

**MEMOTECH
FLOPPY DISC
SYSTEM**

**DISC BASIC
& UTILITIES**

CONTENTS

page

DISC BASIC FDXB.COM	1
OVERLAY8 and CFIG8	7
CONTACT	9

Memotech

80 Column Disc Basic

FDXB.COM

This document should be regarded as an addition to the MTX Series User manual, and should be used in conjunction with that manual. The commands described in the following section are additional to those in standard MTX Basic, and relate to file handling.

Filenames used should conform to CP/M's requirements. (For example, all Basic programs should use the extension .BAS).

All file handling commands are prefixed by DISC, e.g.

```
10 DISC SAVE "PROG.DAT"
```

except EOF, because it is a function. For a more detailed explanation see the descriptions that follow.

It should also be noted that the screen format in 80 column FDX Basic is slightly different to that in MTX Basic. The List Screen is one line longer i.e. 20 lines, and the Edit Screen is one line shorter, i.e. 3 lines.

There are a few differences between standard MTX Basic and the 80 Column Basic described here. The 80 Column Basic has been made more user friendly, and the LET and GOTO statements which are mandatory in MTX Basic are now optional.

There are two versions of Basic on the disc.

1. MTX.COM and
2. FDXB.COM

MTX.COM converts your computer back to an MTX512. FDXB.COM is the disc version of Basic. To run Disc Basic on an 80 column monitor type:

```
FDXB <RET>
```

It is also possible to run Disc Basic using a television or 40 column monitor plugged into your MTX. To do this, type:

```
FDXB 40 <RET>
```

Note that FDXB.COM supports 5¼" floppy drives only, and in order to access 8" drives and Silicon Discs it is necessary to use FBASIC.COM, which will be available shortly.

PROGRAM FILE COMMANDS

Program file manipulation:

SAVE

Format: SAVE <filename>

Purpose: To write the program that is currently in memory to Disc.

Notes: <filename> is a string expression that conforms to CP/M's requirements for filenames.

If <filename> already exists, the existing file will be overwritten.

Examples: 10 DISC SAVE "PROG.BAS"

10 DISC SAVE "STAT.BAS"

LOAD

Format: LOAD <filename>

Purpose: To load a file from disc into memory

Notes: <filename> is a string expression that conforms to CP/M's requirements for filenames.

<filename> is the name that was used when the file was SAVED. LOAD deletes all variables and program lines currently residing in memory before it LOADs the designated program.

Information may be passed between programs using their disc data files.

Examples: 10 DISC LOAD "PROG.BAS"

10 DISC LOAD "STAT.BAS"

OPEN

Format: OPEN# <channel no>,<filename>,<type>,(<reclen>)

Purpose: To allow Input/Output to a disc file.

Notes: A disc file must be OPENed before any disc I/O operation can be performed on that file.

OPEN determines the mode of access that will be used with the disc buffer.

<type> is a string expression whose first character is one of the following:

O = sequential output

I = sequential input

R = random input/output

<channel no> is an integer expression whose value is between one and four. The number is associated with the file for as long as it is OPEN, and is used to refer other disc I/O statements to the file.

<filename> is a string expression containing a name that conforms to CP/M's rules for disc filenames.

<reclen> is an integer expression which, if included, sets the record length for random files.

Examples: 10 DISC OPEN #2,"PROG.BAS","O"

KILL

Format: KILL# <channel no>

Purpose: To close or erase a currently open file.

Notes: <channel no> is an integer expression whose value is between one and four. The number is associated with the file for as long as it is OPEN, and is used to refer other disc I/O statements to the file.

Examples: 10 DISC OPEN #2, "TEMP.DAT", "O"
20 DISC KILL #2

CLOSE

Format: CLOSE # <channel no>

Purpose: To conclude Input/Output to a disc file.

Notes: <channel no> is an integer expression whose value is between one and four. The number is associated with the file for as long as it is OPEN, and is used to refer other disc I/O statements to the file.

<channel no> is the number under which the file was opened.

The association between a particular file and channel number terminates upon execution of a CLOSE. The file may be reopened using the same or a different channel number; likewise, that channel number may now be used to OPEN any file.

A CLOSE for a sequential output file inserts an EOF (End Of File) marker.

