MINNIE AND HSPICE FOR ANALOGUE CIRCUIT SIMULATION

Derek C. Barker

CHAPMAN & HALL

MINNIE and HSpice for Analogue Circuit Simulation

DEREK C. BARKER
Department of Electrical Engineering and Electronics
University of Manchester Institute of Science and Technology

CHAPMAN & HALL
University and Professional Division
London · New York · Tokyo · Melbourne · Madras

Published by Chapman & Hall, 2–6 Boundary Row, London SE1 8HN

Chapman & Hall, 2–6 Boundary Row, London SE1 8HN, UK

Van Nostrand Reinhold Inc., 115 5th Avenue, New York NY10003, USA

Chapman & Hall Japan, Thomson Publishing Japan, Hirakawacho Nemoto Building, 7F, 1-7-11 Hirakawa-cho, Chiyoda-ku, Tokyo 102, Japan

Chapman & Hall Australia, Thomas Nelson Australia, 102 Dodds Street, South Melbourne, Victoria 3205, Australia

Chapman & Hall India, R. Seshadri, 32 Second Main Road, CIT East, Madras 600 035, India

First edition 1991

Typeset in 10½/12pt Times by Excel Typesetters Company
Printed in England by Clays Ltd., St Ives Plc

0 412 42760 5 0 442 31478 7 (USA)

A catalogue record for this book is available from the British Library

Library of Congress Cataloging-in-Publication data available

Printed on permanent acid-free text paper, manufactured in accordance with the proposed
ANSI/NISO Z 39.48–199X and ANSI Z 39.48–1984

MINNIE and HSpice for Analogue Circuit Simulation

Preface

After many years of teaching circuit theory and analogue electronic circuits the author believes that for most students the main path to obtaining a good understanding of the principles involved, as measured by their ability to apply them in a correct and intelligent manner, is through problem solving and design exercises.

In an ideal world the student would be able to construct the circuit being analysed or designed, and so directly test the calculated or predicted results. Indeed, experience leads to the conclusion that typical students like to see their own circuits perform as intended, with a consequent increase in motivation.

At present, however, time and facility constraints mean that most of this work is of the pencil and paper variety, students having few opportunities to see the consequences of their efforts in a practical situation. At best they have to accept sample solutions or simply numerical answers. This path can seem tedious to all but the most motivated of students, so an alternative which can provide many of the benefits of direct circuit testing within the time and resources available is of immediate interest. This is where the MINNIE and HSpice simulation package can assist the undergraduate teaching activity.

Interest in simulation of analogue electronic circuits has existed for many years both in industry and in educational establishments. Software packages using advanced mathematical techniques allied to reasonable device models are widely available – Spice, HSpice and other Spice derivatives are well known and are not unique. However, the size of these programs and the time required for simulation of even a simple circuit implies computing power until recently available only on mainframes and minicomputers. They were written by and for experienced circuit designers, and did not incorporate the 'user friendly' approach now popular in the personal computer market.

MINNIE provides just such a user friendly interface for HSpice. By accepting circuit information in circuit diagram form (**schematic capture**) it builds directly on students' experience as obtained in lectures and laboratories. Consequently they have immediate understanding of the procedures to be followed to simulate the performance of their circuits. Allied to the vastly increased computing power and greatly improved screen resolutions of the workstations or personal computers now generally available, students have a very powerful tool at their disposal.

When MINNIE/HSpice first arrived at UMIST the author realized that the existing documentation, the *MINNIE User Guide*, was written for experienced users and would not be appropriate for first year undergraduates. This led to the preparation of three introductory tutorial documents aimed specifically at undergraduates. The success of these documents persuaded the author to develop his ideas further, leading to the preparation of this book. It is intended to be an initial guide to circuit simulation using MINNIE/HSpice, aimed at undergraduates (and any other interested readers), and does not attempt to replace the *MINNIE User Guide*. Consequently some of the more advanced features of MINNIE/HSpice, of interest only to experienced circuit designers, have been omitted in the interest of brevity.

