PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 3 : G09G 3/16

A1

(11) International Publication Number: WO 83/ 01333

(43) International Publication Date: 14 April 1983 (14.04.83)

(21) International Application Number: PCT/US82/01444

(22) International Filing Date: 6 October 1982 (06.10.82)

(31) Priority Application Number: 309,039

(32) Priority Date: 6 October 1981 (06.10.81)

(33) Priority Country: US

(71) Applicant: GULTON INDUSTRIES, INC. [US/US]; 212 Durham Avenue, Metuchen, NJ 08840 (US).

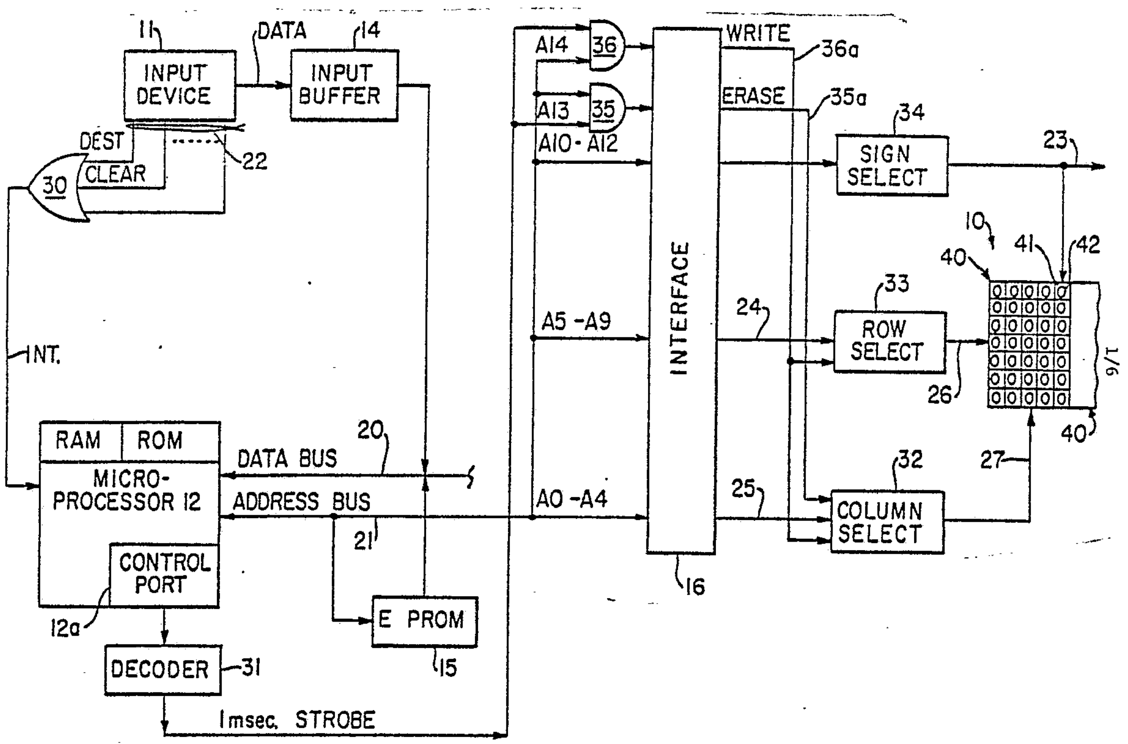
(72) Inventors: RECTOR, Robert, Edwin ; 333 Melrose 6A, Richardson, TX 75080 (US). YARBERRY, Scott, Houston ; 3813 McKinley Street, Plano, TX 75023 (US). GRASSI, Richard, Joseph ; Route 1, Box 135 W, Royse City, TX 75089 (US).

(74) Agents: RELSON, Morris et al.; Darby & Darby, 405 Lexington Avenue, New York, NY 10174 (US).

(81) Designated States: AT (European patent), AU, BE (European patent), CH (European patent), DE (European patent), FR (European patent), GB (European patent), JP, LU (European patent), NL (European patent), SE (European patent).

Published
With international search report.

(54) Title: MAGNETIC DISPLAY ENERGIZATION SYSTEM AND METHOD HAVING IMAGE DATA COMPARISON



(57) Abstract

A control system changes a first image displayed on a display module (40) to a second image. The display module comprises a matrix of display elements (42) each of which has a light and a dark state. The system includes means for storing data defining the first image and means for selecting data defining the second image. The first image data and the second image data is compared to determine which display elements must change their state to display the second image.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	LI	Liechtenstein
AU	Australia	LK	Sri Lanka
BE	Belgium	LU	Luxembourg
BR	Brazil	MC	Monaco
CF	Central African Republic	MG	Madagascar
CG	Congo	MR	Mauritania
CH	Switzerland	MW	Malawi
CM	Cameroon	NL	Netherlands
DE	Germany, Federal Republic of	NO	Norway
DK	Denmark	RO	Romania
FI	Finland	SE	Sweden
FR	France	SN	Senegal
GA	Gabon	SU	Soviet Union
GB	United Kingdom	TD	Chad
HU	Hungary	TG	Togo
JP	Japan	US	United States of America
KP	Democratic People's Republic of Korea		

MAGNETIC DISPLAY ENERGIZATION SYSTEM AND METHOD HAVING IMAGE DATA COMPARISON

Background of the Invention

This invention relates to display system controls and more particularly to a system for controlling the changes in the characters and symbols displayed by a display system of the type utilizing magnetically actuated elements.

Display systems utilizing magnetically actuatable display elements have been in use for a number of years. Each of the display elements used in such systems includes a "dot" which is, for example, a disc or sphere having one dark-colored hemisphere or surface and an opposing light-colored hemisphere or surface. The disc or sphere may have a permanent magnet secured to it and be rotatably mounted in close proximity to an inductive element. A pulse of current to the inductive element creates magnetism which causes the dot to rotate to a position at which its desired light or dark surface faces the viewer. Typically, a plurality of display elements are mounted and arranged in a module in the form of a matrix (e.g., five columns by seven rows). By causing a predetermined set of dots to have their light-colored faces visible, a symbol or character may be displayed by the module. A group of these modules may be arranged in one or more rows to form a display board, and a complete sign may comprise a number of display boards. Display systems such as those described above have a number of advantages which make them particularly useful in applications such as displaying the destination of public transportation vehicles. One advantage of the display in such applications is that it produces indications which are highly visible even in daylight. As a further feature, this type of display consumes power only when the symbols displayed are being changed and retains its display without requiring application of power.

A number of systems for controlling the displaying of messages by the above apparatus have been devised. One such system is particularly adapted for use on moving vehicles such as buses. It allows a message (e.g., the origination point, the destination and intermediate stops) to be displayed which is longer than the display sign. This is accomplished by sequentially displaying portions or lines of the message one after the other, where the number of characters in each portion or line is less than or equal to the number of modules in the sign. After each line of the message is set and displayed for a predetermined time, that line is replaced with the next line of the message. After the last line of the message is displayed for a predetermined time, the first line is again displayed, and this cycle may be repeated as often and frequently as desired.

In one currently available display system each message line is caused to be displayed by "writing" one character after another until the entire line is set and can be read. Each character is written by setting the elements (e.g. dots) of the corresponding display module row by row to the required dark or light-colored faces, until all seven rows of display elements are set to their required new positions. In this operation, five elements are set simultaneously. In another form, each character can be set a column at a time, with seven elements set simultaneously. This mode of changing the display is quite rapid, but uses five row driver circuits (or seven column driver circuits) to simultaneously flip up to five (or seven) display elements. This simultaneous manner of flipping the display elements has the disadvantage of consuming a relatively high amount of power.

U.S. Patent No. 4,216,471 describes a system which avoids this problem of high power consumption by flipping the display elements one element at a time, in sequence. This sequential method of operation has the disadvantage, however, of being relatively slow. Moreover, the relatively long period of time in which the display is in transition from one line of a message to another often presents a distracting and unappealing appearance to the viewer.

In all of the above-mentioned prior art systems the display elements are flipped row by row in a top to bottom or bottom to top manner or, alternatively, column by column in left to right manner. In all of the above cases the viewer tends to be distracted by the partially formed characters which are present during the transition period in which the second message is being formed.

It is an object of the invention, therefore, to provide a display control system having relatively low power consumption and simple circuitry.

It is a further object of the invention to provide a system capable of switching from displaying one line of a message to displaying another line in a relatively rapid manner.

