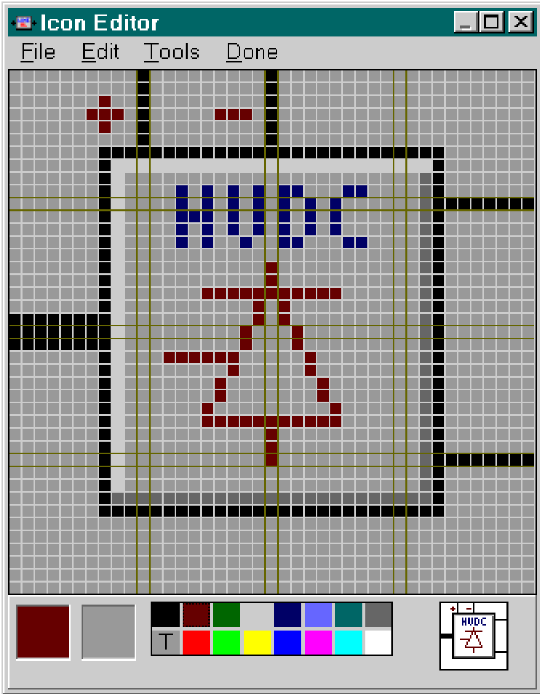


Fig. 5.14 - The icon associated with the new object

Finally, after clicking the *Save* or *Save As* buttons, the new support file will be saved to disk. The file name can be specified in a standard *Save...* dialog box. Support files for user specified components are normally located in the `\USP` folder and their extension is `.SUP`. The file name may be different from the DBM file name. (`C:\atpdraw\usp\HVDC_6.SUP` is used in this example).

You can reload the support files of the user specified objects whenever you like, using the *Edit User Specified* item in the *Objects* main menu.

Now a completely new object in ATPDraw has been created. The object is found under *User Specified | Files* in the component selection menu. If you select `HVDC_6.SUP` in the appearing file window, the new object is drawn in the circuit window and can be used and edited like any other objects.

5.4.3 Example circuit. 12-pulse HVDC station *(Exa_6.cir)*

The 6-pulse rectifier bridge can be used when a 12-pulse HVDC station is to be simulated. An example of such an application is shown in Fig. 5.15. The example is based on exercise 54 in [2].

The HVDC station is supplied by a 3-phase AC source via two transformers. In this example the new object `HVDC_6` is used two times in series. Fig. 5.16 shows the data input dialog box of the new object. The external library file (`HVDC_6.LIB`) created using DATA BASE MODULARIZATION feature of ATP is $Include-d in this example.

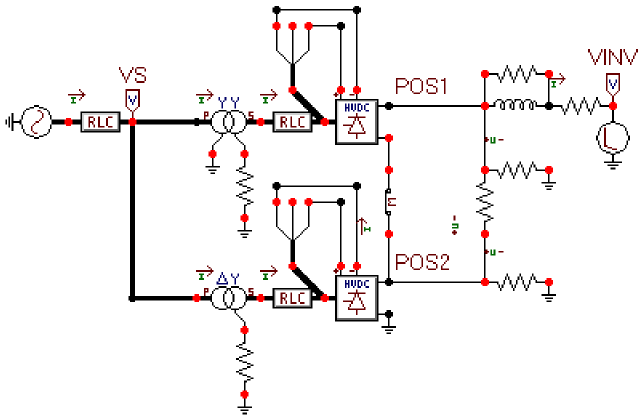


Fig. 5.15 - 12-pulse HVDC station (`EXA_6.CIR`)

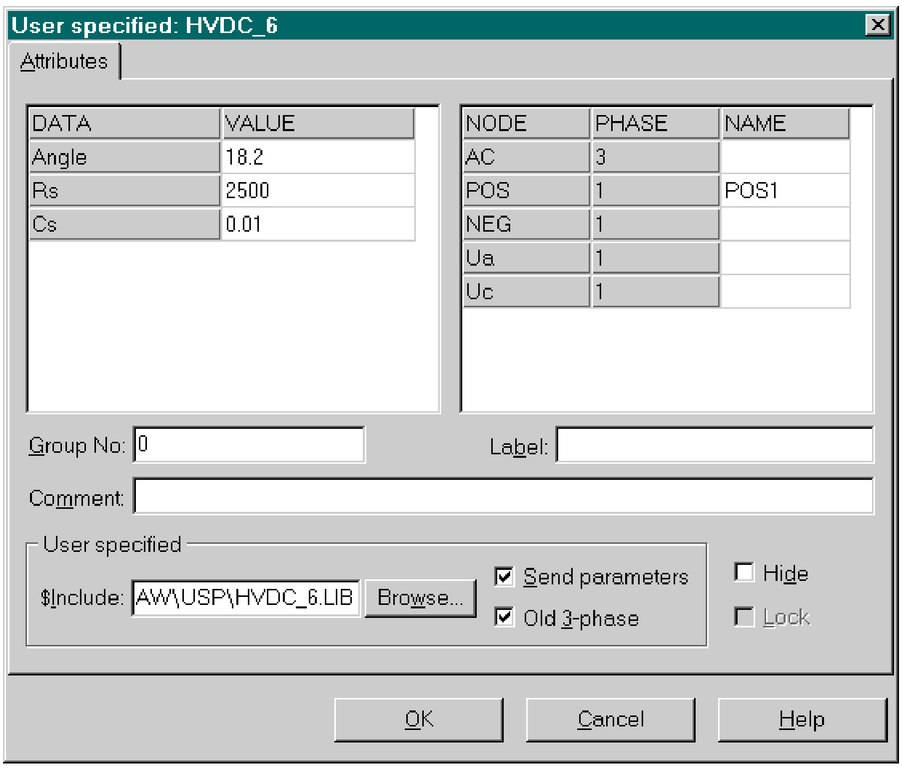


Fig. 5.16 - Input window of the users specified 6-pulse rectifier bridge

The ATP file created by ATPDraw is shown here:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  Tue, 30 Jun - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
$PREFIX,C:\ATPDRAW\USP\
$SUFFIX, .LIB
$DUMMY, XYZ000
C Miscellaneous Data Card ....
POWER FREQUENCY                     60.
C  dT   >< Tmax >< Xopt >< Copt >
  .00002    .033
     500       1       1       1       1       0       0       1       0
TACS HYBRID
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
  TRANSFORMER                         TX00011.0E12                             1
            9999
 1VA                       .1 98.57    1.
 2VS1MA XX0017           .022 21.59  .468
  TRANSFORMER TX0001                  TX0002
 1VB
 2VS1MB XX0017
  TRANSFORMER TX0001                  TX0003
 1VC
 2VS1MC XX0017
  VS1MA VS1XXA           .0001                                                 1
  VS1MB VS1XXB           .0001                                                 1
  VS1MC VS1XXC           .0001                                                 1
        XX0017          1.00E7                                                 0
  VSA   VA               .0001                                                 1
  VSB   VB               .0001                                                 1
  VSC   VC               .0001                                                 1
  TRANSFORMER                         TX00041.0E12                             1
            9999
 1VA    VC                 .3 295.7    1.
 2VS2MA XX0032           .022 21.59 .2702
  TRANSFORMER TX0004                  TX0005
 1VB    VA
 2VS2MB XX0032
  TRANSFORMER TX0004                  TX0006
 1VC    VB
 2VS2MC XX0032
  VS2MA VS2XXA           .0001                                                 1
  VS2MB VS2XXB           .0001                                                 1
  VS2MC VS2XXC           .0001                                                 1
        XX0032          1.00E7                                                 0
  POS1  VLINE                       100.                                       0
  POS1  VLINE            8000.                                                 0
  VLINE VINV             15.35                                                 1
  POS1                  1.00E7                                                 2
  POS2                  1.00E7                                                 2
  POS2  POS1            1.00E7                                                 2
C User specified object: C:\ATPDRAW\USP\HVDC_6.SUP
C  Angle = 1.8E+0001
C     Rs = 2.5E+0003
C     Cs = 1.0E-0002
C     AC = VS1XXA
C    POS = POS1
C    NEG = XX0011
C     Ua = VS1XXA
C     Uc = VS1XXC
```

```
$INCLUDE, HVDC_6, VS1XX, POS1##, XX0011, VS1XXA, VS1XXC,  18.2, 2500.,   .01
C User specified object: C:\ATPDRAW\USP\HVDC_6.SUP
C  Angle = 1.8E+0001
C     Rs = 2.5E+0003
C     Cs = 1.0E-0002
C     AC = VS2XXA
C    POS = POS2
C    NEG =
C     Ua = VS2XXA
C     Uc = VS2XXC
$INCLUDE, HVDC_6, VS2XX, POS2##, ######, VS2XXA, VS2XXC,  18.2, 2500.,   .01
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
  POS2  XX0011                                        MEASURING                1
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
14VSA     0   187794.       60.      -60.                                  -1.        1.
14VSB     0   187794.       60.     -180.                                  -1.        1.
14VSC     0   187794.       60.       60.                                  -1.        1.
12VINV    0   229660.                 .01
BLANK TACS
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
  VA     VB     VC     VINV
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

The part of the file marked bold (**TACS HYBRID**, **BLANK TACS** and **POWER FREQUENCY**) is added to the ATP file only if the *TACS* and *Power Frequency* check boxes in the *File format* page of the *ATP | Settings* menu are selected. Leaving the *TACS* box unchecked, ATPDraw would not know that TACS is present inside the $Include file, so the user must have to add it manually.

It is also important to note that a full 5 characters long node name (e.g. **VS1XX**) must be used for the 3-phase node of the HVDC_6 object, since ATPDraw adds *A*, *B* or *C* to the end of the user supplied node name. This restriction is required by the structure of the `HVDC_6.LIB` file, which was prepared supposing that *A*, *B* or *C* phase sequence identifiers are located in the 6th character position of the node names. Furthermore, the *Old 3-phase* indicator must be selected (see Fig. 5.16) to pass the first parameter as a 3-phase node rather than 3 single phase ones, to be conform with the structure of the DBM file. Consequently, only the first example below is suitable to $Include the `HVDC_6.LIB`, into the ATP file properly, where the node name has 5 characters and the *Old 3-phase* is on. The other two examples are equally wrong and will generate an error message or will produce incorrect results at the simulation, because in the 2^{nd} example only 3 characters long node name has been specified, or in the 3^{rd} one the *Old 3-phase* parameter was not selected.

```
1) $INCLUDE, HVDC_6, VS1XX, POS1##, XX0011, VS1XXA, VS1XXC . . .
2) $INCLUDE, HVDC_6, VS1##, POS1##, XX0011, VS1A##, VS1C## . . .
3) $INCLUDE, HVDC_6, VS1XXA, VS1XXB, VS1XXC, POS1##, XX0011, VS1XXA, VS1XXC
```

In Fig. 5.17 four curves from the simulation are shown. The results are the same as in [2].

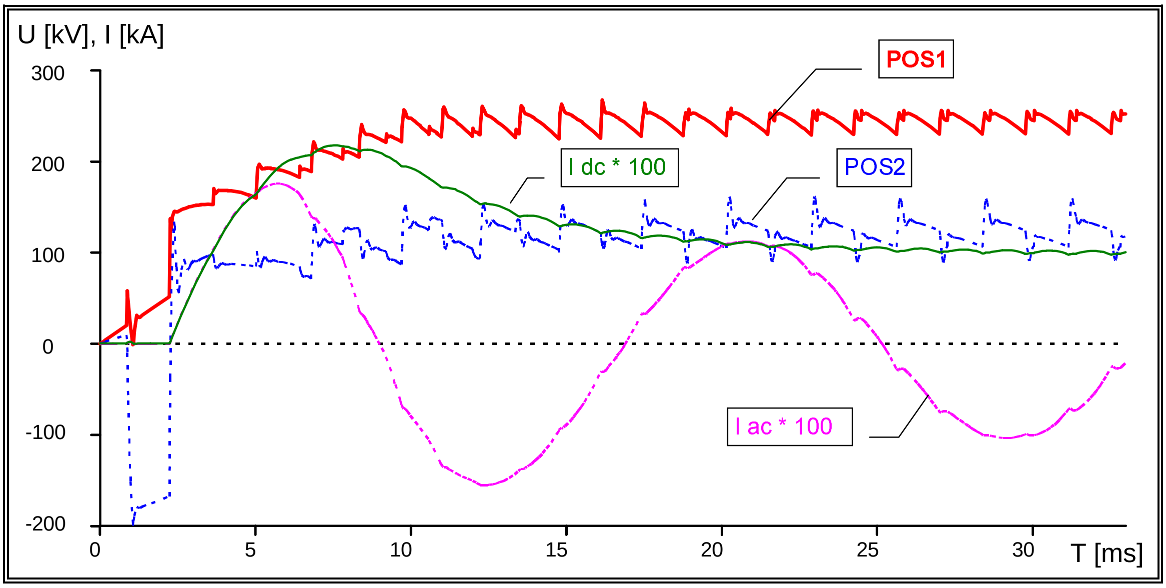


Fig. 5.17 - ATP simulation results

5.5 Using Overhead Line (PCH) objects

This example demonstrates how to use the *Overhead Line (PCH)* objects, the new program module at the bottom of the Component selection menu, for modeling transmission lines and cables in ATPDraw. PCH objects are also based on the Data Base Modularization option in ATP, but users do not need to create supporting files, because they are already pre-defined for the most frequently used single and multi-phase frequency dependent JMarti lines, constant parameter distributed (KCLee/Clarke) lines and PI-equivalent lines. When the pre-defined supporting files do not meet the user's requirements, it is also possible to create own supporting files, similarly to that in the previous example.

The Overhead Line (PCH) menu item enables loading of punch files created by the Line/Cable constants supporting routines. After selecting a punch file, the format is checked and if recognized, a .LIB file with the same name as the .PCH file will be created automatically. ATPDraw converts the .PCH files to be compatible with the Data Base Modularization format internally, so you do not need to run ATP to compile a .LIB file which then will be $Includ-ed at the simulation. This module thus makes it possible to include overhead lines and cables efficiently in ATPDraw.

5.5.1 Creating Line Constants data files

This example shows how to create frequency dependent JMarti line models and how to interface them with ATPDraw as an Overhead Line (PCH) object. To use this feature, the user must be familiar with the usage of the LINE CONSTANTS and the JMARTI SETUP supporting routines of ATP (see in [3]). A completely new program called ATP_LCC has been developed to perform this task easily for the user (see Section 6 - Line/Cable Manual of this book). In this program the user specifies the cross section data and material data for an overhead line or a cable. Based on this data, the ATP_LCC generates an input file in the correct format for the LINE CONSTANT or JMARTI SETUP support of ATP. As a result, ATP produces a punch output (`.PCH`) file.

The example line is a 138 miles long 500 kV overhead line (from John Day to Lower Monumental) taken from the benchmark `DCN3.DAT` in ATP. The JMarti setup file (`JMARTI_1.ATP`) created by the ATP_LCC program (see Chapter 6 of this manual) is printed below:

```
BEGIN NEW DATA CASE
JMARTI SETUP
$ERASE
LINE CONSTANTS
ENGLISH
  1 .364  .05215 4           1.602  -20.75     50.     50.
  1 .364  .05215 4           1.602  -19.25     50.     50.
  2 .364  .05215 4           1.602   -0.75    77.5    77.5
  2 .364  .05215 4           1.602     .75    77.5    77.5
  3 .364  .05215 4           1.602   19.25     50.     50.
  3 .364  .05215 4           1.602   20.75     50.     50.
  0   .5    2.61 4            .386   -12.9    98.5    98.5
  0   .5    2.61 4            .386    12.9    98.5    98.5
BLANK CARD ENDING CONDUCTOR CARDS
    100.     5000.         1                     138.     1            1
    100.       60.         1                     138.     1            1
    100.       .01         1                     138.     1   9 10     1 0
BLANK CARD ENDING FREQUENCY CARDS
BLANK CARD ENDING LINE CONSTANT
               0       0       0       0
       0       .3      30              1       1
       0       .3      30              1       1                        0
$PUNCH
BLANK JMARTI
BEGIN NEW DATA CASE
BLANK CARD
```

The optional `BRANCH` request cards are not supported in the present version of ATP_LCC. However no such card is needed, because ATPDraw will specify branches with 6 characters long node names beginning with `IN___`, `OUT__` and having *A*, *B* and *C* at the end.

Running this file through ATP will produce the punch file printed below (`JMARTI_1.PCH`) which is not really suitable for human reading, but ATPDraw is able to interpret this file and to create a corresponding Data Base Module `.LIB` file from it, as shown next.

```
C  <++++++>  Cards punched by support routine on  30-June-98  11.41.29  <++++++>
C     ***** UNTRANSPOSED JMARTI line segment ******
C JMARTI SETUP
C $ERASE
C LINE CONSTANTS
C ENGLISH
C   1 .364  .05215 4           1.602  -20.75     50.     50.
C   1 .364  .05215 4           1.602  -19.25     50.     50.
C   2 .364  .05215 4           1.602   -0.75    77.5    77.5
C   2 .364  .05215 4           1.602     .75    77.5    77.5
C   3 .364  .05215 4           1.602   19.25     50.     50.
C   3 .364  .05215 4           1.602   20.75     50.     50.
C   0   .5    2.61 4            .386   -12.9    98.5    98.5
C   0   .5    2.61 4            .386    12.9    98.5    98.5
C BLANK CARD ENDING CONDUCTOR CARDS
C     100.     5000.         1                     138.     1            1
C     100.       60.         1                     138.     1            1
C     100.       .01         1                     138.     1   9 10     1 0
C BLANK CARD ENDING FREQUENCY CARDS
C BLANK CARD ENDING LINE CONSTANT
C               0       0       0       0
C       0       .3      30              1       3
-1                            2.  0.00              -2 3
      24        4.7483546645741773800E+02
```

```
   1.61937912421391417E+02   2.31271248199659422E+04  -2.27381516291362205E+04
   1.35631236309925783E+02   1.57786715004263443E+02   3.02545079995990478E+02
   1.92718142213451842E+02   3.06916979107774637E+02   1.40565095310774655E+03
   5.84974602951421639E+03   2.37637828336455205E+04   7.57213487861937320E+04
   2.49726250955637108E+05   7.17969011577112601E+05   2.18739783208210301E+06
   8.35988485893377196E+06   1.78579105617950857E+07   9.25495139350984991E+06
   6.26011551835864131E+06   1.25978962097961307E+07   1.25527002542942185E+07
   2.80032780539078378E+07   1.92913450647414551E+07   4.03216752009965107E+07
   2.84904979620343335E-01   4.72899113362671653E-01   4.73642438007043920E-01
   7.05449950994055608E-01   1.09341761282020467E+00   2.05612703682514742E+00
   3.59754576341474009E+00   8.68253903635959290E+00   3.50926881498620773E+01
   1.47307801542858158E+02   6.23008701713353276E+02   2.11240321074656277E+03
   7.33111306083741056E+03   2.25091621950892732E+04   7.30745289484094538E+04
   2.95643915322969959E+05   1.31201751479643095E+06   2.77646717695323378E+06
   3.79339438676111680E+06   7.29980910128741153E+06   7.90922963082673308E+06
   1.71681772938710191E+07   1.17125935092542936E+07   2.51135726015069299E+07
      16         8.67728887124226300E-04
   1.11203817425757010E-02   2.04652610945998876E-01   3.14550427345624384E-01
   4.07808702949896907E-01   5.02522923182327386E-01   6.87364165349635848E-01
   2.62610601854125259E+00   1.00822089482542179E+01   6.95284062460513610E+01
   2.77269184278038893E+02   4.71527278407562335E+03  -2.59664340760941969E+03
   5.82641011690053802E+03   5.31255526957045659E+04   1.77357634368528910E+07
  -1.77971956629667096E+07
   4.38541538138593801E+00   7.82493081676832106E+01   1.19646401465681536E+02
   1.52666237349410978E+02   1.90741605922539890E+02   2.51348826612552387E+02
   5.23962430599736763E+02   4.42352317124616946E+02   1.29246742864321846E+03
   2.35058685797840099E+03   5.11049238969389899E+03   5.40193598424134689E+03
   1.16678754042265010E+04   2.67944226558872652E+04   2.03919613417938527E+04
   2.04123533031356529E+04
-2                              2.   0.00                -2 3
      18         2.85808760446322253000E+02
   2.38967853101196994E+02   9.94796071525319008E+02  -8.28470217129222534E+02
   1.63430465199159244E+02   3.09062705764837916E+02   3.05699254724883190E+02
   3.73769873937279784E+02   3.62703527974681265E+02   1.85566471109298619E+02
   7.66234942279643150E+01   1.37531836430123406E+02   6.68492111664738502E+01
   7.01118207557536124E+01   1.06728341414327645E+02   2.07647691919385853E+03
   7.12042639966596199E+03   2.05902541118038440E+05   5.04179092207166739E+06
   2.33459891876313053E-01   3.21996573664507236E-01   3.30191941571867210E-01
   6.09242577054224355E-01   1.20782380234957044E+00   2.07863012470610720E+00
   3.61893572243173534E+00   6.05698807947469220E+00   9.68249708472598237E+00
   1.25991008725694549E+01   2.14714699714826374E+01   3.55831313559903748E+01
   6.49984039582018909E+01   9.65570139039255509E+01   1.68151638281782789E+03
   5.79679105116642677E+03   1.67796518984579511E+05   4.12644312401545374E+06
      13         7.42711274958369832000E-04
   2.71241953077073876E-02   2.72206819366755078E+01   1.73847763313030584E+01
   2.15264403892287071E+01   4.71240893706049463E+02   8.77197062686300797E+02
   2.02572720900844216E+04   7.43395452398797352E+04   2.34951357245135587E+06
   1.03201563722517002E+09  -1.10651980722473622E+09   2.21803386621226025E+09
  -2.14597522119946790E+09
   4.71050503746044758E+00   4.78035106832746260E+03   3.38171988134823505E+03
   3.22431840769873952E+03   1.91095033425645016E+04   1.16518123239540030E+05
   1.05067867695843554E+05   2.38533313358663407E+05   6.30474419322799310E+05
   9.35226293313646806E+05   9.36161519606960589E+05   9.86806241973502562E+05
   9.87793048215475283E+05
-3                              2.   0.00                -2 3
      20         2.72444060840364557000E+02
   2.31571202247750279E+02  -2.16307119459785372E+03   2.33546384594222764E+03
   1.22845012919678411E+02   3.54336122050243433E+02   3.18643276409119836E+02
   2.35527195101345456E+02   6.08766963248312436E+02   3.62944414646835015E+01
   8.11839999548129896E+01   1.28680509614660452E+02   1.23062774458542834E+02
   3.12012656363901364E+01   4.43215003806123378E+02   1.27420103535939893E+03
   1.41309094008195939E+03   2.05905201080906955E+03   1.03845444837157648E+04
   7.21869346611934307E+04   1.24129379505568464E+06
   2.26111821626729992E-01   3.23929478349008815E-01   3.20111183092964269E-01
   5.62029362012544720E-01   1.23831178335615300E+00   2.11154765184744520E+00
   3.37279747295096044E+00   6.76195852503021833E+00   9.28893460154047546E+00
   1.27363060075060127E+01   2.11725714798945220E+01   3.47900803374305739E+01
   6.16650632171658160E+01   7.45728229261837669E+02   2.26318661547996272E+03
   2.19351924883939274E+03   3.46475868553182500E+03   1.74239139420778920E+04
   1.21071679475079990E+05   2.08614843021669332E+06
      19         7.48524124296435969000E-04
```

```
   4.06697226028513836E-03   3.96407150505539152E-02   4.87274752414044255E+00
   5.16090318045150820E+00   5.87806130838518826E+00   2.39530511102252675E+00
   1.86765626044075326E+01   3.40955023404691673E+01   5.40831887205668636E+02
   2.18482792829798291E+03   1.85180981888789866E+04  -2.25285650191682635E+02
   1.81349636745061056E+05   1.16893721888192435E+05   5.14441140087258828E+05
   1.43233527443105126E+09  -1.44639603189362073E+09   6.36152834019420863E+08
  -6.22925850650732518E+08
   1.40342595069212628E+00   1.37261357003022831E+01   1.74588746692735595E+03
   1.65587497328505015E+03   2.07348587209993867E+03   3.68965907569957608E+03
   3.67600426776633868E+03   1.11439047359345896E+04   2.19338558668988335E+04
   4.53906408949455509E+04   1.58569128013578855E+05   4.85035764955933847E+05
   4.03562005910227600E+05   7.83388464155936265E+05   1.33468956336463523E+06
   3.82287537962081144E+06   3.82669825500042877E+06   4.46413234689150751E+06
   4.46859647923839652E+06
  0.57153211  0.70710678 -0.41762016
  0.00000000  0.00000000  0.00000000
  0.58881414  0.00000000  0.80696147
  0.00000000  0.00000000  0.00000000
  0.57153211 -0.70710678 -0.41762016
  0.00000000  0.00000000  0.00000000
```

5.5.2 Creating new Overhead Line (PCH) objects

The punch file created by a JMarti setup ATP simulation, needs to be converted to a Data Base Module library file. This is done automatically by ATPDraw if you choose a `.PCH` file in the *Open Punch File* dialog box appearing when you click on the *Overhead Line (PCH)* option in the component selection menu. In this example, electrical parameters of the 500 kV transmission line are described in the `JMARTI_1.PCH` file. Selecting this file from the list, ATPDraw interprets the format and creates a `.LIB` file with the same name as the `.PCH` file. The results of the interpretation are reported to the user in dialog boxes, as show in Fig. 5.18.

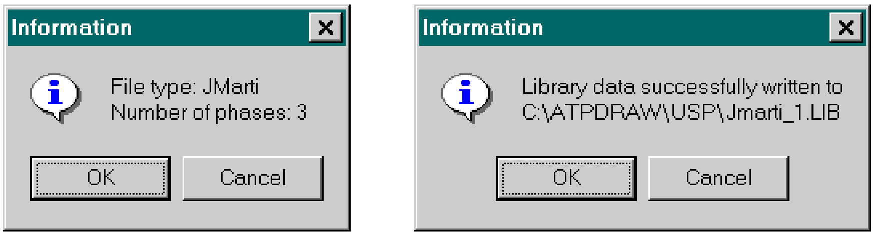
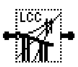


Fig. 5.18 - Punch file interpretation

Finally, ATPDraw selects a support file from the pre-defined alternatives based on the number of phases and the type of the line (constant parameter or frequency dependent) and displays the new line object icon in the middle of the circuit window.



