

WORLD METEOROLOGICAL ORGANIZATION

AGRICULTURAL METEOROLOGY PROGRAMME

CAGM Report No. 33B (Revised)

Practical Manual Series No. 2

**MANUAL ON USE OF PC, MS-DOS, EDIT AND CSMP
FOR SIMULATION OF PRIMARY PRODUCTION**

Prepared by

**The Centre for Agrobiological Research (CABO) and
The Department of Theoretical Production Ecology (TPE)
Wageningen, Netherlands**

WMO/TD-NO. 249 (Rev.)

Geneva, July, 1990

NOTE: This manual was prepared at the request of WMO for use as a supplementary text for its series of roving seminars on Simulation of Primary Production of Natural Pastures and is reproduced as submitted by the authors with their permission. It does not necessarily represent the views of the Organization.

WORLD METEOROLOGICAL ORGANIZATION

AGRICULTURAL METEOROLOGY PROGRAMME

CAGM Report No. 33B (Revised)

Practical Manual Series No. 2

**MANUAL ON USE OF PC, MS-DOS, EDIT AND CSMP
FOR SIMULATION OF PRIMARY PRODUCTION**

Prepared by

**The Centre for Agrobiological Research (CABO) and
The Department of Theoretical Production Ecology (TPE)
Wageningen, Netherlands**

WMO/TD-NO. 249 (Rev.)

Geneva, July, 1990

CONTENTS

Preface	iii
Instructions to the reader	iv
1. Use of the Personal Computer AT	1
2. The Operating System MS-DOS	5
2.1 Introduction	5
2.2 Commands	6
2.3 Files	6
2.4 Memory division	7
2.5 Referring to files	9
2.6 Path command	9
2.7 Operations on files	10
2.8 Disks	11
2.9 Formatting	11
2.10 Backing up	12
3. The editor	13
3.1 Introduction	13
3.2 Summary of EDIT	14
3.3 A short EDIT session	16
3.4 The screen layout	18
3.5 Two more EDIT modes	19
4. CSMP on IBM's or compatibles (PCSMF)	21
4.1 Every day use of PCSMP	21
4.2 The separate stages of PCSMP and intermediate files	22
Appendix I. Installation of PCSMP	25

PREFACE

During the last decades the application of simulation and systems analysis in agricultural research has increased considerably. The Department of Theoretical Production Ecology (TPE) and the Centre of Agrobiological Research (CABO) in Wageningen, the Netherlands, have contributed to this development and much of the insight, knowledge and methodology have been published in the series Simulation Monographs. One of these monographs describes a course taught by teachers of CABO and TPE in 1984: 'Modelling of agricultural production: weather, soils and crops'. Participants of the course were research workers in various developing countries. The course was sponsored and organized in close cooperation with the World Meteorological Organization (WMO) in Geneva, Switzerland.

As a result of this course, interest in simulation and systems analysis in agricultural research has increased in many developing countries. WMO asked CABO/TPE to give short courses on simulation of primary production in various countries. CABO/TPE is interested to teach such courses. The contents of the course is tailored to the level of beginners in this field and comprises various readers from books of the Simulation Monographs series. It is organized in such a way that after the course, participants have become acquainted with the approach and are able to use and develop elementary models on primary production under various circumstances. We hope that the participants will have a fruitful course and thank WMO for the initiative and willingness to sponsor.

H. van Keulen
CABO
Wageningen, the Netherlands

R. Rabbinge
TPE
Wageningen, the Netherlands

INSTRUCTIONS TO THE READER

During the roving seminar 'Simulation of Primary Production' on the use of crop-, soil- and agrometeorological data to assess primary production, organized by WMO (Geneva), participants will use personal computers for running simulation models.

Use this textbook as a user's manual while you work at the personal computer. To arrange this user's manual, the following literature has been used:

Dierkx, R.T., 1986. MS-DOS, a Disk Operating System. Internal report Department of Theoretical Production Ecology, Agricultural University, Wageningen, the Netherlands. (Chapter 2 of the manual).

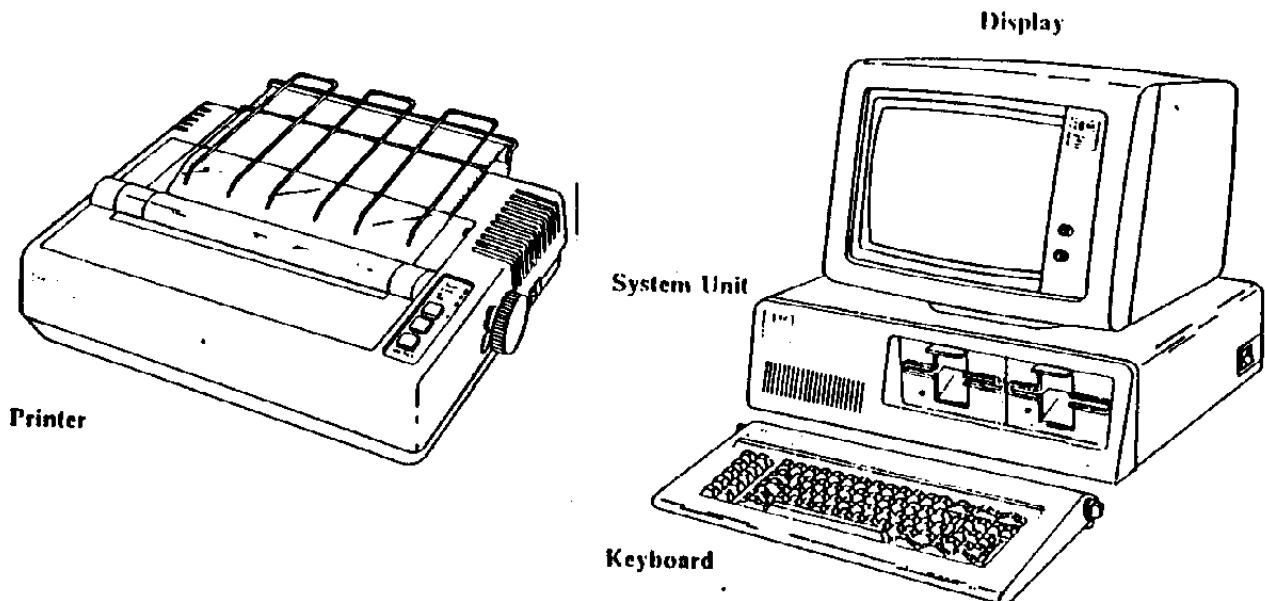
Dierkx, R.T., 1986. The IBM professional editor. Internal report Department of Theoretical Production Ecology, Agricultural University, Wageningen, the Netherlands. (Chapter 3 of the manual).

