

EUROPEAN PATENT APPLICATION

Application number: 88104228.7

Int. Cl.4: **G09G 1/16**

Date of filing: 17.03.88

Priority: 02.04.87 GB 8707851

Date of publication of application:
05.10.88 Bulletin 88/40

Designated Contracting States:
DE FR GB IT

Applicant: **International Business Machines Corporation**
 Old Orchard Road
 Armonk, N.Y. 10504(US)

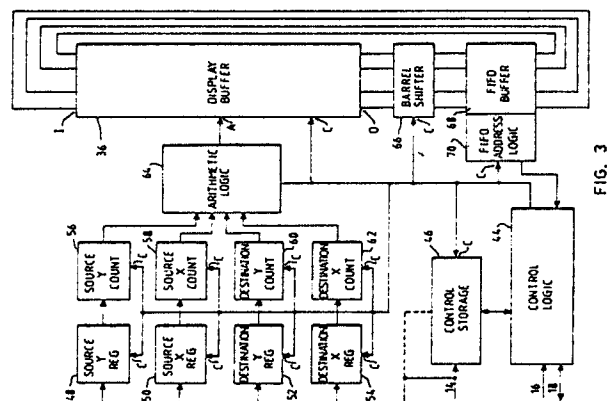
Inventor: **Harrison, Roy Bernard**
 5 Chichester Close
 East Wellow Hampshire, SO5 0EY(GB)
 Inventor: **Butler, Nicholas David**
 52 Staplerford Close Woodley
 Romsey Hampshire, SO51(GB)
 Inventor: **Bowater, Ronald John**
 Apeldoorn Whitenap Lane
 Romsey Hampshire, SO5 8ST(GB)
 Inventor: **Norris, Paul William**
 5 Ashbridge Rise
 Chandlers Ford Hampshire SO5 1SA(GB)

Representative: **Blake, John**
 IBM United Kingdom Limited Intellectual
 Property Department Hursley Park
 Winchester Hampshire SO21 2JN(GB)

Display system.

The present invention relates to a display system comprising a display buffer 36 data indicative of picture elements with a facility for moving an image from a source to a destination position by manipulating the data in the display buffer. The present invention provides this facility by having means for reading, processing and storing bursts of display buffer words. In the described embodiment of the invention control logic 44 first causes a burst of display buffer words containing bytes representative of the image in its source position to be read out from the display buffer. These words are passed through a barrel shifter 66, where the word is rotated as necessary and stored in appropriate byte locations in a FIFO buffer 68. The FIFO buffer has the capability to write to two word addresses simultaneously so that data words are created in the FIFO buffer which contain data in the correct alignment for writing unmodified into the corresponding display

buffer words. When the source has been completely read or the buffer is full (whichever comes first) the control logic automatically switches from reading to writing and stores the burst of data words it has in its FIFO buffer into memory at the destination locations.



DISPLAY SYSTEM

The present invention relates to a display system comprising a display buffer for data indicative of picture elements with a facility for moving an image from one position to another by manipulating the data in the display buffer.

In order to move images (eg. a character) around or onto a display screen a technique commonly called "bit blit" or "bit blt" has been developed. Essentially, the technique involves the provision of logic for automatically copying data indicative of the picture elements (pels) forming an image in response to commands (eg. using a mouse) identifying the image in its original, or source, position and in its destination position. In conventional bit-blit implementations a memory location is read and the data thus obtained is then written to another memory location thus copying one pel of the image. By incrementing the source (read) and destination (write) addresses appropriately, and repeating the operation, images (eg text or graphical characters) of any size can be moved a pel at a time.

This process can be used, both for copying images from one position to another in a part of a display buffer which is actually scanned in order to generate a screen of displayed images, (ie. in an on-screen portion of the display buffer) and for copying images between an off-screen (ie. non visible) and an on-screen portion of a display buffer. This latter technique is used in all points addressable (APA) display buffers for copying symbol definitions which are held in an off-screen part of a display buffer into an on-screen part of the display buffer when the symbols are actually to be displayed.

Conventionally, this is done by the system microprocessor which may use string move instructions if it is provided with them. The rates of blit obtained are, however, relatively low because of the limited bus bandwidth of microprocessor system and the very large amounts of data involved with large displays. As a consequence of this, it is not normally possible to move the image (unless it is very simple) around the screen in real time in response, for example, to mouse movements. Also a not inconsiderable time is taken to move an image from off-screen to on-screen storage.

The object of the present invention is to provide a display system with a mechanism for copying an image from a source position to a destination position which does not suffer from the disadvantages of prior art mechanisms.

This object is met by a display system comprising a word organised display buffer for storing

bytes of data indicative of picture elements (pels) with a plurality of bytes of data stored in each buffer word, and means for copying an image from a source position to a destination position, said means for copying comprising means for reading one or more bursts of source words from a first set of word locations in the display buffer, the source words including bytes of data indicative of pels for the image in its source position, means for rearranging the bytes of data in a burst of read source words so as to form a set of destination words in which the bytes of data indicative of pels for the image are in destination order within the words and means for writing the destination words generated from each burst of read source words in a write burst to a second set of word locations in the display buffer such that the bytes of data indicative of the pels for the image are stored in the correct display buffer byte locations for representing the image in its destination position.

In a preferred embodiment of the invention the means for rearranging the bytes of data in a burst of source words comprises a word-wide barrel shifter for rotating a data word by a selectable integral number of bytes, a plurality of word-wide registers organised as a first-in-first-out (FIFO) buffer for temporarily storing said bytes of data indicative of pels for the image, FIFO buffer addressing logic such that individual bytes within the registers can be addressed, and control logic for providing control signals to the barrel shifter and to the FIFO buffer addressing logic such that the bytes of data in a burst of read source words are rearranged so as to form a set of destination words in which said bytes of data indicative of pels for the image are in destination order within the words.

In a first example the barrel shifter is connected to the data output of the display buffer for receiving a source word read out of the display buffer and for rotating the source word by a selectable integral number of bytes. The data input of the FIFO buffer is connected to the output of the barrel shifter for receiving a rotated word and the data output of the FIFO buffer is connected to the data input of the display buffer. The control logic provides control signals to the barrel shifter and to the FIFO buffer addressing logic such that any byte in a rotated word which was rotated across the word boundary in the barrel shifter is stored in the FIFO buffer in an appropriate byte location in a register adjacent to the register in which is stored any byte in the rotated word which was not rotated across the word boundary in the barrel shifter, whereby the set of destination words in which said bytes of data indicative of pels for the

image are stored in destination order within the words is stored in the FIFO buffer.

In an alternative example the FIFO buffer is connected instead to the data output of the display buffer for receiving a source word read out of the display buffer. The input to the barrel shifter is connected to the output of the FIFO buffer for receiving a data word from the FIFO buffer to be rotated by a selectable integral number of bytes and the output of the barrel shifter is connected to the data input of the display buffer. The control logic provides control signals to the barrel shifter and to the FIFO buffer addressing logic such that any byte of said data word which is to be rotated across the word boundary in the barrel shifter is read from an appropriate byte location in a register adjacent to the register from which is read any byte in said data word which is not to be rotated across the word boundary in the barrel shifter, whereby the set of destination words in which said bytes of data indicative of pels for the image are in destination order is formed by the rotated word at the output of the barrel shifter.

