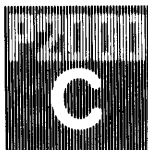




PHILIPS

CP/M REFERENCE MANUAL

CP/M is a trademark of Digital Research Inc., Pacific Grove, Cal.



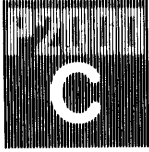
Preface

PREFACE

This Reference Manual is provided for the use of programmers intending to develop and install their own applications on the P2000C in the CP/M environment. The companion manual, the P2000C CP/M User Guide (993 21921), provides an introduction to the operating system and to the configuration program. This program also includes options for editing various internal tables for the control of peripheral devices. These are described in this manual.

The manual consists of two parts: Part A, which is concerned with the implementation of CP/M on this computer; and Part B, which contains original Digital Research documentation, and which provides detailed information on various aspects of the operating system. A table of contents for both parts of the manual follows this preface.

The System Reference Manual (993 30421) for the P2000C may also provide useful information.



Contents

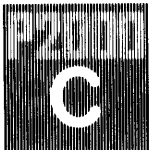
TABLE OF CONTENTS

	page
PREFACE	
TABLE OF CONTENTS	
<u>PART A : PHILIPS IMPLEMENTATION OF CP/M</u>	
1 INTRODUCTION	1-1
2 SUPPORTED HARDWARE	2-1
2.1 Keyboard and Video Display Unit	2-2
2.2 Disk Storage Media	2-3
2.3 Printer	2-4
2.4 Serial I/O	2-5
3 CONFIGURATION PROGRAM	3-1
3.1 Introduction	3-1
3.1.1 Working Copy of System Software	3-1
3.1.2 Purpose of CONFIG	3-1
3.2 Files	3-2
3.3 Displays	3-3
3.3.1 CP/M Environment Display	3-3
3.3.2 Runtime BASIC Display	3-4
3.3.3 Advanced BASIC Display	3-4
3.3.4 BASIC DS Display	3-4
4 OPERATION OF CONFIG	4-1
4.1 Calling the Program	4-1
4.2 Language Selection	4-1
4.3 Leaving the Program	4-1
4.4 Main Menu	4-2
4.5 System Configuration	4-3
4.6 System Generation	4-4
4.7 Disk Table Editing	4-5
4.7.1 Copy to Temporary Buffer	4-6
4.7.2 Edit Temporary Buffer	4-6
4.7.3 Save Temporary Buffer to Disk	4-7
4.7.4 Delete Table From Disk	4-7
4.7.5 Return to Main Menu	4-7



Contents

	page	
4.8	Keyboard Table Editing	4-8
	4.8.1 Keyboard Codes	4-8
	4.8.2 Select 1,3 and 4	4-9
	4.8.3 Edit Temporary Buffer	4-9
	4.8.4 Return to Main Menu	4-9
4.9	Video Table Editing	4-10
	4.9.1 Video Codes	4-10
	4.9.2 Select 1,3 and 4	4-10
	4.9.3 Edit Temporary Buffer	4-11
	4.9.4 Return to Main Menu	4-11
4.10	Printer Table Editing	4-12
	4.10.1 Printer Tables	4-12
	4.10.2 Select 1,3 and 4	4-13
	4.10.3 Return to Main Menu	4-13
	4.10.4 Edit Temporary Buffer	4-14
4.11	Notes on the Preparation of National Versions	4-17
	4.11.1 Example: '#' Translated to '£'	4-17
5	THE UTIL PROGRAM	5-1
5.1	Starting the UTIL Program	5-1
5.2	The Main Menu	5-2
	5.2.1 Format Floppy Disk	5-3
	5.2.2 Format Hard Disk	5-3
	5.2.3 Copy Floppy Disk	5-4
	5.2.4 Redefine Language/Disks	5-6
6	BACKUP FOR HARD DISKS	6-1
6.1	Using BACKUP.COM	6-1
7	CP/M 'ASM' COMMAND	7-1
8	CP/M 'LOAD' COMMAND	8-1
9	DISK INFORMATION	9-1
9.1	Five-Inch Disks	9-1
	9.1.1 Single Sided	9-1
	9.1.2 Double Sided	9-1
9.2	Hard Disk	9-2
10	ACCESS TO THE VIDEO CONTROLLER	10-1
10.1	Screen Specifications	10-1



Contents

	page	
10.2	Screen and Screen Codes	10-2
	10.2.1 Single Codes	10-2
	10.2.2 Escape Sequences, Set Attrib.	10-3
	10.2.3 Escape Sequences, Scrn. Control	10-4
	10.2.4 Description of Control Codes	10-5
	10.2.5 Graphic Control	10-8
	10.2.6 Description of Graphic Commands	10-9
10.3	Status Information	10-10
11	RESERVED LOCATIONS IN PAGE ZERO	11-1
12	SYSTEM MEMORY LOCATIONS	12-1
APPENDIX A	CHARACTER CODE TABLES	App A-1

PART B : DIGITAL RESEARCH CP/M 2 MANUALS

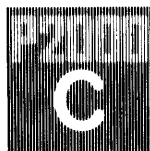
CP/M 2 USER'S GUIDE

1	AN OVERVIEW OF CP/M 2 FACILITIES	2/1
2	USER INTERFACE	2/3
3	CONSOLE COMMAND PROCESSOR INTERFACE	2/4
4	STAT ENHANCEMENTS	2/5
5	PIP ENHANCEMENTS	2/8
6	ED ENHANCEMENTS	2/10
7	THE XSUB FUNCTION	2/11
8	BDOS INTERFACE CONVENTIONS	2/12
9	CP/M 2 MEMORY ORGANIZATION	2/27
10	BDOS DIFFERENCES	2/28

Contents



	page
ED : CONTEXT EDITOR USER'S MANUAL	
1	ED TUTORIAL 3/1
1.1	Introduction to ED 3/1
1.2	ED Operation 3/1
1.3	Text Transfer Functions 3/1
1.4	Memory Buffer Organization 3/5
1.5	Memory Buffer Operation 3/5
1.6	Command Strings 3/7
1.7	Text Search and Alteration 3/8
1.8	Source Libraries 3/11
1.9	Repetitive Command Execution 3/12
2	ED ERROR CONDITIONS 3/13
3	CONTROL CHARACTERS AND COMMANDS 3/14
CP/M ASSEMBLER (ASM) USER'S GUIDE	
1	INTRODUCTION 4/1
2	PROGRAM FORMAT 4/2
3	FORMING THE OPERAND 4/4
3.1	Labels 4/4
3.2	Numeric Constants 4/4
3.3	Reserved Words 4/5
3.4	String Constants 4/6
3.5	Arithmetic and Logical Operators 4/6
3.6	Precedence of Operators 4/7
4	ASSEMBLER DIRECTIVES 4/8
4.1	The ORG Directive 4/8
4.2	The END Directive 4/9
4.3	The EQU Directive 4/9
4.4	The SET Directive 4/10
4.5	The IF and ENDIF Directives 4/10
4.6	The DB Directive 4/11
4.7	The DW Directive 4/12



Contents

	page	
5	OPERATION CODES	4/12
5.1	Jumps, Calls and Returns	4/13
5.2	Immediate Operand Instructions	4/14
5.3	Increment and Decrement Instructions	4/14
5.4	Data Movement Instructions	4/14
5.5	Arithmetic Logic Unit Operations	4/15
5.6	Control Instructions	4/16
6	ERROR MESSAGES	4/16
7	A SAMPLE SESSION	4/17
CP/M DYNAMIC DEBUGGING TOOL (USER'S GUIDE)		
1	INTRODUCTION	5/1
2	DDT COMMANDS	5/3
2.1	The A (Assemble) Command	5/3
2.2	The D (Display) Command	5/4
2.3	The F (Fill) Command	5/4
2.4	The G (Go) Command	5/4
2.5	The I (Input) Command	5/5
2.6	The L (List) Command	5/6
2.7	The M (Move) Command	5/6
2.8	The R (Read) Command	5/6
2.9	The S (Set) Command	5/7
2.10	The T (Trace) Command	5/7
2.11	The U (Untrace) Command	5/8
2.12	The X (Examine) Command	5/8
3	IMPLEMENTATION NOTES	5/9
4	AN EXAMPLE	5/10
CP/M 2 INTERFACE GUIDE		
1	INTRODUCTION	6/1
2	OPERATING SYSTEM CALL CONVENTIONS	6/3
3	A SAMPLE FILE-TO-FILE COPY PROGRAM	6/29
4	A SAMPLE FILE DUMP UTILITY	6/34
5	A SAMPLE RANDOM ACCESS PROGRAM	6/37
6	SYSTEM FUNCTION SUMMARY	6/46



Introduction

1 INTRODUCTION

This part of the manual contains a collection of information relating to the implementation of CP/M on the P2000C microcomputer. The configuration program is described in its entirety; it provides, for example, access to an internal table to store the codes generated by the keys in shifted and unshifted positions, which may be set as required. Similar tables are available for printer, video and disks.