Examples: 10 DISC CLOSE #2

INPUT #

Format: INPUT# <channel no>,arguments

Purpose: To read data items from a disc file and assign them to program variables.

Notes: <channel no> is an integer expression whose value is between one and four. <channel no> is the number under which the file was opened.

The arguments are the variable names that will be assigned to the items in the file.

N.B. The variable type must match the type specified by the variable name.

No question mark is printed.

The data items in the file must appear in the same way they would if data were being typed in response to an INPUT statement.

If End Of File is reached when a numeric or string item is being INPUT, the item is terminated.

With numeric values or strings, the number terminates on a carriage return and line feed or comma.

N.B. Line feed followed by carriage return does not act as a string delimiter.

Examples: 10 DISC INPUT #2,A\$,B\$

LINE INPUT #

Format: LINE INPUT #<channel no>,arguments

Purpose: To read an entire line from a disc data file to a string variable.

Notes: <channel no> is an integer expression whose value is between one and four. <channel no> is the number under which the file was opened.

The arguments are the string variable names that will be assigned to the items in the file.

LINE INPUT # reads all characters in the file up to a carriage return/line feed. The next LINE INPUT # reads all the characters up to the next carriage return/line feed. The carriage return/line feed sequence itself is skipped over.

N.B. If a line feed/carriage return is encountered, it is preserved.

PRINT

Format: PRINT #<channel no>,<list of expressions>

Purpose: To write data to a sequential or random access disc file.

Notes: <channel no> is an integer expression whose value is between one and four.
<channel no> is the number under which the file was opened.
The expressions in <list of expressions> are the numeric and/or string expressions that will be written to the file.

REC

Format: REC #<channel no>,<logical record no>

Purpose: Positions the pointer in the file at the record number specified.

Notes: <channel no> is an integer expression whose value is between one and four.
<channel no> is the number under which the file was opened.
The next record READ after this command will be the one specified in <logical record no>.

Example: 10 DISC REC #1,7

EOF

Format: EOF (<channel no>)

Purpose: Places an End Of File marker after the last record in the file.

Notes: <channel no> is an integer expression whose value is between one and four.
<channel no> is the number under which file was opened.
EOF is a function and not a statement.

Example: 10 IF EOF(3)<>0 THEN STOP

WRITE

Format: WRITE "<filename>",<start address>,<no of bytes>

Purpose: To WRITE a block of memory to disc.

Notes: <filename> is a string expression containing a name that conforms to CP/M's rules for disc filenames.
<start address> and <no of bytes> are in decimal and can be numeric expressions.

Example: 10 DISC WRITE "TEST.DAT",17000,100

READ

Format: READ "<filename>",<start address>

Purpose: To READ a block of memory from disc.

Notes: <filename> is a string expression containing a name that conforms to CP/M's rules for disc filenames.
<start address> is in decimal.

Example: 10 DISC READ "TEST.DAT",17000,100

DIR

Format: DIR (string expression)

Purpose: To list the files on Disc

Notes: DIR is used in the same way as CP/M uses it.

Example: 10 DISC DIR "*.BAS"

ERA

Format: ERA (string expression)

Purpose: To erase files on Disc

Notes: ERA is used in the same way as CP/M uses it.

Example: 10 DISC ERA "*.DAT"

Disc Basic Display Formats Text and Graphics

The FDX display resolution is 80 characters by 24 rows in TEXT mode and 90×160 in graphic mode. There are 8 ink colours, 8 paper colours and 4 character sets.

If you use RGB output, the colours available are:

0 Black	1 Red
2 Green	3 Yellow
4 Blue	5 Magenta
6 Cyan	7 White

If you select Black and White output, the INK and PAPER commands cause the following:

INK

0 Normal	1 Underline
2 same as 0	3 same as 1
4 Bright	5 Bright and Underline
6 same as 4	7 same as 5

PAPER

0 Normal	1 same as 0
2 Reverse Video	3 same as 2
4 Background	5 Background and Reverse
6 same as 5	7 same as 5

An 80 column graphics screen is created on the FDX using the CRVS command.

e.g. CRVS 6,3,0,0,80,24,80

A type 3 screen has no user characters or SPRITES. However, user characters and SPRITES may be defined, and SPRITES controlled from within a type 3 screen. The SPRITES will not appear on a type 3 screen, but will appear on a type 1, (the MTX graphics screen.)

