

[54] **ELECTRONIC TEXT DISPLAY SYSTEM
WHICH SIMULATES A TYPEWRITER**

[75] Inventors: **Arnold J. Goldman**, Encino;
Stephen L. Kurtin; **Carver A. Mead**,
both of Pasadena, all of Calif.

[73] Assignee: **Lexitron Corporation**, Burbank,
Calif.

[22] Filed: **July 12, 1971**

[21] Appl. No.: **161,554**

[52] U.S. Cl. **340/172.5, 340/324**

[51] Int. Cl. **G06f 3/10, G06f 3/14**

[58] Field of Search **340/324 A, 324 AD,**
340/172.5

[56] **References Cited**

UNITED STATES PATENTS

3,648,271	3/1972	McConnell et al.	340/324 A
3,593,310	7/1971	Kievit.	340/172.5
3,540,012	11/1970	Ehrman	340/324 A
3,242,470	3/1966	Hagelbarger et al.	340/172.5
3,639,736	2/1972	Sutherland.	340/324 A

Primary Examiner—Gareth D. Shaw

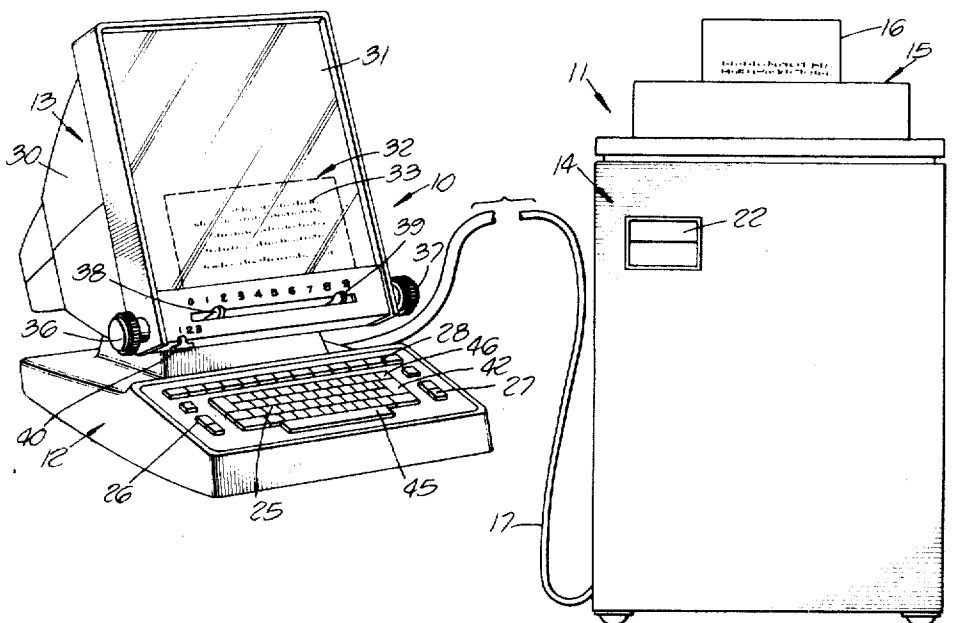
Attorney—John G. Mesaros

[57]

ABSTRACT

There is disclosed herein an electronic typewriter system including an operator station and desk-side console. The operator station includes a keyboard and a visual display for typed text, and the console includes suitable control and memory means and a print-out device. The system enables textual material to be entered via the keyboard and to be displayed in a full page format on the display, as well as to be stored in a memory. The system provides editing capabilities for changing, editing and manipulating the text, and a hard copy of the displayed text can be typed out by a printer upon command. The keyboard and display are constructed and organized to simulate the operational features of a conventional electric typewriter. Special equipment includes simulated platen knobs, margin indicators, space and back space keys, and a vertical spacing lever similar in use to those found on conventional electric typewriters, along with electronic subsystems acting in cooperation therewith to enable display of a page outline, page movement on the display, setting of margins for the typed text, controlled movement of a position indicator or cursor, and so forth.

5 Claims, 11 Drawing Figures



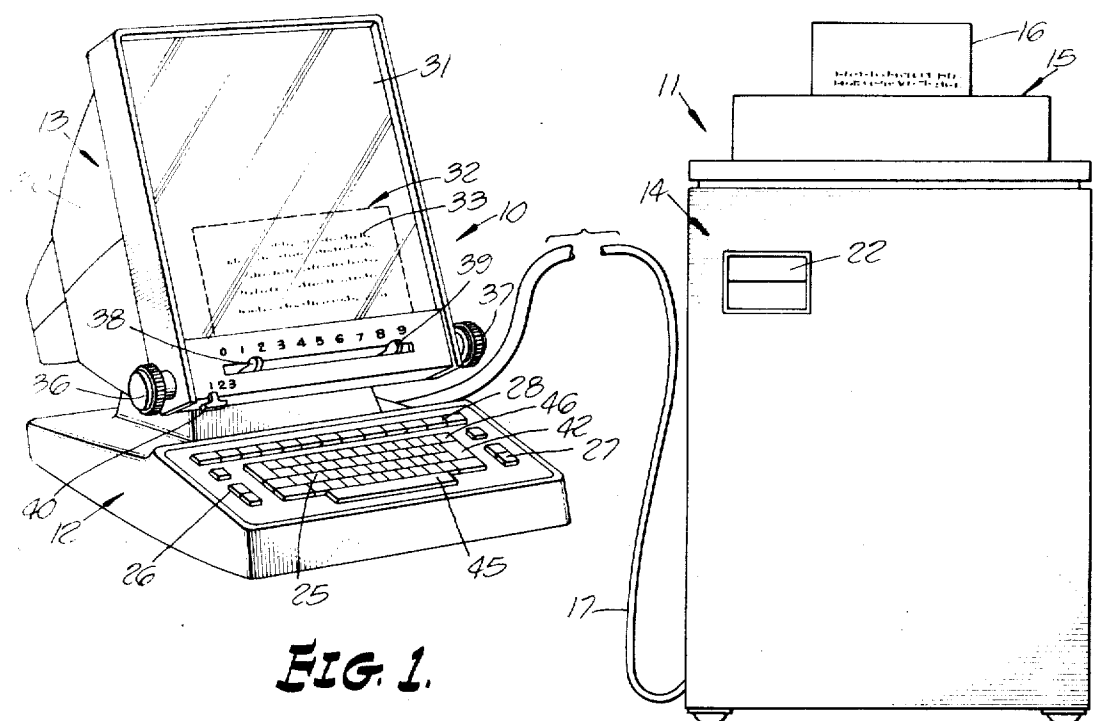


FIG. 1.

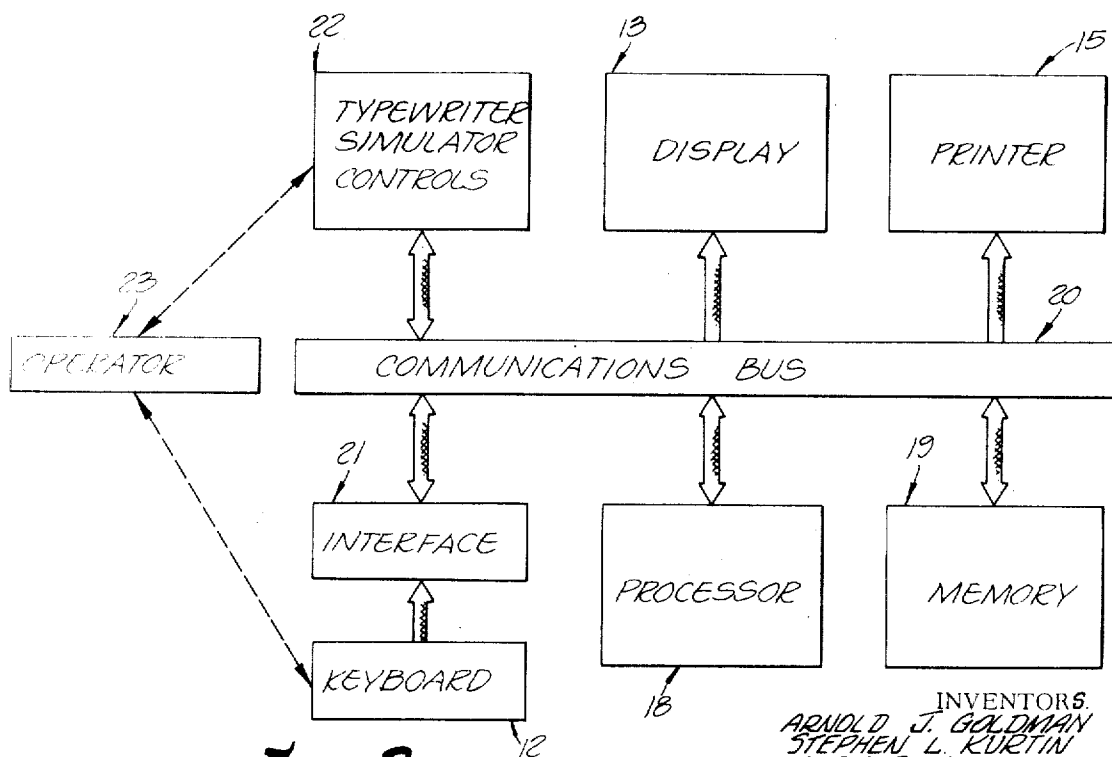


FIG. 2.

INVENTORS:
ARNOLD J. GOLDMAN
STEPHEN L. KURTIN
BY CARVER A. MEAD

Lyon & Lyon
ATTORNEYS.

