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

(11) Publication number: **0 413 483 A2**

(12) **EUROPEAN PATENT APPLICATION**

(21) Application number: **90308591.8**

(51) Int. Cl.5: **G09G 1/28**, G09G 5/06

(22) Date of filing: **03.08.90**

(30) Priority: **14.08.89 IL 91303**

(43) Date of publication of application:
20.02.91 Bulletin 91/08

(84) Designated Contracting States:
DE FR GB

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

(72) Inventor: **Chevion, Dan Shmuel**
43 Hantkeh Street
Haifa 34806(IL)
Inventor: **Walach, Eugeniusz**
1 Ciegel
Kiriat Motzkin, 26418(IL)
Inventor: **Gilat, Ittai**
38 Henrieta Szole
Haifa, 34722(IL)

(74) Representative: **Harris, Ian Richard**
IBM United Kingdom Limited Intellectual Property Department Hursley Park
Winchester, Hants. SO21 2JN(GB)

(54) **A display system.**

(57) A display system is described that allows the rapid display of colour images on screen, and removes the need for extensive image pre-processing prior to display on screen. A universal palette is utilised, divided into several sub-palettes, each one containing data pertaining to shades of different, individual colours. For example, in a palette with an 8 bit address, 64 shades each of red, green, blue and grey could be stored. Source colour data are supplied to logic which selects a particular colour component's data, and this data is then utilised to address the sub-palette containing data pertaining to shades of that particular colour. For example, red source data are utilised to address the section of the palette (sub-palette), containing data yielding red colour shades. Pixels are grouped on screen to form a macro-pixel, each pixel of the macro-pixel being devoted to one particular colour shade. The eye then acts to merge these colour pixels on screen to form the composite colour of the macro-pixel. In an alternative embodiment, all of the source colour data are used to address the palette. Overall contouring of the image on screen is removed without the need for extensive image pre-processing, thus allowing the rapid display of guality images on screen.

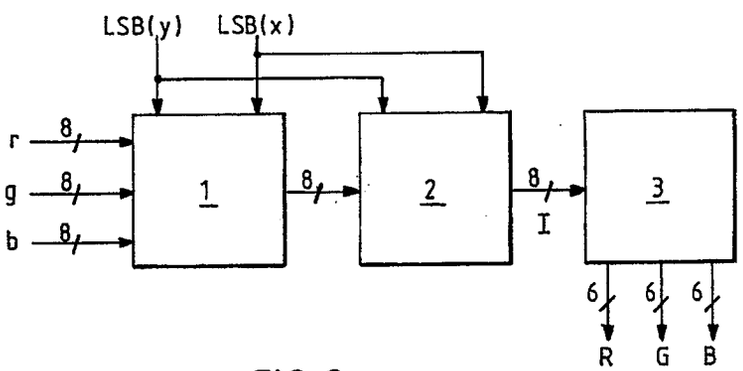

FIG. 3

EP 0 413 483 A2

A DISPLAY SYSTEM

The present invention relates to a display system for displaying images on a raster-scanned cathode ray tube (CRT).

In raster-scanned display systems, a bit map or refresh buffer is employed to store data representing the image to be displayed. This data is output to digital to analog convertors (DACs) to control the intensity of an electron gun(s), and hence the nature of the image as it appears on screen.

In its simplest form, bit map data merely dictates whether the electron gun is active. However, in the majority of modern systems, it is necessary for greater image detail to be stored and manipulated within the bit map, particularly when a colour representation is desired on screen. For example, at any one bit map location, data may be stored referring both to the intensity of red, green and blue colour components, and the overall luminance or intensity of the pixel.

Source colour data stored within a bit map, is usually divided into its red, green and blue (RGB) components, a set number of bits being assigned to each colour. In a typical modern system, such as that utilised in the IBM PS/2 (Trade Mark) computer, 24 bits of data are stored at each bit map location, 8 bits being assigned to each of the three colour components which when combined form the final pixel colour on screen.

In direct colour mode, data from the bit map can be supplied directly to DACs that control the three electron guns within the system (one for each colour), the final colour of each individual pixel on screen being a mixture of the three colour components generated by each gun. To increase the flexibility of data manipulation, many systems include a palette or look-up table. The palette stores data which corresponds to a series of colours, the bit map data being used to address particular locations within the palette. As with direct colour mode, the final colour of the pixel on screen is a mixture of RGB components, but the guns are controlled, via the DACs, by the data stored in the palette.