Dierkx, R.T. & H.H. van Laar, 1987. PCSMP on IBM's or compatibles. Internal report Department of Theoretical Production Ecology, Agricultural University, Wageningen, the Netherlands. (Chapter 4 of the manual).

H.J.W. van Roermund

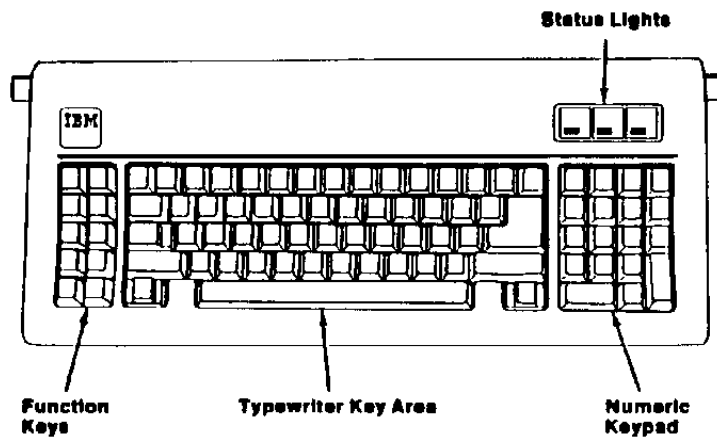
1. Use of the IBM personal computer AT

The IBM personal computer -AT or a compatible Personal Computer-AT (from now on just IBM PC-AT) consists of a system unit, a keyboard, a display and a printer.

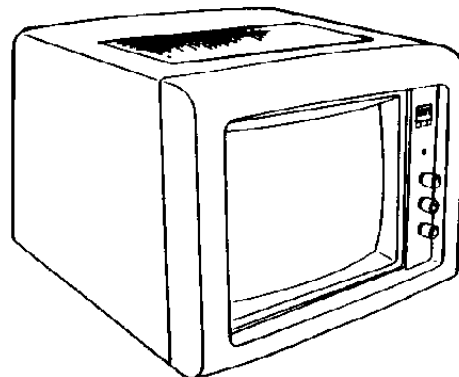


The system unit is the central part (the computer itself) of your IBM PC-AT. In the system unit, computer programs are run and calculations are made. It contains the hard disk drive (fixed disk drive, drive c:) and the mathematical coprocessor if installed.

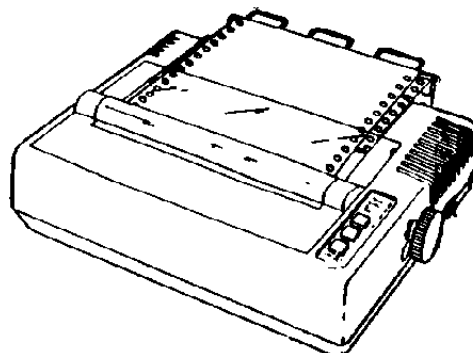
The keyboard is the primary input to the system unit (standard input): you are typing information (text) from the keyboard.



The display is the primary output of the system unit (the screen, VDU, standard output): information (text) is going from the system unit to the display. This is also called echo.



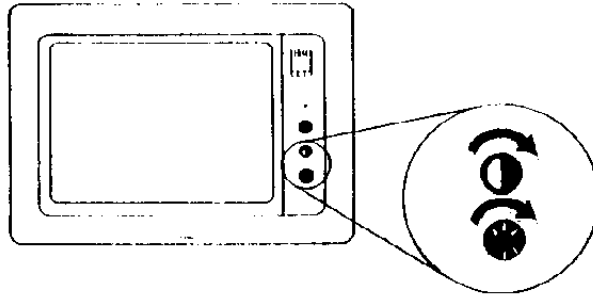
The printer is also output of the system unit: information is going from the system unit to the printer and will be printed on paper (if you first instruct the computer to do so).



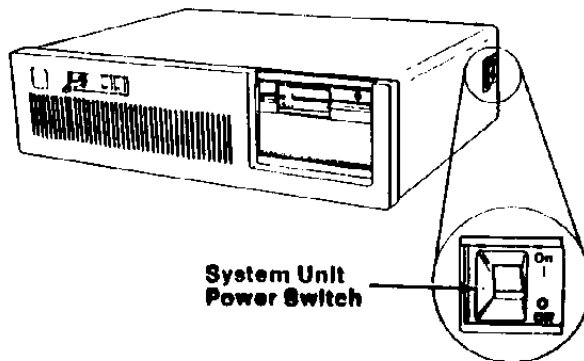
Because the hard disk is sensitive to changes in voltage of the electricity supply, use of an uninterrupted power supply (UPS) is recommended .