It is still another object of the invention to minimize the transition period between displaying one line of a message and another so as to present a more pleasing appearance to the viewer.

SUMMARY OF THE INVENTION

In accordance with the invention, an apparatus is provided which sets only one display element at a time and thus requires only one driver circuit to operate at a time. This feature results in an appreciable saving of power and economy of construction when compared to former apparatus requiring a plurality of such driver circuits operating simultaneously. In order to decrease the time required to switch from one displayed line of a message to another, the apparatus compares a coded version of each character or symbol currently being displayed by a particular module with a coded version of the next character or symbol to be displayed by the same display module. From this comparison signals are produced which serve to select the particular individual display elements in each display module which must be reset from dark to light or light to dark in order to display the next character. The selected display elements are then set sequentially.

This sequential manner of setting the display elements results in appreciable savings in hardware and decrease in average power consumption. Moreover, the present invention allows a significant decrease on the time consumed in switching from one line of display to another as will be seen from the following.

A drive signal to set a typical display element must be active for approximately one millisecond. Thus, the time consumed to set all of the display elements of a typical 5x7 display module in sequence would be approximately 35 milliseconds. In accordance with the present invention, however, only those display elements which the comparing operation indicates must be changed are, in fact, energized to be set. For example, to change the character displayed by a module from an "I" to a "T" requires only four elements to be changed from dark to light. This requires approximately 4 milliseconds. While the time saving for various changes will vary, the present invention thus makes possible a considerable saving in the time required to complete the change of an entire displayed message in comparison to former systems such as the one set forth in the '471 patent referred to above. At the same time, less power is required than for simultaneously changing an entire row or column of elements.

It is possible for the elements which require change to be addressed randomly or in any sequence. In a preferred embodiment, however, the elements are addressed in accordance with a particular raster or sequence which proceeds column by column in a right to left manner within each module and left to right from module to module. It has been found that the use of this raster results in the most pleasing appearance of the display to the viewer during transition periods.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing brief description, as well as further objects, features and advantages of the present invention will be more fully understood by reference to the following detailed description of a presently preferred (but nonetheless illustrative) embodiment of the present invention when taken in conjunction with the accompanying drawings in which:

Fig. 1 is an overall, primarily functional, block diagram of an apparatus in accordance with the present invention;

Fig. 2 is a table illustrating a raster arrangement which defines the sequence in which individual elements of the display are actuated; and

Figs. 3A through 3D comprise a simplified flow chart of programming means defining operations of a microprocessor for producing operations in accordance with the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENT

Fig. 1 shows the basic components of a system embodying the invention. The system includes one or more signs 10 which include a plurality of display modules 40 organized into one or more boards. In a presently preferred embodiment each sign includes three boards having five modules each.

Each display module 40 comprises a matrix of display elements 41 such as the five column by seven row matrix shown. Each element 41 includes a movable "dot" 42 which may, for example, be a sphere or disc rotatably mounted in a support. Each dot 42 has a light colored or bright surface and an opposite dark surface. Typically, the dot 42 is permanently magnetized and the support of the element 41 has an electromagnetic coil assembly (not shown) mounted on or within it. The dot 42 may be "flipped" to one of its surfaces or the other by the application of a pulse of current to the coil assembly in the appropriate direction. After the dot 42 is flipped, it remains in its new position until another pulse of current is applied through the coil assembly in the opposite direction to cause the dot 42 to flip again. The display element 41 thus has the advantage of requiring power only when its dot 42 is flipped and no power is dissipated otherwise. Display elements and modules of the type described above are commercially available from companies such as Ferranti-Packard Electronics Ltd. of Ontario, Canada.

In a presently preferred embodiment, particularly advantageous for the display of destinations and the like on public transportation vehicles, thirty-five display elements 41 are arranged to form a module 40 comprising a matrix having seven rows and five columns of elements 41. Five modules are arranged side by side to form a display board and three boards are arranged side by side to have a sign having the capacity to display a line of text having up to fifteen characters.

The preferred apparatus for controlling the characters or images to be displayed by the sign 10 includes a microprocessor 12 having means such as a ROM (Read Only Memory) or a PROM (Programmable Read Only Memory) for storing a program for determining the functioning of the apparatus, and a Random Access Memory (RAM) for storing data and intermediate results from manipulations of that data. The apparatus further includes a memory, such as an erasable programmable read only memory (EPROM) 15, in which is stored data representing the repertoire of messages it is desired to display selectively. An operator may select any desired message for display, by means of an input device 11 which in a presently preferred embodiment is a keyboard. The input device 11 communicates with the remainder of the system through an input buffer 14 which in turn is connected to a data bus 20, illustratively carrying 8 bits in parallel. The data bus 20 is connected to an input to the microprocessor 12.

The entire set of messages which may be selected to be displayed by the sign 10 are prestored in ASCII (American Standard Code for Information Interchange) format in the EPROM 15. In the presently preferred embodiment, each message may contain one or more lines, each of 15 or fewer characters. Each of these lines has its characters stored in sequence in the EPROM, followed by a special character indicating the end of a line (EOL). Each of these lines may be stored in the actual manner in which it is to be displayed (that is, including stored blank spaces preceding or following the letters and/or numerals to be displayed). Alternatively, each line may be stored as a character string with some or all of its blanks deleted. At the end of the last line of each message a special character indicating the end of message (EOM) is stored instead of the EOL character. The successive lines of each message are stored sequentially within the EPROM 15, that is, the first character of each line is stored at the next sequential memory address after the EOL character for the previous line. Messages are also stored sequentially within the EPROM 15, the first character of a message being stored at the memory address directly succeeding the EOM character of the previous message.

In addition to the above, the EPROM 15 may have a number of memory sections set aside for the storage of special characters, such as the ASCII Code for a blank space.

A keyboard which is adapted for use as the input device 11 of the present invention may have keys corresponding to the digits 0 through 9, and also a number of additional "function keys", each indicating an operation which is to be performed. For example, the keyboard may contain a "destination" key for indicating that the operator wishes a new destination to be displayed, and a "clear" key to indicate that all information should be cleared from the indicator signs 10. Other function keys may also be included. Each of the function keys of the keyboard actuates a corresponding signal path 22, all of which are converted to an OR circuit 30. The output of the OR circuit 30 is connected to an "interrupt" terminal of the microprocessor 12.

To operate the system, the operator actuates one of the function keys, which activates a corresponding signal path 22 to the OR circuit 30. The output signal from the OR circuit 30 interrupts any operations being currently performed by the microprocessor 12, which then puts itself into a state to accept data from the data bus 20. The operator then enters into the input device 11 a numerical code which corresponds to a predetermined message. Each digit of this code is presented to the data bus 20 by the input buffer 14 for a sufficient period of time for the microprocessor 12 to read the data and set the data into its internal RAM.

Microprocessor 12 may communicate with the remainder of the system via the data bus 20 and an address bus 21, which may carry 16 bits in parallel. In addition, microprocessor 12 has a control port 12a through which various specialized control signals generated by the microprocessor may be routed to the remainder of the system. A decoder 31 decodes the signals from the control port 12a to produce discrete signal paths. One of these signal paths carries a one millisecond strobe signal which is generated by the microprocessor 12 at appropriate times for activating the system to "write" or "erase" dots on the display modules as described below.

The one millisecond strobe signal from the decoder 31 is fed to a pair of AND gates 35 and 36. The other inputs to these AND gates are address bus lines A13 and A14, respectively, forming part of the 16-line address bus 21. The output of gate 36 feeds an interface 16 which converts the signal into a one-millisecond "write" signal appearing on lead 36a, and the output of the gate 35 is similarly converted into a one-millisecond "erase" signal appearing on lead 35a.

Lines A10 through A12 of the address bus 21 carry binary encoded signals specifying the address of one of the possible eight signs in the system. These signals are fed through interface 16 to feed a sign-select decoder 34. In response to the signals at its input, decoder 34 activates an appropriate discrete signal path 23 which selects the particular sign on which a message is to be written, erased or changed.

Address bus lines A5 through A9 carry binary encoded row address signals. These signals are fed to the interface 16 where they are converted to a set of row-select lines 24. These row select lines 24 are fed to a decoder 33 along with the "write" signal and the "erase" signal. Decoder 33 decodes these signals to activate an appropriate line 26 which selects the particular row in which a selected dot 42 is to be appropriately flipped.