The information contained in this manual should be sufficient to cover program development, whether under CP/M directly, or under one of the two BASIC development environments - the Advanced BASIC Interpreter (P2509), or the BASIC Development System (P2511).

The UTIL program is a format and copy utility for preparing floppy disks to run on the P2000C. The contents of hard disk volumes can be stored for security on 5.25" floppy disks with the BACKUP program.



Supported Hardware

2 SUPPORTED HARDWARE

The P2000C that you are using may be one or another of three models:

The P2010-1 has two floppy disk drives for single sided 5.25" disks of 160 kilobyte each.

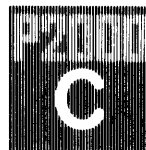
The P2010-2 has the same drives, and has in addition the IEEE 488 interface card.

The P2012 has two double sided disk drives with a capacity of 640K each.

In addition two more single or double sided disk drives and, for the P2010-1 and the P2012, in which the IEEE card is not installed, a 256K RAM disk may be configured under CP/M.

Standard on all three models are: the SASI interface for (up to two 10MB) hard disk drives, the high speed serial input/output interface and a video terminal containing its own processor for screen handling and graphics.

Supported Hardware

2.1 KEYBOARD AND VIDEO DISPLAY UNIT

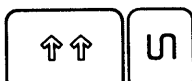
The codes generated by all of the character keys on the computer may be set in memory and preserved on disk by the 'Keyboard Table Editing' option of the configuration program (described in chapter 4 section 8). The keys on the function pad are pre-set to standard values for many CP/M based packages, (and for the screen handler utility supplied with the P2511 BASIC DS and the P2506 BASIC RTS).

In addition the code 80H is the capslock toggle, and is initially implemented with this key:



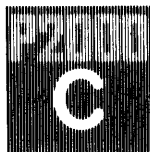
This key in its 'supershifted' position will, at any time print the contents of the screen. The code that produces this function is 90H.

print screen:



The printscreen facility may be inhibited by user programs by setting the byte at memory address 0008H to a non-zero value. On a 'cold boot' this location is set to zero.

The codes that control the characters displayed on the screen on the video display unit may be set in a similar way, via an internal table. For the procedure for setting these codes, refer to the 'Video Table Editing' part of the documentation on the configuration program (chapter 4 section 9).



Supported Hardware

2.2 DISK STORAGE MEDIA

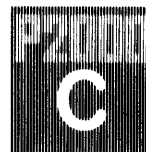
The disk from which the system is booted will be designated drive 'A' under CP/M. Any other logical disks may be given any designation from B to P. The arrangement of disks is described to the operating system by means of the disk table editing procedure, which is a part of the configuration program (chapter 4 section 7).

With regard to hard disk volumes: upper and lower volumes on a 10 megabyte disk may be set up as either 2 and 8 MB, or as 5 and 5. The hard disk is connected to the SASI board at the rear of the CPU.

A third and fourth floppy drive may be connected via the outlet at the rear of the unit.

The 256K RAM disk may optionally be fitted to your P2000C computer and may be connected to the main board inside the unit. For further details on the RAM disk refer to the System Reference Manual (993 30421).

Supported Hardware

2.3 PRINTER

Characters will be sent from the CPU to the printer according to the tables prepared for a particular printer. A detailed account of the structure of the printer tables and a description of the procedure for changing them is given in the section on 'Printer Table Editing' (chapter 4 section 10). An explanation of how national variable characters are treated is given in chapter 4 section 11.

The printer may be accessed via the logical LST: option, which is part of the CP/M 'STAT' command. This is possible if the I/O byte is set to LPT:. If the I/O byte is set to CRT: then the LST: will be echoed to screen. (Details of the STAT command are available in the User Guide and in part B page 2/5 of this manual).

If you issue a command to the printer and the device happens to be off-line, the system will display the following message, after a time-out period (which may be set by the configuration program.)

```
IGNORE PRINTER WITH ESC
```

Pressing any key other than 'ESC' will cause the system to try again to communicate with the device - after the same specified time-out period. The 'ESC' key allows you to continue, ignoring the error condition of the printer.

Printer speeds may be set up to 19200 Baud. The option is available within the 'Configure System' part of the configuration program (see chapter 4 section 5).

2.4 SERIAL I/O

The serial I/O port is accessible via BDOS PUNCH (for output) and READER (for input) calls. Described in part B page 6/9.

The port offers a fully duplexed communications interface at Baudrates of up to 19200.



Configuration Program

3 THE CONFIGURATION PROGRAM

3.1 INTRODUCTION

The Philips P2000C System Software, included on diskette with the machine, includes a basic system configuration. This may be sufficient for your immediate needs but, at some time in the future it may be necessary to add to or change some of the parameters. To make this possible, your System Software includes the program CONFIG and this section will describe the program in some detail as well as how to use it.

3.1.1 Working Copy of System Software

A word of warning seems appropriate at this point. It is possible to destroy information on a diskette, or at least make it impossible to retrieve. For this reason it is advisable not to run programs that may affect your System Software on the only copy you have. Before attempting to run CONFIG, make a copy of your System Disk, and always work on a COPY. Do this, either by using the method described in the CP/M User Guide, or with the 'copy disk' option of the UTIL program, (chapter 5 of this manual).

3.1.2 Purpose of CONFIG

What does the Configuration program do? Basically, it defines the equipment that is included in your system and 'takes note' of various operating parameters. This allows YOUR system to work to its' full operating capability. Any future add-ons, or changes, can be accomodated and can even be stored away on your System Software diskette 'just in case'.

Configuration Program



The configuration program, then, asks for information concerning your set-up and writes the information back into files on the System Software diskette. When you come to reload or RESET the system, the Initial Program Load (or IPL) will take account of the last configuration it was aware of. A change from the previous set-up may call for a RECONFIGURATION or, more likely, just a call for an alternative and previously defined system.

3.2 FILES

Not all the files on your System software diskette are required for system generation, but you will need the following configuration files:

CONFIG.COM: the normal executable CONFIG file
CONFIG.MSG: the text file
CONFIG.DAT: the tables

In addition to the configuration files, the system will require the CBIOS and CP/M files. The system generation will give you the largest possible memory for application programs, and has a choice of operating systems. For this reason, the following files should also be available at system generation:

CBIOS63.COM: 63K basic I/O system
CPM63.COM: 63K operating system
CBIOS62.COM: 62K basic I/O system
CPM62.COM: 62K operating system
CBIOS61.COM: 61K basic I/O system
CPM61.COM: 61K operating system

The last two of the above files will only be required in the very largest configurations, and may be omitted in normal use.

If you are working in the BASIC environment, the following file will also be required:

BASIC.COM or MBASIC.COM



Configuration Program

3.3 DISPLAYS

Your initial display on boot or RESET depends on the environment in which you are working, which may be either CP/M or BASIC. The names of the keyboard, video and printer tables depends on the current configuration.

3.3.1 CP/M Environment Display

```
IPL 1.XX
PHILIPS P2000C   63K CP/M 2.2

..welcome message..

KEYBOARD ASCII, VIDEO ASCII, PRINTER EMPTY

A:640K-FL1   B:640K-FL2

..result of any autostart program..
A>
```

This display informs the operator of the version of the IPL and the size and version of the CP/M system. After the optional welcome message it then gives details of the current keyboard, video and printer tables, followed by a summary of the current disk table.

Configuration Program



3.3.2 P2506 Runtime BASIC Display

```
PHILIPS P2506
BASIC interpreter run-time system.
Release 1.1
36056 Bytes free
..welcome message..
..result of any autostart program..
```

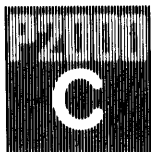
Ok

3.3.3 P2509 Advanced BASIC Interpreter

```
BASIC-80 Rev.5.21
(CP/M Version)
Copyright 1977,78,79,80 by Microsoft
32682 Bytes free
```

3.3.4 P2511 BASIC Development System Display

```
BASIC-80 Rev. 5.22
[CP/M Version]
Copyright 1977-1982 (C) by Microsoft
34831 Bytes free
```

Operation of CONFIG

4 OPERATION OF CONFIG4.1 CALLING THE PROGRAM

To call up the CONFIG program, enter:

```
CP/M environment: CONFIG(CR)
Basic environment: RUN"CONFIG(CR)
```

where (CR) represents the Carriage Return key.

While the program is being loaded, a 'Please Wait' message is displayed. Once loaded, the program automatically goes to 'Language Selection'.

4.2 LANGUAGE SELECTION

The display for this part of the program consists of an inverted line giving the name of the section in which you are working and the name, version and generation date of the program. This is followed by a numbered list of up to 16 languages (0 to F), with the current operating language inverted. You may enter the number against the language in which you wish all further messages to be written or, if you wish to accept the current language, you may enter (CR). After a short pause, during which the required message file is read from the System Software, the program goes automatically to the Main Menu.