To turn on the IBM PC-AT, follow the next steps:

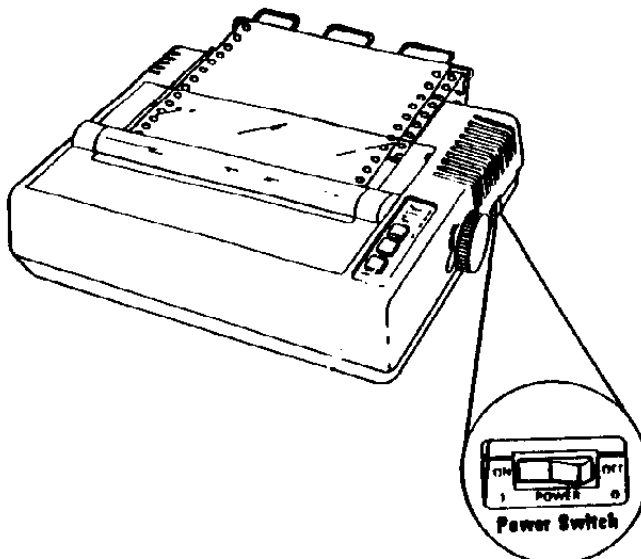
- turn on the UPS.
- turn on the display.



- turn on the system unit.



- If you want to print, turn on the printer.



The system unit (with hard disk and coprocessor), the keyboard, the display and the printer are called the hardware (machinery) of the IBM PC-AT.

The information you give the computer (computer program) is called the software of the IBM PC-AT.

A special computer program is used to handle the hardware. This program is called an Operating System (OS). The IBM PC-AT Operating System is MS-DOS (Disk Operating System).

When you want to calculate something with the computer, you have to instruct the computer by means of a computer program. A computer program is written in a certain language, so the computer knows what it has to. Examples of some computer languages are: BASIC, FORTRAN, PASCAL, CSMP. The computer language we use to simulate biological systems is CSMP (Continuous System Modelling Program)

A file is a collection of data's, for example a text or a computer program, which is stored on the hard disk (disk c:) or on a floppy disk (diskette, disk a: or disk b:). You can change the text of a file by a computer program, called EDITER, such as EDLIN, EDIT or WORDSTAR. The editor we use on the IBM PC-AT is the IBM Professional Editor EDIT.

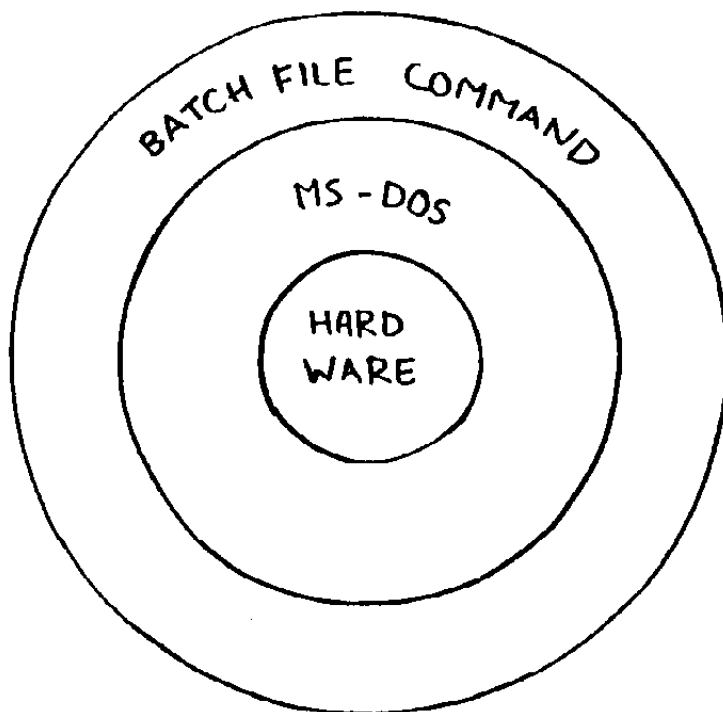
2. THE OPERATING SYSTEM

2.1 INTRODUCTION

On personal computers like the IBM PC-AT the MS-DOS is the most commonly used Disk Operating System (D.O.S.).

WHAT IS AN OPERATING SYSTEM ?

An O.S. is a program which hides the hardware from the user. The user only has to have knowledge of the O.S. to handle the machine.



2.2 COMMANDS

When you power up, DOS answers with its PROMPT, it will wait patiently for you to enter your command.

The prompt looks like: C:\>

The prompt means something like "tell me what you want me to do".

DOS provides a set of commands which you may enter. When you enter a command, DOS takes care of all things which have to be done to perform that command.

Eg the command: TYPE <file_name>

This command shows the contents of the specified file on the screen.

You type in: TYPE fred.txt

Then you hit the 'Enter' key

and fred.txt will be shown on the screen.

2.3 FILES

A file is a collection of data items of any kind.

Examples:

- text
- source code of a programming language
- data base information
- binary code for a certain processor

A file has a name and an extension,

syntax: <file_name>.<extension>

eg: fred.txt (no spaces allowed !)

The extension normally gives information on the contents of the file (eg: .txt or .dat or .csm).

2.4 MEMORY DIVISION

WHAT ARE TREE STRUCTURED DIRECTORIES?

This means that the memory is divided into logical areas.

