UNIVERSITY OF EDINBURGH

Edinburgh
Regional
Computing
Centre

SCREED: A Screen Editor

SCREED is a screen editor, available on EMAS. It can support several different types of video terminal.

by
John M. Murison

First Edition
June 1981

Edinburgh Regional Computing Centre

SCREED: A Screen Editor

SCREED is a screen editor, available on EMAS.
It can support several different types of video terminal.

by John M. Murison

Contents

1. Introduction to SCREED

SCREED is a screen editor available on EMAS at ERCC. Its purpose is to provide a simple means of editing character files. It does this by exploiting the fact that a video screen has 24 lines (usually) and that it is possible to move the "cursor" (current position marker) to any position on any of the lines. When characters are typed thereafter they appear on the screen at the cursor position.

Now, if a video screen were filled with some lines from a character file, the effect of moving the cursor to some position on the screen and then typing would be to overwrite the text at that position. If a text editor running on EMAS had caused the file lines to appear on the screen in the first place, then this editor could note the user's subsequent movement of the cursor, and when the user overwrote some of the text on the screen it could make an exactly similar change to the original file.

The main advantage of this approach to editing over that used by, say, EDIT or ECCE is that instead of typing a "command string" to cause the editor to change the file, the user is "changing it himself", or so it appears, and the effect on the file is immediately obvious.

This summary of the principle of screen editing leaves a number of questions unanswered. For example:

- How do I tell the editor which part of my file is to be displayed on the screen?

- What happens if the text to be added at some point in the file is smaller (or larger) than the text to be removed?

- The cursor control keys on my terminal (the "arrow keys") do not work with EMAS.

To take the last point first: the communications network rejects most of the "control" characters if you type them on your keyboard. However, the network has been modified to allow a new mode of operation in which all characters are acceptable.

The other points are covered in the following sections.

1.1 Changing text

To replace text by new text, you move the cursor to the start of the text to be replaced and overtype it with the new text. Of course, this is only satisfactory if the new text is exactly the same size as the existing text.

To delete text without replacing it by new text, a delete character is provided. If you overtype an existing character with the delete character the effect is to delete the existing character from the file without replacing it by anything else. The delete character is '^' by default, but it can be changed (see Section 4.10).

To add text to the file you need to use "two-letter commands", described in Section 3.

1.2 Starting

Before using SCREED for the first time, you must give the following command:

```
                Command:INSERT(KNTLIB.SCREED7)
```

To invoke SCREED itself:

```
                Command:SCREED(input file, output file)
```

If the input file is to be edited back into the same file, you do not need to specify an output file - in this case the preceding comma is omitted also. If you are creating a new file, the form of the call should be

```
                Command:SCREED( , output file)
```

IF YOU ARE TRYING OUT SCREED FOR THE FIRST TIME:

* Choose as input an unwanted file or a copy of a file.

* BEFORE STARTING, read the next section on stopping (1.3). SCREED is different in this respect from any other EMAS program, and - for example - Int:A will NOT bring you back to EMAS command level.

Once you have invoked SCREED you may be asked what type of video you are using. (SCREED calls the command TERMINAL TYPE if your terminal type is not known by the Subsystem). When you have replied, you will be asked if this information is to be saved, to avoid having to ask you in future. Thereafter, if SCREED is able to work with your terminal, the screen is blanked and the first two lines of the input file appear at the top of the screen (or nothing appears if the file is new).

Holding the 'Control' button down, type 'R' followed by 'S' (or, in the notation used in this description, "type (R)(S)"). The first screenful of the input file will appear on the screen. You will find that the cursor moving buttons will operate - Line Feed, Return, Up Arrow, Down Arrow, Backspace, Home, etc.

1.3 Stopping

TO TERMINATE AN EDIT SESSION, TYPE

```
     STOP   followed immediately by Ctl X (Ctl Z on Newbury 7000
  or                                              series terminals)
     ABORT  followed immediately by Ctl X (Ctl Z on Newbury 7000
                                                  series terminals)
```

(Details of this type of edit command are given in Section 4.)

1.4 Points to note

* Long lines: when a file line is longer than a screen line, it is "wrapped round" onto the next screen line. The line is broken after a word, if possible, and then following the break the symbol '>' is printed (to indicate that the file line is continued). Of course, this symbol does not form part of the file - it is merely an indication on the screen. If you wish, you can overtype it.

* If your terminal is one of those which automatically moves the cursor to the next line when you type past the end of the screen line, then doing this causes a single long line to be created in your file, 'wrapped' onto two screen lines (but not indicated by '>'). If you want to terminate a line which you are typing in, then you must move off the line explicitly and not let the terminal do it for you.