In modern systems, such as those in the IBM PS/2 (Trade Mark) family, the palette has an 8 bit address. Thus the system has 256 possible colours to display on screen at any one time. The PS/2 (Trade Mark) computers are typical in that 18 bits of data are stored in each palette location ready to be supplied to the DACs, 6 bits for each of the primary colour components.

From the above description, it is clear that a typical display system must overcome two problems; firstly which colours to place within the palette (for example, with 18 bits of data at each location, there are 2**18 possible colours that could potentially be placed within the palette) and, secondly, how to transform source data such that it can address the palette.

The first problem stems from the fact that normally only a single palette is available for the entire screen. IBM Technical Disclosure Bulletin Vol. 26, No. 7A, P3409-3418, does describe a system where more than one look-up table is utilised, such that multiple applications can share the same screen. In this system, the screen is divided into discrete areas or viewports, each individual look-up table being asigned to one or more of these viewports, and containing data pertaining to a particular application. The system does not include a "universal" palette, that is one palette containing data for all applications for controlling the display in any section of the screen.

The transformation of, for example, 24 bit source data to an 8 bit address is costly in processing time and usually results in the image having to be prepared for display off line. One existing solution to this problem is to have fixed bit allocation for each of the colours; for example, 4 bits for green, and 2 each for red and blue. These 8 bits are then utilised to address the palette. However, fixed bit allocation leads to contouring, that is where transitions from one colour shade to another are conspicuous, and an unacceptable image quality. Contouring of the image will almost inevitably result due to the limited shades of any one colour available within the palette. Error diffusion, or other dithering techniques, can be used to improve the final image quality, but this is costly as regards processing time. One solution in the prior art has been to store the processed image for later use, but this usually involves the loss of the original source data.

The essence of the present invention is to provide a display system that allows the rapid presentation on screen of quality images, without the need for extensive image preprocessing, or reloading of the palette.

According to the invention there is now proposed a display system comprising a bit map for storing source data for an image pixel, a palette containing data for controlling pixel colour on screen, logic for addressing the palette, wherein the palette is a universal palette divided into sub-palettes, at least one sub-palette containing data for controlling pixel colour for display anywhere on screen, said data corresponding to shades of dominantly one colour, wherein the data within said sub-palette are addressed via an index generated from an associated portion of the source data.

The invention avoids the need to reload the palette for different applications. The screen can be utilised for several different applications at the same time, all applications utilising the same universal palette. An additional benefit is that many more shades of a particular colour are available for display than in a conventional display system, hence removing the need for error diffusion or other dithering techniques to be applied to the data prior to display, in order to remove contouring effects. This is of particular value when a grey-scale image is to be displayed.

In one arrangement of the invention where a multi-coloured image is to be displayed (that is not grey-scale) the image pixels are grouped to form a macro-pixel, wherein the portion of source data utilised to generate the index to address a sub-palette and the sub-palette addressed are dependent on the position of the image pixel within the macro-pixel (M). The advantage of this is that individual pixels can be shades of different colours, the eye acting to merge the colours together to form the composite colour of the macro-pixel. In this way many more composite colours are available for display than in a conventional display.

In a preferred arrangement the source data includes red (r), green (g) and blue (b) source data and the palette is divided into a first, a second, a third and a fourth sub-palette, the first sub-palette for storing red colour data, the second for storing green colour data, the third for storing blue colour data, and the fourth for storing grey colour data. Logic is included for selecting colour component data to be supplied to index generation logic such that the red colour component source data is utilised to address the sub-palette storing dominantly red colour data, blue colour component source data to address the sub-palette storing dominantly blue colour data and the green colour component source data to address the sub-palette storing dominantly green colour data.

In an alternative arrangement of the invention all source data can be utilised to generate the index to the sub-palette.

The display system can be implemented in the form of a general purpose computer, for example a personal computer, including a keyboard and/or other user-input devices, memory, a processor, a display adapter and display device. It could also be implemented as a display adapter card for use, for example, with such a general purpose computer.