In a type 3 screen the COLOUR and ATTR commands have been slightly altered.

COLOUR n,m

n = 0 Print ink
 1 Print paper
 2 Non-print ink
 3 Non-print paper
 4 No effect

m = 0 to 7 as described above.

ATTR n,l

Print attributes:

n = 0 Inverse print
 1 Colour inverse
 2 Alternative character set
 3 Graphics set
 4 Flash

Non-print attributes:

n = 5 Unplot
 6 'Exclusive-or' plot (Over plot)
 7 Flash

All inverse characters are normally printed in the logical B&W inverse (including the cursor).
 The effect of the command

ATTR 1,1

is that all inverse characters are printed in the true colour inverse.

CHARACTER SETS

There are four character sets:

1. Standard ASCII + special graphics
2. Alternate ASCII + special graphics
3. Graphics
4. Alternate graphics

These are selected using a combination of the print attributes 2 and 3.

	ATTR 3,0	ATTR 3,1
ATTR 2,0	Chr Set 1	Chr Set 2
ATTR 2,1	Chr Set 3	Chr Set 4

Characters from all four sets may be mixed on one Basic screen. NODDY will only display characters from one set only.

All screen types (types 0,1 and 3) are available in all versions of Disc Basic.

Type 0: 40 column text on 40 column monitor or television.

Type 1: 32 column text and 256 × 192 graphics on 40 column monitor or television.

Type 3: 80 column monitor.

From this it can be seen that a television and an 80 column monitor can be used simultaneously.

OVERLAY8 and CFG8.

OVERLAY8 is a utility program which enables Televideo format discs to be read and written on Memotech FDX computers, this facility is provided to enable MTX users to purchase third party software on this disc format and to exchange data to/from Televideo computer systems. It is accompanied by the latest version of the configure program CFG8.COM. When OVERLAY8 is run, simply by typing its name, it will overlay your current disc handler (at FA00 . . . FFFF) which must be version 3, with a modified version supporting the new config code 8. The price of this support is the loss of config codes 0,1,4 and 5 – all single density formats, rarely used. The program works by effectively performing a cold boot sequence from the currently logged disc, consequently the operating system is restarted from scratch and the current logged disc becomes drive A:. At this point the user may configure his other drives to type 8 and perform file read/writes to Televideo format discs. FORMAT and RCHECK will not work with type 8 discs, furthermore SYSCOPY, WRTBIOS and WRTCPCP will not perform meaningfully and may destroy data. Sispooled printer output will not survive the overlay process. Make sure that the disc in the currently logged drive is bootable before running OVERLAY8.

We suggest that a separate system disc is prepared to support the Televideo format. This should be done as follows:

With a standard system disc in drive B:

format a clean disc in drive C:

```
FORMAT C:<RET> . . . <RET>
```

copy the system onto the disc, i.e.:

```
SYSCOPY C: <RET>
```

copy the following files to the disc using PIP. (See section on PIP).

1. START.COM
2. PIP.COM
3. CONFIG.COM
4. COLDBOOT.COM
5. OVERLAY.COM
6. CFG8.COM

Use this newly formatted disc as a special system disc for supporting the televideo format.

Re-boot with this new disc in drive B.

Run a new "start-up" as follows:

```
STARTUP CFG8 B:03, C:03 <RET>
```

```
STARTUP <RET>
```

```
COLDBOOT <RET>
```

then run OVERLAY8 and then configure drive C to the Televideo® format type 8

```
OVERLAY8 <RET>
```

```
CFG8 C:08 <RET>
```

Drive C is now Televidio compatible and files may be copied from C to B using PIP. The files on the disc in drive B will be in standard MTX format.

CONFIG 8 FORMAT

Type 8 discs are double sided, double density and 48 TPI, they have 18 sectors per track, 40 tracks and 256 bytes per sector, no logical sector interleaving is performed, the first two physical tracks are not used by the file system.

