

CURSOR LOCATION SYSTEM FOR MESSAGE EDITOR

This invention relates to a system for controlling the position of a cursor indicator on a display screen of a data processing system, for example a computer, a telex terminal or a word processor. The cursor is used in editing a message and/or a program on the screen. (RTM)

Figure 1 of the accompanying drawings shows a conventional cursor position control in a message editor which includes a data input with a keyboard (not shown), a message generator 2, a message memory 3 for storing a message and a display screen 4 on which a cursor 4a is shown. A cursor control unit 5 for moving the cursor 4a comprises an up/down cursor control 5-1 and a right/left cursor control 5-2. A message editor 6 is provided for editing a message.

A message is generated by the message generator 2 according to the input fed in by the data input 1, and the message thus generated is stored in the message memory 3 and is also indicated on the screen 4.

The feeding-in of a message is carried out by using the keyboard on the data input 1. That keyboard has a plurality of keys (for example alphanumeric keys and a return key).

The message thus generated is indicated on the screen 4 as shown in Figure 3(a), and the message is stored in the memory 3 as shown in Figure 3(b). In Figure 3(a), each line starts at the left-hand edge of the screen, and the location of the return (carriage return) (B,D,F) depends upon the length of each line. On each line of Figure 3(a), the right-hand area after the return symbol (B,D,F) is a null area which contains no data. The screen therefore has a data area and a null



area. The memory 3 does not have a null area, in order to save memory cells.

The cursor control 5 moves the cursor on the screen, so that the message on the screen can be edited.

5 The editing of a message includes correction, deletion and addition of a character positioned in a data area on the screen, and the destination of the cursor is usually in the data area, and not in the null area. Editing in a null area is not required, since no message
10 exists in the null area. Therefore, it should be noted that if the cursor moves into a null area, it is only transient and is on the way to the data area.

In a conventional method of locating a cursor, an "up" key and a "down" key in the up/down control 5-1
15 moves the cursor in a vertical direction to any position on the screen, and a "left" key and a "right" key in the right/left control 5-2 moves the cursor to any position in a horizontal direction.

In the example of Figure 3(a), it is assumed that
20 the cursor locates initially at the position D, and is to be moved to the position B. There are two ways of effecting the move, as follows.

In the first method, the cursor at D is first moved to B', immediately beneath the position B, by using the
25 right/left control 5-2, and is then raised by one line to the position B, by using the up/down control 5-1.

In the second method, the cursor at D is first raised by one line to B" by using the up/down control 5-1, and then it is shifted to the position B by
30 depressing the left key of the right/left control 5-2.

Accordingly, it should be noted that both the up/down control 5-1 and the right/left control 5-2 must be operated in order to move the cursor from D to B. Operation of the keyboard is therefore not simple.

It is an object of the present invention to overcome this disadvantage by providing a new and improved cursor control system.

According to the invention, there is provided a
5 cursor location system for a message editor which comprises a keyboard, a display screen, a message memory for storing a message, and a message processor for generation and editing of the message according to an input from the keyboard, the cursor location system
10 comprising a cursor memory which stores data representing the present location of a cursor on the screen, and X-addresses of the locations of the last data of the displayed message line in which the cursor is at present located and of the immediately above line and the
15 immediately below line; and a cursor processor for controlling the cursor memory and the position of the cursor on the screen, so that when the cursor is instructed by depression of an up key or a down key of the keyboard to move upward or downward to a destination
20 line, and the destination cursor location resulting from such depression would be one at which no data is displayed, the cursor is automatically positioned at the last data location on the destination line.

An embodiment of the invention will now be
25 described, by way of example, with reference to the accompanying drawings, in which

Figure 1 is a block diagram of a message editor including a prior cursor location system as described above,

30 Figure 2 is a schematic block diagram of a message editor including a cursor location system according to the present invention,

Figures 3(a) and 3(b) show an example of a message format and a memory allocation in a message memory,
35 respectively,

Figure 4 shows examples of the shifting of a cursor, Figure 5 is a flow diagram showing the operation of the cursor location system, and

Figures 6A and 6B are detailed block diagrams of a cursor location system according to the invention.

Figure 2 of the drawings shows a brief schematic block diagram of an embodiment of a cursor location system according to the present invention, in which the reference numerals 1 - 6 indicate the same components as in Figure 1. The apparatus further includes a cursor processor 7 and a cursor memory 8 which feeds the cursor location data to the cursor processor 7.

The cursor memory 8 has four cells a)-d) which store;

a) the address (X,Y) of the present cursor location,

b) the X-address of the last data location of the preceding (higher) line,

c) the X-address of the last data location of the present line,

d) the X-address of the last data location of the succeeding (lower) line.