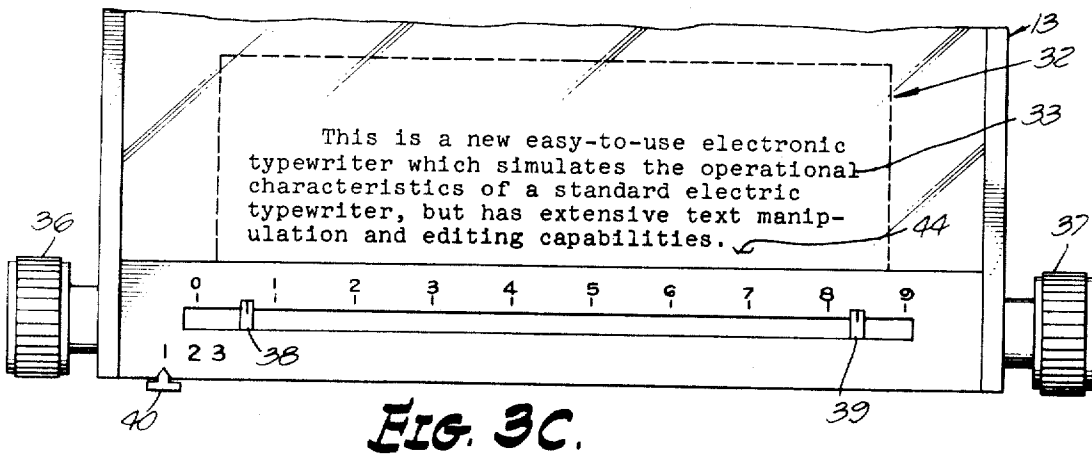
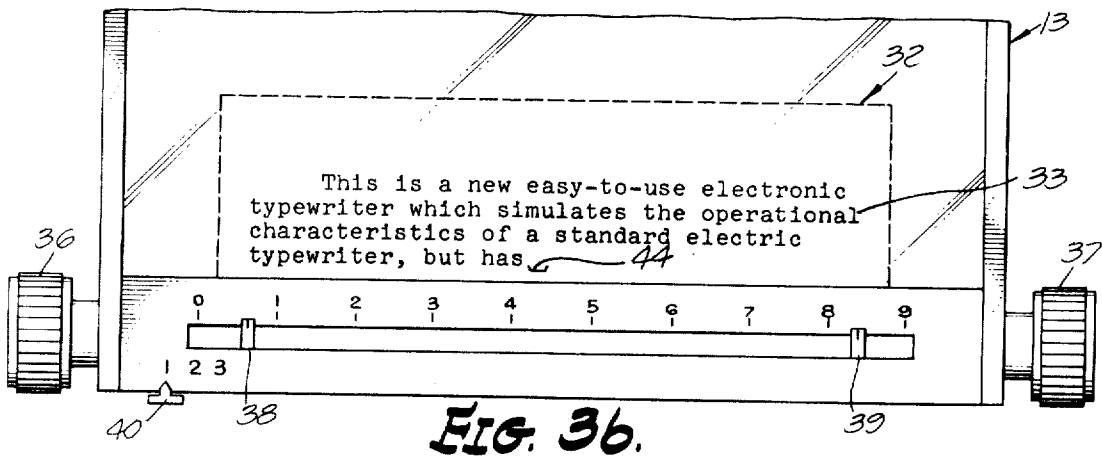
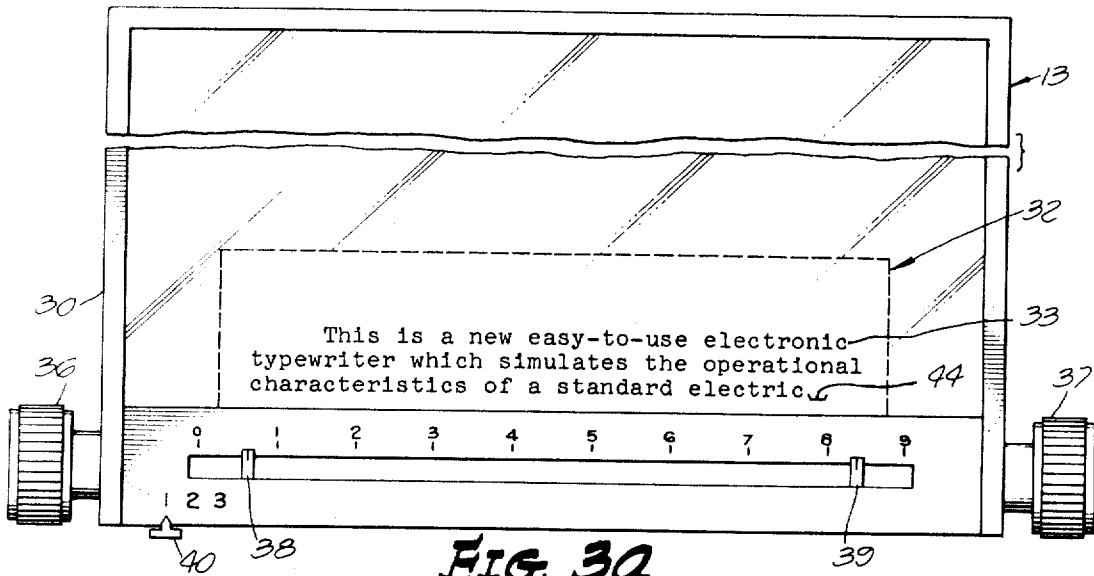
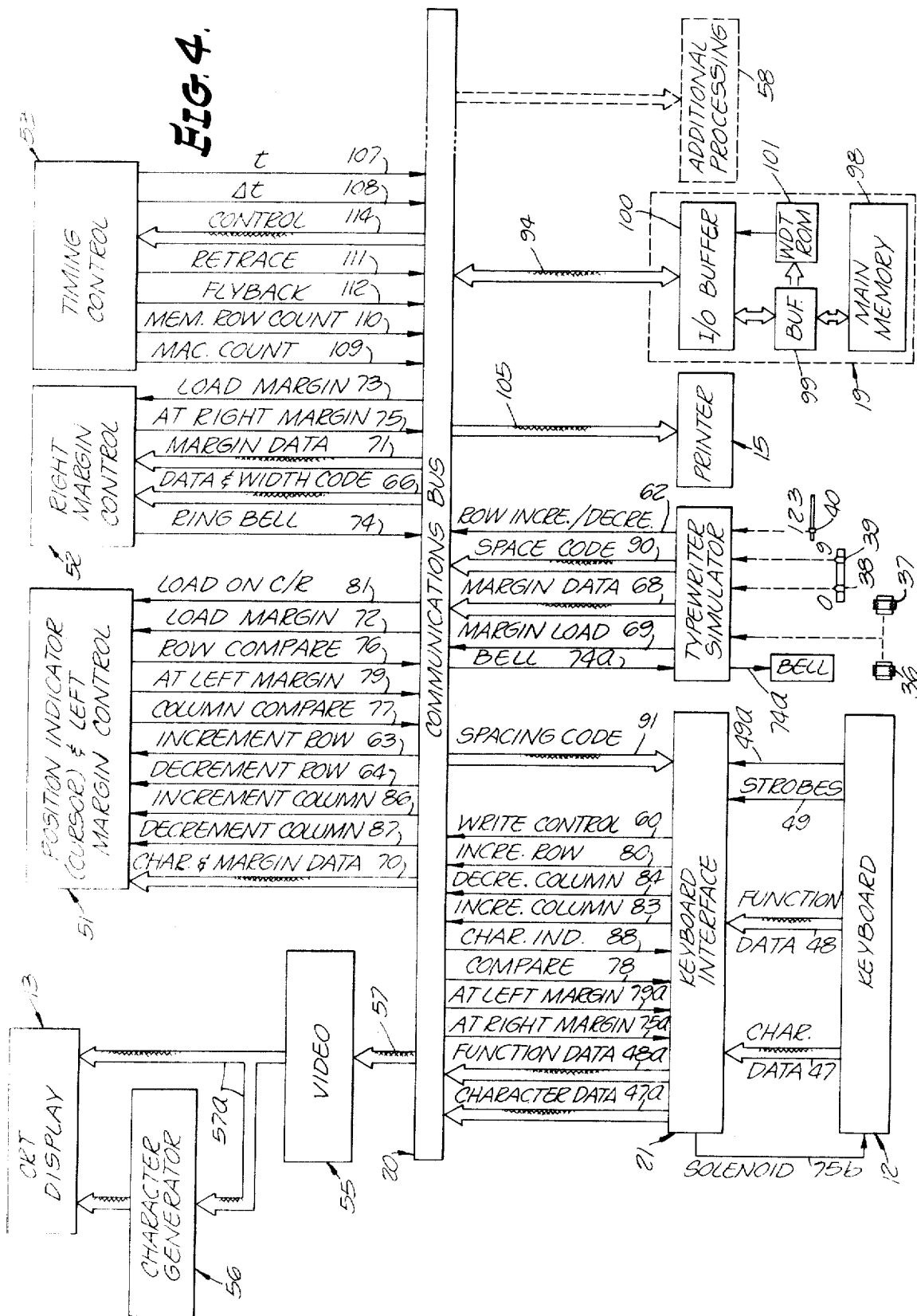


FIG. 4.



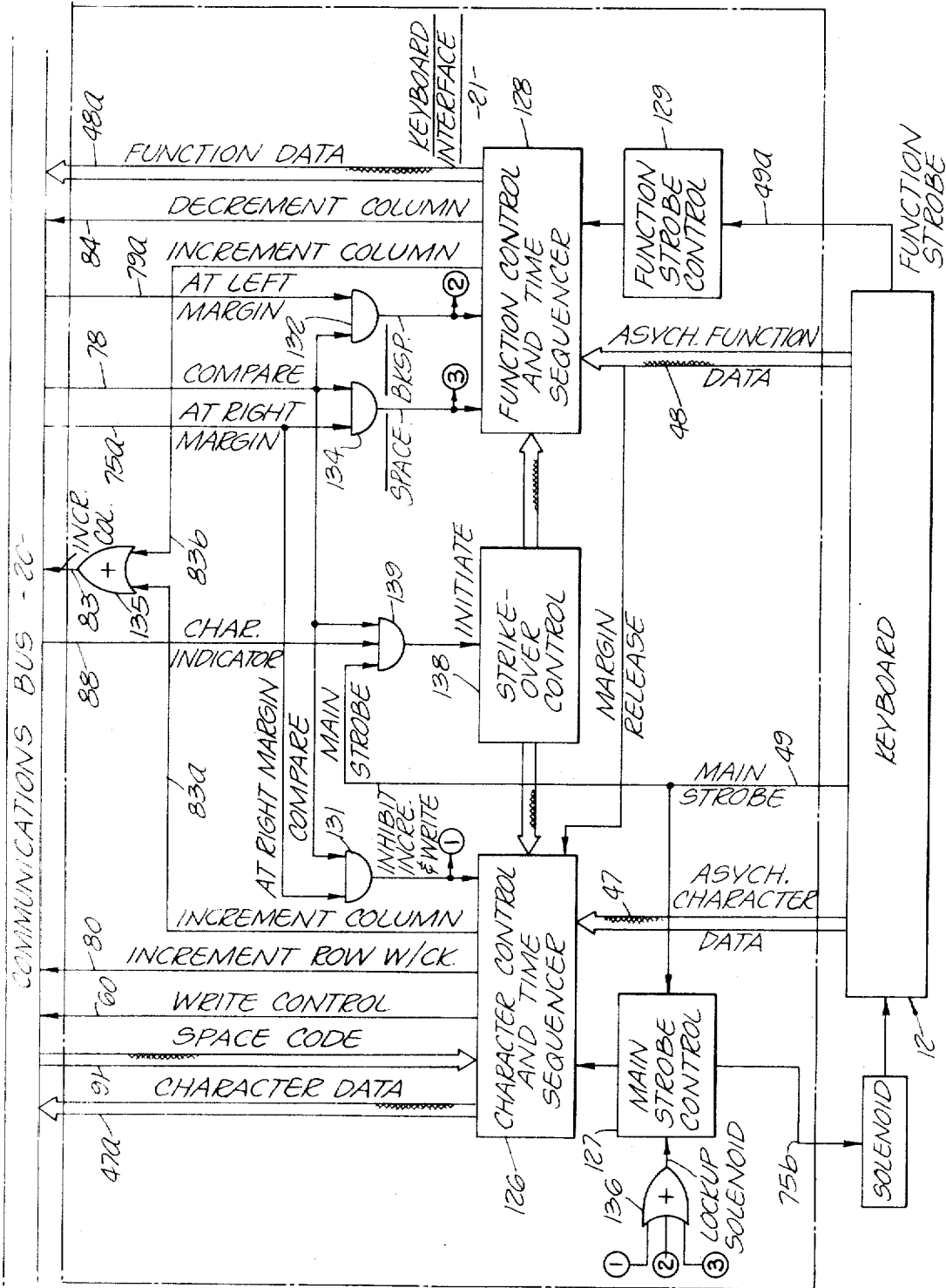


FIG. 5.

