UNIVERSITY OF EDINBURGH
Edinburgh Regional Computing Centre

User Note 94

(June 1987)

Title:

KERMIT on EMAS-3

Author: C.J.Adie, A.Albert-Recht

Contact: Your User Support Team

Software Support Category: n/a

Synopsis

This User Note describes how to use the implementation of KERMIT on EMAS-3. It is intended to provide enough information for someone with a little previous exposure to Kermit to be able to transfer data between EMAS-3 and another Kermit system on a microcomputer.

The information in this edition applies to version 4.

Keywords

File transfer, KERMIT

Edinburgh Regional Computing Centre

James Clerk Maxwell Building, The King's Buildings, Mayfield Road, Edinburgh, EH9 3JZ. Telephone 031-667 1081

CONTENTS

1: INTRODUCTION

This user guide describes the Kermit implementation on EMAS-3. It is intended to provide enough information for you to be able to transfer data between EMAS-3 and another Kermit system, but it does not go into great detail about the Kermit philosophy. User Note 78 (1987 edition or later) is an introduction to Kermit if you are completely unfamiliar with it. Other Kermit systems are described only in passing: thus you would usually need to consult the user guide for your microcomputer. The EMAS end of the transfer is always referred to as "EMAS Kermit", while the other end is referred to severally as "local Kermit", "micro Kermit" or just "other Kermit".

The guide is divided into several chapters. Chapter 2 is a comparison of EMAS Kermit with X-Talk, another microcomputer file-transfer package in use in Edinburgh. Chapter 3 describes the command language that EMAS Kermit uses. Chapters 4 and 5 describes how to use EMAS Kermit to transfer files, while the last chapter comprises the "reference section", in which each EMAS Kermit command is described in detail, in alphabetical order.

Note that the EMAS-3 operating system and the mainframe running it, are both referred to indiscriminately as "EMAS" in this User Note.

2: X-TALK VERSUS KERMIT

X-Talk has been used for many years in Edinburgh to transfer data between micros and the EMAS mainframes. It has some advantages over Kermit, but its disadvantages are now becoming significant. This chapter describes the important differences.

Transmission Errors

Kermit detects and corrects transmission errors during transmission, whereas X-Talk detects and reports transmission errors at the end of a file transfer, and you must retransmit the file yourself: this can make it virtually impossible to transfer medium to large files over a dial-up connection.

Availability

Kermit is available for a wide range of micros and hosts. X-Talk is available for only a limited range of machines, and there are no plans to make it more widely available. X-Talk is very parochial: it is used nowhere except in Edinburgh. Kermit on the other hand is very widely used in the UK and US academic communities, and its popularity is growing. If you use Kermit you will be compatible with a wide range of University mainframes, not just EMAS.

Maintainability

Kermit is much easier for us to maintain than X-Talk, for technical reasons. Any small change in the communications network, the EMAS front-ends, or the EMAS operating system, might mean that X-Talk would have to be changed. This could mean providing new versions for each micro attached to the network. Partly for this reason, X-Talk will become obsolete over the next 3 years or so.

Micro-to-micro communications

All micro Kermits can be used for micro-to-micro file transfers, while X-Talk, being an asymmetric protocol, requires one micro to pretend to be a host. The "host" end of the protocol is only implemented on some UCSD p-System micros.

Speed and cost

Kermit is generally much slower than X-Talk. Kermit also costs more to use (in terms of EMAS resources). Because Kermit is being updated and "tuned", check the help information (use the HELP command) for an up-to-date comparison between Kermit and X-Talk speeds and costs.

Ease of use versus amount of function

Kermit is not particularly easy to use. X-Talk, however, is more user-friendly. On the other hand, Kermit has more "functionality".

Communications path

X-Talk will only work if you are connected to a Computing Service-maintained PAD (because certain of the PAD settings must be set up correctly by the PAD Operator). It will not work through a "soft" PAD in a VAX, for instance. Kermit is very much less dependent on the nature of the communications path between the micro and EMAS, and it is sufficiently flexible to be able to change to accommodate some very "odd" communications pathways!

Which should I use?