The CONTACT Communications Program

INTRODUCTION

The CONTACT program, and its CP/M support package, (H,I,R & T) provides a comprehensive and powerful communications facility for use with MEMOTECH Computer Systems. The software makes extensive use of hardware facilities available on these machines and consequently will not run on other machines. The CONTACT support programs however are designed to run on any CP/M machine, their function is to provide a suite of utilities, running on a remote machine, to facilitate the transfer of text and binary files to and from the CONTACT host computer.

When CONTACT is running, the computer system is transformed into a terminal device working from one of the two serial ports on the machine. Anything typed at the keyboard is sent down the line and anything received from the line is displayed on the screen. Thus the CONTACT host machine may be used as the console (or a terminal) to the remote computer. Once operating in this way local files may be transmitted down the line (in straight ASCII or as a HEX image) with due regard for one or both of the 2 available flow control protocols. Also any ASCII or HEX image data received from the line may be deposited in a local CP/M file – note that characters are received under interrupt control, and disc accesses during data transfers will not cause any loss of data.

The CONTACT program may be used to communicate with a variety of remote computer systems including micros, minis and mainframes; ASCII file transfers from a remote machine to the CONTACT machine are performed without the need for any flow control protocol as data can be displayed and dumped to disc faster than the highest serial transmission rate. File transfers to a remote computer are a little more complex since some kind of flow control will almost certainly be required to allow the remote computer time to dump its data into file store. Two common protocols are provided to achieve this: XON/XOFF which is always active and an optional protocol which involves the transmission of a line at a time in response to a prompt character sent from the remote machine.

These options and others are invoked by means of the CONTACT control menu which is rapidly swapped onto the screen when a control-G character is typed, after the interaction with the menu the user can simply return to the CONTACT screen by typing CR whereupon the original display is swapped back in and updated with any characters received during the interaction.

When the CONTACT program is terminated, a file (LAST.LNK), is saved on disc which contains its status information. On re-entering CONTACT this status information is retrieved and the previous operating mode restored. A further refinement is provided whereby if the LAST.LNK file is renamed (to say UNIX.LNK) and CONTACT is invoked quoting the new name (CONTACT UNIX) on the command line then CONTACT will use this instead of the "LAST.LNK" file to obtain its default status information.

USING CONTACT

The CONTACT program may be called up from CP/M in a variety of ways, whichever is used the CONTACT status information will always be saved in "LAST.LNK".

CONTACT	This command causes CONTACT to search the current default disc for "LAST.LNK". If found the file will be used to setup the initial state of CONTACT. If not CONTACT will adopt its default state. "LAST.LNK" will be saved on the default disc.
CONTACT name	With this form of command CONTACT will search the current default disc for "name.LNK" and, if found, will use it to setup its initial state. If the file is not found then a warning message will be generated and CONTACT will go to its built-in default state.
CONTACT d:name	CONTACT will search drive d: for "name.LNK" and save "LAST.LNK" on drive d:.
CONTACT d:	Drive d: is searched for "LAST.LNK".

Once CONTACT is running, a free dialogue may be conducted with the remote machine; the CONTACT host may be used just as if it were a simple terminal. The only exception to this is that if ^G is typed it will not be transmitted but will result in the CONTACT control menu being swapped onto the screen. At this point, if you really do want to send a ^G character, type another one and it will be transmitted, the menu will disappear and the original CONTACT screen restored.

While the CONTACT control menu is active characters received from the line are buffered up (under interrupt control) for processing as soon as the user returns to the CONTACT screen. The state of the buffer is indicated on the second line of the STATUS BOX, located in the top right hand corner of the menu display. Two HEX numbers are given indicating the number of characters currently in the buffer and its total capacity. Should the buffer reach three-quarters full, CONTACT will send out a ^S or XOFF character in an attempt to halt the data flow, if this attempt fails, after a few more characters, the CONTACT screen will be automatically swapped back to avoid data loss. If, when CONTACT sent its ^S character, the data flow **did** stop then CONTACT will, at an appropriate time when the buffer is emptying, send a ^Q or XON character in order to allow data flow to re-commence.

Once active, the CONTACT control menu provides the user with a powerful set of single-keystroke commands and, by means of the STATUS DISPLAY BOX, provides a constant visual indication of the operational state of the CONTACT.

MENU COMMANDS AND OPTIONS

X To exit back to CP/M.