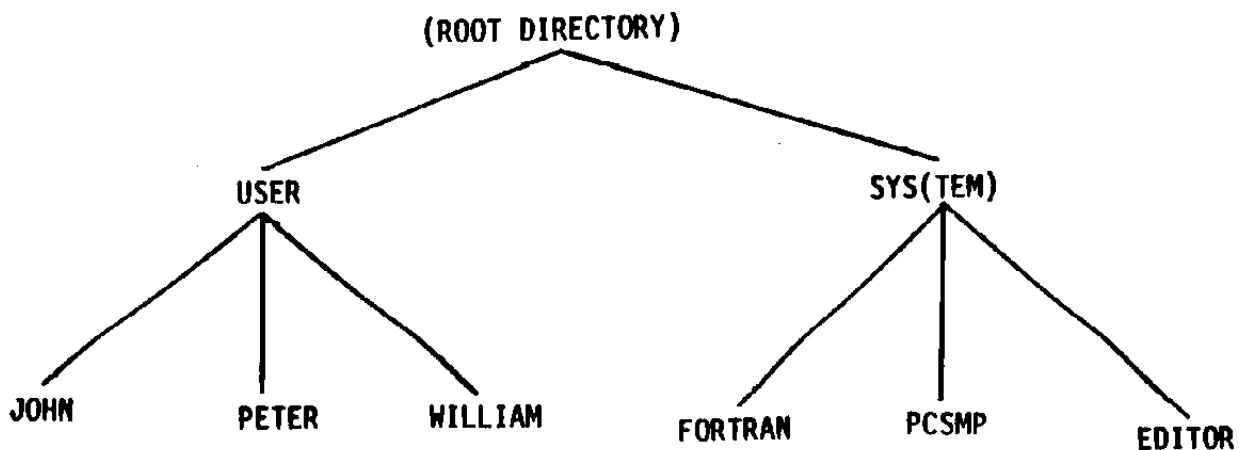
Such an area is called a DIRECTORY. The directories are interconnected in a tree structure.

A directory may contain

- files
- connections to other directories

The advantage of tree structured directories is: a logical division of your information.

TREE STRUCTURED DIRECTORIES AN EXAMPLE



TREE STRUCTURED DIRECTORIES: SOME COMMANDS

To use a tree structure you should be able to:

- add 'branches' (directories)
- remove 'branches'
- move about in the 'tree'
- show the files and connections to the other directories in any directory

Here are the commands you need:

MD <dir_name>
creates a new directory (a 'branch')

RD <dir_name>
removes a directory

CD <dir_name>
changes to another directory

DIR
shows the contents of a directory

An additional command is TREE. TREE will give you an overview of the complete directory tree.

DIRECTORY NAMES

How do you refer to a directory ?

By calling its name, eg: \USR\JOHN,
refers to directory JOHN,
which is a subdirectory of USR, (parent directory)
which is a subdirectory of \

Reserved identifiers are:

root directory	\
current directory	.
parent directory	..

2.5 REFERRING TO FILES

Current directory

When you enter the command `TYPE fred.txt`
you refer to `fred.txt`
`fred.txt` will only be found if it is in the current
directory, that is the directory in which you are
currently working.

Other directory

If `fred.txt` is in another directory, you have to
specify a `PATH` to refer to `fred.txt`,
so `TYPE <path> <file_name>`

eg. `TYPE \USR\PETER\fred.txt`

2.6 PATH COMMANDS

In DOS it is possible to specify a search path with
the command `PATH`.

A search path is a list of paths which DOS follows
in search of a file or command.

DOS remembers this search path during the session.

Eg.: if you enter the command `PATH \USR\PETER`
you have specified a search path.

If you now enter the command `TYPE fred.txt`,
DOS first searches for `fred.txt` in the current
directory, and if `fred.txt` is not found, it
searches in directory `\USR\PETER`

2.7 OPERATIONS ON FILES

What are the things I can do with a file ?

There are four major commands:

COPY <original_file_name> <new_file_name>
to make a copy of your file

REN <old_file_name> <new_file_name>
to give your file a new name (rename)

DEL <file_name>
to delete your file from the memory

TYPE <file_name>
to show the contents of your file on the screen

If a printer is connected to your PC, you can get a hard copy of your file by entering the command

PRINT <file_name>

2.8 DISKS

A disk is the data storage medium. A disk drive is the hardware and the mechanics which perform the storing operation.

DOS uses several disk drives, so they have to be identified:

drive		drive_ID
floppy disk drive		A:
floppy disk drive		B:
hard disk		C:
virtual disk		D:

To select a drive, enter its ID, so if you enter

A: , then A:\>

appears on the screen.

This means that DOS is currently at drive A:

A: is the command you gave DOS, and it meant:

" go to floppy drive A: "

2.9 FORMATTING

Before you can use a floppy disk, it needs to be initialised. This is called formatting, and it is done with the command: `FORMAT`

Eg. `FORMAT A:` or

`FORMAT A:/v`

`/V` is a parameter which you pass to the `FORMAT` command. This gives you the possibility to give a name to the floppy disk.

2.10 BACKING UP

Making back ups is to make sure that you can recover your work in case something goes wrong completely. People always make mistakes, not on purpose, but by accident. Even experienced programmers do. ('Errare humanum est') That is the main reason to have a back up.

There are several ways:

1. make a back up of your file on the same disk
(short time use)
2. make a back up of your file on another disk
(physical separation)
3. make a back up of all files at once
(system back up)

Eg. 1. COPY fred.txt oldfred.txt
2. COPY c:\fred.txt a:\fred.txt
3. The BACK UP (and RESTORE command)

Back up regularly. If you back up each day, only one days work is lost if something goes wrong. Always take good care of your back up, that is both physically and logically.

Physical:

floppy disk cannot stand heat, dust (cigarettes), magnetic field (all electronic equipment) and severe stress (always put it in a solid box)

Logical:

register carefully and completely which files are kept on your back up media (disks), date, owner, and any other item which might be of importance.

3 THE EDITOR

3.1 Introduction

WHAT IS AN EDITOR ?

An editor is a utility.

WHAT IS ITS FUNCTION ?

- insert something in a file
- delete something in a file
- change something in a file

A file is a collection of data items of any kind

Most likely a file will contain:

- text
- source code of a programming language

WHAT KIND OF EDITORS EXIST ?

Editors come in many forms, depending on

- type of file you want to edit (ASCII, NON-ASCII)
- price you want to pay
- system on which you are working (considering speed, available memory etc)

With the MS-DOS there are many editors. The standard editor is named EDLIN, a line oriented editor.