4.3 LEAVING THE PROGRAM

It is only possible to leave the program by pressing RESET. Any changes you have made will not affect the current System, unless a new system has been generated, and the original configuration will be set-up by the IPL.

Operation of CONFIG

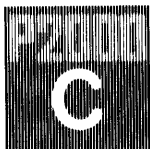
4.4 MAIN MENU

The display for this part of the program consists of a numbered list of 6 choices for further action (0 to 5), under an inverted line giving the name of the section in which you are working and the name, version and generation date of the program.

- 0 = Return To Language Selection
- 1 = System Configuration
- 2 = Disk Table Editing
- 3 = Keyboard Table Editing
- 4 = Video Table Editing
- 5 = Printer Table Editing

Select:

Entering the number of the action you require to carry out will take you to that part of the program. We will first describe the configuration and generation of a new system using already defined tables. After that we will take a closer look at the various tables and the methods of producing and editing them.



Operation of CONFIG

4.5 SYSTEM CONFIGURATION

The display for this part of the program gives the usual inverted line giving the name of the section in which you are working and the name, version and generation date of the program.

You are then informed, by the display, exactly what the current configuration consists of and which additional tables have been set up. At this stage you may select, from the displayed choice, your Disk, Keyboard, Video and Printer set-up. There are also two columns to allow you to select, if required:

- Printer Speed (default value 1200 baud)
- Communication Speed (default value 9600 baud)

The current configuration is indicated by a block inversion on one of the items in each column, with one block brighter than the rest. This indicates the active column. Selection is made by entering the number against your selected table. Selection will cause the next column to become active. Cursor left and right keys allow you to move the active column left or right.

Once the required system has been selected it may be brought into use by a system generation. You can move to the system generation part of the program by pressing (CR). If the table you wish to use requires editing, or if a new table has to be created, you must press (ESC) to return to the MAIN MENU and make the necessary selection.

Operation of CONFIG

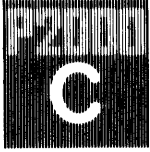
4.6 SYSTEM GENERATION

To generate a new Configuration, once the correct tables have been built up and stored for the disk, keyboard, video and printer, it is only necessary to select which of the available tables you wish to include under each of the four main headings, as explained in the previous section. In addition, you must select appropriate printer and communications speeds from the list given. Entering (CR) after selection will initiate the system generation.

To complete the set-up for the new generation, a few parameters will have to be entered as prompted by the system. These are:

PROMPT	EXPECTED INPUTS
Set CAPSLOCK on startup (y/n) followed by (CR)	Y(es), N(o)
Enter Printer timeout (1-1024 sec) followed by (CR)	Numerical input System will round up input to a multiple of 4
Enter Autostart string or (CR)	List of commands to be executed on IPL (CAPITALS) or (CR)
Enter Welcome message or (CR)	String or (CR). A message entered at this point will be displayed at IPL
Enter floppy drive number on which to write new system (1,2,3,4) followed by (CR)	1 - 4. (No prompt if hard disk is to receive system)
Check that the correct disk is in the selected drive, then enter (Y).	Y

When the final prompt has been entered, the system will generate the new system configuration and return to the MAIN MENU of the Configuration program.



Operation of CONFIG

4.7 DISK TABLE EDITING

The display for this part of the program is headed by the usual inverted line giving the name of the section in which you are working and the name, version and generation date of the program.

The name of the Disk Table in the current system was inverted on the System Configuration display.

The display for Disk Table Editing gives you 5 choices:

- 0 = Return To Main Menu
- 1 = Copy Table To Temporary Buffer
- 2 = Edit Temporary Buffer
- 3 = Save Temporary Buffer To Disk
- 4 = Delete Table From Disk

Select:

There is also a list of the available tables.

Warning Do not create a Disk Table without a valid bootloading device (floppy or hard disk) defined for Drive A. Any attempt to save such a table will be prevented by the system.

We will now discuss the various choices open to you.

Operation of CONFIG

4.7.1 Select 1 - Copy to Temporary Buffer

With this selection it is possible to replace the table in the temporary buffer with any of the listed tables. The program asks for the name of the table to be moved and then returns to the Disk Table Editing Menu. Copying a table means the loss of the existing table in the temporary buffer. If this will be required again it must be saved.

4.7.2 Select 2 - Edit Temporary Buffer

The table in the temporary buffer is not the one inverted on the System Configuration display. It may be the table that is to be edited by the program or the buffer may be empty for table creation. A table in the temporary buffer may not have been saved, in which case it will not be named or listed.

A list is displayed showing the 16 drives names (A to P), and the device that is allocated to each name. An unused name is marked 'not selected'.

A	floppy drive 1, double track drive, 640 kb disk
B	floppy drive 2, double track drive, 640 kb disk
C	hard disk 1, 10 MB drive, 5 MB volume low
D	hard disk 1, 10 MB drive, 5 MB volume high
E	ramdisk 256 kb
F	not selected
G	not selected
H	not selected
I	not selected
J	not selected
K	not selected
L	not selected
M	not selected
N	not selected
O	not selected
P	not selected



Operation of CONFIG

The cursor control (up and down) keys allow you to move the cursor to the name you wish to allocate or alter. After entering the letter N(ot selected), F(loppy), H(ard disk) or R(amdisk) the system will request drive number and capacity information as required. Pressing (ESC) will return you to the Disk Table Editing Menu.

4.7.3 Select 3 - Save Temporary Buffer to Disk

With this selection the current table in the temporary buffer is saved on disk. You will be asked for the name (up to 8 characters) and, if the name already exists, whether the old table should be over-written. After writing the table to disk, the program returns to the Disk Table Editing Menu.

4.7.4 Select 4 - Delete Table From Disk

With this selection the program asks for the name of the table that must be deleted. A (CR), (carriage return), in answer to the request for a name will return to the Disk Table Editing Menu.

4.7.5 Select 0 - Return to Main Menu

With this selection you will be returned to the Main Menu.

Operation of CONFIG

4.8 KEYBOARD TABLE EDITING

The display for this part of the program is headed by the usual inverted line giving the name of the section in which you are working and the name, version and generation date of the program.

The name of the Keyboard Table in the current system was inverted on the System Configuration display.

The display for Keyboard Table Editing gives you 5 choices:

- 0 = Return To Main Menu
- 1 = Copy Table To Temporary Buffer
- 2 = Edit Temporary Buffer
- 3 = Save Temporary Buffer To Disk
- 4 = Delete Table From Disk

Select:

There is also a list of the existing tables.

4.8.1 Keyboard Codes

The 86 keys with NO SHIFT, SHIFT, SUPERSHIFT and SHIFT + SUPERSHIFT give the possibility of more than the 256 possible hexadecimal outputs. For this reason, some codes will be duplicated (i.e., numeric keys) and some will not be changed by action of the shift keys. The Configuration program shows you, and allows you to change, the codes that are generated by each key under each of the four possible shift conditions. The operation is similar to that of the previous part, only the editing - SELECT 2 - being different.



Operation of CONFIG

4.8.2 Select 1, 3 and 4

The selections 1, 3 and 4 are virtually identical to the Disk Table Editing, giving control over a number of variations of your keyboard for different applications. Copying a table means the loss of the existing table in the temporary buffer. If this will be required again it must be saved.

4.8.3 Select 2 - Edit Temporary Buffer

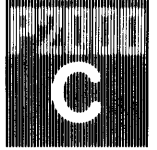
In this part of the program, the TAB key takes you from level to level (shift condition), while the cursor control keys move the cursor around the keyboard diagram. (CR) allows you to access and alter the CAPSLOCK LIMIT. To briefly explain Capslock, the generation of the code 80H causes an inversion of NORMAL/SHIFT key codes BETWEEN the code 41H (A) and a preset limit. For the UK keyboard, this limit will normally be set to 5AH (z), but for other national keyboards it will vary according to the number of additional characters in the alphabet.

By use of this part of the Configuration Program special codes, within the ASCII character set, may be generated by user defined keys. This is done by inserting the ASCII code of the required character in the key position and in the correct shift condition.

4.8.4 Select 0 - Return to Main Menu

With this selection you will be returned to the Main Menu.

Operation of CONFIG

4.9 VIDEO TABLE EDITING

The display for this part of the program is headed by the usual inverted line giving the name of the section in which you are working and the name, version and generation date of the program.

The name of the Video Table in the current system was inverted on the System Configuration display.

The display for Video Table Editing gives you 5 choices:

- 0 = Return To Main Menu
- 1 = Copy Table To Temporary Buffer
- 2 = Edit Temporary Buffer
- 3 = Save Temporary Buffer To Disk
- 4 = Delete Table From Disk

Select:

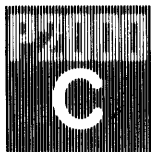
There is also a list of the existing tables.