Policy on which micro file transfer protocol to use is changing at the time at which this User Note is being written, so it is therefore not appropriate to give advice on this question here and now. From time to time, articles will appear in the *Computing Newsletter* detailing the position. However, Kermit will continue to be an integral factor in the micro file transfer scene for at least the next 5 years.

3: USING EMAS KERMIT

In this section we shall look at how you invoke EMAS Kermit, and how its command language operates. Output from the computer is shown `like this` while what you type is shown `like this`. If you must type a control character - say control P - it will be shown as `Ctrl+P`.

3.1 Starting and stopping EMAS KERMIT

To access the Kermit program on EMAS, enter the once-and-for-all command:

```
Command: SEARCHDIR MICROS:EMASLIB
```

Thereafter, to run Kermit, type:

```
Command: KERMIT
```

The KERMIT command may optionally be followed by any command which you can type in response to the `Kermit-EMAS>` prompt. This facility is normally used to start the program in server mode (see section 4.7):

```
Command: KERMIT SERVER
```

but it can also be used to execute any other Kermit command. Note, however, that a Kermit command line which would normally contain spaces must be enclosed in quotes:

```
Command: KERMIT "SEND MYFILE"
```

To leave EMAS Kermit, you would normally use the EXIT command, or its synonym QUIT.

3.2 EMAS Kermit command language

You control what you want EMAS Kermit to do, and how it should do it, by giving it commands in its "command language". The format of the command language closely follows that used on most other Kermit implementations on other machines.

3.2.1 Command format

When you enter Kermit, you will see a prompt on the screen

```
Kermit-EMAS>
```

This indicates that EMAS Kermit is expecting you to type a command. You can type either a Kermit command, or an EMAS command preceded by EMAS (e.g. EMAS FILES). A synonym for EMAS is !, which need not be followed by a space. Thus to see what files you have, you could type:

```
Kermit-EMAS>!FILES
```

Like EMAS commands, Kermit commands can be typed in upper case, lower case, or any mixture of the two as you please.

Kermit's own commands all take the form of a command name, such as SET, often followed by one or more further pieces of information or "parameters", which must be separated from each other by spaces. For example, one command you might use is:

```
Kermit-EMAS>SET FILE TYPE ASCII
```

to select text file transfer. Here the command is SET, while FILE, TYPE and ASCII are parameters to the command.

The reference section at the end of this guide gives the complete specification of all the EMAS Kermit commands.

3.2.2. The TAKE file facility

As an alternative to typing commands in on the keyboard, you can place them in a file known as a "TAKE file", and tell EMAS Kermit to read the commands from there instead of from the keyboard. This is done with the TAKE command; for example:

```
Kermit-EMAS>TAKE COMMANDFILE
```

EMAS Kermit will read the file line by line, and will obey each line in the file as a command. You can include either Kermit commands or EMAS commands in the file, the only exception being another TAKE command, which is not allowed. Additionally, you can place comment lines in the file to describe what it does: any line that starts with a semicolon (";") will be ignored by Kermit. Commands executed from a TAKE file are not displayed on the screen.

When the end of the TAKE file is reached, Kermit will revert to reading input from the keyboard.

If there are several commands which you wish EMAS Kermit to execute every time you start it, you can create a TAKE file called KERMINIT. When Kermit starts up, it looks for a file called KERMINIT in the current group, or failing that, at the top level (the "root") of your file index. If it finds KERMINIT, Kermit will automatically treat it as a TAKE file, and execute the commands which it contains. However, if you start Kermit with a parameter in the EMAS command line, as described in section 3.1, the KERMINIT file (if it exists) is ignored, and the command line parameter is executed as a Kermit command instead.

3.2.3. Context-sensitive help

When typing an EMAS Kermit command line, you may forget exactly what parameters the command you are using requires. Typing a question mark at any point in the line and then <return> will show you what the possibilities at that point in the line are. Thus for instance typing:

```
Kermit-EMAS>SET FILE ?
```

results in the following output:

```
TYPE   EXTENSION    PROTECTION
```

Note that there must be a space before the question mark, or you will get an error message. A little experimentation will show you how useful this facility is. You can even type ? at the Kermit-EMAS> prompt to find out what the commands are.