The editor to be described is called The IBM Professional Editor,
in short: EDIT

3.2 Summary of 'EDIT'

EDIT

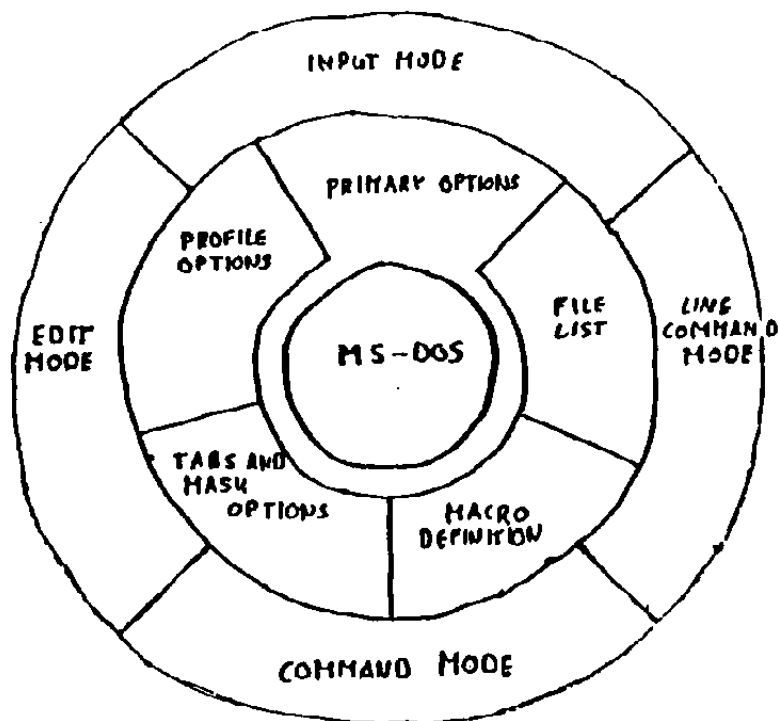
EDIT is a combined line/screen editor which only works on files which contain ASCII code (readable), so EDIT can be used for:

- text
(eg documentation, articles)
- program development
(eg CSMP source code)

EDIT contains

- a) 5 menus in which
 - the characteristics of EDIT may be defined
 - the subjected file is to be defined
- b) 4 edit modes,
that is four ways of approaching the contents of the file you are editing

THE LAYOUT OF EDIT



THE MENUS

1. PRIMARY OPTIONS MENU

This menu contains the basic settings

2. PROFILE OPTIONS MENU

This menu contains settings which are saved in a separate file, the profile file. With each edit session this file is read.

3. TABS AND MASK OPTIONS MENU

4. MACRO DEFINITION MENU

5. FILE LIST MENU

THE EDIT MODES

In each mode you may insert/delete/change something.

1. EDIT_MODE : characters
2. INPUT_MODE : lines
3. LINE_COMMAND_MODE : groups of lines
4. COMMAND_MODE : complete file

3.3 A SHORT EDIT SESSION

PRIMARY OPTIONS MENU.

items of major importance in this menu are:

INPUT FILE

The file which EDIT takes as input data

OUTPUT FILE

The file to which EDIT will write the result of the edit session

PROFILE FILE

The file which contains the characteristics of EDIT during the session

INPUT MODE

In input mode, you may insert, delete and undelete lines in the file.

The place of insertion is denoted by means of a tag which looks like this:
INSERT>>

Function keys involved:

05	"INSERT"
06	"DELETE"
16	"UNDELE"

INSERT: enter this mode by pressing function key 05, labeled "INSERT". To exit enter a null line or Cursor Up or Cursor Down

DELETE: place the cursor at the to be deleted line. Press "DELETE". The line disappears

UNDELETE: for recovering the last deleted line. Press "UNDELE". The line reappears

EDIT MODE

In EDIT MODE you
insert,
delete (-) undelete,
overwrite (-) restore characters.

