KTX12 * * * 02

Printed in Japan

M10 PORTABLE COMPUTER

Quick Reference

olivetti

INTRODUCTION

The aim of this Quick Reference is to provide the user not only with a reference list of the BASIC Commands, Statements and Functions but also a summary of the essentials of operation.

Because of its applications in the field, we believe this to be a necessary requirement. However, the user is advised first to become thoroughly familiar with the M10 using the M10 Operations Guide and to use this booklet afterwards as an "aide-memoire".

The M10 comes in two versions - the M10 MODEM which has a built-in modem and a version without modem. Where differences occur between models, these are clearly indicated in the text.

Distribution: General (G)

1st Edition: September 1983

Reference: M10 Portable Computer
Operations Guide

PUBLICATION ISSUED BY:

Ing. C. Olivetti & C., S.p.A.
Direzione Documentazione
77, Via Jervis - 10015 Ivrea (Italy)

TABLE OF CONTENTS

GENERAL INFORMATION

The M10 comes equipped with five built-in application programs:

- BASIC interpreter, allowing the user to write, run and store his own programs.
- TEXT for creating and editing text files.
- ADDRSS for maintaining a computer-based address book.
- SCHEDL for keeping an updated daily schedule of appointments etc.
- TELCOM for direct or remote data exchange with another computer.

INTERFACES

The M10 has the following independent interfaces on the rear panel.

RS-232C - the industry standard for equipment requiring a serial interface (e.g. serial printer, modem coupler etc.)

PRINTER - This Centronics standard parallel interface is used to connect a parallel printer or microplotter to the M10.

PHONE - M10 MODEM only. In the TELCOM application the M10 MODEM can be directly connected to a modular telephone line for remote data communication with a host computer.

TAPE - The M10 can be connected to a cassette tape recorder for storing files.

BCR - The M10 can be connected to a bar code reader (Hewlett-Packard HEDS-3050 or HEDS-3000) for stocktaking applications.

KEYBOARD

The M10 MODEM is equipped with a USA-ASCII keyboard. The version without integrated modem can have a UK, Italian, French or German standard keyboard (these models are named M10 UK, M10 I, M10 F and M10 D respectively).

In addition to the standard character set given by the keyboard, a further 94 special characters are available using <GRPH> and <GRPH + SHIFT> (see Figures 1 and 2).

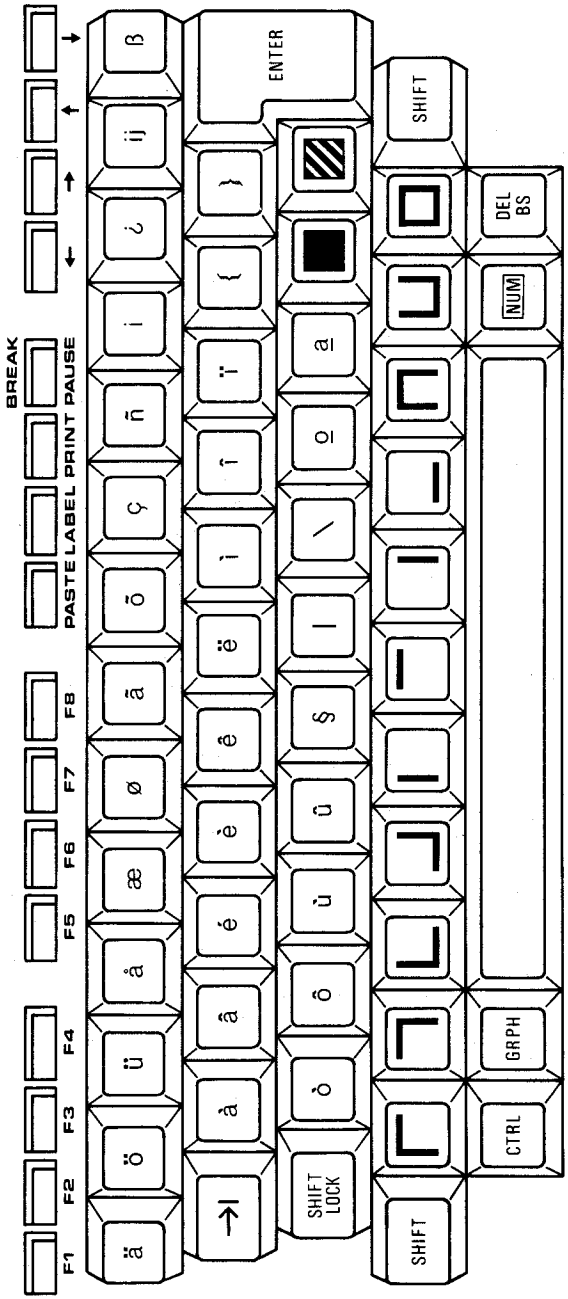


Figure 1 Keyboard When <GRPH> is Pushed

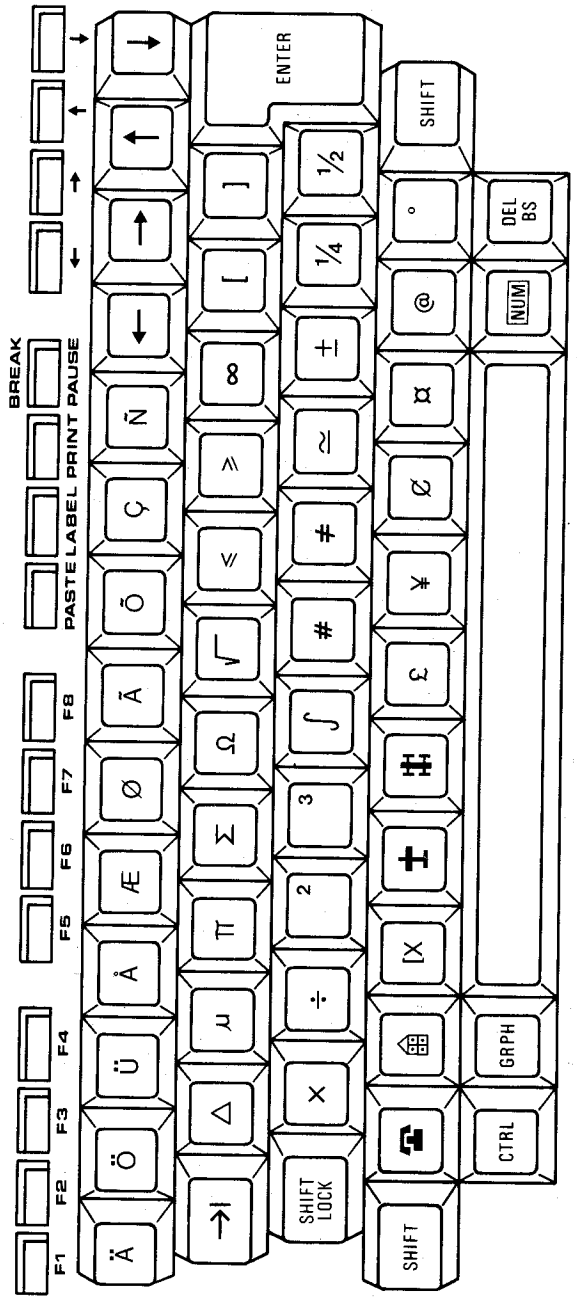


Figure 2 Keyboard When <GRPH + SHIFT> is Pushed

Cursor movement key functions are summarised in Table 1.