4.9.1 Video Codes

Here the arrangement of the program is identical, but a short explanation of the use of the table is required. It is quite common, but not necessary, for the character that is represented by the keyboard generated code to be displayed directly on the monitor. By use of Video Tables it is possible to arrange for all or some of the keyboard generated codes to be changed before they are offered to the character generator. A table of all the characters available as screen output from the character generator is provided in Appendix A (Figure A.3 - 8-bit Character Generator Table).

4.9.2 Select 1, 3 and 4

The selections 1, 3 and 4 are virtually identical to the Disk Table Editing, giving control over a number of Video tables for different applications. Copying a table means the loss of the existing table in the internal buffer. If this will be required again it must be saved.



Operation of CONFIG

4.9.3 Select 2 - Edit Temporary Buffer

This selection will show you the current table in the temporary buffer. The display should be interpreted as:

"The console output code equivalent to the Vertical Column + the Horizontal Column (in HEX) will produce the character code at the junction."

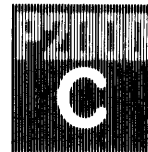
In the following example, all codes between 00 and 1F are displayed as ASCII code 24 (\$). Other characters are not translated.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
10	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
20	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F

4.9.4 Select 0 - Return to Main Menu

With this selection you will be returned to the Main Menu.

Operation of CONFIG

4.10 PRINTER TABLE EDITING

The display for this part of the program is headed by the usual inverted line giving the name of the section in which you are working and the name, version and generation date of the program.

The name of the Printer Table in the current system was inverted on the System Configuration display.

The display for Printer Table Editing gives you 5 choices:

- 0 = Return To Main Menu
- 1 = Copy Table To Temporary Buffer
- 2 = Edit Temporary Buffer
- 3 = Save Temporary Buffer To Disk
- 4 = Delete Table From Disk

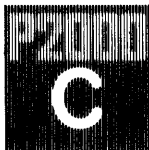
Select:

There is also a list of the existing tables.

4.10.1 Printer Tables

Here the arrangement of the program is identical, but a short explanation of the use of the table is required. As with the video codes, it is quite common, but not necessary, for the character that is represented by the keyboard generated code to be printed directly. By use of the Printer Tables it is possible to arrange for some of the keyboard generated codes to be changed or modified before they are offered to the printer.

In some cases it is necessary to replace one character with 1, 2 or 3 other characters. This is done with 1st, 2nd and 3rd level TRANSPARENT tables respectively.



Operation of CONFIG

In other cases, a character must be replaced by 2, 3 or 4 characters with some correcting character between them - such as a back-space to superimpose two or more normal characters. This is done with 2nd, 3rd and 4th level NONTRANSPARENT tables respectively.

Note: In both cases the 'level' indicates the number of characters that will be printed for each character sent to the printer.

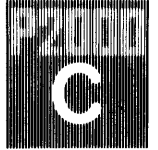
4.10.2 Select 1, 3 and 4

The selections 1, 3 and 4 are virtually identical to the Disk Table Editing, giving control over a number of variations of Printer tables to allow for the use of different printers. Copying a table means the loss of the existing table in the temporary buffer. If this will be required again it must be saved.

4.10.3 Select 0 - Return to Main Menu

With this selection you will be returned to the Main Menu.

Operation of CONFIG

4.10.4 Select 2 - Edit Temporary Buffer

This selection will show you the current table in the temporary buffer. A typical printer table is shown in figure 1.1.

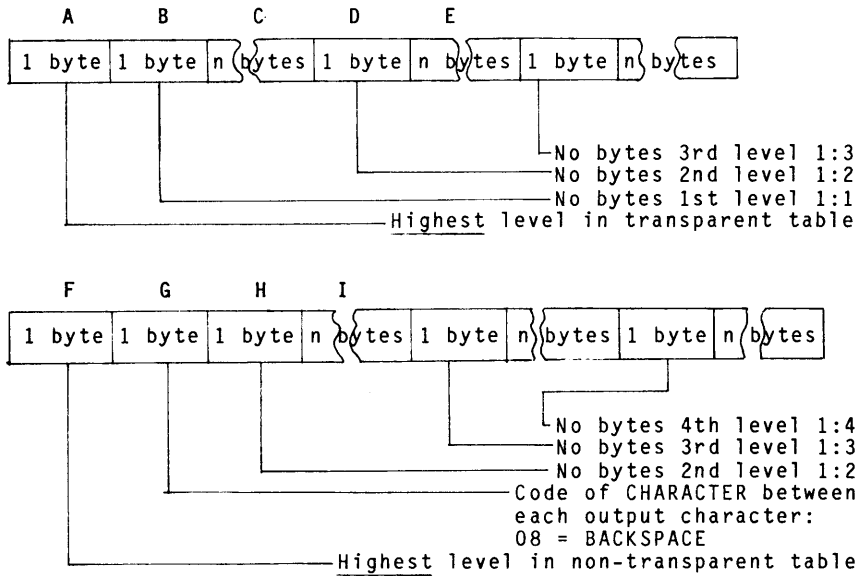
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	02	16	5C	7C	60	3C	A4	4F	A6	3E	B0	60	B7	6F	C2	7E
10	DE	7B	EE	41	FA	7C	FE	61	03	A3	1B	20	02	08	4E	3C
20	2D	7E	3E	2D	3C	7B	28	2D	7D	29	2D	7E	3E	7E	C5	41
30	3E	C6	45	7E	CE	43	2C	CF	4E	7E	D3	61	3C	D5	61	3E
40	D6	65	7E	D7	65	3C	DC	69	3C	DF	6E	7E	E2	41	60	E5
50	4F	3E	E9	4F	25	ED	55	3E	F2	61	60	F5	6F	3E	F7	6F
60	3C	F9	6F	2F	FB	7C	33	FC	75	3C	FD	75	3E	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Figure 1.1 Configuration Program - Typical Printer Table

This printer table can be considered as two continuous strings, with a total maximum length of 256 bytes, which describe any translations between sent and printed characters. The format of this string is shown in figure 1.3, the first string defining the transparent tables, the second defining the non-transparent tables. The table shown in figure 1.1 is repeated in figure 1.2, with the various areas indicated. To clarify the use of the printer table, the translations made by the table in the above example are listed in figure 1.3.



Operation of CONFIG



	A	B	C	D	E	F	G	H	I							
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	02	16	5C	7C	60	3C	A4	4F	A6	3E	B0	60	B7	6F	C2	7E
10	DE	7B	EE	41	FA	7C	FE	61	03	A3	1B	20	02	08	4E	3C
20	2D	7E	3E	2D	3C	7B	28	2D	7D	29	2D	7E	3E	7E	C5	41
30	3E	06	45	7E	CE	43	2C	CF	4E	7E	D3	61	3C	D5	61	3E
40	D6	65	7E	D7	65	3C	DC	69	3C	DF	6E	7E	E2	41	60	E5
50	4F	3E	E9	4F	25	ED	55	3E	F2	61	60	F5	6F	3E	F7	6F
60	3C	F9	6F	2F	FB	7C	33	FC	75	3C	FD	75	3E	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Figure 1.2 The Printer Table in Detail

Operation of CONFIG

Transparent Conversions
LEVEL 1

INPUT	PRINTER
5C	7C
60	3C
A4	4F
A6	3E
B0	60
B7	6F
C2	7E
DE	7B
EE	41
FA	7C
FE	61

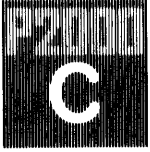
LEVEL 2

INPUT	PRINTER
A3	1B + 20

Non-Transparent Conversions
LEVEL 2

INPUT	PRINTER
3C	2D + 08 + 7E
3E	2D + 08 + 3C
7B	28 + 08 + 2D
7D	29 + 08 + 2D
7E	3E + 08 + 7E
C5	41 + 08 + 3E
C6	45 + 08 + 7E
CE	43 + 08 + 2C
CF	4E + 08 + 7E
D3	61 + 08 + 3C
D5	61 + 08 + 3E
D6	65 + 08 + 7E
D7	65 + 08 + 3C
DC	69 + 08 + 3C
DF	6E + 08 + 7E
E2	41 + 08 + 60
E5	4F + 08 + 3E
E9	4F + 08 + 25
ED	55 + 08 + 3E
F2	61 + 08 + 60
F5	6F + 08 + 3E
F7	6F + 08 + 3C
F9	6F + 08 + 2F
FB	7C + 08 + 33
FC	75 + 08 + 3C
FD	75 + 08 + 3E

Figure 1.3 Printer Table Conversions



Operation of CONFIG

4.11 NOTES ON THE PREPARATION OF NATIONAL VERSIONS

The following notes describe how the configuration program handles the restrictions imposed by the fact that CP/M is designed for the uninhibited I/O of 7-bit character codes. It is particularly important to take note of this if you intend to develop different national versions of an application, since many of the characters necessary for different languages are defined by the P2000C 8-bit character table (see App A, figs 2 & 3). Briefly, what the program does is to look at the national video table for any codes that are translated. All codes translated there are translated accordingly in the printer table. This means that it is not necessary to have a separate printer table for each national language character set.

