(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 283 927 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **88104226.1**

(51) Int. Cl.[4]: **G09G 1/16**

(22) Date of filing: **17.03.88**

(30) Priority: **27.03.87 GB 8707409**

(43) Date of publication of application:
28.09.88 Bulletin 88/39

(84) Designated Contracting States:
DE FR GB

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

(72) Inventor: **Harrison, Roy Bernard**
5 Chichester Close
East Wellow Hampshire. SO5 OEY(GB)
Inventor: **Norris, Paul William**
5, Ashbridge Rise
Chandlers Ford Hampshire. SO51 1SA(GB)
Inventor: **Wood, Roger Timothy**
108 Woodley Lane
Romsey Hampshire(GB)

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

(54) Display adapter.

(57) A display adapter is provided for connecting the system bus 12 (Fig. 1) of a workstation to a display means 34 (Fig. 1) having a scanned display screen. The display adapter comprises input means, including means 48, 50, 46 for receiving positioning data specifying an area on the screen within which an image is to be displayed and means 54 for receiving a stream of data items representing the image. The individual data items in the received stream 42 (Fig. 2C) define respective pels of the image and the stream is ordered to define the image line-by-line and within each line, pel-by-pel. The display adapter also comprises a display buffer 38 for storing, in display scanning sequence, data defining the pels to be displayed on the display screen, and addressing logic 44, 49, 51, 58 for computing from the positioning data, individual storage locations (Fig. 2b) in the display buffer to be addressed for individual data items in the received stream in order that the data for the individual pels of the image are stored in the display buffer in said display scanning sequence and in order that the image is displayed correctly on the display screen.

The means for receiving a stream of data items preferably comprises a multiplexer which enables the stream of data items to be selectively supplied from the system bus or from a source external to the workstation.

EP 0 283 927 A2

DISPLAY ADAPTER

The present invention relates to a display adapter for connecting the system bus of a workstation to a display means having a scanned display screen and to a workstation comprising such an adapter.

The term "workstation" is a general term applied to computing equipment and includes both stand alone units such as a personal computer and units which are used in a host connect mode such as a terminal or a personal computer provided with a terminal emulator.

The heart of a workstation is a central processing unit such as a microprocessor. The processor is responsible for carrying a large range of different management tasks in addition to the processing of the user programs. One management task which has been carried out by the processor in prior workstations is the formatting of image data in preparation for display.

It is common for the display of a workstation to be a raster scanned display device such as a cathode ray tube device (CRT) which requires a refresh buffer. In order that the display may be scanned at a sufficiently high rate without resorting to expensive solutions, the image data for display is normally stored in the display buffer in accordance with the scanning cycle for the display. The data for the picture elements (pels) to be displayed is stored in a linear fashion in the display buffer. In many prior art systems the display buffer is in practice a part of the workstation's RAM.

To store new image information in the display buffer often involves complex calculations, even to store the image data for a small part of the screen image, because of the requirement to store the image data in accordance with the scanning cycle. Although it is usual for the data for each pel to comprise a number of bits of information (eg.4) defining colour, grey level, etc, a number of pels are usually stored in each byte of display buffer storage (eg. two pels can be stored per byte if each pel is defined by 4 bits). In such cases it is necessary to address part of a byte in order to access the data for a particular pel. Also, it is usual for the display to be refreshed using interleaved scanning on alternate lines in order to reduce perceived flicker. This has the effect that the data relating to adjacent lines of the display as seen may not be stored adjacently in the display buffer. As a result of the aspects mentioned above, the processor has a high processing overhead in creating the display.

The disadvantages of the prior art are overcome by the provision of a display adapter in accordance with the present invention for connecting the system bus of a workstation to a display means having a scanned display screen, the display adapter comprising:

input means for receiving positioning data specifying an area on the screen within which an image is to be displayed and for receiving a stream of data items representing the image, individual data items in the received stream defining respective pels of the image and the stream being ordered to define the image line-by-line and, within each line, pel-by-pel;

a display buffer for storing, in display scanning sequence, data defining the pels to be displayed on the display screen;

and addressing logic for computing from the positioning data, individual storage locations in the display buffer to be addressed for individual data items in the received stream in order that the data for the individual pels of the image are stored in the display buffer in said display scanning sequence and in order that the image is displayed correctly on the display screen.

In a particular embodiment of a display adapter according to the present invention the input means comprises register means connectable to the processor bus for receiving initial screen position information, control storage connectable to the processor bus for receiving size and orientation information which, in combination with the initial screen position information, form the positioning data defining the area on the screen within which the image is to be displayed and gate means connectable to the system bus for receiving the stream of data items representing an image to be displayed within that area. The addressing logic in this embodiment comprises counter means connected to the register means for receiving the initial screen position information contained therein, arithmetic logic connected to the counter means for receiving the instantaneous values of information contained therein and for generating display buffer address signals therefrom and control logic which is connected to the control storage and, in response to the size and orientation data stored therein causes the counter means to be loaded from the register means and/or the content of the counter means to be adjusted to identify the screen location for each successive data item, causes the arithmetic logic to perform appropriate arithmetic operations on instantaneous values contained in the counter means so as to generate display buffer

addresses for successive data items in the received stream and causes the data for the individual pels of the image to be stored in appropriate locations in the display buffer using the display buffer addresses so generated such that the image is displayed on the screen within said area.

Some processors, for example the 80286 processor manufactured by the Intel Corporation, are provided with block move instructions which are significantly faster for moving a block than are the corresponding series of individual instructions. However, the block move instructions cannot be used if the order of the data within the block has to be changed by the processor and consequently they cannot be used to move a block of image data arranged in perceived display order into a display buffer. A workstation with a display adapter according to the invention, however, does allow the block move instructions to be used. A stream of image data can be sent from the workstation RAM to the display adapter using a processor block move operation in the same way that blocks of information can be transferred to an I/O device such as direct access storage devices (DASDs). The display adapter can either process the stream of image data directly as it arrives, or, if the display adapter is additionally provided with an image buffer for the temporary storage of the stream, the processing of the stream can be performed subsequently.

Providing a workstation with a display adapter according to the present invention enables performance to be improved. Firstly, it enables faster updating of displayed images. This is a result, primarily, of the fact that the workstation processor no longer has to calculate display buffer addresses. In processors in which fast block move operations to a single address or series of addresses are possible, further improvements in the speed of updating can be achieved because these block moves can be exploited. Performance is also improved because the display buffer need not be contained within the processor address space due to the display adapter being responsible for addressing the individual the display buffer locations and consequently, processor address space is saved for other uses.

Workstations are known which provide hardware features which draw rectangles of a programmable size, colour and mix. Generally the colour of the resulting rectangle is determined by a pattern register which selects between a pair of constant colour values and logic functions which determine how the selected colour is mixed into the existing picture. In these prior art workstations, however, there is no provision for receiving a stream of data items representing an image, individual data items in the received stream defining respective pels of the image and the stream being ordered to define the image line-by-line and, within each line, pel-by-pel, and for storing the data items in a display buffer such that the image is correctly displayed on a display screen.