* End-of-line: the behaviour of the cursor when text is typed at the end of a screen line depends upon the terminal in use. On some terminals it stays put on the last printing position; on some it moves to the right of the last printing position and then moves to the next line when another key is struck; and on some terminals the cursor is moved to the next line as soon as the last character position on a line has been typed into. This latter behaviour causes difficulties when scrolling and so such terminals are treated as though their lines were one position shorter than they actually are (typically 79 positions instead of 80). If your terminal is one of these (see Section 5), do not type into the final position on any line - the character typed will be ignored if you do.

* Page mode: as specified in Section 5, some terminals operate in page mode. This means that they cannot scroll the displayed text upwards while adding adding new lines of text at the bottom of the screen. The implications for individual two-letter commands are detailed in Section 3.
Note that where a terminal has a key marked 'Page', this must NOT be depressed during the operation of SCREED if the terminal operates in scroll mode.

* When SCREED is being used, there are no "command line" or "menu" areas on the screen - all that you see is file, apart from '^' characters and wrapped-line ('>') indicators. In addition, some blank lines on the screen may not correspond to blank lines in the file. Such screen lines are called "null".

* "Int:" character
  Normally an EMAS user can generate an interrupt ("Int:") by use of the ESC key. When SCREED is in use the ESC key does not have this effect. However, typing Ctl] (Ctl \ with Newbury 7000 series terminals) does, and it is then possible to abandon the edit in the usual way. PLEASE NOTE the following points: a) once the "Int:" prompt appears on the screen, all the editing is effectively lost, and you should force a return to command level by typing A or C; b) if you do return to command level by this means, you should then call SETMODE or log off immediately. From this it will be understood that you should only force an interrupt during a SCREED session as a last resort.

* Glasgow users
  Users of Glasgow University TCPs should note that, after leaving SCREED (either returning to Command level or branching to EDIT, ECCE or VIEW), they should reset the TCP by giving the following command:

  ```
  ESC I  space  Ctl P  Ctl A
  ```

  This causes Ctl A to be the 'Int:' character, which is the norm with the Glasgow TCPs.

* Finally, note that in this description of SCREED, the notation "(A)" stands for "Ctl A", or "the code produced by typing the letter 'A' while holding down the Control button at the same time".

2. Cursor-moving keys

To move the cursor to a specific part of the screen the user can make use of whatever keys are available on his keyboard: Tab, Backspace, Return, Line Feed, the "arrow" keys, Home - but not the space bar since it overwrites the current position on the screen with a space character.

With the exception of Return (Section 2.1), the terminal behaves as though it were in local mode. You should try out the effects of these keys in local mode or when using SCREED, as they differ for different terminals. For example: 1) Backspace (BS) typed when the cursor is at the start of a line causes the cursor to move to the end of the previous screen line with some terminals, while having no effect with others (while in the case of the Lynwood DAD-1, it causes the cursor to move down!); 2) The Tab key - (I) with some terminals - also has different effects with different terminals. It normally causes a move to the next tab stop, tab stops being 8 character positions apart (usually); but the effect may change towards the end of a screen line.

In general, it is valid to use whatever keys the terminal happens to have in order to move the cursor around.

Some terminals do not have "arrow" keys. A position on a line of the screen can be reached by moving past it by repeated use of the Tab key, followed by repeated use of Backspace. However, it will usually be found simpler to use the two-letter commands, described in Section 3, especially the (W) commands.

With ALL terminals, it is necessary to use the two-letter commands whenever you wish to see another part of the file, or to get space on the screen to insert text at some point.

2.1 Return

The effect of using the Return key depends on the context:

* If the cursor is not on the last screen line, it is moved to the next screen line. If there is text on this line, the cursor is positioned at the left hand margin. If this line has no text in it, however, the cursor is positioned at the current indentation level of the text. This is determined from the indentations of the previous three non-blank screen lines. Advice: try it!

* If the cursor is on the last screen line and the terminal operates in scroll mode, Return normally causes the screen to scroll and the next line of the file to appear. However, if the current line is (or was) a null line, then the screen scrolls and another null line appears. (This is so that when (A)(S) - described in Section 3.3 - has been used to add a lot of text to the file, using the Return key on the bottom line simply provides another null line to type into).

3. The two-letter commands

The user commands SCREED by means of pairs of control characters. The first indicates an action and the second the text unit to which the action applies. If the action control character is omitted the most recent action control character given is assumed.

Action		Text Unit
(A)dd		
(B)eginning of		
(C)onjoin		(W)ord
(D)isjoin	followed	
(E)nd of		(L)ine
(O)mit	by	
(N)ext		(S)creen
(P)revious		
(R)ewrite		
(T)runcate		

As explained earlier, when a capital letter is given in parenthesis in this description, for example '(A)', it represents the control character produced by typing the letter while holding down the Control key. In some documents this is expressed as Ctl A or Ctl+A.

For example, (R)(S) means Ctl R Ctl S - "Rewrite Screen" - and this is what happens when you type (R)(S). (N)(L) means "Next Line" - the cursor is moved to the next line. If you then type (L) on its own, the (N) would be assumed, since it was the last action control given.