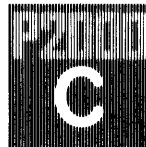
4.11.1 Example: '#' Translated to '£'.

Printer tables are supplied for many types of printer. Each of them contains in addition to translations in the normal ASCII range (20-7F), all of the translations likely to be needed by nationally dependent characters (A0-FF), as defined in the P2000C 8-bit character table (App A, fig 3). The diagram on page 4.15 shows how the tables have been set up.

At the last stage of working with the configuration program, when you instruct the program to generate the system, as you have configured it, CONFIG will do the following:

- 1 For each printer table input code the program finds the first occurrence of the same code in the video table. If the code in this position matches the video table position; that is, if there is no translation, nothing happens and the next printer table input code is read. In the UK video table, for example, in position 23 the code A3 (£) appears, and in position A3, to ensure no duplication, the code 23 (#) is found.

Operation of CONFIG



2 In such a case, where the video table code does not match with its position, the program patches the printer table input code according to the value found in the video table at that position. Thus A3 becomes 23 in the printer table also and falls within CP/M's range of characters.

3 It must be emphasised that if you modify the video table in this way and replace a character in the lower part with one from the upper part, you must also insert the character that you are replacing into the upper part, so that there are no duplicate codes.

4 The printer tables themselves, existing in the file CONFIG.DAT, are not patched. But when the CP/M system tracks are written to the disk, the table to be copied, along with the other configuration data, is modified.



The UTIL Program

5 THE UTIL PROGRAM

The UTIL program on the P2000C administers the following functions:

- Formatting P2000C floppy disks (160k and 640k)
- Formatting hard disk
- Copying floppy disk contents to P2000C floppy disks

The 'copying' option is available to copy from other P2000C disks or from earlier Philips disk formats (for p-System as well as for CP/M).

The program offers the choice of which language the messages appear in and whether the messages in other languages should be discarded (to save on disk space).

5.1 STARTING THE UTIL PROGRAM

To call the UTIL program type the name against the CP/M prompt. At startup the program will be in the 'Select Language and Disk Arrangement' mode, and will display the following message at the top of the screen:

SELECT LANGUAGE AND DISK ARRANGEMENT

During this operation, the disk from which the UTIL program was executed must remain in its drive until the MAIN MENU is displayed.

Press any key when ready

If your version of the program includes language selection, you will be prompted at this point to specify the language in which messages are to appear, and also whether other language messages are to be retained.



The UTIL Program

Once this part of the program has been completed, in future, on startup you will be taken directly into the UTIL main menu.

The next screen will have the same header, and will display the following prompt lines:

Define your FLOPPY DISKETTE arrangement by typing the appropriate number against each drive:

- 1 - 160k
- 2 - 640k
- 3 - Not existing

DRIVE - 1:
DRIVE - 2:
DRIVE - 3:
DRIVE - 4:

5.2 THE MAIN MENU

Once you have completed the diskette arrangement section, the program will display the UTIL main menu at the top of the screen.

P2000C UTIL (Rel: 1.0X) - MAIN MENU

SELECT:
1 - FORMAT FLOPPY DISK
2 - FORMAT HARD DISK
3 - COPY FLOPPY DISK
4 - REDEFINE LANGUAGE/FLOPPY DISK ARRANGEMENT

TO ESCAPE PRESS 'ESC' OR '0'



The UTIL Program

5.2.1 Select 1 - Format Floppy Disk

Option 1 allows the formatting of disks according to the arrangement setup during the initial stage of the UTIL program. The main menu is replaced by this screen:

FORMAT FLOPPY DISK TO P2000C FORMAT

Enter (sequentially) the DRIVE NUMBER(S) containing the disks to be formatted, terminate with (CR)
Exit with (ESC) or (0)
Formatting continues cyclically until an empty drive is found

Drive(s) to format (1 to 4):

A warning message will be displayed before the formatting process continues any further.

When formatting begins, the program will indicate which disk and which drive is currently being processed. If an error occurs at this point, (for example, an open drive or a faulty disk), an error message will be displayed at the bottom of the screen. If this happens, pressing any key will return you to the main menu.

5.2.2 Select 2 - Format Hard Disk

This option will format every track of a 10M byte hard disk connected via the SASI port. The screen that replaces the main menu, if option 2 is selected, is shown on the following page.

The UTIL Program



FORMAT HARD DISK

Which DRIVE (1or 2)?

When input is received at this point, before the program will start the format process, a warning message will be displayed, to give you the chance to check that everything is in order.

On input, UTIL will begin formatting and indicate on the screen which physical hard disk and which track is currently being verified. If an error occurs you must press any key to return to the main menu.

5.2.3 Select 3 - Copy Floppy Disk

The function of this part of the program is to copy data from other Philips diskettes to P2000C format diskettes. The size of the destination disk (i.e. 160k or 640k), that was defined at the initial stage of the UTIL program, will determine the allowable source disk selections. The following screen will be displayed, each line appearing as input is received for the previous line:

COPY FLOPPY DISK TO P2000C FORMAT

Enter SOURCE drive (1 to 4) :
Enter destination drive (1 to 4) :
do you wish to verify whilst copying (y/n) ?



The UTIL Program

Having entered information in response to these prompts, you will be presented with either one of the two displays shown below.

For 160k drive as destination:

Select SOURCE disk type:
1 - P2000C 160k
2 - P2000M

to escape enter (ESC) or (0)

For 640k drive as destination:

Select SOURCE disk type :
1 - P2000C 640k
2 - P2000C 160k
3 - P2000M
4 - P2500 300K
5 - P2500 600K

to escape press 'ESC' or '0'

The UTIL Program



If your source is not a P2000C 640K disk, an additional prompt will appear, to which you must respond by specifying which system was used to store files on the disk: CP/M or BASIC, or CP/M or p-System.

On both the 160k and 640k screens, once a selection has been made, the following will be displayed along with the escape message:

WARNING : Destination disk will be DELETED!
Press any key when ready

The program will then indicate which disk is currently being accessed and, in the case of physical copying, indicate also which track is being addressed. If logical copying is taking place, the program will indicate which file is currently being copied.

5.2.4 Select 4 - Redefine Language/Floppy Disk Arrangement

If you select option four from the main menu, you will be taken back to the initial part of the program, (described in section 5.1)



Backup for Hard Disks

6 BACKUP FOR HARD DISKS6.1. Using BACKUP.COM

The purpose of this utility is to copy hard disk volumes onto 5.25" floppy disks, or to restore volumes from these backup disks.

In order to provide an additional degree of data security, the floppy disks used to backup a volume are serialised during the process and cannot then be overwritten immediately. The next backup must be made onto another set of disks. Furthermore, the backup program 'links' the disks to either the upper or the lower volume. For example: the contents of a lower volume, stored on a particular set of disks, cannot then be restored to an upper volume, and, similarly, an upper volume cannot be copied to disks that have been used to store the contents of a lower volume. The number of occupied blocks on the hard disk volume is also checked by the program to ensure that the volume is the correct size.

To call the program under CP/M type:

A> BACKUP

The first screen displayed by the program will look like the one shown at the top of the following page. Each of the lines to the right of the screen are displayed individually; as you type in a valid response to a prompt, (example responses are given), the next line appears.

Backup for Hard Disks

HARD DISK BACKUP PROGRAM

```
Enter todays date : 29/11/83
Enter hard disk volume (A,B...P) : A
Enter 1st backup volume (A,B...P) : C
Enter 2nd backup volume ( or (CR) ) : D
```

- 1 - Backup
- 2 - Restore

Select : 1

Enter 1, 2 or (ESC) to indicate whether you wish to backup, restore or exit. (It is not necessary to press return after any entry in connection with this program). If you select 1 or 2, an additional prompt will be displayed beneath the others. Depress any key at this point and the backup process will start.

The following information applies equally to the process for restoring data from backup since both procedures use virtually the same prompts.

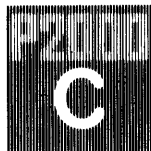
HARD DISK BACKUP PROGRAM

```
Enter todays date : 29/11/83
Enter hard disk volume (A,B...P) : A
Enter 1st backup volume (A,B...P) : C
Enter 2nd backup volume ( or (CR) ) : D
```

- 1 - Backup
- 2 - Restore

Select : 1

Insert disk 1 in Drive C
Type any key to continue ((ESC) to abort) :



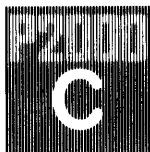
Backup for Hard Disks

A message indicating which drive is currently receiving or sending data to the hard disk will be displayed. This will change as the program fills up the disk in one drive and starts transferring data to or from the next disk. You will be prompted when it is necessary to insert a new disk. The process continues until a complete volume has been transferred.