In this example the number of phases is 3 and a frequency dependent JMarti line is described in the punch file, so the icon shown left will be chosen.

The structure of the ATPDraw created library file (`JMARTI_1.LIB`) is identical with that of the DBM punch files produced by DATA BASE MODULE supporting routine of ATP (see next).

```
KARD  2  2 33 33 58 58
KARG  1  4  2  5  3  6
KBEG  3  9  3  9  3  9
KEND  8 14  8 14  8 14
KTEX  1  1  1  1  1  1
/BRANCH
-1IN___AOUT__A                2.  0.00             -2 3
      24       4.74835466457417738OE+02
```

```
      1.61937912421391417E+02   2.31271248199659422E+04  -2.27381516291362205E+04
      1.35631236309925783E+02   1.57786715004263443E+02   3.02545079995990478E+02
      1.92718142213451842E+02   3.06916979107774637E+02   1.40565095310774655E+03
      5.84974602951421639E+03   2.37637828336455205E+04   7.57213487861937320E+04
   ..
   ..!!! Same as in the file JMARTI_1.PCH, so it is not repeated here !!!
   ..
   -2IN___BOUT__B               2.   0.00              -2 3
   ..
   ..!!! Same as in the file JMARTI_1.PCH, so it is not repeated here !!!
   ..
   -3IN___COUT__C               2.   0.00              -2 3
   ..
   ..!!! Same as in the file JMARTI_1.PCH, so it is not repeated here !!!
   ..
   $EOF
   ARG, IN___A, IN___B, IN___C, OUT__A, OUT__B, OUT__C
```

The new line object's data input window is shown in Fig. 5.19. In the *$Include* field the name of the library must be specified which is `JMARTI_1.LIB` in this example. When *Auto path* option is selected in the *ATP | Settings | File format* page, ATPDraw inserts the prefix (`C:\ATPDraw\USP\`) and the extension (`.LIB`) automatically in the ATP file. If the *Auto path* option is disabled, the full path of the library file must be used as the first parameter of the $Include field.

Send parameters attribute must be selected here. Deselect the *Old 3-phase* if the library file has been created via the Overhead Line (PCH) menu. When the library file has been created by the user outside ATPDraw, the *Old 3-phase* option must be used in accordance with the number of arguments in the ARG card of the DBM file. If this option is selected, ATPDraw sends only the first 5 characters of the 3-phase nodes to the $Include argument list. When this option is unchecked, 3x6 character node names will be sent.

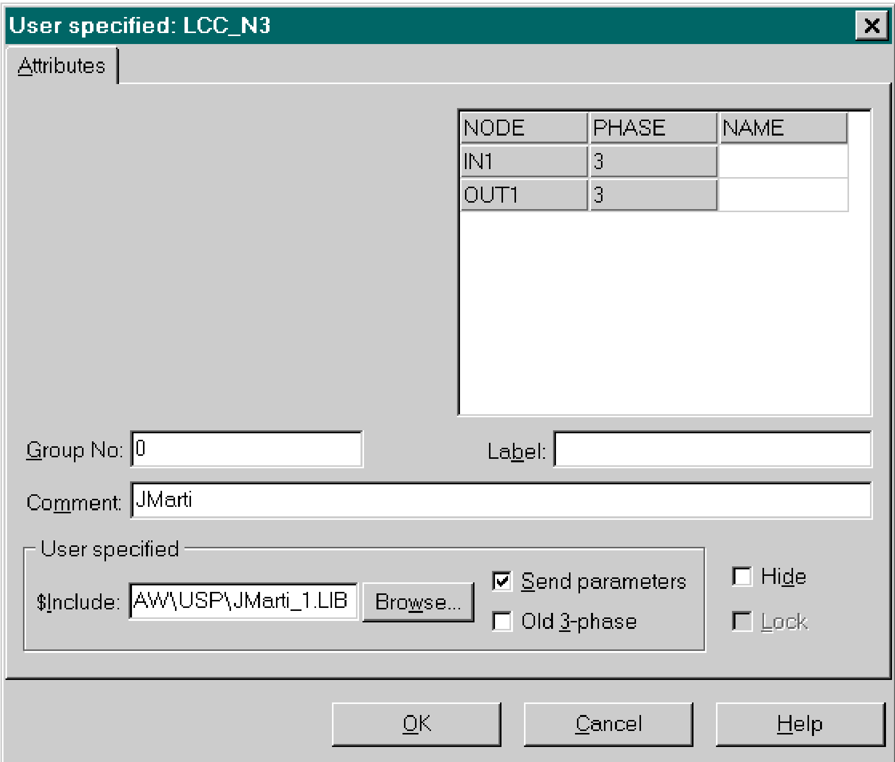


Fig. 5.19 - JMarti object input window

5.5.3 Switching study using the JMarti line *(Exa_7.cir)*

In this chapter the usage of the JMarti line object is demonstrated. The 3-phase switching example (*Exa_3.cir*) created in section 5.1 of this manual is repeated here; this time with a JMarti line instead of the PI-equivalent. The example circuit is shown in Fig. 5.20.

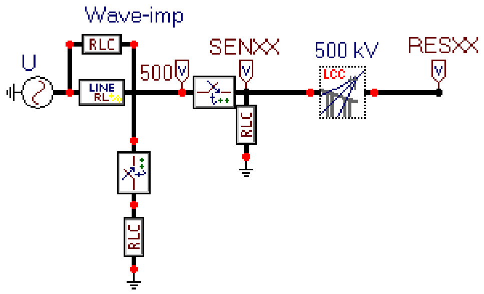


Fig. 5.20 - JMarti line in switching study (`EXA_7.CIR`)

No restriction exists in node naming of the 3-phase nodes at the sending/receiving end of the line, because the corresponding support file and the automatically generated library file support any legal node names. The ATP file generated by ATPDraw for this 500 kV example circuit is listed below:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  Tue, 30 Jun - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
$PREFIX,C:\ATPDRAW\USP\
$SUFFIX, .LIB
$DUMMY, XYZ000
C Miscellaneous Data Card ....
POWER FREQUENCY                      60.
C  dT   >< Tmax >< Xopt >< Copt >
  .00005      .2
     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
51X0001A500A               .55      8.98
52X0001B500B              .711     11.857
53X0001C500C
  X0001A500A              200.                                              0
  X0001B500B              200.                                              0
  X0001C500C              200.                                              0
        X0008A                        2.51                                  0
        X0008B                        2.51                                  0
        X0008C                        2.51                                  0
        SENXXA          1.00E7                                              0
        SENXXB          1.00E7                                              0
        SENXXC          1.00E7                                              0
C JMarti
C User specified object: C:\ATPDRAW\USP\LCC_N3.SUP
C    IN1 = SENXXA
C   OUT1 = RESXXA
$INCLUDE, Jmarti_1, SENXXA, SENXXB, SENXXC, RESXXA, RESXXB, RESXXC
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
```

```
    500A  SENXXA    .03333      10.                                         0
    500B  SENXXB     .0361      10.                                         0
    500C  SENXXC     .0388      10.                                         0
    X0008A500A      .13333      10.                                         0
    X0008B500B       .1361      10.                                         0
    X0008C500C       .1388      10.                                         0
  /SOURCE
  C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
  14X0001A 0   408248.       60.                                     -1.        1.
  14X0001B 0   408248.       60.      -120.                          -1.        1.
  14X0001C 0   408248.       60.       120.                          -1.        1.
  BLANK BRANCH
  BLANK SWITCH
  BLANK SOURCE
    SENXXASENXXBSENXXCRESXXARESXXBRESXXC500A  500B  500C
  BLANK OUTPUT
  BLANK PLOT
  BEGIN NEW DATA CASE
  BLANK
```

The results of the simulation are shown in Fig. 5.21.

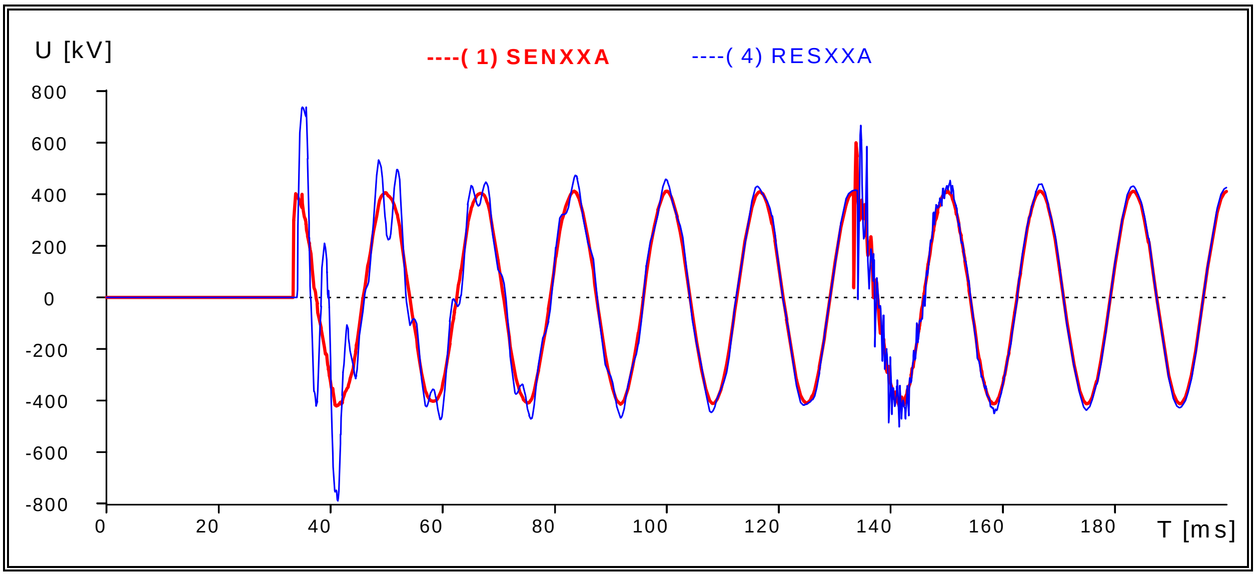


Fig. 5.21 - Switching overvoltages at both ands of a JMarti line

5.5.4 Line to ground fault and fault tripping transients *(Exa_7a.cir)*

In this example the usage of the Overhead Line (PCH) objects in combination with transposition objects are demonstrated. In the DOS version of the program users had to be careful with the usage of transposition objects with the JMarti objects together, since the phase sequence different from ABC was not recognized inside the DBM library file. This limitation does not exist any more in the present version of ATPDraw.

The one-line diagram of the simulated network is shown in Fig. 5.22. In this study, transients caused by a single-line to ground fault on a 750 kV interconnection are investigated. The results of the simulation are validated by means of field test records obtained at a staged fault test on the line.

To simplify the comparison, it was assumed in the simulation that the fault appears at the peak of the phase voltage and the circuit breakers operate exactly at same time as observed during the field test.

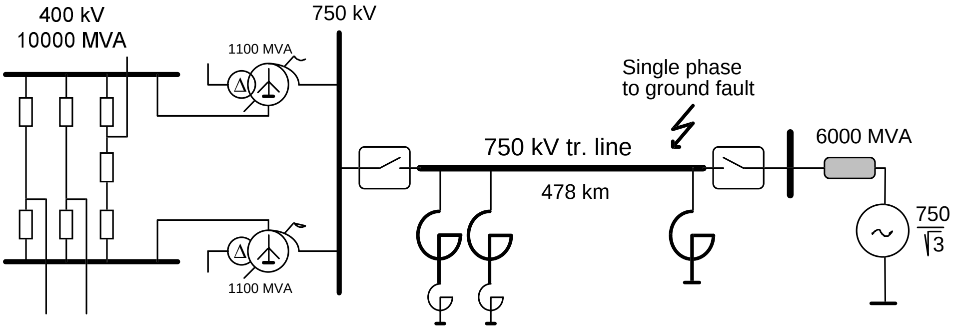


Fig. 5.22 - Line-to-ground fault example

The surplus of capacitive power generation of the 478 km long line is compensated by shunt reactors. At the sending end, two sets of neutral reactors are applied to reduce the amplitude of the secondary arc current. The staged fault has been initiated at the receiving end of the line.

ATP_LCC program has been used to calculate the electrical parameters of the line and to create the JMARTI SETUP input files. Three transposition points exist along the route, so each LINE CONSTANTS case represents a line section with the length equal to 84.6 km, 162.7 km, 155.9 km, 75.7 km, respectively. You can find all LCC input files (`LIN750_1.LIN - LIN750_4.LIN`) for this case in the ATP_LCC directory. The JMARTI SETUP file created by LCC for the first line section is shown below as an example.

```
BEGIN NEW DATA CASE
JMARTI SETUP
$ERASE
LINE CONSTANTS
METRIC
  1  .32   .0585 4           3.105   -17.5    27.9     13.     60.   45.       4
  2  .32   .0585 4           3.105       0    27.9     13.     60.   45.       4
  3  .32   .0585 4           3.105    17.5    27.9     13.     60.   45.       4
  0 .187    .304 4             1.6   -13.2   41.05   26.15       0     0       0
  0 .187    .304 4             1.6    13.2   41.05   26.15       0     0       0
BLANK CARD ENDING CONDUCTOR CARDS
     20.     5000.         1                    84.6     0           1
     20.       50.         1                    84.6     0           1
     20.       .05         1                    84.6     0   7 10    1 0
BLANK CARD ENDING FREQUENCY CARDS
BLANK CARD ENDING LINE CONSTANT
               0       0       0       0
       0      1.      30                       1
       0      1.      30                       1                     .05
$PUNCH
BLANK JMARTI
BEGIN NEW DATA CASE
BLANK CARD
```

Processing the LCC created files with ATP separately, four punch files (`LIN750_1.PCH - LIN750_4.PCH`) will be generated which can be used as input for the Overhead Line (PCH) objects of ATPDraw, as it was discussed in the previous example.

The layout of the complete circuit is shown in Fig. 5.23.

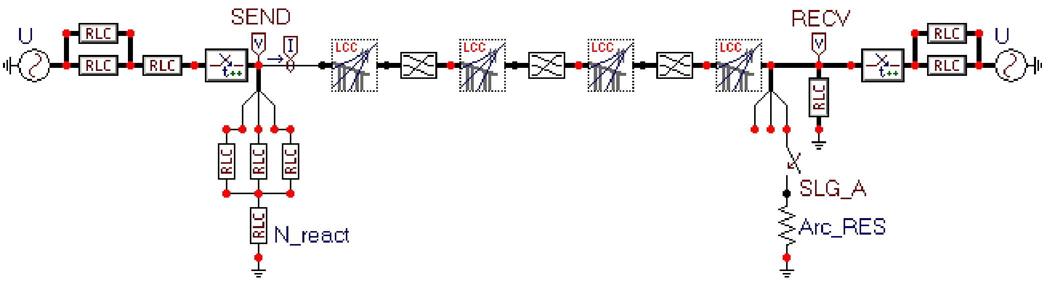


Fig. 5.23 - Line-to-ground fault study (`EXA_7a.CIR`)

The supply side networks connecting to the 750 kV transmission line at the sending end and at the receiving end are rather simple. Only the positive sequence short circuit capacity of the supplies has been taken into account by 3-phase RLC objectc, connected parallel with a resistor representing the equivalent surge impedance seen from the terminals. An uncoupled series reactance simulates the short circuit inductance of the transformer bank consisting of three single-phase units. The shunt reactors, -which are also single-phase devices- have been represented by linear RLC components, because the predicted amplitude of the reactor voltages are far below the saturation level of the air gapped core. The arc resistance of the fault is assumed to be constant, and as an approximation 2 ohms were applied in the study.

The ATPDraw generated file for this 750 kV example circuit is listed below:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  Tue, 30 Jun - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
$PREFIX,C:\ATPDRAW\USP\
$SUFFIX, .LIB
$DUMMY, XYZ000
C Miscellaneous Data Card ....
C  dT   >< Tmax >< Xopt >< Copt >
  .00002       .5
     500       5       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
  SLG_A                        2.                                               0
  XX0011SENDA                 10. 3000.                                         0
  XX0011SENDB                 10. 3000.                                         0
  XX0011SENDC                 10. 3000.                                         0
        XX0011                 1.  300.                                         0
  X0022AX0024A                 5.  180.                                         0
  X0022BX0024B                 5.  180.                                         0
  X0022CX0024C                 5.  180.                                         0
  X0022AX0024A               150.                                               0
  X0022BX0024B               150.                                               0
  X0022CX0024C               150.                                               0
  X0032AX0031A                 5.  300.                                         0
  X0032BX0031B                 5.  300.                                         0
  X0032CX0031C                 5.  300.                                         0
  X0032AX0031A               150.                                               0
  X0032BX0031B               150.                                               0
  X0032CX0031C               150.                                               0
```

```
         RECVA                  20. 6000.                                     0
         RECVB                  20. 6000.                                     0
         RECVC                  20. 6000.                                     0
   X0024AX0027A                  2.  200.                                     0
   X0024BX0027B                  2.  200.                                     0
   X0024CX0027C                  2.  200.                                     0
C JMarti
C User specified object: C:\ATPDRAW\USP\LCC_N3.SUP
C     IN1 = LN1A
C    OUT1 = TRAN1A
$INCLUDE, LIN750_1, LN1A##, LN1B##, LN1C##, TRAN1A, TRAN1B, TRAN1C
C JMarti
C User specified object: C:\ATPDRAW\USP\LCC_N3.SUP
C     IN1 = TRAN1C
C    OUT1 = TRAN2C
$INCLUDE, LIN750_2, TRAN1C, TRAN1A, TRAN1B, TRAN2C, TRAN2A, TRAN2B
C JMarti
C User specified object: C:\ATPDRAW\USP\LCC_N3.SUP
C     IN1 = TRAN2B
C    OUT1 = TRAN3B
$INCLUDE, LIN750_3, TRAN2B, TRAN2C, TRAN2A, TRAN3B, TRAN3C, TRAN3A
C JMarti
C User specified object: C:\ATPDRAW\USP\LCC_N3.SUP
C     IN1 = TRAN3A
C    OUT1 = RECVA
$INCLUDE, LIN750_4, TRAN3A, TRAN3B, TRAN3C, RECVA#, RECVB#, RECVC#
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
  RECVA SLG_A      .0285      .225       10.                                  0
  X0027ASENDA        -1.      .075                                            0
  X0027BSENDB        -1.        1.                                            0
  X0027CSENDC        -1.        1.                                            0
  SENDA LN1A                                               MEASURING          1
  SENDB LN1B                                               MEASURING          1
  SENDC LN1C                                               MEASURING          1
  RECVA X0032A       -1.      .075                                            0
  RECVB X0032B       -1.        1.                                            0
  RECVC X0032C       -1.        1.                                            0
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
14X0022A 0   612300.       50.                                        -1.        1.
14X0022B 0   612300.       50.      -120.                             -1.        1.
14X0022C 0   612300.       50.       120.                             -1.        1.
14X0031A 0   612300.       50.        10.                             -1.        1.
14X0031B 0   612300.       50.      -110.                             -1.        1.
14X0031C 0   612300.       50.       130.                             -1.        1.
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
  SENDA SENDB SENDC RECVA RECVB RECVC
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

Fig. 5.24 shows the results of the simulation. The first three curves are the phase-to-ground voltages at the receiving end of the line. When the secondary arc current extinguishes an oscillating trapped charge appears on the faulty phase, which is the characteristics of the shunt compensated lines. The next figure shows the phase currents during and after the fault.

Fig. 5.25. shows the oscillograms of the phase voltages and currents recorded by high speed transient disturbance recorder at the staged fault test. The field test records are shown here just for verification of the simulation.

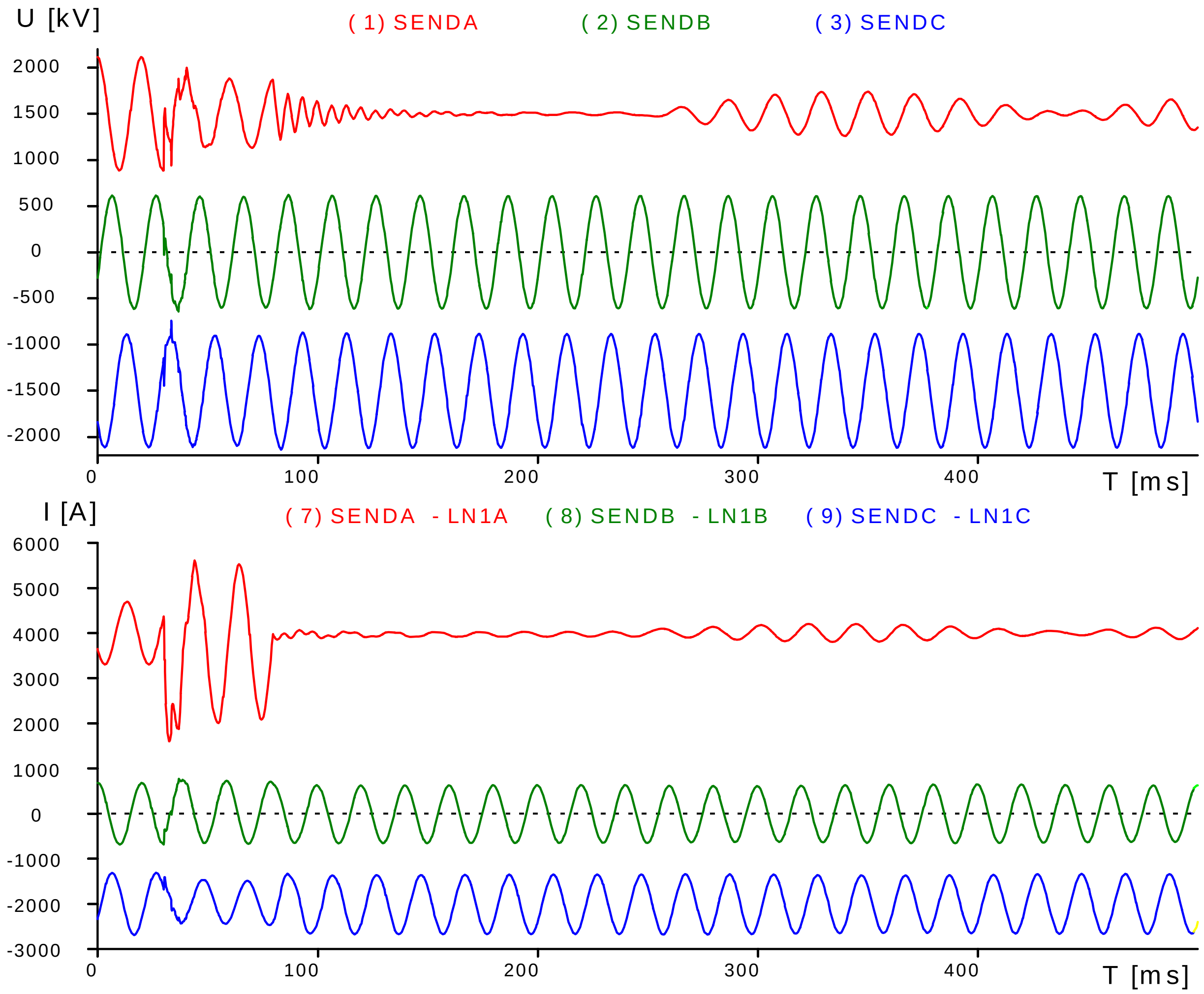


Fig. 5.24 - SLG fault and fault clearing transients (simulation)

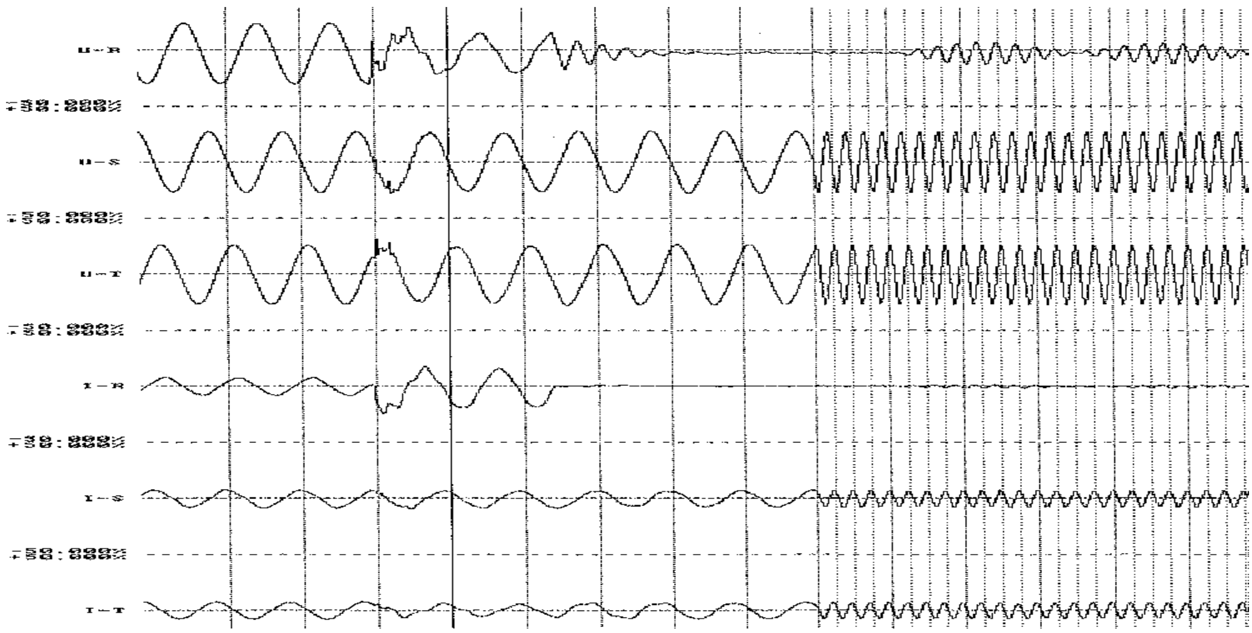


Fig. 5.25 - SLG fault and fault clearing transients (field test)

5.6 Usage of MODELS

In this chapter the usage of MODELS [4] in ATPDraw will be explained. ATPDraw supports only a simplified usage of MODELS. The following facilities are not included:

- nested models
- expressions in the use statements
- record of models variables

The example in this chapter is taken from benchmark `DC68.DAT`, subcase 7. The data case is a 500 kV system from Coulee to Raver with a series capacitor at Columbia. The capacitors are protected with ZnO arresters and with MODELS controlled switches.

5.6.1 Creating the model file

ATPDraw handles the input/output to the MODELS section of the ATP file and the usage of each model. The actual model file describing the model must be written outside of ATPDraw. The model in `DC68.DAT` is printed below. The model has been modified a bit since it is not allowed to use expressions in the USE section of a model. Instead of calculating the voltage across the arrester in the USE statement, the two node voltages at each side are sent as input parameters and the difference is calculated inside the model `vcap:=V1-V2`.

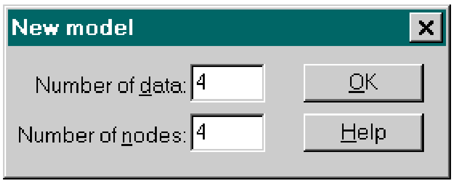
```
MODEL FLASH_1

comment *****************************************************************
*                                                                       *
*    Function: set or cancel the gap firing control signal              *
*    Inputs  : voltage and current across ZnO resistor                  *
*    Output  : the firing signal to the electrical ZnO component        *
*                                                                       *
************************************************************* endcomment