Any combination of an action followed by a text unit may be given, but may not be valid in some circumstances. For example, (O)(W) (Omit Word) will fail if the cursor is not on a "word" (a piece of text delimited by spaces or the beginning of a line or the end of a line). The editor signals failure by wiggling the cursor back and forth on the current line. It is probably simpler to start by trying out the effects of these commands on a file than to read the detailed description of their effects (in Sections 3.1, 3.2, 3.3). In fact, it should be possible to predict what most of the commands will do - possible exceptions being (A)(W), (A)(L), (C)(S), (D)(S), and the (T) commands:

(A)(W) moves the rest of the current line to the right and inserts delete ('^') characters in the gap. These can then be overwritten with new text. Note that it is not necessary to 'tidy up' the line by use of (R)(L) thereafter - SCREED will automatically remove delete characters from a line when it moves off the screen.

(A)(L) opens up a vertical gap before the current line, by moving it down to the bottom of the screen. The gap is filled with null lines, which again may be typed into or left to be removed by SCREED in due course.

(C)(S) "conjoins" the current screenful of text to the next screenful. It then displays the "join" in the middle of the screen. The effect is thus to move the text up by half a screenful.

(D)(S) "disjoins" the current screenful of text, at the current line. The new screenful of text starts at the current line, i.e. the screen is scrolled or rewritten so that the current line is at the top.

(T) stands for "truncate": the text unit - (W), (L) or (S) - is thus truncated, i.e. deleted from the current position onwards. Thus (T)(W) removes the rest of the current word, (T)(L) removes the rest of the current line, (T)(S) deletes the rest of the current screen (including the current line).

The following sections describe the (W) commands, then the (L) commands, then the (S) commands.

Please note that some terminals operate in "page" mode (which means that they do not scroll the text). Where a two-letter command normally implies a scrolling operation, e.g. (D)(S), in the case of a page-mode terminal the screen is entirely rewritten. Differences in behaviour to accommodate page mode terminals are noted in the detailed descriptions following.

3.1 The (W)ord commands

* (A)(W) shifts the remainder of the current line about 15 places to the right and fills the gap with '^' characters. These can then be overtyped as required. If the end of the line was moved off screen in the process, it reappears on (R)(L), although you do not have to follow (A)(W) with (R)(L). (Indeed, there is no command which must be followed by some other specific command - they are all entirely independent.)

* (B)(W) moves the cursor to the beginning of the current word. The command fails if the cursor is not currently on a word.

* (C)(W) "conjoins" the word on which the cursor is positioned with the next word. It does this by replacing the intervening gap with '^' characters. The command fails if the cursor is not currently on a word, or is on the last word of a line.

* (D)(W) "disjoins" the current word at the cursor position, i.e. it breaks it into two words by inserting a space character just before the cursor position. The command fails if the cursor is not currently on a word.

* (E)(W) moves the cursor to the end of the current word, i.e. to the first position following the word. The command fails if the cursor is not currently on a word.

* (O)(W) causes the current word and appropriate space characters beside it to be replaced by '^' characters.

* (N)(W) moves the cursor to the next word on the screen. The command fails if the cursor is on or beyond the last word on the screen.

* (P)(W) moves the cursor to the previous word on the screen. The command fails if the cursor is on or before the first word on the screen.

* (R)(W) rewrites the current word. In fact, it rewrites the rest of the current line.

* (T)(W) truncates the current word. It replaces each character of the current word, from the cursor position to the end of the word, by '^' characters. The command fails if the cursor is not currently on a word.

3.2 <u>The (L)ine commands</u>

Note that (R)(L), (D)(L) and (C)(L) refer to file lines (which may occupy more than one screen line). All the other (L) commands refer to screen lines.

* (A)(L) causes the current line and all the lines below it to be replaced on the screen by null lines, the current line then to be rewritten at the bottom of the screen and finally the cursor to be moved back to the first of the null lines. This is to be interpreted as a gap having been opened up in front of the current line, into which text may be typed. The gap consists of null lines, which can then be used as necessary. Null lines below the last line added will be discarded on typing (R)(S), but note that null lines <u>traversed</u> prior to text being added lower down will be turned into blank lines (i.e. they become part of the file).

  [If you want to add blank lines to your file, use (A)(L) at the appropriate point, then traverse as many null lines as you want turned into blanks and type a '^' character. If you then use (R)(S) you will find that all the null lines traversed and the line on which the '^' was typed have been preserved as blank lines, while the remaining null lines have been discarded.]

  If your terminal is operated by SCREED in page mode (e.g. the Lynwood DAD-1) or if it has no "clear rest of line" function (e.g. the ADM-3A), then (A)(L) causes the screen to be blanked, the line previously containing the cursor to be written at the bottom of the screen and the line before that to be written at the top of the screen. The screen lines between are all null.