In order to save typing, you can use .. at the beginning of a command. This is shorthand for the last command which invoked context-sensitive help, up to the ?. Thus, having just issued the above command, you could say:

```
Kermit-EMAS>.. TYPE BINARY
```

to set the file type to binary.

More detailed help is available with the "HELP" command.

3.2.4. Abbreviating commands

It is possible to abbreviate Kermit commands and parameters. The rule is that a shortened form of the command or parameter may be used so long as it is unambiguous - ie it can be distinguished from other similar commands. Thus, to set binary file transfer mode, you can type:

```
Kermit-EMAS>SET F T B
```

which is completely unambiguous. However, you could not type:

```
Kermit-EMAS>SE F T B
```

because SE could refer to the SEND command or the SET command.

4: TRANSFERRING FILES WITH KERMIT

The primary use of EMAS Kermit is to transfer files between EMAS and a microcomputer. The methods used will be substantially the same whatever the microcomputer system is, since any Kermit system should be able to communicate with any other. Though the general techniques will be the same, the exact

commands used to control the micro Kermit will vary from one system to another. You will need to consult the user guide for the micro Kermit to discover how it should be controlled. In this section we shall cover in detail how to use EMAS Kermit to transfer files.

4.1 Principles

Transferring files with Kermit involves several discrete steps.

1. The micro Kermit program is started and set up for the transfer. In particular, you may wish to tell the micro Kermit what types of file are to be moved. You may also need to set any parameters for terminal emulation.

2. Terminal emulation mode is entered (usually by typing CONNECT at the micro), and you log in to EMAS as though you were using an ordinary terminal.

3. EMAS Kermit is started.

4. Commands can then be given to EMAS Kermit (from terminal emulation mode) and to the local Kermit (from local Kermit command mode). The two Kermit systems will communicate with each other using the standard Kermit protocol.

5. After the transfers are done, EMAS Kermit is terminated, and you may log out if you wish.

In practice, the steps taken will range up and down this list as required. For example, EMAS Kermit parameters can be changed at any time, not only at the start, and if you are moving several types of file you will need to change them frequently. In the sections below we shall consider the various actions you will need to take: the order of doing them is up to you.

4.2 Setting file type

4.2.1. Binary files

These files contain data that is not primarily printable text, such as machine-code programs or word processor sources for your micro. When you transfer these files, you wish every byte in the file on the micro to appear unchanged in the file on EMAS, regardless of what it is.

You tell Kermit that you are handling binary files with the command:

```
Kermit-EMAS>SET FILE TYPE BINARY
```

which tells it not to change any data that it either sends or receives. Note here that you may need to issue a comparable command to the local Kermit, to prevent it trying to manipulate the data. Some Kermits may not allow you to send pure binary data.

4.2.2. Printable text (ASCII) files

These files contain printable text. When you transfer one of these files, you do not necessarily want a byte-for-byte transfer, since the two machines may differ in how they store text files. The Kermit protocol defines the way in which things such as end-of-line are transferred: for instance if you are transferring a file from EMAS to your micro, EMAS Kermit will translate your data to this standard format, and the micro Kermit will then translate the standard format into whatever its own specific

requirements are.

You tell EMAS Kermit that ASCII text files are to be transferred with the command:

```
Kermit-EMAS>SET FILE TYPE ASCII
```

4.2.3 Choosing the right file type

It is important to note that "binary" files mentioned here do *not* include executable objects on EMAS. All files read by and written by EMAS Kermit are of type CHARACTER (on EMAS), even if they contain an executable image for running on another machine. A good rule of thumb to use is that if the file is text, and you wish to be able to look at it on EMAS, then use an ASCII transfer. Otherwise, use BINARY mode. The default when Kermit is started is ASCII mode.

4.2.4 Sending eight bit data

A further point to consider when transferring files concerns whether you can transfer all the 8 bits in every byte. It is common for communications systems to mainframes to restrict data to only 7 bits in each byte: thus you can only normally send characters whose ASCII codes are in the range 0 to 127. However, some text files and every binary file will contain bytes from the whole character set, with codes from 0 to 255.