A workstation including a specific embodiment of a display adapter in accordance with the present invention is described in the following with reference to the accompanying drawings in which:

- Figure 1 is a block diagram of a workstation including a display adapter in accordance with the present invention;
- Figures 2A, 2B and 2C are schematic diagrams illustrating, respectively, the relationship between perceived picture element positions on a display screen (Figure 2A), the corresponding storage positions in a display buffer for the data corresponding to those positions (Figure 2B) and a stream of data items defining an image (Figure 2C);
- Figure 3 is a block diagram showing details of a display adapter in accordance with the present invention;
- Figure 4 is a block diagram of a modified version of the display adapter shown in Figure 3;
- Figure 5 is a block diagram of another modified version of the display adapter shown in Figure 3; and
- Figure 6 is a flow diagram illustrating aspects of the operation of the display adapter of Figure 3

Figure 1 shows an overview of a workstation comprising 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 connects the system bus to a display device 34. It also allows the connection of an external image data source (not shown) to the workstation via the line 37. 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 is preferably implemented as special purpose hard-wired logic to take advantage of the speed of such logic. The present invention does not, however, exclude the possibility of implementing the present adapter with a high speed microprocessor and storage elements and appropriate code.

The display adapter includes a display buffer

36 which is not addressed directly by the processor. The display device can, however access the display buffer in order to fetch the data corresponding to the individual picture elements on the screen (38, 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.

Figure 2A illustrates the display screen as perceived. The screen has "Y" lines of "X" pels each. As shown the lines of the screen are numbered from 0 to Y-1 starting from the top and moving down the screen. The pels in each row are likewise shown numbered, from left to right, 0 to X-1.

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

As shown, the display buffer comprises a single unit and is organised on 8 bit bytes. Each pel is represented by 4 bits of information defining the intensity and/or colour of the pel. However, it will be understood that other display buffer organisations are possible. For example, the buffer could be organised in the form of a plurality of parallel bit planes, each comprising one bit per pel, whereby the combination of the bit planes defines the complete image and whereby the complete pel information for a location on the screen is defined by the combination of the information in the corresponding position in each of the bit planes. A different number of bits might be also be chosen.

The display screen is refreshed in this embodiment using an interleaved scanning technique as is often the case in the art so that the scanning of even numbered rows alternates with the scanning of odd numbered rows. Thus, the image data for the individual pels of the display screen shown in Figure 2A are stored as shown in Figure 2B, with the data relating to the even numbered rows stored in a first sequence from a base address 361 of the refresh buffer and the data relating to the odd numbered rows stored in a second sequence starting at an offset address 362 at or after the end of the first sequence.

The data relating to each of the sixteen pels will not normally be generated in the workstation in the same order as that in which the data is stored in the display buffer. Commonly the data will be generated as a string or sequence 42 of data items, in which the individual pels of the image are defined line-by-line, and within each line, pel-by-pel. Figure 2C illustrates a string in which the image is defined row-by-row, and within each row, pel-by-pel. The first four data items relate to the pels on a first row (b), the second four relate to the four pels on the adjacent second row (b+1), the third four to the pels on the third row (b+2) and the fourth four to the four pels on the fourth row (b+3). This particular order is, however, merely illustrative. It may be that the data is generated column-by-column and within each column, pel-by-pel. Also, the ordering of the rows and/or columns may be inverted (eg for the rows, from bottom to top rather than, as shown in the Figures, from top to bottom).

It will be appreciated that a significant amount of address calculation is needed to determine the locations in the display buffer at which individual data items are to be stored. This is true, for example, even if it is intended to write an image which only is to occupy a part of the screen, say an image comprising a block 40 of $4^{\times}4$ pels to be displayed within the area where the following conditions apply:

$$a <= x <= a+3 \text{ and } b <= y <= b+3.$$

These calculations are performed by the display adapter 32 which is shown in more detail in Figure 3. 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. Also connected to the data bus for the receipt of initialisation data from the workstation RAM are first and second registers 48 and 50. A first input 52 of a two input multiplexer 54 is also connected to the data bus for the receipt of image data from the workstation RAM. The second input 56 of the multiplexer is connected to an external source of data (not shown), either directly, or via a data gearbox as will be explained later with reference to Figure 4. The output 57 of the multiplexer is connected to the data port 68 of the display buffer 38. The display adapter also comprises first and second counters 49 and 51 connected, respectively, to the first and second registers for receiving the count stored therein. An arithmetic logic unit 58 has first, 60, and second, 62, inputs connected, respectively, to the first and second counters. The output 64 of the arithmetic unit is connected to the address input of the display buffer 38. A driver is connected between the data port of the display buffer and the data bus and is used for the transfer of data in the buffer to the bus. The control unit is connected via control inputs C to the control storage, the first and the second registers, the first and second counters, the multiplexer, the arithmetic logic unit and the driver.

To illustrate the operation of the display adapter the steps which are performed in displaying the image 40 referred to with regard to Figure 2A is set out below. It is assumed that the processor, in performing some task, has generated image data

for a four-by-four block of pels and this is stored in RAM as a sequence 42 (Figure 2C) of sixteen data items in which the first four relate to the pels on a first row, the second four relate to the four pels on the adjacent second row, the third four to the pels on the third row and the fourth four to the four pels on the fourth row. It is also assumed that the image is to be displayed upright within an area having its upper left corner at a screen position b,a, where b is the vertical displacement in pels from the top of the screen and a is the horizontal displacement in pels from the left hand side of the screen.

In order to update the display the processor first initialises the display adapter by sending positioning data to the adapter over the data bus. The positioning data comprises initial screen position information which is sent to the first and second registers and size, orientation and data stream format information which is sent to the control storage.

The initial screen position information comprises the "x" and "y" values ("a" and "b") for the screen position of one corner of the area (eg. the top left hand corner) in which the image is to be displayed. The area size information defines the length of the horizontal and vertical sides of a rectangular area in which the image is to be displayed in terms of a numbers of pels. The orientation information effectively defines which corner of the rectangular area is identified by the "x" and "y" values stored in the first and second registers. The orientation information defines whether successive image items in the stream are to be displayed at increasing or decreasing y and x values (ie. x increasing / decreasing, y increasing / decreasing) and whether the image data is ordered row-by-row or column-by-column (ie. y major or x major). The data stream format information identifies the format of the stream of image data items to be received (ie. the number of bits per data item and/or the number of data items per byte). The adapter assumes that each data item comprises the complete definition of a pel (ie. for all the constituent bit planes).

The positioning data for the stream of image data in Figure 2C is as follows:

initial screen position information ..... x = a, y = b;
size information ..... horizontal 4 pels, vertical 4 pels;
orientation information ... x increasing, y increasing, x major;
data stream format information.....2 data items per byte.