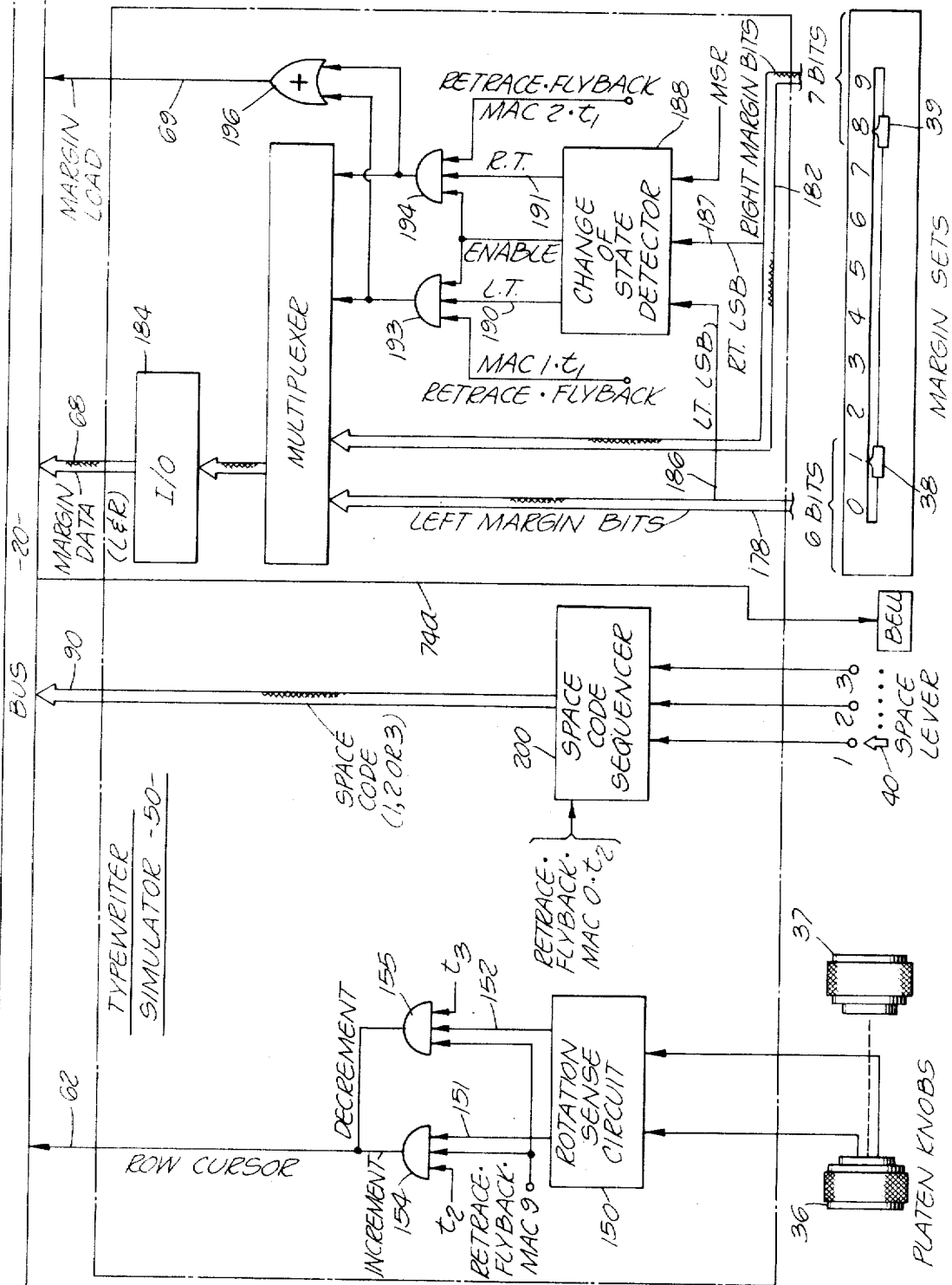


FIG. 6.

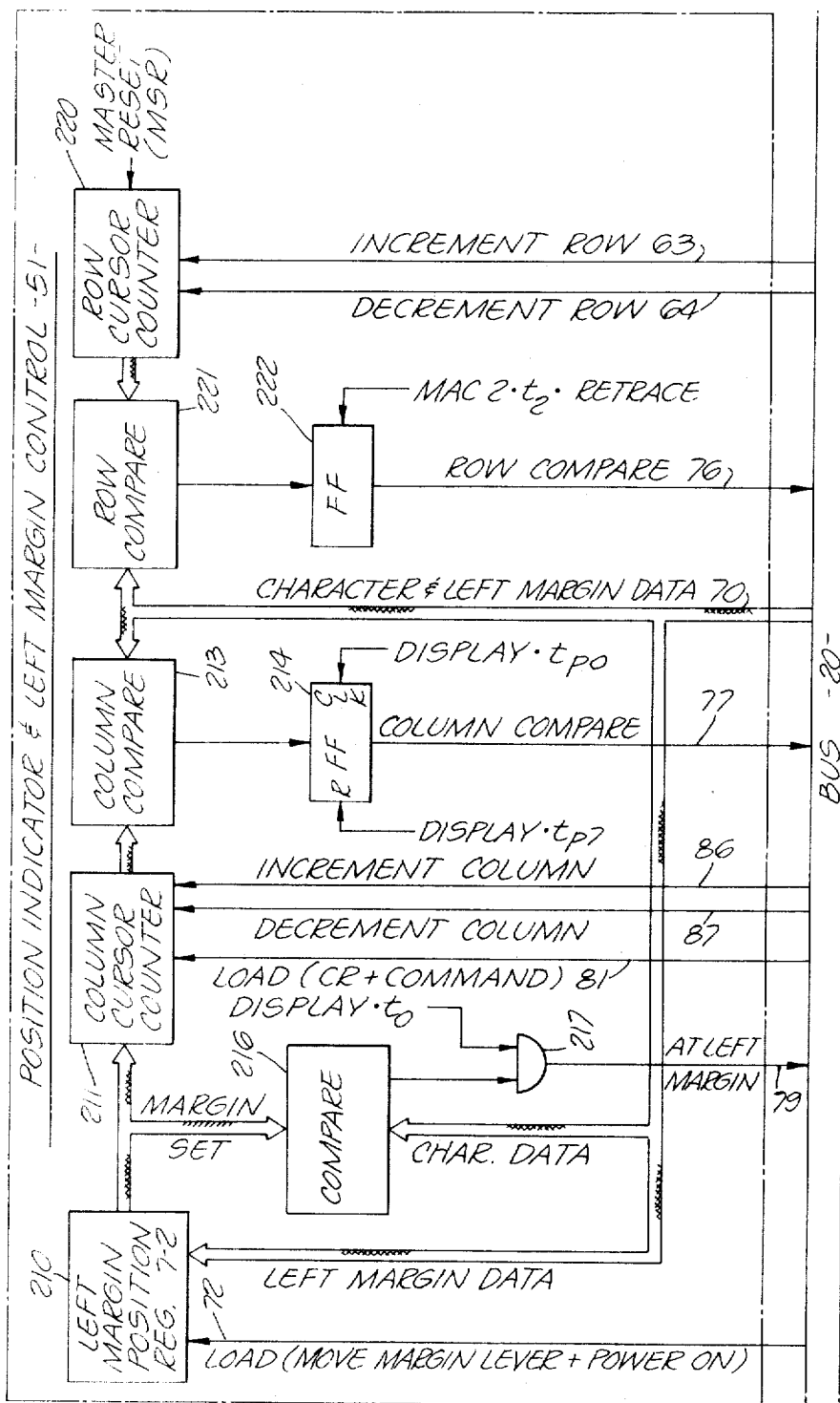
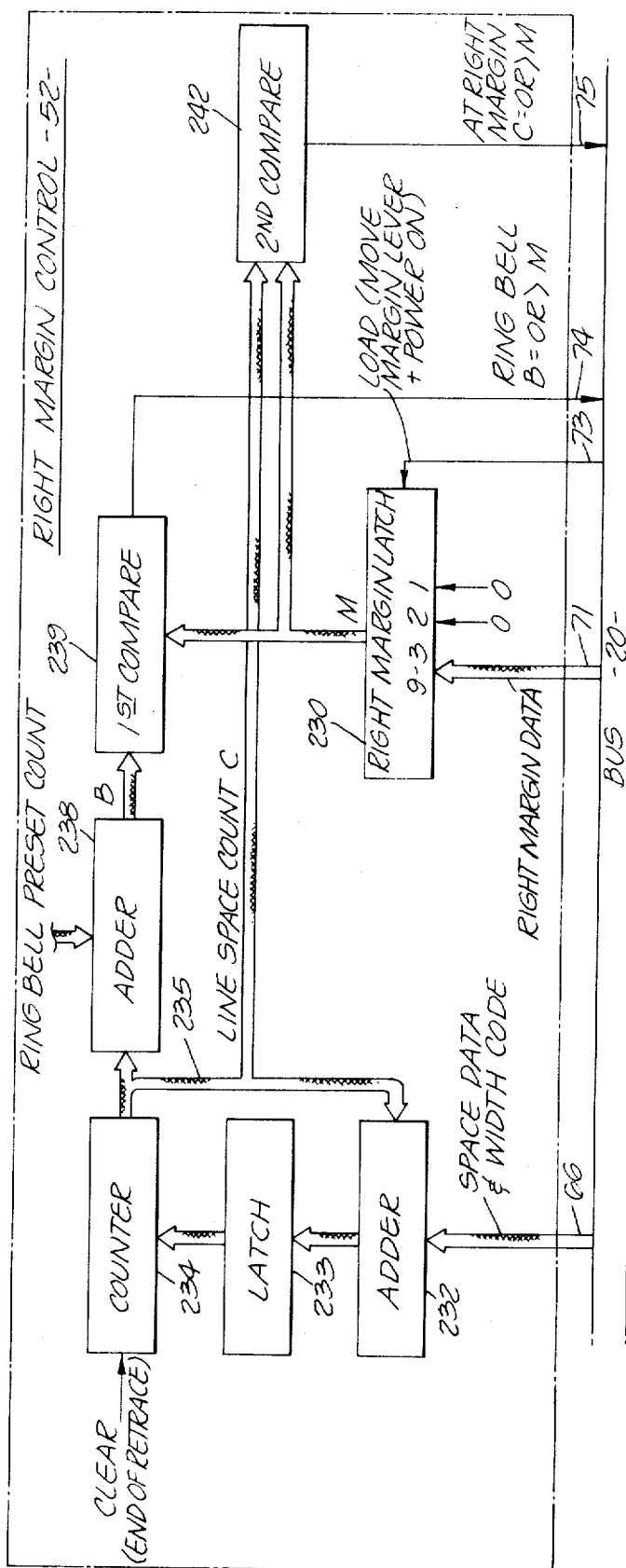
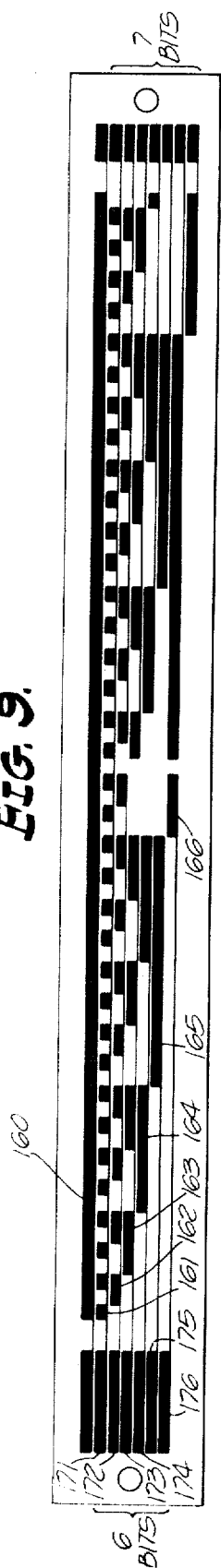


FIG. 7.

ஆ



857

ELECTRONIC TEXT DISPLAY SYSTEM WHICH SIMULATES A TYPEWRITER

This invention relates to an electronic typewriter system and more particularly to an electronic typewriter and text display system which simulates the operational features of a conventional typewriter.

BACKGROUND OF THE INVENTION

Many display systems are known in the data processing field. Display systems having high information density usually use a display of the cathode ray tube type. Such display systems often are line organized so as to display high level language computer programs, and may present columnar information and thus may have one or more tabular positions. If the display is to be used in a data entry or interactive mode, it usually features a position indicator or cursor, free to be directed in some manner to any printable position on the display, to indicate the location at which the next display related operation is to occur. Typically, the position indicator is moved or otherwise positioned by specifically assigned buttons, keys, a joy stick, or other manually operative devices. Furthermore, such systems generally have, and rely upon, certain added operator controls for data entry, data manipulation and cursor movement which are unfamiliar to the average typist.

SUMMARY OF THE INVENTION

The present invention is directed to an improved display system for text editing and manipulation wherein various control functions are provided by control devices which simulate those of a conventional typewriter. The display system is particularly useful as an interactive display subsystem of a real-time text processing computer. A significant objective of the invention therefore is to provide a display system which simulates, from the operator's viewpoint, a conventional typewriter. By fulfilling this objective, operator training in the use of the system, as well as errors in use, are significantly minimized.