Insert press Ins key and type
 your text
 exit: press Ins key
 again

Delete press Del key

Undelete press Esc key
 In this way deleted
 characters are recovered

Overwrite just type your text.
 Any previous text is
 overwritten

restore field press CTRL End key
 You recover overwritten
 characters

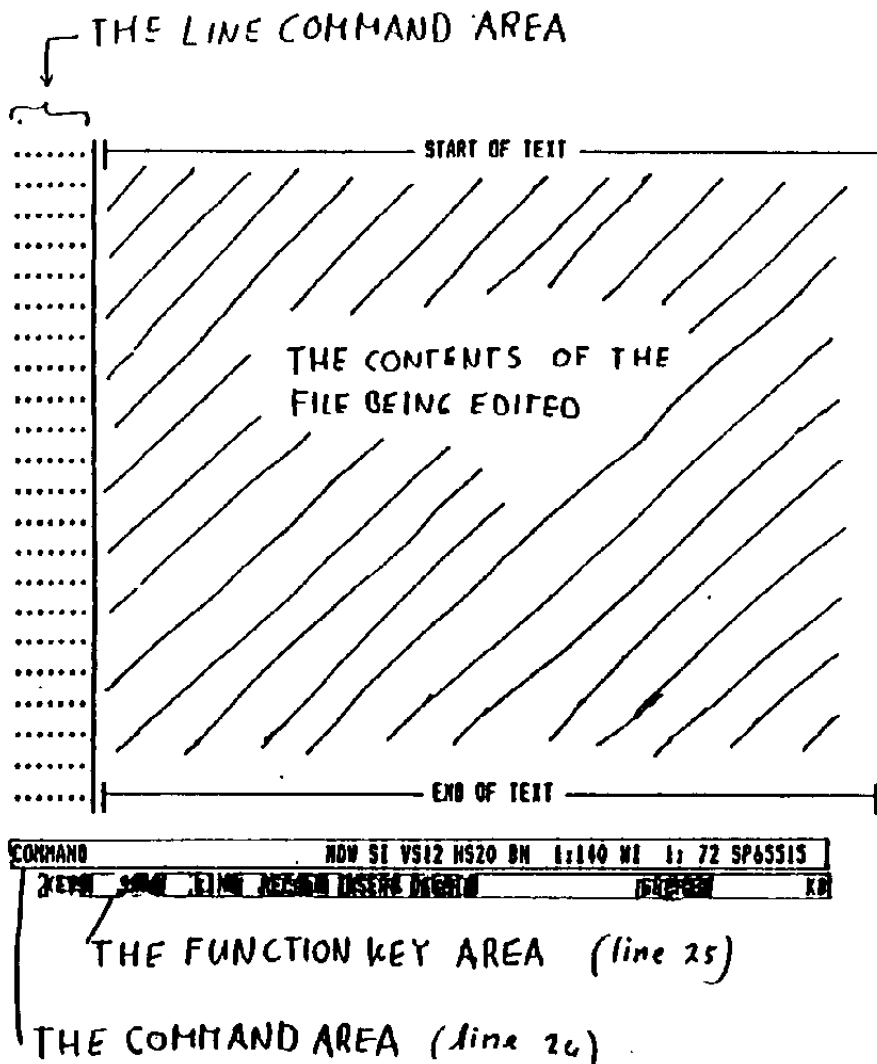
3.4 THE SCREEN LAYOUT

The LINE COMMAND AREA (column 1 to 7) is used for entering line commands

The COMMAND AREA (line 24) is used to enter commands

The FUNCTION KEY AREA is used to display the labels of the function keys

What is left over, a square of 72 characters by 23 lines, is used to display the contents of the file



3.5 Two more edit modes

LINE COMMAND MODE

Line commands affect groups of lines. They are being typed in the dotted area on the left hand of the display

syntax: <command>[count_operand]

The count operand denotes the number of lines affected by the command, starting from the current line

example: D25
 means : delete 25 lines starting
 from the current line

Another way to denote a range of lines is using two character commands

example: DD...! blablabla
 ! fhehnnshbdfvdfv
 DD...! dvddvgffrf

function keys involved with line command mode are:

03 "LINE" to get into line
 command mode
07 "CANCEL" to cancel a line
 command

LINE COMMAND MODE ; summary

CC	C	copy lines
CCR	CR	copy lines repeatedly
DD	D	delete lines
EE	E	extract lines (store them in another file)
GG	G	gather lines (make one line out of more lines)
MM	M	move lines
PP	P	print lines
LL	L	translate lines upper>lowercase
UU	U	translate lines lower>uppercase
COL		display column grid
A		AFTER CURRENT LINE
B		BEFORE CURRENT LINE

COMMAND MODE

The commands entered in the command mode affect the complete buffer (or file)

Two function keys are involved :

O2 "SWAP" get into command mode

O4 "RECALL" recall a previously entered command

Commands may be entered at the 24th line of the screen behind the prompt:

COMMAND->

This prompt appears when pressing "SWAP"

The most important commands are:

SEARCH search a string
CHANGE change a string
NEXT write lines from the buffer to
outputfile and read lines from
input file into buffer
EOPEN open extract file for output
ECLOSE close extract file
MERGE merge the contents of a file
inbetween the text

Examples: S /name/
C /name/newname/
EOPEN filename
MERGE filename

4 PCSMP ON IBM PC'S OR COMPATIBLES

4.1 EVERY DAY USE OF PCSMP

An example demonstrates the every day use. A model for exponential growth is simulated:

1. power up; wait until prompt appears
2. C: <enter> {go to disk C:}
3. change the search path if this is not already done in the file AUTOEXEC.BAT.
type: PATH <enter>
on the screen will appear PATH=.....<text>.....
now type: PATH \CSMP;.....<text>.....
4. CD \USR <enter> {go to the users area, not necessary when using the supplied AUTOEXEC.BAT}
5. PCSMP <enter> {invoke CSMP; P stands for personal computer}
6. wait for prompt 'INPUT FILE : '
7. EXPONENT.CSM <enter> {input file}
8. when translation, compilation, linking and execution were successfully completed, wait for DOS prompt

At this stage 3 more files are present on \USR,
FOR06.DAT - contains simulation results
FOR03.DAT - contains translator output listing
UPDATE.LST - contains a line-numbered UPDATE.FOR

[Note: step 2-4 can be omitted when using the supplied AUTOEXEC.BAT.

4.2 THE SEPARATE STAGES OF PCSMP AND INTERMEDIATE FILES

Invoking just PCSMP causes translation, compilation, linking and execution to take place. Intermediate files are deleted. Only the files FOR03.DAT, FOR06.DAT and UPDATE.LST are saved.

A detailed description:

Personal Computer Continuous System Modeling Program.
Executes a command procedure with the following phases:

- Translator phase : -translates your input file into two FORTRAN subroutines called UPDATE.FOR and DATA.FOR
- Compiler phase : -error messages in FOR03.DAT
- : -compiles the FORTRAN subroutines
- : -error messages in UPDATE.LST
- Linker phase : -links them with other PCSMP routines into the executable image UPDATE.EXE
- Execution phase : -runs the image UPDATE.EXE
- : -error messages in FOR06.DAT or UPDATE.LST

With special options it is possible to skip one or more stages, or to save intermediate files.

Options:

Spaces between file-specification and options are obligatory. Possible options are:

- C : start with compiler phase.
- L : start with linker phase
- E : start with execution phase.
- S : save all intermediate files.

Example: PCSMP C <enter>

The translation phase now is skipped; compilation is started immediately. However, the compilation stage requires input; (UPDATE.FOR and DATA.FOR), so the first time you have to run PCSMP with the option S.

Format:

PCSMP [option]

When C, L, E or S - option is specified, all the files necessary to make a restart from compilation, linker or execution phase are saved. These files are otherwise destroyed.

By using the option C you can modify UPDATE.FOR after a run and rerun the program with the command " PCSMP C ".

By using the option L you can start again from the linker phase, with the command " PCSMP L ".