A method for processing image pixel data is presented wherein a palette is divided into sub-palettes, at least one sub-palette containing data for controlling pixel colour anywhere on screen, said data corresponding to shades of dominantly one colour, wherein the sub-palette is accessed by an associated portion of image pixel source data.

Specific embodiments of the invention will now be described with the aid of the accompanying diagrams, in which:

Figure 1. is a unit cell model of colour

Figure 2. illustrates a pixel grouping to form a macro-pixel

Figure 3. is a block diagram of part of a display system

Figure 4. indicates the variation of LSB(x) and LSB(y) with screen position.

Figure 5. illustrates the workings of the select component logic.

Figure 6. illustrates processing for a grey-scale image.

Figure 7. illustrates a pixel grouping to form a macro-pixel.

As previously stated, RGB colour components are mixed in varying proportions in order to yield a composite colour for each pixel as it appears on screen. In a conventional display system utilising a colour palette, the source data is manipulated such that the nearest colour approximation available is chosen from the palette. Figure 1 shows a conceptual unit cell model of colour, where each of the axes represents one of the main RGB colour components. The colours represented by an equal mixture of these three components lie along the diagonal axis (dotted line), and are various shades of grey. In conventional systems, there are limited available mixtures of shades of red, green and blue); a major disadvantage is the lack of shades of grey colour obtainable from these available mixtures of RGB.

To overcome these problems, the present invention utilises a universal palette, that is one palette for all applications, divided into sub-palettes. In one embodiment the palette is divided into four sub-palettes, each with 64 entries, making 256 entries in total, the output of each sub-palette being devoted to shades of one particular colour; for example, red for sub-palette 1, green for sub-palette 2, blue for sub-palette 3 and grey for sub-palette 4. In this way, 64 shades of each colour are stored within the palette and are available for display on screen.

In order to utilise these available colour shades to display a multi-colour image (that is not a black and white/grey-scale image), the pixels are grouped into "macro-pixels" to form each picture element. Figure 2 shows an embodiment where the macro-pixel (M) consists of four pixels, a,b,c and d. Pixel 'a' is converted according to palette 2 (it is green), 'b' according to palette 1 (red), 'c' by palette 3 (blue) and 'd' by palette 2 (green). In essence, instead of the colour output of three electron guns being merged at one pixel on screen to form a composite colour, the guns are aimed at individual pixels grouped to form one picture ele-

ment. The eye then acts to merge the individual colours, thus forming the desired composite colour for that picture element (or macro-pixel). Due to the human eye's greater sensitivity to the colour green, it is desirable for a greater number of individual pixels within a macro-pixel to be devoted to green shades, rather than to either of the alternative colours.

Figure 3 illustrates part of the logic of a display adapter for a personal computer in which a multi-colour picture is to be produced on screen and 24 bits of source colour data (8 bits each for RGB) are available at each bit map location in the display buffer of the display adapter. The colour component utilised to address the palette must be "selected", and then used to generate an index to the "physical" palette (Fig 3). Colour source data (r,g,b) is supplied from the display buffer to select component logic (1). Least significant bit data LSB-(x) and LSB(y) is supplied from the Cathode Ray Tube Controller (CRTC) of the display adapter to the select component logic (1) during scanning of the display screen (not shown). The least significant bit data identifies which pixel (a,b,c or d) within a macropixel (M) is currently being scanned. This LSB data is used to select the appropriate colour component data (r, g or b) to be supplied to the index generator logic (2), which utilises these data to access a particular sub-palette, or portion of the palette (3). The output (R,G,B) from the palette is used to control a colour display device (not shown) connected thereto. The display buffer, the CRTC and other elements of the display adapter which are not shown, can be conventional. Accordingly these elements will not be further described herein.

In Fig. 4, x and y coordinates correspond to the coordinates of the pixel on screen (4). As a raster scan moves from left to right across the screen (x=0 to Xmax) and from top to bottom (y=0 to Ymax), the least significant bits (LSB) of data defining the pixel x,y coordinate change from 0 to 1 alternating with each column or row respectively. These LSB values are supplied to logic (1 in Fig. 3) to select the colour data utilised to generate the palette index (via logic block 2).