This information will have been available to the processor in RAM as a result of the generation or selection of the image for display.

In addition to the initialisation data referred to above, the control logic also requires organisation data defining the organisation of the display buffer in order to be able to generate the display buffer addresses for a stream of incoming data items. The organisation data are loaded into the control storage by the processor. This can be either done at the same time as the initialisation data is supplied, or it can be done at some earlier time. The organisation data comprise the base address of the display buffer, (ie. the address from which the even numbered lines of pels are stored - 361, Figure 2B), the number of pels per byte, the total number of pels per line on the screen, the total number of scan lines on the screen and the address from which the odd numbered scan lines are stored (362, Figure 2B).

The control logic includes logic for determining the particular operations to be performed in the display adapter on the basis of the initialisation data, including the particular arithmetic operations to be performed by the arithmetic logic. In the example illustrated in Figures 2, the arithmetic operations to be performed by the arithmetic logic for a particular screen location y, x in order to generate the display buffer address for that location are:

address = base address + (((y*X) + x)/2) if y is an even number; and
address = offset address + ((((y-1)*X) + x)/2) if y is an odd number,

where y is the vertical screen displacement value for that location, x is the horizontal screen displacement value for that location and X is the total number of pels in a row. The offset address is the base address plus an offset.

Once the display adapter has been initialised, the image data can then be sent by the processor as a stream of data at high speed from the workstation RAM via the data bus to a single address (ie. the input 52 of the multiplexer 54). This is done by the processor using a single block move instruction.

The control unit 44 contains interlock mechanisms of a conventional type, as are provided in other adapters, for controlling the exchange of information with the host. By means of these mechanisms it is possible for the host to transfer the data as a block (ie. as an uninterrupted string or sequence) when both the processor and the adapter are ready. As the display adapter is configured as hard logic, it is able to process the information in real time as the stream of image data comes over the bus from the RAM.

The image data received at the multiplexer is

directed to the appropriate locations in the display buffer under the control of the control unit 44. In response to the area size and orientation data, the data stream format information and the organisation data stored in the control storage 46, the control logic causes the content of first and second counters to be automatically updated for each data item received in the sequence of data items and controls the arithmetic logic to perform appropriate arithmetic operations. The updating of the counters and the operation of the arithmetic logic is synchronised by the control logic with the receipt of successive data items in the data stream as they are received so that the data items may be stored directly in the appropriate display buffer locations.

Figure 6 is a flow diagram giving an overview of the operations performed by the display adapter in generating the display buffer addresses for storing the successive data items. The steps outlined in Figure 6 are illustrated in the following with reference to the example shown in Figures 2).

IN STEP 80, the initial screen displacement value, b, for the y ordinate (identified as being the major ordinate in the initialisation data and stored in the second register 50) is loaded into the second counter 51.

IN STEP 82, the initial screen displacement value, a, for the x (ie. the remaining) ordinate is loaded into the first counter 49.

IN STEP 84, the control logic causes the arithmetic logic to calculate the address of the location in the display buffer for the data item to be display at the screen displacement values currently in the first and second counters.

(In the case of the initial screen address b,a, where both b and a are odd, the display buffer address is the second half of the display buffer address "offset address + ((((b-1)*X) + a)/2)")

IN STEP 86, the control logic determines from the horizontal length of the rectangle (ie. the horizontal size) whether there are any more display buffer addresses to be calculated 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 increments the first counter as the initialisation data specified "x increasing" (It would have been decremented if the initialisation data had specified "x decreasing"). The control logic then returns (via loop 88) to step 84 and causes the arithmetic logic to calculate the address for the second data item using the formulae indicated above.

If there are no more pels in the current row of the image, the logic proceeds by path 90.

IN STEP 92 the control logic determines from the vertical length of the 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 increments the second counter as the initialisation data specified "y increasing" (It would have been decremented if the initialisation data had specified "y decreasing"). The control logic then returns (via loop 94) to step 82 and causes the initial screen displacement value, a, for the x ordinate to be loaded into the first counter.

If there are no more rows to be processed the control logic then exits via the path 96 and the transfer of the image data stream is complete.

As mentioned above, the second input 56 of the multiplexer is connected to an external source of data. The procedure for receiving image data from an external source is essentially the same as that for receiving image data from the data bus. The initialisation data would still be provided over the data bus by the processor, but in this case the source would be identified as that connected to the input 56 of the multiplexer. The alternative source could be a video source such as a video camera with a digital interface. It could be an image data stream output from, or for, another display adapter with a different scanning rate. The adapter therefore allows emulation of existing display adapters. This is a useful facility as it provides compatibility with existing workstation systems. (By appropriate use of the orientation information it is an easy matter to accommodate image data from a display adapter based on a different display buffer format from that illustrated in Figure 2b. For example data based on a vertical scan can be accommodated by specifying the y ordinate as the major ordinate.)

The connection to the alternative video source could be via a direct path as in Figure 3 if the display adapter is able to cope with the video rates of all of the external sources which might be connected to it. If, however, it is intended to connect video sources having very high video data rates, it is preferable to employ a data gearbox 70 as shown in Figure 4. Preferably also, the gearbox will have a reduction ratio which is variable under the control of the control logic. The purpose of the data gearbox is to select only the Nth pel of the image

data received from the external video source. The control logic controls the reduction gearbox and the other elements of the adapter to alter only 1 in N pels on each scan of the incoming image data in such a manner that the complete incoming image is captured every Nth scan. This technique has been used successfully to capture the output of an external display adapter running at many times the pel drawing rate of the present adapter.

As mentioned above, the adapter shown in Figure 3 had a driver connected between the data port of the display buffer and the workstation bus. In the case of the Figure 3 adapter this driver can be used to save a screen, or part of a screen of data in the workstation memory, after, for example, compiling a screen of data from a number of constituent images. In the Figure 4 version of the adapter, the provision of the driver also means that it is possible to capture image data from an external source by reading a stream of image data from the external source into the display buffer and then transferring that data to the workstation RAM.

Figure 5 illustrates an alternative version of the display adapter shown in Figure 3 which enables a stream of image data to be received at very high burst rates which exceed the display buffer addressing capability (or pel drawing rate) of the display adapter. The alternative display adapter is additionally provided with an image buffer 72 which is connected between the multiplexer and the display buffer in the image data path. In this adapter the control logic is arranged to read the image data received via the multiplexer. The provision of the image buffer also allows the image data transfer rate to be increased in certain circumstances as it means that space and/or data is/are always available and as a consequence processor wait states can be saved. The ability to buffer the data in the image buffer in this way is a result of the control logic rather than the processor addressing the individual locations. The version of adapter shown in Figure 5 could additionally be provided with a data gearbox as shown in the Figure 4 version.

A detailed description of the logic which makes up the various logical units shown in the Figures has not been included in this description as it is considered to be a matter of routine for the person skilled in the art to implement that logic in view of the description given above of the functions to be performed.