→	One space to the right
<SHIFT> + →	To beginning of next word
<CTRL> + →	To end of current line
←	One space to the left
<SHIFT> + ←	To beginning of last word (or current word)
<CTRL> + ←	To beginning of current line
↑	One line up
<SHIFT> + ↑	To top line of screen
<CTRL> + ↑	To beginning of file
↓	One line down
<SHIFT> + ↓	To bottom line of screen
<CTRL> + ↓	To end of file

Table 1 Cursor Movement Keys

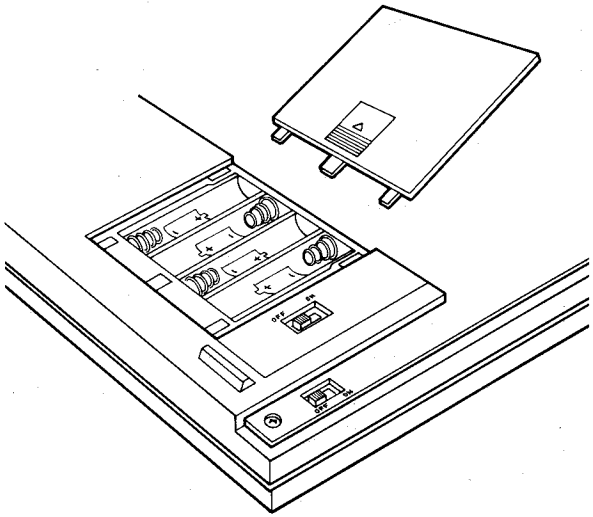
SWITCHES, BUTTONS AND CONTROLS

- On the right hand side panel, a circular control adjusts the screen contrast.
- On the underside panel are the following switches:
  - . Power ON/OFF
  - . Memory power ON/OFF
  - . DIR/ACP (M10 MODEM only; used in TELCOM)
  - . CAL/ANS (M10 MODEM only; used in TELCOM)
- At the left of the rear panel is the general RESET button to be used if the M10 "freezes" and cannot be unblocked.

In addition, the battery compartment, RAM/ROM compartment and M10 system bus compartment are located on the underside.

CHANGING THE M10 BATTERY

The M10 operates from a 6V DC supply derived either from its internal battery or from the mains supply, using an AC adaptor. The battery is installed as shown in Figure 3.



When changing the battery, first switch off all peripherals connected to the M10. Do NOT turn off the memory power ON/OFF switch as this causes the RAM contents to be lost.

When the cells start to run low, the 'Battery Low' indicator lamp on the front panel of the M10 will light up.

If the M10 is switched off, power to the memory is maintained by a Ni-Cd rechargeable battery. The RAM contents are protected for between 8 and 30 days depending on the amount of memory installed.

To preserve the battery the M10 has automatic switch-off after 10 minutes if no keys are pressed.

In normal operation, the M10 is switched on and off using the power ON/OFF switch. Always make sure that all peripherals are turned off before switching the M10 off.

THE MAIN MENU

The main menu of the M10 appears on the screen when the computer is first switched on.

The current day, date and time is displayed on the top line of the menu.

SETTING THE TIME

To reset the day, date and time on the menu, proceed as follows:

1. Access BASIC and after the **Ok** prompt, enter the following command:

   DAY$="day"<ENTER>

   where day is one of the following three-letter abbreviations:

   Mon - Monday
   Tue - Tuesday
   Wed - Wednesday
   Thu - Thursday
   Fri - Friday
   Sat - Saturday
   Sun - Sunday

   When this is accepted by the M10 the **Ok** prompt appears again.

2. Now enter the command:

   DATE$= "mm/dd/yy"<ENTER> if you have the M10 MODEM.

   or

   DATE$="dd/mm/yy"<ENTER> if you have another model.

   where

   dd is a number from 01 to 31 representing the day

   mm is a number from 01 to 12 representing the month

yy is a number from 00 to 99 representing the year

When the **Ok** prompt returns, the date has been fixed.

3. Enter the command:

TIME$="hh:mm:ss"<ENTER>

where

hh is a number from 00 to 23 representing the hour

mm is a number from 00 to 59 representing the minute

ss is a number from 00 to 59 representing the second

The **Ok** prompt returns to the screen when this is accepted.

4. Press <F8> to return to the main menu and check that the new values are displayed.

Example:

Ok
DAY$="Mon"<ENTER>
Ok
DATE$="07/24/83"<ENTER> for the M10 MODEM
or
DATE$="24/07/83"<ENTER> for other models
Ok
TIME$="10:16:35"
Ok

The M10 will register the time forward from 10:16:35 on Monday, July 24, 1983. In BASIC, the commands PRINT DAY$, PRINT DATE$ and PRINT TIME$ can be used to check the current values.

THE BASIC FACILITY

BASIC is the high-level programming language of the M10. There are three operating modes in BASIC:

Direct - This is the mode in force when the **Ok** prompt appears on the screen and it is entered automatically when you access BASIC. It is used to enter programs and immediate lines.

Execute - This mode is entered when the M10 is running a program or executing a command. The M10 returns automatically to Direct mode.

Text or Edit - This mode is entered with the command EDIT <ENTER> and it is used to edit programs. It incorporates all the features of the TEXT application program except automatic word-wrapping.

Files created in BASIC are suffixed .BA by the M10.

THE FUNCTION KEYS IN BASIC

The function of keys F1-F8 varies according to the operating mode. They are summarised in Tables 2 and 3.