By using the option E you can modify CONTROL.SYS after a run and rerun the program with the command " PCSMP E ".

Translator-phase

The translator phase produces the following files:

- FOR03.OAT : containing a copy of the input file, error messages in case errors are detected and some statistics.
 - UPDATE.FOR : the fortran program, into which the structure statements of the input file are translated.
 - DATA.FOR : containing the values of some common variables.
 - CONTROL.SYS : containing initial values, execution control and output control statements.
 - TABLE.TMP : containing the dimensions of arrays.
- All versions of these last four files are deleted by the command procedure, unless an option is specified.

Compiler phase

The compiler phase compiles the FORTRAN-files UPDATE.FOR and DATA.FOR, resulting in the files UPDATE.OBJ and DATA.OBJ. Also a list-file of UPDATE.FOR is made, called UPDATE.LST, containing the program of UPDATE.FOR with line numbers and paging.

Linker phase

The linker phase links the object-files MAIN2.OBJ, UPDATE.OBJ and DATA.OBJ and puts the result in the file UPDATE.EXE, which is executed during the execution-phase.

Execution phase

The execution phase produces the file FOR06.DAT containing the execution output requested by CONTRO.SYS. After execution all intermediate files produced are deleted except FOR03.DAT, FOR06.DAT and UPDATE.LST, unless you specified any of the possible options C, L, E or S. In case you want to run your program with other parameter values or constants, you can change these values in the file CONTRO.SYS and then use the option E. You should follow the next procedure:

1. PCSMP S
2. input file: example.csm
3. now the normal computation procedure follows, resulting in the files:FOR03.DAT,FOR06.DAT and UPDATE.LST.
4. change the parameter values in CONTRO.SYS
5. PCSMP E
6. new files FOR03.DAT,FOR06.DAT and UPDATE.LST are produced.

!!BEWARE!! the files of no. 3 are overwritten by the ones of no. 6, so if you are interested in both outputs you better rename the files of no. 3 first before you make reruns.

APPENDIX I
INSTALLATION OF PCSMP

1. DESCRIPTION OF THE DISKETTE CONTENTS

PCSMP is a version of CSMP for use on IBM PC's or compatibles. The program is not meant for distribution. The present version 4.0 is suited for PC's with the following specifications:

512 Kb working memory, but preferably more
10 Mb hard disk or up
1 diskette drive 360 Kb or 1.2 Mb
DOS 2.11 or later versions

A coprocessor is not necessary, but the performance of PCSMP is faster if a 80287 mathematical coprocessor is added to the CPU.

The PLOTCSM program that enables plotting of output made while running a PCSMP program, is available for PC's with a HERCULES MONOCHROME GRAPHICS CARD (or compatible). To enable plotting on the screen, also a MONOCHROME MONITOR is needed. A graphics printer (e.g. EPSON FX-85 or LX-800) is required to obtain hard-copies of the plot-output.

The plot program is not required for adequate running of PCSMP on the PC, it merely makes its output more enjoyable. Thus, even if the system in use does not fulfill the plot requirements, the user can still run PCSMP.

PCSMP is available on two high density (1.2 Mb) diskettes or four (360 Kb) double density diskettes.
The files on the diskettes are:

'FORTRAN' diskette(s):

FOR1.EXE
FOR2.EXE
FOR3.EXE
IBMFOR.LIB
LINK.EXE
LIB.EXE
8087.COM
AUTOEXEC.BAT
ANSI.SYS
CONFIG.SYS
EMULATOR.LIB
8087ONLY.LIB
SUBR.LIB

'PCSMP' diskette(s):

PLOT.BAT
C8087.FOR
PCSMP.BAT
STOP.BAT
README.TXT
SUCROS.CSM
PLOT.TXT
TPLGSA.DAT
PCSMPFIG.EXE
SR0M.FNT
C8087.EXE
STPL.FNT
TRANS.EXE
MAIN2.OBJ
ZLINK.BAT
PLOTCSM.EXE
EXECUT.LIB

1.1 FILE DESCRIPTION

FOR1.EXE, FOR2.EXE, FOR3.EXE

Together these files are the IBM MS-DOS FORTRAN compiler version 2.0. This compiler supports the subset of the FORTRAN-77 ANSI standard. It is a 3 stage compiler. The first (FOR1) and second (FOR2) stage generate the object code, the third stage (FOR3) generates an assembly listing. In everyday use of PCSMP, this third stage is not needed.

IBMFOR.LIB

This is the FORTRAN library. It is constructed in such a way, that if a coprocessor is available and enabled, most of the calculations are performed by the coprocessor. If not present or disabled, the coprocessor is emulated in software.

LINK.EXE

The MS-DOS linker, version 2.10. It is a one stage linker. Therefore the order of presenting the object code and libraries is important. In the batch file PCSMP.BAT the correct order is provided.

LIB.EXE

The FORTRAN library manager. It enables combining of OBJECT MODULES into an OBJECT LIBRARY. It is used during the creation of a user-made subroutine library.

8087.COM

Command file to switch the 80287 chip (if present) on or off.

AUTOEXEC.BAT

The general command procedure. One of its actions is to set the system path. This file should be present in the root directory.

ANSI.SYS

System file to control the video display. This file should be present in the root directory.

CONFIG.SYS

The system configuration file, which should be present in the root directory. It should contain the commands 'FILES=20', 'BUFFERS=40', 'DEVICE=C:\ANSI.SYS', 'FCBS=8,0', and 'BREAK=ON'.

EMULATOR.LIB

The MS-FORTRAN EMULATOR library. If a 80287 mathematical coprocessor is available (and enabled), most of the calculations are performed by this coprocessor. If not available or disabled, the coprocessor is emulated in software. If the 80287 chip is enabled, also the 80287ONLY.LIB could be used.