In the embodiments of the invention mentioned above, the control logic comprises read logic for defining a first set of display buffer addresses from which to read the source words and write logic for defining a second set of display buffer addresses to which to write the destination words. The read logic causes a sequence of addresses from said first set to be generated, for a burst of source words to be read, until the FIFO buffer is full or until the words in which bytes of data indicative of pels for the image in the source position are exhausted, whichever occurs first, and then the write logic causes a sequence of addresses from said second set forming a burst of write addresses to be generated until FIFO buffer is emptied of bytes of data indicative of the image. In addition, the control logic can synchronise the addressing of the display buffer and the operation of the means for rearranging the bytes of data in a burst of read source words such that, after an appropriate number of bursts of source words have been read, rearranged and stored in the destination locations in the display buffer, the image is copied from the source position to the destination position.

A display system as above may include a display buffer which comprises an on-screen portion for storing bytes of data indicative of pels to be displayed on a display screen and an off-screen portion for bytes of data indicative of pels which are not displayed on a display screen. In this case the means for copying an image from a source position to a destination position will enable enable an image to be copied within one of the on-screen or off-screen portions and to be copied between said portions in either direction.

With a display system in accordance with the present invention alpha numerics, text and even graphics images can be moved around a display screen in real time (eg in response to mouse movements) which can facilitate the editing of screens containing images of all types (eg for page composition in desk-top publishing). It is possible to rapidly copy images from an off-screen portion to an on-screen portion of an all points addressable (APA) display buffer in order to display symbols (eg. characters) whose pel definitions are held in the off-screen portion. In this way a screen of images, including alpha numerics, text and even graphics, can be created on a graphics display which is driven using an APA graphics display buffer with the performance normally associated with coded buffer alphanumeric only displays.

In order to enable a more complete understanding of the present invention, there follows a particular description of a specific embodiment of a display system in accordance with the present invention with reference to the accompanying drawings in which:

Figure 1 is a block diagram of a workstation;

Figures 2A and 2B are schematic diagrams illustrating, respectively, the relationship between perceived picture element (pel) positions on a display screen and the corresponding storage positions in a display buffer for data indicative of those pels;

Figure 3 is a schematic block diagram showing aspects of a display adapter included in Figure 1; and

Figures 4 and 5 form a flow diagram illustrating aspects of the operation of the display adapter of Figure 3.

Figure 1 shows an overview of a workstation including a display adapter in which the invention may be implemented. The workstation comprises a number of different system units connected via a system bus 12. The system bus comprises a data bus 14, an address bus 16 and a control bus 18. Connected to the system bus is a microprocessor 10, random access memory 20, a keyboard adapter 28, a display adapter 32, an I/O adapter 22 and a communications adapter 26. The keyboard adapter is used to connect a keyboard 30 to the system bus. The display adapter, which forms a specific embodiment of the present invention, connects the system bus to a display device 34. The I/O adapter likewise provides a connection between other input/output devices 24 (eg. DASDs) and the system bus, and the communications adapter allows the workstation to be connected to and to communicate with an external processor or processors such as a host processor (not shown).

The display adapter includes an all points addressable (APA) display buffer 36 which can be

accessed by the display device in order to fetch the data corresponding to the individual picture elements on the screen (see Figure 2A). The data are fetched in synchronism with the scanning of the display screen. To facilitate this the information in the display buffer is organised in accordance with the scanning sequence of the display refresh circuitry. In addition to the on-screen, or visible portion of the display screen which is scanned by the display device, the display buffer also comprises an off-screen, or non-visible portion in which are stored images (eg. character or symbol definitions) which are not visible on the screen.

Figure 2A illustrates the display screen 38 as perceived. The screen has "Y" lines of "X" pel positions each. As shown the rows of the screen are numbered from 0 to Y1 starting from the top and moving down the screen. The pel positions in each row are likewise shown numbered, from left to right, 0 to X1. It will be appreciated that the numbering is purely an arbitrary choice and that other numbering schemes could have been chosen.

In order to generate the screen shown in Figure 2A, data is stored in a display buffer 36, the on-screen portion of which is represented in Figure 2B.

This specific display buffer is organised on words of 32 bits. Each pel is represented by an 8-bit byte of data defining the intensity and/or colour of the pel. It will be understood however, that other display buffer organisations are possible. For example, the buffer could be organised on words of another length and/or a different number of bits could make up a byte defining the intensity and/or colour of a pel (eg. 8 bit words and 4 bit bytes).

It will be noted that the display buffer is shown, for reasons of simplicity of explanation, as being organised from the bottom up as a linear stream of bytes within the buffer starting with the byte for the top left-most pel, followed by the bytes for the pels, from left to right, in the first row, the second row, and so on until the byte for the bottom right-most pel. The display buffer could be organised for a screen which is refreshed using an interleaved scanning technique (i.e. where the scanning of even numbered rows alternates with the scanning of odd numbered rows) by arranging for the image data for the individual pels of the display screen shown in Figure 2A to be stored with the data relating to the even numbered rows stored in a first sequence from a first address in the display buffer and the data relating to the odd numbered rows stored in a second sequence starting at a second address at or after the end of the first sequence. Alternatively, the display buffer could be organised as a single sequence as shown in Figure 2B and the display scanning logic provided with the capability to extract the data it needs from the buffer in

order to support an interleaved scanned display screen.

Figure 2A shows two rectangular blocks 40 and 42 which represent the source position 40 and the destination position 42 on the screen for a rectangular image to be moved. The rectangular image comprises a row of 6 pels, the top-left of which is initially located at the screen position b,a. The top-left of the destination position is d,c. Figure 2B illustrates the source rectangle to be represented in the display buffer by a set of 6 bytes of data at 41. Likewise the destination rectangle is represented in the display buffer by a set of 6 bytes of data at 43. It will be noted that the alignment of the sets of data for the source and destination rectangles with respect to the word boundaries of the display buffer (ie. the left/right edges of the block at bit address 0/31 shown in Figure 2B) is different. It should be noted that a simple image lying on one row only of pels has been illustrated for reasons of ease of explanation. In practice the images to be moved will normally be larger and will normally cover a plurality of rows.

The display buffer in this adapter is formed from a 32 bit wide array of D-RAMS. The minimum addressing unit is one 32 bit word. The display buffer is, however provided with separate byte enables so that 0 to 4 bytes (each of 8 bits) may be written to as required for each write access with the remaining of the four bytes remaining unchanged. In this way the display buffer can be configured as an APA buffer so that each pel (ie. each 8-bit byte) can be selectively written to or not as the case may be even though four bytes (ie. a word) at a time are (is) addressed.

In view of this it would be possible to apply the prior art approach to moving an image on a display screen (ie. to successively read single bytes and each time to write the data thus obtained to the equivalent destination location in the buffer, thereby copying successively single pels until the whole image has been copied). This approach is, however, very slow because each byte is processed in turn. It would not be possible to simply read and write a word at a time because the position of a byte within the word boundaries may be different in the source and destination words as illustrated in Figure 2B.