Briefly, in accordance with the concepts of the present invention, a display system is provided employing a display such as a cathode ray tube or other high information density display means; and a set of controls including (1) a keyboard for data entry, the data and function data for formatting and manipulating text which is displayed, and (2) special purpose controls, such as those which simulate the operation of the platen knobs, margin indicators, and a vertical spacing lever. Additionally, the system includes a control subsystem or real-time text processor which interacts with the display and controls, as well as with a memory, for properly routing and acting upon the data and controlling the display of displayable data, and directing storage of textual information and control signals.

The text is displayed on the cathode ray tube in the same format as a typed page. The displayed page outline and text thereon are moveable on the display by operator controls which simulate the action and operation of those found on a conventional typewriter. The text is rolled past the print line by rotatable pseudo platen knobs such as paper is moved past the printhead of a common office typewriter. A position indicator or cursor is allowed to move along the bottom line of the page on the display screen, this line being referred to as a print line, under control of operator controls, such

as the keyboard "space" and "backspace" keys. Vertical spacing of one or more lines is accomplished upon depression of the carriage return key, depending upon the positioning of a vertical spacing lever which thus determines the number of blank lines to be inserted between successive depressions of the keyboard carriage return key. Horizontally sliding margin indicator levers are provided to set text margins at positions on the display screen (and on the print-out copy), relative to the edge of the page outline displayed on the screen, which correspond with the physical position of the margin levers.

Accordingly, it is a principal object of the present invention to provide a new electronic typewriter system.

An additional object of this invention is to provide an electronic typewriter system which simulates the operational features of a conventional typewriter.

A further object of this invention is to provide an improved electronic text display system which simulates a typewriter.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects and features of the present invention will become better understood through a consideration of the following description taken in conjunction with the drawings in which:

FIG. 1 is a perspective view of a system according to the present invention including an operator's station and control console;

FIG. 2 is a simplified block diagram of the system;

FIGS. 3a through 3c illustrate the display of text material on the display screen of the operator's station;

FIG. 4 is a detailed block diagram of the present system;

FIG. 5 is a more detailed block diagram of a keyboard and keyboard interface of the system of FIG. 4;

FIG. 6 is a more detailed block diagram of the typewriter simulator control and subsystem of the system of FIG. 4;

FIG. 7 is a more detailed block diagram of the position indicator and left margin control of the system of FIG. 4;

FIG. 8 is a more detailed block diagram of the right margin control of the system of FIG. 4; and

FIG. 9 is a detailed view of the position decoder for the margin indicator levers.

DESCRIPTION OF THE PREFERRED EMBODIMENT

General Discussion of System and Simulation

Turning first to FIG. 1, there is illustrated an operator station 10 and a desk-side console 11. The operator station 10 includes a keyboard 12 and a cathode ray tube display 13, or other suitable high information density display means. The console 11 houses the electronic system within a cabinet 14 and a printer 15 for printing-out a typed page 16 on command. The station 10 and console 11 are interconnected by a cable 17.

Considering FIG. 2 in conjunction with FIG. 1, the console 14 may house an electronic processor 18, memory 19, and communications bus 20, as well as the electronic interface and control logic 21 of the keyboard 12. Associated with the operator station 10 is a typewriter simulator and controls 22. The overall system, as will appear subsequently, may include both an internal dynamic memory 19 as well as a magnetic tape memory of cassette form for static storage of data, and

the console 14 of FIG. 1 includes a drawer 23 for receiving a tape cassette.

An operator, or typist, 24 is included in the block diagram of FIG. 2 inasmuch as the operator forms an important link in the overall system which is a real time, interactive text processing and display system which places the operator in direct control of text content, format, and page layout. The operator views the display 13 and communicates with the system by means of the keyboard 12 and typewriter simulator and controls 22. Turning more specifically to the operator environment, that is the operator station 10 of FIG. 1 which includes the keyboard 12, display 13 and controls 22, the keyboard 12 includes a central cluster of typewriter-like keys which is used by the operator for text generation, i.e., text entry. The output of alphanumeric keys of cluster 25 is in the form of electronic signals representing, in coded form, textual information. In addition to the central cluster of keys 25, additional groups of keys 26, 27 and 28, which are essentially control switches, may be provided and used by the operator to perform text editing and manipulating functions as will be described in more detail subsequently. The outputs of these switches 26-28, as well as function switches (e.g., space bar) of the cluster 25, likewise are electronic signals but these signals are interpreted by the processor 18 of FIG. 2 to initiate and control the various functions, including editing and manipulation of text desired without requiring the operator to perform or know any complicated commands or programs.

The display 13 may include a cathode ray tube mounted within a housing 30 having a suitable transparent cover 31. The cathode ray tube presents to the operator an image of an outline 32 of a page as well as an image of typed text 33 thereon. All textual information entered on the keyboard is directed by the processor 18 to the memory 19 and is displayed on the display 13. The format of the displayed text is controlled by the processor according to pre-established format rules, the bounds of which are set by the operator by means of controls of a familiar nature, such as margin indicator levers, platen knobs, vertical space lever, and the like, which forms the controls of the typewriter simulator and controls 22.

Considering these latter controls in more detail, a pair of rotatable platen knobs 36 and 37 are provided and are conveniently mounted on the display 13 as seen on FIG. 1. The knobs enable the page outline and text to be moved up and down on the display. Additionally, a pair of movable margin indicator levers 38 and 39 also are conveniently placed at the lower portion of the display 13. The levers allow the left and right margins of the text to be established. A vertical space lever 40 is provided to determine vertical spacing of one, two or three lines upon depression of a carriage return button 42 disposed in a conventional position in the central keyboard cluster 25.

A principal feature of the present invention is the operator station 10 shown in FIG. 1 and the associated electronic subsystems for enabling display and movement of the page outline 32 and text 33 thereon in the same manner as a typed sheet of paper within a conventional typewriter under control of the operator in a manner familiar to a typist. The display 13 essentially simulates in every detail a typed page. The displayed page is the same aspect ratio as a final typed copy, and

visually presents the displayable contents of the memory 19 to the operator.

The display interactive controls further aid in typewriter simulation. These controls include the platen knobs 36 and 37 whose operation generates electrical signals interpretable by the processor as a command to move the displayable contents of the memory on a line-at-a-time basis, past the print line (i.e., the bottom line of the display) so as to simulate the platen knob function on a typewriter and thereby make every line of text accessible in a familiar manner as well as accessible for editing functions. These controls also include the movable margin indicator levers 38 and 39 which are set by the operator by manual positioning thereof. The positioning of these levers generates electrical signals interpretable by the processor 18 as margin positions for the displayed text 33. The lever 38 establishes the left margin of the text 33, and the lever 39 enables a bell ring and keyboard "lock-up" to occur as with a standard typewriter. Positioning the vertical space lever 40 likewise generates an electrical signal interpretable by the processor 18 as the number of blank lines desired between typed lines, and causes the displayed material to advance a given number of lines (such as one, two or three) each time the carriage return key 42 is struck.

The display 13 not only serves to display the displayable contents of the memory 19 but also serves as the means by which the operator "gets her hands on" (i.e., selects and manipulates) those portions of the text 33 which she may wish to edit or revise; such as, delete one or more characters, add new characters or symbols, add or delete words, and so forth. The bottom line of the display 13 is, as when typing on a sheet of paper in a typewriter, the print line as noted before. A position indicator or cursor 44 as seen in FIGS. 3a-3c moves along the print line under control of a "space" key 45 and "backspace" key 46 located in conventional positions in the keyboard cluster 25. This position indicator identifies the position in which, or at which, the next operation will occur. Typical operations may include test generation (i.e., typing characters), editing (i.e., deleting and/or adding one or more characters), and so forth. These operations are performed by the processor 18, as will be explained in more detail subsequently, as it interprets the direct control commands of the operator. The simplest direct control command, for example, is the depression of an alphanumeric key of the cluster 25 which indicates that the operator desires that alphanumeric character to appear on the display in the correct print position. The coded signal coming from the keyboard is entered in memory, and the processor may modify any pre-existing code in memory as required, and instruct that the result be displayed. The operator thus immediately sees on the display 13 the result of operations she performs. Hence, she can interact directly, via the display, with the contents of the memory 19.

Unlike interpretive computer systems, in the present system the operator is not required to write instructions into a computer. The commands are direct and interactive in real-time in a manner familiar to a typist. Text positions on the display correspond to memory locations, and the operator selects (e.g., via the platen knobs 36 and 37, space bar 45, etc.) the location on the page at which she wishes to perform a text operation, and depresses the appropriate keyboard button. The

processor interprets the electrical signal from the keyboard, and modifies memory contents appropriately, with the result being immediately displayed. The operator thus may be considered to interact directly with the text stored in memory.

Turning briefly to the processor 18, the same is an electronic subsystem which monitors and augments the flow of text and control signals in response to operator commands. Some of its principal functions in the present system involve interpretation of the positions of the margin indicator levers 38 and 39 and proper positioning of the text 33 on the display 13, movement of the page outline 32 and text 33 in response to operation of the platen knobs and carriage return, control of the positioning of the cursor, proper routing of data, including character or text data and function or control data. The processor acts to control text held within the memory 19, and the memory in turn acts in conjunction with the processor. Text manipulation (e.g., deletion or addition of a character) is performed within the memory but under processor control.

The memory 19 is line organized and holds the text being displayed, edited and manipulated. The displayed text is a subset of information stored in memory as determined by the processor as it interprets operator commands. For example, depression of an alphanumeric key causes a code to be stored in memory corresponding to the desired character; depression of the carriage return key causes a carriage return code to be stored; and so forth. The text in memory is available for display and is manipulated by the operator by keyboard commands issued to the processor. For example, an entire page may have been typed and displayed, but the typist can return to a particular portion (such as the first three lines as seen in FIG. 3a) for editing or correcting as desired through use of the platen knobs and space or backspace keys.

The basic memory is essentially a dynamic recirculating memory such that its contents (i.e., successive coded signals representing successive characters, spaces and control codes) are continually refreshed. The memory stores internal control codes which are not displayed but are required for the processor to recognize and operate on the text in memory. Such control codes include, for example, carriage return, space, and identification of a strike-over (i.e., ϕ) to be explained in more detail subsequently.

As will be readily apparent, the printer 15 serves to produce upon command a hard copy record of the text generated. The text flows from memory to the printer 15 under processor control. The printed copy is identical in content and format to the page image as seen on the display 13. The printer need be activated only when a hard copy is desired. All text entry and manipulation is performed in the interactive mode using the display 13 as a "preview image" prior to final print-out.

As noted earlier, the system also may include an auxiliary tape memory. This may take the form of a magnetic tape cassette which serves as an auxiliary, static, and quasi-permanent text record. Interactive text generation and manipulation functions are performed between the processor and the dynamic memory, but when the operator wishes to indefinitely store the contents of the dynamic memory for future retrieval, it may be transferred to the tape cassette. Similarly, the text stored on the cassette may be retrieved and displayed on the display 13.

BRIEF DESCRIPTION OF SYSTEM

A brief description of the system as illustrated in FIG. 4 will be given before embarking upon a more detailed discussion thereof and the display and simulation concepts. In the system as illustrated in FIG. 4, all but the keyboard 12, the display 13, printer 15, memory 19 and communications bus 20 are assumed to be a part of the processor 18 of FIG. 2. The keyboard interface is shown separate in FIG. 2 for facilitating an understanding of the system, but more logically is considered a part of the processor. The processor, as previously explained, monitors and augments the flow of text and control signals in response to operator commands. It includes a keyboard interface 21 which provides communication between the keyboard 12 and the bus 20. The interface 21 supplies, for example, character data representative of the alphanumeric keys which have been struck by the typist, as well as function data indicating control functions (i.e., carriage return, character delete, character erase, edit, etc.) directed by the typist. It also enables control of the flow of such data to the communications bus 20 so that the data can be appropriately used by other subsystems as needed. The character data and function data are applied by cables 47 and 48 to the keyboard interface 21, which in turn supplies such data in proper time sequence by cables 47a and 48a to the bus 20. Strobe signals, which are essentially gating signals which occur whenever keys are struck, are applied by lines 49 and 49a to the keyboard interface 21. Each alphanumeric key of the keyboard causes a character code to be generated and applied to the interface 21. The function and control keys supply function signals to the interface. The communication of the keyboard interface 21 with the bus and other subsystems will be described in more detail later, but briefly this subsystem serves to provide character and function data to the bus 20 and memory 19 and to enable writing of data into the memory by a write control signal on a line 60.