The author has adopted a self-teaching approach in order to reduce the demands that would otherwise be made on already heavily loaded staff. The reader proceeds at his or her own pace, avoiding periods of peak usage of terminals or workstations and dispensing with the necessity for scheduled formal lectures. First year students are soon brought up to the point at which they are able to confirm the theoretical circuit behaviour described in textbooks and lectures, the answers obtained analytically for set problems, or the results obtained experimentally in the laboratories. The tutorial (demonstration) examples and the exercises used were selected bearing in mind the progress of students as they advance through their courses, so they will be able to simulate the standard circuits they meet in the second and third years as well as the more complex ones they may design for undergraduate and postgraduate project work.

In Chapter 1 the author is concerned with introducing the beginner to the rudiments of MINNIE/HSpice and the host system. Some items are specific to the Apollo system installed at UMIST, and are mentioned only as being representative of the systems the user is likely to encounter.

The next six chapters, dealing with the response of linear systems to sinusoidal stimulation, form the bulk of the book. While Chapter 2 deals exclusively with passive components (linear circuit theory), the next five, Chapters 3 to 7, deal mainly with circuits which include active devices (electronic circuits). The electronic circuit used for tutorial purposes is the standard bipolar common emitter, so the discussions are accessible to the typical second year (and in many cases first year) undergraduate. The exercises cover a wider range of achievement, some being taken from second and third year design exercises as implemented at UMIST. Due to the inherent similarities between bipolars and FETs, students will have little difficulty in adapting to the latter and so they are not specifically covered. Operational amplifiers, on the other hand, are introduced because they make possible the discussion of active filters and feedback oscillators (Wien Bridge) in a relatively simple manner, as well as having certain peculiarities of their own.

Many of the techniques introduced in these six chapters are of general application. The reader is introduced to the MINNIE filing and retrieval system and to its post-processing facilities in Chapter 4, while

Using this book

This book is aimed at the beginner – the first year student who is just having an initial organized look at computers, and is meeting circuit theory for the first time. If he or she is not at the same time entering the world of analogue electronic engineering it will not be long before this happens. The student has probably used a personal computer before, at school or at home so is 'computer-aware', not an absolute novice. It is also intended to be of immediate use to second and third year undergraduates.

More experienced users, postgraduates, circuit designers, etc., who are interested in the MINNIE/HSpice simulation package, will also find this book to be a useful introduction.

To meet the needs of all readers, within this book the material is presented in three different forms.

1. Guides to specific MINNIE facilities: these are comprehensive and detailed, and each step is illustrated with the relevant screen picture, so that it is clear at all times exactly what to do and what results should be obtained.
2. Tutorials: these are fully documented simulations, in which the reader is taken step by step through each stage, from setting up the initial circuit to presenting the final results. Again each step is fully documented with all the relevant screen pictures. The problem and any relevant issues are discussed, with particular attention to factors affecting the analysis. Where appropriate the commonality between simulation and laboratory procedures is emphasized.

   The choice of demonstration circuits is based on the author's experience and should provide additional insights into important topics in both circuit theory and analogue electronics.
3. Exercises: as the name implies they are presented to give further experience in the use of the facilities presented in the immediately preceding guide sections or tutorials. The assistance provided is less comprehensive than in the other sections, but is still sufficient to lead the student to the correct results.

   In several of these exercises design work is encouraged. The problem is set and one may then proceed to one's own design and simulation or alternatively use the sample solution provided. Where suitable these problems intentionally leave considerable scope for making decisions – a good incentive to understanding the factors influencing circuit performance.

the presentation and 'measurement' of simulation results is discussed in Chapter 5. Of particular interest is Chapter 6 covering independently variable passive components – though specific to variable resistors the reader can immediately extend the procedures to capacitors, etc. Finally, Chapter 7 is devoted to the single most useful activity in the electronics laboratory: the presentation and examination of waveforms. This chapter includes four quite comprehensive analysis/design exercises suitable for second/third year undergraduates (and postgraduates).