The present invention provides a solution to this problem by providing means for reading, processing and storing bursts of display buffer words.

Although, in the specific embodiment of the invention described herein the display buffer is implemented in D-RAMS and can be accessed on both word and byte boundaries, the present invention is not limited to such an organisation. It can equally be implemented with a display buffer in some other technology and where the buffer can

only be accessed on the word boundaries and not on the byte boundaries.

If the display buffer is implemented in D-RAMS, however, and if consecutive words in the display buffer are arranged to define consecutive sets of four pels, the burst operation of the present invention allows, for example, a D-RAM "page mode" feature to be utilised which provides a further significant performance advantage. "Page mode" is a feature conventionally provided in D-RAMS which, as a consequence of their internal chip organisation, allows faster access to sequential locations than to random locations.

Figure 3 is a schematic block diagram showing the interrelationship between various functional elements in the display adapter shown in Figure 1 which forms the specific embodiment of the present invention. The adapter comprises a control unit 44 which is connected to the address bus 16 and to the control bus 18. Associated with the control unit is control storage 46 which is connected to the data bus 14 for the receipt of initialisation data from the workstation RAM.

The inputs to Y and X source registers 48 and 50 and Y and X destination registers 52 and 54 are also connected to the data bus 14 for receiving initial position data as part of the initialisation data to be explained later. Connected to the outputs of the Y and X source and the Y and X destination registers are Y and X source and Y and X destination counters 56, 58, 60, 62, respectively. An arithmetic logic unit 64 has four inputs each connected to the output of a respective one of the four counters 56, 58, 60, 62. The output of the arithmetic unit is connected to the address input of the display buffer 38.

The data outputs of the display buffer 36 are connected in parallel to the inputs of a barrel shifter 66 known per se, and whose purpose is to rotate a data word input thereto by an integral number of byte positions. The data outputs of the barrel shifter are connected in parallel to the data inputs of a FIFO buffer 68 which is formed from a plurality (in this display adapter, eight) word-wide registers. The FIFO buffer is provided with addressing logic 70 such that individual bytes in the FIFO buffer may be separately addressed and such that individual bytes presented simultaneously on the data inputs to the FIFO buffer may be stored in different words. The FIFO buffer addressing logic contains byte pointers for indicating the start and finish of data in the FIFO buffer, whereby the FIFO buffer addressing logic can determine when the buffer is full, and when it is empty. The data outputs of the FIFO buffer are connected back to the data inputs to the display buffer. Although shown as being separate, the data inputs and outputs of the display buffer may in fact be common -

likewise for the FIFO buffer.

The control unit is connected via control inputs C to the control storage, the Y and X registers, the Y and X counters, the arithmetic logic unit the display buffer, the barrel shifter and the FIFO buffer address logic. The FIFO buffer address logic is also connected to the control logic for passing "buffer full" and "buffer empty" signals.

In this particular display adapter the various functional units shown are provided in the form of a special purpose circuitry. The logic units for example are provided by combinatorial logic. The present invention does not however, exclude the possibility of implementing the logic in a software-and-processor-based system.

An image to be copied is processed by the display adapter in bursts. The control logic first causes a burst of display buffer words containing bytes representative of the image in its source position to be read out from the display buffer. These words are passed through the barrel shifter, where the word is rotated as necessary and stored in appropriate byte locations in the FIFO buffer. The FIFO buffer has the capability to write to two addresses simultaneously to obtain the desired result which is that the buffer locations match the destination display buffer word locations byte for byte and indeed, bit for bit.

When the source has been completely read or the FIFO buffer is full (whichever comes first) the display adapter automatically switches from reading to writing and stores the burst of data words it has in its FIFO buffer into the display buffer at the destination locations. It should be noted that the processes described above cause the buffer to contain data in the format it is required for writing. That is each buffer word contains data in the correct alignment for writing unmodified into the corresponding display buffer words.

In any case where the source rectangle is too large to fit into the FIFO buffer the display adapter automatically switches back and forth between read bursts filling the FIFO buffer and write bursts dumping the FIFO buffer into the destination display buffer locations. The size of the X dimension of the image to the size of the FIFO buffer is irrelevant to this process. A large X dimension will result in many bursts being required to move one line of the image. A small X dimension will result in many lines of the image being transferred in a single burst. The buffer is therefore logically invisible.

In the example illustrated in Figure 2A and Figure 2B the source image is read completely in a burst of two reads. (It will be understood that in a trivial case where the image is four by one pels in size, the burst might only include one read). The control logic calculates from the source and des-

termination X addresses that a rotation of one to the right is required in order to get the pels into the positions within the destination word in which they will be required during the write burst. Also further logic calculates that pels occupying the last position within the word need to be written in the FIFO buffer one location ahead of pels in the first, second and third positions. Although the bytes "b,a+2", "b,a+1", " b,a" lie in the same display buffer word at the source position for the image (41, Figure 2A), it can be seen that they the equivalent bytes "d,c+2", "d,c+1", " d,c" lie in different (adjacent) memory words at the destination position (43, Figure 2B).

In order to explain more fully the manner in which an image is moved on the display screen of a display system according to the present invention, reference will be made to the flow diagrams in Figures 4 and 5 and to Figures 2A and 2B.

Before the display adapter can actually modify the data in the display buffer in order to update the display screen, an initialisation phase is, however, necessary.

In the workstation of Figure 1, the processor initialises the display adapter by sending positioning data to the adapter over the data bus. The positioning data comprise image size information, information about the initial, or source position and the final, or destination position for that image and scan direction information. These data can result from operations using a mouse or instructions from a keyboard, and so on, and are supplied over the data bus to the control storage.

The source and destination position information comprises the "x" and "y" values for the screen position of one corner of the image (eg. the bottom right hand corner) in its source and destination positions. The image size information defines the length of the horizontal and vertical sides of the image in terms of numbers of pels. The scan direction information determines the directions in which the source and destination image positions should be scanned from the indicated screen position in order to avoid corrupting data. If the image positions are scanned in the wrong directions when the image positions overlap it might be that a location could receive new pel data before the old pel data had been read out.

The size and position information in the case of the block in Figure 2 is as follows:

source position information x = a+5,
y = b;
destination position information x = ..
c+5, y = d;
size information horizontal 6 pels, vertical 1
pel;
scan direction information ... y decreasing, x

decreasing.

In order to compute the scan direction information, it is necessary to determine in which direction the pels of the image need to be processed (ie in which direction the image needs to be scanned) in order that data shall not be corrupted in the display buffer. Figures 2A and 2B illustrate a case where the source and destination rectangles do not overlap, though in practice it is common for images in the source and destination positions to overlap when an image is copied within the on-screen portion of the display buffer. The aim is to avoid the need to write to a location which contains a byte of pel data which has not yet been read out of the display buffer. This can be done by comparing the source and destination positions of the image. The direction of scanning of the image in the X or Y direction, or both as appropriate, will be opposite to the direction of movement of the image. If, for example, destination position of the image is to the right of the source position (ie. the direction of movement is to the right), the scanning will be from right to left.

The source and destination position information mentioned above identifies the position the screen location from which the "scanning" of the rectangles should start. These values can be calculated from the image position information, the image size information and the scanning direction information.