  INPUT  V1     --  Voltage on positive side of ZNO        [V]
         V2     --  Voltage of negative side of ZNO        [V]
         iczn   --  ZNO current                            [Amps]

  DATA   Pset   --  power setting                          [Megajoules/msec]
         Eset   --  energy setting                         [Megajoules]
         fdel   --  firing delay                           [msec]
         fdur   --  firing duration                        [msec]

  VAR   power   --  power into ZnO resistor                [Watts]
        trip    --  gap firing control signal              [0 or 1]
        energy  --  energy into ZnO resistor               [Joules]
        tfire   --  time at which the gap was last fired   [sec]
        vcap    --  voltage difference across series caps  [Volts]

  OUTPUT  trip

  HISTORY  INTEGRAL(power) {DFLT:0}

  INIT trip:=0
       tfire:=0
  ENDINIT

  EXEC
    ------------------------------------------------------------------
    vcap:=V1-V2
    power:=vcap*iczn
    energy:=INTEGRAL(power)
    ------------------------------------------------------------------
```

```
      IF  trip>0              -- is already firing
      AND t-tfire>fdur*1.e-3  -- has exceeded firing duration
      THEN
        trip:=0     -- cancel the firing signal
        tfire:=0    -- null the firing time
      ENDIF
      ------------------------------------------------------------------
      IF trip=0                         -- is not signaling to fire
      AND tfire=0                       -- firing condition not yet detected
      AND (     power  >= Pset * 1.e9   -- power setting exceeded
            OR energy >= Eset * 1.e6 ) -- energy setting exceeded
      THEN
        tfire:=t    -- set the firing detection time
      ENDIF
      ------------------------------------------------------------------
      IF trip=0                  -- is not signaling to fire
      AND tfire>0                -- firing condition has been detected
      AND t-tfire>=fdel*1.e-3    -- firing delay exceeded
      THEN
        trip:=1     -- set the firing signal
      ENDIF
      ------------------------------------------------------------------
    ENDEXEC
  ENDMODEL
```

The model file must be given a name with extension `.MOD` and it must be stored in the `\MOD` folder. In this example the name `FLASH_1.MOD` was chosen. The name of the model file must be the same as that of the actual model.

5.6.2 Creating new MODELS object in ATPDraw

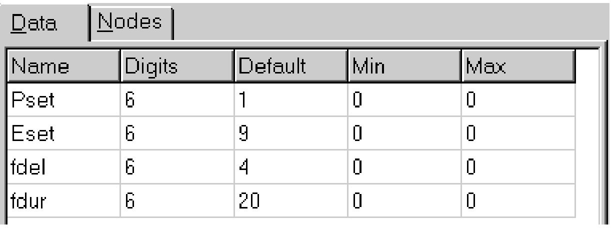
To be able to use the just created model definition inside ATPDraw, a model object must be created. This process is similar to that established in section 5.4.2 of this manual. To create a new object start up ATPDraw, enter the *Objects* menu and select the *New Model* field.



A window appears where the user must specify the size of the model. The model shown above has 4 nodes (input+output) and 4 data. All MODELS nodes must be single phase.

Fig. 5.26 - Specifying size of model

After you have specified the values shown in Fig. 5.26, click the *OK* button and the window closes. Then a tabbed notebook style dialog box appears with two attribute tabs: *Data* and *Nodes*. Select the *Data* page where you must specify the values shown in Fig. 5.27.



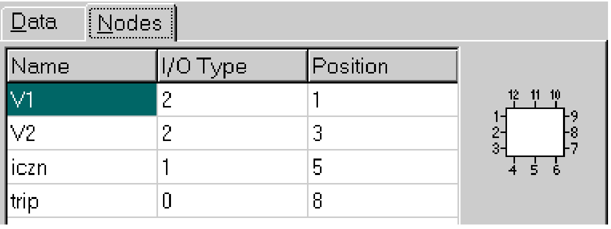
Data | Nodes

Name	Digits	Default	Min	Max
Pset	6	1	0	0
Eset	6	9	0	0
fdel	6	4	0	0
fdur	6	20	0	0

The name of the data must be the same as those used in the Model file `.MOD`.

Fig. 5.27 - Specifying data values for the model object

The default values are taken from the Use Model statements in `DC68.DAT` (you can of course change these values individually for each use of the model). The value specified in the *Digit* field is the number of digits used in the Use Model statement. No restriction is applied to data values, so *Min=Max* is set. After you have specified the values shown in Fig. 5.27 click the *Nodes* tab and enter to the node window as shown in Fig. 5.28:



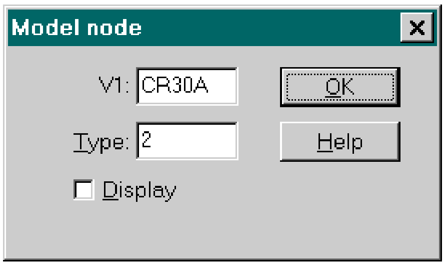
The name of the nodes must be the same as that used in the model file.

Fig. 5.28 - Specifying node values

The *I/O Type* is a code for the type of parameter:

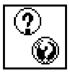
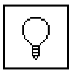
0: output	1: current input
2: voltage input	3: switch status input
4: machine variable input	

The *Position* field is the node position on the icon border as shown at the right side of the figure.



The *I/O Type* values can be changed later in the node input window (menu *Type*) for each model objects, as shown in Fig. 5.29. This window appears when the user clicks on a Model node with the right mouse button.

Fig. 5.29 - Input window of model nodes

Model objects also must have an icon which represents the object on the screen and an optional help which describes the meaning of parameters. If no user supplied help text is given the Help viewer will display the model definition file (`*.MOD`) automatically, so it is advised to use this feature. If you really need your own help file click the  button at the right hand side of the New Model dialog box to open the help editor. The icon editor appears similarly, by clicking the  button. Here you can be creative and draw a suitable icon as shown in Fig. 5.30. When you finished select the *Done* menu item.

Finally, when you click the *Save* or *Save As* buttons the support file of the new model object will be saved to disk. A file window appears where you must specify the name of the object. The object does not need to have the same name as the model file. The object should have the extension `.SUP` and be stored in the `\MOD` folder. You can reload and modify the support file of the model objects whenever you like, using the *Edit Model* item in the *Objects* menu.

The new model object has now been created and it is ready for use. It can be found in the component selection menu under the *MODELS* field. After you have selected this field, a file window appears where you can select the file `FLASH_1.SUP`. This will draw the Model and it can be edited like any other ATPDraw objects.

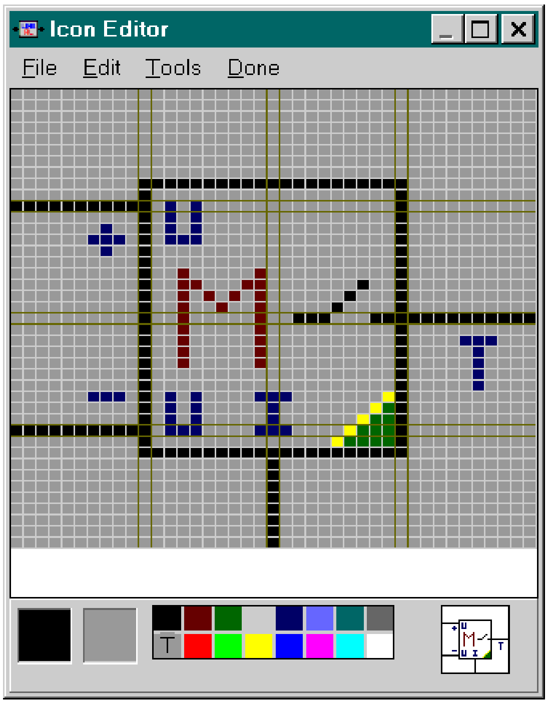


Fig. 5.30 -The icon of the new model object

5.6.3 Using MODELS controlled switches (DC68.DAT) *(Exa_8.cir)*

The Model created in chapter 5.6.2 corresponds to the one used in the benchmark file `DC68.DAT`, used to control three switches protecting a series capacitor circuit. The actual circuit is shown in Fig. 5.31.

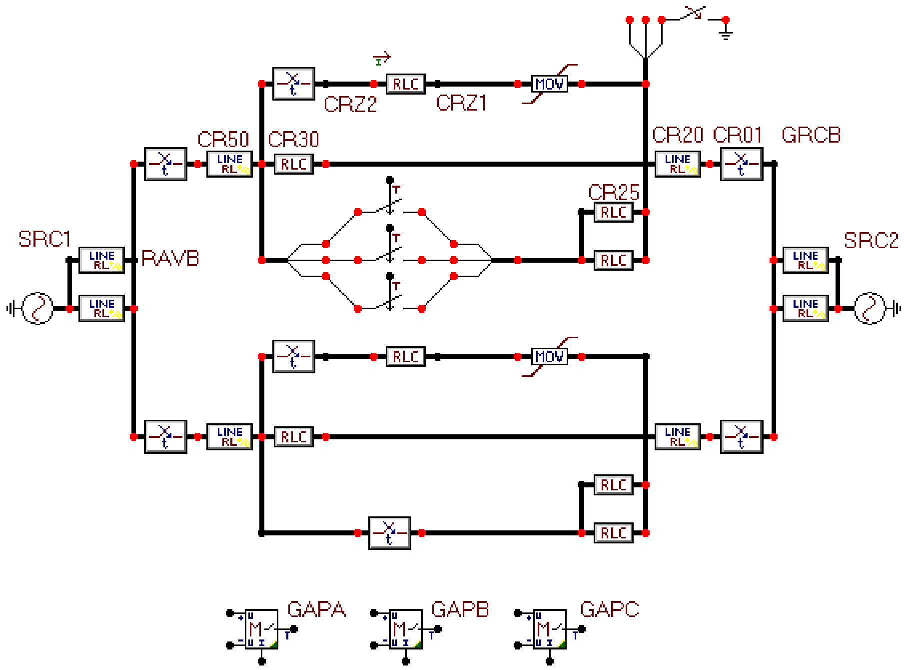


Fig. 5.31 - DC68.DAT , usage of MODELS in ATPDraw (`EXA_8.CIR`)

No connections are drawn between the three models in Fig. 5.31 and the nodes in the electrical network. Instead, the same node names are specified both in the models and in the circuit. This results in a warning such as shown in Fig. 5.32, when the ATP file is created, but it can be ignored in this case by clicking *OK*. If you click *Abort*, ATPDraw marks the node names with identical name in cyan color. If you wish, you can use the *Draw relation* feature available under TACS in the component selection menu, to visualize the connection between the model objects and the electrical network.

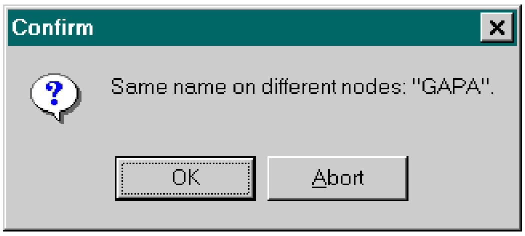


Fig. 5.32 - Warning message that can be ignored here

The three Models must have different **Use As** names. If you click the right mouse button on the Model controlling GAPA, an input window will appear where you can specify the parameters shown in Fig. 5.33. The *Model file* field is the name and path of the file which contains the model definition written in MODELS simulation language [4]. The *Use As* field can contain any name, but the name must be different for all the three Models in Fig. 5.31.

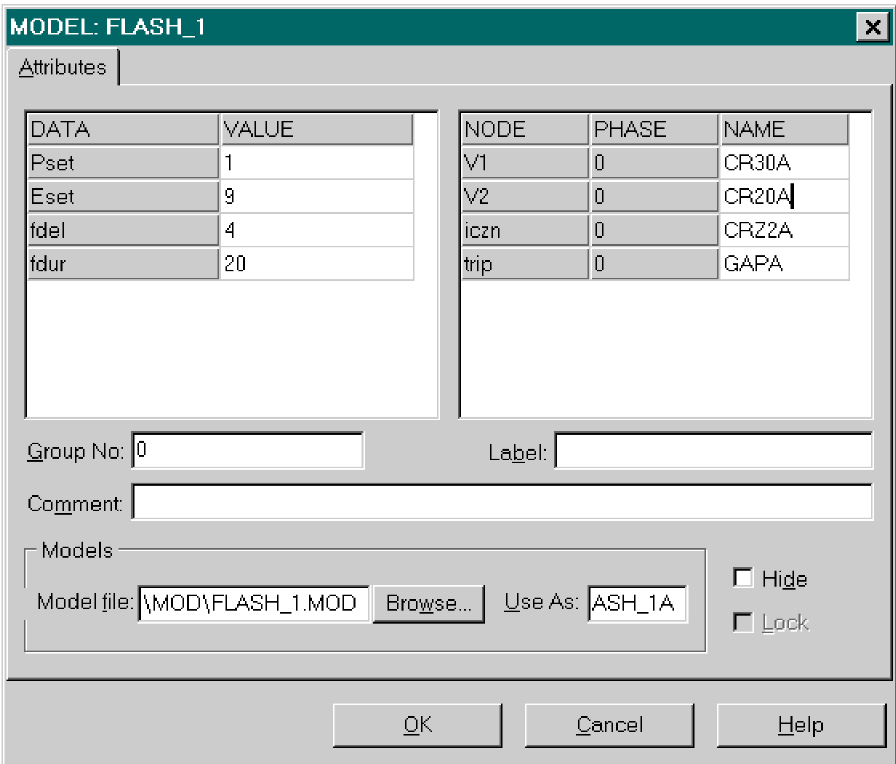


Fig. 5.33 - Attributes of Model objects

If you right click on one of the arresters, an input window appears where you can specify the parameters. In section 4.2.9.3 of the Reference Manual the usage of the *Attributes* and *Characteristic* page of ZnO arresters are explained in detail, so it is not repeated here.

The characteristics of ZnO arresters can not be specified directly by exponential functions in ATPDraw, since ATPDraw uses the current/voltage characteristics and performs an exponential fitting internally. However the nonlinear characteristics can be taken into account using $Include. This requires a text file named e.g. `ZNO_1.LIB` stored in the `\USP` directory in the following format:

```
                     1.0                      40.                        .80
                   9999.
```

The *$Include* field on the Characteristic page must contain the name of this file and the *Include characteristic* button must be on in such cases. If you choose the *Browse* button a file window will appear where you can select a file for the characteristic.

You must be careful with using $SUFFIX and $PREFIX options, when the files you want to include are located in different subdirectories. As in this case, where the nonlinear characteristics of the ZnO blocks are in the `\USP` folder and the models description file is in the `\MOD` folder.

To avoid conflicts, you are advised to use the full path of all included files, because ATPDraw does not support the $PREFIX/$SUFFIX in such a combination. This could result in some problems when exchanging ATPDraw circuit files with an other user who has installed the program in a different drive and or in a different directory, but this can be solved under *Tools | Options | Directories*. Alternatively, this difficulty can be avoided by installing all include files in the same subdirectory, called for example: `\LIB`.

The ATP file of Fig. 5.31 is listed below. The cards marked bold (2nd integer miscellaneous card, **RECORD** in MODELS and **PRINTER PLOT** cards) are added to the file using the *Edit file* option in the ATP menu. The direction of rotation of the source must also be changed (B ⇔ C) since `DC68.DAT` uses a negative sequence AC source rotation.

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW Tue, 30 Jun - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
C Miscellaneous Data Card ....
POWER FREQUENCY                        60.
C  dT   >< Tmax >< Xopt >< Copt >
  .00005     .05     60.
       1       9       1       3       1      -1       0       1       0
       5       5      20      20     100     100
MODELS
INPUT
  IX0001 {v(CR30A )},
  IX0002 {v(CR20A )},
  IX0003 {i(CRZ2A )},
  IX0004 {v(CR30B )},
  IX0005 {v(CR20B )},
  IX0006 {i(CRZ2B )},
  IX0007 {v(CR30C )},
  IX0008 {v(CR20C )},
  IX0009 {i(CRZ2C )},
OUTPUT
  GAPA   ,
  GAPB   ,
  GAPC   ,
$INCLUDE, C:\ATPDRAW\MOD\FLASH_1.MOD
USE FLASH_1 AS FLASH_1A
```

```
INPUT
  V1:= IX0001
  V2:= IX0002
  iczn:= IX0003
DATA
  Pset:= 1.000E+0000
  Eset:= 9.000E+0000
  fdel:= 4.000E+0000
  fdur:= 2.000E+0001
OUTPUT
  GAPA  :=trip
ENDUSE
USE FLASH_1 AS FALSH_1B
INPUT
  V1:= IX0004
  V2:= IX0005
  iczn:= IX0006
DATA
  Pset:= 1.000E+0000
  Eset:= 9.000E+0000
  fdel:= 4.000E+0000
  fdur:= 2.000E+0001
OUTPUT
  GAPB  :=trip
ENDUSE
USE FLASH_1 AS FLAS_1C
INPUT
  V1:= IX0007
  V2:= IX0008
  iczn:= IX0009
DATA
  Pset:= 1.000E+0000
  Eset:= 9.000E+0000
  fdel:= 4.000E+0000
  fdur:= 2.000E+0001
OUTPUT
  GAPC  :=trip
ENDUSE
RECORD
  FLASH_1a.vcap   AS VCAPA
  FLASH_1a.iczn   AS IZNA
  FLASH_1a.power  AS PZNA
  FLASH_1a.energy AS EZNA
  FLASH_1a.trip   AS GAPA
ENDMODELS
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
51SRC1A RAVBA                 .1         20.
52SRC1B RAVBB                 .1        18.5
53SRC1C RAVBC
51SRC1A RAVBA               300.
52SRC1B RAVBB               150.
53SRC1C RAVBC
51CR50A CR30A                 .5         14.
52CR50B CR30B                 8.         52.
53CR50C CR30C
  CR30A CR20A                           93.4                                   0
  CR30B CR20B                           93.4                                   0
  CR30C CR20C                           93.4                                   0
  CRZ2A CRZ1A                .01                                               1
  CRZ2B CRZ1B                .01                                               1
  CRZ2C CRZ1C                .01                                               1
/BRANCH
92X0017AX0068A                         5555.                                   0
 1.4750000000000000E+0005-1.0000000000000000E+0000 0.0000000000000000E+0000    1
```

```
$INCLUDE, C:\ATPDRWN2\USP\ZNO_1.LIB
92X0017BX0068BX0017AX0068A             5555.                                0
92X0017CX0068CX0017AX0068A             5555.                                0
  CR25A CR20A               5.    .23                                       0
  CR25B CR20B               5.    .23                                       0
  CR25C CR20C               5.    .23                                       0
  CR25A CR20A             200.                                              0
  CR25B CR20B             200.                                              0
  CR25C CR20C             200.                                              0
51CR20A CR01A               .4          9.
52CR20B CR01B               5.         36.
53CR20C CR01C
51X0064AX0083A              .5         14.
52X0064BX0083B              8.         52.
53X0064CX0083C
  X0083AX0068A                          93.4                                0
  X0083BX0068B                          93.4                                0
  X0083CX0068C                          93.4                                0
  X0069AX0017A             .01                                              0
  X0069BX0017B             .01                                              0
  X0069CX0017C             .01                                              0
  X0079AX0068A              5.    .23                                       0
  X0079BX0068B              5.    .23                                       0
  X0079CX0068C              5.    .23                                       0
  X0079AX0068A            200.                                              0
  X0079BX0068B            200.                                              0
  X0079CX0068C            200.                                              0
51GRCBA SRC2A               .1          7.
52GRCBB SRC2B               .1         10.7
53GRCBC SRC2C
51GRCBA SRC2A             350.
52GRCBB SRC2B             150.
53GRCBC SRC2C
51X0068AX0095A              .4          9.
52X0068BX0095B              5.         36.
53X0068CX0095C
/BRANCH
92CRZ1A CR20A                          5555.                                0
 1.4750000000000000E+0005-1.0000000000000000E+0000 0.0000000000000000E+0000    1
$INCLUDE, C:\ATPDRWN2\USP\ZNO_1.LIB
92CRZ1B CR20B CRZ1A CR20A              5555.                                0
92CRZ1C CR20C CRZ1A CR20A              5555.                                0
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
  RAVBA CR50A        -1.       10.                                          0
  RAVBB CR50B        -1.       10.                                          0
  RAVBC CR50C        -1.       10.                                          0
  CR30A CRZ2A        -1.        1.                                          0
  CR30B CRZ2B        -1.        1.                                          0
  CR30C CRZ2C        -1.        1.                                          0
13CR30A CR25A                                        CLOSED          GAPA   0
13CR30B CR25B                                                        GAPB   0
13CR30C CR25C                                                        GAPC   0
  CR01A GRCBA     -0.006       10.                                          0
  CR01B GRCBB     -0.006       10.                                          0
  CR01C GRCBC     -0.006       10.                                          0
  RAVBA X0064A       -1.       10.                                          0
  RAVBB X0064B       -1.       10.                                          0
  RAVBC X0064C       -1.       10.                                          0
  X0083AX0069A       -1.        1.                                          0
  X0083BX0069B       -1.        1.                                          0
  X0083CX0069C       -1.        1.                                          0
  X0083AX0079A       -1.       10.                                          0
  X0083BX0079B       -1.       10.                                          0
  X0083CX0079C       -1.       10.                                          0
  X0095AGRCBA        -1.       10.                                          0
  X0095BGRCBB        -1.       10.                                          0
  X0095CGRCBC        -1.       10.                                          0
  CR20A           .01998       10.                                          0
```

```
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1    ><    T1    >< TSTART >< TSTOP  >
14SRC1A  0   440000.       60.       -20.                                -1.       10.
14SRC1C  0   440000.       60.      -140.                                -1.       10.
14SRC1B  0   440000.       60.       100.                                -1.       10.
14SRC2A  0   440000.       60.                                           -1.       10.
14SRC2C  0   440000.       60.      -120.                                -1.       10.
14SRC2B  0   440000.       60.       120.                                -1.       10.
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
BLANK OUTPUT
  PRINTER PLOT
 194 5. 0.0 40.         CR25A CR30A
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

5.7 Lightning overvoltage studies in a 400 kV substation *(Exa_9.cir)*

This example demonstrates how to use ATPDraw effectively in a substation lightning protection study. The one-line diagram of the investigated 400 kV substation is drawn in Fig. 5.34.

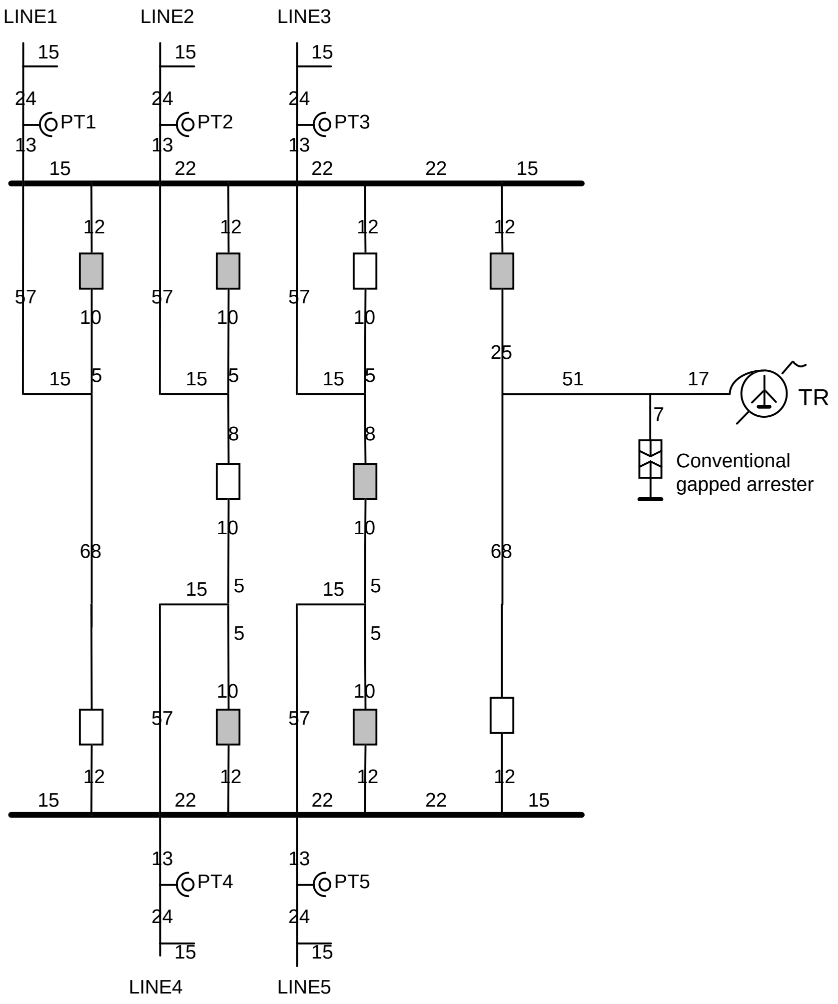


Fig. 5.34 - The one-line diagram of the substation

The figures written to the buses give the length of each section in meters. The circuit breakers drawn as empty boxes are assumed to be disconnected, so in that configuration only two transmission lines are connected to the transformer which is protected by a conventional, gapped arrester. The simulated incident is a single phase backflashover arising 1.2 km away from the substation which was assumed to be caused by a 100 kAmps 2/50 μs direct lightning to the ground wire. The ATPDraw circuit of the complete network (substation+incoming line) is shown in Fig. 5.35.

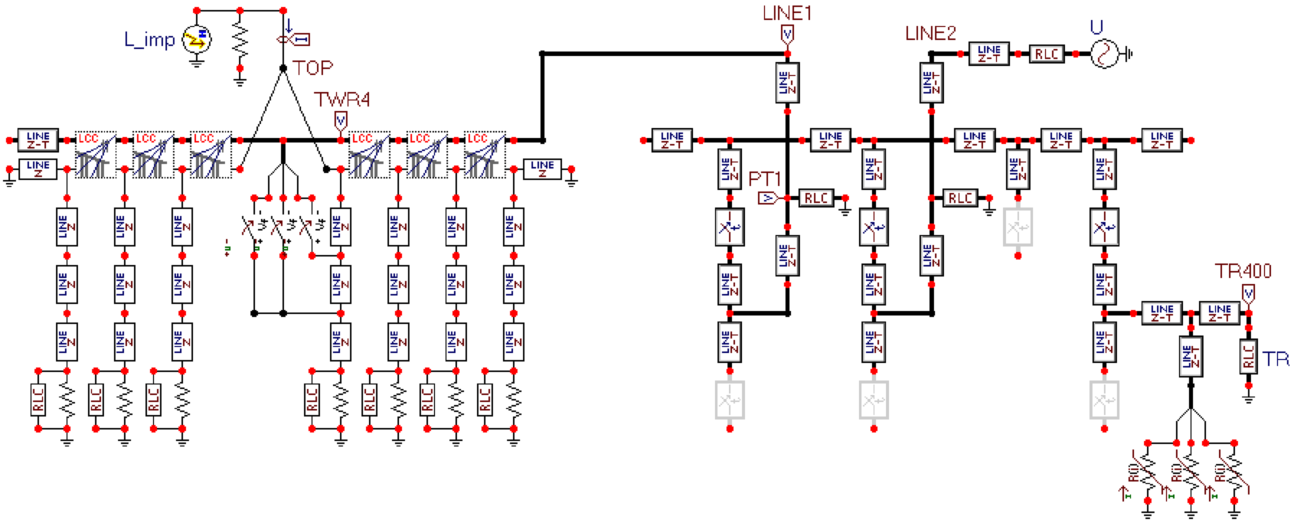


Fig. 5.35 - Example circuit (`EXA_9.cir`)

The single circuit overhead line spans in the vicinity of the lightning stroke have been represented by four-wire JMarti lines. The surge propagation along the transmission tower and the response of the tower grounding have been approximated by a single-phase constant parameter transmission line and lumped R-L branches. The JMarti line was included in ATPDraw via the Overhead Line (PCH) objects again, as it was shown earlier. The library file (`C_400kV.LIB`) created this way can be found in the `\USP` folder

As the previous figure shows, the model contains many identical blocks, so Copy&Paste support of ATPDraw can be exploited effectively in such kind of simulations. You need to define the object parameters only once and copy them as many times as needed.

The power frequency voltage is assumed to be on the maximum with opposite polarity at the moment of lightning on the faulty phase. So the backflashover appears when the voltage stress exceeds the flashover voltage of the voltage dependent switches simulating the insulator gaps.

All substation components were represented by three-phase objects except the conventional arresters which were simulated by single-phase nonlinear resistors with sparkover voltage, since no such three phase components are available in ATPDraw. The nonlinear characteristics of the device is shown in Fig. 5.36. The voltage and current values are given here in kilovolts and kAmps, so the amplitude of the power frequency source and the lightning current also have to be scaled accordingly.

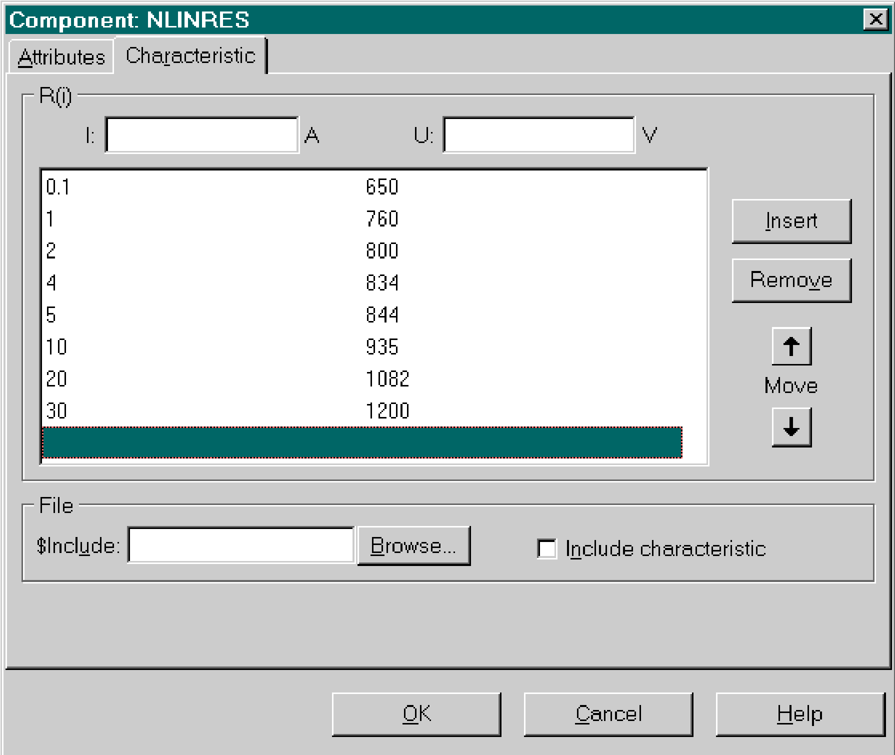


Fig. 5.36 - Nonlinear characteristics of the SiC arrester

The ATP file created by ATPDraw is printed below. It must be noted that this case does run only if you increase the DEAFULT parameter in the `LISTSIZE.DAT` file of ATP from 3.0 to 6.0:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW Tue, 30 Jun - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
$PREFIX,C:\ATPDRAW\USP\
$SUFFIX, .LIB
$DUMMY, XYZ000
C Miscellaneous Data Card ....
C  dT  >< Tmax >< Xopt >< Copt >
2.000E-8  .00003
     500       3       0       0       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
-1XX0014XX0010               .01  200.2.50E8    8. 1
-1XX0016XX0014               .01  200.2.50E8    7. 1
-1XX0018XX0016               .01  200.2.50E8   18. 1
        XX0021               40.                                               0
-1      XX0023               .02  600.2.90E8  400. 1
        XX0021               13.  .005                                         0
-1XX0028XX0012               .01  200.2.50E8    8. 1
-1XX0030XX0028               .01  200.2.50E8    7. 1
-1XX0032XX0030               .01  200.2.50E8   18. 1
        XX0035               40.                                               0
-1X0038AX0039A               .02  650.2.40E8 3000. 1 00
-2X0038BX0039B              .002  400.2.90E8 3000. 1 00
-3X0038CX0039C
        XX0035               13.  .005                                         0
-1XX0046                     .02  600.2.90E8  400. 1
-1XX0060XX0045               .01  200.2.50E8    8. 1
```

```
-1XX0068XX0060               .01  200.2.50E8    7. 1
-1XX0070XX0068               .01  200.2.50E8   18. 1
        XX0070               40.                                               0
-1XX0081XX0055               .01  200.2.50E8    8. 1
-1XX0083XX0081               .01  200.2.50E8    7. 1
-1XX0021XX0083               .01  200.2.50E8   18. 1
        XX0018               40.                                               0
        XX0036              400.                                               0
-1XX0091XX0046               .01  200.2.50E8    8. 1
        XX0018               13.   .005                                        0
-1XX0095XX0091               .01  200.2.50E8    7. 1
        XX0070               13.   .005                                        0
-1XX0100XX0023               .01  200.2.50E8    8. 1
-1XX0035XX0095               .01  200.2.50E8   18. 1
-1X0049AX0125A               20.  400.2.40E5  .008 1 00
-2X0049BX0125B                2.  260.2.90E5  .008 1 00
-3X0049CX0125C
-1XX0066TOP                  .01  200.2.50E8    8. 1
-1X0063AX0110A               20.  400.2.40E5  .012 1 00
-2X0063BX0110B                2.  260.2.90E5  .012 1 00
-3X0063CX0110C
-1X0132AX0112A               50.  650.2.40E5  .015 1 00
-2X0132BX0112B               10.  360.2.90E5  .015 1 00
-3X0132CX0112C
-1X0051AX0168A               20.  400.2.40E5  .068 1 00
-2X0051BX0168B                2.  260.2.90E5  .068 1 00
-3X0051CX0168C
-1X0115ALINE2A               20.  650.2.40E5  .024 1 00
-2X0115BLINE2B                2.  360.2.90E5  .024 1 00
-3X0115CLINE2C
-1X0117AX0118A               20.  400.2.40E5  .012 1 00
-2X0117BX0118B                2.  260.2.90E5  .012 1 00
-3X0117CX0118C
-1X0119AX0120A               20.  650.2.40E5  .015 1 00
-2X0119BX0120B                2.  360.2.90E5  .015 1 00
-3X0119CX0120C
-1X0125AX0121A               20.  400.2.40E5  .015 1 00
-2X0125BX0121B                2.  260.2.90E5  .015 1 00
-3X0125CX0121C
-1X0125AX0115A               20.  400.2.40E5  .085 1 00
-2X0125BX0115B                2.  260.2.90E5  .085 1 00
-3X0125CX0115C
  X0115A                               .0005                                   0
  X0115B                               .0005                                   0
  X0115C                               .0005                                   0
-1X0118AX0110A               20.  650.2.40E5  .022 1 00
-2X0118BX0110B                2.  360.2.90E5  .022 1 00
-3X0118CX0110C
-1X0110AX0132A               20.  650.2.40E5  .022 1 00
-2X0110BX0132B                2.  360.2.90E5  .022 1 00
-3X0110CX0132C
/BRANCH
99      SICC               1100.    1.                                         1
 1.0000000E-0001 6.5000000E+0002
 1.0000000E+0000 7.6000000E+0002
 2.0000000E+0000 8.0000000E+0002
 4.0000000E+0000 8.3400000E+0002
 5.0000000E+0000 8.4400000E+0002
 1.0000000E+0001 9.3500000E+0002
 2.0000000E+0001 1.0820000E+0003
 3.0000000E+0001 1.2000000E+0003
            9999
-1X0025AX0136A               20.  400.2.40E5  .068 1 00
-2X0025BX0136B                2.  260.2.90E5  .068 1 00
-3X0025CX0136C
-1X0137AX0132A               20.  400.2.40E5  .012 1 00
-2X0137BX0132B                2.  260.2.90E5  .012 1 00
-3X0137CX0132C
-1X0139ATR400A               20.  650.2.40E5  .017 1 00
-2X0139BTR400B                2.  360.2.90E5  .017 1 00
-3X0139CTR400C
```

```
-1X0136AX0141A              20.  400.2.40E5  .025 1 00
-2X0136BX0141B               2.  260.2.90E5  .025 1 00
-3X0136CX0141C
/BRANCH
99       SICB             1100.    1.                                        1
 1.0000000E-0001 6.5000000E+0002
 1.0000000E+0000 7.6000000E+0002
 2.0000000E+0000 8.0000000E+0002
 4.0000000E+0000 8.3400000E+0002
 5.0000000E+0000 8.4400000E+0002
 1.0000000E+0001 9.3500000E+0002
 2.0000000E+0001 1.0820000E+0003
 3.0000000E+0001 1.2000000E+0003
            9999
-1PT1A   LINE1A             20.  650.2.40E5  .024 1 00
-2PT1B   LINE1B              2.  360.2.90E5  .024 1 00
-3PT1C   LINE1C
-1X0149AX0120A              20.  400.2.40E5  .012 1 00
-2X0149BX0120B               2.  260.2.90E5  .012 1 00
-3X0149CX0120C
-1X0168AX0151A              20.  400.2.40E5  .015 1 00
-2X0168BX0151B               2.  260.2.90E5  .015 1 00
-3X0168CX0151C
         TR400A                        .003                                  0
         TR400B                        .003                                  0
         TR400C                        .003                                  0
-1X0136AX0139A              20.  650.2.40E5  .051 1 00
-2X0136BX0139B               2.  360.2.90E5  .051 1 00
-3X0136CX0139C
-1SICA   X0139A             20.  400.2.40E5  .007 1 00
-2SICB   X0139B              2.  260.2.90E5  .007 1 00
-3SICC   X0139C
/BRANCH
99       SICA             1100.    1.                                        1
 1.0000000E-0001 6.5000000E+0002
 1.0000000E+0000 7.6000000E+0002
 2.0000000E+0000 8.0000000E+0002
 4.0000000E+0000 8.3400000E+0002
 5.0000000E+0000 8.4400000E+0002
 1.0000000E+0001 9.3500000E+0002
 2.0000000E+0001 1.0820000E+0003
 3.0000000E+0001 1.2000000E+0003
            9999
  X0163AX0164A               1.   50.                                        0
  X0163BX0164B               1.   50.                                        0
  X0163CX0164C               1.   50.                                        0
-1XX0078XX0066              .01  200.2.50E8    7. 1
-1X0168APT1A                20.  400.2.40E5  .085 1 00
-2X0168BPT1B                 2.  260.2.90E5  .085 1 00
-3X0168CPT1C
  PT1A                                .0005                                  0
  PT1B                                .0005                                  0
  PT1C                                .0005                                  0
-1X0120AX0118A              20.  650.2.40E5  .022 1 00
-2X0120BX0118B               2.  360.2.90E5  .022 1 00
-3X0120CX0118C
-1XX0178XX0078              .01  200.2.50E8   18. 1
         XX0032             40.                                              0
         XX0032             13.  .005                                        0
-1XX0184XX0100              .01  200.2.50E8    7. 1
-1XX0186XX0184              .01  200.2.50E8   18. 1
         XX0186             40.                                              0
         XX0186             13.  .005                                        0
         XX0178             13.  .005                                        0
         XX0178             40.                                              0
-1LINE2AX0163A              20.  650.2.40E5    3. 1 00
-2LINE2BX0163B               2.  360.2.90E5    3. 1 00
-3LINE2CX0163C
C User specified object: C:\ATPDRAW\USP\LCC_N4.SUP
C    IN1 = X0009A
C IN2__A = XX0010
```

```
C    OUT1 = X0011A
C OUT2_A = XX0012
$INCLUDE, C_400KV, X0009A, X0009B, X0009C, XX0010, X0011A, X0011B, X0011C $$
  , XX0012
C User specified object: C:\ATPDRAW\USP\LCC_N4.SUP
C     IN1 = TWR4A
C IN2__A = TOP
C    OUT1 = X0044A
C OUT2_A = XX0045
$INCLUDE, C_400KV, TWR4A#, TWR4B#, TWR4C#, TOP###, X0044A, X0044B, X0044C $$
  , XX0045
C User specified object: C:\ATPDRAW\USP\LCC_N4.SUP
C     IN1 = X0044A
C IN2__A = XX0045
C    OUT1 = X0054A
C OUT2_A = XX0055
$INCLUDE, C_400KV, X0044A, X0044B, X0044C, XX0045, X0054A, X0054B, X0054C $$
  , XX0055
C User specified object: C:\ATPDRAW\USP\LCC_N4.SUP
C     IN1 = X0054A
C IN2__A = XX0055
C    OUT1 = LINE1A
C OUT2_A = XX0046
$INCLUDE, C_400KV, X0054A, X0054B, X0054C, XX0055, LINE1A, LINE1B, LINE1C $$
  , XX0046
C User specified object: C:\ATPDRAW\USP\LCC_N4.SUP
C     IN1 = X0011A
C IN2__A = XX0012
C    OUT1 = TWR4A
C OUT2_A = TOP
$INCLUDE, C_400KV, X0011A, X0011B, X0011C, XX0012, TWR4A#, TWR4B#, TWR4C# $$
  , TOP###
C JMarti
C User specified object: C:\ATPDRAW\USP\LCC_N4.SUP
C     IN1 = X0039A
C IN2__A = XX0023
C    OUT1 = X0009A
C OUT2_A = XX0010
$INCLUDE, C_400KV, X0039A, X0039B, X0039C, XX0023, X0009A, X0009B, X0009C $$
  , XX0010
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
  XX0036TOP                                          MEASURING                1
  XX0078TWR4C                   1.               1870.                        2
  XX0066TWR4A                   1.               1870.                        2
  XX0078TWR4B                   1.               1870.                        2
  X0121AX0117A        -1.      1.001                                          0
  X0121BX0117B        -1.      1.001                                          0
  X0121CX0117C        -1.      1.001                                          0
  X0141AX0137A        -1.      1.001                                          0
  X0141BX0137B        -1.      1.001                                          0
  X0141CX0137C        -1.      1.001                                          0
  X0151AX0149A        -1.      1.001                                          0
  X0151BX0149B        -1.      1.001                                          0
  X0151CX0149C        -1.      1.001                                          0
/SOURCE
C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
15XX0036-1      100.   .000002   .00005        5.              .000002        1.
14X0164A 0     -330.       50.                                     -1.        1.
14X0164B 0     -330.       50.      -120.                          -1.        1.
14X0164C 0     -330.       50.       120.                          -1.        1.
BLANK BRANCH
BLANK SWITCH
BLANK SOURCE
  LINE1ALINE1BLINE1CTWR4A TWR4B TWR4C TR400ATR400BTR400CPT1A  PT1B  PT1C
BLANK OUTPUT
BLANK PLOT
BEGIN NEW DATA CASE
BLANK
```

In Fig. 5.37, results from the simulation are displayed. The blue (thick) line shows the voltage at the transformer terminal, while the (red) thin lines are the incoming voltage measured at the substation entrance and the discharge current of the arrester.

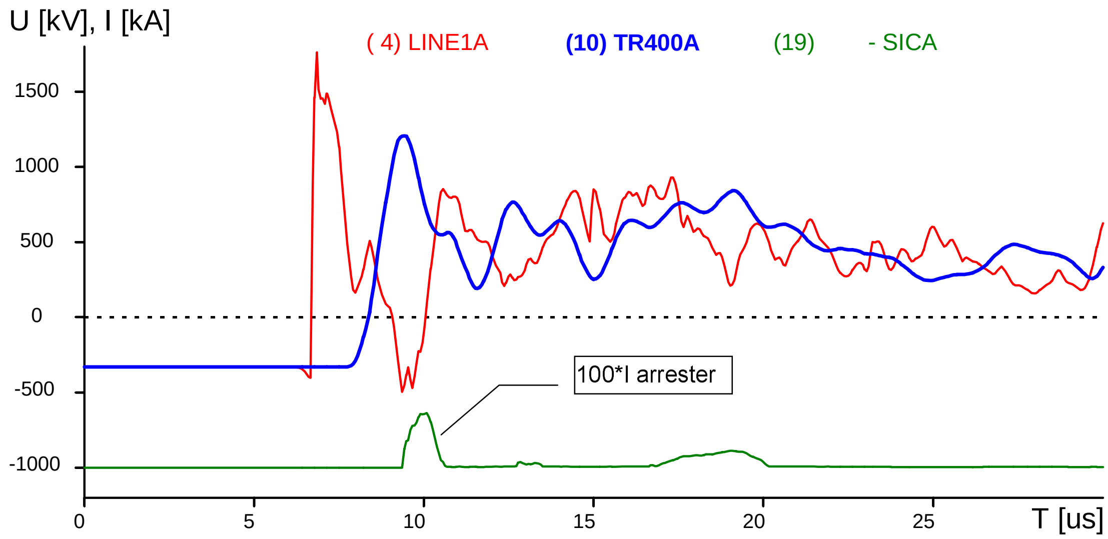


Fig. 5.37 - ATP simulation results

5.8 Simulating transformer inrush currents

This example shows how to create an ATP input file which can be used in development of controlled switching to reduce the inrush current of large power transformers substantially.

In such kind of studies the magnetic coupling between transformer windings and the nonlinear behavior of the magnetizing reactance are the main influencing parameters, so they must be taken into account precisely. The three-winding, three-phase, low reluctance transformer investigated in this study has been simulated by a BCTRAN model completed with three hysteretic, nonlinear inductances. Since neither of these ATP components are supported as standard ATPDraw objects, a new user specified component has been developed for the BCTRAN model and a special use of the standard Type-93 nonlinear inductance object has been applied to represent the hysteresis loop of the core material.

This chapter of the manual explains how to include the BCTRAN model into your ATPDraw circuit and finally gives a practical example (*Exa_10.cir*) of the usage of the new object.

5.8.1 Creating a new user specified BCTRAN object

Supporting routine BCTRAN can be used to derive a linear representation for a single or 3-phase two-, three or multiple winding transformers, using test data of both the excitation test and the short circuit test. The nonlinear behavior can not be included in the BCTRAN model, however adding Type-93 or Type-96 (saturation or hysteresis) elements, connected to windings closest to the core, the nonlinearities can be taken into account.

The BCTRAN model is based on test data can be obtained from the transformer manufacturers:

Voltage rating $V_{high}/V_{low}/V_{tertiary}$:		400/132/18 kV
Winding connection:		Yyn0d11
Power rating:		250 MVA (75 MVA tertiary)
Excitation losses:		140 kW
Excitation current:		0.2 %
Short circuit losses:	High to Low:	710 kW
	High to Tertiary:	188 kW
	Low to Tertiary:	159 kW
Positive seq. reactances:	High to Low:	15 %
	High to Tertiary:	12.5%
	Low to Tertiary:	7.2%

The three phase, three winding auto-transformer has a tertiary delta winding, so neither the value of the zero sequence exciting current nor the value of the zero sequence excitation loss are critical (see the ATP Rule Book [3] chapter XIX-C for more details). Based on this data, the input file for the BCTRAN supporting routine requesting [A] -[R] output will be as shown below:

```
BEGIN NEW DATA CASE
ACCESS MODULE BCTRAN
$ERASE
 3       50.     0.060      250.       140.      0.060      250.       140. 0 3 3 0
  1    154.73            HVBUSALVBUSAHVBUSBLVBUSBHVBUSCLVBUSC
  2     76.21            LVBUSA      LVBUSB      LVBUSC
  3      18.0            TVBUSCTVBUSATVBUSATVBUSBTVBUSBTVBUSC
 1 2      710.      15.0      250.      15.0      250. 0 1
 1 3      188.      12.5       75.      12.5       75. 0 1
 2 3      159.       7.2       75.       7.2       75. 0 1
BLANK
$PUNCH
BLANK
BEGIN NEW DATA CASE
BLANK
BLANK
```

Because the magnetizing inductance will be added to the model as an external element, only the resistive component of the excitation current has been inputted here, otherwise the magnetizing inductance would be taken into account twice.

5.8.2 Creating a Data Base Module file for BCTRAN

The first step is to create a DBM file which is an ATP file for the specific circuit along with a header explaining variables in the DBM file. The ATP Rule Book [3] chapter XIX-F explains in detail how to create such a file.

Since only 3x3 node names will be sent to the module from the outside, the header of the DBM file shows only `ARG` parameters following the `DATA BASE MODULE` request card. These variables are used in the final ATP files as the parameters of the $INCLUDE expression. The arguments to this procedure are:

`HVBUSA, HVBUSB, HVBUSC` :	The 3-phase node of the high voltage terminal
`LVBUSA, LVBUSB, LVBUSC` :	The 3-phase node of the low voltage terminal
`TVBUSA, TVBUSB, TVBUSC` :	The 3-phase node of the tertiary terminal

The closing and header section are the same as in any other Data Base Module cases. The rest of the file describes the transformer model as produced by the BCTRAN supporting routine of ATP. The structure of the DBM input file is shown below:

```
BEGIN NEW DATA CASE --NOSORT--
DATA BASE MODULE
$ERASE
ARG,HVBUSA,HVBUSB,HVBUSC,LVBUSA,LVBUSB,LVBUSC,TVBUSA,TVBUSB,TVBUSC
..
<<<< the .PCH file generated by the >>>>
<<<< BCTRAN supporting routine must >>>>
<<<< be inserted here               >>>>
..
BEGIN NEW DATA CASE
C
$PUNCH
BEGIN NEW DATA CASE
BLANK
```

Running this DBM module file through ATP will produce a `.PCH` punch file listed here:

```
KARD  3  3  4  4  6  6 10 10 11 13 13 16 16 20 25 25 31 31 38 46 46
KARG  7  9  7  8  8  9  1  4  4  7  9  2  5  5  7  8  3  6  6  8  9
KBEG  9  3  3  9  3  9  3  9  3  9  3  3  9  3  3  9  3  9  3  3  9
KEND 14  8  8 14  8 14  8 14  8 14  8  8 14  8  8 14  8 14  8  8 14
KTEX  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
$ERASE
C  <++++++>  Cards punched by support routine on  06-Sep-98  20.02.21  <++++++>
C ACCESS MODULE BCTRAN
C $ERASE
C  3       50.     0.060      250.      140.     0.060      250.      140. 0 3 3
C    1    154.73           HVBUSALVBUSAHVBUSBLVBUSBHVBUSCLVBUSC
C    2     76.21           LVBUSA      LVBUSB      LVBUSC
C    3      18.0           TVBUSCTVBUSATVBUSATVBUSBTVBUSBTVBUSC
C  1 2      710.      15.0      250.      15.0      250. 0 1
C  1 3      188.      12.5       75.      12.5       75. 0 1
C  2 3      159.       7.2       75.       7.2       75. 0 1
C BLANK
$VINTAGE, 1,
 1TVBUSCTVBUSA             6942.8363020897
 2TVBUSATVBUSB                         0.0
                           6942.8363020897
 3TVBUSBTVBUSC                         0.0
                                       0.0
                           6942.8363020897

  USE AR
 1HVBUSALVBUSA              7.327656539452 .59310761130933
 2LVBUSA                   -15.70576584821             0.0
                           52.452809134348 .05405278589067
 3TVBUSCTVBUSA             3.5071177190576             0.0
                           -87.07085746858             0.0
                           338.51816785593       .02445984
 4HVBUSBLVBUSB                         0.0             0.0
                                       0.0             0.0
                                       0.0             0.0
                            7.327656539452 .59310761130933
 5LVBUSB                               0.0             0.0
                                       0.0             0.0
                                       0.0             0.0
                           -15.70576584821             0.0
                           52.452809134348 .05405278589067
 6TVBUSATVBUSB                         0.0             0.0
                                       0.0             0.0
                            -.22427138E-17             0.0
                           3.5071177190576             0.0
                           -87.07085746858             0.0
                           338.51816785593       .02445984
```

```
   7HVBUSCLVBUSC                                      0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                           7.327656539452  .59310761130933
   8LVBUSC                                            0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                          -15.70576584821             0.0
                                          52.452809134348  .05405278589067
   9TVBUSBTVBUSC                                      0.0             0.0
                                                      0.0             0.0
                                           -.22427138E-17             0.0
                                                      0.0             0.0
                                                      0.0             0.0
                                           -.22427138E-17             0.0
                                          3.5071177190576             0.0
                                          -87.07085746858             0.0
                                          338.51816785593       .02445984
  $VINTAGE, 0,
  $UNITS, -1.,-1.
    USE RL
  C ----------------- << case separator >>> -----------
  $EOF   User-supplied header cards follow.         07-Sep-98  17.19.10
  ARG,HVBUSA,HVBUSB,HVBUSC,LVBUSA,LVBUSB,LVBUSC,TVBUSA,TVBUSB,TVBUSC
```

This file is very similar to the DBM input file, but with a different header and with the DBM file header given at the bottom instead. This file will be $Included in the ATP file by ATPDraw. It is suggested to give this file a name with extension `.LIB` and store it in the `\USP` directory. The name `LINTR400.LIB` is used in this example.