There are a number of help or warning messages that you may encounter during the operation of the BACKUP.COM program. Some of them are described below:

<u>Message</u>	<u>Indicates:</u>
Latest backup, use another	That you have a disk from the most recent backup series for that volume in the drive.
Bad serial-number of the disk	That there is a disk that has not been serialised for use with that volume.
Restore disk out of sequence	That the disk in the drive does not have the expected number for correct restoration of data.
Restore volume is wrong size	That the floppy disk is serialised for use with a bigger or smaller hard disk volume.

When the process has finished, the 'insert disk' message will be replaced with a 'process complete' message, and you will be able to restart the process directly, should you wish to copy or restore another hard disk volume.



CP/M 'ASM' Command

7 CP/M 'ASM' COMMAND

The CP/M assembler is described in full in the Digital Research documentation in part B of this manual. This chapter provides an introduction to the use of the ASM command and its requirements.

The ASM command loads and executes the CP/M 8080 assembler. The command must be followed by a source filename containing assembly language statements, in which the extension to the filename is assumed to be ASM, and so need not be specified. The two-pass assembler is automatically executed and any errors occurring are printed at the console.

The assembler produces a file with the extension PRN which contains a listing of the source program along with the machine code generated for each statement and diagnostic error messages, if any. The PRN file can be listed at the console using the TYPE command or sent to a peripheral device using PIP. It can be edited by removing the leftmost 16 characters of each line, (this can be done by issuing a single editor 'macro' command). The resulting file is identical to the original source file and can be renamed from PRN to ASM for subsequent editing and assembly.

A file with the extension HEX is also produced. This contains 8080 machine code in hexadecimal format suitable for subsequent loading and execution (see next chapter).



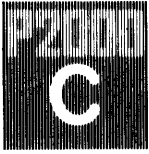
CP/M 'LOAD' Command

8 CP/M 'LOAD' COMMAND

The LOAD command reads the file specified in the command line, which is assumed to contain machine code in hexadecimal format, and produces a memory image file which can be executed. The filename is assumed to have the extension HEX, and so only the primary filename need be given.

The command creates a file with the extension COM, which marks it as containing machine executable code. The hex file need only be loaded once; it can be subsequently executed any number of times by simply typing the primary name.

The hex file must contain valid Intel format hexadecimal machine code records (as produced by the ASM program, for example), which begin at 100H, the beginning of the TPA. Further, the addresses in the hex records must be in ascending order; gaps in unfilled memory regions are filled with zeros by the LOAD command as the hex records are read. Thus LOAD must be used only for creating standard CP/M 'COM' files which operate in the TPA. Programs which occupy regions of memory other than the TPA can be loaded using the DDT utility.



Disk Information

9- DISK INFORMATION9.1 FIVE-INCH DISKS

Five-inch disks, as supported by the P2000C, may be either single sided, single density (48 tpi) - 160K - or double sided, double density (96 tpi) - 640K.

9.1.1 Single Sided

The characteristics of the single sided disk is as follows, (as displayed by the CP/M STAT command):

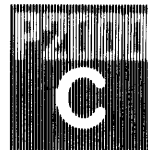
```
A: Drive Characteristics
1216: 128 Byte Record Capacity
 152: Kilobyte Drive Capacity
   64: 32 Byte Directory Entries
   64: Checked Directory Entries
  128: Records/ Extent
    8: Records/ Block
   32: Sectors/ Track
    2: Reserved Tracks
```

9.1.2 Double Sided

The characteristics of the double sided disk are as shown below:

```
A: Drive Characteristics
5056: 128 Byte Record Capacity
 632: Kilobyte Drive Capacity
  128: 32 Byte Directory Entries
  128: Checked Directory Entries
 512: Records/ Extent
   32: Records/ Block
   32: Sectors/ Track
    2: Reserved Tracks
```

Disk Information

9.2 HARD DISK

The characteristics of a 10 megabyte hard disk are as follows:

When the disk is partitioned into 2 volumes, each of 5MB:

5MB lower

A: Drive Characteristics
39104: 128 Byte Record Capacity
4888: Kilobyte Drive Capacity
256: 32 Byte Directory Entries
0: Checked Directory Entries
256: Records/ Extent
32: Records/ Block
32: Sectors/ Track
2: Reserved Tracks

5mb upper

B: Drive Characteristics
39168: 128 Byte Record Capacity
4896: Kilobyte Drive Capacity
256: 32 Byte Directory Entries
0: Checked Directory Entries
256: Records/ Extent
32: Records/ Block
32: Sectors/ Track
1224: Reserved Tracks



Disk Information

When the disk is partitioned into 2 volumes, the lower of 2 MB and the upper of 8MB:

2MB lower

A: Drive Characteristics
15616: 128 Byte Record Capacity
1952: Kilobyte Drive Capacity
256: 32 Byte Directory Entries
0: Checked Directory Entries
256: Records/ Extent
32: Records/ Block
32: Sectors/ Track
2: Reserved Tracks

8MB upper

B: Drive Characteristics
62656: 128 Byte Record Capacity
7832: Kilobyte Drive Capacity
256: 32 Byte Directory Entries
0: Checked Directory Entries
256: Records/ Extent
32: Records/ Block
32: Sectors/ Track
490: Reserved Tracks



Access to the Video Terminal

10 ACCESS TO THE VIDEO TERMINAL10.1 SCREEN SPECIFICATIONS

8-bit code (national versions) 24 lines/80 characters

Character Mode

- Bidirectional scrolling
- Split-screen capability (partial scroll)
- Three Attribute modes:
 - Manual Read and write data from/to attribute page (normal memory access).
 - Auto duplicate: The read attribute data will be duplicated.
 - Block mode: For "block moves" such as scrolling, the attribute page will be scrolled automatically.
- Attributes: Underline)
 Invert) and all
 Blink) combinations
 4 Intensity levels)
- Adjustable TABs
- Teletext graphics
- Text and Attributes back-transfer from screen possible.

High Resolution Graphics Mode

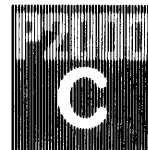
- 2 selectable modes
 - 512 x 252 dots (no attributes)
 - 256 x 252 dots (3 intensity levels + background)

Each dot addressable

Simple vector handling:

- In Cartesian and Polar co-ordinates.
- Combination with character mode (characters: 21 lines/64 characters).

Access to the Video Terminal

10.2 SCREEN AND SCREEN CODES

The screen is controlled by both single codes and ESCape sequences.

10.2.1 Single Codes

A C T I O N	Code	

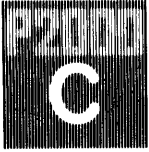
Cursor Home	SOH	01H
Cursor Forward	ACK	06H
Cursor Down	LF	0AH
Cursor Up	SUB	1AH
Cursor Back	NAK	15H
Bell-Beep	BEL	07H
Backspace (same as Cursor Back)	BS	08H
TAB	TAB	09H
Clear Screen	FF	0CH
CR	CR	0DH
End of Page (column 80, row 24)	EOT	04H
Reset Terminal	CAN	18H
CAPS LOCK	SI	0FH
Lock Keyboard	EM	19H
Unlock Keyboard	STX	02H

10.2.2 Escape Sequences - Set Attribute

```
SET ATTRIBUTE = ESC,0,b
               where b=attribute byte
                   and 0=numeric zero
```

An attribute can be set at any time and is valid until a new attribute is selected.

```
Attributes: Underline - UL
             Blink     - BL
             Invert    - INV
             4 Intensities
             res       - reserved
```



Access to the Video Terminal

```
Attribute byte: |res|in1| UL|INV|res|res| BL|in2|
                |---|---|---|---|---|---|---|---|
bit:            7   6   5   4   3   2   1   0
```

Intensities	in1	in2
Quarter bright	0	0
Bold	0	1
Normal	1	0
Half bright	1	1

For example, the ESCape sequence ESC,0,99 (or ESC,0,&H63) would cause screen characters to be produced:

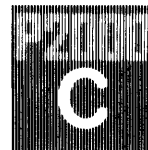
- half brightness
- underlined
- blinking

NOTE: CP/M and BASIC will not send directly all possible escape sequences to the terminal, since some of these contain values outside the normal range for printing. For example, the set graphic dot command 'ESC, D,xy' requires that xy is 3 bytes in graphic mode 2 or 2 bytes in graphic mode 1. However, these bytes may have values from 0 - 255, which includes many outside the ASCII range.