In the example shown in Figure 2, as both the "x" and "y" values of the destination position are higher than those of the source position it is possible to compute that the image should be scanned from bottom right to top left. Also, if the positions of the source and destination images are defined by their top left hand corners (ie. b,a and d,c) at some stage during processing of the images, it is possible to compute the bottom right hand pel positions. In the present workstation these computations are performed by software under the control of the processor and not in the adapter. In an alternative embodiment of the invention it would be possible for special purpose combinatorial logic to be incorporated for this in the display adapter. In this case the initial Y and X values would first be read into the control storage and the computed values then passed via the dashed line shown in Figure 3 between the control storage and the registers. In either case, an indication of the scanning direction is stored in the control storage and the initial Y and X positions for the source and destination images are stored in respective ones of the registers 48, 50, 52 and 54. The values stored in the registers for the example in Figures 2 are as follows:

Source Y register b;
Source X register a+5;

Destination Y register d;
 Destination X register c + 5.

Specific combinatorial logic is provided in the control logic for determining the number of bytes of rotation which needs to be performed by the barrel shifter in order to align the bytes of the source words within the destination words. The rotation required is calculated by subtracting the value of the source X position from the destination X position and truncating the result to the number of bits required to describe the number of pels in the word. This assumes, as is indeed the case, that there is an integral number of words per row on the display screen and that the number of pels per word is an integral binary power (eg. 4 as in this case). In the present example the source address is "a+5" whilst the destination address is "c+5" and the number of bits required to describe the number of pels in one memory word is 2 (ie. there are 4 bytes per word). Thus the calculation is:

$$\text{Rotation} = (c - a) \text{ trunc } 2.$$

As illustrated in Figure 2B, the rotation necessary is in fact one byte to the left. This rotation value is stored in the control storage and supplied to the barrel shifter.

Once this initialisation information has been determined and stored, the display adapter is then ready to start the operations on the display buffer. All the calculations described above are performed directly by combinatorial logic and do not therefore take time either in the microprocessor during setting up of the display adapter hardware or during operation of the display adapter hardware.

Figure 4 illustrates the logical flow of operations performed by read burst logic in the control logic and Figure 5 illustrates the logical flow of operations performed in write burst logic in the control logic during the modification of the display buffer contents. These flow diagrams are not intended to suggest that each step shown in the Figures relates to a separate machine state. Some steps involve a plurality of operations and others are in practice done in parallel by combinatorial logic. The aim in any implementation is to minimise the number of clock states.

IN STEP 80 the screen position value, b, for the y ordinate of the bottom right-most part of the image in its source location is loaded into the source Y counter 56 from the source Y register 48.

IN STEP 81 the screen position value, a + 5, for the x ordinate of the bottom right-most part of the image in its source location is loaded into the source X counter 58 from the source X register 50.

IN STEP 82 the control logic tests whether the FIFO buffer is full. This is done by examining the control output from the FIFO buffer addressing logic.

IN STEP 83 (ie. the FIFO buffer is full) control is passed to the logic controlling a write burst as set out in Figure 5 and expanded later.

IN STEP 84 on returning from a write burst, or alternatively if the FIFO buffer was not full, the control logic causes the arithmetic logic to compute the address of the word in the display buffer containing the byte of data for the pel to be displayed on the screen at position indicated by the values currently in the source Y and X counters. The actual computation to be performed will depend on the display buffer organisation, the number of pels per word and so on.

IN STEP 85 the control logic determines whether the word address just calculated is a new source word address. If the address just calculated is the same as the last word address calculated for the source image, control passes to step 87.

IN STEP 86 (ie. the address just calculated was new) the control logic then causes the display buffer word pointed to by the arithmetic logic to be read from the display buffer into the barrel shifter where it is rotated by the number of bytes determined to be necessary by the control logic. The barrel shifter is formed from combinatorial logic and consequently allows pels to be rotated from their position in the source word to their position in the destination word without extra clock cycles.

The rotated word appears at the output of the barrel shifter and is read into appropriate byte locations in the FIFO buffer as addressed by the FIFO addressing logic under the control of the control logic. It will be remembered that the byte locations in the registers can be separately enabled whereby the FIFO addressing logic can address appropriate bytes in two adjacent registers at once. Any byte in a rotated word which was rotated across the word boundary in the barrel shifter is stored in the FIFO buffer in an appropriate byte location in a register adjacent to the register in which is stored any byte in the rotated word which was not rotated across the word boundary in the barrel shifter. In the present example, the number of bytes per word is 4 and the rotation is one so only pels occupying byte 3 within the source word are rotated across the word boundary (ie. bytes in the left-hand word position as shown in Figure 2B).

IN STEP 87, following on from step 86, or step 85 in the case the display buffer source word address was the same as the previously determined display buffer source word address, the control logic determines from the horizontal length of the image rectangle (ie. the horizontal size) whether any more display buffer addresses calculations are to be performed for the current row of pels. This is done by maintaining a count in the control logic of the number of pels processed in

the current row.

If there are, the control logic decrements the source X counter as "x decrementing" was stored in the control storage. The control logic then returns to step 82.

If there are no more pels in the current row of the image, the logic proceeds to step 88.

IN STEP 88 the control logic determines from the vertical length of the image rectangle (ie. the vertical size) whether there are any more rows to be processed for the image. This is done by maintaining a count in the control logic of the number of rows processed for the current image.

If there are, the control logic decrements the source Y counter as "y decreasing" indication was stored in the control storage. The control logic then returns to step 81 and causes the position value, $a+5$, for the x ordinate to be loaded from the source X position register into the source X position counter.

IN STEP 89 the control logic passes control to the write burst logic as there are no more bytes of data relating to image pels in the source position in the display buffer. When control returns from the write burst logic, the image copying operation is complete and the control logic terminates its operation.

IN STEP 90 on entering a write burst, control is either passed to step 91 on the first entry into a write burst during processing of an image, or to step 95 otherwise.

IN STEP 91 the screen position value, d, for the y ordinate of the bottom right-most part of the image in its destination location is loaded into the destination Y counter 60 from the destination Y register 52.

IN STEP 92 the screen position value, $c+5$, for the x ordinate of the bottom right-most part of the image in its destination location is loaded into the destination X counter 62 from the destination X register 54.

IN STEP 93 the control logic tests whether there is another FIFO buffer word containing bytes to be written into a destination word location in the display buffer. This is done by examining the control output from the FIFO buffer addressing logic.

IN STEP 94 (ie. there are no more FIFO buffer words to be read out) control is returned to either step 83 or 89 in the read burst logic from which control was passed to the write burst logic.

IN STEP 95 (ie. there are FIFO buffer words to be written into the display buffer) the control logic causes the arithmetic logic to compute the address of the word in the display buffer containing the byte of data for the pel to be displayed on the screen at the position indicated by the values currently in the destination Y and X counters. The actual computation to be performed will depend on

the display buffer organisation, the number of pels per word and so on.

IN STEP 96 the control logic determines whether the address just calculated is a new display buffer address. If the address just calculated is not new, control passes to step 98.