KEY NAME	FUNCTION
F1 File	Lists on the screen all the files on the menu
F2 Load	Used to load a file from an external device (e.g. a cassette tape recorder) or from RAM
F3 Save	Saves the current program to an external device such as a cassette tape recorder or to RAM
F4 Run	Runs the current program
F5 List	Lists the current program on the screen
F6	Not used
F7	Not used
F8 Menu	Returns to the main menu
PASTE	Inserts the contents of the PASTE buffer at the location of the cursor
LABEL	Displays the functions of F1-F8 on the screen
PAUSE	Halts the program momentarily; program can be resumed by pressing <PAUSE> again
BREAK (SHIFT+PAUSE)	Stops execution of the program

Table 2 Function Keys in Direct Mode

KEY NAME	FUNCTION
F1 Find	Allows you to specify a string to be located in the file under edit
F2 Load	Loads a program from a cassette tape recorder
F3 Save	Saves a program file to a cassette tape recorder
F4	Not used
F5 Copy	Stores a selected string in the PASTE buffer to be copied at the location specified by the cursor
F6 Cut	Stores a selected string in the PASTE buffer and removes the original from the file
F7 Sel	Allows you to select a string for cut, copy and paste operations
F8 Menu	Exits from the Text Mode
PASTE	Inserts the contents of the PASTE buffer at the location of the cursor
LABEL	Displays the functions of F1-F8 on the screen

Table 3 Function Keys in the Text Mode

TEXT APPLICATION PROGRAM

This application program, used to create and edit text files, is equipped with a wide range of word-processing features. These include automatic word-wrapping, finding a specified character string, copying text, Cut and Paste operations etc.

On accessing TEXT, the prompt **File to edit?** appears on the screen. Enter a file name, not exceeding 6 characters in length. A file name may not begin with a number or with any of the following characters:

! " # $ % & '

( ) + * : , . /

and may not include a colon (:) or period (.). All files are automatically suffixed .DO by the computer.

When the file name has been accepted, the cursor appears as a flashing arrow-head at the top of the screen and the text can be typed in. Once a file has been created, pressing <F8> closes the file, saves it in RAM and returns the user to the main menu. The file name, suffixed .DO by the M10, will appear on the main menu.

To remove a TEXT file from RAM, access BASIC, type in

KILL"filename" (include the double quotes and the suffix .DO)

and press <ENTER>. This deletes the file.

THE FUNCTION KEYS IN TEXT

The function keys F1-F8 are used as follows in TEXT:

F1 (Find) - This function is used to find a specified character string in a TEXT file.

Press <F1> and, in response to the prompt **String:** type in the character string you are looking for and press <ENTER>. The cursor will move to the first occurrence of the string after its current position. If you wish to search the whole file, start with the cursor at the beginning of the file.

Press <F1> then <ENTER> again to find the next occurrence and so on. If the string is not found, the message **No match** is displayed.

F2 (Load) - Used to load a TEXT file from cassette tape into RAM (see "USING A TAPE RECORDER WITH THE M10").

F3 (Save) - Used to save a TEXT file on a cassette tape (see "USING A TAPE RECORDER WITH THE M10").

F5 (Copy) - Pressing <F5> copies a selected block of text into the PASTE buffer for subsequent duplication. Pressing <PASTE> inserts the contents of the PASTE buffer into the current file at the cursor location. This function may be used to duplicate text within a TEXT file or to copy text from one file to another.

F6 (Cut) - Used to delete a selected block of text from the current file and put it in the PASTE buffer. Pressing <PASTE> inserts this block at the cursor location. This function can be used to carry out 'Cut and Paste' operations in a TEXT file or from one file to another.

F7 (Select) - Used to select a block of text for the 'Copy' or 'Cut and Paste' operations. Position the cursor at the start of the desired text block and press <F7>. Now move the cursor to the end of the block, using the cursor movement keys. The selected block appears "in negative" on the screen. You can select a text block from the current cursor position onwards. All the cursor movement key combinations (with <SHIFT> and <CTRL>) can be used in a 'Select' operation. To select a text block from the current cursor location to a specified word or phrase, press <F7> then <F1>, entering the word or phrase in response to the prompt **String:** . The length of the selected text block is limited by the amount of memory available in RAM; if you exceed this, you will receive the message **Memory full.**

F8 (Menu) - This key closes and saves the file and returns the user to the main menu.

THE ADDRSS PROGRAM

This application allows the user to set up and maintain an easily accessible address book. In the case of the M10 MODEM, the TELCOM program has access to the ADRS.DO file and can use the information in it for automatic dialling of a number and automatic log-on to a host computer. The use of the ADDRSS program can be summarised as follows:

1. Access TEXT and create the file ADRS.DO.

2. Enter the names, telephone numbers and addresses you need in this file in the following form:

   Name : telephone number : address

   The number must be preceded and followed by a colon (:) if the autodial and auto log-on options on the M10 MODEM are to be used.

3. Press <ENTER> after each entry to allow the computer to distinguish between entries.

4. You can now at any time access ADDRSS and use the function keys to examine selectively the contents of the ADRS.DO file.

THE FUNCTION KEYS IN ADDRSS

F1 (Find) - Used as in TEXT.
Type in the target string and press <ENTER>. All entries containing the specified string are displayed on the screen. If you have used codes, entering the code as target string provides a list of all entries prefixed by that code.

F5 (Lfnd) - Identical to F1, except that the result is output at the printer.

F8 (Menu) - Returns the user to the main menu.

By using code letters in the entries, they can be grouped by category (e.g. profession, geographical location etc.).

THE SCHEDL PROGRAM

The SCHEDL application program allows the user to set up a daily schedule of activities and notebook containing all important information concerning appointments, travels, expenses etc. This information can be arranged and presented in a variety of ways using the SCHEDL program. The essentials of using this program are as follows:

1. Access TEXT and create the file NOTE.DO.

2. Enter your list of appointments, meetings, expenses, travel, items requiring action and, indeed, anything that is relevant to organising your day. Keep entries neat and easy to read. It is good practice to prefix entries by the date and time. They can also be prefixed by a symbol to identify the entry as an appointment, an item requiring attention, an expense etc.

3. Press <ENTER> after each entry to keep them separate.

4. You can now at any time access SCHEDL and use the function keys to examine selectively the contents of the NOTE.DO file.

THE FUNCTION KEYS IN SCHEDL

F1 (Find) - Used as in TEXT.
Type in the target string and press <ENTER>. All entries in which the target string occurs are displayed on the screen. By using the code symbol for appointments, for example, as target string you will obtain a list of your appointments.

F5 (Lfnd) - Identical to F1, except that the result is output on the printer.

F8 (Menu) - Returns the user to the main menu.

TELCOM PROGRAM

The TELCOM program allows the M10 to communicate with another computer or host system. Connection may be either direct, using the special null modem connecting cable, or remote, by means of a telephone line. There are substantial differences between the M10 MODEM and the other versions.

There are two operating modes in TELCOM, Entry and Terminal. When you access TELCOM you are automatically in Entry mode.