Address bus lines A0 through A4, which carry binary encoded column address signals, are connected into a set of column-select lines 27 by the interface 16. These column select lines are fed to a decoder 32 along with the "write" and "erase" signals. Decoder 32 activates an appropriate line 27 which selects the particular column of the sign in which a dot is to be flipped. Thus the combination of the activation of an appropriate line 26 and the activation of an appropriate line 27 uniquely selects and activates one particular dot 42 to be flipped.

The decoders 32, 33 and 34 referred to in the previous discussion comprise commercially available circuits which take a binary encoded input and respond by producing a discrete output signal for each input code. Each output signal is connected to a suitable current driver which is also commercially available. The current drivers produce current sufficient to flip the dots of the sign.

For example, the sign select decoder 34 is driven by binary codes "000" through "111" appearing on address lines A10 through A12. Each of these codes activates and drives a respective one of eight output signal paths which selects one of eight possible signs.

GENERAL MODE OF OPERATION

A detailed discussion of the mode of operation of the present system in conjunction with a flow chart of the program contained in the microprocessor 12 may be found below. It is believed, however, that the following general discussion of the operation will be helpful in understanding the more detailed explanation to follow.

Each message that may be displayed by the system is assigned a unique code or number. This number corresponds to the order in which the message is written into the EPROM 15. For example, the 15th sequential message within the EPROM 15 is assigned the message code 15. Thus, in order to select a particular message to be displayed, the operator enters the message's code number into the system using the input device 11. As each digit of this number is keyed in, it is stored temporarily in the input buffer 14 and presented to the data bus 20. Data bus 20 routes the digit to the microprocessor 12 where it may be stored in the microprocessor's random access memory (RAM). After the microprocessor 12 has received the message's entire code number, it begins to sequentially read the data stored in the EPROM 15, utilizing the address bus 21 to select the location from which data is to be read and the data bus 20 to communicate the data read. Each time an end of message (EOM) character is read from the EPROM 15, a count in the microprocessor 12 is incremented by one. (The incrementation may be accomplished by either programming or hardware means.) This process is continued until the count becomes equal to the code number previously entered by the operator. This equal condition occurs when the EOM character designating the end of the message desired to be displayed has been reached. It will be appreciated that a similar result could be achieved by setting the message code into a counter decrementing the count upon reading of each EOM until the count reaches zero.

At this point, the system "backs up" to the first EPROM 15 memory location after the preceding EOM.

This memory location is the beginning of the "character string" comprising the letters and numerals of the first line of the message to be displayed. A character string comprises the alpahnumeric characters and blank or "imbedded" spaces (if any) between them. It preferably does not, however, include blank spaces before the first letter or numeral in the string or after the last letter or numeral. A character string generally not equal to the number of display modules 40 in the sign 10.

In a preferred embodiment, a determination is made in any desired way as to the number of characters in the character string and this number is stored in a predetermined location of the RAM. Utilizing this data the system calculates the number of preceding blanks and following blanks needed to center the character string on the sign 10, and this data is stored. The system also stores the address of the first character of the character string. All of the above data is retained, not only during display of this message line, but also until the display of the next line of the message is completed. Thus, while the display function is being performed, the system has available to it appropriate data defining the length, centering and location of both the current character string and the previously displayed character string.

Assuming that there are leading blanks in the message line currently being displayed, the system addresses the predetermined location in EPROM 15 or any other convenient storage medium which contains the ASCII code corresponding to a "blank" and transfers this data into the microprocessor 12 RAM. If the message does not contain any leading blanks, the ASCII code for the first character of the character string is transferred to the RAM. In either of the above cases, the ASCII code stored in the RAM is applied to a decoder (hardware or software) in the microprocessor 12. This decoder converts this ASCII code into data defining the pattern of light and dark dots which should be displayed by the display module 40 to display the selected character or blank.

This decoding process is performed in a column by column manner. When the dot pattern for the first column is determined it is stored. The dot pattern for the first column of the character previously displayed in the same module was similarly determined and stored. The two patterns are then compared to determine which of the dots 42 of the display module 40 must be flipped from light-to-lark (erased) or dark-to-light (written). The appropriate write or erase pulses are then generated by the microprocessor 12 in accordance with a display sequence or raster to be described below. For example, if the previous character displayed and the current character are both blanks, no write or erase pulses need be generated by the microprocessor 12. The write and erase pulses then cause the appropriate dots to be flipped, changing the column from that forming part of the previous character to the condition forming part of the character being currently displayed. The above process is continued column by column until the new character is completely formed and then repeated for all fifteen modules 40 of the sign 10.

DETAILED DESCRIPTION

The Raster

In the apparatus in accordance with the present invention, all of the display elements 41 or dots 42 of the display are given an order or sequence called a "raster". In the present embodiment there may be up to eight signs 10, each sign 10 having three boards, each boards comprising five display modules 40, and each module 40 comprising seven rows of five display elements 41 and dots 42. Each element 41 is given an individual designation or address in the binary numbering system to facilitate its being addressed for flipping its dot 42 from light-to-dark or dark to light. The lower order five bits of this binary number designate the column in which the element 41 is located (there are 25 columns in each board, numbered from 0 to 24 in binary numbers). The next higher order three bits designate the row (from 0 to 6), the next higher order two bits desingate the board (0 to 2), and the next higher order three bits designate the particular sign (illustratively 0 to 7). The usual 16-bit word or "byte" may thus be used.

Fig. 2 illustrates the manner in which the addresses are assigned to the display elements 41 of the first board (board 0) of the first sign (sign 0). Each box on the diagram of Fig. 2 represents an individual display element and includes the binary address of that element converted to the hexadecimal (base 16) numbering system for legibility. For the second board of the first sign, the numbering scheme is repeated except that the third hexadecimal digit from the right becomes a 1 instead of a 0. For example, the upper lefthand display element of the lefthand module of the second board is designated "100" (hex). Similarly, in the third board, the third hexidecimal digit from the right becomes a "2" instead of a "0". Thus, the upper lefthand display element of the leftmost module of this board is designated "200" (hex).

For additional signs, the leftmost digits of the element designation are changed accordingly. For example, in the second sign (sign 1) the upper lefthand element of the leftmost module of the first board is designated "400" (hex), and the same element in the third board (board 2) of this sign is designated "600" (hex).

For the 8th sign (sign 7) the hexidecimal designation of the upperleftmost element of the leftmost module of board 0 is "1C00", and the same element in the third board is "1E00".

It will be appreciated from the above that since each element 42 has a unique designation or "address", it is possible to utilize an addressing scheme which randomly selects the desired dot 41 for flipping. In the present embodiment, however, it has been found to be convenient to scan through the elements 42 in a particular order or raster and the bit designations forming the address of each element 41 have been selected to enable the performance of this scan in a most advantageous manner.

Referring to Fig. 2, a scan of the elements 41 of a board of the present embodiment (whose elements are disignated by 00 hex through D8 hex begins at the upper rightmost element ("04" hex) of the leftmost module 40. The scan continues down the column through elements "24", "44", "64", "84", and "A4", until it reaches the bottom element of the column designated as "C4". The scan then resumes at the next column on the left starting at element "03" to "C3", and continues similarly to scan in a column by column manner until it reaches element "C0" at the lower left corner of the module. The scan then resumes at location "09" of the second leftmost module and continues in a similar manner.

In accordance with the above raster, the manner of changing a message displayed on a sign is right to left by column for each module, and left to right by module. This raster has been found to produce a visual appearance of a faster message change than do alternative rasters such as one in which columns and modules are both changed in a left to right manner. Nevertheless the use of alternative rasters, or of random selection of display elements is within the scope of the present invention.

<u>The ASCII Translation Table</u>

In order to enable the microprocessor 12 to translate ASCII coded character information into information indicating the particular dots 42 of a display board 40 which must be flipped, the microprocessor 12 of a presently preferred embodiment includes, in a special block of storage, an ASCII-to-dot conversion table. This table includes a set of five bytes (8 bits each) corresponding to each ASCII character. Each byte represents one column of dots for the character. Thus, for the character "H", the first and last bytes would include seven binary "1"'s (the 8th bit is ignored). This indicates that all seven dots of the rightmost and leftmost columns of the display module displaying that character must be colored. The second through fourth bytes would be a binary "0001000" since only the middle dot of the second through fourth columns of the character "H" are colored.

DETAILS OF OPERATION

Figs. 3A through 3D illustrate, in simplified flow chart form, the manner of operation of the apparatus of the present invention. The following discussion of Figs. 3A through 3D may be best understood by further reference to the block diagram of the system of Fig. 1.