8087ONLY.LIB

MS-FORTRAN 8087ONLY library (see EMULATOR.LIB)

PLOT.BAT

The command procedure to invoke plotting of output generated by PCSMP. It should not be called on subdirectory C:\PCSMP, as it also deletes the files STPL.FNT, SROM.FNT, and TPLGSA.DAT (which are stored on C:\PCSMP for use on other directories). Protecting these files against deleting (with ATTRIB +R) will cause errors during plotting.

C8087.FOR

The FORTRAN source code for C8087.EXE. When another directory structure is used than default of the installation procedure, the indication of the path in the C8087.FOR has to be adapted accordingly. After compilation and linking of the adapted C8087.FOR, the newly made C8087.EXE should replace the existing file (by default on C:\PCSMP).

PCSMP.BAT

The command procedure controlling the process of translating, compiling, linking and executing.

STOP.BAT

A command procedure to cause ending of another command procedure with a beep to notify user.

README.TXT

File containing some information about PCSMP.

SUCROS.CSM

A Simple and Universal CROp growth Simulation model. This model in CSMP can be used to test the performance of PCSMP.

PLOT.TXT

File containing some information about PLOTCSM.

PCSMPFIG.EXE

The routine producing the PCSMP logo on the screen.

C8087.EXE

Executable file to make the linking procedure depending on whether the 808287 chip is enabled (linking with 8087ONLY.LIB) or not (linking with EMULATOR.LIB).

TRANS.EXE

The PCSMP translator from PCSMP to FORTRAN. During TRANS, the following files are made: FOR03.DAT, UPDATE.FOR, DATA.FOR, and TABLE.TMP. The latter three are used for further compilation and execution.

PLOTCSM.EXE

The routine producing plots. Invoked by PLOT.BAT.

MAIN2.OBJ

The PCSMP main program in OBJECT code. It takes care of a correct sequence of sections in the program made by the user. Also it updates the program variable TIME (depending on the method of integration) in this part. It always is LINKED before the OBJECT code of the program made by the user. This is done by commands in PCSMP.BAT.

ZLINK.BAT

Batch file to facilitate linking with right subroutine library.

EXECUT.LIB

The PCSMP library, containing OBJECT code modules for various subroutines and functions that are typical for PCSMP.

! NOTE: CONFIG.SYS, ANSI.SYS and AUTOEXEC.BAT should be present in the root directory (C:\).

2. INSTALLING PCSMP

Conditions:

- some acquaintance with DOS 2.11 or compatible
- hard disk available with over 2 Mb free space
- default startup from hard disk (operating system resides on hard disk)
- hard disk identification is C:
- 5 1/4" diskette drive present, with identification A:

Sequence of actions,
execute all **bold** instructions:

1. Power up: wait until prompt appears
2. **CD**
3. Check whether the directories **C:\SYS\PCSMP**
C:\SYS\FORTRAN
C:\SYS\MACROS\SUBR
and **C:\USER** exist.

if not, create these directories.
(for commands, see chapter 2.4)
4. **CD**
5. Insert the 'FORTRAN' diskette(s) in drive A:
6. **COPY A:*.* \SYS\FORTRAN/V**
7. Insert the 'PCSMP' diskette(s) in drive A:
8. **COPY A:*.* \SYS\PCSMP/V**
9. **COPY \SYS\FORTRAN\SUBR.LIB \SYS\MACROS\SUBR\SUBR.LIB**
10. **COPY \SYS\FORTRAN\AUTOEXEC.BAT C:\AUTOEXEC.BAT**
11. **COPY \SYS\FORTRAN\CONFIG.SYS C:\CONFIG.SYS**
12. **COPY \SYS\FORTRAN\ANSI.SYS C:\ANSI.SYS**
13. Reset (!!!) the system: power off/on or CTRL_ALT_DEL
14. **DEL \SYS\FORTRAN\AUTOEXEC.BAT**
15. **DEL \SYS\FORTRAN\CONFIG.SYS**
16. **DEL \SYS\FORTRAN\ANSI.SYS**
17. **DEL \SYS\FORTRAN\SUBR.LIB**
18. **CD C:\USER**
19. **COPY \SYS\PCSMP\SUCROS.CSM \USER\SUCROS.CSM**
20. Run the SUCROS.CSM model to test the performance of PCSMP on your PC:

PCSMP SUCROS.CSM D (example to execute a PCSMP-program)
21. The PCSMP-program SUCROS.CSM is now running ! PATIENCE !
Results can be found in the file FOR06.DAT (tabular)
or by using the PLOT command (plotted output).
Errors can be found in the files FOR03.DAT and/or
UPDATE.LST

3. IN CASE OF TROUBLE

- a. Problems concerning file PCSMP.BAT may be made visible by changing the first line
ECHO OFF
into
:ECHO OFF

and remove the phrase '>PCSMP.BAT' (three occurrences)

A person familiar with DOS should now be able to recognize problems.

- b. When you are desperate, because you cannot trace the error, you can run PCSMP step by step:

```
TRANS<enter>
FOR1<enter>
    UPDATE<enter>
    <enter>
    UPDATE<enter>
    <enter>
FOR2<enter>
FOR1<enter>
    DATA<enter>
    <enter>
    <enter>
    <enter>
FOR2<enter>
LINK<enter>
    \CSMP\MAIN2+UPDATE+DATA<enter>
    UPDATE<enter>
    <enter>
    \CSMP\EXECUT+\CSMP\FORTRAN<enter>
UPDATE<enter>
```

- c. For more information on error messages during the various stages of PCSMP, consult the installation manual:

PCSMP on IBM PC-AT's or PC-XT's and compatibles.
Jansen, D.M., R.T. Dierkx, H.H. van Laar & M.J. Alagos
Simulation Report CABO-TT nr. 15.

or the CSMP reference manual:

Continuous System Modeling Program III (CSMP III),
Program Reference Manual. 206pp.
IBM SH19-7001-3
Techn. Publ. Dept., White Plains, USA.