ENTRY MODE

This mode allows you to access Terminal mode, to select the communications parameters and provides the M10 MODEM with the auto-dial option.
In Entry mode, the function keys have the following role:

F1 (Find) - M10 MODEM only.
This key allows you to find any entry in the ADRS.DO file.

F2 (Call) - M10 MODEM only.
Gives automatic dialling of a number contained in the ADRS.DO file or manual dialling from the keyboard.

F3 (Stat) - This lists the current communications parameters and enables you to modify them.

F4 (Term) - Changes the mode to Terminal.

F8 (Menu) - Returns you to the main menu.

TERMINAL MODE

This mode is used for all data communications operations. The function keys in Terminal are as follows:

F1 (Prev) - Displays the previous eight lines of text on the screen.

F2 (Down) - Downloads incoming data from a host computer into a file.

F3 (Up) - Uploads prepared data from a file for transmission to a host computer.

F4 (Full/Half) - Commutates between full duplex and half duplex modes of data transmission.

F5 (Echo) - When the M10 is connected to a printer, this key prints what is on the screen.

F8 (Bye) - Returns user to Entry mode. The response to the prompt **Disconnect?** must be Y (Yes) to free the telephone line.

DATA COMMUNICATION PARAMETERS

The different parameters, their meaning and their range of values are shown in the table below.

Parameter	Possible Values	Meaning
Baud Rate	M	Modem (300 baud)
	1	75 baud
	2	110 baud
	3	300 baud
	4	600 buad
	5	1200 baud
	6	2400 baud
	7	4800 baud
	8	9600 baud
	9	19200 baud
Word Length	6	6 bits
	7	7 bits
	8	8 bits
Parity	O	Odd
	E	Even
	N	No Parity
	I	Ignore Parity
Stop Bit	1	1 Stop bit
	2	2 Stop bits
Line Status	E	Enable
	D	Disable
Dial Pulse Rate	10	10 pps
	20	20 pps

Table 2 Data Communications Parameters

DATA COMMUNICATIONS AVAILABLE TO M10 MODEM ONLY

The first requirement for remote data communications using the M10 MODEM is to connect the computer to a modular telephone line, as shown in Figure 4.

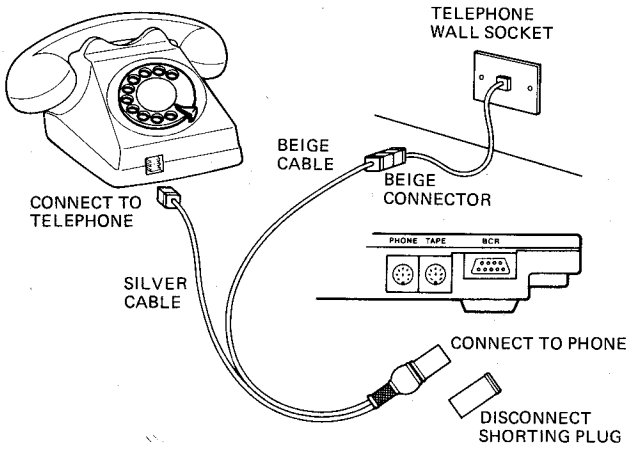


Figure 4 Connection of the M10 MODEM to a Telephone Line

If direct connection is not possible, connect an acoustic coupler to the RS-232C interface and place the cups over the telephone receiver, with the coupler microphone to the telephone loudspeaker and vice-versa.

AUTOMATIC DIALLING

1. Connect the M10 MODEM to the telephone as shown in Figure 4. Set the DIR/ACP switch to DIR.

2. Access TELCOM and press <F1> (Find).

3. Type in the name of the person whose number you require from the ADRS.DO file and press <ENTER>. The name and telephone number will appear on the screen.

4. Press <F2> (Call). The message **Calling .....** appears, where the dots represent the digits which appear one by one on the screen as they are dialled.

5. Lift the receiver before the last digit is dialled.

This operation is available only when the M10 MODEM is directly connected to a telephone line.

MANUAL DIALLING

1. Access TELCOM and press <F2> (Call).

2. In response to the prompt **Call** enter the number you require via the keyboard and press <ENTER>.

3. The digits appear as they are dialled; pick up the receiver before the last digit appears on the screen.

AUTOMATIC LOG-ON PROCEDURE

This procedure, which incorporates automatic entry into Terminal mode, is used to log-on automatically to a host system.

1. Access the ADRS.DO file and make an entry of the form:

   Host system name: telephone number < >

   where the angle brackets contain the log-on protocol for the host system (see below).

2. Verify that the M10 MODEM is directly connected to a telephone line, the DIR/ACP switch in the DIR position and the ANS/CAL switch set to CAL.

3. Access TELCOM and press <F1> (Find); in response to the prompt, type in the name of the host system and press <ENTER>.

4. When the name and number are displayed on the screen, press <F2> (Call).

5. The number is now dialled automatically, the digits appearing as they are dialled.

6. When communication with the host system is established, the M10 MODEM enters Terminal mode automatically, emitting a high-pitched sound; the designation of the function keys F1-F8 in Terminal mode appear at the foot of the screen.

The M10 MODEM will now proceed to log-on to the host system, in accordance with the instructions entered in the angle brackets < > in the ADRS.DO file. There are four special command keys to enable you to set up the log-on protocol.

? This character tells the M10 MODEM to wait until the next character in the string is received before continuing e.g. ?P means wait until the character P is received from the host before proceeding.

= This character imposes a delay of two seconds in the procedure.

! The exclamation mark tells the M10 MODEM to transmit the next character as it stands e.g. != means transmit the character = rather than wait for two seconds. The character ! allows the M10 MODEM to distinguish between special characters and transmitted characters.

^ This symbol instructs the M10 MODEM to send the next character as a control character e.g. ^C means transmit Control-C.

An example of a log-on protocol would be:

< = ^C ?U User Identity ^M ?P Password ^M >

which translated means:

1. Wait two seconds (=) then send Control-C (^C).

2. Wait (?) until a prompt containing U is received, then enter User Identity (^M is equivalent to <ENTER>).

3. Wait (?) until a prompt containing P is received, then enter Password (^M is equivalent to <ENTER>).

If such a protocol is entered between the angle brackets < > in the ADRS.DO file, it will be followed each time Terminal mode is entered automatically.

DATA COMMUNICATIONS USING THE M10 WITHOUT INTEGRATED MODEM

This requires the use of a modem coupler. The Olivetti MC 10 Modem Coupler is ideal for this application, being specially designed to operate with the M10.