Although a specific 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 adapter could be arranged to only accept a single input data format. In this case control logic could be so configured that constant data stream format information (eg. the number of data items per byte) would be incorporated in the logic and consequently this information would not need to be provided by the processor with the initialisation information.

On the other hand it might be desired to accept the image data for each of a number of bit planes one after another. In this case, there would be a separate series of data items for each of the bit planes. Each data item would comprise a single bit and the complete image would be formed from the combination of the information for each of the bit planes. The adapter could be configured to accept the input information in this form by causing the contents of appropriate display buffer locations to be modified by, rather than replaced by the data items in the incoming data stream.

Instead of the driver 69 shown in Figures 3 to 5, the adapter could alternatively be provided with an output multiplexer so that image data from the screen could be supplied to the data bus of the workstation, or alternatively to an external device.

In the described embodiment the area displayed on the screen is rectangular. If desired, however, the adapter could also be provided with a facility to display images which were not rectangular by incorporating masking logic in the adapter. In simple terms this could be achieved transferring mask boundary information from the workstation RAM to the control storage and subsequently transferring 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 discarded so that only that part of the image within the boundary would be written to the buffer and be displayed.

Claims

1. A display adapter for connecting the system bus (12) of a workstation to a display means (34) having a scanned display screen, the display adapter comprising:

input means (46, 48, 50, 54) for receiving positioning data specifying an area on the screen within which an image is to be displayed and for receiving a stream (42) of data items representing the image, individual data items in the received stream defining respective pels of the image and the stream (42) being ordered to define the image line-by-line and within each line, pel-by-pel;

a display buffer (38) for storing, in display scanning sequence, data defining the pels to be displayed on the display screen;

and addressing logic (44, 49, 51, 58) for computing from the positioning data, individual storage locations in the display buffer to be addressed for individual data items in the received stream (42) in order that the data for the individual pels of the image are stored in the display buffer in said display scanning sequence and in order that the image is displayed correctly on the display screen.

2. A display adapter as claimed in claim 1 wherein the input means comprises:

register means (48, 50) connectable to the processor bus for receiving initial screen position information;

control storage (46) connectable to the processor bus for receiving size and orientation information which, in combination with the initial screen position information, form the positioning data defining the area on the within which the image is to be displayed; and gate means (54) connectable to the system bus for receiving the stream (42) of data items representing an image to be displayed within that area.

3. A display adapter as claimed in claim 2 wherein the addressing logic comprises:

counter means (49, 51) connected to the register means for (48, 50) receiving the initial screen position information contained therein;

arithmetic logic (58) connected to the counter means for receiving the instantaneous values of information contained therein and for generating display buffer address signals therefrom; and

control logic (51) which is connected to the control storage (46) and, in response to the size and orientation data stored therein:

- causes the counter means (49, 51) to be loaded from the register means (48,50) and/or the content of the counter means (49,51) to be adjusted, in synchronism with the receipt of successive data items in the received stream, to identify the screen location for each successive data item;

- causes the arithmetic logic (58) to perform appropriate arithmetic operations on instantaneous values contained in the counter means so as to generate display buffer addresses for successive data items in the received stream; and

- causes the data for the individual pels of the image to be stored in appropriate locations in the display buffer (38) using the display buffer addresses so generated such that the image is displayed on the screen within said area.

4. A display adapter as claimed in claim 2 or in claim 3 wherein the gate means comprises a multiple input multiplexer (54), a first input (52) of which is connectable to the system bus (12,14) of a workstation for receiving a stream of image data from the workstation RAM, a second input (56) of which is connectable to an alternative image source (via 37) external to the workstation and the output of which is connected to the data port of the display buffer.

5. A display adapter as claimed in claim 4 additionally comprising a data gearbox (70), the output of which is connected to the second input (56) of the multiplexer (54) and the input of which is connectable to the alternative image source, the control input (c) of the gearbox (70) being connected to the control logic (44) whereby the data rate from the alternative image source can be reduced, if necessary, to the data rate which may be handled by the display adapter.

6. A display adapter as claimed in claim 2 or in any claim dependent thereon wherein the display adapter is additionally provided with an image buffer (72) for the temporary storage of the received stream of image data, said image buffer (72) being connected between the gate means (57) and the display buffer (38) and additionally connected to the control means (44) via a control input (c).

7. A display adapter as claimed in any one of the preceding claims additionally comprising a driver (69), the input of which is connected to the data port (68) of the display buffer and the output of which is connectable to the system bus (12,14) so that image data may be transferred, in use, from the display buffer (38) to the system bus.

8. Workstation comprising a central processing unit (10), random access memory (20), a system bus (12), display means (34) having a scanned display screen and a display adapter (32) as claimed in any one of the preceding claims connected between the system bus and the display means.

9. A Workstation as claimed in claim 8 wherein the workstation is so configured that the display adapter (32) is connected to the processor via one of the processor I/O ports and the stream of image data may be provided from the workstation RAM in response to a processor block move operation.