Kermit in general has a technique for overcoming this restriction, by encoding characters in the range 128 to 255 into special sequences that can be sent down any communications line. EMAS Kermit will use this technique automatically when appropriate. Almost all modern Kermits will use this technique, which is known as "eight bit prefixing", but you may occasionally encounter an older implementation that does not support it. In this case your 8-bit data will be garbled in transmission, and Kermit will report this fact to you (perhaps by a message saying "too many retries" or something similar). There is, regrettably, no way round this problem from within Kermit.

4.3 Simple File Transfer

(See also section 4.7 for an alternative method which can often be used.)

4.3.1. Sending files to EMAS

To send a file to EMAS you use the command SEND on the local machine. You must also tell EMAS Kermit that a file is on its way.

a. In terminal mode, start EMAS Kermit, and issue the RECEIVE command. This tells it to expect a file from the local system. EMAS Kermit will then wait for something to happen.

b. Type the control sequence to return to the micro Kermit command mode. (See micro Kermit User Guide for details.)

c. Issue the micro Kermit SEND command.

Example:

```
Kermit-Micro> connect

[Connecting to host.  Type Ctrl+] C to return to micro]

Command: kermit

EMAS Kermit version 4.0

Kermit-EMAS> receive doc

Ctrl+] C

[Back at Micro]

Kermit-Micro> send info.doc
```

Transfer then proceeds, and the micro will normally show a display of how much data has been transferred. When transfer is complete, you may go back into terminal mode and look at your file on EMAS.

4.3.2 Sending files from EMAS

Transferring files to a local machine from EMAS is the exact reverse of the above RECEIVE procedure: all you need to do is reverse the roles of the two machines.

a. In terminal mode, start the EMAS Kermit program, and issue a SEND command. This tells it to transfer a file to the local system. There will normally be a delay before anything happens - the interval is 10 seconds (by default) and is intended to let you do the next step before the transfer starts.

b. Type the control sequence to return to the micro.

c. Issue the RECEIVE command to the local Kermit. When EMAS Kermit's delay time expires, it will start to send the file. The RECEIVE command tells the local Kermit to sit and wait until this happens.

Example:

```
Kermit-Micro> connect

[Connecting to host.  Type Ctrl+] C to return to micro]

Command: kermit

EMAS Kermit version 4.0

Kermit-EMAS> send doc

Ctrl+] C

[Back at Micro]

Kermit-Micro> receive info.doc
```

When transfer is complete, you may issue a CONNECT command locally to return to terminal emulation mode.

4.4 Handling Problems

By design, Kermit is a highly reliable file transfer system. The error-detection capabilities of Kermit ensure that data is transferred correctly. That said, there are some cases where you may need to abort a transfer.

The simplest way out of possible problems is for you to keep an eye on the progress of the transfer and see when it appears to be in trouble. The local Kermit will probably have some sort of display facility to show you how the transfer is going. If things seem to have ground to a halt, and the local Kermit is not able to do anything, you can abort the transfer at the EMAS end by re-entering terminal emulation mode, and typing Ctrl+P B to generate an Int: prompt, then typing ABORT followed by return. This will not be echoed on the screen, but should return you to the Kermit prompt in a few seconds. If you type just A in response to the Int: prompt, you will be returned to the EMAS Command: prompt.

If you cannot get any response at all from the mainframe, type Ctrl+P A to get a PAD> prompt, then type CLR to clear the call. You should then be able to log on again.

4.5 Protocol control

The rules by which files are transferred between Kermit systems are termed the "Kermit protocol". These rules define in detail how data should be transferred: they specify how much can be sent in one chunk or packet, what control sequences indicate the start and end of a packet, what character encoding is to be used, and so on. In almost every case you will have no need to change any of these settings, since they are carefully chosen so that any Kermit can communicate with any other Kermit in just about every circumstance.

However, it is possible that you may come across cases where you need to change some of the protocol values, either to improve the performance of the file transfer mechanism, or because the standard settings are inappropriate and do not work.

The protocol values are changed by the SET command, and EMAS Kermit allows you to change all the possible values. The reference section details all the SET commands and their effects. A detailed discussion of the various possibilities is beyond the scope of this user guide, since some understanding of the Kermit protocol is needed. You will find this explained in the "Kermit Protocol Manual".