Referring to the block 100 of Fig. 3A, the display of a desired message is initiated by the operator pressing the "destination" key on the input device 11. The pressing of this key activates a "destination" signal to the OR circuit 30 which responds by generating an "interrupt" signal and supplying it to the microprocessor 12. Upon receipt of the "interrupt" signal, the microprocessor 12 commences a sub-routine which reads any data which subsequently appears in the input buffer 14 and stores this data into the microprocessor RAM. The operator then keys in the numeral code corresponding to the desired message and this code is stored, one numeral at a time, in the microprocessor RAM (see blocks 102 and 103). The "enter" key is then pressed causing the program to proceed to block 104.

In the present embodiment, the message code preferably has a direct relationship to the position at which its corresponding message is stored in the EPROM 15. For example, a message code of "10" would correspond to the tenth sequential message stored in the EPROM 15.

Proceeding to block 104 of the flow chart, the microprocessor now begins a subroutine in which the data stored in the EPROM 15 is sequentially read one character at a time. Addressing of the EPROM 15 for reading purposes is performed by the microprocessor via the address bus 21 and the data read from the EPROM 15 is presented to the microprocessor 12 via the data bus 20.

As each character is read, it is decoded by the microprocessor 12 to determine whether or not it is one of the special characters denoting the end of a message (EOM) or the end of a line (EOL) of a message. As indicated by diamond 105, if the character is not an EOM, the program proceeds to diamond 106. As indicated by diamond 106 if the character is not an EOL character, the program returns to block 104 to read the next character from the EPROM 15. If, on the other hand, the character is an EOL character, the program proceeds from diamond 106 to block 107 and the memory address of that EOL character is stored in the microprocessor RAM. Thereafter the program returns to block 104 to read the next character. The EOL address stored at block 107 is always that of the last EOL read, and changes if a later EOL is read.

Returning to diamond 105, if the character just read was an EOM character, a count within the microprocessor 12 is incremented by one. This incrementing may be performed, for example, in hardware by a counter, or by a software routine, or by a combination of the two.

After incrementing the EOM count, the program proceeds to diamond 109 where the EOM count is compared with the previously stored message code (see block 103). If the two numbers do not correspond, the program proceeds to block 110, and the address of the EOM character just read is stored in the microprocessor RAM for future reference. The program then returns to block 104 for the reading of the next sequential character. The EOM address stored at block 110 is always that of the last EOM read, and changes if a later EOM is read.

If the comparison in diamond 109 indicates that the EOM count corresponds to the input code, then the EOM character for the desired message to be displayed has been found. At this point a number of bookkeeping-type operations are performed by the system. A character counter is set to 15 (block 111). The system backs up to the address the first character stored after the preceding EOM (the address of which was stored in block 110). The system notes or stores the address of this first character of the string as the current EPROM 15 address (MEMADDR).

This character is the beginning of the "character string" for the first line of the new message to be displayed. A character string is formed by the alphanumeric characters and imbedded blank spaces, if any, between two successive EOL characters or between an EOL character and an EOM character in the EPROM 15.

Since it is not necessary to store leading and trailing blanks in the EPROM 15, each character string stored in it may contain fifteen or fewer characters in the present illustrative embodiment having 15 modules 40 in each sign 10.

As indicated by block 113, the system at this time also determines the number of characters in the character string. This may be done, for example, by actual counting from the preceding EOM to the first EOL, or by subtracting from the address of the first EOL the address of the first character of the character string. This character string count is stored for later use.

Proceeding to block 114, the microprocessor 12 now determines how many blank spaces are required to be displayed before and after the character string. The number of leading blanks required is determined by subtracting the number of characters in the character string from the number of modules in a sign (here 15) and dividing the result by 2, either ignoring any remainder or adding it to the dividend. This number of required Leading blank spaces is stored in a "blank counter" within the microprocessor 12.

The program then proceeds to diamond 115 (Fig. 3B) where a determination is made as to whether the blank count is zero. Assuming that a number other than zero has just been set into the blank counter, the "no" branch out of diamond 115 is taken and the blank count is decremented by 1 in block 116. At this time the character count is also decremented by 1 (see block 117). The program then proceeds to the display routine as indicated by block 118.

At this point it will be noted that upon entering the display routine, the data concerning the previous line displayed normally has previously been stored in microprocessor 12. This data includes the number of its leading blanks, the EPROM 15 address of the first character of its character string, the number of characters in its character string, and the number of trailing blanks.

The display subroutine starts at diamond 119 where it is determined whether the current character to be displayed is a blank. Assuming that it is, the "yes" branch out of diamond 119 is taken and diamond 120 is entered. At this point the microprocessor utilizes the previously stored data concerning the previous message line displayed to determine whether the corresponding character in the previous message line was a blank. If the previous character was a blank, all of the dots 42 of the relevant display module 40 are already turned to their dark sides and nothing further need be done. Therefore, the "yes" branch out of diamond 120 is taken and, assuming that the system is still in the process of supplying leading blanks, the program returns to diamond 115.

If, on the other hand, it is determined in diamond 120 that the previous character was not a blank, the "no" branch out of diamond 120 is taken and block 122 is entered. As indicated by block 122, the microprocessor now retrieves the ASCII code for a blank from a predetermined fixed storage location in the EPROM 15. In block 123 this ASCII code is applied to a decoder sub-routine which, using the ASCII translation table, finds the five "control words" or bytes having patterns of ones and zeros corresponding to the respective light and dark faces to be displayed by dots to display the desired current character. In the case of a blank character all five of these control words contain all zeros.

As block 123 also indicates, the system at this point retrieves the five control words corresponding to the previous character which had been displayed. The system then proceeds to block 124 where the control words for the current character to be displayed are compared with the control words for the previously displayed character.

In this comparison subroutine the system proceeds dot-by-dot in the sequence determined by the previously discussed raster to determine whether a change from light to dark or dark to light is necessary in order to display the current character.

The comparison subroutine starts in Fig. 3C at block 201 at position 004 (hex) of the raster table. In the following discussion the symbol "A" represents the logical state of the previous dot and the symbol "B" represents the logical state of the currently desired dot. In block 202, A and B are compared in accordance with the Boolean expression

$$(A \oplus B) \cdot B$$

here the symbol $\oplus$ indicates a logical "exclusive OR" operation. If this expression reduces to a logic "1" , it is necessary to rotate the dot 42 from its dark face to its light face (write operation). It may be noted at this point that if a blank is to be displayed it will never be necessary to perform a write operation. If, on the other hand, a non-blank character is to be displayed and a write operation is necessary, the "yes" branch is taken out of diamond 203 and block 207 is entered . As indicated by block 207 the now microprocessor 12 generates a one millisecond strobe pulse at its control port 12a. This strobe pulse is routed through the decoder 31 to the AND gates 35 and 36. The microprocessor also activates the appropriate address line A14 on the address bus 21 which is AND'ed with the strobe at gate 36 to produce a 1 millisecond write strobe at the output of interface 16 corresponding to gate 36. The microprocessor also activates the appropriate address lines A0 through A12 on the address bus 21 to indicate the current position in the raster scan (here 004 hex). All of these lines are supplied to the interface 16 which converts them to supply the appropriate row-select, column-select and sign select signals to apply the write strobe to the selected display element (the upper right-hand element of the leftmost module of the display board).

If, as is the situation when a blank is to be displayed, the comparison in block 202 indicates (by a zero) that the relevant dot is not to be flipped from dark to light, the "no" branch out of block 203 is taken and block 204 is entered. In block 204, A and B are again compared, in accordance this time with the Boolean expression

$$(A \oplus B) \cdot A$$

to determine whether a dot which previously displayed its light face is to be flipped to its dark face (erased).

It may be noted at this point that when power is first turned on on the system, the storage locations which store the control words for the previous character are automatically set to all ones. A special subroutine (not illustrated) is then entered whereby all blanks are displayed. Thus, when block 204 is entered as a result of the power-on subroutine, an erase is performed for every position in the raster. In an alternative embodiment, the power-on sub-routine may be performed each time the "enter" key of the input device 11 is activated.

If an erase is to be performed, the "yes" branch out of diamond 205 is taken and block 206 is entered. In accordance with block 206, address bus line A13 (indicating an erase) is activated by the microprocessor 12, a one millisecond strobe pulse is generated at control port 12a, and the lines A0 through A12 in the address bus correspon-ding to the current position in the raster are activated. As in the write operation, the drivers 16 decode the address lines and supply an erase pulse to the appropriate display element 40, thereby flipping its dot 42 to its dark face.