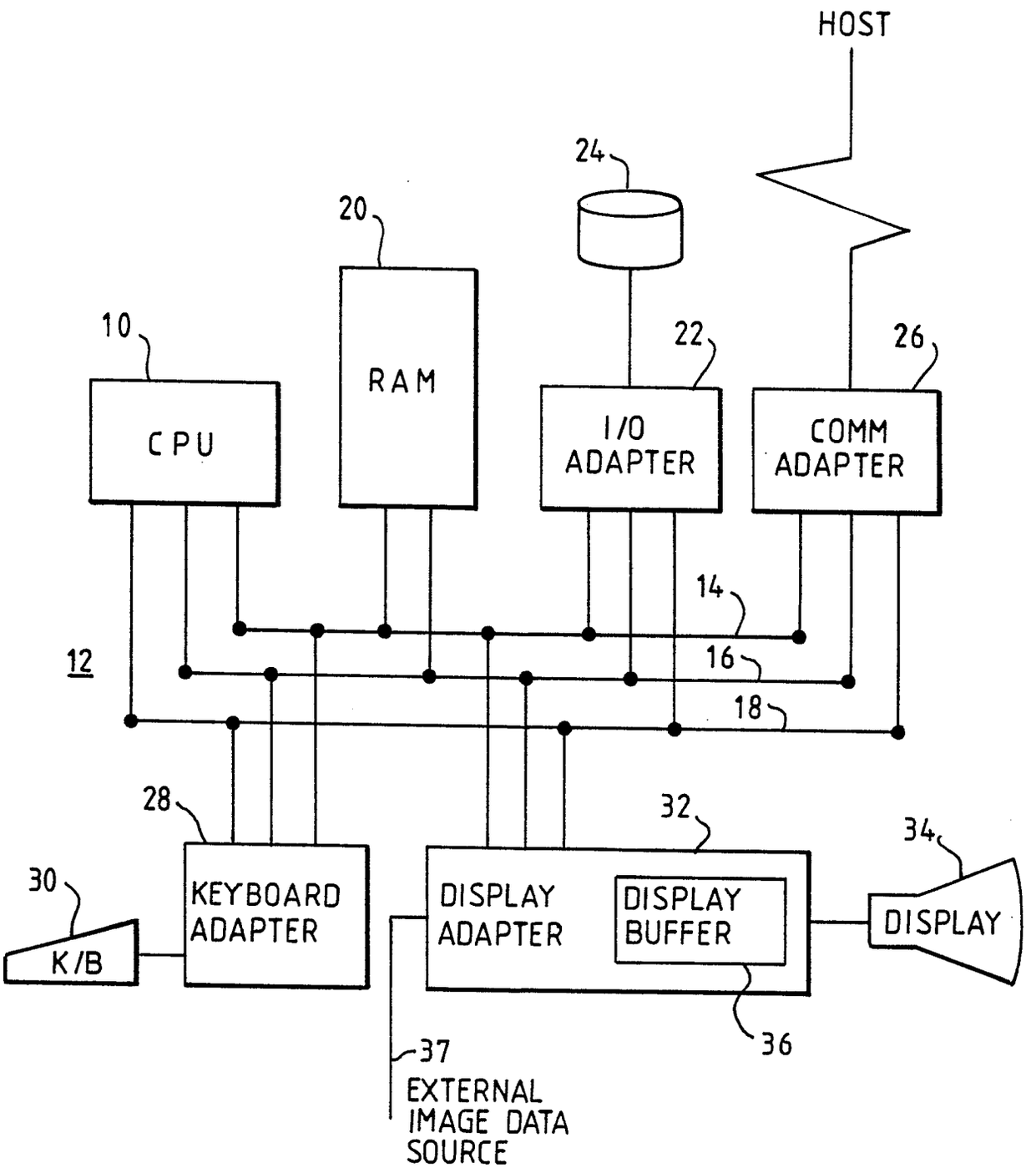
HOST
24
20
10
22
26
RAM
CPU
I/O ADAPTER
COMM ADAPTER
14
12
16
18
28
32
34
30
KEYBOARD ADAPTER
DISPLAY ADAPTER
DISPLAY BUFFER
DISPLAY
K/B
36
37
EXTERNAL IMAGE DATA SOURCE

FIG. 1

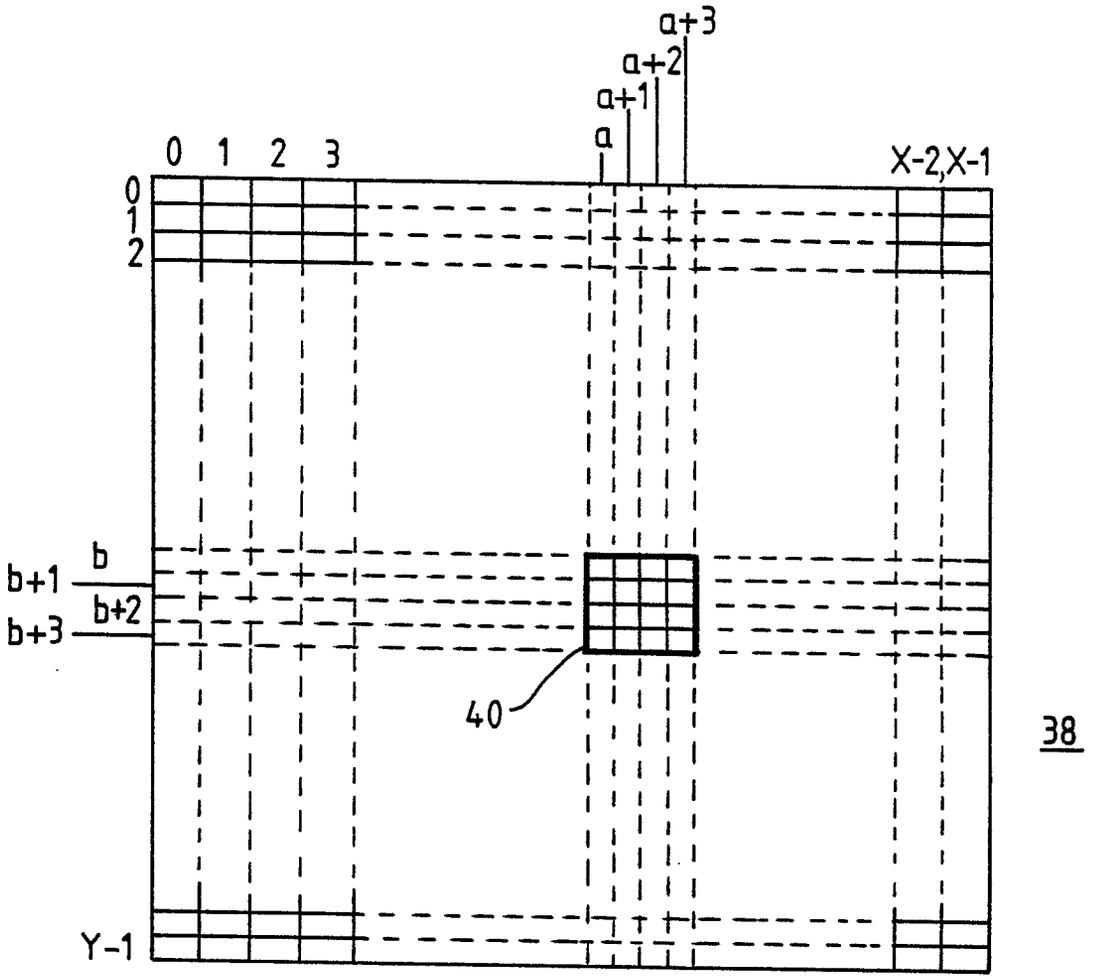
a+3
a+2
a+1
a
0 1 2 3
X-2,X-1
0
1
2
b
b+1
b+2
b+3
40
38
Y-1

FIG. 2a

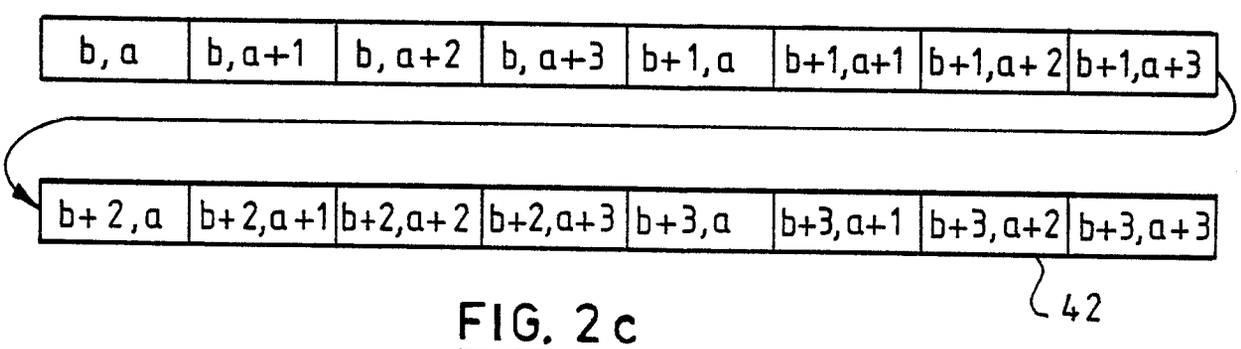
b, a
b, a+1
b, a+2
b, a+3
b+1, a
b+1, a+1
b+1, a+2
b+1, a+3
b+2, a
b+2, a+1
b+2, a+2
b+2, a+3
b+3, a
b+3, a+1
b+3, a+2
b+3, a+3
42