* (B)(L) moves the cursor to the beginning of the current line (i.e. to the left hand edge of the screen).

* (C)(L) "conjoins" the current line with the next line. The intervening newline character is removed and the two screen lines are rewritten. The command fails if the current line is already "wrapped" onto the next screen line, or is the last line on the screen.

* (D)(L) causes the current line to be broken ("disjoined") at the cursor. Normally the remainder of the screen has to be rewritten to accommodate the new line produced.

* (E)(L) moves the cursor to the end of the text on the current screen line.

* (O)(L) turns the current screen line into a null line and deletes ("omits") the corresponding text from the file.

* (N)(L) moves the cursor to the next screen line. If the cursor is already on the last screen line, it causes the screen to scroll and the next file line to appear. The command fails if the cursor is on the last screen line and the terminal operates in page mode.

* (P)(L) moves the cursor to the start of the previous screen line. The command fails if the cursor is on the top line of the screen.

* (R)(L) removes all '^' characters from the current screen line in the process of rewriting it.

* (T)(L) truncates the current line, i.e. deletes the text on the line from the cursor position onwards.

3.3 The (S)creen commands

* (A)(S) is similar to (A)(L) except that the current line is not rewritten at the bottom of the screen. Thus an arbitrarily large number of lines can be added simply by scrolling the screen when you get to the bottom. Use of a "Clear Screen" key, if your terminal has one, is equivalent to (A)(S) - the resulting screenful of null lines is understood to come before the current line.

  If your terminal is operated by SCREED in page mode or if it has no "clear rest of line" function, (A)(S) causes the screen to go blank, as though you had used a "Clear Screen" key (as described above).

* (B)(S) moves the cursor to the top left hand position on the screen.

* (C)(S) implies that this screenful and the next are to be "conjoined". The resulting screenful shows the join exactly in the middle of the screen; i.e. the effect is to scroll up by half a screen. With page mode terminals the screen is rewritten to display the appropriate text.

* (D)(S) means that the screen is to be "disjoined" at the current line, which becomes the first of the new screen; i.e. the screen scrolls until the current line is at the top. With page mode terminals the screen is rewritten with the current line at the top.

* (E)(S) moves the cursor to the bottom right hand edge on the screen.

* (O)(S) causes all the lines displayed on the screen to be removed ("omitted") from the screen and the file.

* (N)(S) clears the screen, then displays the next screenful of file.

* (P)(S) clears the screen, then displays the previous screenful of file. The command fails if the start of the file is already on the screen.

* (R)(S) rewrites the screen, removing all '^' characters and null lines. If there are null lines above the cursor when (R)(S) is typed then file lines prior to those currently displayed are brought down onto the screen to replace the null lines. If there are null lines below the cursor when (R)(S) is typed then file lines following those currently displayed are brought up onto the screen to replace the null lines. When there are insufficient previous file lines to bring down, the first line of the file appears at the top of the screen. If there are insufficient file lines to bring up onto the screen, null lines appear at the bottom of the screen.

* (T)(S) deletes, from the screen and the file, the current line and all screen lines below it.

4. The (X) commands

(Note in this description that '(X)' stands for 'Ctl X'.)

The two-letter commands are useful, but they are not sufficient. The following questions make this clear:

- * How do I terminate (or abandon) the edit session?
- * How do I get to the middle of a large file (typing (N)(S) fifty times is rather tedious)?
- * Can I save the editing done to date?
- * How can I correct twenty occurrences of the same mis-spelled word?
- * Can I change the delete character?

All these facilities are provided by means of another control character, (X) [(Z) on Newbury 7000 series terminals]. If you type in some text and follow it with (X), SCREED rewrites the line as it was before you typed the text, and then examines the text typed. If it is one of the following, in upper or lower case and ignoring spaces, then a special action is implied:

```
HELP
CLOSE (or STOP or END or EXIT)
ABORT
SAVE
EDIT
ECCE
text followed by '>' or '<', e.g. bananas<
REPORT=<integer>
MESSON or MESSOFF
DEL=<character>
```

If the text was none of these then SCREED takes no action.

If (X) is typed on its own, i.e. without preceding text, the effect is to move the cursor to the left hand end of the current line. If the cursor is already at the left hand end and (X) is typed on its own, then "thumbing" takes place (see Section 4.11).

4.1 HELP

```
          HELP (X)
```

This causes a branch from SCREED to the VIEW command, and you can then examine this description of SCREED. On exiting from VIEW by means of the command Q or E, the screen is rewritten as it was when you typed HELP, and the SCREED session resumes.

Note: '(X)' stands for 'Ctl X'

4.2 CLOSE/STOP/END/EXIT

```
          CLOSE (X)

          STOP (X)

          END (X)

          EXIT (X)
```

Any of these causes the edit session to be terminated.