The processor also includes a typewriter simulator subsystem 50 which receives and decodes the signals from the platen knobs 36-37, margin indicator levers 38-39 and vertical space lever 40, and applies this information to the bus 20 for the purposes described earlier. The processor also includes a position indicator and left margin control 51, right margin control 52, and timing control 53. The former serves to direct the positioning of the position indicator 44 as directed by the typist and as seen in FIGS. 3a-3c, and to establish the left margin of the typed text 33 as set by margin indicator lever 38. In similar manner, the right margin control 52 directs the limits of the right margin of the typed text 33 by providing appropriate control signals for causing a bell to ring and keyboard "lock-up" when the typist reaches the margin. The timing control 53 merely serves to provide basic timing, or time-based gating, signals to the system.

The processor further includes a video circuit 55 and character generator 56, supplied with appropriate data by cables 57 and 57a, which in turn serve to cause generation of the appropriate information (i.e., page outline and text) on the display 13. An additional processing subsystem 58 also may be provided for control of other text manipulation functions, such as justification of text, as may be desired.

Description of Simulation and Portions of the System Involved

Page Outline

Turning first to a discussion of display of the page outline 32 and text 33 on the display 13, an example thereof is illustrated in FIGS. 3a through 3c. Both the manner and procedure of generating the page outline, displaying the same, generating text, and editing and manipulation of text, will be discussed in more detail subsequently but an initial discussion is appropriate at this point.

Creation of the page outline 32 is accomplished by displaying horizontal trace segments on the cathode ray tube of the display 13 to form the top and bottom lines of the page. The right and left page outlines are created by a sequence of short vertical line segments. The two types of segments, horizontal and vertical, are contained in a character generator repertoire and displayed at appropriate times during the beam position. The horizontal and vertical symbols are addressed just as alphabetic characters, by appropriately addressing the character generator 56 of FIG. 4. Normally, all characters are displayed sequentially from left to right in lines and from top to bottom. Therefore, a full line of horizontal segments may be displayed to form the top of the page outline. This action is directed by the video circuit 55, during the time corresponding to the line above (i.e., line 0) the first line of text, which supplies the address of the horizontal line segment to the character generator 56 and directs it to display the horizontal line "character" in a sequence one line long. The page bottom is similarly traced out, using a line of the horizontal segments placed just below (i.e., line 59) the last line (i.e., line 58) in which text may be displayed. While drawing the intervening lines (i.e., lines 1-58) of text, the character generator 56 is directed to trace a vertical line immediately to the left of a first possible character position in each line at the end of retrace. This procedure draws the left side of the page outline. The actual first character of text displayed is usually inset a number of spaces, as selected by the setting of the left margin lever 38. The right hand page edge may be created in a similar fashion (i.e., a vertical line segment to the right of the last possible character position in each line), when monospaced text is being displayed, at a predetermined character and space count. In the case of proportionately spaced text, the right page edge may be generated by displaying a vertical line segment when a specific horizontal position or sweep voltage has been reached. The former is a function of system timing only; the latter is a function of beam position.

It can be seen from FIGS. 3a through 3c that the page outline 32, as well as the text 33, moves upwardly on the display 13 as additional lines are typed in the same manner as an actual page being typed. Additionally, the position indicator or cursor 44 is generated by the character generator 56 and moves horizontally on the display as typing occurs to indicate where the next typing event is to take place. The platen knobs control the rows or lines of text displayed, but the undisplayed text below the print line is not lost but remains stored in memory. This lower portion of the page, although the data therefor is applied from the memory via the bus to the display circuits, is blanked so as to prevent display thereof on the upper portion of the display screen. In

this manner the display of data is blanked until the occurrence of the top line of the page outline, and only the material from the top page outline through the print line is displayed.

FIGS. 3a-3c also illustrate the progression of the typed material. FIG. 3b is an illustration of the text after the first three lines have been typed, followed by a carriage return, and a portion of the fourth line ("typewriter, but has") has been typed. In a similar manner, FIG. 3c illustrates how the fourth line is completed, followed by a carriage return, and then the fifth line is typed. The page (i.e., the outline 32 and text 33) may be moved up or down on the display by rotation of either platen knob 36 or 37. The page as seen in FIG. 3c may be, for example, rolled down to the position shown in FIG. 3a for editing purposes, such as for changing the word "standard" in the third line in FIG. 3a to "conventional," by appropriate positioning of the cursor 44 and depression of a function key, and then typing in the new word. For example, one of the keys 28 of the keyboard 12 in FIG. 1 may be an "edit" key such that the cursor 44 can be placed under the first letter of a word to be shifted, and then depression of the edit key will shift down in memory the remainder of the line involved. Other special purpose keys can be provided for editing and manipulation functions. In this example, however, the fourth and fifth lines as seen in FIG. 3c are not lost inasmuch as they have been stored in the dynamic memory, and they may be redisplayed as desired by rotation of a platen knob to raise the page again to the position seen in FIG. 3c in the same manner as an actual typed page. The platen knobs change the displayed position of the page on the CRT, but the contents of the memory 19 are not changed thereby allowing absolute addressing of memory locations regardless of a display position. That is, line one of the text 33 remains line one in memory no matter what physical position line one of text occupies on the display screen. The platen knobs thus reposition, or "roll," the contents of memory up and down on the CRT of the display 13.

Platen Knobs

As explained above, the platen knobs 36 and 37 are used to "roll" the contents of the display screen, that is, the page outline 32 and text 33 thereon, up and down in the same manner that a piece of paper is rolled up and down in a standard typewriter. In this system, the electron beam of the cathode ray tube of the display 13 is assumed to start at the top left-hand corner of the display and sweep to the right, as with conventional operation of cathode ray tubes, but without interlace. After reaching the right end of the display, the video amplifier means associated with the cathode ray tube receive a retrace signal and return the beam to the left of the display and down a predetermined distance below the first sweep, again as is conventionally known. After the vertical amplifier receives a flyback signal, the display beam returns to its starting position.

When the typist rotates one of the platen knobs 36 and 37, which may be physically interconnected, the typewriter simulator 50 of FIGS. 4 and 6 senses the direction of rotation and generates an increment or decrement "row cursor" signal. The increment signal indicates that an additional row or line of text is to be displayed; whereas, the decrement signal indicates that one less line of text is to be displayed. In any event, the cursor remains, and is displayed on, the bottom print

line as seen in FIGS. 3a-3c. For example, if three lines are displayed as shown in FIG. 3a, an increment row cursor signal indicates that an additional line as seen in FIG. 3b is to be displayed. Similarly, a decrement signal occurring with the text as displayed in FIG. 3b causes the display of one less line as shown in FIG. 3a. The number of increment or decrement signals generated depends upon the number of detent positions the platen knobs are moved. In any event, the increment and decrement action raises and lowers the displayed page 32 and text 33 on the display 13.

The increment or decrement row cursor signal is applied through the bus 20 to the position indicator control 51 to change the contents of a row cursor counter. More specifically, the increment or decrement row cursor signal is applied by a line 62 from the typewriter simulator 50 to the bus 20. If it is an increment signal, it is applied to the bus at one time (t_1) during retrace and flyback; and if it is a decrement signal, it is applied at another time (t_2) during retrace and flyback. These increment and decrement signals are available from the bus 20 and are applied on respective lines 63 and 64 to the position indicator and left margin control 51 and serve to increment or decrement a row cursor counter in the control 51 which ultimately causes a greater or fewer number of lines of text from memory to be displayed, as will be explained in more detail in conjunction with a description of FIG. 7.

Thus, every time a platen knob is rotated one detent position in one direction (such as counter-clockwise), the row cursor counter is incremented; and when rotated one detent position in the other direction (clockwise), this counter is decremented. The contents of this counter are compared with a memory row count from a memory address counter. The row cursor count represents the line of text where the typist desires the cursor to be. The cursor position is completely under the control of the typist. She moves the cursor up and down the page by using the platen knobs 36-37 and/or the carriage return key. The memory row count corresponds to the row of text being read out of the memory 19. For example, when the first row of data is being read out of the memory, the memory row count is 1, and when the n^{th} row of memory is being read, the memory row count is n . During usual operation, a system clock drives the memory row counter, and the counter is incremented sequentially from zero to 59 and then back to zero. The row cursor count is constantly compared with the memory row count, and when these counts are equal (which they will be for the bottom displayed line of text) a "row compare" signal goes true and remains true until the memory row count is incremented. As the row compare signal goes false, initiating flyback of the display, the display beam returns to the starting position. In this manner, the row containing the cursor or position indicator 44 is always on the bottom row of the display; that is, the typing line. When the typist rolls a platen knob and increments the row cursor, the row compare signal occurs one memory row count later, thereby initiating flyback one row later which in turn causes an additional line of information to be displayed.

Margin Indicators

Turning now to the margin indicator levers 38 and 39, these enable the margins of the displayed text to be set in the same manner as on a standard typewriter. The typist moves the left lever 38 and the right lever 39 to

the desired locations to set the text margins. When either of these levers is moved, or the system is turned on, binary codes corresponding to the respective margin lever positions are generated within the typewriter simulator 50 and pass through the bus 20 into the respective controls 51 and 52. The margin data and load signal are applied by a cable 68 and a line 69, respectively, to the bus 20, and from there to the controls 51 and 52. The left margin data is applied from the bus 20 by a cable 70 to the control 51, and the right margin data is applied by a cable 71 to the control 52. These margin positions are stored in the respective controls 51 and 52 under direction of "load margin" signals. A load margin signal (originating from line 69) is applied from the bus 20 by a line 72 to the control 51. Similarly, a load signal is applied by a line 73 to the control 52. These load signals indicate that left and/or right margin data is available and enable the same to be loaded into latches or registers of the respective controls 51 and 52. Also, a load signal is generated whenever system power is turned on. Whenever one of the margin levers is moved, the new position is similarly decoded and stored.

The left margin control 51 basically sets the left margin limit of the text 33 appearing on the display. The right margin control principally serves to provide a "ring-bell" signal on a line 72 to cause a bell to ring prior to reaching the right margin, and to provide an "at right margin" signal on a line 75 to indicate that the right margin has been reached. This latter signal is used to cause "lock-up" of the keyboard and to prevent further typing in that row or line, and to inhibit space bar signals to prevent the cursor from being moved past the right margin, unless a margin release key is depressed. For these purposes, the keyboard 12 of FIG. 1 includes a bell and a solenoid associated therewith, both of which are electrically operated. The solenoid is operated and strikes the keyboard housing whenever a key is struck to simulate the noise and action of a typewriter, and upon "lock-up" the solenoid is inhibited so as to simulate the lock-up characteristic of a keyboard and inhibits application of character and function data from the keyboard 12 to the interface 21. Lines 74a and 75b provide the ring bell and inhibit solenoid signals to the keyboard 12.

Additionally, the position indicator and left margin control 51 is involved in providing row and column "compare" signals which indicate comparison between cursor position and memory location. A column cursor count indicates the position of the operator on a particular line; whereas, the row cursor count indicates which line. Each time the operator strikes a key, the character signal is applied from the keyboard 12 through the interface 21 to the bus 20, and the character is entered into the memory 19, and the column cursor count is incremented thereby directing display of the cursor, and thus directing the operator, to the next position on the line. The control 51 provides the row and column compare signals on lines 76 and 77 to the bus 20 to in turn direct the location of the cursor 44 on the display. This is accomplished upon row and column compare by directing the character generator 56 to generate the cursor. The row and column compare signals are "anded" and applied as a "compare" signal by data cables 57 and 57a to the character generator 56. This compare signal is also applied by a line 78 to the keyboard interface 21 for gating functions.