5.8.3 Creating support file

When the library file from the DBM file has been created, the next step is to create a *New User Specified* object. So start up ATPDraw and enter the *Objects* field in the main menu. The process of creating a new object consists of two steps: creating parameter support and creating an icon.

First select the *New User Specified* field in the popup menu. You must specify the number of data and nodes of the new object. Since no `NUM` card exists in the DBM header, the *Number of data* is 0 in this example. The *Number of nodes* is set to 3 as shown in Fig. 5.38.

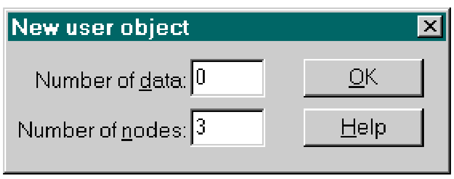
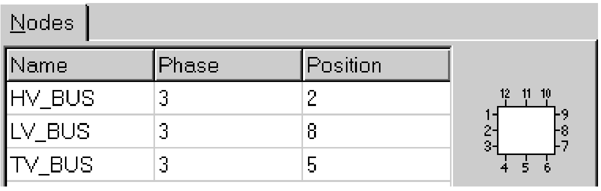


Fig. 5.38 - Size of the new BCTRAN object

When the user has determined the size of the new object and clicks *OK*, a notebook-style dialog box appears. In this case the *Data* tab does not exist, so only the *Node* settings need to be specified.



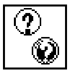
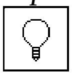
In this window you must specify a name of each nodes in the *Name* field, the number of *Phases* (1 or 3) and the node position on the icon border. Codes for the available node positions are shown in the icon at right in Fig. 5.39.

Fig. 5.39 - Node window of the new object

The names of the nodes do not have to be the same as the names used in the DBM punch file, but the sequence of the nodes must be the same as used in the `ARG` card.

5.8.4 Creating icon

Each user specified component can have an icon which represents the object on the screen and an optional on-line help which describes the meaning of the parameters. These properties can be edited using the built in *Help* and *Icon Editors.*

The  button and the  button at the right hand side of the *New User Specified* dialog box is used to call the help and the icon editor. All the functions and menus of the editor are described in the Reference part, so this information is not repeated here. The icon specified as the graphical representation of the new BCTRAN object on the screen is shown in Fig. 5.40.

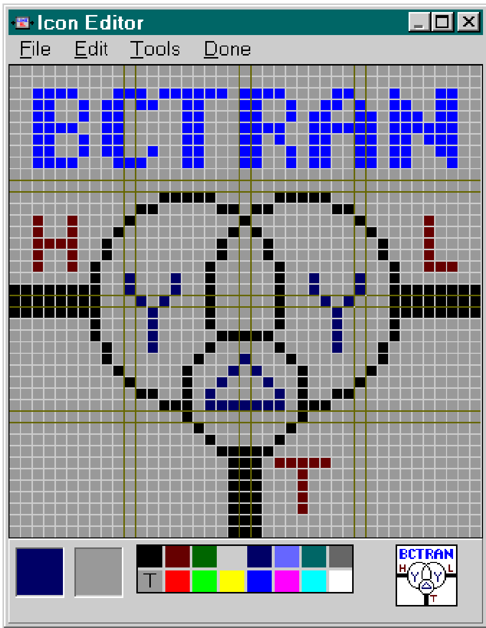


Fig. 5.40 - The icon specified for the BCTRAN object

When you have finished the icon drawing, click on the *Done* button. Finally, after clicking the *Save* or *Save As* buttons, the new support file will be saved to disk. The file name can be specified in a standard *Save...* dialog box. Support files for user specified components are normally located in the `\USP` folder and their extension must be `.SUP`. The file name may be different from the DBM file name. The new object is found under *User Specified* | *Files* in the component selection menu. If you select `LINTR400.SUP` in the appearing file window, the new object is drawn in the circuit window and can be used and edited like any other objects.

5.8.5 Example circuit. Transformer energization *(Exa_10.cir)*

The application of the new BCTRAN object is shown in this example. The study case is controlled energization of an unloaded transformer. Controlled switching is the most effective way of avoiding high inrush currents, which may stress the transformer windings dynamically. A three-phase, three-winding Yyd connected transformer is switched from the earthed star connected 400 kV side, while the 132 kV and the 18 kV delta connected buses were unloaded. The single line diagram of the simulated system is shown in Fig. 5.41.

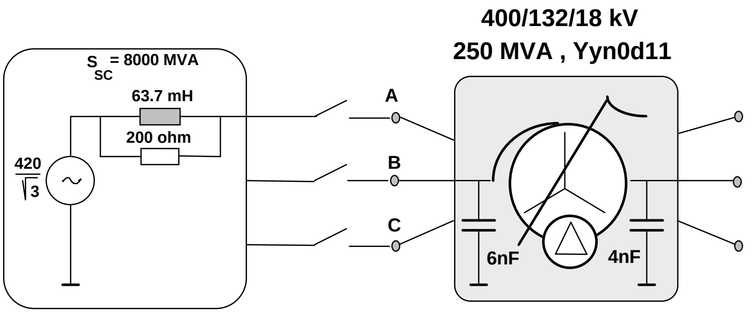


Fig. 5.41 - One-line scheme of the simulated case

The ATPDraw circuit of the study case is shown in Fig. 5.42.

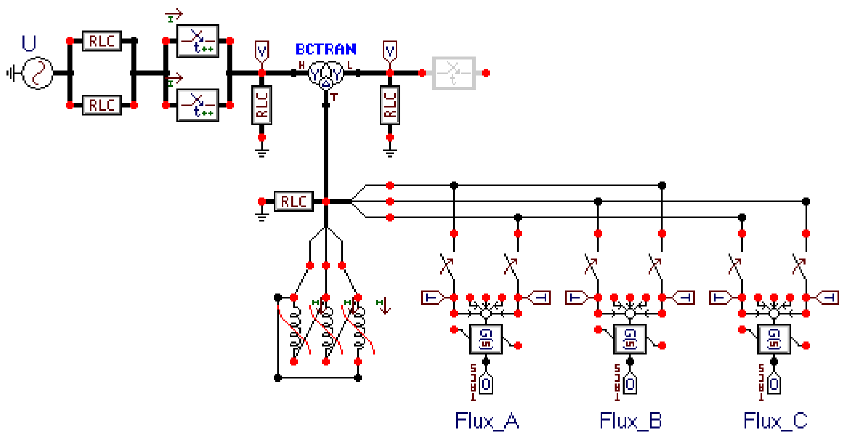


Fig. 5.42 - Transformer energization (`EXA_10.CIR`)

The 400 kV supply is represented by lumped RLC branches. The external library file (`LINTR400.LIB`) created using Data Base Module feature of ATP is $Included in this example. The 3-phase, parallel connected time controlled switches are applied to simulate the “steady on - off-on” operating cycle of the circuit breaker. The TACS transfer function objects are used as flux sensors in this example. It must be noted that such a “generalized” transfer objects are available only in version 1.2 and above of ATPDraw. So, if you have and older version of the program, you must replace them by the older G(s) blocks.

The magnetizing branch is represented by a hysteretic nonlinear inductor model generated by the HYSDAT supporting routine of ATP. At present ATP supports only one type (Armco M4) of magnetic material, but it could be used as good enough approximation for other oriented silicon steel materials, too. Fig. 5.43 shows the hysteresis loop of the Type-1 material of ATP and of the magnetic core of the transformer investigated in this study.

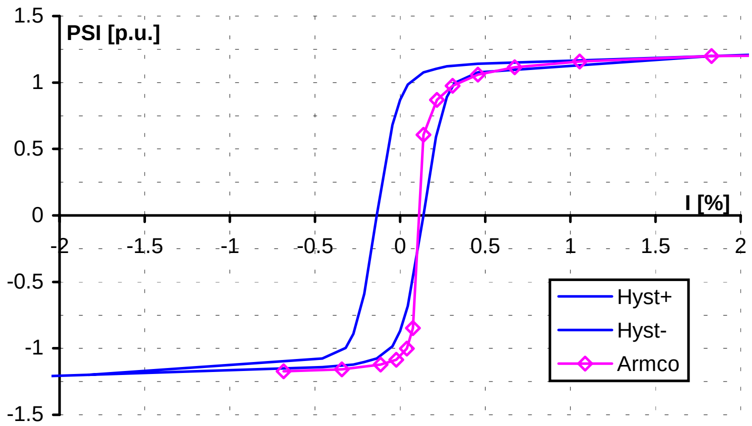


Fig. 5.43 - The shape of the hysteresis loop of the transformer

The output file generated by the HYSDAT supporting routine is listed below. In this example the file is given a name `HYSTR400.LIB` and stored in the `/USP` folder.

```
C  <++++++>  Cards punched by support routine on  06-Sep-98  18.19.25  <++++++>
C HYSTERESIS
C $ERASE
C C   ITYPE    LEVEL    { Request Armco M4 oriented silicon steel -- only 1 availab
C         1        4    { That was ITYPE=1.  As for LEVEL=2, moderate accuracy outp
C      98.2     97.2    { Current and flux coordinates of positive saturat
 -3.68250000E+01 -9.49129412E+01
 -2.45500000E+01 -9.43411765E+01
 -1.10475000E+01 -9.23400000E+01
 -4.91000000E+00 -9.03388235E+01
 -1.84125000E+00 -8.86235294E+01
  6.13750000E-01 -8.51929412E+01
  2.14812500E+00 -8.11905882E+01
  3.55975000E+00 -7.43294118E+01
  4.29625000E+00 -6.28941176E+01
  4.91000000E+00 -4.57411765E+01
  6.13750000E+00  3.05894118E+01
  6.75125000E+00  4.23105882E+01
  8.59250000E+00  5.71764706E+01
  1.10475000E+01  6.86117647E+01
  1.33797500E+01  7.43294118E+01
  1.74918750E+01  8.00470588E+01
  2.39362500E+01  8.51929412E+01
  3.28356250E+01  8.91952941E+01
  4.29625000E+01  9.20541176E+01
  6.13750000E+01  9.49129412E+01
  9.82000000E+01  9.72000000E+01
  1.35025000E+02  9.77717647E+01
           9999.
```

In spite of the Type-96 inductor is not a native object of ATPDraw, it can be included by a special use of the Type-93 nonlinear inductor object. Since ATPDraw supports to include an external file, such as the hysteresis loop specified in the `HYSTR400.LIB`, the nonlinear flux-current characteristics for the Type-96 elements can be connected to a standard Type-93 object, too. When the .ATP has been created in the *ATP | Make File* procedure, the user must change the Type ID in column 1-2 from 93 to 96 manually as shown in the next page. It can be done easily using the *ATP | Edit File* feature of ATPDraw, and can be automated with the *Find&Replace* command of the Text editor.

The ATPDraw generated ATP file for this study case is listed on the next page:

```
BEGIN NEW DATA CASE
C --------------------------------------------------------
C Generated by ATPDRAW  Wed, 30 Sep - 1998
C A Bonneville Power Administration program
C Programmed by H. K. Høidalen at EFI - NORWAY  1994-1997
C --------------------------------------------------------
$PREFIX,C:\ATPDRAW\USP\
$SUFFIX, .LIB
$DUMMY, XYZ000
C Miscellaneous Data Card ....
C  dT  >< Tmax >< Xopt >< Copt >
 .000005     .15
     500       5       0       0       1       0       0       1       0
TACS HYBRID
/TACS
90XX0036                                                                      1.
90XX0028                                                                      1.
 1FLUXA   -XX0036                         +XX0028     1.
        1.
                  1.
90XX0051                                                                      1.
90XX0049                                                                      1.
 1FLUXB   -XX0051                         +XX0049     1.
        1.
                  1.
90XX0066                                                                      1.
90XX0064                                                                      1.
 1FLUXC   -XX0066                         +XX0064     1.
        1.
                  1.
33FLUXA
33FLUXB
33FLUXC
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
/BRANCH
96T_BUSBT_BUSA             8888.                                               1
$INCLUDE, HYSTR400
        L_BUSA                          .004                                   0
        L_BUSB                          .004                                   0
        L_BUSC                          .004                                   0
  SOURCASUPLA                 2.  63.7                                         0
  SOURCBSUPLB                 2.  63.7                                         0
  SOURCCSUPLC                 2.  63.7                                         0
  SOURCASUPLA               200.                                               0
  SOURCBSUPLB               200.                                               0
  SOURCCSUPLC               200.                                               0
        H_BUSA                          .006                                   0
        H_BUSB                          .006                                   0
        H_BUSC                          .006                                   0
        T_BUSA                           .01                                   0
        T_BUSB                           .01                                   0
        T_BUSC                           .01                                   0
/BRANCH
96T_BUSAT_BUSC             8888.                                               1
$INCLUDE, HYSTR400
/BRANCH
96T_BUSCT_BUSB             8888.                                               1
$INCLUDE, HYSTR400
C User specified object: C:\ATPDRAW\USP\LINTR400.SUP
C HV_BUS = H_BUSA
C LV_BUS = L_BUSA
C TV_BUS = T_BUSA
$INCLUDE, LINTR400, H_BUSA, H_BUSB, H_BUSC, L_BUSA, L_BUSB, L_BUSC, T_BUSA $$
  , T_BUSB, T_BUSC
/SWITCH
C < n 1>< n 2>< Tclose ><Top/Tde ><   Ie   ><Vf/CLOP ><  type  >
```

```
    SUPLA H_BUSA        -1.        .045        1.                                    1
    SUPLB H_BUSB        -1.        .045        1.                                    1
    SUPLC H_BUSC        -1.        .045        1.                                    1
    XX0028T_BUSA                     1.                                              0
    SUPLA H_BUSA       .0735         1.                                              1
    SUPLB H_BUSB       .0785         1.                                              1
    SUPLC H_BUSC       .0785         1.                                              1
    XX0036T_BUSC                     1.                                              0
    XX0049T_BUSB      .00666         1.                                              0
    XX0051T_BUSA      .00666         1.                                              0
    XX0064T_BUSC      .00333         1.                                              0
    XX0066T_BUSB      .00333         1.                                              0
  /SOURCE
  C < n 1><>< Ampl.  >< Freq.  ><Phase/T0><   A1   ><   T1   >< TSTART >< TSTOP  >
  14SOURCA 0    326600.       50.                                    -1.        1.
  14SOURCB 0    326600.       50.      -120.                         -1.        1.
  14SOURCC 0    326600.       50.       120.                         -1.        1.
  BLANK TACS
  BLANK BRANCH
  BLANK SWITCH
  BLANK SOURCE
    L_BUSAL_BUSBL_BUSCH_BUSAH_BUSBH_BUSC
  BLANK OUTPUT
  BLANK PLOT
  BEGIN NEW DATA CASE
  BLANK
```

The results of the simulation are shown in Fig. 5.44. Initially the upper switch is assumed to be disconnected, and the lower one connects the transformer to the source. Thus only a small magnetizing currents flow in the steady-state. Then the lower switch operates at 45 ms, and high remanent flux remains in two of the phases. The subsequent energization of the transformer could result in high inrush current as simulated in this case by closing the upper switches.

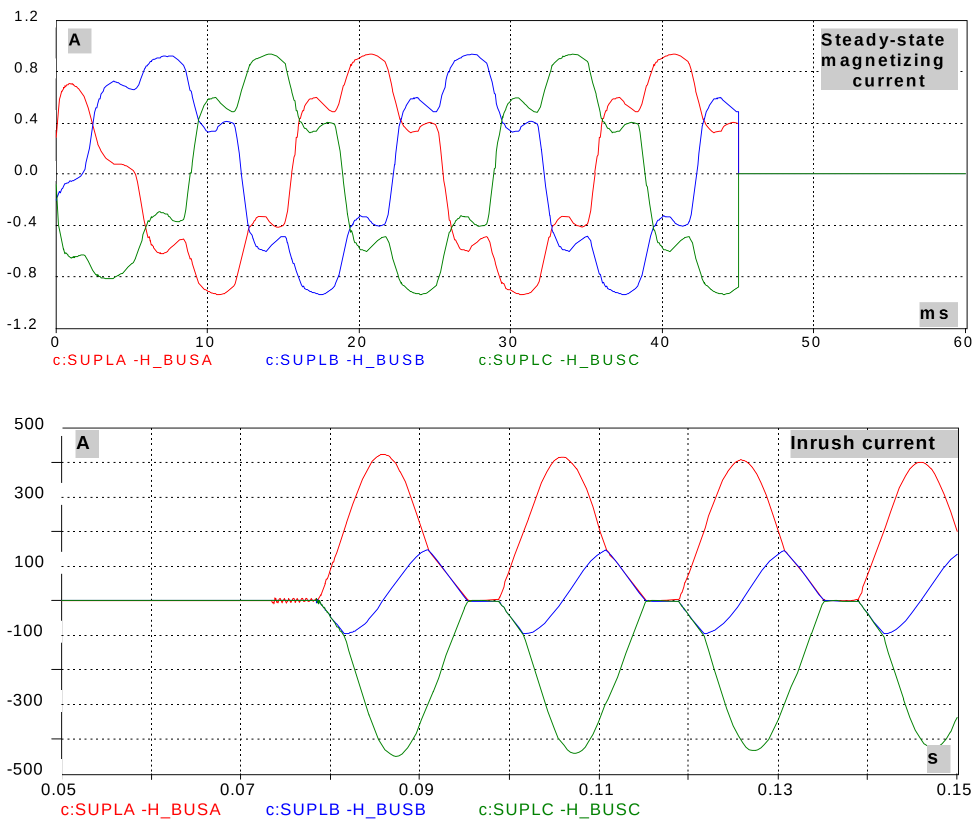


Fig. 5.44 - ATP simulation results

6. Line/Cable Manual . . .

. . . .

ATPDraw™
for Windows
1.0

This chapter is the manual for a new program called ATP_LCC, a mouse controlled preprocessor for the LINE CONSTANTS/CABLE CONSTANTS supporting routines of ATP. It was first issued with the ATPDraw for Windows version 1.0 and was not available in earlier versions. The ATP_LCC manual is included here as the last chapter of the ATPDraw user's guide because the purpose of ATP_LCC is very similar to that of the ATPDraw itself; i.e. to provide a graphical user interface. While the ATPDraw has the functionality to create ATP input files for the time and frequency domain simulations, ATP_LCC prepares such files for the line constant or cable constant calculations. ATP_LCC is written in Borland Delphi 1.0, which is a 16-bit compiler, so ATP_LCC is running on all MS-Windows compatible operating systems. This chapter includes installation manual as well as user's guide for ATP_LCC.

6.1 Introduction

In the ATP_LCC, the user specifies the geometrical and physical data for an overhead line or a cable system in input windows. The program consists of two parts; one for LINE CONSTANTS support and one for CABLE CONSTANTS. These two parts are handled independently in the program with separate input windows. The geometrical and electrical data of the system can be specified in tabbed notebook style dialog boxes, consisting of two pages for overhead lines and five pages for cables. These windows appear, when you select the *New Line* or *New Cable* option in the File menu, or when an existing line/cable data file is selected in the *Open* dialog of the File menu.

For later modification the input data can be saved to disk files with the default extension `.LIN` for lines and `.CBL` for cables.

The specified line and cable geometries are drawn in the main window. Zooming in and out operations are supported, as well as the copy of the bitmap of the conductor arrangement to the Windows clipboard.

Considering the user specified data, the corresponding ATP file is generated in the *ATP | Create data case* menu of the program. This file is ready to be processed by ATP to create punched output (`.PCH` file) or matrix output (`.LIS` file). The punch files created by ATP is in most cases readable directly by the ATPDraw for Windows via the last menu item on the component selection menu: *Overhead lines (PCH).*

The ATP_LCC program now supports the following cases for LINE CONSTANTS:

- Constant parameter (KCLee and Clack lines)
- JMarti lines
- PI-equivalents
- Single frequency output
- Logarithmic frequency output
- Mutual coupling output

and for the CABLE CONSTANTS:

- Overhead lines
- Single core cables
- Enclosing pipes

6.2 How to get the program?

ATP_LCC is available via FTP from an anonymous sever at `ftp.ee.mtu.edu` (IP 141.219.23.120) in the directory `/pub/atp/gui/atp_lcc`, or on the mirror servers operated in Europe and in Japan. The Username is: anonymous, the password is your full E-mail address. Download all files found in the `/atp_lcc` directory on the FTP sites. At present, you will find there a self extracting archive `LCC.EXE` and a readme file. Later, there might be some patch files, too.

Inexperienced Internet users can find a short guide to the ATP related Internet resources in section 2.10 of the Installation Manual.

6.3 How to install the program?

ATP_LCC does not need any special installation. You just simply need to copy the auto-decompressing archive `LCC.EXE` to the desired ATP_LCC directory. Then change to this directory and run `LCC.EXE`, which will decompress the archive. Delete the `LCC.EXE` file to free up some hard disk space as the final step of the installation. After running `LCC.EXE`, the following files should be there in your ATP_LCC directory:

```
.                <DIR>        98.08.13  13.43 .
..               <DIR>        98.08.13  13.43 ..
LCC       EXE        219 836  97.06.16  21.20 LCC.EXE
ATP_LCC   EXE        529 664  97.06.13  23.27 ATP_LCC.EXE
H_CABLE   TXT          9 604  97.06.07  16.15 H_CABLE.TXT
H_LINE    TXT          5 204  97.05.19  18.58 H_LINE.TXT
TEST      CBL          7 861  97.06.12  13.31 TEST.CBL
TEST      LIN            791  97.05.19  13.22 TEST.LIN
          7 file(s)         772 960 bytes
```

Creating a program icon for ATP_LCC slightly depends on the version of the operating system you are using, as shown below. You can create a separate program group for the ATP_LCC, but it is also practical to put its icon into your existing ATP or ATPDraw for Windows folder.

<u>Windows 3.1x/NT3.5x</u>

Select the *New* item in the Program Manager's File menu. Create a new group by selecting the *Program Group* radio button first, then press *OK*. The description field in the next window specifies the name of the program group you are creating. The *Group File* field is optional, so you can leave it empty. After pressing *OK*, the new program group is created with the specified name. Select *File* | *New* again, but now leave the *Program Item* radio button to be selected, then press *OK* and *Browse* the `ATP_LCC.EXE` in the appearing window. Finally press *OK* and the new icon must appear in the specified program group.

<u>Windows 95/NT4.0</u>

Select *Settings* | *Taskbar* option from the *Start* menu. Choose *Add* on the *Start Menu Programs* page and *Browse* the `ATP_LCC.EXE`. Select a folder in which you are going to place the ATP_LCC shortcut, or create a new folder by clicking the *New Folder* button. Finally press *Finish* and *OK*. The new icon must appear in the *Programs* folder of the Start menu.

6.4 The Main window and the Main menu

ATP_LCC has a standard Windows user interface. Fig. 6.1 shows the main window of the program. The *Main menu* of the program has four sub-menus: *File*, *Edit*, *ATP* and *Help*. The *Toolbar* below the main menu provides easy access to the most frequently used file (Open, Save, Edit) and bitmap (Zoom-to-fit, Copy to clipboard) operations. In the middle of the main window, the specified line and cable geometry is drawn. In that window zooming is supported by the Zoom-to-fit toolbar icon or by the *Zoom fit* command in the Edit menu. Zooming is also supported by a left or right mouse click on the desired position of the diagram. At the bottom of the main window a status bar is located with *Modified* field and menu option hints. When the modified field is visible, the line/cable data has been changed since the previous *Save* or *Save as* operation and must be saved to disk before quitting the program.

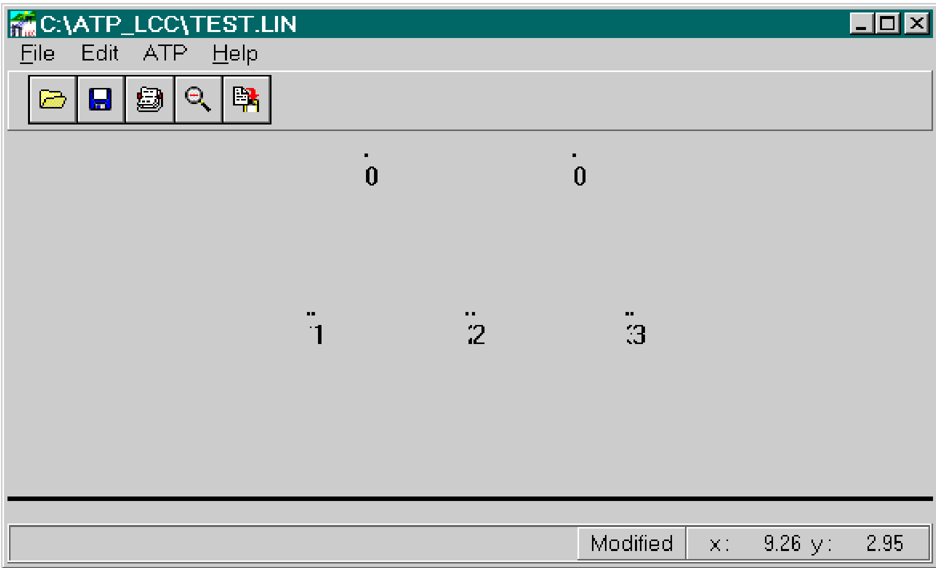


Fig. 6.1 - The main window of ATP_LCC

6.5 The File menu

This field contains actions for input/output of line/cable constant data cases. Selecting the File menu will result in a pop-up menu shown in Fig. 6.2.