Note: '(X)' stands for 'Ctl X'

4.3 ABORT

```
          ABORT (X)
```

The edit session is terminated without the output file being written to.

Note: '(X)' stands for 'Ctl X'

4.4 SAVE

SAVE (X)

The file is copied in its present state to the output file, the screen is blanked and then the current line and adjacent lines are written in the middle of the screen (the other screen lines are null).

Note: '(X)' stands for 'Ctl X'

4.5 EDIT

EDIT (X)

A branch is made to the Subsystem editor EDIT. The current file position is preserved. You can terminate the edit session from within EDIT in the usual way, or return to SCREED if you wish. This is done by hitting the ESC key and replying "SCREED" (or "screed") to the "Int:" prompt. On hitting the "Return" key the "Edit:" prompt will appear as usual, but if you hit "Return" again SCREED will be reverted to. The current line and adjacent lines will then be displayed, as for SAVE.

Note that you can start your editing in EDIT and then branch to SCREED by the use of the "Int:". To do this you must have invoked EDIT from command level by the command TEDIT.

Note: '(X)' stands for 'Ctl X'

4.6 ECCE

ECCE (X)

A branch is made to the editor ECCE. The current file position is preserved. You can terminate the edit session from within ECCE in the usual way, or return to SCREED if you wish. This is done by hitting the ESC key and replying "SCREED" (or "screed") to the "Int:" prompt. On hitting the "Return" key the ">" prompt will appear as usual, but if you hit "Return" again SCREED will be reverted to. The current line and adjacent lines will then be displayed, as for SAVE.

Note that you can start your editing in ECCE and then branch to SCREED by the use of the "Int:". To do this you must have invoked ECCE from command level by the command TECCE.

Note: '(X)' stands for 'Ctl X'

4.7 Text location

```
        text> (X)
     or
        text< (X)
```

If you type some text followed by '>' (followed by (X)), SCREED will search from the current position to the end of the file for the text. If the first occurrence of the text is on the screen, the cursor is moved to it; if it is off the screen, the screen goes blank and then the line with the occurrence is displayed in the middle of the screen with the adjacent lines. If there is no occurrence of the text you typed then the cursor is wiggled as for a fault, and nothing else happens - in particular, the cursor position is not moved.

If you type text followed by '<' (followed by (X)), SCREED will search for the text back from the current position to the start of the file.

If you type just '>' followed by (X), SCREED will search for an occurrence of the last search text typed in. (This saves typing if you found an occurrence of the text but not the one you wanted.)

The effect is similar if you type just '<' followed by (X) - SCREED searches backwards for an occurrence of last search text typed in.

On most terminals the '>' key also has '.' on it. To save using the shift key, '.' followed by (X) has the same effect as '>' followed by (X). Similarly ',' followed by (X) has the same effect as '<' followed by (X).

Note: '(X)' stands for 'Ctl X'

4.8 REPORT=

```
REPORT=n (X)
```

(n is integer, $n \geq 0$) In certain situations, the screen is blanked and then only two or three lines appear; e.g. on entering or re-entering SCREED, or after text location or "thumbing". The integer following REPORT= indicates how many lines before the current line and how many after the current line you wish to have appear on the screen in these situations. The default is 1 (i.e. 1 before and 1 after). Setting it to 23 or more (for most terminals) guarantees a whole screenful.

Note: '(X)' stands for 'Ctl X'

4.9 MESSON/MESSOFF

```
   MESSON (X)
or
   MESSOFF (X)
```

By default, TELL messages and Operator messages are not output during a SCREED session. This can cause you to be logged off without warning (because the System was closing down and you didn't get the Operator messages). Specifying MESSON causes the messages to appear. However, note that SCREED cannot tell when such a message has appeared on the screen, so you should rewrite the screen - (R)(S) - after getting a message.

Using MESSOFF causes TELL and Operator messages to be held until you exit from SCREED. This is the default.

Note: '(X)' stands for 'Ctl X'

4.10 DEL=

```
DEL=char (X)
```

The specified character becomes the delete character in future.

```
e.g. DEL=* (X)
```

Any existing occurrences in the file of the newly-selected delete character are noted by SCREED, and are NOT treated as delete characters.

Note: '(X)' stands for 'Ctl X'

4.11 Thumbing

You can get to an approximate point in your file by a technique known as "thumbing": ignoring the current contents of the screen, suppose that the top line of the screen corresponds to the first line of the file and that the bottom line of the screen corresponds to the last line of the file; then halfway down the screen ≡ halfway down the file, three quarters down the screen ≡ three quarters down the file, etc. If you position the cursor on the left hand margin of a screen line and then type (X), the effect is to move to the corresponding position of the file. The screen is blanked and three lines from the implied file position are written, at the specified screen position. If this is not in fact the right place then you can adjust your position, up or down, and type (X) again. This is known as "thumbing", as you are effectively thumbing through the file.