Chapter 8 forms a third part concerned with the response of linear circuits to square waves and step functions. It commences with passive circuits directly relevant to transient analysis as covered in basic circuit theory, and then proceeds to the response of the common emitter stage operating in its linear region. The transient responses are drawn and the harmonic contents (Fourier analysis) are determined. Typical circuit theory exercises are presented which emphasize the use of initial conditions, the simulation path following closely that of analysis.

Finally, Chapter 9 introduces non-linear circuits through a very comprehensive exercise set to UMIST students during their second year. The reader is guided through a detailed examination and comparison of the response of simple *RC*, diode-capacitor and transistor-capacitor circuits. The concepts of independently variable components, previously introduced for passive components, are extended to include voltage (and, by inference, current) sources.

The tutorial and exercise examples used in this book were primarily selected to demonstrate particular facilities of MINNIE in a logical order. These facilities will individually first be required at different points of the first and second years of the undergraduate course, and it was considered important that the examples should relate to corresponding parts of the syllabus. Design work is encouraged where feasible and a number of questions are asked at various points, answers being provided in the Appendix.

Derek C. Barker

Department of Electrical Engineering
and Electronics,
University of Manchester Institute of
Science and Technology.

Beginners should not attempt to work their way through all the chapters of this book immediately, but should start by familiarizing themselves with the local system, with the contents of Chapter 1 and with the first parts of Chapter 2. Then this experience can be added to as other topics are covered in the circuit theory or electronic engineering lectures.

There are opportunities offered by laboratory work, set tutorial questions, old examination papers, etc. to increase familiarity with this very powerful aid to design work. It is only through active use that the expertise necessary for efficient use of the MINNIE/HSpice package is acquired.

It should be noted that the sequence of topics, though quite logical from the organizational viewpoint of the book, is not necessarily the chronological order in which they appear in courses. Thus parts of Chapters 8 and 9 could be encountered in the first year, much earlier than the circuits of Chapter 7. Consequently progress through the book need not be strictly sequential.

Finally, do persevere. When you achieve familiarity with MINNIE you will have at your disposal a very powerful tool for all your analogue design work.

Derek C. Barker

Contents

Getting started 1

1.1 THE SYSTEM

A typical suite of workstations dedicated to undergraduate CAD activities might consist of a network of 30 Apollo (or similar) monochrome computers. Each computer or 'node' would operate independently using its own processor and memory, but have free access to files or directories belonging to any other node via a high speed communications network. This arrangement gives the user the ability to work from any node and 'transparently' use the collection of nodes as one large computer. MINNIE (and HSpice) would run on this system, together with a variety of other software packages.

The UMIST system, based on Apollo DN3000 monochrome computers, has at present about 1.4 G of disk storage capacity, which is not sufficient to meet the requirements of nearly 600 undergraduates plus interested postgraduates and academic staff. Consequently it is necessary to restrict the individual's overall memory requirements, and users are strongly encouraged to store files on personal floppy disks. This also has the virtue of protecting data from unauthorized entry.

Experience also shows that it is desirable to limit student output. It is easy and very tempting to request a screen dump for every circuit change or new result, but uncontrolled output leads to long print queues.

1.2 INITIAL CONDITIONS

The procedures described here are for an Apollo DN3000 network. Other installations will have similar start up sequences.

1. When you arrive at the workstation it should already be on. They are normally left on all the time, to reach and maintain stable operating conditions.

   If the screen is blank, press any key on the keyboard or move the mouse to wake up the terminal (the screen is blanked if it is not used within a time limit).
2. At the bottom of the screen you will see the **display command line**. On it you are asked to **login**. Use the mouse to move the arrow cursor to the display command line and type in the user name you have been given. Thus if your user name is 'apprentice', you will have

Login: apprentice

3. Then, the machine will ask for your **password**. You will have been given an initial password so at least one other person knows it. (Facilities exist for you to change it to a private code, but if you make use of them do not forget the new password you have selected). If your initial password is 'plumber', type it and you will then have

   Password: . . . .