IN STEP 97 (ie. the address just calculated was new) the control logic then causes the next FIFO buffer word to be written to the display buffer word pointed to by the arithmetic logic. In practice only bytes in the word which form part of the image are written to the display buffer. The control logic only enables the locations in the destination word for bytes indicative of pels of the image to be written within the display buffer address pointed to by the arithmetic logic, in order to avoid corrupting bytes in the display buffer not forming part of the image. The control logic keeps track of which bytes are to be written by counting the bytes in the FIFO buffer.

IN STEP 98, following on from step 97, or step 96 in the case the display buffer destination word address was the same as the display buffer destination word address last calculated, the control logic then determines from the horizontal length of the image rectangle (ie. the horizontal size) whether any more display buffer destination word calculations are to be performed for the current row of pels. This is done by maintaining a count in the control logic of the number of pels processed in the current row.

If there are, the control logic decrements the destination X counter as "x decreasing" was stored in the control storage. The control logic then returns to step 93.

If there are no more pels in the current row of the image, the logic proceeds to step 99.

IN STEP 99 the control logic decrements the destination Y counter as a "y decreasing" indication was stored in the control storage. The control logic then returns to step 92 and causes the X position value, $c+5$, for the x ordinate to be loaded from the destination X position register into the destination X position counter. It is not necessary to determine whether there are any more rows to be processed for the image as the FIFO buffer will be emptied of bytes before this can happen.

Although a particular embodiment of the invention as claimed has been described it will be appreciated numerous modifications and alternative structures are possible within the scope of the invention as claimed.

For example the positions of the barrel shifter and the FIFO buffer could be reversed so that the words of data indicative of the image in its source location are read straight into the FIFO buffer, and then the special addressing mechanism of the FIFO buffer used to compile data words which can

be fed to the barrel shifter where they are rotated to form destination words and then stored in the destination locations in the display buffer. In such an embodiment of the display system the FIFO buffer is connected to the data output of the display buffer for receiving a source word read out of the display buffer. The input to the barrel shifter is connected to the output of the FIFO buffer for receiving a data word from the FIFO buffer to be rotated by a selectable integral number of bytes and the output of the barrel shifter is connected to the data input of the display buffer. The control logic provides control signals to the barrel shifter and to the FIFO buffer addressing logic such that any byte of said data word which is to be rotated across the word boundary in the barrel shifter is read from an appropriate byte location in a register adjacent to the register from which is read any byte in said data word which is not to be rotated across the word boundary in the barrel shifter. In this way the set of destination words in which said bytes of data indicative of pels for the image are in destination order is formed by the rotated word at the output of the barrel shifter.

In the particularly described embodiment the area displayed on the screen is rectangular. If desired, however, the adapter could also be provided with the facility to copy images which are not rectangular by incorporating masking logic in the adapter. In simple terms this could be achieved in the workstation of Figure 1 by transferring mask boundary information from the workstation RAM to the off screen portion of the display buffer and subsequently copying image data for a rectangular area as described in the preceding description. In this case however the control logic would cause the data items relating to screen positions outside the mask boundary to be disregarded so that only that part of the rectangular image area within the boundary would be displaced.

As an alternative to merely writing bytes of data into the display buffer, it may be desired to logically combine the byte indicative of a pel for the image with the byte already stored at the destination location in order to achieve a particular effect on the display screen. This can be achieved by providing further arithmetic logic to allow arithmetic and logical operations to be performed between the source and destination locations. Each pel could effectively be processed separately in such logic although several pels could in practice be processed concurrently. The operation on each pel could be of the type:-

Destination := Source (Operation) Destination

where the operation could be an exclusive-or, a logical AND, and so on.

In addition, scissoring logic may be employed to protect areas of the display buffer from being written to (but not from being copied). For example, areas outside a window on the display screen, or areas in off-screen storage can be protected in this way.

In the above, the display adapter shown in part in Figure 3 has been identified as a specific embodiment of a display system according to the invention, as rightly it is. However, the term display system as used in the claims is not limited thereto. The workstation including the display adapter could equally be described as a display system. Consequently it should be understood that the term display system as used herein includes within its scope, a workstation, or indeed any other system falling within the scope of the claims, whether or not it is possible to separately identify a display adapter in that system. Also, although the logic in the specific embodiment of the invention is in the form of combinatorial hardware logic, the present invention includes within its scope that one or more of the functions provided by that logic is implemented in software.

Claims

1. A display system comprising a word organised display buffer (36) for storing bytes of data indicative of picture elements (pels) with a plurality of bytes of data stored in each display buffer word, and means for copying an image from a source position to a destination position, said means for copying comprising means (48, 50, 56, 58, 64) for reading one or more bursts of source words from a first set of word locations in the display buffer, the source words including bytes of data indicative of pels for the image in its source position, means (44, 66, 68, 70) for rearranging the bytes of data in a burst of read source words so as to form a set of destination words in which said bytes of data indicative of pels for the image are in destination order within the words and means (52, 54, 60, 62, 64) for writing the destination words generated from each burst of read source words in a write burst to a second set of word locations in the display buffer whereby the bytes of data indicative of the pels for the image are stored in the correct display buffer byte locations for representing the image in its destination position.

2. A display system as claimed in claim 1 in which the means for rearranging the bytes of data in a burst of source words comprises a word-wide barrel shifter (66) for rotating a data word by a selectable integral number of bytes, a plurality of word-wide registers organised as a first-in-first-out (FIFO) buffer (68) for temporarily storing said bytes

of data indicative of pels for the image, FIFO buffer addressing logic (70) such that individual bytes within the registers can be addressed, and control logic (44) for providing control signals to the barrel shifter (66) and to the FIFO buffer addressing logic (70) such that the bytes of data in a burst of read source words are rearranged so as to form a set of destination words in which said bytes of data indicative of pels for the image are in destination order within the words.