The left hand margin may be defined as the column on the page to which the cursor 44 returns after the typist hits the carriage return button. Each time the operator hits the carriage return button, an increment row signal is applied by a line 80 from the keyboard interface 21 to the bus 20. A load signal then is applied from the bus 20 on a line 81 to the left margin control 51 to transfer the contents of a left margin latch (which is loaded earlier by the load signal on line 72) into a column cursor counter in the control 51. This operation, as will be apparent enters a preset count, which is a function of the set position of the left margin indicator lever 38, into the column cursor counter to indicate the initial column position of the cursor on the line; that is, to cause the cursor to be displayed at the left margin rather than at or near the left edge of the page. An "at left margin" signal also is generated by the control 51 and applied by a line 79, the bus 20, and a line 79a to the keyboard interface 21 to inhibit backspace key signals (i.e., to prevent a backspace signal from moving the cursor to the left) on compare, unless a margin release key is depressed.

The position of the cursor along the line changes as typing progresses as directed by increment and/or decrement column signals on respective lines 83 and 84 from the keyboard interface 21 to the bus 20. These signals are applied from the bus 20 by lines 86 and 87 to the control 51 to increment or decrement the column cursor along the line. The column position of the cursor along the line is changed by hitting an alphanumeric character key (increment), space (increment), backspace (decrement), and by editing keys such as an erase key (decrement) of the group 28 of FIG. 1; whereas, the cursor row position is always the bottom typing line, but this row position with respect to memory row position is changed by depressing the carriage return key or by rotating the platen knobs as explained earlier to display more or fewer lines of text.

Turning again to the right margin control 52, the same keeps track of the character spaces and blank spaces used in typing a line so as to provide the "ring bell" and "at right margin" signals. In an exemplary system, the information in a line of text on the display and stored in a row of memory is 128 characters long. Thus, there are a discrete number of characters per line of display and per line of memory. In order to keep track of the number of characters and spaces used during typing a line of text, the characters and spaces are counted in the right margin control 52. Additionally, in a proportional spacing text system, different characters have different lengths (such as wide and narrow) and, thus, width codes are generated so as to keep track of the number of character spaces used on a line. The width codes also are used by the character generator to control the length of the cursor. The depression of an alphanumeric key on the keyboard 12 sends a coded signal to the memory 19 which in turn uses this coded signal to address a read-only memory to derive the proper code for this character. This latter code includes character width information, and the code is stored in the memory and also is used to address the character generator 56 for display of the character. The width code data is derived from the bus 20 by a cable 66 to indicate the width of characters and spaces, and the width code is applied to a counter within the right margin control 52 and counted to keep track of the number of spaces (character spaces and blank

spaces) being used in typing the line. The count is used to determine when the ring bell and at right margin signals should be generated; in other words, it is used to tell when the right margin is reached. When the counted number is equal to or greater than the right margin count, as determined by the setting of the right margin indicator lever 39, the at right margin signal is generated and is applied to the line 75. The ring bell signal on the line 74 is generated a predetermined number of spaces prior to the at right margin signal. For example, the control 52 can be preset to cause the ring bell signal to occur 15 unit spaces before the at right margin signal is to be generated. More specifically, space and width code data are applied by the cable 66 to an adder within the control 52, along with the previous space count for the line which is accumulated internally. The space count is updated and applied to a counter, this counter keeping track of the total number of spaces as they are used. The output of this counter is then compared with the right margin limit as set by the right margin indicator lever 39 to provide the at right margin signal, and compared with a preset number (such as one indicating 15 spaces) to provide the ring bell signal.

In addition to typing characters, it also is possible to provide an overstrike, e.g., ¢. Another feature, underlining, is accomplished by merely changing the most significant bit in the character code. When it is desired to provide an overstrike, the cursor is placed under a character (c) by means of the space bar or backspace key (and with the platen knobs if the character to be overstruck is on a line other than the then current typing line), and the second character (/) is typed over the first character (c) to provide the overstrike. This is accomplished by sensing the fact that the cursor is under an existing character and directing the display of the two characters, one over the other. Also, a code (overstrike or backspace code) is stored in the memory to indicate the existence of the overstrike.

More specifically, a character indicator line 88 provides a signal from the bus 20 to the keyboard interface 21 to indicate that a character is on-line at a particular time. This character indicator signal, along with the "compare" signal on the line 78, indicates that the cursor is under a character, and is used by the keyboard interface 21 upon depression of a character key to initiate an over-strike sequence or program. The purpose of this sequence is to generate a backspace code which is stored in a memory location in-between the two characters involved (i.e., c, backspace code, /) to indicate that the two characters are to be displayed one over the other. This code also directs the printer to make the overstrike upon print out and, of course, the overstrike code itself is not printed or displayed. Furthermore, during editing functions, the code likewise indicates the existence of the overstrike; for example, when it is desired to erase the overstrike upon depression of an erase key, it will be apparent that all three (two characters and backspace code) should be erased. This is accomplished by placing the cursor under the overstrike on the display and depressing the erase key. The first character is erased from the memory location, and the cursor is incremented thereby giving backspace/cursor compare which in turn indicates that both the backspace code and the second character likewise are to be deleted from the memory.

Vertical Space Lever

The vertical or line spacing control lever 40 serves the same function as the equivalent lever on a standard typewriter. The typist sets the lever 40 for the desired number of spaces (such as single, double, or triple space), and the displayed text is spaced accordingly, just as in regular typing. The typewriter simulator 50 generates a binary space code representing the position of the switch 40, and this code is applied by a cable 90 to the bus 20, and from the bus through a cable 91 to the keyboard interface 21 and is stored therein. When the typist strikes the carriage return key, one, two, or three carriage returns are generated by the keyboard interface 21 which are applied by the cable 47a to the bus 20. These coded signals are applied by the bus 20 through data cable 94 to the memory 19 and are stored therein. The resulting stored carriage returns are used to increment the printer 15 upon printing out so as to obtain the proper number of spaces between lines during printing.

Memory

The memory 19 includes a main memory 98 communicating through a buffer 99 with an input/output buffer 100. A "width" read-only memory (ROM) 101 communicates with the buffers 99 and 100. The ROM 101 is addressed by the character code, and serves as a table look-up to determine the width of characters. It provides a two bit output signal to indicate several (such as five) different character widths. This width information is supplied, along with the character code, through the cable 94 to the bus 20 for purposes as described previously (e.g., keeping track of the number of spaces used in typing a line and for determining the length of the cursor).

The memory 98 is organized to provide sixty rows (corresponding to display lines 1₀ through 1₅₉), each row including 1024 bits broken down into 128 characters, each eight bits long. The memory address count (MAC) directs which character is being read in or out of a given line or row of the memory 98, and a memory row count addresses the rows of memory. As noted earlier, coincidence (compare) of a cursor position (row and column) with the MAC count determines the precise location in the memory 98 in which a character is to be stored. Each row of memory corresponds to a text line on the page 32, but the first and last rows not used for text display. Lines of information are each held serially in respective memory rows, and the information in each memory row is recirculated. The characters stored therein are periodically applied such as approximately 35 times per second, to the character generator 56 for continuous display of the stored text.

In typical prior memory display systems, there is direct correspondence between a memory character position and a display character position, and an entire page of text is stored in a one line recirculating memory. Although in the present system, memory rows correspond to displayed lines, character positions within a line of the display do not necessarily correspond with locations in a row of memory because of the inclusion of coded signals, or non-displayable characters such as the backspace code for overstrikes and carriage return codes noted previously, along with the text. The present arrangement allows absolute addressing of the memory even as the displayed lines of text are changed (rolled up or down, as by platen knob motion). That is, a given line on the display (e.g., line 5) always corresponds with a given row (e.g., row 5) of memory no

matter where this particular line may physically appear on the display. This means that by merely changing the cursor row count (with the platen knobs or on carriage return as noted earlier), the display position of the text on the display can be changed, but this change does not change the contents of the main memory 98.

Row compare (line 76 from control 51) occurs when the memory row address and cursor row count are equal. This signal stays true until the next row of memory is addressed (at which time there is no longer comparison between cursor and memory rows). When the row compare signal goes false, flyback of the display 13 is triggered, and the electron beam of the cathode ray tube returns to the top of the screen. If the row cursor count is incremented, for example, with the platen knobs, the row compare signal occurs one line later, thereby giving flyback one line later, thereby causing display of an additional line of text. As a specific example, assume that line 3 of the text is being displayed as shown in FIG. 3a. Line 4, and subsequent lines, are being read from the main memory 98 and applied, via the character generator 56, to the display 13 for display at the top of the display screen; however, all lines (line 4, etc.) after the typing line, although supplied to the display, are blanked or inhibited by the video circuit 55 to prevent display thereof. Now, upon increment of the row cursor counter by rotating the platen knobs one position, the row compare signal occurs, as explained above, one row later thereby causing the fourth line of text, as partially shown in FIG. 3b, to be displayed. Flyback occurs at the end of the fourth line, and now the fifth and subsequent lines of the text are supplied from the memory to the top of the display but are blanked as noted earlier.

A similar operation occurs upon decrement. In this case, the row compare signal occurs a line earlier thus causing flyback to occur at the end of the earlier line. Again, as a specific example, assume that line 5 of text is being displayed as illustrated in FIG. 3c. Opposite rotation of a platen knob decrements the row cursor count and, thus, row compare (and flyback, at the end of the line) occurs one line earlier, at line 4.

As should be apparent, by merely changing one count, that is the cursor row count, either up or down, the position of the text on the display screen can be changed. This enables the text 33 and page outline 32 to be simply rolled up or down on the display screen, and the arrangement used and described above still allow absolute addressing to the memory without shifting up and down the contents of memory, which would require a change in memory addressing.

Timing Control

The timing control 53 supplies the various timing signals to the overall system. It includes a system clock for providing the timing signals. The basic systems time is a t time, supplied by a line 107, as will become more apparent upon a consideration of the detailed block diagrams of the subsystems shown in FIGS. 5 through 8. Each t time is broken down into eight "delta t " times supplied on a line 108 for memory control purposes (i.e., to serially shift character bits), and eight t times equals one MAC count or time. The MAC counts are provided by a line 109 and memory row counts (which occur after a predetermined number of MAC counts) are supplied by a line 110. Each MAC count corresponds to a character position in memory as noted earlier, and 128 MAC counts or times is the display time

for one entire line on the display. At the end of the display time, retrace (which is ten MAC counts long) occurs which increments the row cursor counter and causes addressing of the next memory row. The retrace signal is provided by a line 111. 60 retraces plus one flyback time (which is 128 MAC counts constitutes the time of display of a complete page of text which, as noted earlier, is repeated approximately 30 times per second. The flyback signal is provided by a line 112. A cable 114 supplies control signals (such as compare) used in generating flyback and retrace signals.

The timing allows, as will be apparent to those skilled in the art, time sharing of the communications bus 20. Thus, the data bus can be time shared with a large number of signals, with each class of information being transferred over the bus during an allocated time. The time is divided into three major time frames, display time, retrace time, and flyback time. The display time is subdivided into the eight t time slots, $t_0 - t_7$. This allows data to be placed on the bus at an early time slot and read therefrom as needed during a later time slot. The bus 20 preferably can be broken down into several separate buses; such as, a data bus, control and indicator lines, address bus, timing bus and voltage distribution lines. The data bus transmits character data and width data; the control and indicator lines transmit control signals, such as row compare, column compare, carriage return, and so forth; the address bus is used in addressing for transfer of information from subsystem to subsystem principally for editing and text manipulation operations; the timing bus transfers timing signals such as strobe, flyback and retrace; and the voltage lines supply the various necessary voltages throughout the system. The first three operate synchronously and the last two asynchronously.