   Note that the password that you typed is not shown on the screen. If you make a mistake you will be asked to **Login** again.
4. If you are successful in completing steps 1–3 the rest of the screen comes into use. At the bottom of the **main display**, just above the display command line, you will see a **dollar ($) prompt**. Let us:

   (a) **adjust the screen display mode**. The monochrome Apollos will present either white lines on a black background or black lines on a white background.

   To change between modes, place the cursor on the command line (bottom line). Then use the command **inv (invert)**:

   command: inv

   and when you press **return** the display will change. (For these notes I will assume that you have selected black lines on white background. Consequently, the description **inverse video** will always imply a black background, also referred to as **enhanced**.)

   (b) **see what directories there are** in your name. Your **user name** (apprentice) and **password** (plumber) have given you access to a working area, which may be compared to a large, multi-drawer, private filing cabinet, having the ability to grow to suit your demands. You could then liken a directory to a drawer within that filing cabinet. You can create new directories as appropriate.

   At the **$** prompt use the command **ls (list)**. NB: use lower case letters.

   $ ls

   You will then see a list of the directories at present in your working area.

   (c) **make a working directory** (to put your MINNIE work in). Use the (lower case) command **mkdir ⟨directory-name⟩** (make a directory called, say, kitchen)

   $ mkdir kitchen

   (NB: you only need to make a working directory if you do not have one from a previous use).

   (d) **check that the directory was created**. Repeat step (b) and see what directories are listed.

   $ ls

kitchen should now appear in its place on the directory list.

(e) **start to work** in this new directory. Individual jobs – circuits etc. – may be allocated to different **files** within this directory, so you may store a lot of work in it.

Use the (lower case) command **cd** ⟨**directory-name**⟩ (change directory to, in this case, kitchen):

$ cd kitchen

(f) check that you are now in **kitchen**. Use the command **pwd**:

$ pwd

and you should see the full **path** (all the directory names) from the lowest (**root**) level of the operating system up to the level at which you are at present operating. The *last* item in the list is your **current directory** – **kitchen** in this case.

//serf/u/apprentice/kitchen

(g) **call** MINNIE. Just type **minnie** (lower case),

$ minnie

and wait for the MINNIE start-up screen to appear.

1.3 START UP SCREEN

When MINNIE appears you will see Fig. 1.3-1, containing

1. A right-hand section full of items and functions to be used for laying out your circuit.
2. A larger left-hand area for drawing your circuit. This area is covered with a regular array (grid) of points. (NB: this is only a small section of the total drawing area available).
3. A small black/white two-coloured square in the centre of the screen, on one of the grid points. This square, your **current position marker**, indicates your actual working point on the screen. Components will attach to the long edge of the black half of this marker.
4. A circle with a cross in it. This is your **cursor** or **pointer**. Note that in some modes it will change shape (a dagger, a pen, a magnifying glass, etc.).

   (If this cursor is not visible it may be in the drawing area off screen so move the MOUSE to bring it back into view).
5. At the top of the right-hand area, a **highlighted** (inverse video) box indicating that you are immediately in `DRAWING` mode. The alternative modes, to be selected as required, are `ANALYSIS` and `RESULTS`. In this area you will also find `EXIT`, `CAPTIONS` and `GLOBALS`.