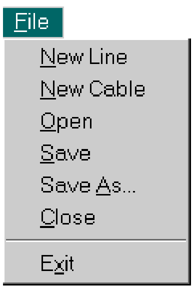


Fig. 6.2 - File menu commands in ATP_LCC

6.5.1 New Line

This selection starts a new line constant session. ATP_LCC supports working on either a Line Constant or on a Cable Constant case. When the *New Line* is selected, a tabbed notebook style dialog box appears where the electrical and material data of conductors, as well as the geometrical arrangement of the system can be specified. The structure of the line data windows and the interpretation of input field parameters are described in section 6.9 of this manual.

6.5.2 New Cable

Selecting this field starts a new cable constant session. ATP_LCC supports working on either a Line constant or on a cable constant case at a time. Following the *New Cable* selection, a tabbed notebook style dialog box appears where the electrical and material data of conductors, as well as the geometrical arrangement of the cable system can be specified. The structure of the cable data windows and the meaning of input field parameters are shown in section 6.10 of this manual.

6.5.3 Open

To open an existing line or cable data file, select the *Open* command. This will produce a standard Windows 3.1 style Open dialog box shown in Fig. 6.3, where you select a file to load. This command is also available via the toolbar icon:

The existing line and cable constant files in the \ATP_LCC folder are shown below the *File name:* field. Selection can be made either by typing the file name directly or by a left mouse click in the file list. Clicking *OK* will perform the selection made and the file is loaded. Clicking on *Cancel* will simply close the window.

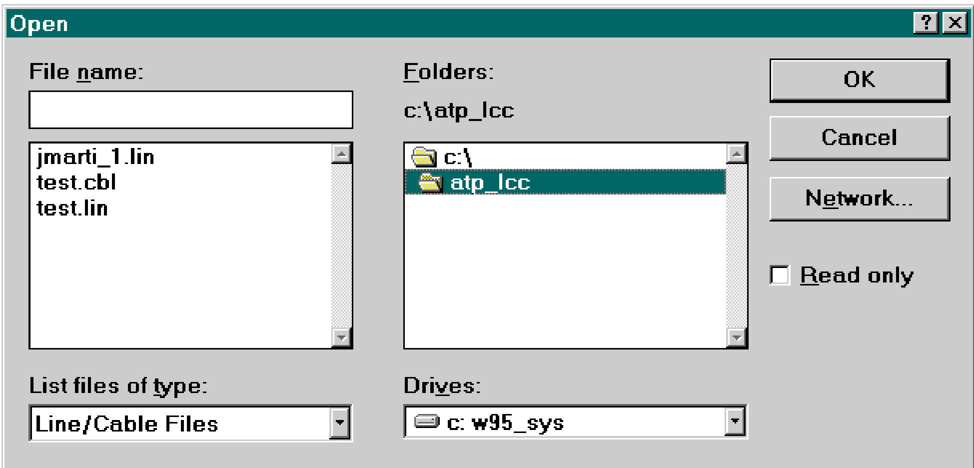


Fig. 6.3 - Opening an existing line or cable data file

6.5.4 Save

This command allows you to save the current line or cable constant data file to a disk file. This command is also available via the toolbar icon. The suggested extension for a line constant case is `.LIN`, and for a cable constant case is `.CBL`. If the name `noname.lin` or `noname.cbl` is shown in the Main window's header field, a Save As dialog box will be performed where the user must specify the name of the disk file.

6.5.5 Save As

If the actual line constant or cable constant case still does not have a name (`noname` can be seen in the header field), it will be requested in a *Save As* dialog box. Here the user must specify a file name for the current data case. This command allows you to save the current case under a name other than that is already used.

6.5.6 Close

The current line or cable constant case will be closed. The user will not be warned if the current case has not been saved yet, but the last session can be retrieved by clicking the Edit toolbar icon or by using the *Edit data* command in the *Edit* menu.

6.5.7 Exit

This command closes the ATP_LCC program. If any changes to the current session have not been saved yet, the user will be requested as shown in Fig. 6.4 to confirm before the application is terminated.

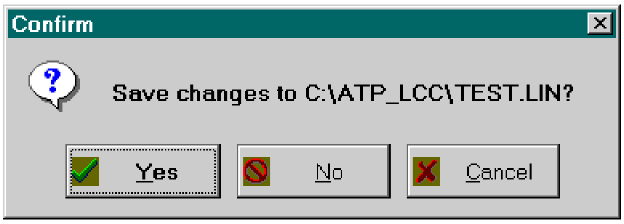


Fig. 6.4 - Warning message before terminating the program

6.6 The Edit menu

The Edit pop-up menu is shown in Fig. 6.5.

Fig. 6.5 - Edit menu commands

6.6.1 Edit data

This command displays either the Line or a Cable data window, where the line model (constant parameter, frequency dependent, etc.), the conductor geometry and the electrical data can be specified. This command is also available via the toolbar icon: .

All input data of the system must be specified in tabbed notebook style dialog boxes. This notebook has 2 tabs for a line constant case and 5 tabs for cable systems. Depending on the simulated cable system, some of the tabs might be inactive. I.e. if there is no enclosing pipe in your cable-system, the *Pipe data* tab will be inactive, so when you click on such a tab it will not come to the front and parameters can not be modified.

The structure of the line and cable data windows and the interpretation of input field parameters are described in the last two sections of this manual.

6.6.2 Zoom fit

The conductor or cable geometry of the system is drawn in the middle of the main window. The function of the *Zoom fit* command is to adjust the drawing to the actual size of the main window. The toolbar icon also executes this command.

Enlarging the drawing is also supported. To activate this feature, click inside the drawing with the left mouse button and the graphics of the line or cable system will be enlarged by one step around the mouse cursor location. If you need further enlargement, click the mouse again. The X and Y coordinates of the actual mouse cursor location are displayed and updated continuously in the lower-right corner of the main window, while you move the mouse.

6.6.3 Copy graphics

When this command is executed, the bitmap representation of the line or cable system drawn in the middle of the main window will be copied to the Windows clipboard. Thus the drawing in the main window can be pasted easily into other Windows application. The toolbar icon also performs this operation.

6.7 The ATP menu

The pull-down ATP menu has only two items: *Create data case* and *Edit data case*.

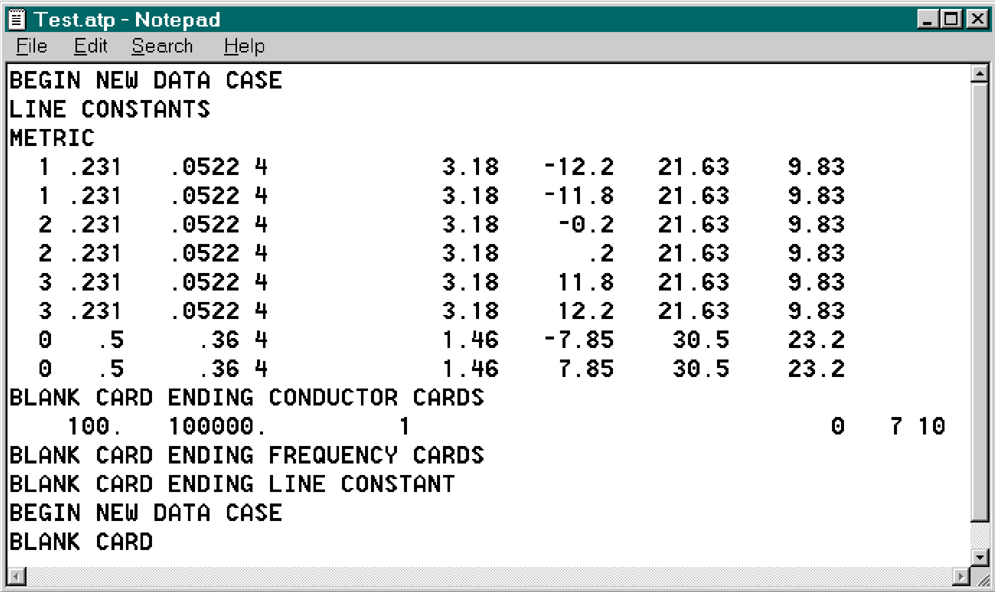
6.7.1 Create data case

Generates an ATP input file in correct format for the actual line or cable constant data case. The name of the disk file will be requested in a standard Save As dialog box. The default filename is the same as that of the current line or cable data file, completed with the extension .atp. When the ATP file is successfully created, it will be reported to the user in a message box such as shown in Fig. 6.6.

Fig. 6.6 - The ATP file is successfully generated

6.7.2 Edit data case

This selection opens the Notepad editor which enables you to contemplate or edit the ATP file. When the *Edit data case* is selected, a file having the same name as the active line or cable data file with the extension .atp is searched for and automatically opened as shown in Fig. 6.7.




```
BEGIN NEW DATA CASE
LINE CONSTANTS
METRIC
  1 .231    .0522 4            3.18   -12.2   21.63    9.83
  1 .231    .0522 4            3.18   -11.8   21.63    9.83
  2 .231    .0522 4            3.18    -0.2   21.63    9.83
  2 .231    .0522 4            3.18      .2   21.63    9.83
  3 .231    .0522 4            3.18    11.8   21.63    9.83
  3 .231    .0522 4            3.18    12.2   21.63    9.83
  0    .5     .36 4            1.46   -7.85    30.5    23.2
  0    .5     .36 4            1.46    7.85    30.5    23.2
BLANK CARD ENDING CONDUCTOR CARDS
    100.    100000.         1                              0    7 10
BLANK CARD ENDING FREQUENCY CARDS
BLANK CARD ENDING LINE CONSTANT
BEGIN NEW DATA CASE
BLANK CARD
```

Fig. 6.7 - Edit ATP file using the Notepad editor

Normally you do not need to use this option, because the ATP_LCC generates a correct ATP file. However not all the ATP options are supported by the graphical preprocessor (i.e. BRANCH request cards are not written to the .ATP file), so this *Edit data case* option is offered for experienced users, to edit the file manually.

6.8 The Help menu

At present, no extensive online help is available in the ATP_LCC. A short description for the input parameters is available however, selecting the *Help* buttons at the lower-right corner in the Line data and Cable data windows.

6.8.1 About

This selection displays an information window about the program developer, the sponsoring institution, as well as the version of the program.

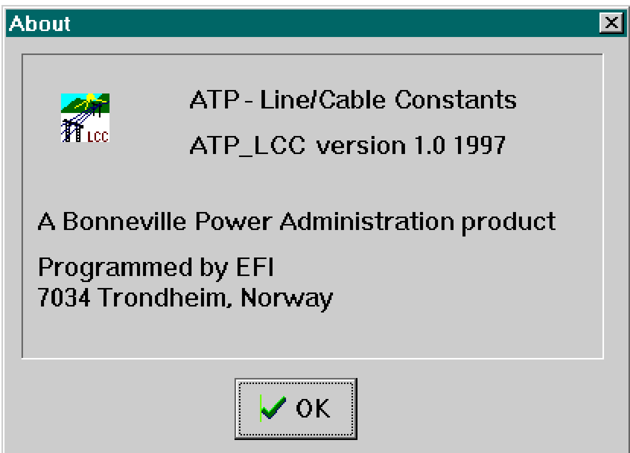
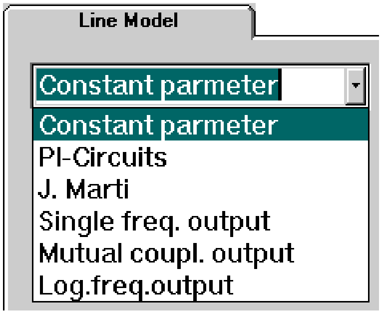


Fig. 6.8 - ATP_LCC's about box

6.9 The Line data window

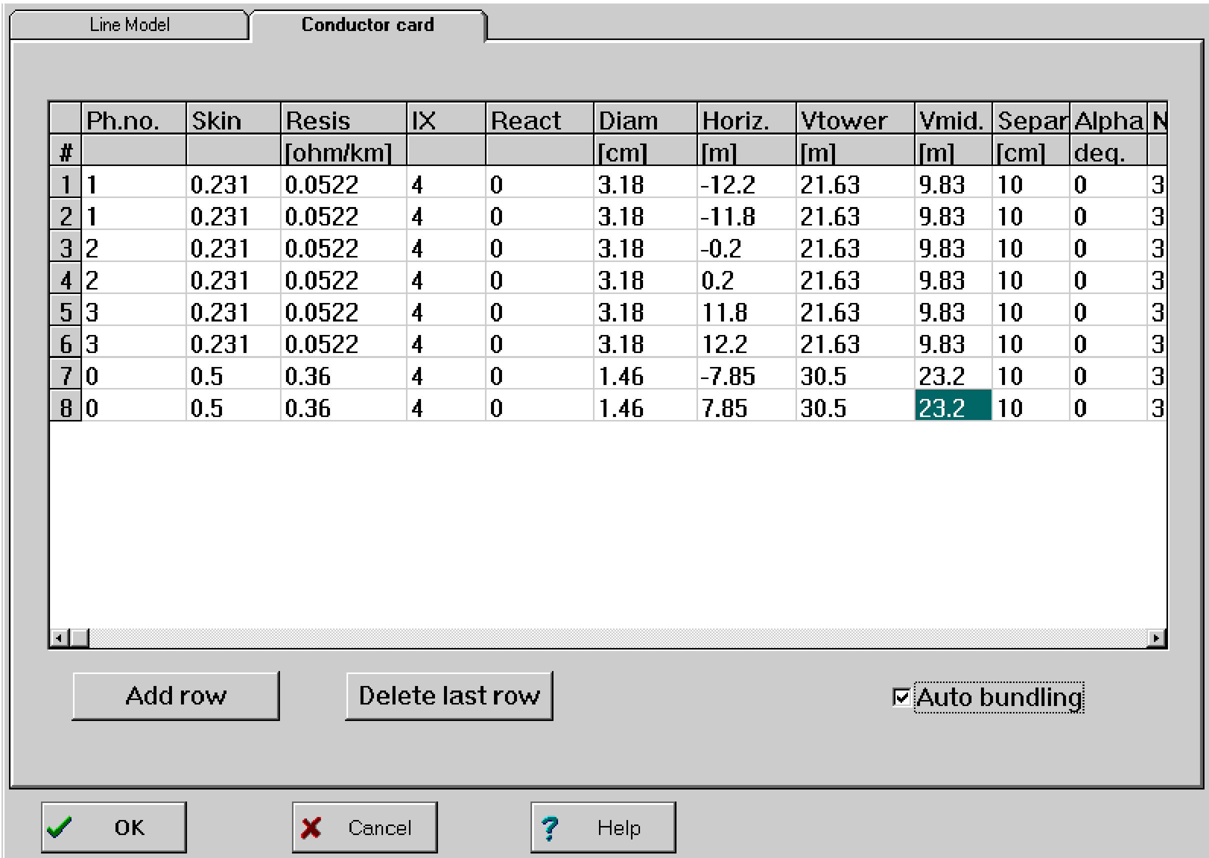
Starting from the electrical and geometrical data, the LINE CONSTANTS supporting routine of ATP calculates the electrical characteristics for overhead lines. The input data for this routine must be specified in a notebook style dialog box with two tabs: *Line Model* and *Conductor card*. On the *Line Model* card you can choose the line model from the list of the supported classes in a combo box shown in Fig. 6.9:



⇒ Constant parameter (KCLee/Clack transposed line model)
⇒ PI-equivalents (equivalent and nominal PI circuit)
⇒ JMarti lines (frequency dependent line model)
⇒ Transfer and/or shunt impedance/admittance at a given frequency
⇒ Calculate the coupling between a line and a 4th conductor
⇒ Pos. and zero sequence line parameters for several frequencies

Fig. 6.9 - Supported line models in ATP_LCC

The *Conductor card* tab is common for all line models. The structure of this tab is shown in Fig. 6.10.



#	Ph.no.	Skin	Resis [ohm/km]	IX	React	Diam [cm]	Horiz. [m]	Vtower [m]	Vmid. [m]	Separ [cm]	Alpha deg.	N
1	1	0.231	0.0522	4	0	3.18	-12.2	21.63	9.83	10	0	3
2	1	0.231	0.0522	4	0	3.18	-11.8	21.63	9.83	10	0	3
3	2	0.231	0.0522	4	0	3.18	-0.2	21.63	9.83	10	0	3
4	2	0.231	0.0522	4	0	3.18	0.2	21.63	9.83	10	0	3
5	3	0.231	0.0522	4	0	3.18	11.8	21.63	9.83	10	0	3
6	3	0.231	0.0522	4	0	3.18	12.2	21.63	9.83	10	0	3
7	0	0.5	0.36	4	0	1.46	-7.85	30.5	23.2	10	0	3
8	0	0.5	0.36	4	0	1.46	7.85	30.5	23.2	10	0	3

Fig. 6.10 - Conductor card data input window

The parameter names at the top of each column are identical with the ones used in Chapter XXI - LINE CONSTANTS of the ATP Rule Book [3]. The meaning and available settings of parameters are described here:

```
Ph.no.: phase number. 0=ground wire
Skin  : =0: no skin effect (RESIS=AC resistance)
        =T/D: Skin effect (RESIS=DC resistance)
        (T/D is the tubular thickness/conductor diameter ratio
         SKIN=0.5 for solid conductors)
Resis : AC or DC resistance in ohm/km or ohm/mile. See Skin.
IX    : 0: REACT=reactance for one unit spacing (m or foot) at Freq.
        1: REACT=reactance for one unit spacing (m or foot) at 60 Hz
        2: REACT=Geometric mean radius (cm or inch)
        3: REACT=GMR/R (Solid conductor: 0.7788)
        4: REACT=Blank. Correction for skin effect.
React : Self-inductance parameter.
Diam  : outside diameter (cm or inch) of one conductor component.
Horiz : Horizontal distance (m or foot) from the centre of bundle
        to a user selectable reference line.
VTower: vertical bundle height at tower (m or foot).
VMid  : vertical bundle height at mid-span (m or foot).
        The height h= 2/3* VMid + 1/3*VTower is used in the
        calculations.
```

The next three columns are displayed only when the *Auto bundling* checkbox is selected.

```
Separ : Distance between conductors in a bundle (cm or inch)
Alpha : Angular position of one of the conductors in a bundle,
        measured counter-clockwise from the horizontal line.
NB    : Number of conductors in a bundle.
```

You can *Add row* to the table or *Delete last row* of the table using the gray buttons under the table. The Add row command extends the table by a new row *and* the parameters of the previous last row are copied automatically into the new one. Before deleting the last row, confirmation is needed as shown in Fig. 6.11. Rows inside the table can also be deleted, but they must be dragged to the last row first. To drag a row click on its # identifier in the first column, hold the button down and drag the selected row to a new location.

Fig. 6.11 - Deleting a row needs confirmation

When you completed parameter settings on the Line model tab, as well as on the Conductor card, click on the *OK* button to return to the main window of ATP_LCC. The Line data window can be re-opened again, as many times as needed using the *Edit data* command in the Edit menu, or by a left mouse click on the Edit toolbar icon.

To obtain on-line help, select the *Help* button at any time.

6.9.1 Constant parameter line model

The line is assumed to be transposed and line parameters (surge impedance, travel time) will be calculated at a given frequency. A short description of parameters is given under the Fig. 6.12.

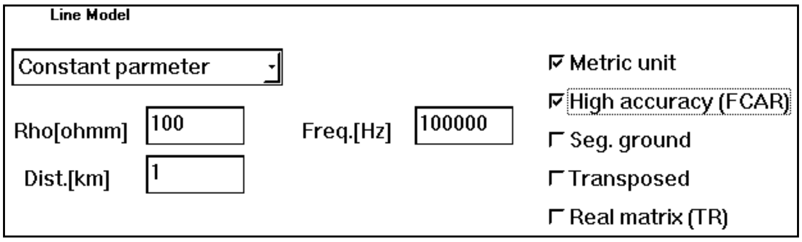


Fig. 6.12 - Settings for a constant parameter line model

```
Rho:          The ground resistivity in ohm of the homogenous earth.
Dist:         Length of overhead line.
Freq.:         Frequency at which the parameters will be calculated.
Metric/English: Switching between the Metric and English unit.
High accuracy (FCAR): If button is checked highest accuracy is used
                      in Carson's formulas.
Seg.ground:    Segmented ground wires. If button is unchecked then the
               ground wires are assumed to be continuously grounded.
Transposed:    The overhead line is transposed if button is checked.
Real/Complex matrix(TR):
            Real matrix: The eigenvectors of the transformation matrix
                         are rotated closer to the real axis so that their
                         imaginary part is assumed to become negligible.
                         Recommended for transient simulations.
         Complex matrix: Full complex transformation matrix will be used.
                         Recommended for steady state calculations.
```

6.9.2 Pi-equivalent line model

Equivalent or nominal PI-circuits will be calculated. Fig. 6.13 shows the setting options.

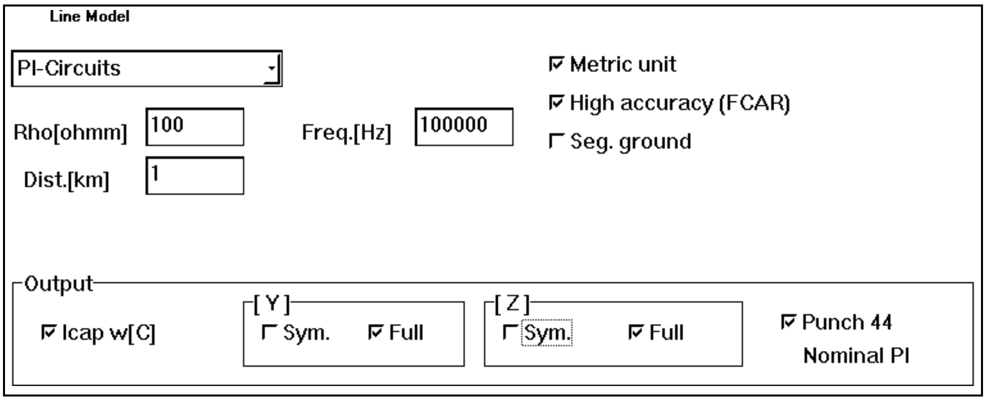


Fig. 6.13 - Settings for PI-circuits output

```
Rho:          The ground resistivity in ohm of the homogenous earth.
Dist:         Length of overhead line.
Freq.:         Frequency at which the parameters will be calculated.
Metric/English: Switching between the Metric and English unit.
High accuracy (FCAR): If button is checked highest accuracy is used
                      in Carson's formulas.
```

Seg.ground: Segmented ground wires. If button is unchecked then the ground wires are assumed to be continuously grounded.
Icap: Specifies selection between the capacitance matrix and the susceptance matrix (ωC).
Y/Z Sym: If selected shunt admittance/impedance and transfer admittance/impedance of the symmetrical components of the system of equivalent phase conductors will be calculated (perfect transposition assumed).
Y/Z Full: If selected shunt admittance/impedance and transfer admittance/impedance of the unreduced system will be calculated.
Punch 44: Request for punching of nominal PI circuit.

6.9.3 JMarti line model

The parameter names used in Fig. 6.14 are identical with those ones of used in Chapter XVII - JMARTI setup of the ATP Rule Book [3].

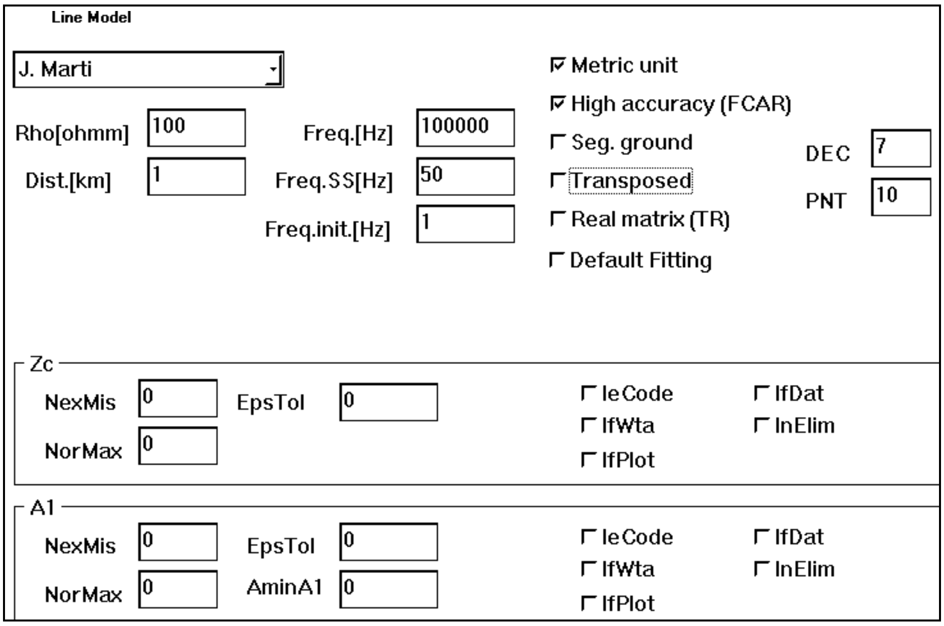


Fig. 6.14 - Settings for JMarti lines

Rho: The ground resistivity in ohm of the homogenous earth.
Dist: Length of overhead line.
Freq.: Frequency at which the transformation matrix is calculated.
Freq.SS: Frequency at which the steady state values are calculated.
Freq.Init: Initial frequency of the logarithmic frequency looping.
Metric/English: Switching between the Metric and English unit.
High accuracy (FCAR): If button is checked highest accuracy is used in Carson's formulas.
Seg.ground: Segmented ground wires. If button is unchecked then the ground wires are assumed to be continuously grounded.
Transposed: The overhead line is transposed if button is checked.
Real/Complex matrix(TR):
Real matrix: The eigenvectors of the transformation matrix are rotated closer to the real axis.
Complex matrix: Full complex transformation matrix will be used.
DEC: Number of decades in the logarithmic frequency loop.
PNT: Number of frequency points per decade.

```
Default fitting: when unselected the default values are used, otherwise:
   Zc:      NexMis: Different fitting (normally blank). Normax>=0
            EpsTol: fitting tolerance [%]
            NorMax: maximum order (default 30)
            IeCode: off: accuracy based on EpsTol ,
                    on : accuracy based on order <= NorMax
            IfWta:  on : output of fitting comparing rational
                         approximation and theory.
            IfPlot: on : character plot of fitting (as above).
            IfDat:  on : output of transmission circuit parameters
                         as function of frequency.
            InElim: on : order increases regardless of improvements
                         in fitting.
   A1:      Same parameters as Zc +
            NexMis: Different fitting (normally blank). Normax<=0
            AminA1: Data points with magnitude less than AminA1 will
                    be ignored during the fitting process.
```

6.9.4 Matrix output of the line parameters

Shunt admittance and series impedance matrixes will be calculated at a given frequency.