FIG. 2c

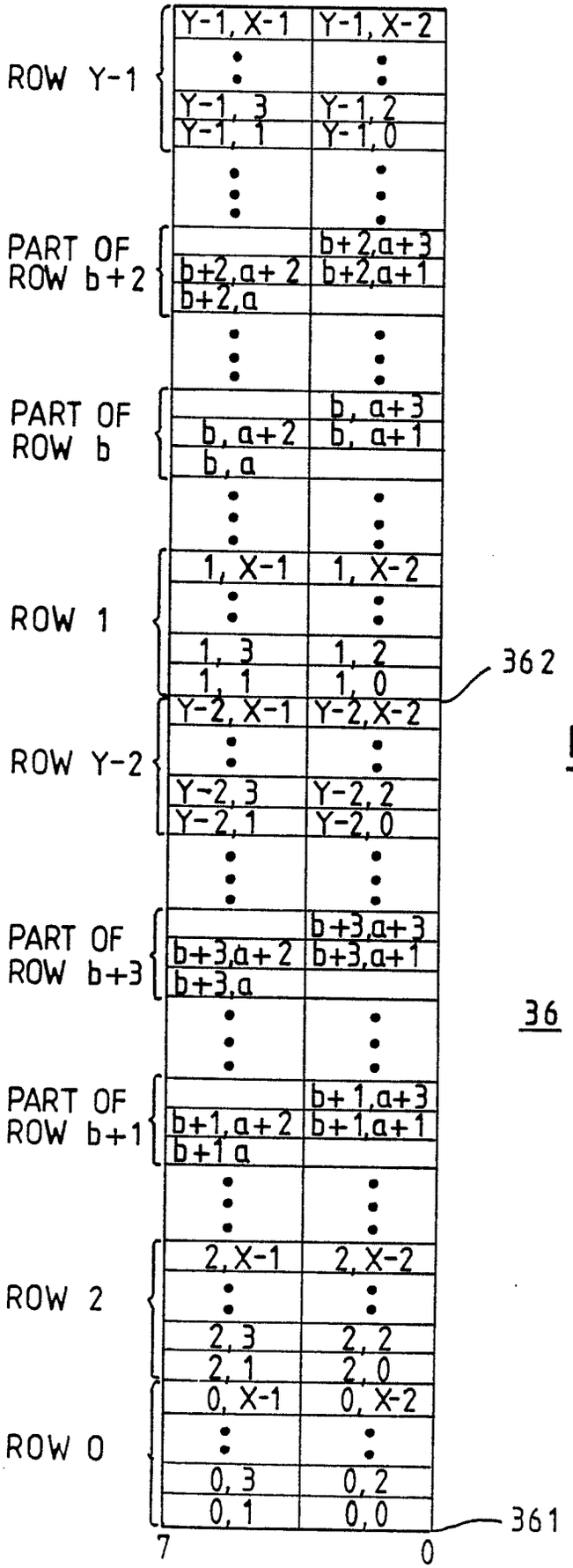
ROW Y-1
Y-1, X-1
Y-1, X-2
Y-1, 3
Y-1, 2
Y-1, 1
Y-1, 0
PART OF ROW b+2
b+2, a+3
b+2, a+2
b+2, a+1
b+2, a
PART OF ROW b
b, a+3
b, a+2
b, a+1
b, a
ROW 1
1, X-1
1, X-2
1, 3
1, 2
1, 1
1, 0
362
ROW Y-2
Y-2, X-1
Y-2, X-2
Y-2, 3
Y-2, 2
Y-2, 1
Y-2, 0
PART OF ROW b+3
b+3, a+3
b+3, a+2
b+3, a+1
b+3, a
36
PART OF ROW b+1
b+1, a+3
b+1, a+2
b+1, a+1
b+1 a
ROW 2
2, X-1
2, X-2
2, 3
2, 2
2, 1
2, 0
ROW 0
0, X-1
0, X-2
0, 3
0, 2
0, 1
0, 0
361
7
0

FIG. 2b

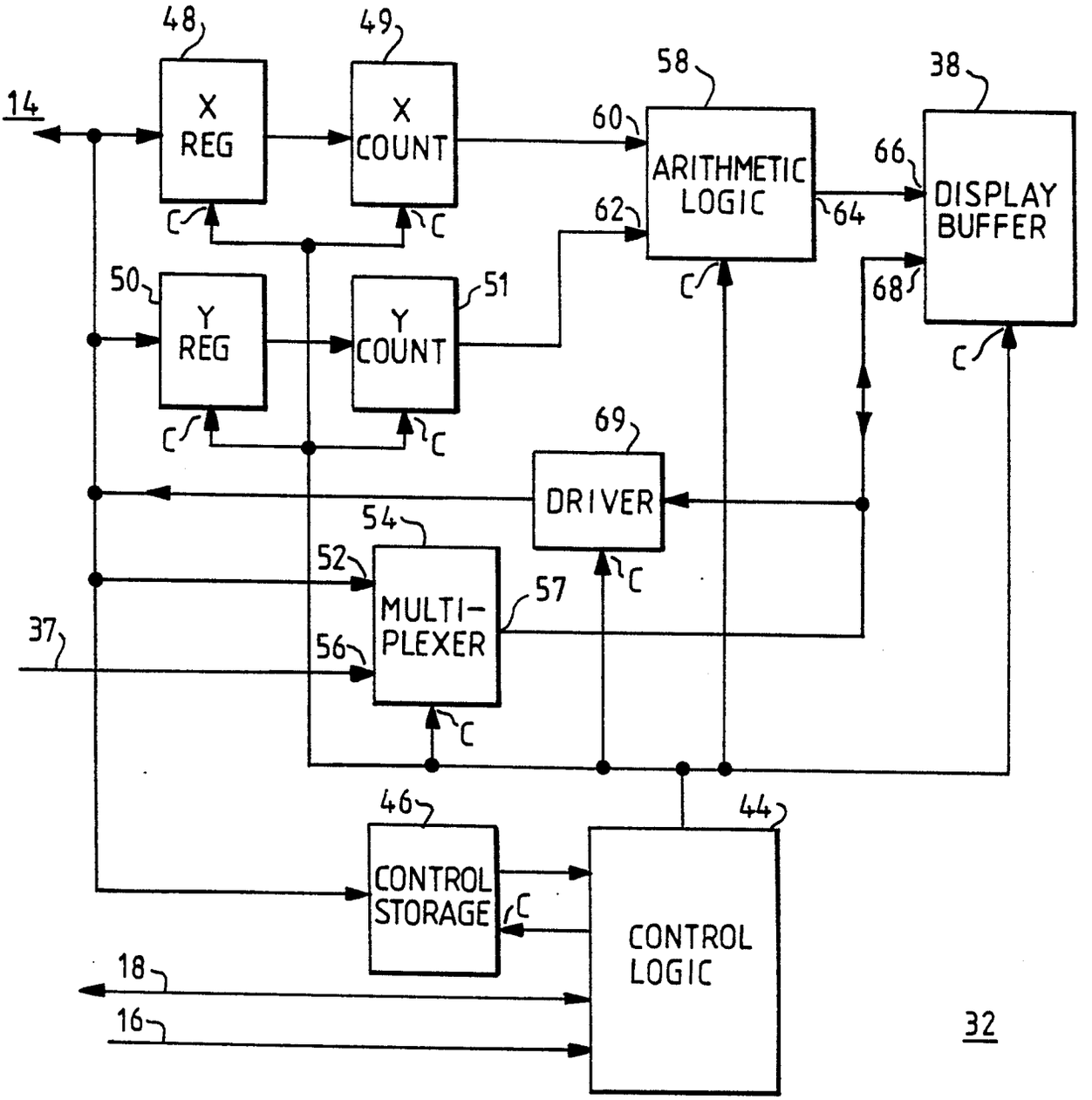
48
49
58
38
14
X REG
X COUNT
60
ARITHMETIC LOGIC
66
DISPLAY BUFFER
62
64
50
Y REG
Y COUNT
51
68
C
69
DRIVER
54
52
MULTI-PLEXER
57
37
56
46
44
CONTROL STORAGE
CONTROL LOGIC
18
16
32