4.6 EMAS file names

When sending a file from the micro to EMAS, you need to be aware of how EMAS Kermit handles file names. Many microcomputers use a dot to separate the file name from the file type, whereas EMAS uses a dot to separate group names in a file specification.

Suppose you issue the following command at the micro:

```
Kermit-Micro> SEND MYFILE.TXT
```

EMAS Kermit will remove the dot and store the file as MYFILETXT. If it were not to do this, the EMAS Subsystem would attempt to store the file in the group MYFILE under the name TXT, which is almost certainly not what you want.

What happens when you *do* want to transfer a file into a group other than the current one? Most micro Kermits provide a way of specifying a different name for

the host to store the file under. Typically, this is done by supplying a second parameter in the SEND command. If you wish to transfer the file MYFILE.TXT on the micro to EMAS, and call it DOCS.JUNE.MYFILE, you should give the command:

```
Kermit-Micro> SEND MYFILE.TXT DOCS/JUNE/MYFILE
```

Note that the group separator used is a slash rather than a dot, because EMAS Kermit removes dots from the received filename, as described above. A backslash can be used instead.

These considerations do not apply to receiving a file from EMAS, because EMAS Kermit always constructs a suitable filename which the micro Kermit translates into a form acceptable to the micro's operating system.

4.7 Server Mode

Server mode is a useful extra facility provided by EMAS Kermit. It allows many operations to be performed at the microcomputer end, rather than having to interact with the EMAS Kermit command line interpreter. To start server mode, enter the SERVER command:

```
Kermit-EMAS>SERVER
```

A message is displayed inviting you to type your local escape sequence to return to your micro Kermit command prompt. For instance, on an IBM PC you would type Ctrl+] C.

You can now send and receive files, and perform certain other operations on EMAS, by issuing commands to the micro Kermit. The commands below are those supported by EMAS Kermit - note that they should be typed in response to the *micro* Kermit prompt when EMAS Kermit is in server mode, and *not* in response to the EMAS Kermit prompt. Furthermore, not all micro Kermit programs support the use of all these commands.

GET <filename>

Initiate a file transfer from EMAS to the micro. Note that the RECEIVE command should not be used when the remote system (EMAS) is in server mode.

SEND <filename>

Initiate a file transfer from the Micro to EMAS. Just the same as the SEND command when EMAS Kermit is not in server mode.

REMOTE <command>

Perform some operation on EMAS. The possible <command> options are described below. They are more or less self-explanatory. Square brackets round a parameter indicate that it is optional.

COPY <filename1> <filename2>

Copy <filename1> to <filename2>

RENAME <filename1> <filename2>

Rename <filename1> as <filename2>.

DIR [<params>]

List files on EMAS. <params> is as for the FILES command.

DEL <filename>

Delete the specified file.

CWD [<groupname>]

Change working group (like the EMAS USEGROUP command).

TYPE <filename>

List the specified file.

KERMIT <command>

Here, <command> can be any valid EMAS Kermit SET or SHOW command. This facility allows you to change EMAS Kermit parameters without having to leave server mode. You can even use the context-sensitive help facility! Note however that many micro Kermits do not allow you to issue this command.

FINISH

Terminate the Kermit program on EMAS and return to the Command: prompt.

BYE

Log off EMAS altogether.

Note that these last two commands are the only way (apart from Int:A or Int:ABORT) to leave server mode.

5: COMMAND REFERENCE SECTION

In the descriptions of commands, {curly brackets} mean "repeat zero or more times". [Square brackets] mean "optional". <Angle brackets> simply indicate a parameter which would be replaced by something meaningful in actual use. The <>, {} and [] brackets are not intended to be typed at the keyboard.

References to a "wild card" filename means that the filename may contain the asterisk character "*". See the EMAS-3: User's Guide, Chapter 4, for an explanation of wild carding, but note that EMAS Kermit only supports the asterisk wild card and not the ampersand.

EMAS <emas command>

executes <emas command> as if typed at command level. A synonym is ! but note that there must be a space between EMAS and the <emas command>, whereas there need not be a space between the ! and the <emas command>.

EXIT

is the same as QUIT. It terminates the current Kermit session.