1. Connect the MC 10 Modem Coupler to the RS-232C interface of the M10.

2. Set the ANS/CAL switch on the M10 to CAL.

3. Turn the power ON/OFF switch on the MC 10 to ON; the indicator lamp shows flashing green.

4. Access TELCOM on the M10 and set the communications parameters to the values recommended by the host system (300 baud maximum signalling rate).

5. Lift the telephone receiver and dial the number of the host system. When communication is established, you will hear a high-pitched continuous tone.
   Press <F4> on the M10 to enter Terminal mode.

   After a moment, the indicator lamp on the MC 10 shows continuous green.

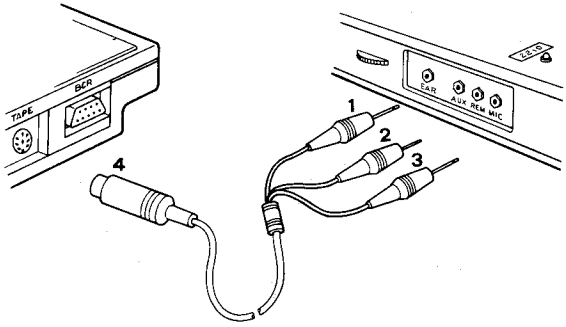
6. When Terminal mode is accessed, the M10 emits a high-pitched sound; the designation of the function keys F1-F8 appears at the foot of the screen.

7. Follow the procedure for log-on provided by the host system.

The M10 / MC 10 combination is now ready for data transmission.

USING A CASSETTE TAPE RECORDER WITH THE M10

A cassette tape recorder can be used with the M10 to save files on tape and to load files from tape to the M10. The connection should be made as in Figure 5 below.



1. WHITE TO EAR
2. BLACK TO REM
3. RED TO MIC
4. CONNECT TO TAPE

Figure 5 Connecting a Cassette Tape recorder to the M10

In addition to the normal functions, the recorder must be equipped with the following features:

- Input microphone jack MIC

- Output jack EAR

- Remote control jack REM

- Tape counter

The BASIC commands MOTOR ON and MOTOR OFF determine whether control lies with the M10 or not.

SAVING A TEXT FILE TO TAPE

1. Press the PLAY and RECORD buttons simultaneously on the recorder.

2. Access the TEXT file you want and press <F3> (Save).

3. In response to the prompt **Save to:** , type in the selected tape file name and presss <ENTER>.

The recorder will turn automatically. When the file is saved the recorder will stop and the prompt will disappear from the screen.

SAVING A BASIC FILE TO TAPE

1. Access the BASIC facility and press <F2> (Load).

2. In response to the prompt **Load "** , type in the RAM file name and press <ENTER>.

3. Now press <F3> (Save) and, in response to the prompt **Save "** , enter the command

   CAS:tape filename<ENTER>

   where tape filename is the cassette file to which the BASIC file is to be saved.

LOADING A TEXT FILE FROM TAPE

1. Set the cassette to the counter position corresponding to the start of the file to be loaded and press PLAY.

2. Access the TEXT file in which the file is to be loaded and press <F2> (Load).

3. In response to the prompt **Load from:** , type in the tape file name and press <ENTER>.

4. When the file is located the message

   FOUND: tape filename

   appears and the file is appended to the TEXT file from which you called.

LOADING A BASIC FILE FROM TAPE

1. Set the cassette to the correct position and press PLAY.

2. Access BASIC and press <F2> (Load).

3. In response to the prompt **Load "** , enter the command

   CAS:tape filename <ENTER>

4. When the file is located, the message

   FOUND:tape filename

   appears and the file is loaded as the current file in BASIC.

BASIC COMMANDS, STATEMENTS AND FUNCTIONS

The following pages give a list of BASIC Commands, Statements and Functions for quick reference. No distinction is made between these three categories. A much fuller explanation can be found in the M10 Operations Guide.

NAME	PURPOSE	FORMAT
ABS	Returns the absolute value of a numeric expression.	ABS(x) where: x is an expression
ASC	Returns the ASCII code for the first character of a string.	ASC(x$) where: x$ is a string
ATN	Returns the arctangent of an angle.	ATN(x) where: x is a number or numeric expression
BEEP	Sounds the buzzer for half a second.	BEEP
CALL	Invokes a machine language program.	CALL address , A ,HL where: address is the starting address in ROM A is an integer to be input to register A . HL is an integer, input to register HL

NAME	PURPOSE	FORMAT
CDBL	Converts a number to double precision.	CDBL(x) where: x is an integer or single precision number
CHR$	Returns an ASCII character for an integer.	CHR$(i) where: i is an integer in the range 0 to 255
CINT	Converts a number to an integer.	CINT(x) where: x is a single or double precision number
CLEAR	Resets memory space.	CLEAR string space, highest location
CLOAD, CLOADM	See under LOAD.	
CLOAD?	Used to compare a BASIC program on tape with one in memory.	CLOAD? "filename"
CLOSE	Closes access to a peripheral or file.	CLOSE file number 1, file number 2, ... ,

NAME	PURPOSE	FORMAT
CLS	Clears the screen.	CLS
COM ON /OFF /STOP	Enables and disables communication via RS-232C.	COM ON COM OFF COM STOP
CONT	Continues program execution after a <BREAK> or a STOP statement.	CONT
COS	Returns the cosine of an angle.	COS(x) where: x is an angle in radians
CSAVE, CSAVEM	See SAVE.	
CSNG	Converts a number to single precision.	CSNG(x) where: x is an integer or double precision number

NAME	PURPOSE	FORMAT
CSRLIN	Returns the vertical position of the cursor.	CSRLIN
DATA	Stores numeric and string data accessed by READ.	DATA constant-1, constant-2 ...
DATE$	Returns or sets current date.	DATE$ DATE$="mm/dd/yy" for M10 MODEM DATE$="dd/mm/yy" for European models
DAY$	Returns or sets current day.	DAY$ DAY$="xxx" where: xxx represents Sun,Mon,Tue,Wed,Thu,Fri or Sat
DIM	Specifies number of dimensions and elements in an array.	DIM x(dim1 ,... ,dimn) ,y(dim1 ,... ,dimn) where: x and y are numeric or string variables dim1 dim2 etc. are the number of elements in each dimension

NAME	PURPOSE	FORMAT
EDIT	Changes M10 from Direct to Text Mode in BASIC.	Edit n - m where: n is the first line to edit m is the last line to edit
END	Terminates the program execution.	END
EOF	Tests for end of file.	EOF(n) where: n is file number in OPEN
ERL/ERR	Gives line and error number in error handling sub-routine.	If ERL = line number THEN ... IF ERR = error number THEN ... where: line number is the line of last error error number is code number of error
ERROR	Simulates error condition or defines error.	ERROR n where: n is error number
EXP	Returns an exponential function.	EXP(x) where: $-87.365 \leqslant x \leqslant +87.365$