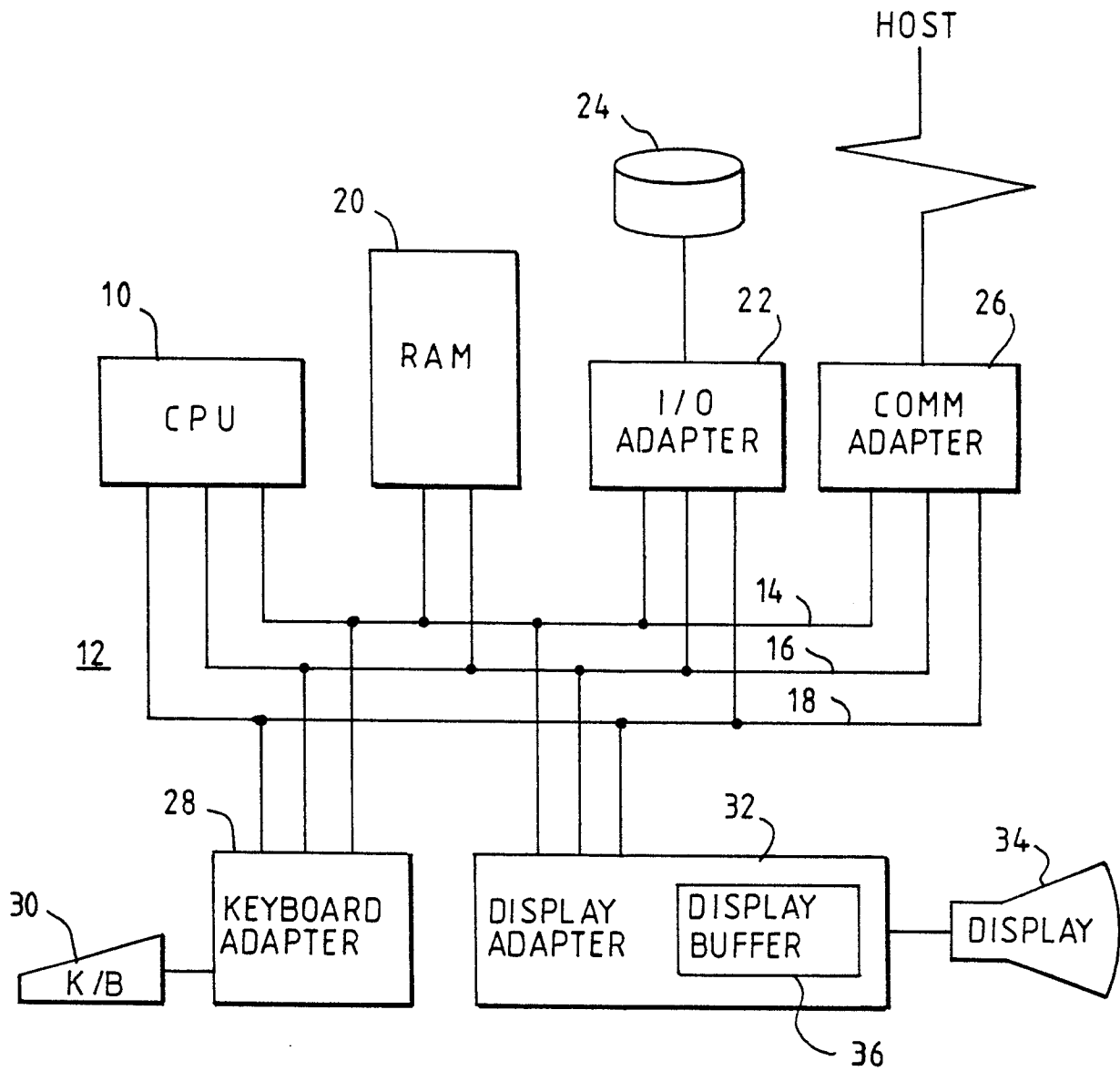
3. A display system as claimed in claim 2 in which the barrel shifter (66) is connected to the data output of the display buffer (36) for receiving a source word read out of the display buffer (36) and for of the FIFO buffer (68) is connected to the output of the barrel shifter (66) for receiving a rotated word and the data output of the FIFO buffer (68) is connected to the data input of the display buffer (36), and in which the control logic (44) provides control signals to the barrel shifter (66) and to the FIFO buffer addressing logic (68) such that any byte in a rotated word which was rotated across the word boundary in the barrel shifter is stored in the FIFO buffer (68) in an appropriate byte location in a register adjacent to the register in which is stored any byte in the rotated word which was not rotated across the word boundary in the barrel shifter (66), whereby the set of destination words in which said bytes of data indicative of pels for the image are stored in destination order within the words is stored in the FIFO buffer (68).

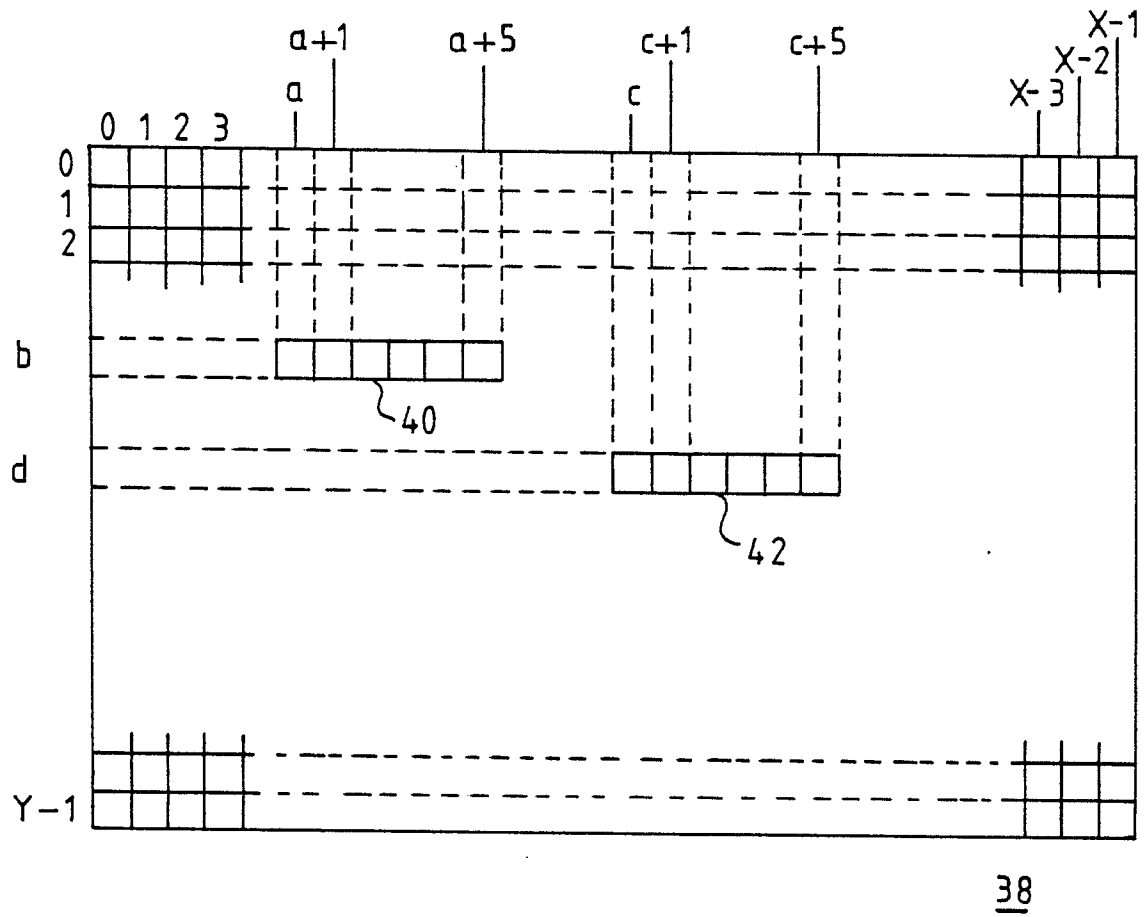
4. A display system as claimed in claim 2 in which the FIFO buffer (68) is connected to the data output of the display buffer (36) for receiving a source word read out of the display buffer (36), in which the input to the barrel shifter (66) is connected to the output of the FIFO buffer (68) for receiving a data word from the FIFO buffer (70) to be rotated by a selectable integral number of bytes and the output of the barrel shifter (66) is connected to the data input of the display buffer and in which the control logic provides control signals to the barrel shifter and to the FIFO buffer addressing logic (70) such that any byte of said data word which is to be rotated across the word boundary in the barrel shifter (66) is read from an appropriate byte location in a register adjacent to the register from which is read any byte in said data word which is not to be rotated across the word boundary in the barrel shifter (66), whereby the set of destination words in which said bytes of data indicative of pels for the image are in destination order is formed by the rotated word at the output of the barrel shifter (66).