Video and Character Generator

The video control circuit 55 serves to latch or store character data from the bus 20 on each memory address count (MAC) for subsequent display by the display 13. It also controls sweep of the display 13. The character generator 56 receives digital codes representative of characters and directs the display of the characters and cursor on the CRT of the display 13. The display may be in the form of a dot matrix; however, it is preferable that the character generator 56 generate analog signals for painting or stroking the characters on the display. The character width code is used by the video control circuit 55 to control the width of the character by controlling the horizontal sweep of the display. Additionally, the video control 55 controls beam sweep for overstrikes as noted earlier; that is, the electron beam, after painting one character (c), is caused to sweep to paint the second character (/) over the first (c). The video control 55 also controls generation of the page outline, and blanks or unblanks the display so as to display only the appropriate portions of the text received from the memory. The video control thus causes display of text information and page outline from line zero until flyback, and blanks everything else. The display also is blanked by the circuit 55 upon receipt of a nondisplayable character, such as the backspace code and carriage return code. The video circuit can cause a character to blink by displaying the same less frequently on command if desired.

The video circuit responds to carriage return to trace right vertical segments to create the right-hand page edge as noted earlier, and to provide the proper space

between the lines of text. The horizontal and vertical segments are represented by codes in the character generator 56. The top line of the page outline is created by generating at line 1_0 (which is always the first line in memory and is the top line of the display) a sequence of the horizontal line segments. The left page outline is displayed just to the left of the first possible character position at column zero by generating vertical segments as noted earlier. The right outline in a monospaced typing system may be generated at a particular column position, but in a proportional spaced system in analog signal controlling the horizontal sweep amplifier for the cathode ray tube is sampled and compared with a fixed reference voltage to trigger the character generator to generate a right-hand vertical segment upon the occurrence of a particular voltage level. In this manner, the right-hand line is displayed at a constant position to create the right page outline. The bottom line of the page is traced at line 1_{59} , and flyback occurs after final retrace.

FURTHER DETAILED DESCRIPTION OF THE SYSTEM AND SIMULATION

While it is believed that the foregoing discussion provides a comprehensive understanding of both the operation of the system and the various simulation features thereof, there is set forth below a more detailed discussion of the components and operation of the various subsystems of the present system. These are illustrated in more detail in FIGS. 5 through 8.

Keyboard Interface

Turning first to FIG. 5, there is illustrated in detail the keyboard interface 21 along with its interconnections between the keyboard 12 and communications bus 20. The keyboard interface 21, as noted earlier, essentially receives the information and data from the keyboard 12 and properly supplies data to the bus 20.

The keyboard 12 includes the various alphanumeric keys and function and control keys. The keyboard 12 may be a keyboard assembly from Clare Pendar Co. of Idaho, and each key operates a switch. When one of these switches is closed, signals are generated which in turn cause the application of coded signals and control signals from the keyboard 12 to the keyboard interface 21. If an alphanumeric character key is depressed, for example, a seven bit code for that particular character is applied on seven data lines of the asynchronous character data cable 47 to a character control and time sequencer 126. As an example, the character "A" may have a code, 0001100, according to standard USASCII encoding. Also, when the key is depressed a main strobe signal is supplied on the line 49 from the keyboard to indicate that the character data is present for storage in, or other use by, the keyboard interface 21. This strobe signal is applied to a main strobe control 127 which in turn controls gating of data into the control 126. The strobe control 127 also energizes the solenoid to provide a sound simulating the operation of a typewriter. The space, backspace and tab keys may generate separate control signals on lines within the data cable 47. The carriage return key likewise may generate a separate signal or a particular code. In any event, the various character and control keys of the keyboard 12 generate signals or coded signals which are suitably latched into the keyboard interface 21 upon command by the main strobe, and are ultimately applied to the bus 20.

Additionally, various types of function keys may be provided as noted earlier. Depression of any of the function keys likewise closes a switch to cause the application of asynchronous function data by cable 48 to a function control and time sequencer 128 in the keyboard interface 21. Also, when one of these function keys is depressed, a function strobe signal is applied by line 49a to a function strobe control 129 which in turn indicates to the control 128 that the function data is available. Although not shown, a function strobe signal may be applied to the solenoid for the same reason as the main strobe.

Most of the character, control and function keys involve single switch closures. However, certain keys may provide double closures (i.e., another switch closure upon full depression of the key) to indicate that the particular event should be repeated, as is the case with a conventional typewriter. For example, full depression of the space bar closes a second switch, indicating that the space bar is fully depressed, to cause a space signal and main strobe signal to be repetitively supplied as long as the space bar is fully depressed; whereas, partial depression causes only one of each of these signals to occur. The same may be true of the backspace key. Certain function keys may also be double depression keys, such as character add, character delete and erase which in turn cause a particular function control signal and function strobe signal to be repetitively applied from the keyboard 12 to the keyboard interface.

The keyboard interface 21 supplies the character and function data to the bus 20 at appropriate times. The "compare" line 78 from the bus 20 is connected to an "and" gate 131, the output of which is connected to an input of the character control 126. Suitable line supplying timing signals from the timing control 53 in FIG. 4 are used throughout the system, but these are not shown in most cases so as to prevent unduly cluttering of the drawings. The character data is applied from the control 126 by data cable 47a upon coincidence of "compare" and timing signal t_3 , and a write control signal is applied on the line 60 to cause the particular character to be written into the correct location in memory. These operations occur unless the "at right margin" signal is present which in turn causes the "and" gate 131 to inhibit the write control and increment column signals. The control 126 also generates increment column signals on an output line 83, and an increment row upon carriage return signal on an output line 80 which are used by the position indicator and left margin control 51 of FIG. 4 as described earlier. The control 126 receives the spacing code by cable 21, and this code is used upon carriage return to supply the proper carriage return code to the memory to indicate the number of spaces to be provided between lines as set by the space lever 40 in FIG. 4.

Inasmuch as experienced typists are capable of high speed bursts of typing, such as when typing the word "the," the control 126 may include a small (e.g., four characters in length) buffer memory to hold bursts of characters during fast typing. The memory then supplies the character data serially to the bus 20 at appropriate times.

The function control and time sequencer 128 operates in a manner similar to the control 126 to supply function data by cable 48a to the bus 20 at appropriate times. It also generates the decrement column signals

on the line 84 and increment column signals on the line 83b for purposes as described previously. As with the control 126, the transfer of function data to the bus occurs upon coincidence of the "compare" signal on line 78 and particular t time for the various functions involved. The compare signal is applied by the line 78 to an "and" gate 132, the output of which is connected to the control 128. The "at left margin" signal is applied by the line 79a to the gate 132, and upon coincidence of this signal and "compare," backspacing is inhibited. Similarly, an "and" gate 134 is responsive to "compare" and "at right margin" signals to inhibit entry of spaces into the text. The increment column signals from both the controls 126 and 128 are applied by lines 83a and 83b to an "or" gate 135 which in turn supplies increment column signals on line 83 to the bus 20. The outputs of the "and" gates 131, 132 and 134 also are applied to the input of an "or" gate 136 to inhibit operation of the keyboard solenoid, as well as the application of main strobe signals to the character control 126, upon the occurrence of any of the inhibit outputs from these gates.

The keyboard interface 21 also controls strike-overs by means of a strike-over control 138. As noted earlier, the codes for two characters which are to be displayed as an overstrike (e.g. ϕ) are stored and separated by a backspace code (e.g., c, backspace code, /). A "character indicator" signal on a line 88 from the bus 20 indicates that a character is on the bus (the first character of the overstrike in this instance). An "and" gate 139 receives the character indicator signal, the compare signal from line 78 and the main strobe signal from line 49 to provide an initiate signal to the strike-over control 138. The character indicator signal 88 is provided by examining whether a character is on the data bus portion of the communication bus 20 at time t_1 , and if so, the character indicator signal is applied on the line 88 at time t_2 . If the compare signal also is present at time t_2 , the cursor is under a character. This information, along with depression of a key (for the second character) as indicated by the main strobe signal from the "and" gate 139 to set a flip-flop within the strike-over control 138 to initiate an overstrike program or routine as noted previously. The strike-over control 138 provides signals to the character control 126 and to the function control 128 for gating the respective character and control data to the bus 20.

As should be apparent, each character code or column storage position in memory stores information for a single character, space or code. Inasmuch as an overstrike involves three items of information (that is, the first character, backspace code, and second character) a sequence of operations occurs to cause the first character to be stored in a first position, the backspace code to be stored in the next succeeding position, and the second character to be stored in the third position even though the cursor is under the first character on the display and the second character is then entered by the keyboard over the first character. An exemplary overstrike routine may include essentially the following steps which are directed by the control 138 after the first character has been typed: store the second character in the next succeeding memory location and increment the cursor; shift this character to the following location (to open a storage location for the backspace code); and store the backspace code in the location between the first and second characters. More precisely,

an exemplary overstrike routine may include the following steps, assuming the first character has already been entered in memory, and the character indicator, compare and main strobe (second character has been struck on the keyboard) signal inputs to the "and" gate 139 are true, thereby initiating the routine which is directed by the control 138: (1) a character add occurs to open one character space (eight bits) in memory inasmuch as both the first and second characters cannot be stored in the same memory location; (2) the character control 126 is activated by the control 138 to supply data to the bus and causes the second character to be stored in the opened memory location, and the cursor is incremented; and (3) a second character add occurs to open a space between the first and second character positions, and the backspace code is stored in this location, and the cursor is incremented twice to cause it to appear after the displayed overstrike.

The routine may be expanded where provision for justifying text are desired by including the additional step of comparing the width codes for the two characters, and then causing the longer width character to be stored before the backspace code. In this case, the width codes for the characters are supplied (not shown) to the strike-over control 138 from the width ROM of memory 19 to enable the width comparison to be made. Storage of the longer width character first, if it is the second character struck, is accomplished by providing auxiliary register means in the memory 19 which is activated by the overstrike routine to allow the second or successive character to be stored and then shifted into the earlier location, and to allow the first character to be shifted to the subsequent location which follows the backspace code, giving the result—widest character, backspace code, narrowest character.

The strike-over program thus causes storage in sequence of the first character, backspace code, and second character. As a specific example of the information stored, and read-out thereof, assume that the text for display and printing stored is: ABØD. At t_1 and a MAC count of zero (i.e., t_1 : MAC 0), the "A" character is "on line" (i.e., available on the bus 20) and is latched into the video circuit 55 at this time. The character is generated by the character generator 56 and painted by the CRT display 13 at a subsequent time, such as during approximately times t_2 to t_3 . Similarly, the character "B" is on the line at t_1 : MAC 1 and is subsequently painted as with the character "A." In a like manner, the character "C" is latched into the video circuit at t_1 : MAC 2 and painted at approximately the time period t_4 through t_5 . At time t_1 : MAC 3 the backspace code is available from the memory. This code indicates that the display should not paint a character, but should go back the unit space count equal to that of the character "C," and the CRT beam sweeps back. At time t_1 : MAC 4, the slant line (/), is latched into the video circuit 55 and painted at approximately t_6 through t_7 . The character "D" is similarly latched at t_1 : MAC 5, and then painted at t_8 through t_9 . The three successive character locations in memory for the strike-over thus have stored therein the respective codes for "C," the backspace code, and "/." In the event the erase function key is depressed, for example, the first character (i.e., C) is detected and deleted; then, by detecting backspace code and cursor compare, both the backspace code and the second character (i.e., /) are deleted.

Typewriter Simulator

Turning now to a more detailed discussion of the typewriter simulator 50, the same supplies to the bus 20 the increment and decrement row cursor signals on line 62, margin data and margin load on respective cable 68 and line 69, and the space code on the cable 90. The platen knobs 36 and 37 are interconnected and operate rotary switches to provide count signals for each detent position thereof. These count signals are sensed by a rotation sense circuit 150 to provide an output pulse for each change in detent position on lines 151 and 152 to control the respective up and down movement of the text on the display.