NAME	PURPOSE	FORMAT
FIX	Truncates a number to an integer.	FIX(x) where: x is a number
FOR...NEXT	Executes a loop a given number of times.	FOR a = n TO p STEP s . . NEXT z,y..., a where: a,y,z are variables n,p,s are numeric expressions
FRE	Returns the number of bytes free.	FRE(x), FRE(x$) where: x is a dummy numeric argument ,usually 0 x$ is a dummy string argument, usually null
GOSUB and RETURN	Used to transfer control to a sub-routine and back to the main program.	GOSUB n . RETURN where: n is the line number of the start of the sub-routine
GOTO	Causes an unconditional branch to another line.	GOTO n where: n is line number

NAME	PURPOSE	FORMAT
HIMEM	Returns the maximum address of memory available to the user.	HIMEM
IF..GOTO..ELSE **IF..THEN..ELSE**	Used to make a conditional branch in the program.	IF x GOTO n ELSE yyy IF x THEN yyy ELSE zzz where: x is a relational expression yyy and zzz are clauses containing statements and line numbers
INKEY$	Returns a string of one character equivalent to a keyboard key.	INKEY$
INP	Returns the byte at a specific port.	INP(x) where: x is the port number
INPUT	Used to input data from keyboard. Also used to input data from a file.	INPUT "prompt";a ,b ... ,n INPUT file number,a ,b ... ,n where: a,b,n are variables

NAME	PURPOSE	FORMAT
INPUT$	Returns a string of characters from keyboard or file.	INPUT$ (n , file number) where: n is number of characters to be read
INSTR	Returns the position of one string within another.	INSTR(i ,x$,y$) where: i is an integer x$ is the string being searched y$ is the string sought
INT	Returns the largest integer less than or equal to a given number.	INT(x) where: x is a number
IPL	Used to run a program immediately on start-up.	IPL"filename"

NAME	PURPOSE	FORMAT
KEY	Used to allocate a string to a particular function key.	KEY n, xxx
	Lists the current labels of the function keys.	KEY LIST
	Controls trapping to a sub-routine.	KEY (n) ON OFF STOP where: n is a function key (1-8) xxx is a string expression
KILL	Deletes a file from RAM.	KILL "filename" where: filename is the full name of the file, including suffix
LCOPY	Lists the current text image on screen at the line printer.	LCOPY

NAME	PURPOSE	FORMAT
LEFT$	Returns a string of characters from the left of a given string.	LEFT$(x$,i) where: x$ is the string i is the integer (0-255)
LEN	Returns the length of a string.	LEN(x$) where: x$ is the string
LET	Assigns a value to a variable.	LET x = a where: x is a variable a is an expression of the same type
LINE	Draws a line on the LCD or specifies a rectangle.	LINE (x1,y1) - (x2,y2) ,c LINE (x1,y1) - (x2,y2) ,c,B F where: x1,y1 and x2,y2 are the co-ordinates of the starting and finishing points of the line; $0 \leqslant x \leqslant 239$ and $0 \leqslant y \leqslant 63$ c is the colour parameter; c=1 to draw, c=0 to delete B specifies box F fills the box

NAME	PURPOSE	FORMAT
LINE INPUT	Used to input a complete line of characters to a string variable.	LINE INPUT "prompt string";a$ where: prompt string is displayed a$ is the string variable
	Also used to input complete line of characters to a string variable from a sequential file.	LINE INPUT file number,a$
LIST	Displays part or all of the current program.	LIST n - m where: n is the first line to display m is the last line to display
LLIST	As LIST but the program is displayed on the line printer.	As for LIST

NAME	PURPOSE	FORMAT
LOAD, LOADM, CLOAD, CLOADM	Used to load a program in BASIC.	CLOAD "filename" ,R LOAD "CAS:filename" ,R LOAD "RAM:filename" ,R LOAD "COM:cbpsx" ,R LOAD "MDM:bpsx" ,R (M10 MODEM only) CLOADM "filename" LOADM "CAS:filename" LOADM "RAM:filename" where: CLOAD, CLOADM load from cassette R runs the program CAS loads from cassette RAM loads from RAM c is the baud rate b is word length p is parity bit s is number of stop bits x is line condition M specifies machine language

NAME	PURPOSE	FORMAT
LOG	Returns the natural logarithm of a number.	LOG(x) where: x is a positive numeric expression
LPOS	Returns the current position of the printer head in the print buffer.	LPOS(x) where: x is a dummy argument, usually 0
LPRINT, LPRINT USING	Prints data at the line printer.	See PRINT, PRINT USING; also MICROPLOTTER commands
MAXFILES	Specifies maximum number of files to be opened or returns previously set maximum.	MAXFILES = n where: $0 \leqslant x \leqslant 15$ MAXFILES
MAXRAM	Returns highest memory address available.	MAXRAM

NAME	PURPOSE	FORMAT
MDM ON/OFF/STOP	M10 MODEM only. Enables and disables trapping to a sub-routine called if characters are received at the internal modem interface.	MDM ON MDM OFF MDM STOP
MENU	Returns user to main menu.	MENU
MERGE	Merges lines of stored program (or from communication or modem interfaces) with current program lines.	MERGE "CAS:filename" see LOAD for definitions MERGE "RAM:filename" MERGE "COM:cbpsx" MERGE "MDM:bpsx"
MID$	Returns a string from another string (returns j characters starting from the ith).	MID$(x$,i ,j) where: x$ is a string expression i and j are integers from 1 to 255

NAME	PURPOSE	FORMAT
MOD	Returns the remainder of an integer division.	n MOD m where: n is the dividend and m the divisor, both integers
MOTOR ON/OFF	Controls the cassette drive.	MOTOR ON MOTOR OFF
NAME	Used to rename a file.	NAME "aaa.xx" AS "bbb.xx" where: aaa is the old filename bbb is the new filename xx is the suffix (e.g. .BA)
NEW	Deletes current program and resets all variables.	NEW

NAME	PURPOSE	FORMAT
ON ... GOSUB	Causes branching to a sub-routine	ON COM GOSUB n ON MDM GOSUB n (M10 MODEM only) ON TIME$="hh:mm:ss" GOSUB n ON KEY GOSUB n ON x GOSUB n where: n is the first line of the sub-routine x is a numeric expression
OPEN	Opens files in RAM or cassette or allocates a buffer to a peripheral.	OPEN "CAS:filename" FOR mode AS file number In place of CAS can be RAM, COM:cbpsx, LCD, LPT (line printer), MDM:bpsx (M10 MODEM only), WAND (bar code reader) filenumber is the number allocated mode is INPUT, OUTPUT or APPEND
OUT	Used to send a byte of data to an output port.	OUT n,i where: n is the port number i is the data byte

