

Kermit and Modem are utilities for transferring files between computers using ordinary RS232 and modem connections.

The file KERMIT.LCOM contains both the Kermit and Modem protocols. Once loaded, it provides a means of transferring files between a Xerox workstation and any other computer that supports either Kermit or Modem, and to which Lisp is able to open a Chat connection.

Of these two file transfer protocols, Kermit is preferred. Modem is much less flexible than Kermit, and cannot be used on RS232 connections requiring parity or flow control. Modem was developed primarily to support file transfers to and from microcomputers running the CP/M operating system. Modem implementations are available for Tops-20, VAX/Unix, and VAX/VMS. Kermit, on the other hand, was designed for file transfers between computers of many types, and there exist implementations of the Kermit protocol on machines ranging in size from eight-bit microcomputers to large IBM mainframes.

For a detailed discussion and tutorial on Kermit, see *Kermit: A File Transfer Protocol* by Frank Da Cruz, Digital Press, 1987.

Requirements

The machine must run Kermit or Modem, and you need the means of reaching it, typically via Chat over an RS232 or a network connection.

You also need the following .LCOM files in order to run this module successfully:

KERMIT, KERMITMENU

as well as CHAT,

and either the RS232C or TCP-IP protocols, or the built-in NS or PUP protocols.

Installation

Load KERMIT.LCOM and the required .LCOM modules from the library.

Establishing a Connection

The first step in using Kermit or Modem is to establish a Chat connection with a desired host. You may use any sort of Chat

connection (e.g., NS, TCP, PUP, or RS232). See the Chat module in this manual.

If you are using an RS232 connection, and plan to transfer files with the Modem protocol, do not establish a connection that requires parity to be used; establish the connection with eight bits per character and no parity (see the RS232 module in this manual). Disable flow control (XOn/XOff) when using Modem.

When you have established a Chat connection to a remote host, log in (if necessary) and start the remote host's Kermit or Modem program. The details of running these programs differ slightly between implementations; you should obtain documentation specific to the version of Kermit or Modem running on the remote host.

Kermit

Remote Kermit in Server Mode

Most mainframe implementations of Kermit have a server mode. This mode causes the remote Kermit to listen for either send or receive requests without your having to type additional commands to the remote Kermit. If the version of Kermit you are using on the remote host does support server mode, give the server mode command to place the program in this mode. In most implementations of Kermit, server mode is entered by your typing **SERVER** to the Kermit prompt:

```
Kermit>SERVER
```

Remote Kermit Not in Server Mode

If the remote Kermit does not support server mode, you must issue individual send and receive requests for each file you transfer. To send a file to a remote Kermit, issue the **RECEIVE** command to the remote Kermit. To receive a file from a remote Kermit, issue the **SEND** command to the remote Kermit. In most cases, these commands are followed by the name of the file to be sent or received.

For example:

```
Kermit> RECEIVE FILENAME
```

or

```
Kermit> SEND FILENAME
```

If you are transferring files between two Xerox workstations connected by an RS232 connection, call (**CHAT 'RS232**) on each machine to establish the connection. Currently, Lisp Kermit does not support a server mode, so you must issue a receive request on one machine, followed by a send request on the other (see below).

After you have started the remote Kermit program, you need to start the local Lisp Kermit program. Lisp provides both functional and interactive interfaces for Kermit (and Modem).

Local Kermit

To start the local side of the Kermit file transfer, use the KERMIT.SEND or KERMIT.RECEIVE functions:

(KERMIT.SEND *LOCALFILE REMOTEFILE WINDOW TYPE*) [Function]

LOCALFILE is the name of the file being sent to the remote Kermit.

REMOTEFILE is the name under which the file should be stored remotely. In most implementations of Kermit, this name overrides any name you specified in the remote receive command.

WINDOW is a pointer to the Chat window over which the transfer will take place. If *WINDOW* is NIL, the value of CHATWINDOW (the first Chat window to be opened) will be used in its place.

TYPE is the type of the file. It should be set to either TEXT or BINARY.

(KERMIT.RECEIVE *REMOTEFILE LOCALFILE WINDOW TYPE*) [Function]

LOCALFILE is the local name of the file to be received from the remote Kermit.

REMOTEFILE is the name of the file on the remote machine.