Fig. 5 illustrates the selection process. In the top row, LSB(y) is always 0 but LSB(x) alternates being either 0 or 1 (Fig. 4). If 0, then the green colour data (g) are utilised to generate the address to the green section of the palette, otherwise the red data (r) are used. Continuing the scan progression, in the next row down, LSB(y) is 1 and LSB(x) again alternates, being either 0 or 1. If LSB(x) = 0 then the blue colour data (b) are utilised to generate the index (I) to the palette, otherwise the green data (g) are employed. These patterns are repeated throughout the raster scan until y=Ymax and x=Xmax (that is the scan reaches the bottom right hand corner of the screen), when fly back occurs and the process repeats. Ultimately, this process results in the pixel pattern highlighted in Fig. 2, where pixels in positions 'a' and 'd' are green, in 'b' red and in 'c' blue.

Once a particular colour component data has been selected, the index generator (2) must utilise this data to generate an index to a palette address in the desired portion, or sub-palette, of the main palette (3), in order that the pixel be the desired colour. If P represents the total size of the palette (in this case 256), Pr, Pg, Pb and Pl the size of the red, green, blue and grey (I represents luminance) sub-palettes respectively (in this example they would all be 64), if the green colour data are selected the index (I) to the palette is:

$I = G(C(x,y)) = Pr + ((Pg/P).g$

where g represents the green colour data (analogous to the desired intensity of the green electron gun) at location x,y.

Conversely if the red colour data are utilised to generate the index:

$I = R(C(x,y)) = (Pr/P).r$

where r represents the red colour data at location x,y.

If the blue colour data are chosen:

$I = Pr + Pg + ((Pb/P).b)$

where b represents the blue colour data at location x,y.

In this way the green colour data accesses the section of the palette containing data yielding a green shade when output to the DAC, the red data a red shade and blue data the section of palette, or sub-palette, yielding a blue shade.

In other embodiments, pixels could be grouped in many different ways to form macro-pixels (an embodiment is illustrated in Fig. 6), the associated processing of the pixel data being modified to generate the correct combinations of individual pixel colours to form the macro-pixel (|1|,|2|,|3| are converted according to the contents of palettes 1,2 and 3 respectively).

Utilising a 24 bit source data system as previously described, Fig. 7 illustrates the processing if the image were to be displayed in black and white (that is a grey-scale rather than a multi-coloured image were desired). In this "black and white" case, the pixel "colour" is not dependent on its position on screen; the pixels are not grouped to form a macro-pixel. There is no need to select a colour component for generating the index to the palette (I). For any pixel position (x,y), the Index (I) to the palette is dependent on the luminance resulting from the combined intensity of the red, green and blue guns (1) at that location:

$I = L(C(x,y))$

where $L(C(x,y)) = Pr + Pg + Pb + ((Pl/P).l)$

and 1 = 0.3r + 0.6g + 0.1b

In this way, only the grey colour/luminance section of the palette is addressed and grey shades output to screen.

In an alternative embodiment, where 8 bits of source data are stored at each bit map location, if a black and white image is desired, then these data (I) are used to address the grey sub-palette directly, to generate a grey-scale pixel where:

I = Pr + Pg + Pb + (Pl/P).I

For a multi-colour display 24 bits of "source data" could be generated from the 8 original bits utilising an alternative "bit map" palette. The processing would then progress as for the multi-colour, 24 bit source data case described above and illustrated in Figs. 3,4 and 5.

Overall, the greatly increased number of shades of each individual colour available within the system described removes contouring effects without extensive pre-processing of the image data. Such a facility is of particular use when displaying a synthetic image where a smooth gradation of colour tones is necessary. Although a system has been described in which 64 shades of each colour are stored in the palette, a system with fewer colour shades could be employed, the remaining memory being used to store other image data.

When using the system described to display grey level images, resolution and brightness are maintained. However, for multi-colour images some resolution is lost, and the brightness of the image is decreased (only one gun is aimed at each individual pixel, rather than three as is the norm). For some applications, this could potentially lead to an unacceptable image quality. To overcome these difficulties, a display system could incorporate both the present invention and traditional image processing technigues, such that image processing could be tailored to meet the needs of the particular application. As an alternative, some shades of a particular colour within the palette could be desaturated, that is a component of the other colours incorporated into that section of the palette, in order to add to the brightness of that colours pixels (that is those pixels whose colour depends on the contents of that particular colour's section of the palette). Clearly, green would be likely to be chosen for desaturation and the associated increased brightness as this is the optimum colour that the eye can detect.