When the process indicated by block 206 or block

207 is completed or when the "no" branch out of block 205 is taken, diamond 208 is entered. Assuming that all thirty-five raster positions of a module have not yet been completed, the "no" branch out of diamond 208 is taken and the process described above from block 202 is repeated for the next position in the raster table. This process continues until each of the thirty-five elements in the display module currently being changed have been addressed, at which point the "yes" branch is taken out of diamond 208 and the program returns to diamond 115 via diamond 210.

The above described process is continued until the blank counter is decremented to zero at which time the "yes" branches are taken out of diamonds 210 and 115 and block 301 is entered. As indicated by block 301, the program now utilizes the EPROM address of the first character of the character string previously stored in the microprocessor RAM to read the ASCII code for the next (here the first) character in the character string. Block 118 is then entered and the program proceeds again to the display subroutine starting at block 119. Assuming that the ASCII character just read does not represent a blank, the "no" branch out of diamond 119 is taken and diamond 126 is then entered. In diamond 126, a determination is made from the previously stored information as to whether the previous character displayed by the same module was a blank. If it was, the "yes" branch out of diamond 126 is taken and the ASCII code for a blank is read from the predetermined location in the EPROM 15 as indicated by block 122. If the previous character was not a blank, the "no" branch out of diamond 126 is taken, and the ASCII code for the previous character is read from the EPROM 15 utilizing the address information previously stored (see block 128). From either block 128 or 122 the program proceeds to block 123 where, as previously set forth, the ASCII codes for both the previous and the current character are decoded into the five control words corresponding to each of them. The system then proceeds as before to compare the previous and current control words and to flip the appropriate dots in the subroutine starting at block 201.

At the end of the compare subroutine the program enters diamond 208 where it is determined whether the writing of the current display module is completed. If it is not, a "no" branch is taken and the program returns to block 202. If it is, the program goes to diamond 210. Since the blank cound is now zero, the yes branch out of diamond 210 is taken and is block 303 is entered where the character count is decremented by one. The program then determines whether any characters are left in the character string in diamond 304. If there are characters left the program returns to block 301 reading the next ASCII code from the updated memory address.

Returning to diamond 304, if it is determined that there are no characters left in the character string (the next character in EPROM 15 is an EOL or an EOM symbol) the "no" branch from diamond 304 is taken and diamond 305 is entered. In block 305 the character count is examined to see whether it has been decremented to zero. If it has, the program proceeds to diamond 306 where it is determined whether the next character in EPROM 15 is an EOM signal. If it is, the "yes" branch is taken and the program again returns to the first character after the previous EOM and begins to repeat the process of displaying the first line of the message as indicated in block 307 after a suitable retention time (e.g. 1.5 seconds).

Returning to diamond 306, if the next sequential character in EPROM 15 is not an EOM (so it must be an EOL), the program continues by displaying the next line of the message which starts at the next sequential character after that EOL as indicated by block 308.

Returning to block 305, if the character count is not zero, but there are no more characters in the character string, then trailing blanks must be supplied. In the case the "no" branch is taken out of diamond 305 and block 309 is entered. As indicated by block 309, the current character count is now inserted into the blank counter and the blanks are inserted by means similar to the means employed to insert leading blanks A. To accomplish this, the program returns to diamond 115, and operates as already described.

It will be noted that the time consumed by the display process described above is attributable mainly to the number of one-millisecond write or erase pulses which are generated since the remainder of the operations take place primarily in the microprocessor 12 at speeds several orders of magnitude faster than the one millisecond necessary to flip a dot. The method and apparatus described above therefore enables an appreciable decrease in the time to change a line of display compared to previous sequential-by-bit systems, which generated a write or erase pulse for each element of each module regardless of whether flipping was necessary. Thus, for example, for a 35-element module (7 rows by 5 columns) previous systems generated 35 sequential pulses requiring at least 35 milliseconds. The present apparatus, on the other hand, needs to generate a write or erase pulse only when a particular dot must be flipped from light to dark or dark to light. Typically, only a fraction of the number of dots of each display module must be changed during each operation. The present system therefore greatly reduces the time needed to replace one displayed line with the next, and thereby facilitates reading of the lines by the viewer, who no longer has to wait as each character changes, and is presented much more briefly with parallel lines in process of change. While the foregoing description has been directed illustratively to signs having modules in a single row, it is to be understood that the sequence of modules may be "folded" or arranged in several rows where desired.

Although specific embodiments of the invention have been described for illustrative purposes, it will be appreciated by one skilled in the art that many modifications, additions and substitutions are possible without departing from the scope and spirit of the invention. For example the completion of the display of a line of characters can be determined by detecting that the raster scan has reached its last position (2D4), rather than by examining a character count as in diamond 305.

What Is Claimed Is:

1. A method for changing a first image displayed on a display module to a second image, said module including a matrix of display elements, each of said elements having a dark state and a bright state, said method comprising the sequential steps of:

storing data defining said first image;

selecting data defining said second image;

comparing said first image data to said second image data for determining which display elements must change their state to display said second image; and

changing the state of the display elements as determined in said comparing step.

2. The method of claim 1, wherein said changing step is performed one element at a time.

3. The method of claim 2 wherein said changing step is accomplished in accordance with a predetermined raster.

4. The method of claim 3 wherein said matrix comprises display elements aligned in rows and columns and wherein said changing step is accomplished by

changing the state of all of those display elements in one of said alignments determined in said comparing step to require change prior to changing display elements in any other of said alignments.

5. The method of claim 4 wherein said comparing step comprises:

deriving from said first image data a set of control words each corresponding to the states of the elements of a respective one of said alignments when displaying said first image;

deriving from said second image data a second set of control words each corresponding to the states of the elements of a respective one of said alignments when displaying said second image; and comparing each control word of said first set with a corresponding control word of said second set to derive a set of control signals representing the elements of each alignment requiring change to cause said module to display said second image.

6. The method of claim 5 wherein each of said control words includes a plurality of bits, each of said bits corresponding to a respective element, and wherein said comparing comprises:

comparing the bits of each of said first set of control words with the corresponding bits of the corresponding control words of said second set to derive a set of signals representing corresponding elements requiring change from the bright state to the dark state; and

comparing the bits of each of said first set of control words with the corresponding bits of the corresponding control words of said second set to derive a set of signals representing corresponding elements requiring change from the dark state to the bright state.

7. A method for changing a first image displayed on a display module to a second image, said module including a matrix of display elements, each of said elements having a dark state and a bright state, said method comprising the sequential steps of:

storing data defining said first image;

selecting data defining said second image;

comparing said first image data to said second image data for determining which display elements must change their state to display said second image; and

activating only the display elements requiring change as determined in said comparing step.

8. A method for changing a first set of images displayed on a plurality of display modules to a second set of images, each of said modules including a plurality of display elements, each of said elements having a dark state and a bright state, said method comprising the sequential steps of:

storing data defining said first set of images,

selecting data defining said second set of images,

comparing said first image data to said second image data for determining which display elements must change their state to display said second set of images; and

changing the state of the display elements as determined in said comparing step.

9. The method of claim 8 wherein said storing step comprises:

storing data defining a location where a representation of said first set of images may be found; and

storing data defining blank images displayed with said first set of images.

10. The method of claim 8 wherein said first set of images comprises a character string and wherein said storing step comprises:

storing data defining where a representation of the first character of said string may be found, and

storing data defining the number of blank images, if any, displayed prior to the first character of said character string.

11. A method of displaying a set of images on an array of display modules each having a matrix of display elements arranged in rows and columns, comprising

the steps of

activating said modules one after the other to display said set of images in succession,

and displaying each image on a respective module by activating the columns of said module in succession in a direction opposite to the direction of successive activation of said modules.

12. The method of claim 11 wherein the succession of activating said modules is from left to right and the succession of activating said columns is from right to left.

13. A system for controlling the display images on a display module including a plurality of display elements each said element having a dark state and a bright state, said system comprising

a memory for storing data defining an image currently displayed image;

an input device for selecting data defining a new image to be displayed;

a microprocessor including a comparator for comparing said current image data to said new image data and to determine which display elements must change their state to display said new image; and

a signal generator responsive to said comparing means for producing signals for changing the state of said selected display elements.

14. The system of claim 13 wherein said generator includes:

means for sequentially addressing each of said display elements in accordance with a predetermined raster; and means for generating a write or erase pulse for changing the state of each selected display element as said selected element is addressed.