5. A display system as claimed in claim 3 or claim 4 wherein the control logic comprises read logic (fig 4) for defining a first set of display buffer addresses from which to read the source words

and write logic (fig 5) for defining a second set of display buffer addresses to which to write the destination words, and wherein the read logic causes the display a sequence of addresses from said first set to be generated, for a burst of source words to be read, until the FIFO buffer is full or until the words in which bytes of data indicative of pels for the image in the source location are exhausted, which ever occurs first, and then the write logic causes a sequence of addresses from said second set forming a burst of write addresses to be generated until FIFO buffer is emptied of bytes of data indicative of the image, the control logic synchronising the addressing of the display buffer and the operation of the means for rearranging the bytes of data in a burst of read source words such that, after one or more bursts of source words have been read, rearranged and stored in the destination locations in the display buffer, the image is copied from the source position to the destination position.

6. A display system as claimed in any one of the preceding claims wherein the display buffer comprises an on-screen portion for storing bytes of data indicative of pels to be displayed on a display screen and an off-screen portion for bytes of data indicative of pels which are not displayed on a display screen and wherein the means for copying an image from a source position to a destination position enable an image to be copied within one of the on-screen or off-screen portions and to be copied between said portions in either direction.

FIG. 1

FIG. 2A

36

	Y-1,X-1	Y-1,X-2	Y-1,X-3	Y-1,X-4
ROW Y-1 {	⋮	⋮	⋮	⋮
	Y-1,7	Y-1,6	Y-1,5	Y-1,4
	Y-1,3	Y-1,2	Y-1,1	Y-1,0
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
43 {	d,c+5	d,c+4	d,c+3	d,c+2
	d,c+1	d,c		
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
41 {		b,a+5	b,a+4	b,a+3
	b,a+2	b,a+1	b,a	
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮
ROW 1 {	1,X-1	1,X-2	1,X-3	1,X-4
	⋮	⋮	⋮	⋮
	1,7	1,6	1,5	1,4
	1,3	1,2	1,1	1,0
ROW 0 {	0,X-1	0,X-2	0,X-3	0,X-4
	⋮	⋮	⋮	⋮
	0,7	0,6	0,5	0,4
	0,3	0,2	0,1	0,0