In an exemplary system, a two deck 24 position rotary switch and a Gray code may be used to provide a repeating cycle of four binary codes. These codes are detected by the circuit 150 to both provide output pulses per detent position and to indicate the direction of rotation. The use of this type of coding, as will be apparent to those skilled in the art, provide sets of unique numbers which will indicate either direction of rotation of the platen knobs, starting from any rest position. Thus, if the platen knobs are rotated counterclockwise, for example, a series of pulses are applied on the line 151 for each detent position movement of the knobs. Similarly, if the same are rotated clockwise, a number of pulses are applied on the line 152. If the number of positions rotated in one direction equals the number rotated in another direction, no pulses are supplied on the lines 151 and 152. These lines are connected to respective "and" gates 154 and 155 to provide the respective increment and decrement row cursor outputs on the line 62 at appropriate times. For example, the increment pulses from the line 151, if they exist, will be gated by the gate 154 at coincidence of retrace and fly-back and MAC 9 and t_4 . The decrement pulses are similarly gated, but at t_5 . The row cursor increment and decrement signals are supplied by the line 62 to the bus 20 to increment or decrement the row cursor as explained previously.

The margin indicator levers 38 and 39 of FIG. 6 operate a pair of multi-contact elements which slide on a code bar which is illustrated in detail in FIG. 9. For example, the lever 38 includes seven electrically interconnected contacts mounted side-by-side to slide along a ground bar 160 and respective conductive coded segments 161 through 166 as seen in FIG. 9. The areas between the segments are non-conductive but the segments of each respective row are connected together. The position of the lever 38 will cause one or more output terminals 171 through 176 to either provide a given output voltage, or to be grounded to the ground bar 160. In this manner, a six-bit code is applied from the left-hand terminals of the bar in FIG. 9 by a cable 178 in the typewriter simulator 50 in FIG. 6 to a multiplexer 180. In a similar manner, the position of the right-hand lever 39 provides an seven-bit code from the right-hand output terminals of the bar in FIG. 9 through a cable 182 to the multiplexer 180 of FIG. 6. The multiplexer is used to enable left margin data to be applied to the bus at one time and the right margin data to be applied to the bus at another time. The left and right margin data is applied by the multiplexer 180 through an input-output buffer 84 and the cable 68 to the bus 20, and latched into respective registers in the left margin control 51 and right margin control 52. The left least significant bit (LSG) and right least significant bit of the re-

spective margin codes are applied by lines 186 and 187 to a change of state detector 188. Any motion of a lever is detected by detecting the least significant bit inasmuch as the LSB signal changes with each position change, the lever position in coded form is applied to the multiplexer. The detector 188 provides left and right margin signals on respective lines 190 and 191, and an enable signal, to "and" gates 193 and 194. Whenever either margin indicator lever 38 or 39 is moved, or power to the system is turned on, the enable output of the detector 188 goes true to cause one or both of the "and" gates 193 and 194 to supply enabling signals to the multiplexer 180 at appropriate times (such as MAC 1 for the left margin, and MAC 2 for the right margin as indicated in FIG. 6) to gate the margin data from the multiplexer 180. Additionally, the outputs from these "and" gates 193 and 194 are applied through an "or" gate 196 to supply the margin load signal on the line 69 to the bus.

The multiplexer, 180, as will be apparent to those skilled in the art, receives and transfers the left and right margin data to the I/O buffer 184 depending upon which margin lever position has been changed. The margin load signal on the line 69 indicates that the margin data is available, and is a load command signal for registers in the left margin control 51 and right margin control 52. It is preferred that the left margin position directly correspond with a memory column address (and at several number spaces from the left hand edge of the page outline 32) so that the left margin position code can be treated as a memory address. In a monospaced text system, the right margin may be treated the same; however, in a proportional spaced text system the right margin is treated as a particular unit space count or distance from the right-hand edge of the page outline 32 as explained earlier. Knowing the number of available character spaces per memory row and the right margin code selected by the lever 39 enables lock-up to occur, and bell ring to occur a predetermined number of spaces before lock-up.

The space lever 40 as discussed previously, indicates the number of spaces to appear between lines upon carriage return in the same manner as the space lever on a conventional typewriter. The space lever 40 is a switch and is connected to a space code sequencer 200 which in turn supplies the selected space code at an appropriate time (retrace and flyback and MAC 0 and t_1) by cable 90 to the bus 20. The position of the lever 40 determines which of the three codes is generated and applied to the bus, the codes indicating one, two, or three carriage returns to be stored. This code is applied by the bus 20 through a cable 91 to the keyboard interface 21 as explained earlier in connection with the discussion of FIG. 5. When the carriage return key is depressed on the keyboard 12, the keyboard interface 21 provides the proper carriage return code (for one, two or three line spaces) back to the bus 20 for entry into the memory 19. This code is supplied by the memory to both the video circuit 55 and the printer 15 to cause proper line spacing on the display 13 and on the hard copy output of the printer. Alternatively, in order to simplify the system the space lever 40 may only include two positions (1 and 2). In this manner only two output signals (for 1 space and for 2 spaces) are available and are used to cause either a single carriage return code or a double carriage return code to be stored, without requiring the generation of three carriage return codes.

Position Indicator and Left Margin Control

Turning now to the position indicator and left margin control 51, the left margin data code from the multiplexer 180 of the typewriter simulator 50 in FIG. 6 is applied by the cable 60 to a left margin position register 210. The load signal is applied by the line 72 to the register 210 to load the left margin code into the register. The left margin data is entered into positions 2 through 7 of the register 210, and bit position 0 of this register is zero. This essentially multiplies the margin count by two to cause each margin position to be doubled and thus be essentially equivalent to four unit spaces.

The contents of the register 210 are loaded into a column cursor counter 211 upon receipt of a load command on a line 81, which occurs upon carriage return or upon command. The contents of the counter 211 are incremented or decremented as a result of typing characters, depressing the space bar, backspacing and so forth, to keep track of the text position (column) within the typing line. The column cursor count in the counter 211 is compared in a column compare circuit 213 with character data (i.e., compared with every MAC count) and during a portion or "window" t_{po} of time t_0 of display time the column compare signal is generated on the line 77 from an RS flip-flop 214. The column cursor count is thus continuously compared with the memory address, and there is, of course, column comparison at least once per text row. The flip-flop 214 is reset during time t_7 of display time. In a similar manner, the memory address count which is available at time t_0 is compared by a compare circuit 216 with the left margin position to provide a signal to an "and" gate 217 which in turn provides the "at left margin" signal on lines 79 at display time t_0 . There is a direct correspondence between memory count and the left margin code, but not for the right margin code in the case of proportional spaced text.

The control 51 also includes a row cursor counter 220 which is incremented and decremented by signals on input lines 63 and 64 to keep track of the row of text being operated upon. A row compare circuit 221 compares the contents of the counter 220 with the memory row address to provide the row compare signal on line 76 from a type D flip-flop 222 at MAC 2 and t_1 and retrace. When the carriage return key is depressed or platen knobs turned one detent position, for example, the row cursor counter 220 is incremented by one to display the next line of text (or a blank line if no information has been typed on the next line). The contents of the left margin register 210 are again gated into the column cursor counter 211 upon carriage return. This causes the cursor to appear at the left margin position as selected by the left margin indicator lever 38 and as indicated by the contents of the left margin register 210 and at left margin signal on line 79.

Right Margin Control

Turning now to the right margin control 52 shown in FIG. 8 which is designed for proportional spaced text, the right margin data from the typewriter simulator 50 in FIG. 6 is gated into a right margin latch 230 whenever the right margin lever 39 is moved or power is turned on as indicated by the load command on line 73. The right margin code includes bits 1 through 7 which are loaded into positions 3 through 9 of latch 230 and zeroes are entered in positions 1 and 2 to multiply the code by four. The right margin code is then available in the latch 230, but is not interpreted as a memory ad-

dress count (in a proportional spaced system) as is the left margin code, but is interpreted as a unit space count. Thus, in order to determine when the right margin is approached, it is necessary to count the number of spaces (blank spaces and character spaces) consumed on a given line. The space data and width code as noted earlier are applied by a cable 66 to an adder 232. At time t_1 , the sum of the character or space data and width code are stored in a latch 233, and at time t_2 the contents of the latch are applied by a cable 235 back to the adder 232. In this manner, at the next time t_1 , the width codes of the previous characters and new character (or spaces or combination of characters and spaces) are added together. This new sum is applied to the counter 234 periodically so that this counter contains the unit space count of the line already consumed. In order to provide the ring bell signal on the output line 74, the output of the counter 234 is added with a present count (such as 15 unit spaces) in an adder 238, and the contents of this adder are compared in a first compare circuit 239 with the contents of the right margin latch 230. The compare circuit 239 provides the ring bell signal if the contents of the adder 238 equal the contents of the latch 230 or if the contents of the adder 238 are greater. In a similar manner, the line space count from the counter 234 is compared in a second compare circuit 242 with the contents of the right margin latch 230 to provide the at right margin signal on line 75. This signal occurs if the line space count is equal to or greater than the content of the right margin latch 230.

In a proportional space system which also enables justification of the text, additional logic can be provided to keep track of the space that the typing line can be expanded and the space that the line can be contracted. This can be accomplished through the use of a pair of counters along with a respective adder and subtractor to add and subtract these numbers from the line space count before the first and second comparison as described in connection with FIG. 8.

It will be appreciated that although particular logic circuitry, sequences of operation, and timing have been discussed and described, others may be used. Thus, for example, although data or signals have been described as applied to the bus or a subsystem at a particular time, such may be applied at other times, it only being necessary that the particular data or control signal be available for use at an appropriate time or times. Furthermore, the present embodiments of this invention are to be considered in all respects as illustrative and not restrictive, the scope of the invention being indicated by the appended claims rather than by the foregoing description, and all changes which come within the meaning and range of equivalency of the claims therefore are intended to be embraced therein.

What is claimed is:

1. In an electronic text display system, a console for simulating a typewriter comprising,
 - keyboard means, said keyboard means having a cluster of alphanumeric and control keys similar to those of a conventional typewriter, and having a plurality of function control keys,
 - visual display means for displaying a page outline and text thereon as entered by said keyboard means, said page outline and text being movable up and down on said display means, and

platen knob means for causing movement of said page outline and text on said display means, said platen knob means including at least a knob member similar to that of a conventional typewriter and which is rotatable in a first and second direction and switch means coupled to said knob member for generating increment signals when said knob member is rotated in said first direction and decrement signals when said knob member is rotated in said second direction, said system further including control means for controlling the portion of said page outline and text thereon being displayed on said display means in response to rotation of said knob member, said control means including means for maintaining a row count which is representative of the portion of said page outline and text thereon being displayed on said display means, means for incrementing and decrementing said row count in response to said increment and decrement signals, respectively, and means responsive to the increment and decrement of said row count for causing said display means to display a greater or lesser portion, respectively, of said page outline and text thereon.

2. A system as in claim 1 including

manually settable margin means for controlling the right and left limits of text display within said page outline, said margin means including margin levers settable by an operator, the settings of said levers causing margin code data to be supplied to said control means and said control means to control the left limit of text display and to provide signals inhibiting further text entry by said keyboard means when a line of text reaches a right margin limit.

3. A system as in claim 2 including

space control means settable to determine the number of intermediate lines between lines of text, said space control means providing signals to said control means for use in combination with a carriage return signal entered by said keyboard means for storage of signals which indicate and control the number of blank lines intermediate lines of text on said display means.

4. In an electronic text display system, a console for simulating a typewriter comprising,

keyboard means, said keyboard means having a cluster of alphanumeric and control keys similar to those of a conventional typewriter, and having a plurality of functional control keys,

visual display means for displaying a page outline and text thereon as entered by said keyboard means, and

manually settable margin means for controlling the right and left limits of text display within said page outline on said display means, said margin means including right and left margin lever means settable by an operator and binary encoder means responsive to the positions of said right and left margin lever means for generating right and left margin binary code signals, respectively, said system further including control means responsive to said left margin code signal for controlling the left limit of text display and responsive to said right margin control signal to provide signals inhibiting further text entry by said keyboard means when a line of text reaches a position of said display means corre-

25

spending to the setting of said right margin lever means.
5. A system as in claim 4 including
space control means settable to determine the number of intermediate lines between lines of text, said
space control means providing signals to said con-

26

trol means for use in combination with a carriage return signal entered by said keyboard means for storage of signals which indicate and control the number of blank lines intermediate lines of text on said display means.

* * * * *

10

15

20

25

30

35

40

45

50

55

60

65