Robert Edwin Rector

Larry Thomas Taylor

Scott Houston Yarberry

Richard Joseph Grassi

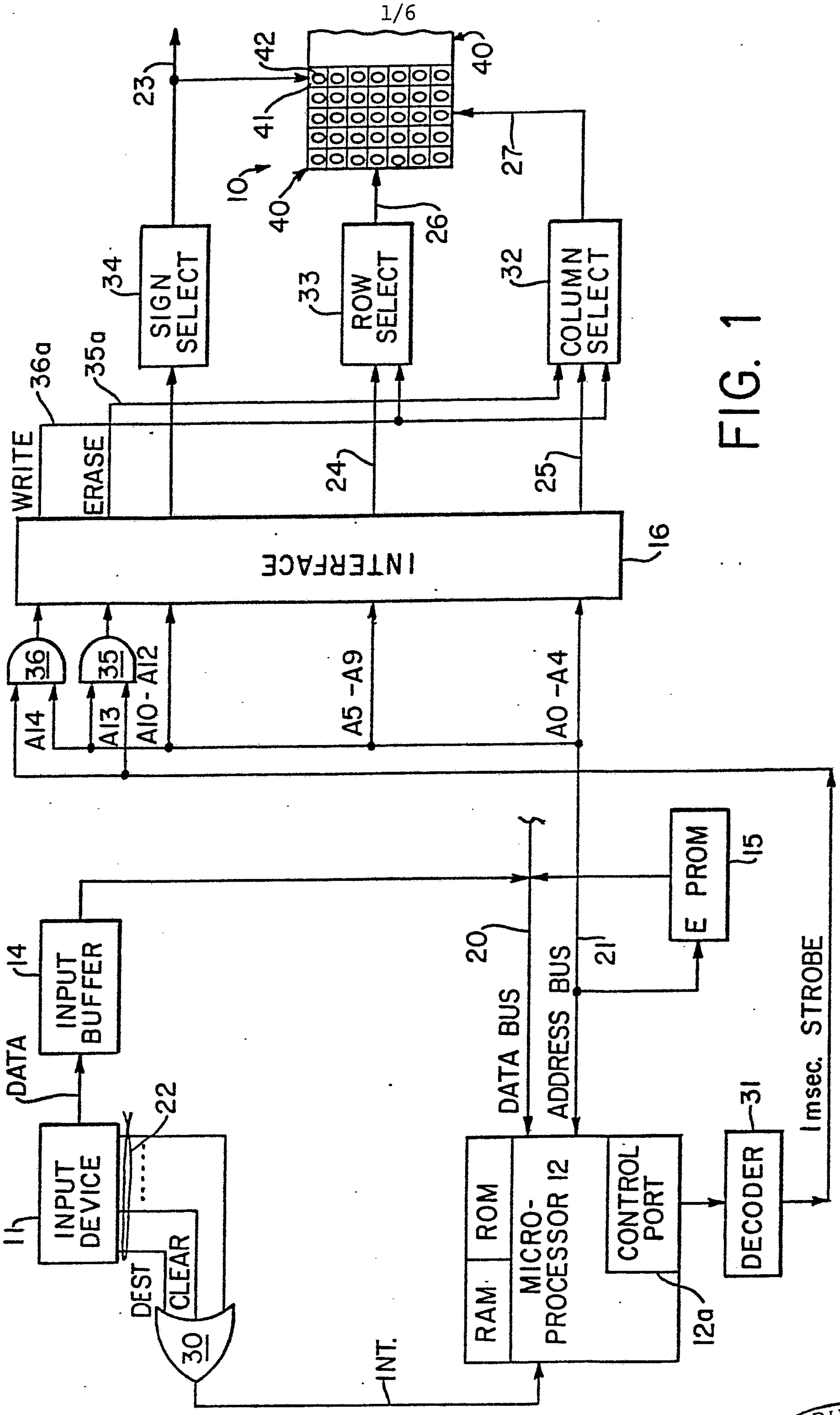

INPUT DEVICE
11
DATA
INPUT BUFFER
14
22
DEST
CLEAR
30
INT.
RAM
ROM
MICRO-PROCESSOR 12
CONTROL PORT
12a
DECODER
31
1msec. STROBE
DATA BUS
20
ADDRESS BUS
21
E PROM
15
A14
A13
A10 - A12
A5 - A9
A0 - A4
36
35
INTERFACE
16
WRITE
ERASE
36a
35a
24
25
SIGN SELECT
34
ROW SELECT
33
COLUMN SELECT
32
23
26
27
10
40
41
42

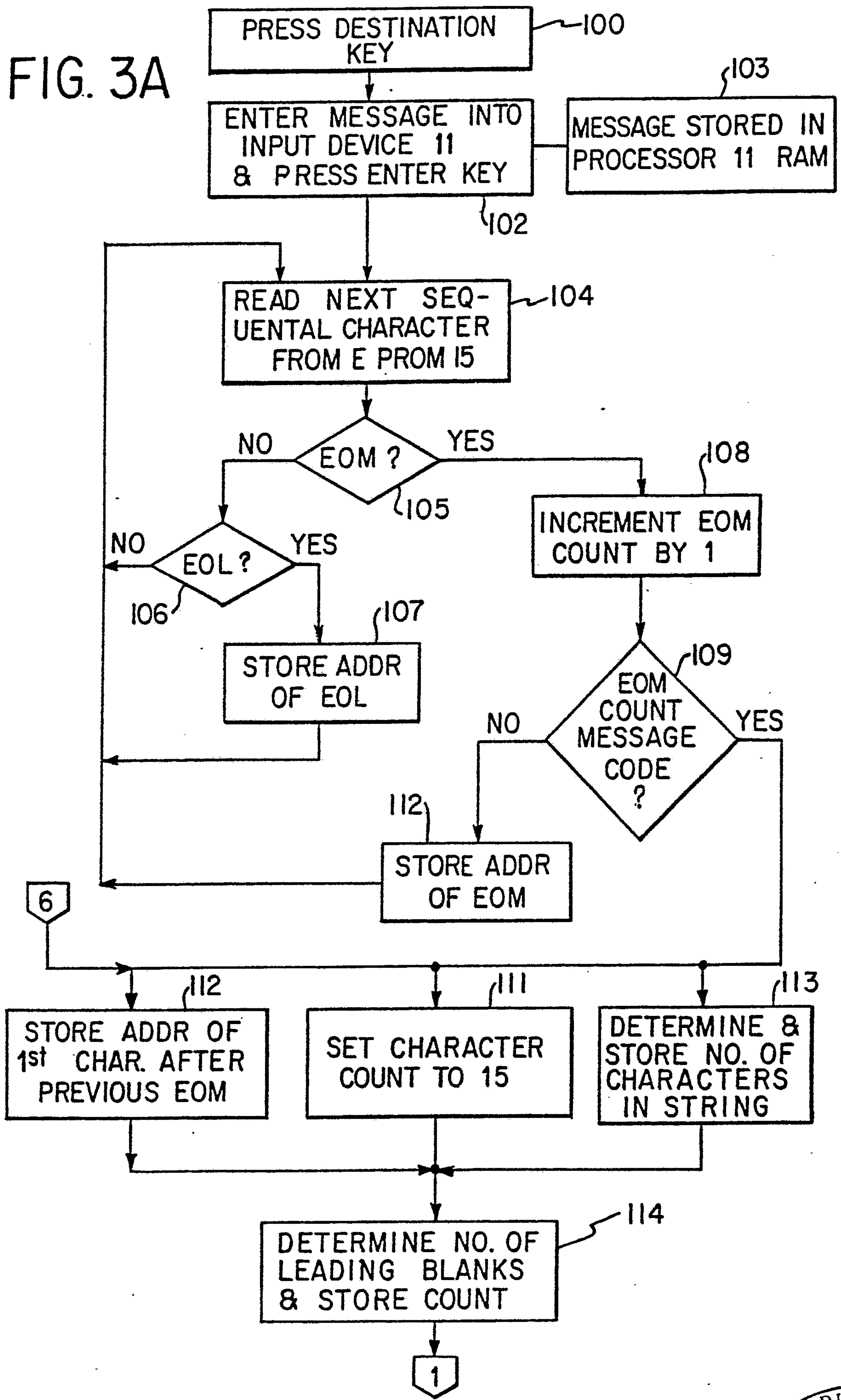
FIG. 1

FIG. 2

ROW \ COLUMN	0	1	2	3	4	5	6	7	8	9	10	11	12
0	00	01	02	03	04	05	06	07	08	09	0A	0B	0C
1	20	21	22	23	24	25	26	27	28	29	2A	2B	2C
2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C
3	60	61	62	63	64	65	66	67	68	69	6A	6B	6C
4	80	81	82	83	84	85	86	87	88	89	8A	8B	8C
5	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC
6	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC

ROW \ COLUMN	13	14	15	16	17	18	19	20	21	22	23	24
0	0D	0E	0F	10	11	12	13	14	15	16	17	18
1	2D	2E	2F	30	31	32	33	34	35	36	37	38
2	4D	4E	4F	50	51	52	53	54	55	56	57	58
3	6D	6E	6F	70	71	72	73	74	75	76	77	78
4	8D	8E	8F	90	91	92	93	94	95	96	97	98
5	AD	AE	AF	B0	B1	B2	B3	B4	B5	B6	B7	B8
6	CD	CE	CF	D0	D1	D2	D3	D4	D5	D6	D7	D8

FIG. 3A



PRESS DESTINATION KEY
100
ENTER MESSAGE INTO INPUT DEVICE 11 & PRESS ENTER KEY
102
103
MESSAGE STORED IN PROCESSOR 11 RAM
READ NEXT SEQUENTAL CHARACTER FROM E PROM 15
104
EOM ?
105
NO
YES
EOL ?
106
NO
YES
STORE ADDR OF EOL
107
INCREMENT EOM COUNT BY 1
108
EOM COUNT MESSAGE CODE ?
109
NO
YES
STORE ADDR OF EOM
112
6
STORE ADDR OF 1st CHAR. AFTER PREVIOUS EOM
112
SET CHARACTER COUNT TO 15
111
DETERMINE & STORE NO. OF CHARACTERS IN STRING
113
DETERMINE NO. OF LEADING BLANKS & STORE COUNT
114
1

FIG. 3B

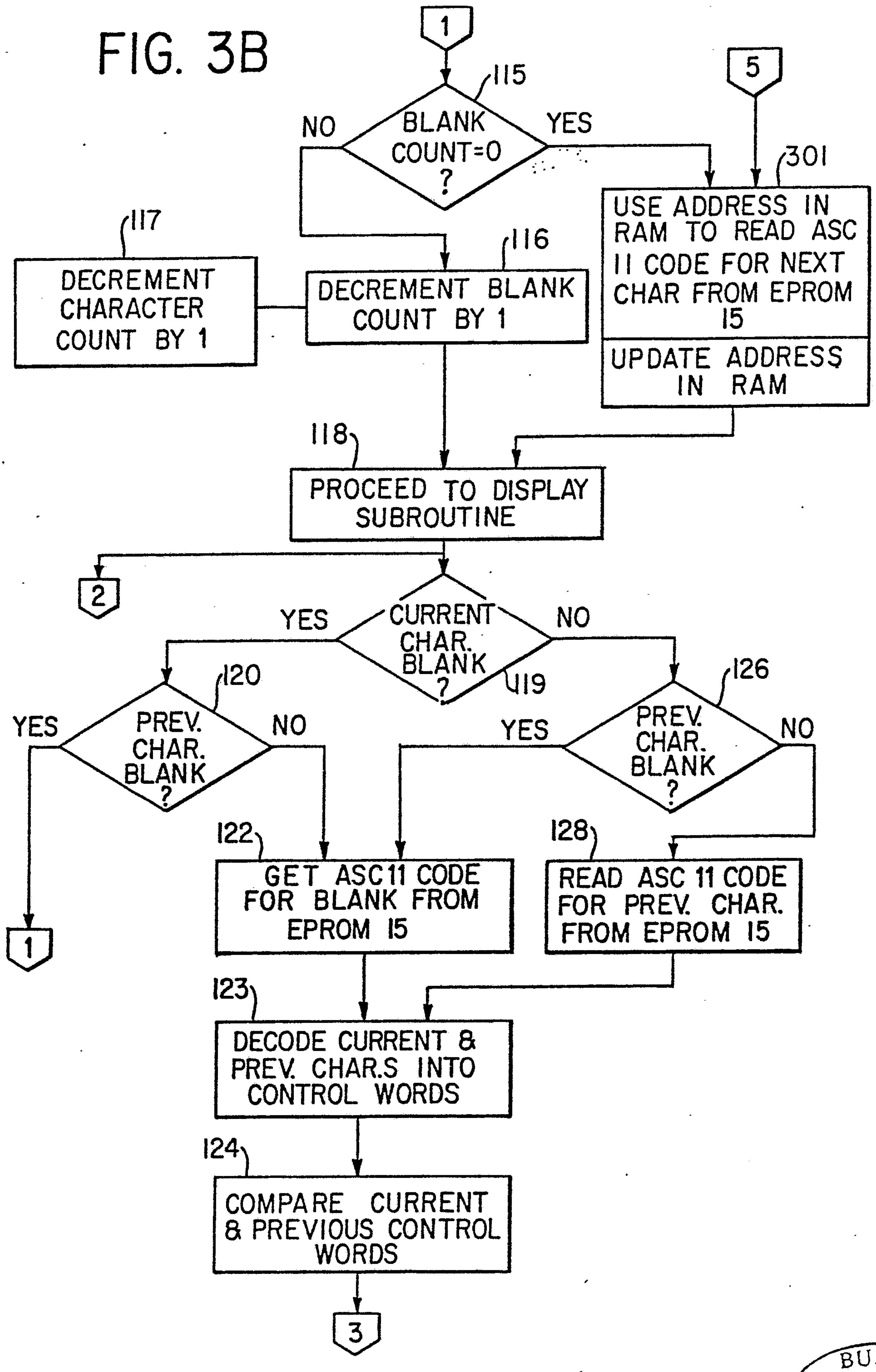
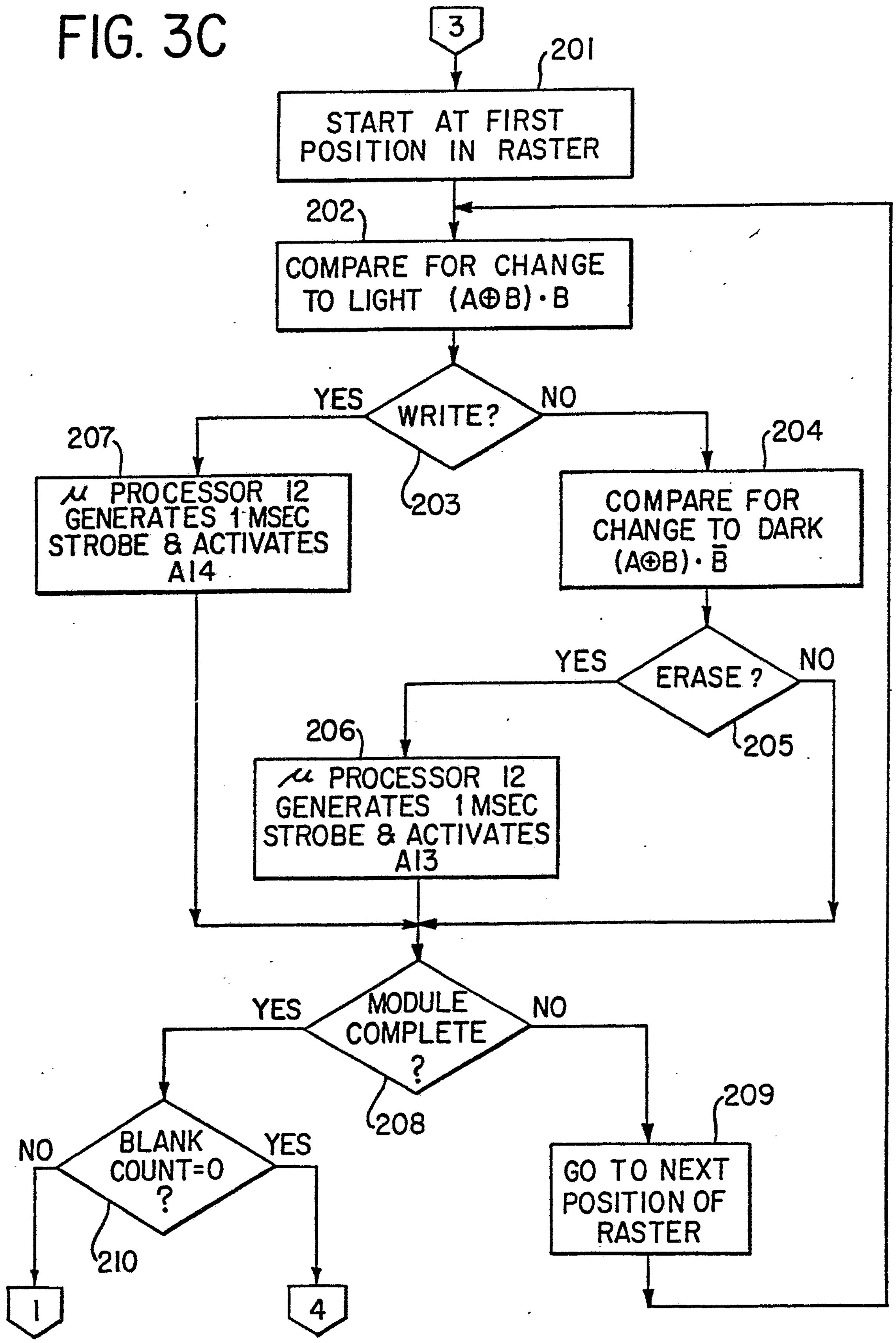