WINDOW is a pointer to the Chat window over which the transfer will take place. If *WINDOW* is NIL, the value of CHATWINDOW (the first Chat window to be opened) will be used in its place.

TYPE is the type of the file. It should be set to either TEXT or BINARY.

While the file transfer is in progress, the associated Chat window will be blank, and cumulative packet counts and other messages will be displayed in a one-line prompt window above the Chat window.

Modem

To transfer files with the Modem protocol, you must run the Modem program on the remote machine. Modem does not support a server mode. Typically, you run the program once per file transferred, with instructions in the command line to indicate whether the file is being sent or received. There are a number of versions of the Modem protocol. On some systems, you run the program called Modem; on other systems, the program is called UModem or XModem.

On Unix, for instance, to send a text file to a Xerox workstation, you would type:

`%XMODEM -ST FILENAME`

On Tops-20, you would type:

`@MODEM SA FILENAME`

Note: % and @ are host system prompts.

As with Kermit, after you have started the remote side of the file transfer, you must start the local (Lisp) side. To do this, use either of the functions `MODEM.SEND` or `MODEM.RECEIVE`:

`(MODEM.SEND LOCALFILE WINDOW TYPE EOLCONVENTION)` [Function]

LOCALFILE is the name of the file to send to the remote Modem program.

WINDOW is the Chat window over which the transfer will take place.

TYPE is the file type, either TEXT or BINARY.

EOLCONVENTION is the end-of-line convention used by the operating system on which the remote Modem program is running. *EOLCONVENTION* should be one of CR, LF, or CRLF. Typically, Unix and VMS require LF, Tops-20 requires CRLF, and other Xerox machines require CR.

`(MODEM.RECEIVE LOCALFILE WINDOW TYPE EOLCONVENTION)` [Function]

LOCALFILE is the name of the file to receive from the remote Modem program.

WINDOW is the Chat window over which the transfer will take place.

TYPE is the file type, either TEXT or BINARY.

EOLCONVENTION is the end-of-line convention used by the operating system on which the remote Modem program is running (see above).

Interactive File Transfers With Kermit or Modem

A more convenient user interface for Kermit and Modem is available via the module `KERMITMENU.LCOM`. It provides a menu-oriented interface for issuing Kermit or Modem commands. To obtain the menu interface, press the middle mouse button in a live Chat window. The standard middle-button Chat menu will contain an entry labeled "Kermit" near its top. If you select this entry, a Kermit menu will appear at the top of the associated Chat window:

```

Kermit/Modem Settings
Send! Receive! Bye! Exit!
Transfer mode: Kermit Modem
Local file: {Dsk}<lispfiles>file.txt
Remote file: file.txt
File type: Text      End-of-line Convention: CRLF

```

The entries on the top line of the menu are action commands:

- SEND Starts sending a file to the remote Kermit or Modem program. The remote program must be prepared to receive the file.
- RECEIVE Starts receiving a file from the remote Kermit or Modem program. The remote program must already be attempting to send the file.
- BYE Closes (severs) the connection.
- EXIT Closes the window containing the menu, but does not close the connection.
- TRANSFER MODE This entry controls whether files are transferred using Kermit or Modem. You may set the state of this entry by selecting either of the Kermit or Modem labels with the mouse. The current transfer mode choice is displayed inverted in the menu.
- LOCAL FILE This entry holds the name of the local file being sent or received. You may set the contents of this field by selecting the LOCAL FILE label and typing the name.
- REMOTE FILE This entry holds the name of the remote file being stored or retrieved. You may set the contents of this field by selecting the REMOTE FILE label and typing the name. The Modem protocol does not use the contents of this field.
- FILE TYPE This field controls whether files are sent in binary or text (ASCII) mode. To set this field, select the FILE TYPE label and choose an entry from the menu that appears.
- END-OF-LINE CONVENTION This field sets the end-of-line convention being used by the remote Modem program (it is not used when files are transferred in Binary mode or with the Kermit protocol). The contents of this field must match the conventions of the operating system on which the remote Modem program is running. To set this field, select the END-OF-LINE CONVENTION label, and choose an entry from the menu that appears.

Limitations

Transfer files between two Xerox machines using the Kermit protocol.

Modem cannot be used on RS232 connections requiring parity or flow control.