31 0

FIG 2 B

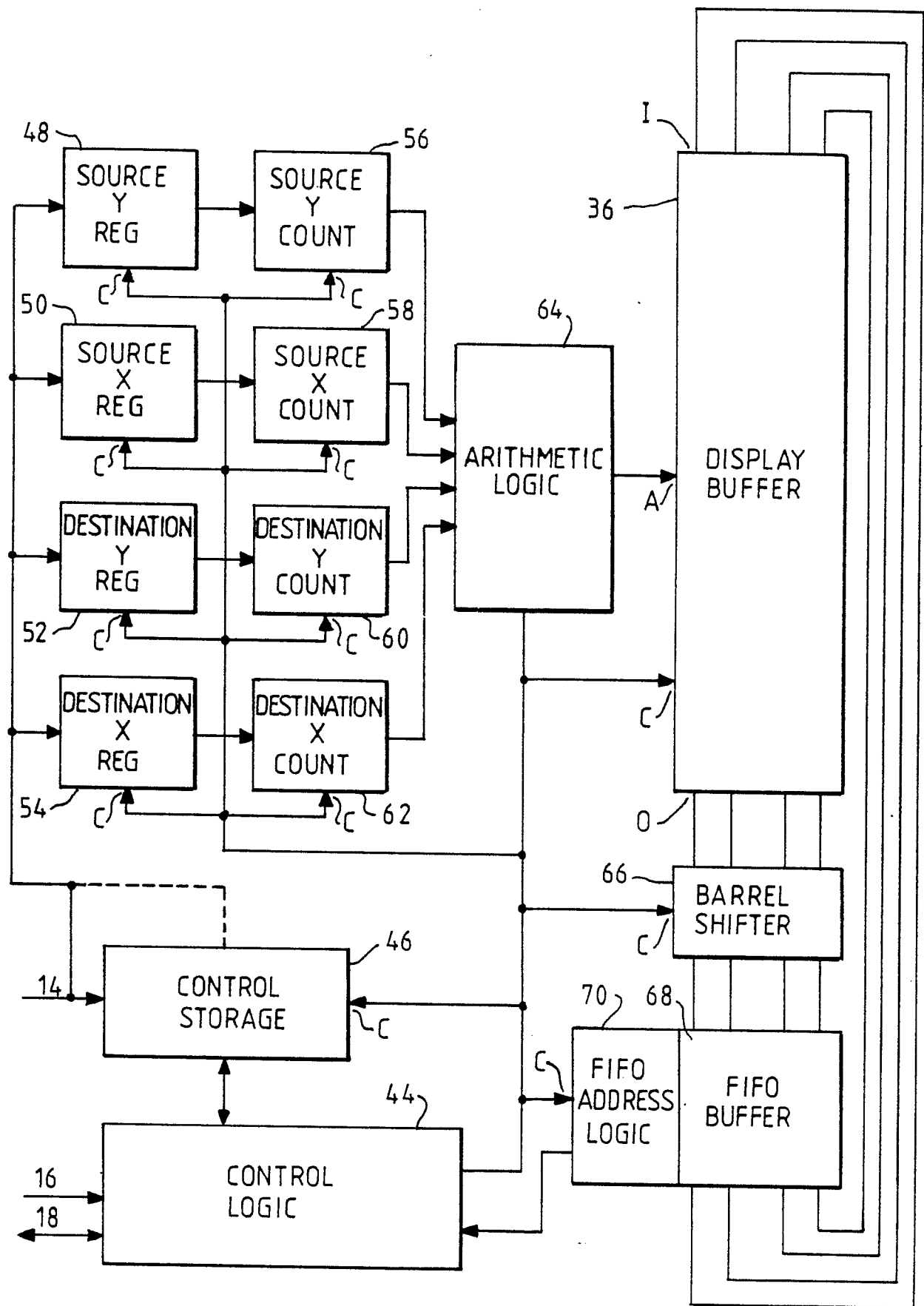
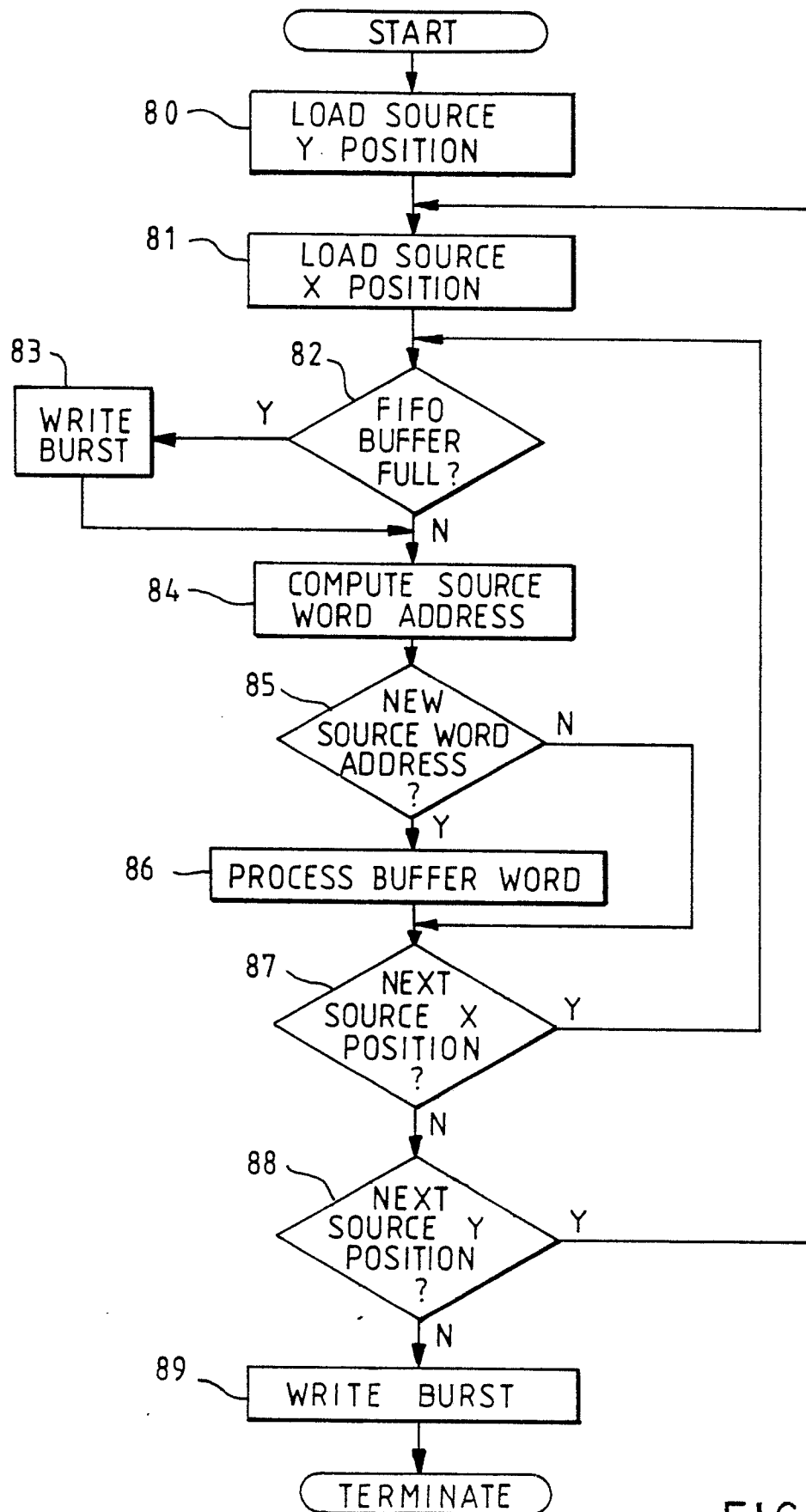
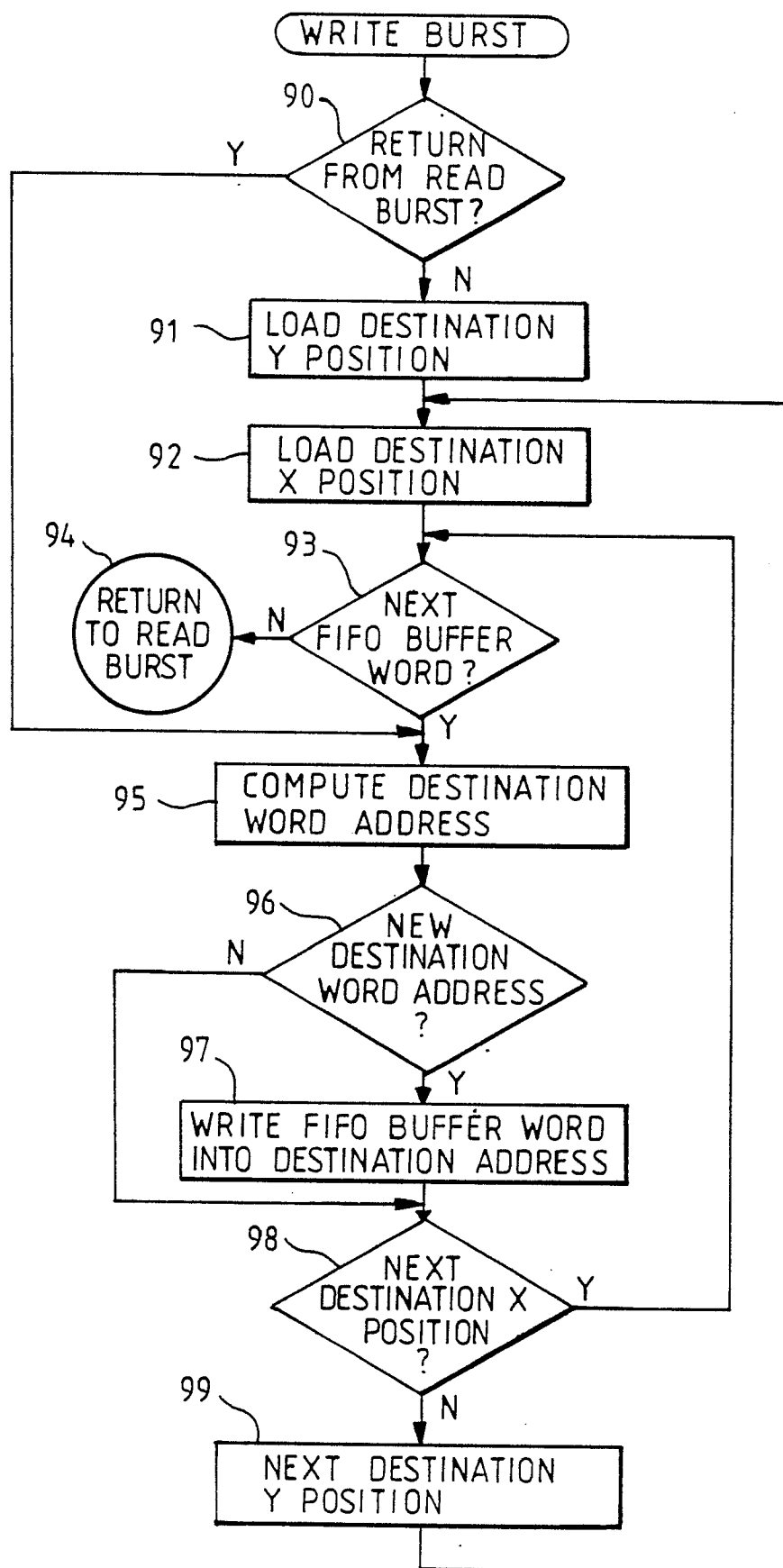


FIG. 3

FIG. 4

FIG. 5