1
115
BLANK COUNT=0 ?
NO
YES
5
301
USE ADDRESS IN RAM TO READ ASC II CODE FOR NEXT CHAR FROM EPROM 15
UPDATE ADDRESS IN RAM
117
DECREMENT CHARACTER COUNT BY 1
116
DECREMENT BLANK COUNT BY 1
118
PROCEED TO DISPLAY SUBROUTINE
2
CURRENT CHAR. BLANK ?
119
YES
NO
120
PREV. CHAR. BLANK ?
YES
NO
126
PREV. CHAR. BLANK ?
YES
NO
122
GET ASC11 CODE FOR BLANK FROM EPROM 15
128
READ ASC 11 CODE FOR PREV. CHAR. FROM EPROM 15
1
123
DECODE CURRENT & PREV. CHAR.S INTO CONTROL WORDS
124
COMPARE CURRENT & PREVIOUS CONTROL WORDS
3

FIG. 3C


3
201
START AT FIRST POSITION IN RASTER
202
COMPARE FOR CHANGE TO LIGHT (A⊕B)·B
YES
WRITE?
NO
203
207
µ PROCESSOR 12 GENERATES 1 MSEC STROBE & ACTIVATES A14
204
COMPARE FOR CHANGE TO DARK (A⊕B)·B̄
YES
ERASE ?
NO
205
206
µ PROCESSOR 12 GENERATES 1 MSEC STROBE & ACTIVATES A13
YES
MODULE COMPLETE ?
NO
208
209
GO TO NEXT POSITION OF RASTER
NO
BLANK COUNT=0 ?
YES
210
1
4

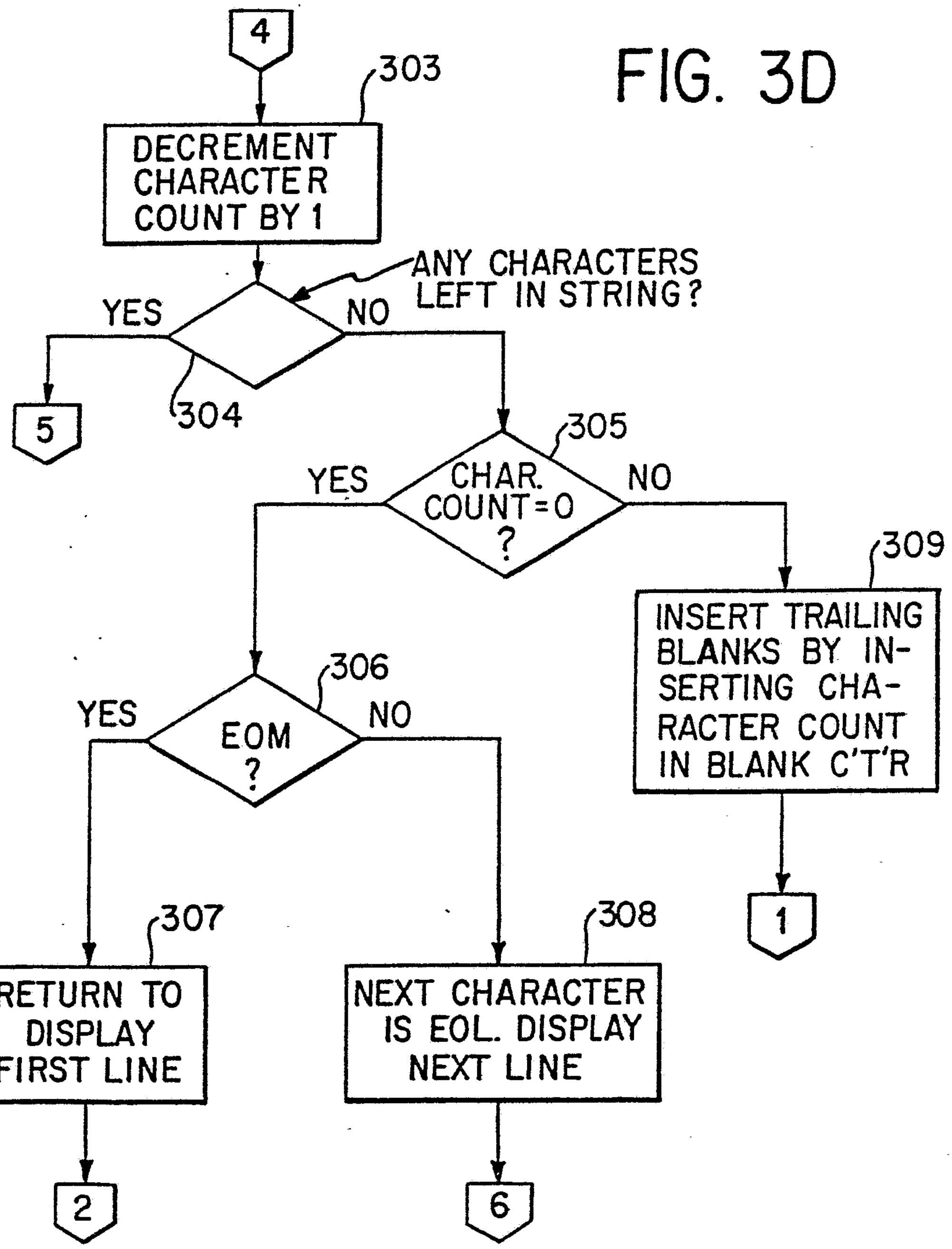

4
303
DECREMENT CHARACTER COUNT BY 1
ANY CHARACTERS LEFT IN STRING?
YES
NO
304
5
305
CHAR. COUNT=0 ?
YES
NO
309
INSERT TRAILING BLANKS BY IN- SERTING CHA- RACTER COUNT IN BLANK C'T'R
1
306
EOM ?
YES
NO
307
RETURN TO DISPLAY FIRST LINE
308
NEXT CHARACTER IS EOL. DISPLAY NEXT LINE
2
6

FIG. 3D

INTERNATIONAL SEARCH REPORT

International Application No PCT/US82/01444

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all) [3]

According to International Patent Classification (IPC) or to both National Classification and IPC

Int. CL.3 G09G 3/16
U.S. 340/764

II. FIELDS SEARCHED

Minimum Documentation Searched [4]

Classification System	Classification Symbols
U.S.	340/764, 783, 811, 815.04, 815.05

Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in the Fields Searched [5]

III. DOCUMENTS CONSIDERED TO BE RELEVANT [14]

Category *	Citation of Document, [16] with indication, where appropriate, of the relevant passages [17]	Relevant to Claim No. [18]
P	US,A, 4,295,127 (Sautter et al) 13 October 1981	1-14
A	US,A, 4,216,471 (Akred, Sr.) 05 August 1980	1-14
A	US,A, 4,069,480 (Helwig) 17 January 1978	1-14
A	US,A, 4,178,575 (Ono) 11 December 1979	1-14

* Special categories of cited documents: [15]
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
"&" document member of the same patent family

IV. CERTIFICATION

Date of the Actual Completion of the International Search [2]	Date of Mailing of this International Search Report [2]
1 DECEMBER 1982	21 DEC 1982
International Searching Authority [1]	Signature of Authorized Officer [20]
ISA/US	Marshall M. Curtis