Before exiting back to CP/M the X (or ^C) command will close any existing files, shut-down the CTC interrupts and supervise the saving of the "LAST.LNK" status file. In the event of BDOS errors occurring while CONTACT is running the user should always type a ^C to invoke an emergency shutdown of CONTACT. An emergency shutdown will not attempt to close any files or save "LAST.LNK" but it will close down the CTC interrupts and return safely to CP/M.

CR To return to the CONTACT screen.

Type carriage-return to return to the CONTACT screen and allow the buffer to empty.

E To erase the CONTACT screen.

The E command erases the CONTACT screen and restores default screen attributes, an immediate return to the CONTACT screen is implied.

^G To transmit a ^G character.

Type control-G to send a ^G to the remote site and return to the CONTACT screen.

DEL To force a BREAK.

Entering a DELETE character forces three one-second duration breaks on the active comms. channel. The CONTACT screen is automatically restored after the break sequence is complete.

ESC To change channel.

Typing an ESCAPE character causes the serial port to toggle back and forth between channels A and B.

B To select a new baud rate for the current channel.

The B command swaps in a baud rate select menu which enables the user to change his current operating speed. Rates available are 300, 600, 1200, 2400, 4800 and 9600 baud.

+, - To select EVEN or ODD parity

0,8 To select NULL parity or RAW 8-bit

EVEN/ODD PARITY forces all data transmissions to 7-bits with even/odd parity, whilst received data is stripped down to 7-bits before further processing. NULL PARITY forces data transmissions to 7-bits with a zero parity or eighth bit, received data is stripped to 7-bits. In RAW 8-BIT mode data transmissions are left unaltered and all 8-bits of the serial output are used, indeed ASCII file transmissions may include several trailing ^Z characters. Received data will also be passed to file unaltered, ^Z characters will be allowed through without causing file closure. In any of the 7-bit modes ^Z will cause any receiving file to be closed; similarly ^Z characters found at the end of local files will not be transmitted. Note that in order to allow full 8-bit data to reach the

screen, the SCREEN FILTER must be switched off. It is not recommended to use 8-bit mode for the purposes of transferring binary files – if possible use the HEX conversion facilities.

P To enter a new prompt character.

Typing a P causes a sub-menu to be activated requesting the user to enter the new PROMPT CHARACTER. The next key typed, whatever it is, becomes the new PROMPT CHARACTER used in flow controlled file transmits. The STATUS DISPLAY BOX contains a copy of this character which, if it is a control code, is displayed in reverse video. See the section on flow control for a full explanation of the PROMPT CHARACTER.

S To switch the screen filter ON/OFF

Typing an S toggles the SCREEN FILTER between the ON and OFF states. The SCREEN FILTER allows the selection of 2 display modes. With the FILTER on, ASCII data is stripped to 7-bits and the only control codes to take effect are: NULL(0), BS(8), TAB(9), LF(10) and CR(13). All other control codes are represented in reverse video but otherwise take no effect. In addition an indication of file transfer activity is provided by the brightness of the display – a high intensity display indicates no file activity, while a low intensity image indicates that a file transfer is in progress. This feature should not be relied upon to provide accurate information regarding individual characters at the start and end of transfers – it is intended only as a guide to file activity. When the SCREEN FILTER is off received data is passed directly to the internal screen handler which interprets all control codes in the usual way (as described in the MTX computer documentation). The high/low intensity file activity indicator feature is lost.

D To select HALF/FULL duplex.

Enter D to toggle between FULL and HALF duplex modes. In FULL DUPLEX, characters typed at the keyboard and files selected for transmission are sent down the line, data received from the line is sent to the screen and to any selected receiving file. In HALF DUPLEX, characters entered at the keyboard and files selected for transmission are sent not only to the line but also to the screen, received data is treated in the same way as in FULL DUPLEX.

F To switch flow control ON/OFF.

Use F to toggle flow control on or off. FLOW CONTROL may be enabled when communicating with computer systems which require and support this protocol. It takes effect only when a file is being transmitted and results in the file being sent a line at a time in response to a PROMPT CHARACTER issued by the remote computer. The local file must be in the form of ASCII records delimited by CR or CR/LF codes, transfer commences as soon as the PROMPT CHARACTER is detected from the remote machine. It is halted immediately after the next CR code has been transmitted. Any LF characters in the file are filtered out. The remote machine, on receiving the CR, will process the record and respond by sending its prompt string, the last character of which should be entered as the CONTACT PROMPT CHARACTER.