The reason for SCREED normally displaying only two or three file lines, on entry or following a (X) command, is to save time - you might want to "thumb" to some other part of the file, and writing a complete screenful of text would thus waste time. You can always use (R)(S) if you want to see a screenful; in this case, note the significance of the cursor position, as explained in Section 3.3. You can also set REPORT (see Section 4.8).

5. Notes on each terminal

This section contains notes on each of the terminals which SCREED can work with (at the time of writing). Contact the ERCC Advisory Service if you have a terminal which is not included below but which you think might be suitable for use with SCREED.

5.1 Perkin Elmer 550 Notes

- There are no "arrow" keys. Thus the cursor has to be moved by the use of Tab, Backspace, Return, Line Feed, and the appropriate SCREED commands (the (W) commands in particular).
- The Clear key (which only operates when Ctl is depressed also) does not transmit any code - the effect is entirely local to the terminal. Thus if you clear the screen by using it, SCREED is not aware of this. Therefore it should not be used.

5.2 Newbury 7000 Series Notes

- In the (X) commands (Section 4), (Z) is in fact used instead of (X).
- (\) is the "Int:" character rather than (]).
- (With respect to the 7001:) The keyboard has a separate row of keys along the top. Most of these are either disabled or do not generate any codes. However the keys from Home rightwards are usable.
- If the terminal does not seem to be operating correctly when used with SCREED, it might not be set up appropriately. This can checked, and corrected if necessary, as described below. Note that during this procedure the terminal is effectively off-line, and so does not interfere with any on-line work in progress.

  Press the Edit key (which is not normally used), and then Ctl and Home together. A screenful of text appears; it details various settings (which on most other terminals are determined by switches on the terminal). When you press Tab the cursor moves from one setting to the next. If you press Return repeatedly the various possible values of the current option appear in turn.

  The settings required to use SCREED are as detailed below; the other settings should not be changed:

First line:

```
    ........BELL=80;DELETE=N;NL=NO....        (BELL=80 turns off bell)

                      TELETYPE = NO

CLEAR FROM HOME
    :
    :
```

Under the heading ATTRIBUTES:

```
    PROTECT = NO
    BLINK = NO
    UNDERLINE = NO
```

Of the codes given under the heading FROM LINE, the following must be underlined:

```
            0C 1A
```

and the following must not be underlined:

```
            08 09 0A 0B 0D 16 18 19 1D 1F
```

The settings of the other FROM LINE codes are immaterial to the operation of SCREED.

You should then finish by pressing the ESC key followed by the Edit key. This puts you back on-line to EMAS.

5.3 ADM-3A Notes

* In the two-letter commands, (_) is used instead of (L).

* Although the terminal has a line width of 80 characters, SCREED (for good reasons) does not use the 80th column of any line, and will ignore any character typed in the 80th column.

* There is a (standard) option on the ADM-3A whereby typing the space bar does not always generate a space character - it merely moves the cursor right. This is not appropriate to the operation of SCREED and should therefore be changed. The person responsible for the terminal should be consulted, as the option switch in question is under the terminal casing.

  The instructions for making the change are given on page 2-3 of the terminal handbook. The switch in question is located at the top of the left-hand set of toggle switches; it is labelled "SPACE 6 / ADV"; it should be set to SPACE (by default it is set to ADV).

5.4 Lynwood DAD-1 Notes

* Note that in the two-letter commands (F) (Following) is used instead of (N) (Next), and that (Q) is used instead of (L) (Line).

* SCREED operates the Lynwood in page mode, which it selects automatically. This affects the operation of (C)(S), (D)(S) and (A)(Q), as explained in Section 3.

* The Line Feed key has been disabled, as it has the effect of clearing any line that it causes the cursor to move onto.

* The two switches at the back of the terminal marked St.CR and St.LF should both be down (i.e. off).

* Sometimes on entry to SCREED, the first two lines of the file appear at the bottom of the screen rather than the top. This has to do with the terminal warming up (I think). SCREED will operate as though the two lines - and the cursor - were at the top of the screen. It is therefore best in this situation either to "thumb" or to use (R)(S) or (F)(S).

* Although the terminal has a line width of 80 characters, SCREED (for good reasons) does not use the 80th column of any line, and will ignore any character typed in the 80th column.

5.5 Visual 200 Notes

* The Visual 200 terminal has a large number of features and settings; each is activated by the user generating - either explicitly or by use of a function key - a string of characters starting with ESC. SCREED can detect when this has been done but unless the feature happens to be one that SCREED uses (such as "Clear screen" or "Clear rest of line"), it will generate an error message and rewrite the screen. This however might not be enough, and to ensure the proper working of SCREED thereafter the user might have to reset the terminal; e.g. if he had issued the command to "clear all tab settings" SCREED would not be aware of this and would operate as though the default tab settings still applied.