NAME	PURPOSE	FORMAT
PEEK	Specifies the byte stored at a specific address.	PEEK(l) where: l is the memory location
POKE	Writes a byte of data into a specified memory location.	POKE l,i where: l is the memory location i is the byte of data (0 to 255, integer)
POS	Returns the current cursor position on the screen.	POS(i) where: i is a dummy argument, normally 0
POWER	Controls the time before automatic power off.	POWER i POWER CONT POWER OFF/RESUME where: i is an integer, 0 to 255
PRESET	Specifies the position of a screen element to be displayed in black or white.	PRESET(x,y c) where: x,y are the screen co-ordinates c is the colour element (default c=1, white)

NAME	PURPOSE	FORMAT
PRINT	Displays data on the screen.	PRINT list ? list PRINT n, list where: list is a list of expressions ? is an abbreviation for PRINT n is the starting position of the print-out
PRINT, **PRINT USING**	Writes data to a sequential file.	PRINT file number, data PRINT file number, USING "x"; list where: x is a string expression defining format list is a list of expressions
PRINT USING	Displays data on the screen using a specified format.	PRINT USING "x"; list where: x is a string expression defining format list is a list of expressions Note: Formatting commands may be: ! \ # + - ** $$ **$, ^^^^ (see M10 Operations Guide)

NAME	PURPOSE	FORMAT
PSET	Specifies the position of a screen element to be displayed in black or white.	PSET(x,y ,c) where: x,y are the co-ordinates c is the colour element (default c=1, black)
READ	Takes values from a DATA statement and assigns them to variables.	READ variable-1, variable-2, ..., variable-n where: variable is a numeric or string variable
REM	Prefixes a remark in a program.	REM or '
RESTORE	Used to re-read DATA statements.	RESTORE n where: n is a line number
RESUME	Continues program execution.	RESUME RESUME 0 RESUME NEXT RESUME n where: n is a line number

NAME	PURPOSE	FORMAT
RIGHT$	Returns a string of characters from the right of a given string.	RIGHT$(x$,i) where: x$ is a string i is an integer 0 to 255
RND	Returns a pseudo-random number between 0 and 1	RND(x) where: x is a number
RUN, RUNM	Used to load and run programs.	RUN n "filename" ,R RUN "CAS:filenmae" ,R RUNM "CAS:filename" ,R CAS can be replaced by RAM, COM:cbpsx, MDM:bpsx (M10 MODEM only); definitions as in LOAD

NAME	PURPOSE	FORMAT
SAVE, SAVEM	Used to store programs.	CSAVE "filename" ,R SAVE "CAS:filename" ,A SAVE "RAM:filename" ,A CSAVEM "filename",a,b,c SAVEM "CAS:filename",a,b,c SAVE "COM:cbpsx" SAVE "MDM:bpsx" (M10 MODEM only) SAVEM "RAM:filename",a,b,c where: a,b,c are addresses M specifies machine language programs See LOAD for other definitions
SCREEN	Determines which device displays output with PRINT, LIST etc.	SCREEN d ,k where: d specifies display (d=0 for LCD, d=1 for CRT) k controls display of function keys (k=0 display disabled, k=1 display enabled)

NAME	PURPOSE	FORMAT
SGN	Returns the sign of a number.	SGN(x) where: x is a number 1 for positive, -1 for negative, 0 for 0
SIN	Returns the sine of an angle.	SIN(x) where: x is an angle in radians
SOUND	Plays a specified note, enables or disables the loudspeaker.	SOUND note, length SOUND ON/OFF
SPACE$	Returns a string of spaces.	SPACE$(x) where: $0 \leqslant x \leqslant 255$
SQR	Returns the square root of a number.	SQR(x) where: x is a number
STOP	Terminates program execution and returns to Direct Mode.	STOP

NAME	PURPOSE	FORMAT
STRING$	Returns a string of a specified number of one character.	STRING$(a,b) STRING$(a,x$) where: a,b are integers between 0 and 255 (b is ASCII code) x$ is a string; first character is returned
STR$	Returns a string representation of a number.	STR$(x) where: x is a number
TAB	Used with PRINT and LPRINT to space to a particular position.	TAB(i) where: i is an integer
TAN	Returns the tangent of an angle.	TAN(x) where: x is an angle in radians
TIME$	Sets or returns the current time.	TIME$="hh:mm:ss" TIME$

NAME	PURPOSE	FORMAT
VAL	Returns the numerical value of a string.	VAL(x$) where: x$ is a string
VARPTR	Returns the address of the first byte of a variable or file.	VARPTR(x) VARPTR(n) where: x is any variable n is the file number
WIDTH	Sets the CRT width to 40 or 80 chartacters.	WIDTH n where: n is 40 or 80

MICROPLOTTER COMMANDS

The BASIC statement for printing with the microplotter is LPRINT. There are some ASCII codes allocated as instructions and these are listed below. To send them the BASIC function CHR$ must be used.

Example: LPRINT CHR$(08)

sends the ASCII code for a backspace (08) to the microplotter.

If the same form is used for an ASCII code for a printable character, the character will be printed.

Example: LPRINT CHR$(90)

prints the character z (ASCII code 90).

The Microplotter has two modes of operation:

- Text Mode (automatically entered at switch-on)

- Graphic Mode.

TEXT MODE COMMANDS

CHR$(08) Backspace; backspaces the pen one character at a time.

CHR$(11) Reverse line feed; moves the paper back one line at a time.

CHR$(18) Select Graphic Mode.

CHR$(29) Rotate pen holder; advances the pen holder one position to the next colour.

GRAPHIC MODE COMMANDS

CHR$(17) Select Text Mode.

A Return to Text Mode.

Cn Change colour; colour defined by n = 0 (black); 1 (red); 2 (green); 3 (blue).

Dx1,y1,x2,y2,x3,y3, ...

Draw; draws a line from the current pen position to the co-ordinates specified by x1,y1. The values of x and y can be from 0 to 999. The following example draws a small square providing the pen is at position 0,0 at the start:

D0,25,25,25,25,0,0,0

H — Home; moves the pen to the origin without drawing a line.

I — Initializes; this sets the origin to the current pen position.

Jx1,y1,x2,y2,x3,y3, ...

Draw relative; acts like D but the co-ordinates are in each case referred to the current pen position, e.g.:

J0,25,25,0,0,-25,-25,0

draws a small square.