The next four commands; R,T,I and H all operate on local disc files. Consequently, before the command is executed, the user is prompted to enter a new file name. At this point, typing a single CR will leave the current file name unaltered (the current file name appears in the status box) whilst typing a ^C will abort the command entirely. Normally however, the user will simply want to enter the name of a CP/M file, this may be done using the usual line editing facilities including BS, DEL and ^X. The FILE NAME must unambiguously specify the name of the local CP/M file to be used in the next file transfer operation. It may be up to 8 characters long and be optionally preceded by a valid drive identifier, delimited from the file name by a colon in the usual way, and followed by an optional 1 to 3 character file type string, delimited by a full stop. E:Z, FRED.DAT, Z, A:L.L, and B:FILE1234.COM are all valid file-names. CONTACT will reject file-names containing wildcard operators.

R To receive into a file.

The receive-into-file command will cause all characters received from the line (including any data received and buffered during the menu interaction) to be placed into the currently named file until either a ^Z character is received (applies only in the 7-bit modes), the file is closed by user intervention or until transfer is suspended by user intervention. Before any data transfers take place the specified file is checked; if it does not already exist it will be created, if it does exist a sub-menu will appear prompting the user for further action. Options available include ABORT command, OVER-WRITE the file, and APPEND new data to the file. Two variations of

APPEND are supported, both involve reading the existing file until the end-of-file condition is satisfied before continuing to write. In an ASCII APPEND, end-of-file is reached when a ^Z character is read. A BINARY APPEND will ignore ^Z codes found in the file and read right through to the end of the last disc sector allocated to the file.

T To transmit from a file.

T will attempt to open the specified file, if it cannot be found an error message will be generated and no further action taken. If the file is opened successfully then transmission will start as soon as the CONTACT screen is returned to. Transmission will continue until either the end of file condition is satisfied (physical end in 8-bit mode, ^Z in the 7-bit modes), the file is closed by user interaction, transfer is suspended by user interaction, a ^S character is received, or until a CR is sent while FLOW CONTROL is on. Once transfer has been suspended by the reception of a ^S character it may be resumed by either the reception of a ^Q or by the user invoking the CONTACT control menu (with ^G) and returning straight back to the CONTACT screen (with CR). If FLOW CONTROL is on and transfer has been suspended by the transmission of a CR, then data flow will re-commence once either a prompt character is received, a ^Q character is received or by the user invoking the control menu and returning back to the screen. NOTE that no terminating characters are sent by CONTACT at the end of a file transmit – it is the user's responsibility to send a suitable termination sequence at the appropriate time, manually. For example when sending a file to another CP/M machine or to a UNIX machine; use ^Z, alternatively an ICL mainframe may well require ++++.

I To receive & decode a HEX image

This command is used for the reception of binary files from remote machines that have the capability to generate standard INTEL HEX format images. The only restriction on this format is that it MUST represent a continuous data image, ie there must be no gaps in the data. Consequently HEX output files from assemblers may be unreadable if Define Storage pseudo ops or if more than one ORG pseudo op has been used. The way around this limitation is to read the hex image as a ASCII file (using R) into a file named "name.HEX" and then use CP/M's LOAD utility to convert it back to binary. As data is received it is interpreted and checked for a variety of possible error modes before being converted and written to file in binary form. It is therefore virtually impossible for any undetected errors to be introduced into a binary file if it is transferred in this way. Any errors which are spotted cause immediate file closure and are flagged in the usual way, note that non-HEX characters are ignored in HEX image data streams provided they do not form part of a HEX record. Received data will continue to be read as a HEX image until either a HEX end-of-file record is detected (:00XXXXXXXX), a ^Z is received, transfer is suspended by user intervention, the file is closed by user intervention or until the error is detected in the image. Apart from the ASCII to binary conversion this command operates in the same way as the R command.

H To transmit the HEX image of a file.