Although a particular embodiment of the present invention has been described herein, it will be appreciated that many modifications are possible within the scope of the amended claims.

For example, although a particular arrangement is illustrated in Figure 3 for processing the data output from the display buffer of a display adapter for addressing a palette, an alternative embodiment could comprise logic (hardware or software) for preselecting the colour component data for respective screen pixels and storing this in the display memory of a display system. In such an embodiment, the select component logic and the index generator logic of Figure 3 would not be required, the palette being addressed directly by the contents of the display memory.

Claims

1. A display system comprising a bit map for storing source data for an image pixel, a palette containing data for controlling pixel colour on screen, logic for addressing the palette, wherein the palette is a universal palette divided into sub-palettes, at least one sub-palette containing data for controlling pixel colour for display anywhere on screen (4), said data corresponding to shades of dominantly one colour, wherein the data within said sub-palette are addressed via an index generated from an associated portion of the source data.

2. A display system as claimed in Claim 1, in which image pixels are grouped to form a macro-pixel (M), wherein the portion of source data utilised to generate the index to address a sub-palette and the sub-palette addressed are dependent on the position of the image pixel within the macro-pixel (M).

3. A display system as claimed in either of the preceding claims, wherein the source data includes red (r), green (g) and blue (b) colour component data.

4. A display system as claimed in any of the preceding claims, wherein the palette is divided into a first, a second, a third and a fourth sub-palette, the first sub-palette for storing red colour data, the second for storing green colour data, the third for storing blue colour data, and the fourth for storing grey colour data.

5. A display system as claimed in claim 4 when dependent on claim 3, including logic (1) for selecting colour component data to be supplied to index generation logic (2) such that the red colour component source data is utilised to address the sub-palette storing dominantly red colour data, blue colour component source data to address the sub-palette storing dominantly blue colour data and the green colour component source data to address the sub-palette storing dominantly green colour data.

6. A display system as claimed in Claim 1, in which all source data are utilised to generate the index to the sub-palette.

7. A display system as claimed in Claim 6 wherein the sub-palette contains grey colour data.

8. A display system as claimed in any of the

preceding claims, wherein the colour data stored within a sub-palette are desaturated.

9. A method for processing image pixel data wherein a palette is divided into sub-palettes, at least one sub-palette containing data for controlling pixel colour anywhere on screen, said data corresponding to shades of dominantly one colour, wherein the sub-palette is accessed by an associated portion of image pixel source data.

10. A method as claimed in Claim 7, in which the portion of the source data utilised to address a sub-palette, and the sub-palette addressed, are dependent on the position of the image pixel when displayed on screen.