The CP/M list output routine cannot output the values 254 and 255. BASIC will intercept and alter some values passed in a print statement when the values are in the range of control characters (0-31). When used with the default width of 80 characters, BASIC will insert a carriage return and line feed after every 80 characters. If this occurs while sending an escape sequence the sequence will be corrupted. This problem can be avoided by using a 'width 255' statement. Similar situations may occur with other high level languages.

To avoid the CP/M list device call and these difficulties with some high level languages, it is possible to call one of the low-level routines in the machine ROM BIOS, (described in the System Reference Manual, chapter 2. The most useful of these are: Console input, Console output and Console status).

Access to the Video Terminal

10.2.3 Escape Sequences - Screen Control

A C T I O N	Code	ASCII
Cursor Addressing	ESC,Y,r,c r=row max = 24 c=column max = 80	Y=59H/89
Erase to End of Line	ESC,K	K=4BH/75
Erase to End of Screen	ESC,k	k=6BH/107
Scroll Up one Line	ESC,S	S=53H/83
Scroll Down one Line	ESC,T	T=54H/84
Set TAB at Cursor Position	ESC,I	I=49H/73
Clear TAB at Cursor Position	ESC,G	G=47H/71
Clear all TABs	ESC,g	g=67H/103
Insert Line	ESC,L	L=4CH/76
Delete Line	ESC,l	l=6CH/108
Insert Character at Cursor Position ON	ESC,Q	Q=51H/81
OFF	ESC,R	R=52H/82
Delete Character at Cursor Position	ESC,P	P=50H/80
Insert Character Wrap-around ON	ESC,N	N=4EH/78
OFF	ESC,R	R=52H/82
Delete Character Wrap-around	ESC,O	O=4FH/79
Back TAB	ESC,i	i=69H/105
Cursor Visible	ESC,C	C=43H/67
Cursor Invisible	ESC,c	
Start Teletext Graphic	ESC,1	1=31H/49
End Teletext Graphic	ESC,2	2=32H/50
Lock Area for Scrolling	ESC,A,n	A=41H/65
	n=number of lines	
Unlock Area from cursor	ESC,a	a=61H/97
Unlock all Areas	ESC,u	u=75H/117
Send Status	ESC,?	?=3FH/63

NOTE: Escape sequences that cause data to be sent back to the processor, eg, Receive picture, Read status, attributes, text - may get data that includes 90H (the print screen function). If so, it will be interpreted as such. Thus it may be necessary to disable this function by changing byte 8 in CP/M memory (see page 11.1).



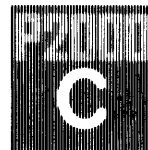
Access to the Video Terminal

Send Text from Cursor Position	ESC,\$,nn	\$=24H/36 nn=number of characters
Send Attributes of Text from Cursor	ESC,%,nn	%=25H/37
Load User Program	ESC,p	p=70H/112 (in INTEL HEX format)
End of INTEL HEX Format (exits loader, normal operation)	ESC,::	:=3AH/58
Execute User Program	ESC,x	x=78H/120
Load New Keyboard Table	ESC,@	@=40H/64
Load New Screen Table	ESC,!:	!=21H/33
Define Caps Lock Key (k=key,nn=upper limit (Hex))	ESC+,k,nn	+ =2BH/43

nn means two bytes in order of low byte, high byte.

10.2.4 Description of Control Codes

Cursor home:	New cursor position is column 1/ row 1
Cursor forward:	Column + 1.
Cursor down:	New line. Scroll if last line or beginning of a locked area.
Cursor up:	One line up. If 1st line then new position is the bottom line.
Cursor back:	Column - 1.
Backspace:	Same as cursor back.
TAB:	Cursor to next TAB position, default every eighth column.
Clear screen:	Erase the whole screen, cursor at home position.
CR:	Cursor at column 1 in current line.
End of page:	New cursor position is column 80 and row 24
Reset terminal:	Initialise hardware and software. After RESET allow 500ms before sending data to terminal.
Lock keyboard:	All keyboard inputs are ignored.
Unlock keyboard:	Keyboard entries are re-enabled.
Reset:	Re-initializes the hardware and software.
Set attribute:	A new attribute is used until the next "Set attribute" command.



Access to the Video Terminal

Cursor address:	Absolute cursor address with an offset of 20H (ESC,Y,20H,20H is the "home position").
Erase to end of line:	Clear all characters including cursor position to column 80.
Erase to end of screen:	Same as "Erase to end of line" to column 80, row 24.
Scroll up 1 line:	Scroll up the whole screen or area and clear last line.
Scroll down 1 line	Same as above, but scroll down.
Set TAB:	Sets a new TAB position at cursor position.
Clear TAB:	A TAB position is removed at cursor position.
Clear all TABs:	Removes all TABs.
Insert line:	Scroll down the lines from cursor line + 1, and clear cursor line.
Delete line:	Scroll up the lines from cursor line +1 to cursor line.
Insert ON:	The next character will be inserted at cursor position, the last character in the line will be lost.
Insert wrap-around:	Same as "Insert" but the last character of the screen is lost
Insert OFF:	Exit insert mode, normal overwrite.
Delete character:	Delete character at cursor position, the last character in the line will be blank.
Delete character wraparound:	Same as "Delete character" but last character of screen will be blank.



Access to the Video Terminal

Back TAB: New cursor position is previous TAB position.
 Cursor visible: Display cursor.
 Cursor invisible: Do not display cursor.
 Start Teletext graphic: All characters between 20H & 3FH, and 60H & 7FH are interpreted as teletext characters.
 End Teletext graphic: Normal character mode.
 Lock area: From cursor line, n lines will be locked. (This area will not be scrolled by Cursor up and down, only by explicit "Scroll up (down)" command.
 Unlock area: The cursor position area is unlocked.
 Unlock all: Normal screen status.
 Send status: 12 bytes terminal status information will be sent (see STATUS INFORMATION).
 Send text: nn characters from cursor position will be sent back.
 Send attribute: nn attribute bytes from cursor position will be sent.
 Load user PGM: Starts INTEL HEX FORMAT loader for down-loading a machine code program in INTEL HEX format.
 End loader: Exits the loader, enables normal operation.
 Execute PGM: Calls a previously loaded program.
 Load new keyboard table: A new keyboard table (national version) will be downloaded. It consists of 4 sub-tables (NORMAL, SHIFT, SUPER SHIFT, SUPER SHIFT-SHIFT) (see STANDARD KEY TABLE).
 Load new screen table: A new screen translation table will be loaded.
 Define Caps Lock Key: The key K is the new caps lock key. There is no default key.

Access to the Video Terminal

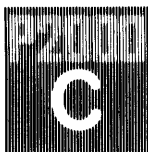
10.2.5 Graphic Control

ACTION	Code	ASCII
Start high resolution mode 1 (256 x 252)	ESC,5	5=35H/53
Start high resolution mode 2 (512 x 252)	ESC,3	3=33H/51
Start character mode (end graphics)	ESC,4	4=34H/52
CARTESIAN CO-ORDINATES:		
Clear dot	ESC,d,xy	d=64H/100
Set dot	ESC,D,xy	D=44H/68
Move to	ESC,m,xy	m=6DH/109
Draw to	ESC,M,xy	M=4DH/77
Clear to	ESC,v,xy	v=76H/118
POLAR CO-ORDINATES:		
Set origin	ESC,z,xy	z=7AH/122
Move to	ESC,y,Aabs	y=79H/121
Draw to	ESC,U,Aabs	U=55H/85
Clear to	ESC,w,Aabs	w=77H/119
Set dot	ESC,F,Aabs	F=46H/70
Clear dot	ESC,f,Aabs	f=66H/102
Send picture to terminal	ESC,r,xy,nn	r=72H/114 nn=number of bytes
Receive picture from terminal	ESC,t,xy,nn	t=74H/116
Note: xy	= co-ordinates:	2 bytes in mode 1 3 bytes in mode 2 (low 'x' byte first, i.e., x(low), x(high), y
	Aabs = A = angle ALPHA	(2 bytes)
	abs = absolute value	(2 bytes)

For sending and receiving picture function, a byte represents the contents of the video RAM (like a dump).

Co-ordinates: x is the horizontal co-ordinate, y the vertical co-ordinate x=0, y=0 is bottom leftmost dot on the screen.

Angle ALPHA in steps of one degree (0 to 360)



Access to the Video Terminal

10.2.6 Description of Graphic Commands

Start mode 1: Start 256 x 252 resolution mode with 3 intensity levels. The change of this level is done by "Set attribute" command.

Start mode 2: Start the 512 x 252 resolution mode.

Start character mode: Exit high resolution graphic mode 1 or 2.

After a start graphic mode command the text on the screen is not cleared but will be re-arranged as 21 lines of 64 characters. The graphic screen will be cleared and the internal cursor and origin set to zero. The start character mode command (exit graphics mode) will clear the text buffer and graphics screen and set the internal cursor and origin field to zero.