HELP

displays a file of helpful information using the VIEW system. Type QUIT to the View: prompt to return to Kermit.

QUIT

is the same as EXIT. It terminates the current Kermit session.

RECEIVE [<filename> { ,<filename> }]

This command tells EMAS Kermit to prepare to receive a file or files from the micro. Issue this command, then escape back to the local machine and tell it to start sending a file or files. If you do not specify a filename in the RECEIVE command, the EMAS files will inherit the same names as on the local machine (but with any "dots" separating the file name and the file extension removed). If you give a filename which already exists, Kermit will rename the file to a unique name, e.g. if you type RECEIVE MYFILE, and MYFILE already exists, the file will be received into MYFIL£. (But see the section on SET FILE PROTECTION.)

More than one filename may be specified in the RECEIVE command, and files received from the micro will be matched up against the specified filenames until the end of the file transfer. Filenames may be separated either by a comma or by a space.

SEND <filename> { ,<filename> }

This command will send a file from EMAS to the microcomputer. Having given this command, escape back to the micro Kermit and initiate the receive end of the transfer there. EMAS Kermit will wait for DELAY seconds before starting to transmit, to give you time to do this. See the section on SET DELAY. Note that <filename> may be wild, so specifying (for instance):

```
Kermit-EMAS> SEND FRED*,JIM*
```

will send all files beginning with FRED and JIM to the micro.

Filenames may be separated either by a comma or by a space.

SERVER

This command puts Kermit into server mode. You should then type your local micro escape sequence to return to your local Kermit, and transfer files using the local GET or SEND commands. You can also use the REMOTE command to interact with EMAS Kermit. The way to leave server mode is to type FINISH or BYE at your local machine. The former will terminate the EMAS Kermit program, but will leave you logged on, while the latter will log you off EMAS altogether.

Note that you can transfer multiple files from EMAS in server mode by specifying a wild card filename in your GET command to the micro Kermit.

SET

The SET command lets you change the way in which EMAS Kermit behaves. SET accepts many parameters, as described below. You will probably never need to use most of these commands, since the default settings are usually adequate. Perhaps the main exception to this generalization is the SET FILE TYPE command - see section 4.2.

SET BINARY-QUOTE <number>

sets the character used as a prefix to characters with the 8th bit set. The binary quote must be in the ranges 33-62 or 96-126

(ASCII), and distinct from both the send and receive quotes (used for quoting control characters), and also the repeat quote (used in repeat count processing). By default, it has a value of 38 (the ampersand &).

SET BLOCK-CHECK-TYPE <number>

This command selects the type of checking used for each Kermit packet. The value can be 1 (the default), 2 or 3. Higher numbers give more protection against errors. Block check type 1 is quite adequate for most situations. Note that the micro Kermit must also be told to use the same block check type, if type 2 or 3 is used.

SET CHERISH ON
SET CHERISH OFF

If CHERISH is ON, then a file received by Kermit will automatically be cherished when the transfer is complete. If CHERISH is OFF, it will not necessarily be cherished automatically - you should cherish it yourself if you wish. The default is ON.

SET DEBUG-FILE <filename>

Specify <filename> as the output file for storage of debugging information. Debugging information is not actually produced until the SET DEBUGGING ON command is issued. Debugging information is useful when trying to isolate a problem with Kermit, but is not required for normal use.

SET DEBUGGING ON
SET DEBUGGING OFF

These commands enable and disable the output of debugging information to the debug file. The default is OFF. Debugging cannot be turned on until a SET DEBUG-FILE command has been issued.

SET DELAY <seconds>

changes the length of time that EMAS Kermit waits before sending its first packet, giving you time to escape back to the micro, and do SEND there. The default is 10 seconds. <seconds> is the time in seconds of the delay.

SET FILE

There are various options starting with SET FILE, as described below.

SET FILE EXTENSION <extension>

tells EMAS Kermit to use <extension> as the final characters of a filename when sending it to a micro Kermit. For instance, if EMAS Kermit send an EMAS file called MYFILE, and the file extension had been set to TXT, then the file would be named as MYFILE.TXT to the micro. (You can of course get the micro Kermit to store the file under some other name if you wish.) The default file extension is EMS.