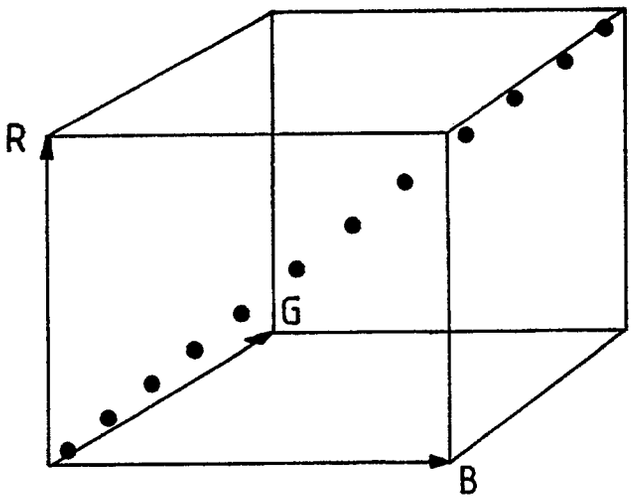

FIG. 1

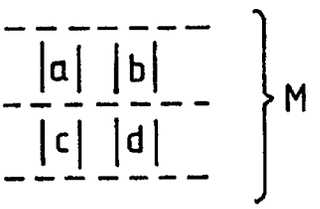

FIG. 2

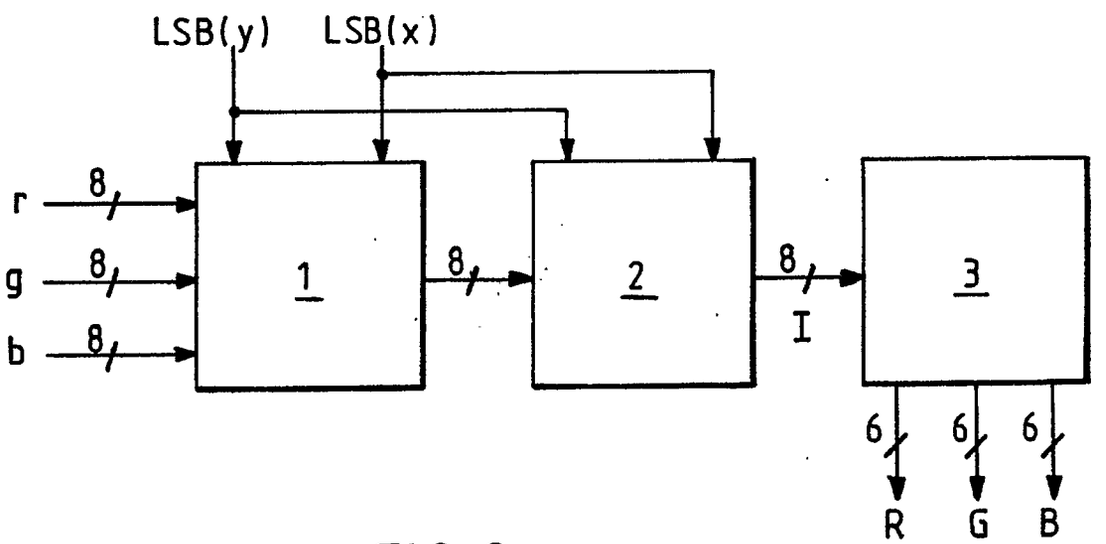

FIG. 3

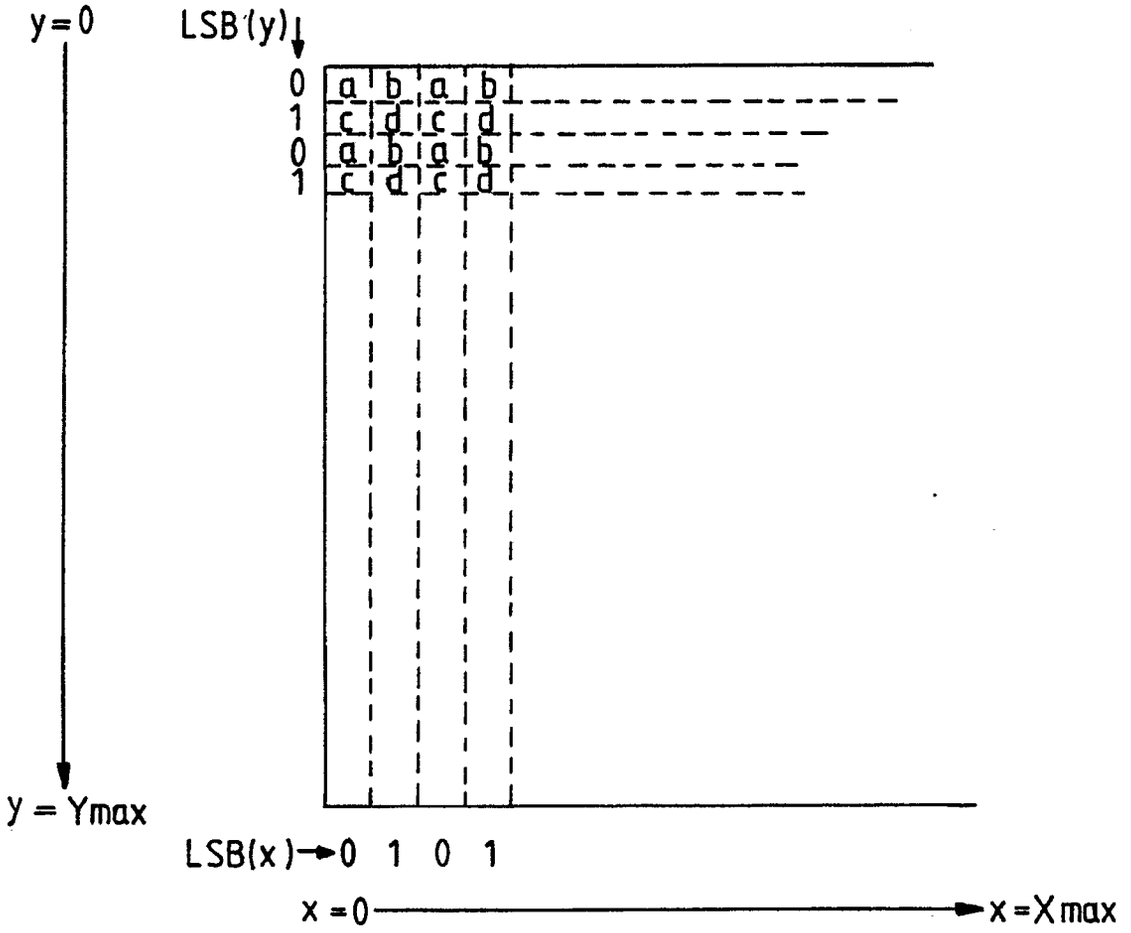

FIG. 4