Mx,y — Move; moves the pen without drawing a line to the position specified by the co-ordinates x,y, with respect to the origin.

Pstring — Print; prints the alphanumeric characters specified in string.

S size — Size; changes the size of characters printed with P. Size can be from 0 (80 characters per line) to 63 (1 character per line). Default is 0.

Q n — Print direction; specifies the direction of printing according to n; n = 0 (left to right), 1 (top to bottom), 2 (right to left upside down), 3 (bottom to top). Default value for n is 0 and at switch-on the Microplotter starts with Q 0.

Rx,y — Move relative; moves the pen without drawing a line to the position x,y with respect to the current position.

Xaxis,step,interval

Draw axis; draws the x or y axis for a graph. The axis is specified by axis which must be 1 (x) or 0 (y). The distance between measurement marks on the axis is given by step (from -999 to 999) and the number of times that step must be repeated is given by interval (from 1 to 255).

BASIC ERROR MESSAGES

ERROR CODE	NO.	DESCRIPTION
AD	53	File already open: a sequential output mode OPEN has been issued for a file that is already open; or a KILL has been attempted on an open file.
BS	9	Subscript out of range: an array element has been referenced with a subscript that is outside the dimensions of the array, or the wrong number of subscripts have been given.
BN	51	Bad file number: a statement or command references a file with a file number that is not open or is out of the range of numbers specified by the MAXFILES statement.
CF	58	File not open: the file specified in a PRINT, INPUT, or similar statement has not been opened.
CN	17	Cannot continue: an attempt has been made to continue a program that cannot be continued for one of the folllowing reasons: 1. It has stopped due to an error. 2. It has been modified during a break in execution. 3. It does not exist.
DD	10	Redimensioned array: two DIM statements have been given for the same array, or a DIM statement has been given for an array after the default dimension has been established.
DS	56	Direct statement in file: a direct statement has been found while loading an ASCII format file and the LOAD is stopped.

EF	54	Input past end: an input statement has been attempted, either after all the data in the file has been input, or for an empty file. Use the EOF function to detect the end of a file to avoid this error.
FC	5	Illegal function call: a parameter that is out of range is passed to a numeric or string function. This errror will also be produced as a result of: 1. A subscript which is negative or greater than the maximum permitted. 2. A negative or zero argument with LOG. 3. A negative argument with SQR. 4. An incorrect argument with: INP, INSTR, LEFT$, MID$, ON ... GOTO, OUT, PEEK, POKE, RIGHT$, SPACE$ or STRING$.
FF	52	File not found: a LOAD, KILL or OPEN statement references a file that does not exist in memory.
FL	57	Too many files: an attempt has been made to create a new file using SAVE or OPEN when all directory entries are full.
ID	12	Illegal direct: a statement that is illegal in direct mode has been entered as a direct command.
IE	50	Internal error: an internal fault has occurred. Contact your local dealer.
IO	18	Input/Output error: an input/output error has occurred on a cassette, printer or VDU operation. It is a fatal error, i.e. BASIC cannot recover.
LS	15	String too long: an attempt has been made to create a string more than 255 characters long.

MO	22	Missing operand: an expression contains an operator with no operand following it.
NF	1	NEXT without FOR: a variable in a NEXT statement does not correspond to any previously executed, unmatched FOR statement variable.
NM	55	Bad file name: an illegal form has been used for a file name specified with LOAD, SAVE, KILL, NAME, etc.
NR	19	No RESUME: an error trapping routine has been started but it contains no RESUME statement.
OD	4	Out of data: a READ statement has been attempted when there are no DATA statements with unread data remaining in the program.
OM	7	Out of memory: a program is too large; has too many files, FOR loops, GOSUBS, or variables; or it contains expressions that are too complex.
OS	14	Out of string space: string variables have caused BASIC to exceed the amount of free memory remaining. BASIC will allocate string space dynamically until it runs out of memory.
OV	6	Overflow: the result of a calculation is too large to be represented in BASIC's number format.
RG	3	RETURN without GOSUB: a RETURN statement has been encountered for which there is no previous, unmatched GOSUB statement.
RW	20	RESUME without error: a RESUME statement has been found before an error trapping routine has been entered.

SN	2	Syntax error: a line has been found that contains an incorrect sequence of characters, e.g. unmatched parenthesis, misspelt keyword, or incorrect punctuation.
ST	16	String formula too complex: a string expression is too long or complex. It should be split into shorter expressions.
TM	13	Type mismatch: a string variable name has been given a numeric value, or vice-versa. Alternatively, a function that requires a numeric argument has been given a string argument, or vice-versa.
UE	21	Unprintable error: an error message is not available for the error condition which exists.
UE	23-49	Unprintable error: these codes are undefined and should be reserved for future expansion in BASIC.
UE	59-255	Unprintable error: these codes are undefined and may be allocated by the user.
UL	8	Undefined line: a line reference in a GOTO, GOSUB, or IF...THEN...ELSE statement is to a nonexistent line.
/0	11	Division by zero: an attempt has been made to divide by zero in an expression, or the operation of involution results in zero being raised to a negative power.

The following table lists the error codes in numerical order.

1	NF	NEXT without FOR.
2	SN	Syntax error.
3	RG	RETURN without GOSUB.
4	OD	Out of data.
5	FC	Illegal function call.
6	OV	Overflow.
7	OM	Out of memory.
8	UL	Undefined line.
9	BS	Sub-script out of range.
10	DD	Redimensioned array.
11	/0	Division by zero.
12	ID	Illegal direct.
13	TM	Type mismatch.
14	OS	Out of string space.
15	LS	String too long.
16	ST	String formula too complex.
17	CN	Cannot continue.
18	IO	Input/Output error.
19	NR	No RESUME.
20	RW	RESUME without error.
21	UE	Unprintable error.
22	MO	Missing operand.
23-49	UE	Unprintable errors.
50	IE	Internal error.
51	BN	Bad file number.
52	FF	File not found.
53	AO	File already open.
54	EF	Input past end.
55	NM	Bad file name.
56	DS	Direct statement in file.
57	FL	Too many files.
58	CF	File not open.
59-255	UE	Unprintable errors.

NOTICE

Ing. C. Olivetti & C. S.p.A. reserves the right to make improvements in the product described in this manual at any time and without notice.

Anything in the standard form of the Olivetti Sales Contract to the contrary not withstanding, all software being licensed to Customer is licensed "as is". THERE ARE NO WARRANTIES EXPRESS OR IMPLIED INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTY OF FITNESS FOR PURPOSE AND OLIVETTI SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES IN CONNECTION WITH SUCH SOFTWARE.