Use the H command to transmit an INTEL HEX format image of a binary, or ASCII, file. The file is read as a series of 8-bit bytes (up to its physical end, disregarding ^Z characters) and converted to a HEX image ASCII data stream for transmission down the line. Apart from this binary to ASCII conversion the H command operates in the same way as the T command.

The last four commands above, R,T,I & H, all result in a specified file being opened for reading or writing. Once the file is open the CONTACT control menu changes to the FILE-ACTIVE state in which the R,T,I and H commands no longer exist but are replaced by the C and Z commands until such time as the file is closed.

C To close the file.

Use the C command to halt file transfers prematurely, or to close received files that were not terminated by a ^Z in the data stream or to close files received in the 8-bit RAW mode. When a received file is closed prematurely, data in the buffer awaiting processing will not be included in the file, only that which has been displayed up to point when the FILE-ACTIVE menu was invoked (with ^G) will be saved.

Z To suspend/resume file transfer

Use the Z command to temporarily suspend file transfers whenever necessary. A second use of the Z command, at a later time, will allow the transfer to continue from the point at which it was suspended.

THE CONTACT STATUS DISPLAY BOX

The CONTACT status display box, which appears in the top right hand corner of the control menu screen, provides the user with a constantly updated display of CONTACT's current status. The items displayed are as follows:

SOFTWARE RELEASE NUMBER	eg.	RELEASE: 041282
BUFFER STATUS	eg.	0000 / A416
CURRENTLY ACTIVE CHANNEL	eg.	CHANNEL A
CURRENT BAUD RATE	eg.	9600 BAUD
CURRENT FILE NAME	eg.	B:UNIX.DAT
CURRENT FILE STATUS	eg.	HEX RECEIVE
TRANSMISSION STATUS IF SUSPENDED	eg.	(SUSPENDED)
DUPLEX MODE	eg.	FULL DUPLEX
SCREEN FILTER STATE	eg.	SCRN FILTER ON
FLOW CONTROL STATE	eg.	FLOW CNTRL OFF
PROMPT CHARACTER	eg.	PROMPT: >
PARITY	eg.	RAW 8-BIT

Certain items from this status list are saved in the CONTACT setup file "LAST.LNK" when CONTACT is terminated, these, together with their allowed values, are as follows:

PARITY	Can be EVEN, ODD, NULL or RAW 8-BIT
SCREEN FILTER	Can be ON or OFF
FLOW CONTROL	Can be ON or OFF
PROMPT CHAR.	Can be any 7-bit ASCII code, inc. control codes
FILE NAME	Can be any unambiguous file name.
CHANNEL	Can be either channel A or B
DUPLEX	Can be either HALF or FULL
BAUD RATE	Can be 300, 600, 1200, 2400, 4800 or 9600

FILE TRANSFERS**RECEIVING AN ASCII FILE**

To receive an ASCII file from a remote computer (or any source of serial data) to which the CONTACT host is connected as a terminal or console device, follow the steps outlined below:

- 1 Make sure that CONTACT is setup properly, ie that the appropriate channel and parity have been selected. Use the CONTACT control menu to check and adjust these as necessary then use CR to return to the CONTACT screen.
- 2 Establish communication with the remote equipment and type the appropriate command line to cause it to transmit the file, for example this might be just "TYPE name". However do not yet press RETURN.
- 3 Type ^G and enter the CONTACT menu, give the R command, enter the required file name and take suitable action if CONTACT informs you that the file already exists, then use CR to return to the CONTACT screen.
- 4 Now type another CR to initiate the command on the remote machine. All the data produced will now not only be displayed on the CONTACT screen but also fed to the local file. The screen intensity should decrease as an indication of file activity.
- 5 Once the data stream stops, or you have collected enough, return again to the CONTACT menu and enter the C command to close the local file, this may have been done automatically if the remote machine sent a ^Z terminating character. The data has now been captured.