FIG. 3

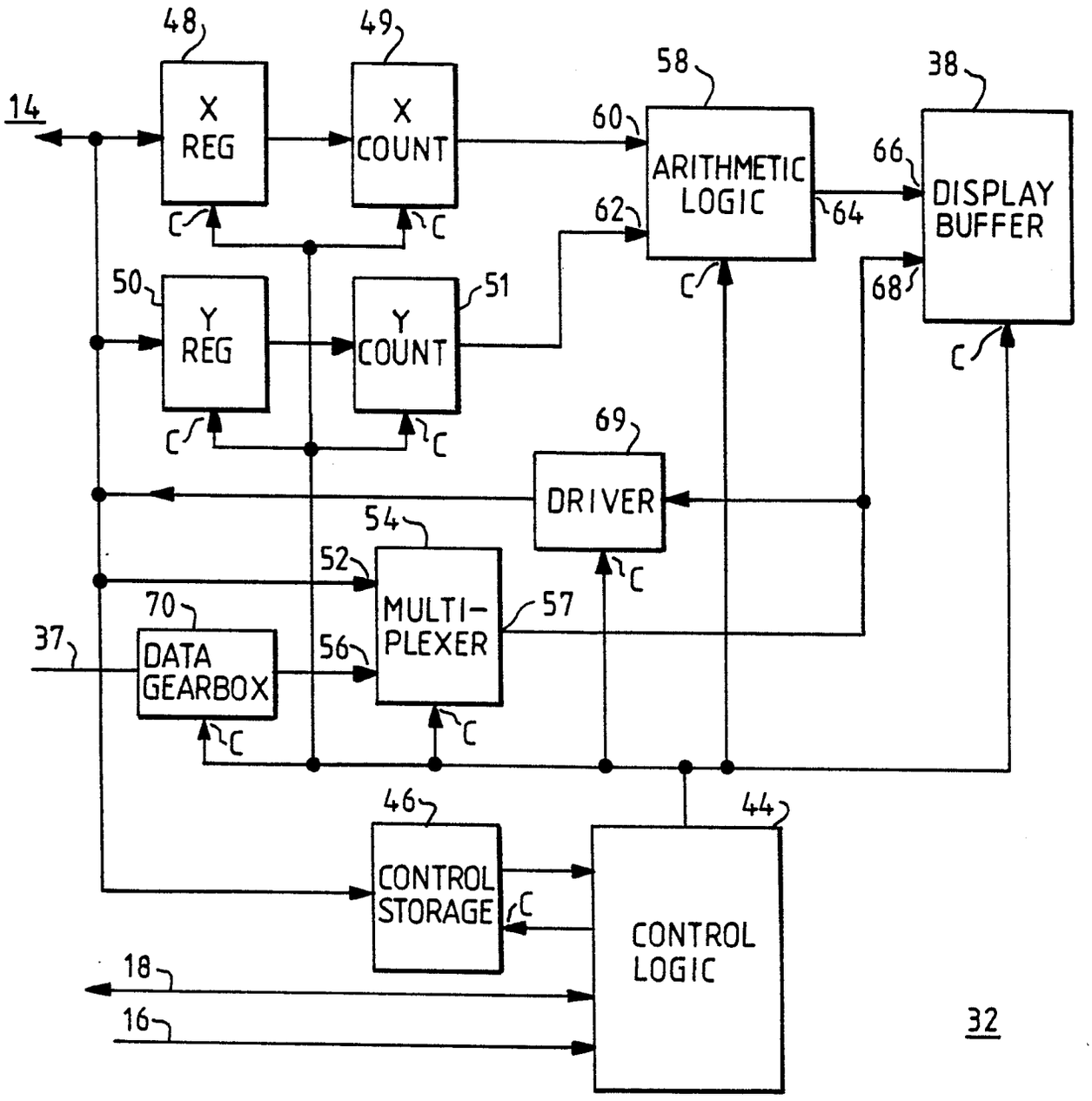
14
48
X REG
49
X COUNT
58
60
ARITHMETIC LOGIC
38
66
DISPLAY BUFFER
62
64
68
50
Y REG
51
Y COUNT
69
DRIVER
54
52
MULTI-PLEXER
57
70
37
DATA GEARBOX
56
46
CONTROL STORAGE
44
CONTROL LOGIC
18
16
32
C