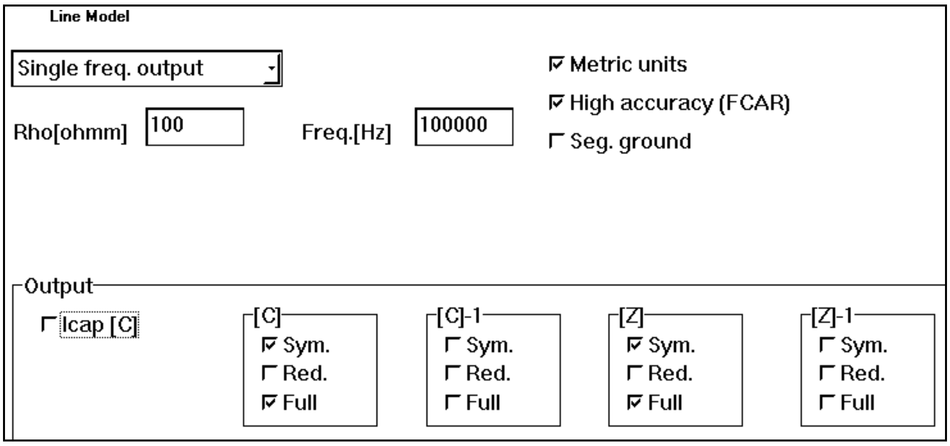


Fig. 6.15 - Settings for shunt and series matrix requests

```
Rho:       The ground resistivity in ohm of the homogenous earth.
Freq.:      Frequency at which the per unit series impedance and the
            shunt capacitance matrices are calculated.
Metric/English: Switching between the Metric and English unit.
High accuracy (FCAR): If button is checked highest accuracy is used
                      in Carson's formulas.
Seg.ground:   Segmented ground wires. If button is unchecked then the
              ground wires are assumed to be continuously grounded.
Icap:         Specifies selection between the capacitance matrix (C)
              and the susceptance matrix (ωC).
C/Z Sym:      If selected shunt capacitance and/or series impedance
              matrix for the symmetrical components of the equivalent
              phase conductor system will be calculated.
C/Z Red:      If selected shunt capacitance and/or series impedance
              matrix for the equivalent phase conductor system after
              elimination of ground wires and the bundling of
              conductors will be calculated.
C/Z Full:     If selected shunt capacitance and/or series impedance
              matrix of the unreduced system will be calculated.
```

6.9.5 Mutual coupling output

Requests output for mutual coupling of a three-phase power carrier line with a communication line running parallel to this line. Available setting options are shown in Fig. 6.16.

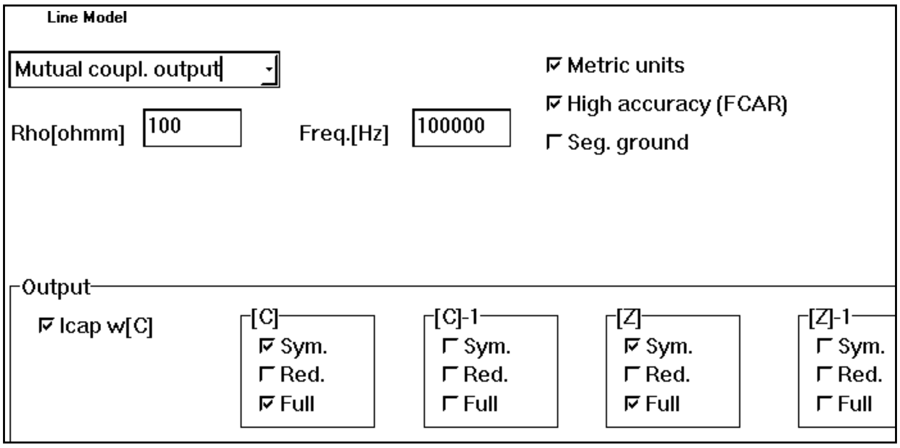


Fig. 6.16 - Settings for mutual coupling output

Parameter settings are identical with that shown in section 6.9.4.

6.9.6 Positive and zero sequence line parameters

Positive and zero sequence R, L, C line parameters will be calculated for several frequencies. This option always assumes full transposition of the n-conductor system. Available setting options are shown in Fig. 6.17.

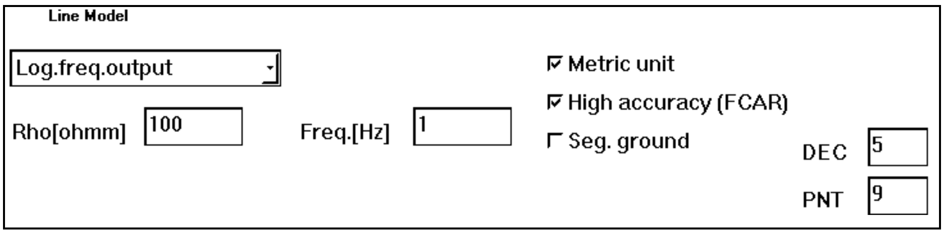


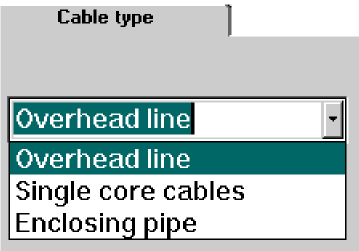
Fig. 6.17 - Settings for sequence values versus frequency

Rho: The ground resistivity in ohm of the homogenous earth.
Freq.: Frequency of the initial frequency point
Metric/English: Switching between the Metric and English unit.
High accuracy (FCAR): If button is checked highest accuracy is used in Carson's formulas.
Seg.ground: Segmented ground wires. If button is unchecked then the ground wires are assumed to be continuously grounded.

Note! “Transposed” is selected internally by the program, because ATP always assumes that the line is transposed in his case.

6.10 The Cable data window

Starting from physical and geometrical data, the CABLE CONSTANTS routine of ATP calculates electrical characteristics for cables and overhead lines. The input data for such a calculation must be specified in a notebook style dialog box having 5 tabs: *Cable type*, *Output*, *Pipe data*, *Cable data* and *Line data* (see Fig. 6.19). The *Cable type* tab has a combo box, in which you can select the configuration classes, as shown in Fig. 6.18:



⇒ Overhead transmission line
⇒ Single core (SC) coaxial cable without enclosing pipe
⇒ System of SC coaxial cables enclosed by a conducting pipe

Fig. 6.18 - Supported cable-type models in ATP_LCC

The tabs *Cable type* and *Output* are accessible for all cable types. The following logic controls the accessibility of the other three tabs:

- Selecting *Overhead line*: only the tab *Line data* will be accessible
- Selecting *Single core cables*: only the tab *Cable data* will be accessible
- Selecting *Enclosing pipe*: the tabs *Cable data* and *Pipe data* will be accessible

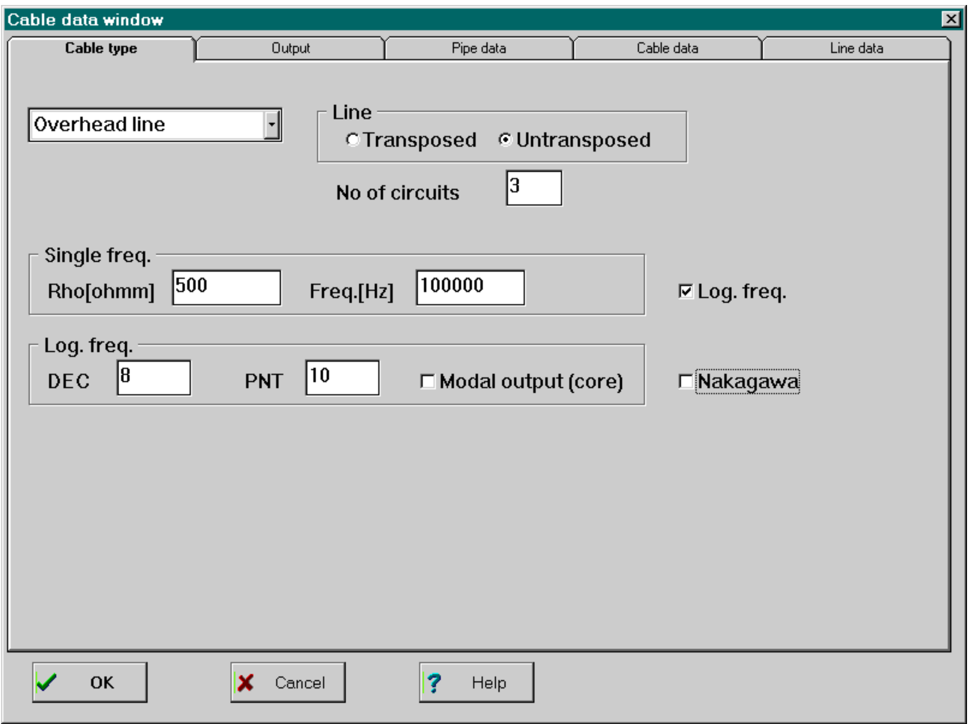


Fig. 6.19 - The Notebook style Cable data window

Once all the entries in this window have been completed, select the *OK* button to return to the main window of ATP_LCC. The Cable data window can be re-opened again as many times as you need using the *Edit data* command. To obtain on-line help, select the *Help* button at any time.

Settings for the three cable classes are described next. The name of the parameters used in the line and cable data windows are similar to the ones used in Chapter XXIII - CABLE CONSTANTS part of the ATP Rule Book [3]. The relatively new CABLE PRAMETERS supporting routine of ATP is not supported yet by ATP_LCC, it should be noted.

6.10.1 Overhead line

Cable type data tab:

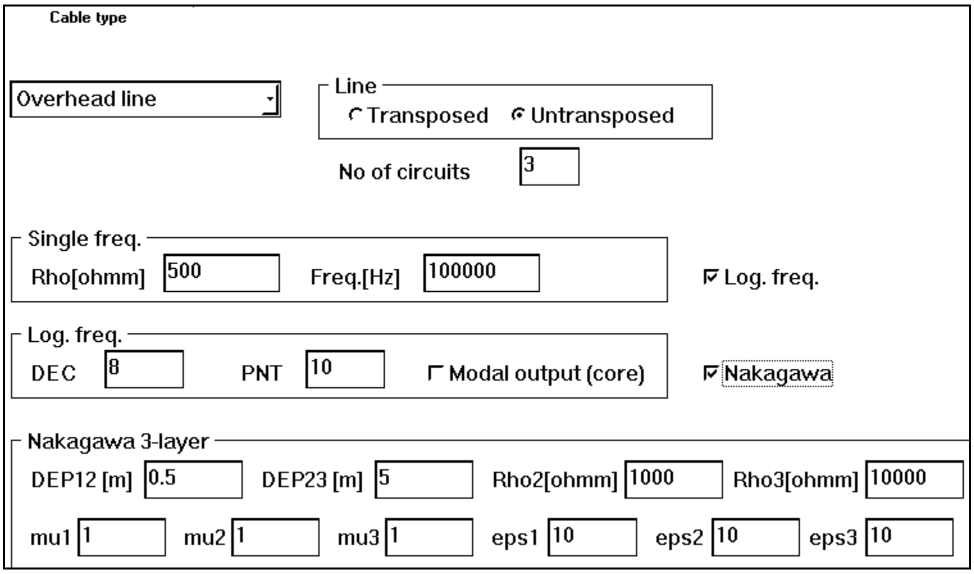


Fig. 6.20 - Cable type settings for overhead line modeling

```
Line: The line is transposed/untransposed
Number of circuits (NPC): Number of circuits which make up the system
                          Each circuit has a uniform conductor style
Rho   : Ground resistivity [ohm·m]
Freq.: Frequency [Hz]
```

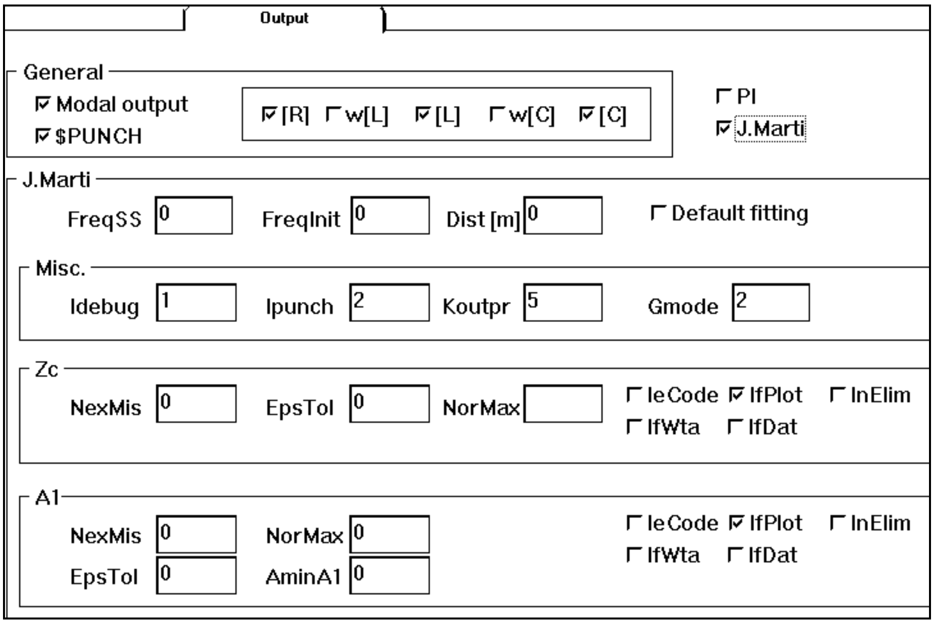
If **`Log.freq.`** is selected:

```
DEC   : Number of frequency decades starting from Freq.
PNT   : Number of frequency points per decade.
Modal output: request printed output of modal quantities for the core
```

When **`Nakagawa`** is selected, a three-layer earth model will be used:

```
DEP12: Depth of first layer [m].
DEP23: Depth of second layer [m].
Rho2 : Resistiviy of second layer [ohm·m]
Rho3 : Resistiviy of third layer [ohm·m]
       (Resistiviy of first layer is defined by Rho)
mu1  : relative permeability of first layer.
mu2  : relative permeability of second layer.
mu3  : relative permeability of third layer.
eps1 : relative permittivity of first layer.
eps2 : relative permittivity of second layer.
eps3 : relative permittivity of third layer.
```

Output data tab:



When **PI** is selected

Fig. 6.21 - Output request settings for overhead line modeling

```
Modal output (KMODE): Modal quantities will be calculated
      [L]: if checked [L] matrix will be calculated
     w[L]: if checked ω[L] matrix will be calculated
           (Always output of [R]), (specifies IZFLAG)
      [C]: if selected [C] matrix will be calculated
     w[C]: if selected ω[C] matrix will be calculated
           (Always output of [R]), (specifies IZFLAG)
$PUNCH checked enables $PUNCH output.
```

If both **PI** and **J.Marti** are unchecked, a KCLee output is created.

```
PI: when selected: equivalent PI circuit will be calculated
    NPAIS : Number of major PI sections
    XMAJOR: Length of each major section [m].
    CNAME : One character to distinguish line sections.

J.Marti: when selelected JMarti line model will be generated
    FreqSS: Frequency at which the steady state values should be
            calculated.
    FreqInit: Initial frequency of the logarithmic frequency looping.
    Dist: Length of the overhead line.
Default fitting: Recommended for inexperienced users.
    Idebug: diagnostic printout flag
    IPunch: punch branch cards
    Koutpr: visibility of JMarti input data card interpretation (SKIP).
    Gmode:  phase to ground conductance. Default: 3e-8 mhos/km.
Zc: NexMis: Different fitting (normally blank). Normax>=0
    EpsTol: fitting tolerance [%]
    NorMax: maximum order (default 30)
    IeCode: on/off: accuracy based on EpsTol or based on order <= NorMax
    IfWta:  on: output of fitting comparing rational approx. and theory.
```

```
     IfPlot: on: character plot of fitting (as above).
     IfDat:  on: output of circuit parameters as function of frequency.
     InElim: on: order increases regardless of improvements in fitting.
 A1: Same parameters as Zc +
     NexMis: Different fitting (normally blank). Normax<=0
     AminA1: Data points with magnitude less than AminA1 will be ignored
             during the fitting process. Default value = 0.05.
```

Line data data tab:

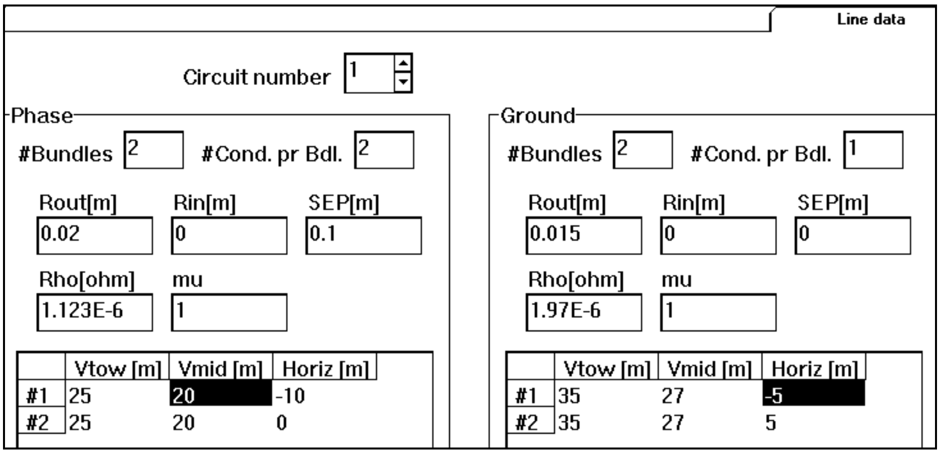


Fig. 6.22 - Line data specification card

```
Circuit number: Select actual circuit number.
                Change the maximum number on the Cable type tab.
Phase:
  #Bundles: Number of phase bundles in current circuit.
            The position grid at the bottom changes size.
  #Cond.pr Bdl.: Number of phase conductors per bundle. 0 handled 1.
  Rout : outer radius of phase conductor.
  Rin  : inner radius of phase conductor. Solid conductor Rin=0.
  SEP  : distance between phase conductors in a bundle.
         0 handled as blank.
  Rho  : resistivity of phase conductors.
  mu   : relative permeability of phase conductors.
  Vtow : height of phase bundle at tower.
  Vmid : height of phase bundle at mid-span.
         h = 2/3*Vmid + 1/3*Vtow used in calculations
  Horiz: horizontal position to user selectable reference line.
         Vtow, Vmid and Horiz must be specified for each bundle.
Ground:
  #Bundles: Number of ground bundles in current circuit.
            The position grid at the bottom changes size.
  #Cond. pr Bdl.: Number of ground conductors per bundle. 0 handled as 1.
  Rout : outer radius of ground conductor.
  Rin  : inner radius of ground conductor. Solid conductor Rin=0.
  SEP  : distance between ground conductors in a bundle.
         0 handled as blank
  Rho  : resistivity of ground conductors.
  mu   : relative permeability of ground conductors.
  Vtow : height of ground bundle at tower.
  Vmid : height of ground bundle at mid-span.
         h = 2/3*Vmid + 1/3*Vtow used in calculations
  Horiz: horizontal position to user selectable reference line.
         Vtow, Vmid and Horiz must be specified for each bundle.
```

6.10.2 Cable without enclosing pipe

This selection specifies a Class-A type cable system which consists of single-core (SC) coaxial cables without enclosing conducting pipe. The cable system might be located underground or in the air.

Cable type data tab:

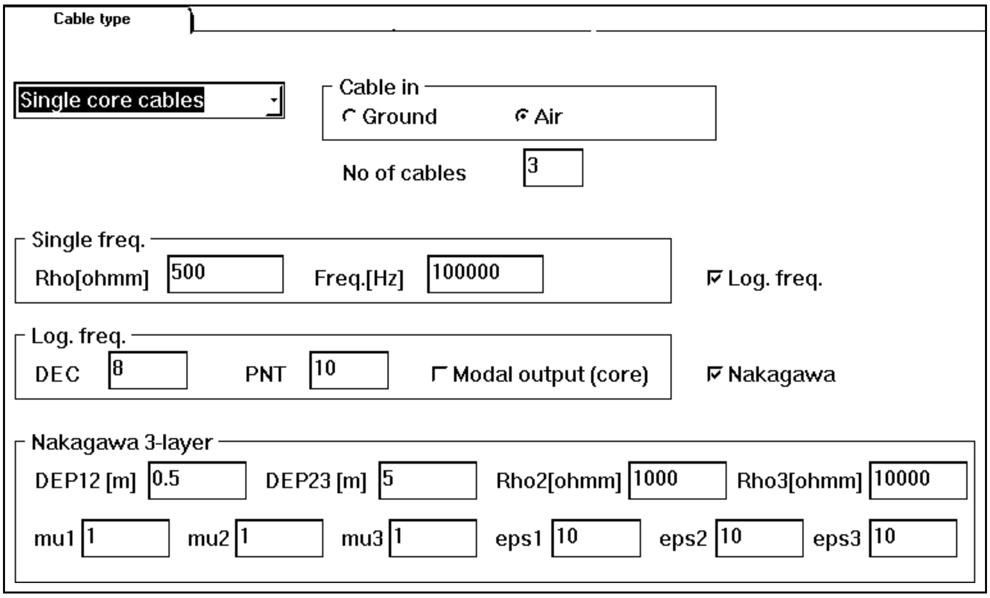


Fig. 6.23 - Cable type settings for single core cable modeling

```
Cable in: The cable is in the air or underground
No of cables (NPC): Number of cables which make up the system
Rho   : Ground resistivity [ohm·m]
Freq.: Frequency [Hz]
```

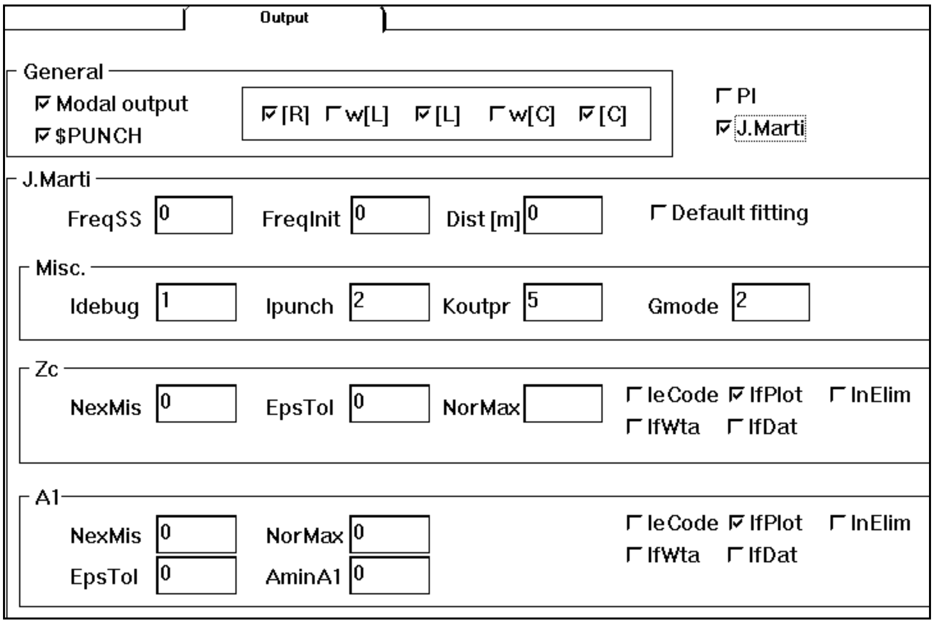
If **`Log.freq.`** is selected:

```
DEC   : Number of frequency decades starting from Freq.
PNT   : Number of frequency points per decade.
Modal output: request printed output of modal quantities for the core
```

When **`Nakagawa`** is selected, a three-layer earth model will be used:

```
DEP12: Depth of first layer [m].
DEP23: Depth of second layer [m].
Rho2 : Resistiviy of second layer [ohm·m]
Rho3 : Resistiviy of third layer [ohm·m]
       (Resistiviy of the first layer is defined by Rho)
mu1  : relative permeability of first layer.
mu2  : relative permeability of second layer.
mu3  : relative permeability of third layer.
eps1 : relative permittivity of first layer.
eps2 : relative permittivity of second layer.
eps3 : relative permittivity of third layer.
```

Output data tab:



When **PI** is selected

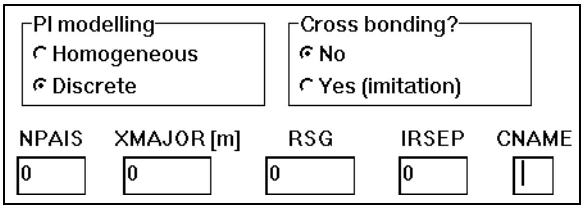


Fig. 6.24 - Output request settings for SC cable modeling

```
Modal output (KMODE): Modal quantities will be calculated
      [L]: if checked [L] matrix will be calculated
     w[L]: if checked ω[L] matrix will be calculated
           (Always output of [R]), (specifies IZFLAG)
      [C]: if selected [C] matrix will be calculated
     w[C]: if selected ω[C] matrix will be calculated
           (Always output of [R]), (specifies IZFLAG)
$PUNCH checked enables $PUNCH output.
```

If both **PI** and **J.Marti** are unchecked, a KCLee output is created.

```
PI: when selected: equivalent PI circuit will be calculated
    Homogeneous selected:
      Cross bonding:
      No : Normal pi-circuit. Recommended NGRND=0 if NPAIS<=3.
      Yes: All sheaths and armors short-circuited.
           Sheath and armor grounded through RSG each major section.
           NGRND must be 0 (not tested). Not valid for transient studies!
           Recommended NPAIS >> 3.
    Discrete selected:
      Cross bonding:
      No : Similar as homogeneous but potential of each conductor kept
           separate for each major section and the interconnection between
           each section is determined by IRSEP.
```

```
                IRSEP=0: Sheaths and armors short-circuited each major section.
                IRSEP<>0: Sheaths and armors are kept separated. Sheaths and
                          armors are grounded through RSG each major section.
                          If IRSEP<>0 and RGS is large this case degenerates to
                          the homogeneous case. NGRND must be 0 (not tested).
           Yes: Similar to the No cross bonding case by each major section is
                split in three parts with actual transposition of the sheaths.
        NPAIS : Number of major PI sections
        IRSEP : Deactivated for homogeneous modeling.
                IRSEP=0: Sheaths and armors short-circuited each major section.
                IRSEP<>0: Sheaths and armors are kept separated.
        XMAJOR: Length of each major section [m].
        RSG   : Sheath grounding resistance [ohm] at the interconnection of two
                major cable sections. Typical value 1-10 ohm.
        CNAME : One character to distinguish cable sections.
     J.Marti: when selelected JMarti line model will be generated
        FreqSS: Frequency at which the steady state values should be
                calculated.
        FreqInit: Initial frequency of the logarithmic frequency looping.
        Dist: Length of the cable system.
     Default fitting: Recommended for inexperienced users.
     No default J.Marti fitting: see under the Fig. 6.21.
```