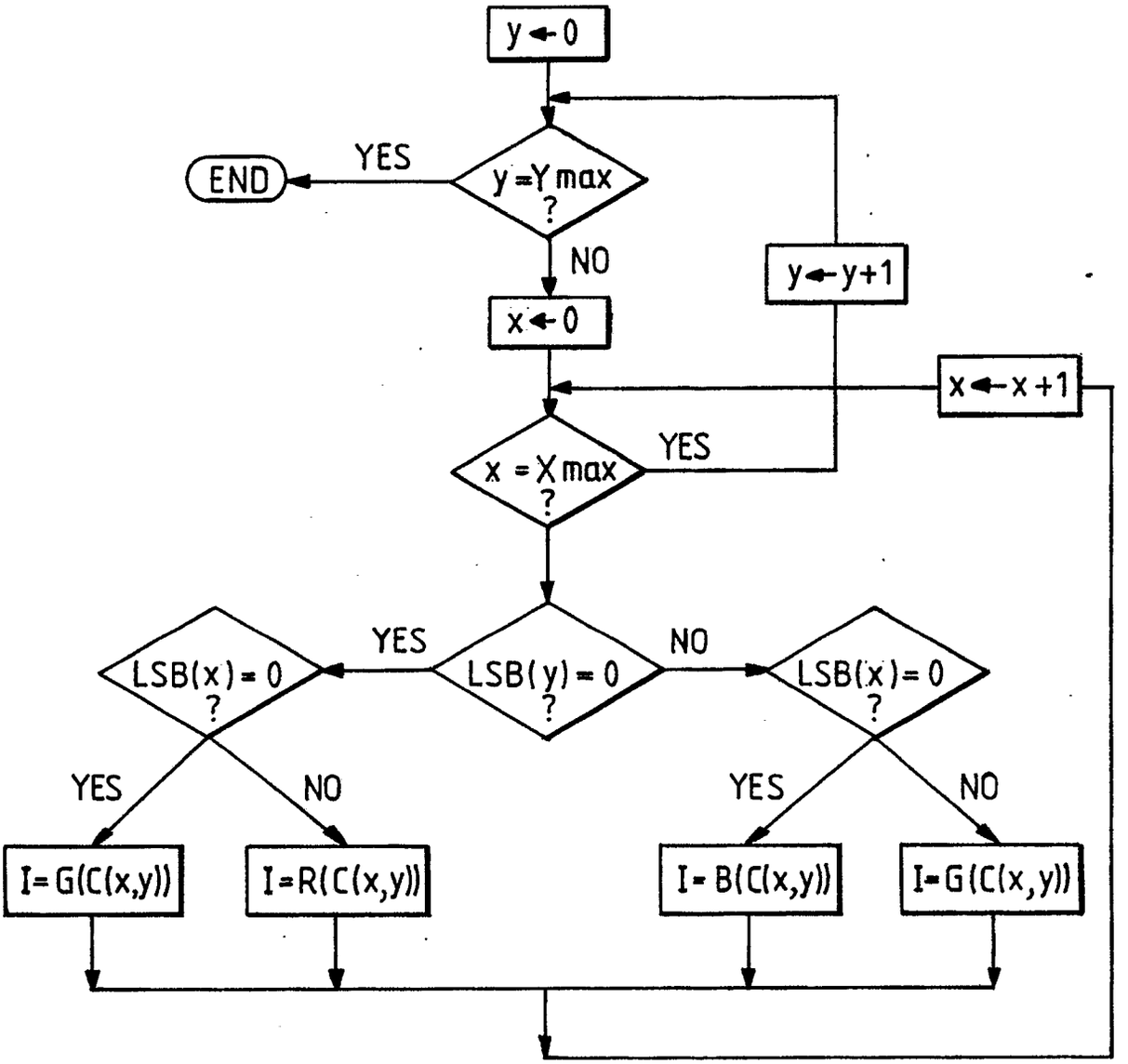
y←0
y=Y max ?
YES
END
NO
y←y+1
x←0
x←x+1
x = X max ?
YES
LSB(x) = 0 ?
YES
LSB(y) = 0 ?
NO
LSB(x) = 0 ?
YES
NO
YES
NO
I= G(C(x,y))
I=R(C(x,y))
I= B(C(x,y))
I=G(C(x,y))