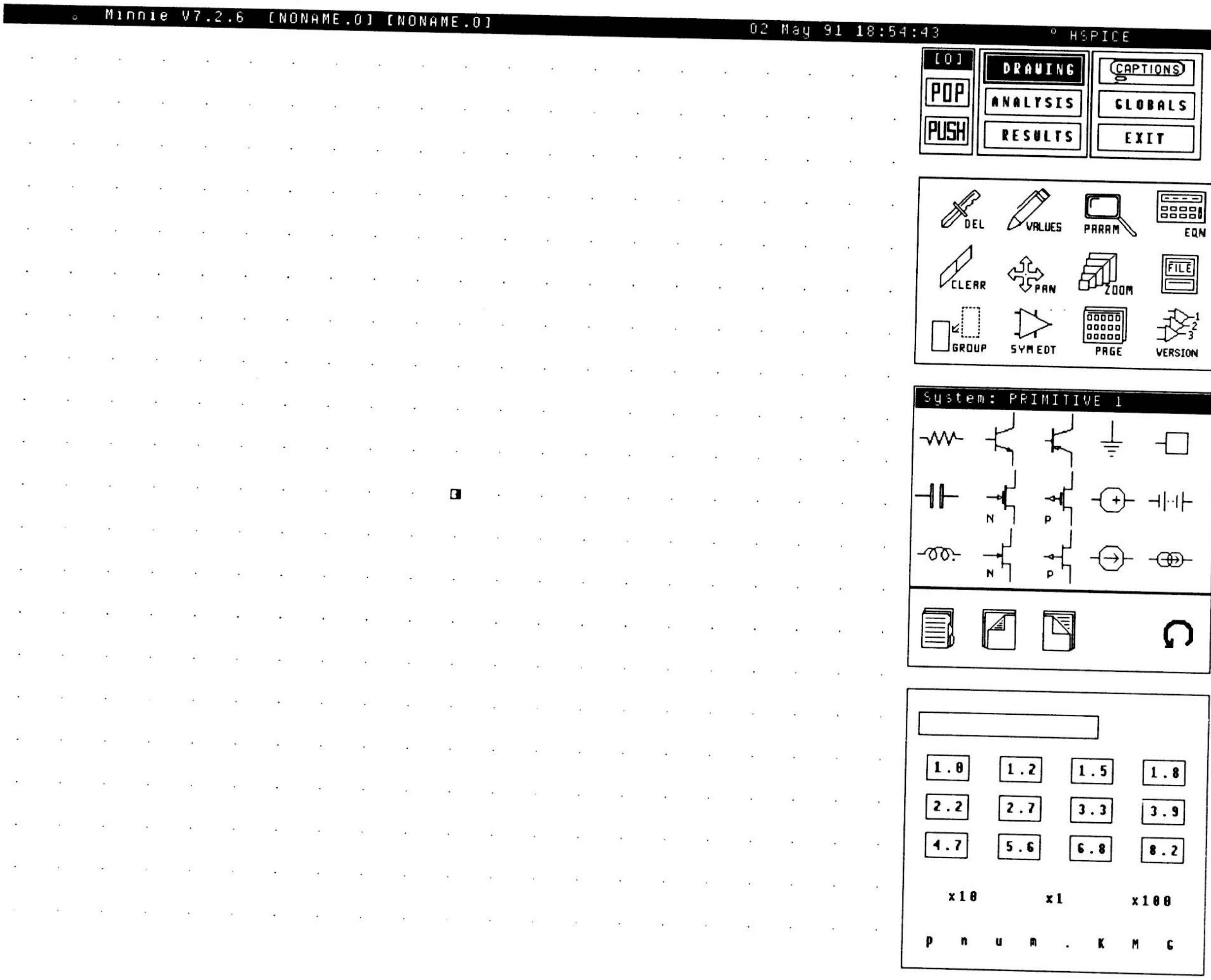

Fig. 1.3-1 Menu on the MINNIE start-up screen

1.4 PRINTING THE SCREEN CONTENTS

To obtain **hard copy** (print out) of your circuits and graphs it is necessary to have a copy of the screen image sent (**dumped**) to a printer.