Cable data tab:

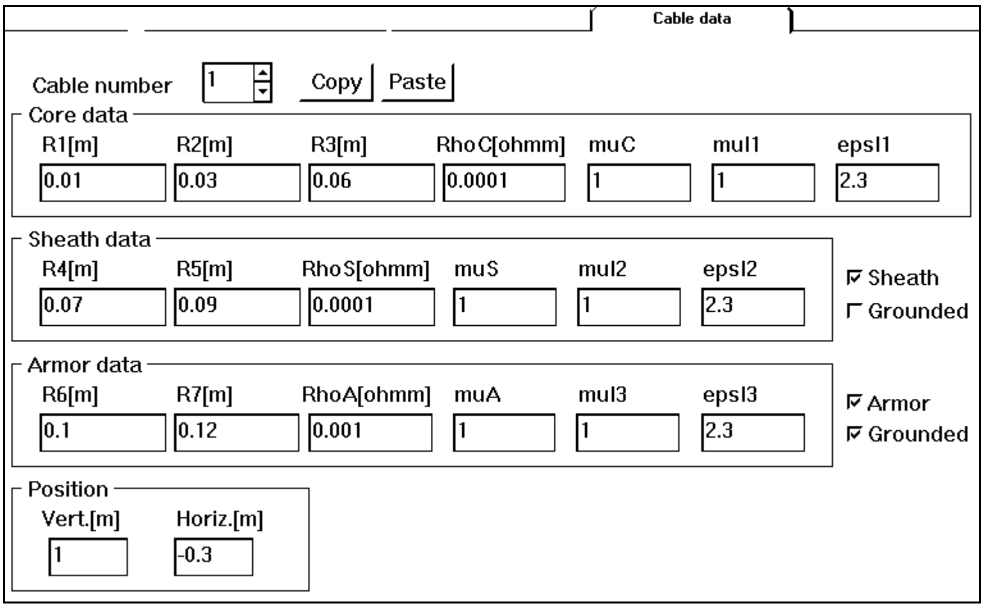


Fig. 6.25 - Cable data settings

This tab has a *Copy* and a *Paste* button, too. The data of the current cable can be copied to an internal clipboard using the Copy button. The Paste button copies the data from the clipboard into the current cable.

```
     Cable number: Select actual cable number. Change the maximum number under
                   Cable type tab.
     Core data:
       R1    : inner radius of tubular core. Specify 0 for solid core.
       R2    : outer radius core.
       R3    : outer radius of 1st insulator.
       RhoC  : resistivity of core conductor.
       muC   : relative permeability of core.
       muI1  : relative permeability of 1st insulator.
```

```
  epsI1: relative permittivity of 1st insulator.
Sheath data: Activate this field by selecting the Sheath check box.
  Grounded: When selected the sheath is grounded.
  R4    : outer radius of sheath.
  R5    : outer radius of 2nd insulator.
  RhoS  : resistivity of sheath conductor.
  muC   : relative permeability of sheath conductor.
  muI2  : relative permeability of 2nd insulator.
  epsI2: relative permittivity of 2nd insulator.
Armor data: Activate this field by selecting the Armor check box.
  Grounded: When selected the armor is grounded.
  R6    : outer radius of armor.
  R7    : outer radius of 3rd insulator.
  RhoA  : resistivity of armor conductor.
  muA   : relative permeability of armor conductor.
  muI3  : relative permeability of 3rd insulator.
  epsI3: relative permittivity of 3rd insulator.
Position:
  Vert  : Positive distance to ground surface.
  Horiz: horizontal position to user selectable reference line.
```

The parameter NGRND of ATP is established automatically based on the selected grounding conditions. NGRND=0: None grounded, 2: All armor grounded, 3: All sheaths and armors grounded, 4: Individual grounding conditions.

6.10.3 Cable system with enclosing pipe

This selection specifies a cable system consisting of single-core (SC) coaxial cables, enclosed by a conducting pipe (referred as class-B type in the ATP Rule Book [3]). The cable system might be located underground or in the air.

Settings on the *Cable type* and on the *Output* tabs are exactly the same as in the case of single core cables (see in the previous section 6.10.2).

Pipe data tab:

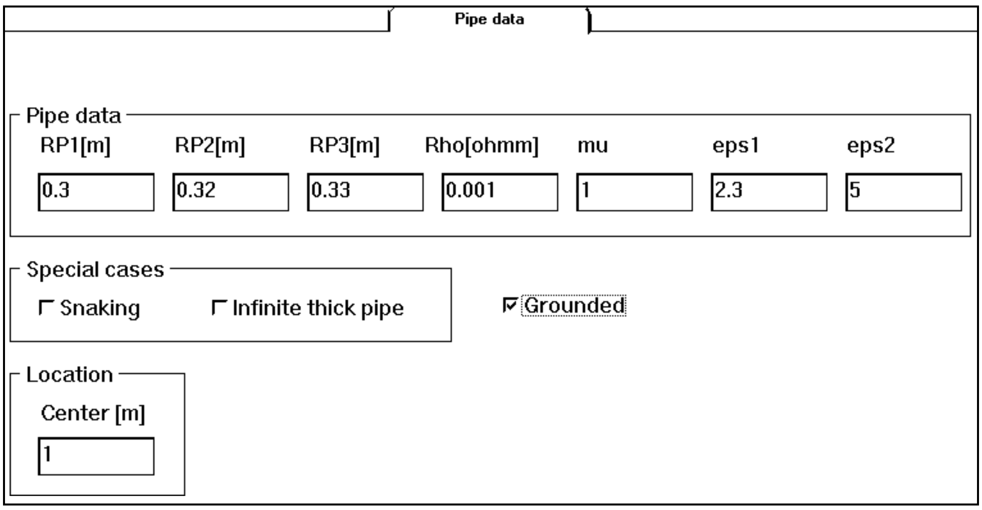


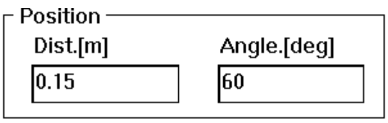
Fig. 6.26 - Pipe data settings

```
RP1:  inner radius of pipe conductor
RP2:  outer radius of pipe conductor
```

```
    RP3:  outer radius of outer insulator
    Rho:  resistivity of pipe conductor [ohm]
    mu:   relative permeability of pipe conductor
    eps1: relative permittivity of inner insulator
    eps2: relative permittivity of outer insulator
  Special cases:
    Snaking: transposition of the cables within the enclosing pipe.
    Infinite thick pipe: ISYST=0 and NGRND<>4 (uniform grounding).
   Grounded: When selected all armors, as well as the pipe is assumed
             to be grounded
     Center: distance between ground surface and the pipe center.
             Always specify Center > 0.
```

Cable data tab:

Settings on the *Cable data* tab are almost identical with that used for single core cables (see the previous section 6.10.2). The reference point now is the pipe center, so the location of cables inside the pipe must be given in polar coordinates in the *Position* field.




```
  Position:
    Vert : Positive distance from pipe center to cable center.
    Angle: Angular position of the cable measured counter clockwise.
```

6.11 Application examples

The Line Constants cases shown here as application examples of the use of ATP_LCC program have already been referenced in section 5.5.1 and 5.5.4 of the Advanced Manual. As it is explained there, ATPDraw recognizes the large majority of punch files created by the Line/Cable constants support of ATP automatically, so such files can be used as input for the *Overhead Line (PCH)* objects. It means that for most of the cases, users do not need to create supporting files to $Include a .PCH file into their ATPDraw circuit. I.e. ATPDraw recognizes the most frequently used file formats (frequency dependent, constant parameter, PI-equivalent) specified in the punch files and creates a DBM compatible library file automatically, and applies the proper supporting files. This section of the manual describes how to create such a punch file by using the ATP_LCC program.

6.11.1 JMarti model of a 500 kV line

The example line is a 138 miles long 500 kV overhead line (from John Day to Lower Monumental) taken from benchmark `DCN3.DAT.` The line configuration is given in Fig. 6.27. All parameters are in English units. The phase conductors are tubular, with:

T/D = 0.364
DC resistance = 0.05215 Ω/mile
Outside diameter of the conductors = 1.602 inch.

The sky wires are solid, so: T/D = 0.5.

DC resistance = 2.61 Ω/mile.

Outside diameter of the ground wires is 0.386 inch.

The earth resistivity is equal to 100 ohm·meter.

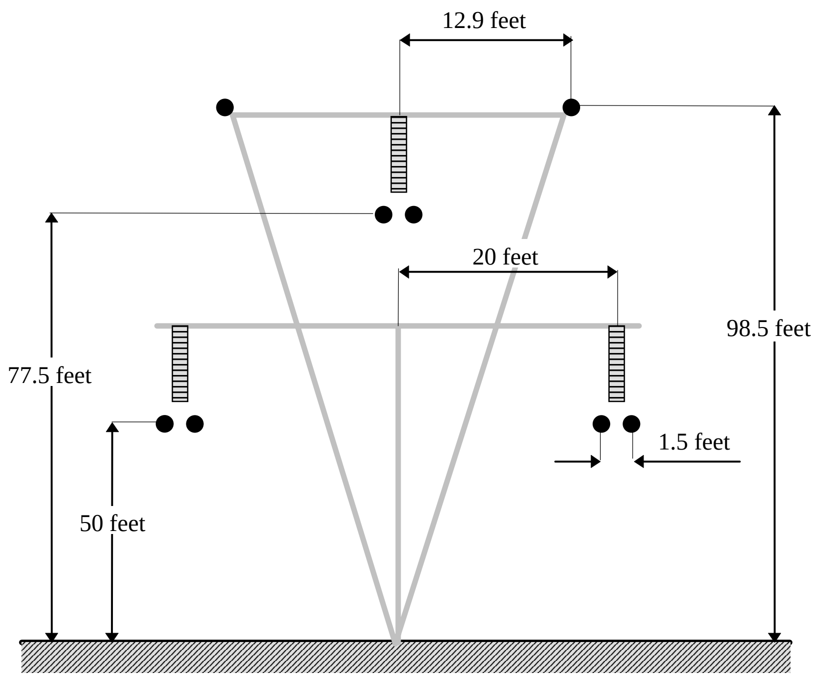


Fig. 6.27 - Line configuration

In ATP_LCC, the geometrical and electrical data of the system can be specified in a two pages tabbed notebook style *Line data* window. This window appears when you select the *New Line* option in the *File* menu, or when an existing line/cable data file is selected in the *File* | *Open*.

Example case *Exa_7.cir* of the Advanced Manual requires an untransposed, frequency dependent line model. Settings for this case are shown in Fig. 6.28 (*Line Model)* and in Fig. 6.29 (*Conductor card).* This tabbed window is displayed on the screen by performing a *File* | *Open* and loading the `JMARTI_1.LIN` into ATP_LCC.

Settings in the *Line Model* and *Conductor cards* for JMarti modeling are described in section 6.9 of this manual and in Chapter XVII - JMARTI setup of the ATP Rule Book [3]. In this example the otherwise powerful “manual bundling” option is not used, so all conductors are specified separately. The “automatic bundling” is described next in section 6.11.2.

The ATP input file is generated when you select the *ATP* | *Create data case* command. For the `JMARTI_1.LIN` case this file is shown next. Processing it with ATP, the resulting punched output (`.PCH`) can be used as input for the ATPDraw’s *Overhead Line (PCH)* objects.

```
BEGIN NEW DATA CASE
JMARTI SETUP
$ERASE
LINE CONSTANTS
ENGLISH
  1 .364   .05215 4              1.602  -20.75     50.      50.
  1 .364   .05215 4              1.602  -19.25     50.      50.
  2 .364   .05215 4              1.602   -0.75    77.5     77.5
  2 .364   .05215 4              1.602     .75    77.5     77.5
```

```
  3 .364  .05215 4           1.602   19.25    50.     50.
  3 .364  .05215 4           1.602   20.75    50.     50.
  0   .5    2.61 4            .386   -12.9    98.5    98.5
  0   .5    2.61 4            .386    12.9    98.5    98.5
BLANK CARD ENDING CONDUCTOR CARDS
    100.     5000.         1                   138.     1           1
    100.       60.         1                   138.     1           1
    100.       .01         1                   138.     1   9 10    1 0
BLANK CARD ENDING FREQUENCY CARDS
BLANK CARD ENDING LINE CONSTANT
               0       0       0       0
       0      .3      30               1       1
       0      .3      30               1       1                     0
$PUNCH
BLANK JMARTI
BEGIN NEW DATA CASE
BLANK CARD
```

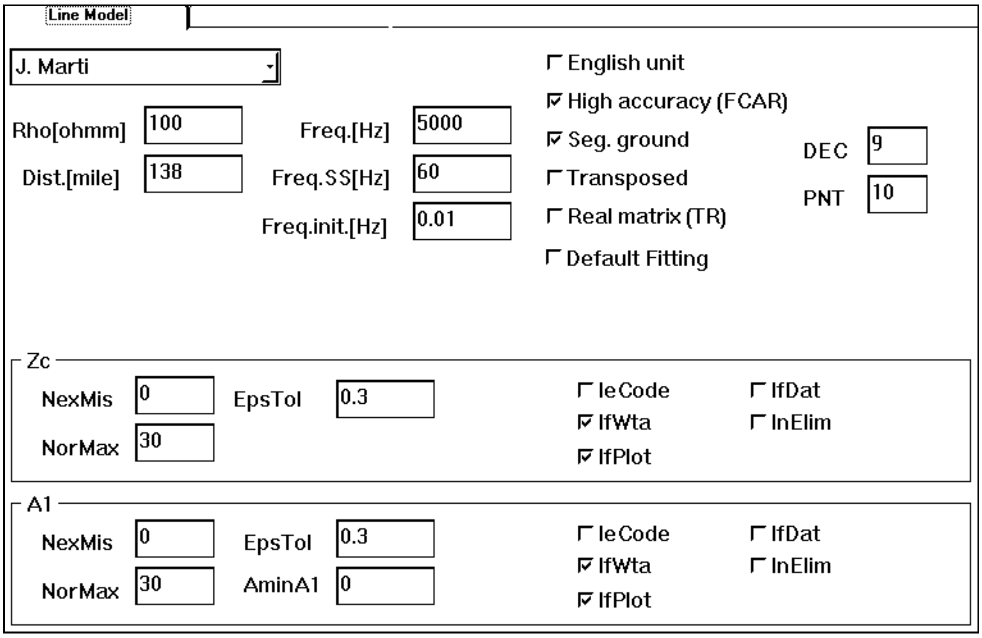
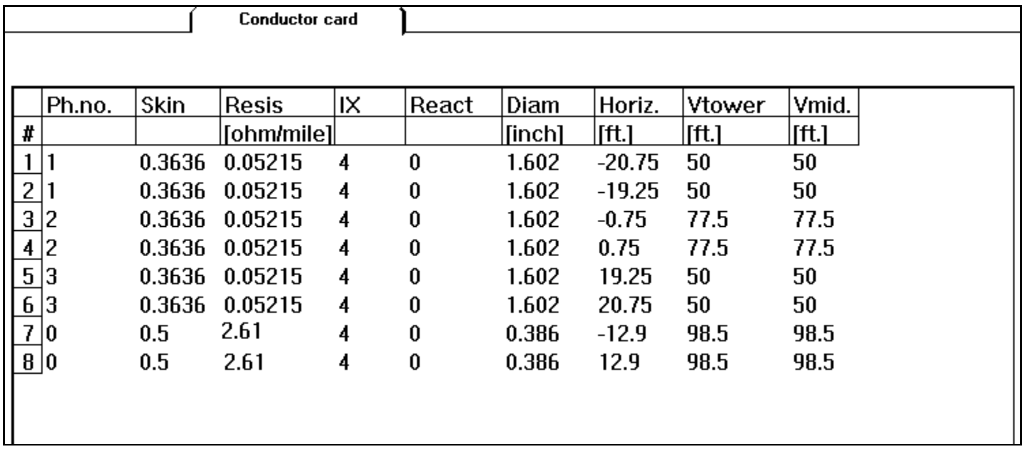


Fig. 6.28 - Line Model window for the case JMARTI_1.LIN


Conductor card

#	Ph.no.	Skin	Resis [ohm/mile]	IX	React	Diam [inch]	Horiz. [ft.]	Vtower [ft.]	Vmid. [ft.]
1	1	0.3636	0.05215	4	0	1.602	-20.75	50	50
2	1	0.3636	0.05215	4	0	1.602	-19.25	50	50
3	2	0.3636	0.05215	4	0	1.602	-0.75	77.5	77.5
4	2	0.3636	0.05215	4	0	1.602	0.75	77.5	77.5
5	3	0.3636	0.05215	4	0	1.602	19.25	50	50
6	3	0.3636	0.05215	4	0	1.602	20.75	50	50
7	0	0.5	2.61	4	0	0.386	-12.9	98.5	98.5
8	0	0.5	2.61	4	0	0.386	12.9	98.5	98.5

Fig. 6.29 - Conductor card window for the case JMARTI_1.LIN

6.11.2 JMarti model of a 750 kV line

The line models calculated here are applied in the example case *Exa_7a.cir* of the Advanced Manual. The LCC supporting program has been used again to create a JMarti model of a 750 kV transmission line with total length of 487 km. Transpositions separate this line into four sections, so the LCC input files ready to be loaded for this example are called: `LIN750_x.LIN`, where x runs from 1-4. The line configuration is shown in Fig. 6.30.

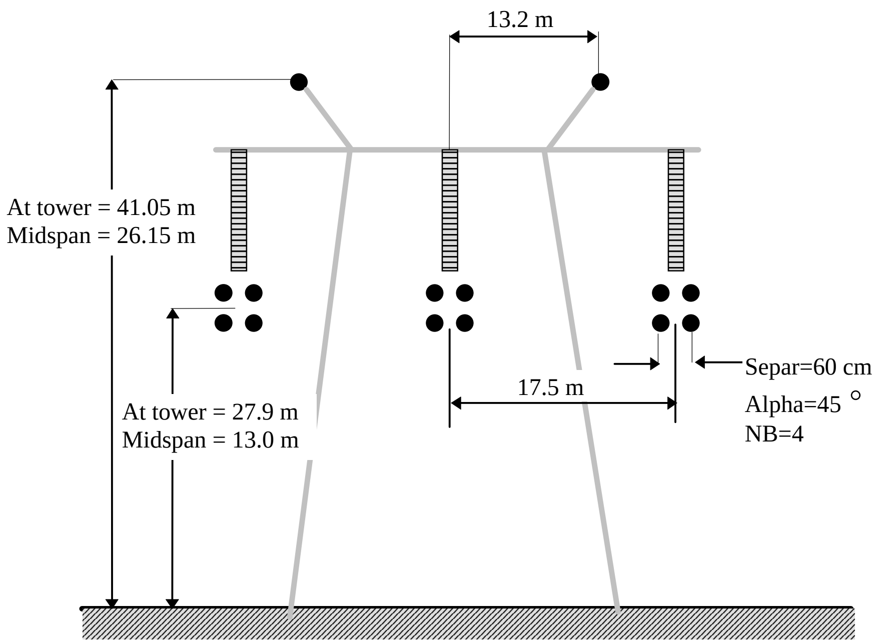


Fig. 6.30 - Tower configuration of the 750 kV line

All parameters are given in Metric units and the “automatic bundling” option of ATP is used in this example. Each phase has four conductors in rectangular arrangement. The cunductors are assumed to be tubular, with:

T/D = 0.32
DC resistance = 0.0585 Ω/km
Outside diameter of the conductors = 3.105 cm.

The sky wires are made from steel reinforced conductors, so it can also be assumed to be tubular, with:

T/D = 0.187
DC resistance = 0.304 Ω/km
Outside diameter of the ground wires = 1.6 cm

Loading the `LIN750_1.LIN` into ATP_LCC the *Line Model* and the *Conductor card* windows are as shown in Fig. 6.31 and in Fig. 6.32 . The JMarti setup input file generated when you activate the *ATP | Create data case* command of the ATP_LCC is shown below:

```
BEGIN NEW DATA CASE
JMARTI SETUP
$ERASE
LINE CONSTANTS
```

```
METRIC
  1  .32   .0585 4            3.105  -17.5    27.9     13.     60.   45.       4
  2  .32   .0585 4            3.105      0    27.9     13.     60.   45.       4
  3  .32   .0585 4            3.105   17.5    27.9     13.     60.   45.       4
  0 .187    .304 4              1.6  -13.2   41.05   26.15       0     0       0
  0 .187    .304 4              1.6   13.2   41.05   26.15       0     0       0
BLANK CARD ENDING CONDUCTOR CARDS
     20.     5000.         1                   84.6      0           1
     20.       50.         1                   84.6      0           1
     20.       .05         1                   84.6      0    7 10   1 0
BLANK CARD ENDING FREQUENCY CARDS
BLANK CARD ENDING LINE CONSTANT
              0       0       0       0
       0      1.      30              1
       0      1.      30      1       1                     .05
$PUNCH
BLANK JMARTI
BEGIN NEW DATA CASE
BLANK CARD
```

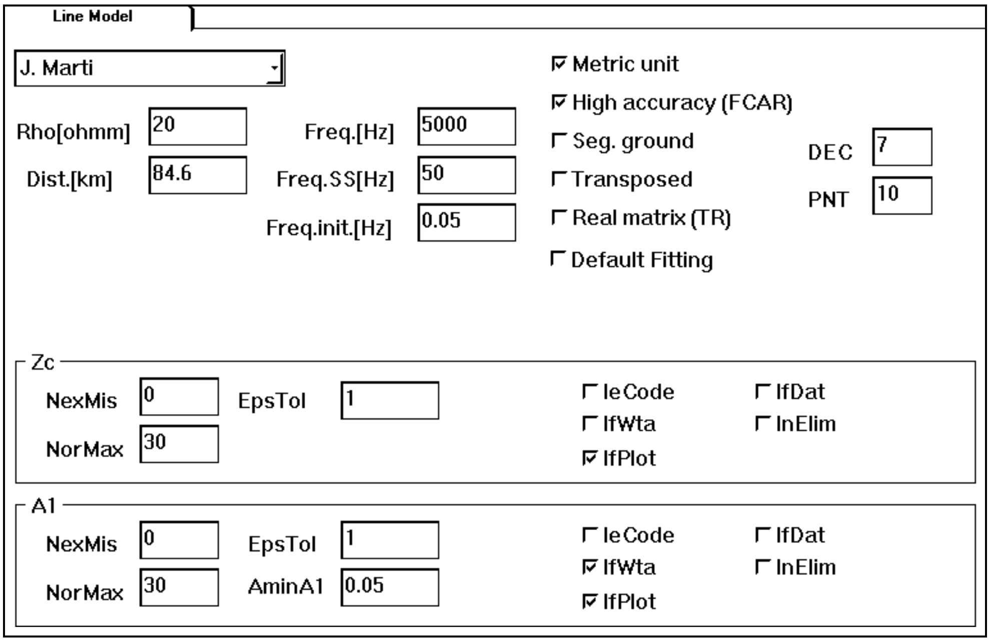
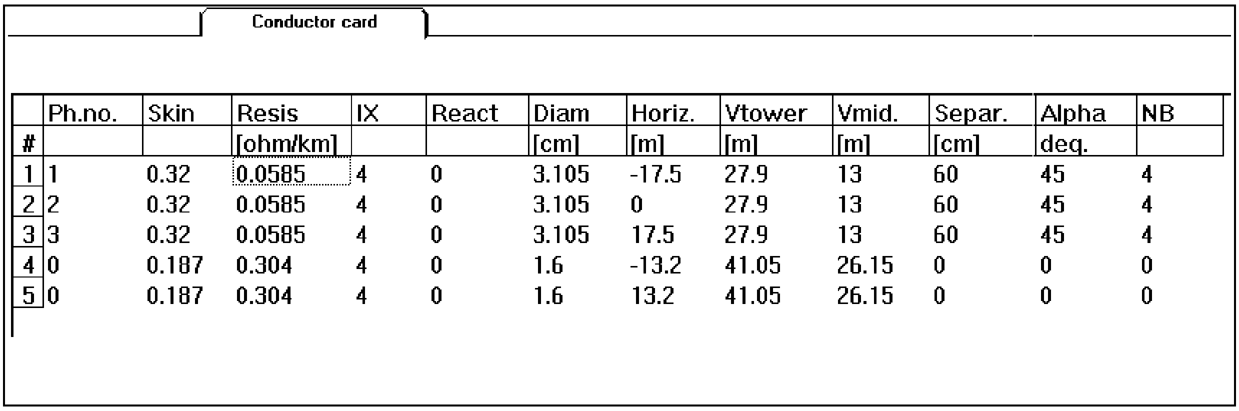


Fig. 6.31 - Line Model window for the case LIN750_1.LIN



#	Ph.no.	Skin	Resis [ohm/km]	IX	React	Diam [cm]	Horiz. [m]	Vtower [m]	Vmid. [m]	Separ. [cm]	Alpha deg.	NB
1	1	0.32	0.0585	4	0	3.105	-17.5	27.9	13	60	45	4
2	2	0.32	0.0585	4	0	3.105	0	27.9	13	60	45	4
3	3	0.32	0.0585	4	0	3.105	17.5	27.9	13	60	45	4
4	0	0.187	0.304	4	0	1.6	-13.2	41.05	26.15	0	0	0
5	0	0.187	0.304	4	0	1.6	13.2	41.05	26.15	0	0	0

Fig. 6.32 - Conductor card window for the case LIN750_1.LIN

7. References

[1] *ATPDRAW version 3*, User Manual, TR A4389, EFI, Norway, 1996

[2] Ned Mohan, *Computer Exercises for Power Electronic Education*, 1990, Department of Electrical Engineering, University of Minnesota.

[3] *ATP-EMTP Rule Book*, Canadian-American EMTP Users Group, 1997

[4] Laurent Dubé, *MODELS in ATP*, Language manual, February 1996