SET FILE PROTECTION ON
SET FILE PROTECTION OFF

If FILE PROTECTION is ON, EMAS Kermit generates a unique filename to RECEIVE into in the case of any clashes. If it is OFF, an existing file of the same name will be overwritten when EMAS Kermit is receiving a file. The default is ON.

SET FILE TYPE ASCII
SET FILE TYPE BINARY

sets up the type of file to transfer. In BINARY mode, each byte is transmitted as it appears in the file, with no translation. In ASCII mode, the local EMAS newline representation (LF) is translated to and from the Kermit standard representation (CR LF). The default is ASCII.

SET PROMPT <prompt>

changes the value of the prompt string from Kermit-EMAS> to whatever you specify.

SET RECEIVE

There are various options starting SET RECEIVE, as described below.

SET RECEIVE PACKET-LENGTH <number>

changes the maximum receivable packet length to <number>. Reduce this value if your connection to EMAS is very noisy, and you are getting many retries. The default is 94, which is also the maximum.

SET RECEIVE RETRIES <number>

changes the number of times that Kermit will try to receive a particular packet before giving up. The default is 10. Note that if the micro Kermit has a lower limit, it will be the one that gives up first and reports an error.

SET RECEIVE START-OF-PACKET <number>

changes the control character which Kermit expects to start a packet, where <number> is the ASCII value of the control character. The default is 1 (ie the control-A character).

SET REPEAT-QUOTE <number>

sets the character to be used in repeat count processing. The repeat quote must be in the range 33-62 or 96-126 (ASCII), and distinct from the send and receive quotes (used for quoting control characters), and also the binary quote (used for quoting characters with the 8th bit set). The default value is 126 (the tilde character ~).

SET SEND

There are various options starting SET SEND, as described below.

SET SEND END-OF-LINE <number>

changes the character used by EMAS Kermit to terminate each packet. <number> is the ASCII value of the character. By default, it is 13 (carriage return).

SET SEND PACKET-LENGTH <number>

changes the maximum packet length which EMAS Kermit will send to <number>. This limit will be changed by the micro Kermit when the file transfer starts, so its value is irrelevant at present. However, a possible future extension to EMAS Kermit (implementing "long packets") will require this command, so it is left in the command set, even though it currently has no function.

SET SEND PADCHAR <number>

changes the value of the character used to pad each packet. <number> is the ASCII value of the character. The default is 0 (ie the NULL character).

SET SEND PADDING <number>

causes each packet to be preceded by <number> PADCHARs. The default is 0 (ie no padding).

SET SEND QUOTE <number>

changes the character that Kermit uses to precede a control character. The default is 35 (the hash character #).

SET SEND RETRIES <number>

changes the number of times that Kermit will try to send a particular packet before giving up. If the micro Kermit has a lower limit for the number of times it tries to receive a packet, it will be the micro Kermit which gives up first. The default is 10.

SET SEND START-OF-PACKET <number>

changes the control character which Kermit uses to start a packet, where <number> is the ASCII value of the control character. The default is 1 (ie the control-A character).

SET TIMEOUT <number>

When a file transfer starts, EMAS Kermit requests the micro Kermit to assume that if the micro receives no response from EMAS Kermit for a given period, it should try again. This command allows you to set that period. The default is 15 seconds, but if EMAS is heavily loaded, you may need to

increase this time somewhat.

SHOW [<parameters>]

SHOW takes the same parameters as SET, and is used to display the current value of that parameter. SHOW ALL shows all of them, as does SHOW without any parameter. See the SET command for more details.

TAKE <filename>

causes commands to be taken from <filename> instead of being typed in at the keyboard. Control is returned to the console when end-of-file is reached. Note that lines beginning with ";" will be ignored by Kermit, and that another TAKE command is not allowed within the file.

If a file called KERMINIT exists (either in the current group or in the "root" group), and provided the KERMIT command has been issued without any parameters, then an implicit TAKE KERMINIT command is executed when EMAS Kermit starts. This provides a simple way to change the default settings without having to type in several SET commands every time.

VERSION

This command displays the version number of EMAS Kermit.