The **control codes** used depend on the manner in which the local system is set up. The dump procedure may use the local operating system's **printer drivers** or drivers provided by ISL, the suppliers of MINNIE. The local system supervisory staff will know which are being used. The codes are:

	System drivers	ISL drivers
Full screen dump (Fig. 1.4-1)	control-F	control-W
Partial screen dump (Fig. 1.4-2)	control-D	control-V

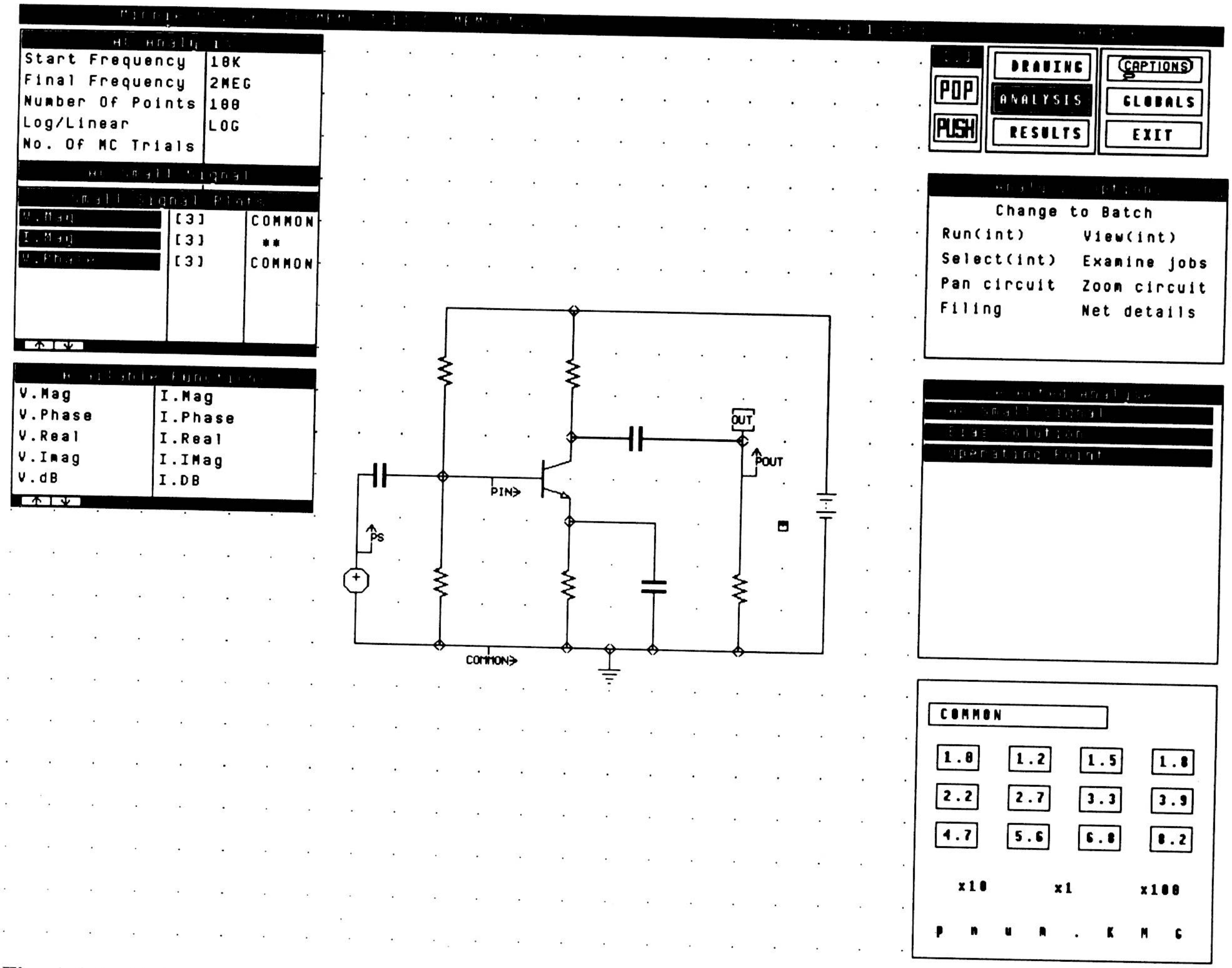

Fig. 1.4-1 Full screen dump: the right-hand menu is not part of the circuit or result presentation area, so it can often be discarded

in which **control-F** means hold down the 'control key' while typing 'F'. Note that **control-f** is the same.