Supposing that the cursor locates at a point Z in Figure 3(a). The cell a) of the cursor memory 8 stores the X-coordinate and the Y-coordinate of the location Z, the cell b) shows the X-address of the location B, the cell c) shows the X-address of the location D, and the cell d) shows the X-address of the location F.

In operation, it is assumed that a message is generated in a conventional manner, and that the message is indicated on the screen.

Figure 4 indicates the movement of the cursor, in which (a) shows the case where the X-address of the present cursor location (cell a)) is equal to or smaller than the X-address of the last data location of the

destination line (i.e. the preceding line or the succeeding line), and (b) shows the case where the X-address of the present cursor location (cell a)) is larger than the X-address of the last data location of the destination line (i.e. the preceding (higher) line or the succeeding (lower) line).

Figure 5 is a flow diagram showing the operation of the cursor processor 7.

In Figure 5, a box 100 indicates the start of the operation. A box 102 tests whether or not the up or down key is depressed. When the down key is depressed, the control goes to a box 114, and when the up key is depressed, the control goes to a box 104.

In the boxes 104 and 114, if the X-address of the present cursor location is equal to or smaller than the X-address of the last data location of the preceding line (box 104), or the succeeding line (box 114), then the control goes to box 108 or the box 118, respectively (see Figure 4(a)). The operation in those cases is the same as that of the prior art of Figure 1, and the cursor shifts upwardly or downwardly as shown in Figure 4(a). Then, the cursor is shifted horizontally to the last data location of the line, as in the conventional system.

On the other hand, if the X-address of the present cursor location system is larger than the X-address of the last data location of the preceding line (box 104) or the succeeding line (box 114), then the control goes to a box 106 or a box 116, respectively, and the cursor moves to the last data location of the preceding line or the succeeding line as shown in Figure 4(b), via a path S. It should be noted that in the conventional system, the cursor at the location A goes to the location B via paths t_1 and t_2 . Therefore, the present invention simplifies and reduces the operation of the cursor.

The X-address of the present cursor location in the cell (a) is then replaced by the last data location (b) of the preceding line (box 106), or the last data location (d) of the succeeding line (box 116). The
5 Y-address of the present cursor location in the cell (a) is also incremented (in the case of a lower line), or decremented (in the case of an upper line).

After the cursor moves, the content of the cursor memory 8 is updated according to the new location of the
10 cursor as shown in the boxes 108, 110, 118 and 120. In the box 108, the content of the cell d) which has the last data location of the succeeding line is substituted by the last data location of the present line in the cell c), and the content of the cell b) which has the last
15 data location of the preceding line enters into the cell c), which then has the updated last data location of the present line. The content of the cell b), which has the last data location of the preceding line, is revised by referring to the message memory (box 110).

20 Similarly, in the box 118, the cell b), which has the last location of the preceding line, is replaced by the last data location of the present line in the cell c), and the content of the cell d), which has the last data location of the preceding line, enters into the cell
25 c), which then has the updated last data location of the present line. The content of the cell d), which has the last data location of the succeeding line, is revised by referring to the message memory 3 (box 120). A box 130 shows the end of the operation.

30 Figures 6A and 6B show block diagrams of the cursor processor in the present invention. The processor comprises a selector 200 which selects one of two input signals U and D according to the output of a cursor key sensor 202. The cursor key sensor 202 provides a signal
35 u when an up key is depressed, and a signal d when a down

key is depressed. A comparator 204, which has two input signals S and T, provides an output signal when the input signal S is higher than the input signal T. The processor further comprises an AND circuit, a memory cell 5 208 which stores the present cursor location (i.e. the cell a) of Figure 2) and has a row register for storing the Y-coordinate of the present cursor location and a column register for storing the X-coordinate of the present cursor location. A memory cell 210 corresponds 10 to the memory cell of Figure 2, and stores the X-coordinate of the last data on the present cursor line. A cell 212 corresponds to the memory cell b), and stores the X-coordinate of the last data location on the preceding line. A cell 214 corresponds to the memory 15 cell d), and stores the X-address of the last data location of the succeeding line. An adder 216 increments its input signal and a subtractor 218 decrements its input signal. The processor includes subtractors 220 and 222, a register 224 for storing the start address of a 20 preceding line, a register 226 for storing the start address of the present line and a register 228 for storing the start address of the succeeding line. An address counter 230 provides an address signal for the message memory 3, and a flag sensor 232 senses a flag in 25 the output of the message memory 3.