CARTESIAN COORDINATES

Clear dot: Erase pixel at screen position xy.

Set dot: Set a pixel at xy.

Move to: Set the internal cursor field to xy.

Draw to: Draw a line from internal cursor field to xy, and set the internal cursor field to xy.

Clear to: Same as "Draw to" but the line is erased.

POLAR COORDINATES

Set origin: Set the internal origin field to xy.

Clear dot: Erase a pixel at A (angle), abs (absolute value) according to the origin.

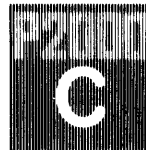
Set dot: Same, but set the pixel.

Move to: Set the internal cursor field to the calculated (using A abs, origin) value xy.

Draw to: Draw a line to the calculated value xy and set internal cursor field to xy.

Note: Any combination of POLAR and CARTESIAN coordinates is possible!

Access to the Video Terminal

10.3 STATUS INFORMATION

When the 'send status' code sequence ESC,? (1BH,3FH) is sent to the terminal, the terminal status information is returned in the form of a 12 byte string, as shown below:

Byte Number	Contents
1	cursor position - column
2	cursor position - row
3	character at cursor position
4	status flag
	Bit:
0	1 = graphics on
1	1 = graphic mode 2
2	1 = teletext on
3	1 = insert mode on
4	1 = insert wraparound on
5	1 = keyboard is locked
6	reserved
7	reserved
	internal graphic cursor field
5,6	x co-ordinate
7	y co-ordinate
8,9	free space pointer (beginning of RAM area for user program)
10,11,12	reserved for future use

The twelve bytes must be read via the normal 'keyboard read' instruction.



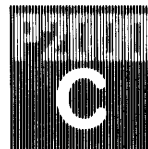
Reserved Locations in Page Zero

1

11 RESERVED LOCATIONS IN PAGE ZERO

Main memory page zero, between locations 00H and 0FFH, contains several segments of code and data that are used during CP/M processing. The code and data areas are given below for reference.

Locations from to	Contents
0000H 0002H	Contains a jump instruction to the warm start entry point at location 4A03H + b. This allows a simple programmed restart (JMP 0000H) or manual restart from the front panel.
0003H 0003H	Contains the Intel standard IOBYTE, which is optionally included in the user's CBIOS.
0004H 0004H	Current default drive number (0=A,...15=P)
0005H 0007H	Contains a jump instruction to the BDOS and serves two purposes: JMP 0005H provides the primary entry point to the BDOS and LHL 0006H brings the address field of the instruction to the HL register pair. This value is the lowest address in memory used by CP/M (assuming that the CCP is being overlaid). The DDT program will change the address field to reflect the reduced memory size in debug mode.
0008H 0027H	08H is used to enable or disable the print screen function. Zero means enabled, not = zero means disabled. (09H-27H, not currently used).
0030H 0037H	(Interrupt location 6, not currently used; reserved).

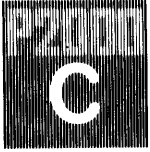


Reserved Locations in Page Zero

0038H 003AH	Restart 7; contains a jump instruction into the DDT or SID program when running in debug mode for programmed breakpoints, but is not otherwise used by CP/M.
003BH 003AH	(Not currently used; reserved).
0040H 004FH	A 16-byte area reserved for scratch by CBIOS; but is not used for any purpose in the distribution version of CP/M.
0050H 005BH	(Not currently used; reserved).
005CH 007CH	Default file control block produced for a transient program by the Console Command Processor.
007DH 007FH	Optional default random record position.
0080H 00FFH	Default 128-byte disk buffer (also filled with the command line when a transient is loaded under the CCP).

This information is set up for normal operation under the CP/M, but can be overwritten by a transient program if the BDOS facilities are not required by the transient.

If, for example, a particular program performs only simple I/O and must begin execution at location 0, it can first be loaded into the TPA, using normal CP/M facilities, with a small memory move program that gets control when loaded (the memory move program must get control from location 0100H, which is the assumed beginning of all transient programs). The move program can then proceed to move the entire memory image down to location 0 and pass control to the starting address of the memory load. If the BIOS is overwritten or if location 0 (containing the warm start entry point) is overwritten, the operator must bring the CP/M system back into memory with a cold start sequence.

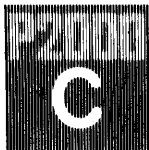


System Memory Locations

12 SYSTEM MEMORY LOCATIONS

Beginning of BDOS
For 61 K Conf. ..E006
For 62 K Conf. ..E406
For 63 K Conf. ..E806

Beginning of CBIOS
For 61 K Conf. ..EE00
For 62 K Conf. ..F200
For 63 K Conf. ..F600



Manual Comment Form

Manual Title: P2519 P2000C CP/M REFERENCE MANUAL

5103 993 11921

Originator:

Name

Address

Comment (if possible, add a copy of the page(s) affected by the comment, marked with the proposed changes)

Please return this form to: Philips Austria
Personal Computer Division EFW-M
Breitenseerstr. 116
A-1141 Vienna
A U S T R I A



Manual Status Control Sheet

Manual Name: P2519 P2000C CP/M Reference Manual

5103 993 11921



Character Code Tables

A CHARACTER CODE TABLES

CP/M will only support 7 bit codes from the standard ASCII code table. Of these 128 codes, fourteen (those left blank in Figure A.1 below), have characters assigned to them according to the requirements of a particular national version. The national variable characters assigned to these ISO character codes are set out in Figure A.2 on the following page. All the available characters mapped in the machine's character generator are shown in Figure A.3.

	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0		P		p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3		3	C	S	c	s
4	EOT	DC4		4	D	T	d	t
5	ENG	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB		7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K		k	
C	FF	FS	,	<	L		l	
D	CR	GS	-	=	M		m	
E	SO	RS	.	>	N		n	
F	SI	US	/	?	O		o	↓

Figure A.1 7-bit ASCII / ISO Characters

Character Code Tables

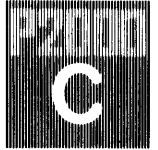


Figure A.2 National Code Table for ISO Characters

HEX:	23	24	27	40	5B	5C	5D	5E	5F	60	7B	7C	7D	7E
ASCII/INT.	#	\$	'	@	[\]	^	_	`	{		}	~
UK (special)	£	\$	'	@	#	\		^	_	`	{		}	~
D/A	#	\$	'	§	Ä	Ö	Ü	^	_	`	ä	ö	ü	ß
F/B	£	\$	'	à	°	ç	§	^	_	`	é	ù	è	¨
I	£	\$	'	§	°	ç	é	^	_	ù	à	ò	è	ì
E	#	\$	'	@	[Ñ]	^	_	`	{	ñ	}	¨
S/SF	#	¤	'	É	Ä	Ö	Å	^	_	é	ä	ö	å	ü
DK/N	#	\$	'	@	Æ	Ø	Å	^	_	`	æ	ø	å	~
P	£	\$	'	@	Ã	Ç	Õ	^	_	`	ã	ç	õ	~
CH	£	\$	'	ç	à	é	è	^	_	`	ä	ö	ü	¨
UK/NL	£	\$	'	@	[\]	^	_	`	{		}	~

Key

ASCII/INT
 UK (special)
 D/A
 F/B
 I
 E
 S/SF
 DK/N
 P
 CH
 UK/NL

German / Austrian
 French / Belgian
 Italian
 Spanish
 Swedish / Finnish
 Danish / Norwegian
 Portuguese
 Swiss
 UK / Dutch



Character Code Tables

Figure A.3 8-bit Character Generator Table

	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	SP	0	@	P	\	p	□	□	◊	◊	◊	◊	◊	◊
1	i	1	A	Q	a	q	□	□	!	±	À	á	Æ	æ
2	"	2	B	R	b	r	□	□	°	z	/	!	À	á
3	#	3	C	S	c	s	□	□	£	3	À	á	ä	ü
4	\$	4	D	T	d	t	□	□	Ö	x	À	á	ö	ö
5	%	5	E	U	e	u	□	□	Y	µ	À	á	ö	ö
6	&	6	F	V	f	v	□	□	..	¶	È	é	ö	ö
7	'	7	G	W	g	w	□	□	§	ö	È	é	ö	ö
8	(8	H	X	h	x	□	□	α	÷	È	é	ü	ü
9)	9	I	Y	i	y	□	□	┌	┌	┌	┌	ø	ø
A	*	:	J	Z	!	z	□	□	└	└	└	└	+	+
B	+	:	K	[k	{	□	□	»	»	!»	!»	ø	ø
C	,	>	L	\	l	l	□	□	→	¼	!	!	ü	ü
D	-	=	M]	m	}	□	□	↓	½	!	!	ü	ü
E	.	<	N	~	n	~	□	□	←	¾	Ç	ç	Ä	ä
F	/	?	O	-	o	º	□	□	↑	?	Ñ	ñ	-	.