* The "typo-matic" keys of this terminal make it particularly suitable for use with a screen editor.

5.6 Pericom 6801 Notes

* SCREED requires this terminal to be in page mode. The user must set the page switch, beside the screen, before invoking SCREED.

* The Auto CR/LF and Auto LF wheels at the back of the terminal should both be rotated upwards to the limit, i.e. switched off.

* The terminal has arrow keys, on a separate pad. Unfortunately, although they move the cursor around the screen, they do not generate any codes and so SCREED is unaware that the cursor has been moved. In other words, DO NOT USE THE ARROW KEYS. The same problem arises with the Tab key, so do not use it either. (The cursor has to be moved by the use of Backspace, Return, Line Feed and the appropriate two-letter commands, the (W) commands in particular.)

* Although the terminal has a line width of 80 characters, SCREED (for good reasons) does not use the 80th column of any line, and will ignore any character typed in the 80th column.

6. Command summary

6.1 Two-letter commands

As explained earlier, when a capital letter is given in parenthesis in this description, it represents the control character produced by typing the letter while holding down the Control key. Sometimes this is expressed as, for example, Ctl A. The equivalent notation here is (A).

Action		Text Unit
(A)dd		
(B)eginning of		
(C)onjoin		(W)ord
(D)isjoin	followed	
(E)nd of		(L)ine
(O)mit	by	
(N)ext		(S)creen
(P)revious		
(R)ewrite		
(T)runcate		

6.2 (X) commands

HELP		
CLOSE (or STOP or END or EXIT)		
ABORT		
SAVE	followed	
EDIT		(X)
ECCE	by	
text followed by '>' or '<'		
REPORT=<integer>		
MESSON or MESSOFF		
DEL=<character>		

If the text given before (X) is none of these then SCREED takes no action.

If (X) is typed on its own, i.e. without preceding text, the effect is to move the cursor to the left hand end of the current line. If the cursor is already at the left hand end, "thumbing" takes place.

EMAS 2900: USING SCREED ON AN ESPRIT TERMINAL CONNECTED TO A PAD

The standard version of the EMAS screen editor Screed has been modified to work with an Esprit or Esprit II terminal connected to a Packet Assembler/Disassembler (PAD), as well as the usual connection via a Terminal Control Processor (TCP). Screed detects automatically whether a PAD or a TCP is in use.

There are several changes necessitated by a PAD connection; this note summarises them.

The PAD causes every character typed on the terminal keyboard to be echoed to the screen. This includes all the 'control characters' which Screed uses for various purposes, particularly the two-letter commands. With most terminals, there are some control characters which have undesirable effects on the screen-editing process, and Screed, when operating through a TCP, causes these not to be echoed. This cannot be done when a PAD is used rather than a TCP, and so the user must rewrite the screen if undesirable effects are produced by pressing the wrong keys.

When the terminal connection is via a TCP, it is possible to ensure that the user's changes to the screen and Screed's changes to the screen are properly interleaved. For example, if the user were typing in new text and carried on typing after a carriage return, without waiting for Screed to put the cursor on the next line at the current indentation, this 'typed-ahead' input would not appear on the screen until Screed had placed the cursor correctly. If this deferred echoing did not occur then the user's input would appear on the screen wherever the cursor happened to be - possibly at the start of the line just completed - and the result would be a mess.

When the terminal connection is via a PAD there is no way that the interleaving described above can be ensured in every case. Screed tries to minimise the effect by rewriting each line typed on receipt of the terminating carriage return, but it will sometimes be necessary for the user to issue a (R)(R)(S) command ('Really rewrite screen'), unless he or she is prepared for Screed to complete its response in every situation.

Calling Screed

Before using Screed for the first time in an EMAS session, it is necessary to set various PAD parameters from the keyboard. This is done as follows:

1) Holding down the 'Shift' key, press 'Break'.

2) The prompt 'PAD>' will appear. Reply as follows:

```
PAD>NOHOSTTABS
PAD>NOPRINTMASK
PAD>NOGREENBOOK
PAD>
```

Press the 'Return' key in response to the final 'PAD>' prompt. This will terminate the PAD command sequence.

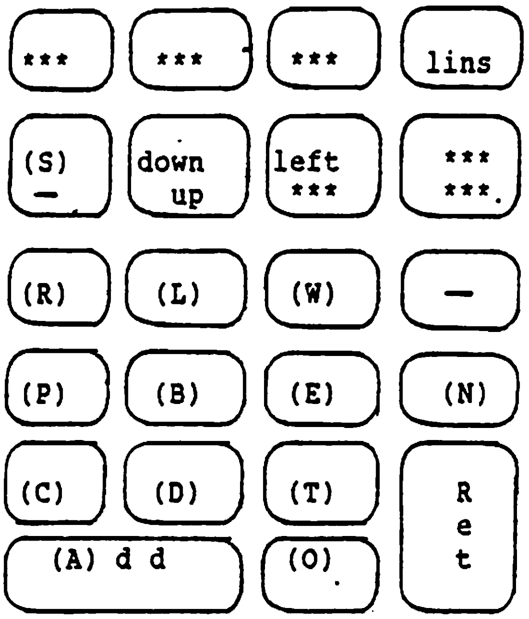
The user is reminded of this sequence on entry to Screed.