1.5 DRAW YOUR CIRCUIT

Circuits are drawn by positioning components using a mouse. The mouse has three keys, which have the following functions:

- **L** left mouse key: rotates position marker
- **M** middle mouse key: all selection & deselection
- **R** right mouse key: cursor movement

1. **Cursor movement**: move the mouse, in contact with its mat so that the ball under it rotates, and notice how the cursor (the circle with a cross) moves in response to the mouse movements.

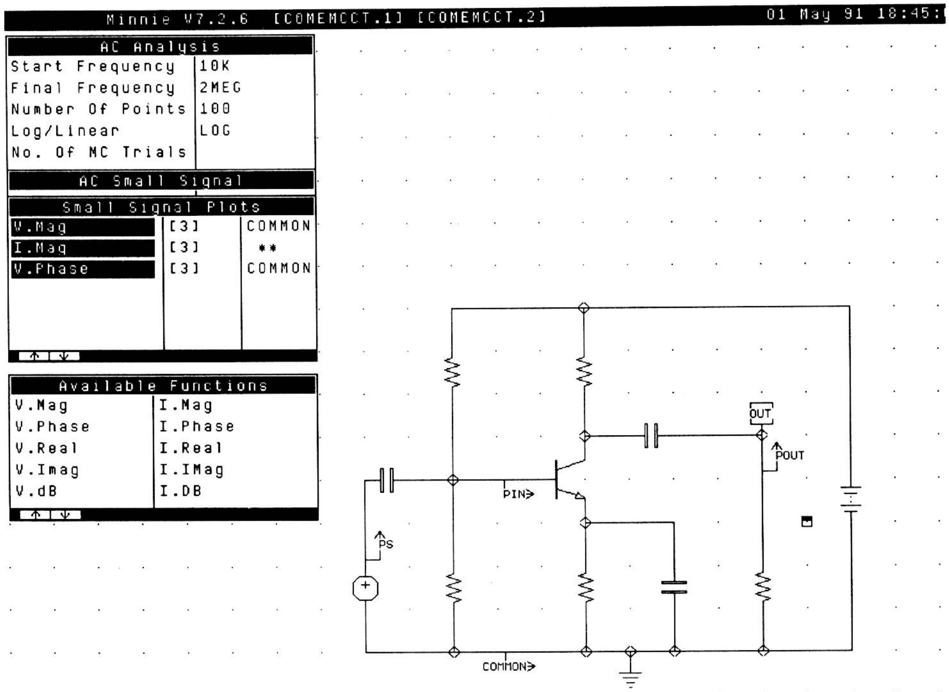

Fig. 1.4-2 Partial screen dump: when printed on the same size of paper the circuit diagram is 1.5 times larger than in the full screen dump

2. **Position marker rotation (component direction)**: position the cursor anywhere within the drawing area, and press the left mouse key **L**. Watch the current position marker, and notice how it rotates clockwise 90° on each press.

   You can use this technique to determine the direction along which a component will be positioned.
3. **Current position (component position)**. Keep the cursor in the drawing area, and press the middle mouse key **M**. See how the marker moves to the **grid point** closest to the cursor. Move the cursor elsewhere and repeat.

   You use this technique to determine the position of a component (which will grow from the grid point covered by the marker).
4. **Line (conductor) drawing**. Keep the cursor in the drawing area, and press the middle mouse key **M**, *holding it down*. Roll the mouse in all directions. Watch how it draws lines while the **M** key is held down. See how the reverse process can be used to **undraw** lines but only if you start from an *open* end. Notice also how these lines snap to the grid lines.

   These lines are your **electrical conductors**. It is important to note that an electrical connection is made *only* when one conductor terminates on another. Consequently if a conductor is drawn across a central portion of another no connection is made. If a connection