In the case of Figure 4(a) where (a) is equal to or smaller than (b) for an upward shift, or (a) is equal to or smaller than (d) for a downward shift, the comparator 204 does not provide an output, and the column address 30 register is not updated, but the row address register is incremented or decremented, according to the depression of an up key or a down key. Therefore, the cursor merely shifts in an upward direction or a downward direction. Then the cells (b), (c) and (d) in the registers 212, 35 210, and 214, respectively, are updated by the previous

content of those registers or the content of the message memory 3.

On the other hand, in the case of Figure 4(b) where (a) is larger than (b) for an upward shift, or (a) is larger than (d) for a downward shift, the comparator 204 provides an output signal. Then the AND circuit 206 provides an output signal when either an up key or a down key is depressed, and the output signal of the AND circuit 206 is applied to the column address register 10 which then accepts one of the data locations (b) or (d). Accordingly, the X-address of the cursor on the destination line is updated to the X-address of the last data location of the destination line itself. Simultaneously, the row register in the present cursor 15 location memory 208 is incremented or decremented by the adder 216 or the subtractor 218. Accordingly, both the row register and the column register in the present cursor location memory 208 are updated, and the updated content of the cell (a) shows the location of the last 20 data location of the destination line. The last data locations (b), (c), and (d) in the registers 212, 210 and 214 are updated also by the previous contents of those registers and the message memory 3.

Next, the updating of the cells (b) and (d) by the 25 message memory 3 (the boxes 110 and 120 in Figure 5) is described. That function is shown in the block diagram of Figure 6B which is connected to Figure 6A through lines *1, *2, *3. Figure 3(b) shows the memory allocation of the message memory 3, in which each screen 30 line has a plurality of characters (indicated by A to B, C to D, and E to F), and at the first address of each line a CR code (carriage return) is stored with a flag bit in the ON state in order to show the end of the preceding line. The start (first) address of the present 35 cursor line, the preceding line, and the succeeding line

are stored in the registers 226, 224 and 228, respectively (see Figure 6B).

When the last data location of the upper line is to be searched (box 110 in Figure 5), the start address of the preceding line in the register 224 is set in the address counter 230 as an initial address, and the content of the memory 3 designated by the address counter 230 is then read out. The address counter 230 is decremented one by one. The flag sensor 232 tests whether the content of the memory 3 thus read out according to the address counter 230 has a flag in the ON state. When a character with a flag in the ON state is read out, the decrementing of the address counter 230 stops, and the difference between the content of the address counter 230 and the content of the register 224 is then calculated by the subtractor 220, which feeds the difference to the cell (b) (see Figure 4A) which stores the X-address of the last data location in the upper line.

Similarly, when the last data location of the lower line is to be searched (box 120 in Figure 5), the start address of the succeeding line in the register 228 is set in the address counter 230, and the content of the counter is then incremented until the flag is sensed by the flag sensor 232. Upon detection of the flag, the decrementing of the address counter 230 stops, and the subtractor 222 provides the difference between the content of the address counter 230 and the content of the register 228. That difference is applied to the cell (d).

As described above, by using the system of the present invention, when the cursor shifts upward or downward, the cursor is automatically located at the last data location of the destination line when the immediately upward location or the immediately downward

26

-10-

location, as the case may be, contains no data, merely by depressing an up key or a down key. Therefore, the number of key operations for locating the cursor is reduced, and the time required for editing a message is
5 shortened. Considering that the cursor is frequently located at the last data location of a line, the present invention is very advantageous.

27



CLAIMS

1. A cursor location system for a message editor which comprises a keyboard, a display screen, a message memory for storing a message, and a message processor for generation and editing of the message according to an input from the keyboard, the cursor location system comprising a cursor memory which stores data representing the present location of a cursor on the screen, and X-addresses of the locations of the last data of the displayed message line in which the cursor is at present located and of the immediately above line and the immediately below line; and a cursor processor for controlling the cursor memory and the position of the cursor on the screen, so that when the cursor is instructed by depression of an up key or a down key of the keyboard to move upward or downward to a destination line, and the destination cursor location resulting from such depression would be one at which no data is displayed, the cursor is automatically positioned at the last data location on the destination line.

2. A system according to claim 1, wherein the cursor processor includes a comparator for comparing the X-address of the present cursor location with the X-address of the last data on the immediately above or immediately below line, according to whether the up key or the down key, respectively, is depressed, the comparator producing an output when the former X-address is larger than the latter X-address, to cause the cursor to be positioned at the last data location on the destination line.

3. A system according to claim 1 or claim 2, wherein the content of the cursor memory includes a flag which indicates the end of a line.



-12-

4. A cursor location system substantially as hereinbefore described with reference to Figures 2 to 6(B) of the accompanying drawings.