TRANSMITTING AN ASCII FILE

To transmit an ASCII file to a remote computer (or any device capable of absorbing serial data) to which the CONTACT host is connected as a terminal or console device:

- 1 Setup CONTACT appropriately: check channel number, parity, flow control and prompt character – if required. Return to CONTACT screen with CR.
- 2 Issue the command which prepares the remote machine to accept data entered from its terminal. On UNIX machines this may be "rt", on an ICL VME/B machine "INTOF(-----)", a CP/M machine may be able to use PIP for files smaller than its data buffer, but in general a purpose written program will be required such as "R" which is supplied in the CONTACT support pack.
- 3 Enter the CONTACT menu with ^G and issue the T command, after entering the file name return to the CONTACT screen (with CR) and allow the data to flow. Some remote machines may not ECHO the transmitted data so, to keep watch on the data transfer you may wish to select the HALF DUPLEX mode.
- 4 When the file has been exhausted CONTACT will automatically close it. However it will now be necessary for you to enter the appropriate terminating sequence to close the file on the remote machine. This is done simply by typing it.

TRANSMITTING AND RECEIVING HEX FILES

This is done in much the same way as with ASCII files, HEX images are, after all, just a special kind of ASCII file. HEX transfers are generally used when an exact binary image of a file is to be moved, as when copying a program file from one machine to another. The remote machine does not have to be able to generate or decode these HEX images (it may merely be storing them in ASCII form), but the CONTACT support package for CP/M is able to do this and it's therefore recommended that these features be made use of when communicating with remote CP/M machines. For example, Word Star files, whilst being text files, carry information in the eighth data bit and as such require binary image transmission, best done using HEX conversion.

MISCELLANEOUS NOTES

ERROR MESSAGES

Error message outputs are requested by CONTACT's internal subroutines as and when error conditions arise, these messages are not sent to the CONTACT screen but are held until the user enters the CONTACT control menu where they are displayed prominently at the top of the screen.

CHANGING DISCS

Floppy discs may be changed at any time while CONTACT is in it's FILE INACTIVE state. They MUST NOT be changed while in the FILE ACTIVE state. Use ^G to enter the CONTACT control menu and check that the file is INACTIVE.

TERMINAL EMULATION

CONTACT contains its own internal and independent screen drivers and so if a terminal emulator overlay is running on the host machine it will be ignored. It is however possible to re-write and re-link CONTACT's screen driver and thus emulate other terminals to some degree, please refer back to MEMOTECH.

This file was downloaded from

www.primrosebank.net

Abridged Terms & Conditions (Downloads)

Disclaimer

www.primrosebank.net, (*the website*) is provided by Dave Stevenson as a service to the public, is provided "as is" and carries no warranties, expressed or implied, of any kind.

Dave Stevenson is not responsible for, and expressly disclaims all liability for, damages of any kind arising out of use, reference to, or reliance on any information contained within the website or made available for download. Whilst the information contained within the website site is periodically updated, no guarantee is given that the information provided on the website is correct, complete, and up-to-date.

A number of articles on the website contain technical data and practical guidance which may be of use in testing and maintaining various items of vintage computer and electronics hardware. Such articles are not intended to cover all aspects of the tasks involved and may omit essential information, including necessary safety precautions. Performance of the tasks described may risk damage to equipment and/or people. The reader is responsible for ensuring that he/she is capable of performing the tasks described and well as assessing the inherent risks involved and taking appropriate measures to mitigate such risks.

Dave Stevenson expressly disclaims all liability for, damages to equipment or injury of any kind arising out of use of such technical data and guidance.

Unless otherwise noted, all data on the website is deemed to be *Copyright (c) Dave Stevenson, 2009-2013*

You are hereby granted permission to download data and software from the website *for your own personal use. Redistribution of any content from the website without written authorisation from Dave Stevenson is expressly forbidden. You are also expressly forbidden from offering for sale any material obtained from the website.*

As far as possible, information included on the website from other sources has been credited to the respective author and/or publisher. The majority of content on the website is derived from material first published in the 1980s. *This material is likely still under copyright of the original author and/or publishers.* The authors and/or publishers may not have given express permission to copy, transmit or make this information available for download, but I believe that they would have no objection to this archive information being placed into the public domain.

However, should the author and/or publisher of the original material find any content on the website for which they wish to assert their rights, they should notify Dave Stevenson (by e-mail to: webmaster@primrosebank.net) who would be pleased to enter into a dialogue to agree a satisfactory resolution of their concerns.

If you obtained this file as part of a paid-for package, you have been scammed! I suggest that you request a refund from the seller, please also advise Dave Stevenson at the e-mail address above.