FIG. 5

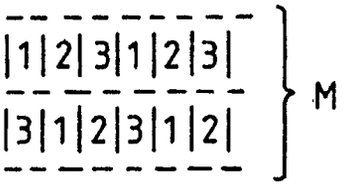
|1|2|3|1|2|3|
|3|1|2|3|1|2|
M

FIG. 6

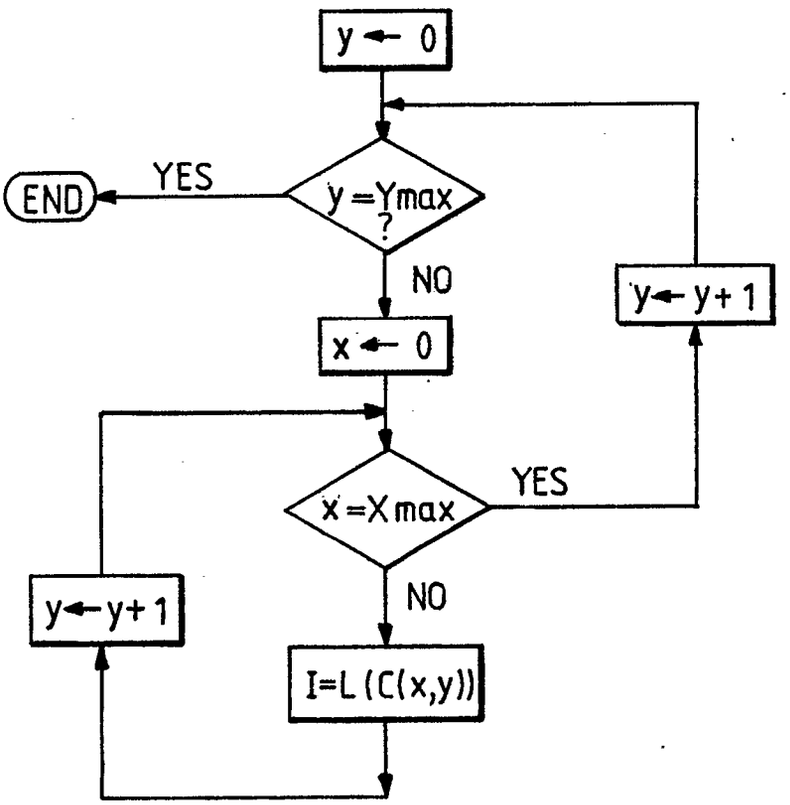
y ← 0
y = Ymax ?
YES
END
NO
x ← 0
y ← y+1
x = Xmax
YES
NO
y ← y+1
I=L(C(x,y))

FIG. 7