Using this version of Screed

To date, the only terminals which can be used via a PAD are the Hazeltine Esprit and Esprit II. These two terminals are almost identical in function; they differ in the meanings of some of the keys on the keypad (see below).

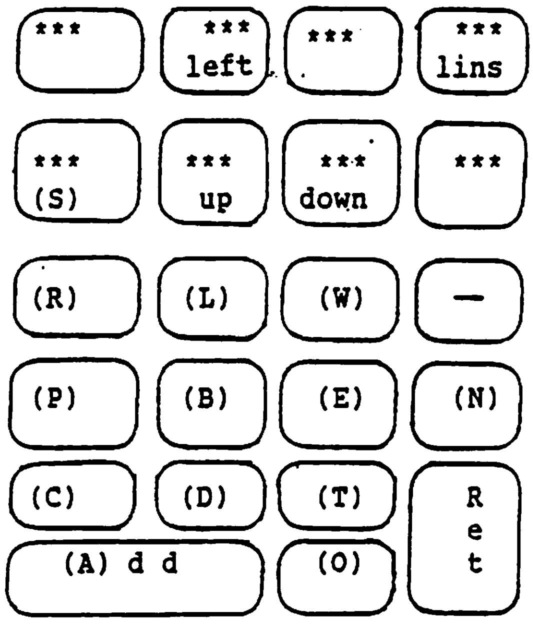
All the two-letter commands are now effected by use of the numeric keypad to the right of the main keyboard. When using the numeric keys, do not hold down the 'Ctrl' key. The mapping of letters to keys on the keypad are as follows:

(*** = Do not use)



Esprit keypad

(*** = Do not use)



Esprit II keypad

A card giving this layout should be available. If the terminal is used frequently with Screed, then it is recommended that the card be stuck onto the vertical surface above the keypad, to the right of the screen.

The 'Do Not Use' keys are so designated either because they are irrelevant or have an undesirable effect. To be specific:

* Cursor right (->) generates 'Ctrl P' (also known as DLE); this is interpreted by the PAD as an escape code - i.e. it affects the PAD, which does not treat it as data to be sent to EMAS.

* Delete Line generates 'ESC Ctrl S'. Ctrl S is also known as DC3, or X-off, and it causes the PAD to suspend all I/O until a Ctrl Q (also known as DC1, or X-on) is typed at the keyboard.

* The 'Local' key and the 'Block' key should not be used. If they are pressed inadvertantly, then the 'Block Mode' light above the keyboard will either light up or start flashing. Press the key again, with shift held down in the case of 'Local', and the light should go out. Do not continue with the edit session until you have managed to put the light out.

Other differences

* The (S) key (on the keypad) causes the cursor to be placed at the 'Home' position (top left-hand corner) on the screen, and can be used for this purpose if not immediately preceded by (R), (N), etc.

* The 'Line Insert' key (top of the keypad) can be used to insert a blank line (not a null line) immediately before the current screen line.

* The 'Line Feed' key is seen by Screed as if it were a Carriage Return. It is therefore equivalent to the 'Return' key except at the bottom of the screen where it causes the screen to scroll. Since Screed has no way of detecting that this has happened, it should not be used.

* The 'Rubout' key is equivalent to three cursor-right commands. Its effect is controlled by Screed directly rather than the PAD, and it may be rather sluggish as a result. However, Screed will also try to unscramble the effects of user type-ahead when it receives the Rubout code and so it can be used for this purpose.

* The only time that the Ctrl key need be used is when sending Ctrl X.

* The 'Tab' key will produce up to three space characters, possibly overwriting text as a result.

* It is possible to generate a System Interrupt ('Int:') from within Screed without all the editing being lost. In the usual way with PAD connections, an interrupt is caused by typing Ctrl+P followed by B, or Shift+Break followed by B and Return. If the interrupt generated terminates the edit session then the editing will be lost; otherwise it will be necessary to rewrite the screen before continuing.

* If a Screed session is terminated by use of an 'Int:' then it is advisable to reset the PAD connection to its 'normal' state. This is most simply done by calling Screed again without specifying any files. An error message will result, but the PAD reset will be effected as a by-product.

Conclusion

I would appreciate feedback concerning this version of Screed. If any users have suggestions for ameliorating the effect of the PAD, then I shall be happy to consider them.

John M. Murison
March 1985