System B.1 Stream package
J. P. Strait
4 December, 1980
Copyright (C) Three Rivers Computer Corporation, 1980

Several improvements were made to the stream package in the Perq
operating system release B.1. The stream package is the implementor
of the Pascal Reset, Rewrite, Get, Put, Eof, Eoln, Read, Write,
Readln, and Writeln routines. The following improvements were made.

1. Control character processing was improved so that most control
   characters may be read from a text file. In previous systems
   control characters were deleted when a text file was read. Some
   control characters are deleted under some circumstances:

   - When reading Console: certain characters have special meanings
     and are not passed through to text files.

        ^c   aborts programs--one ^c if a program is waiting for
              input and two if it is not.

        ^C   (control-shift-C) aborts a program at any time and
              prints a stack traceback.

        ^D   (control-shift-D) prints a stack traceback without
              aborting.

        ^h   (back-space) deletes a character of input.

        ^j   (line-feed) is ignored.

        ^m   (carriage-return) is the end of line character.

        ^q   resumes output if it was suspended with ^s.

        ^s   suspends output.

        ^u   (oops) deletes a line of input.

        ^z   is the end of file character.

   - When reading disk files, ^j (line-feed) is the end of line
     character, and ^m (carriage-return) is ignored. When writing
     disk files, the end of line is represented as a carriage-return
     followed by a line-feed.

- When reading RS:, no characters have special meanings and none are ignored. This means that there is no end of line when reading RS:. When writing RS:, Writeln generates a carriage-return followed by a line-feed.

- When reading RSX: (see below), ^j (line-feed) is the end of line character, and ^m (carriage-return) is ignored. When writing RSX:, Writeln generates a carriage-return (no line-feed).

2. Reset and Rewrite now allow filenames of the form

        <device name>:<anything>.

   This allows dumb programs to unconditionally concatenate extensions onto filenames typed by the user (e.g. Console:.Map, RSX:.List).

3. A new device called RSX: has been added to allow reading and writing files on an RSX-11M system. This device uses the RS-232 port and a very simple protocol to read and write RSX-11M files through its PIP utility. The RSX: device is optional, and can be deleted from the stream package to save code space. See the CompileRSX constant in Stream.Pas.

4. When an output file is opened to RSX:Filename, the stream package sends

        ^qPip FileName=TI:^m

   down the RS-232 wire. Then for each character the stream package follows these steps.

        a) Flush RS232 input buffer.

        b) Send RS232 character.

        c) Wait for RS232 echo--in the case of sending a carriage return or ^z, wait for a line feed.

   An ^z is sent when the file is closed.

5. When an input file is opened to RSX:Filename, the stream package sends

    ```
    ^qPip TI:=FileName^m
    ^s
    ```

   down the RS-232 wire. Then for each 32 character buffer the stream package follows these steps.

   a) Send a ^q.

   b) Read from RS232 until end of line or buffer full.

   c) Send a ^s.

   The end of the file is recognized by a '>' after an end of line. The ^s is not sent after the end of file is received.