FIG. 4

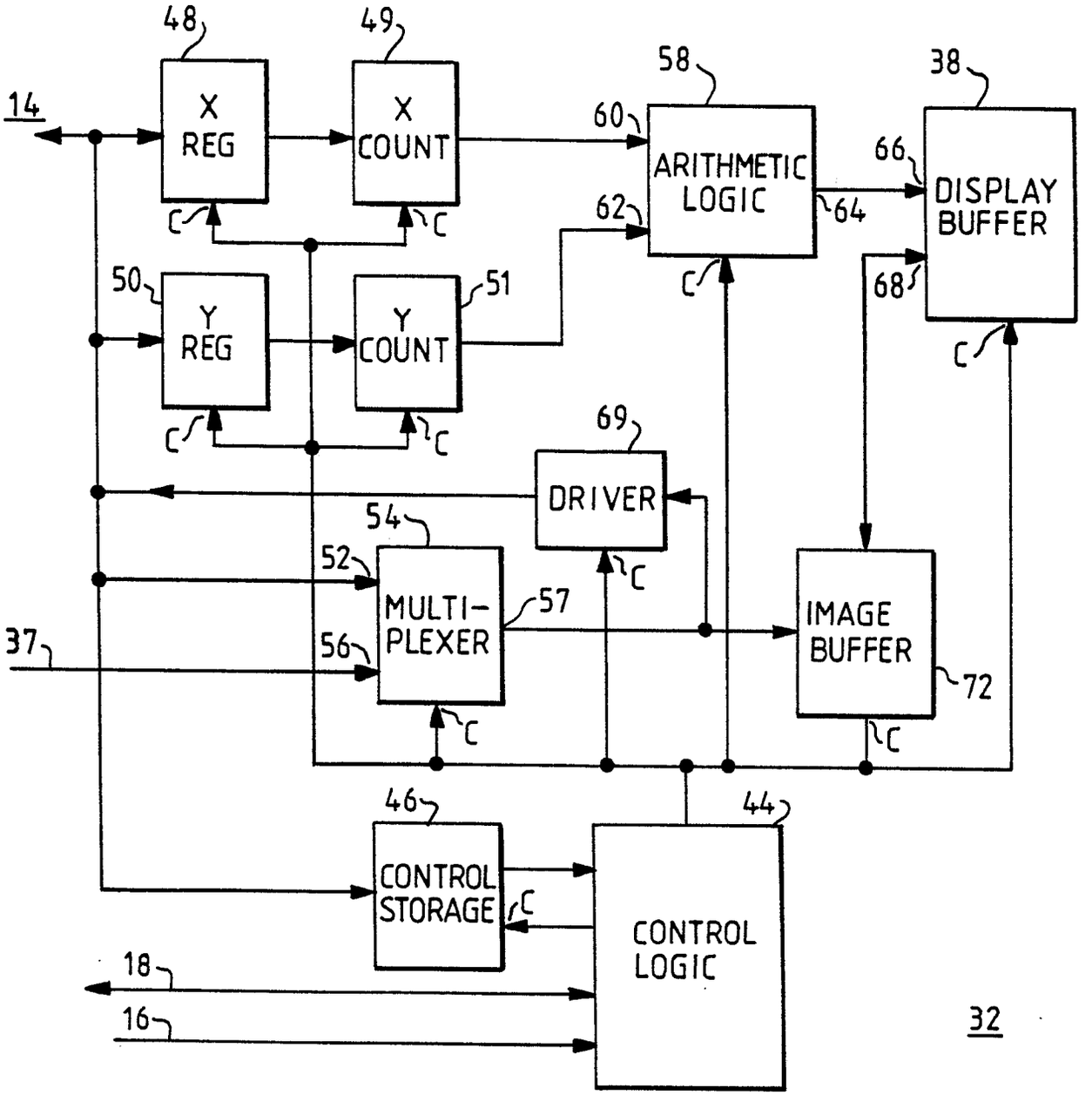
48
49
58
38
14
X REG
X COUNT
60
ARITHMETIC LOGIC
66
DISPLAY BUFFER
62
64
C
50
Y REG
Y COUNT
51
68
69
DRIVER
54
52
MULTI-PLEXER
57
IMAGE BUFFER
37
56
72
46
44
CONTROL STORAGE
CONTROL LOGIC
18
16
32

FIG. 5

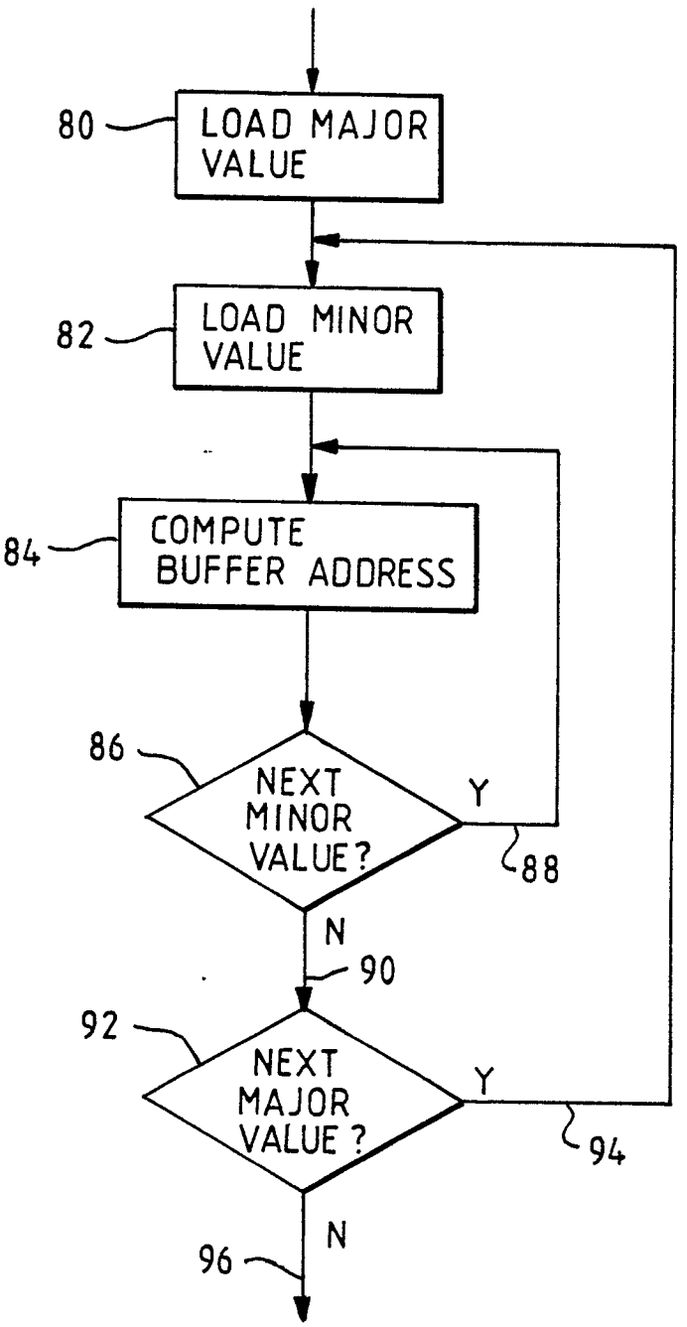
80
LOAD MAJOR VALUE
82
LOAD MINOR VALUE
84
COMPUTE BUFFER ADDRESS
86
NEXT MINOR VALUE?
Y
88
N
90
92
NEXT MAJOR VALUE ?
Y
94
N
96

FIG. 6