

EUROPEAN PATENT APPLICATION

Application number: 85101561.0

Int. Cl.⁴: **G 09 G 1/16**
G 06 F 3/153

Date of filing: 13.02.85

Priority: 20.02.84 JP 28784/84

Date of publication of application:
25.09.85 Bulletin 85/39

Designated Contracting States:
DE FR GB NL

Applicant: Ascii Corporation
Sumitomominamiaoyama Bldg. 11-5, Minamiaoyama
5-chome
Minato-ku Tokyo(JP)

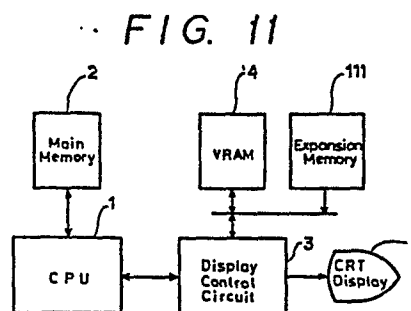
Inventor: Ishii, Takatoshi
23-6 Imai 3-Chome
Oome-shi Tokyo(JP)

Representative: Lehn, Werner, Dipl.-Ing. et al,
Hoffmann, Eitle & Partner Patentanwälte Arabellastrasse
4 (Sternhaus)
D-8000 München 81(DE)

Display control unite.

A display control unit (3) is disclosed which is able to change the processing form of a central processing unit (1) (CPU) for increased processing speeds. In this display control unit (3), when the execution time of its display operation is reduced by reading out data in a source area from a memory unit (4) and then writing the read-out data sequentially into a destination area, a predetermined period of time before the beginning of a vertical or horizontal retrace, the CPU(1) is interrupted so that the CPU (1) can omit its confirmation as to whether a video CPU has completed its previous processing.

Also, in this display control unit (3), an updating processing of color specification in display data can be performed at increased speeds, only desired dots on a display screen (5) can be logically operated, a form or an object having a solid body on a transparent background within a source area can be transferred at higher speeds, and an expansion memory (111) can be used as a "kanji (chinese character)" ROM (pattern memory) or the like.



DISPLAY CONTROL UNIT

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to an improved display control unit for computer systems.

2. Description of the Prior Art

5 In Fig. 1, there is illustrated a block diagram of a conventional color graphics display unit.

As shown in Fig. 1, there is provided a CPU (micro-processor) 1 to control a whole system, and to this CPU 1 are connected a main memory 2 and a display control circuit 3. The main memory 2 is dedicated to holding programs and data, and the display control circuit 3 is used to control the color graphics display. Reference character 4 designates a VRAM (video memory) to hold data for CRT display, and character 5 represents a CRT color display unit.

Fig. 2 shows a block diagram of an example of the display control circuit 3 illustrated in Fig. 1.

In this example, a timing controller 11 generates a clock signal which is in turn inputted to a counter 12 having a column counter and a line counter. This counter 12

is then operated to generate a synchronous signal for CRT display via a display timing circuit 13. On the other hand, the counter 12 creates display addresses which are then outputted as VRAM addresses via a multiplexer 15.

Read data for display access coming from VRAM 4 are inputted to a video output controller 20 via a buffer 19 to produce CRT video signals.

On the other hand, when CPU 1 tries to access VRAM 4, the addresses of VRAM 4 are set in a VRAM address register 14. And, if a write strobe is input to a CPU interface controller 18, then the multiplexer 15 selects the outputs of the VRAM address register 14 given by CPU 1 as VRAM addresses, and also the write data from CPU 1 is written into VRAM 4 via buffers 16, 17.

Fig. 3 illustrates an example of the VRAM 4 which has a series of physical addresses as a memory unit. Logically, it has a display screen as shown and the display screen comprises 256 horizontal dots x 1024 vertical dots.

Normally, a display screen is physically composed of 200 vertical dots. Therefore, the logical existence of 1024 vertical dots means that the screen has unobserved areas or that a plurality of screens are present.

Now let us consider an example: On the display screen shown in Fig. 3, coordinates of X, Y are used to superpose the color code block data in the source area within VRAM 4 on the color code data in the destination area.(area to
5 which data is transferred).

CPU 1 calculates the physical addresses of the VRAM 4 based on the coordinates (Sx, Sy) in the source area and then sets them in the VRAM address register 14 within the display control circuit 3. CPU 1 also produces a read
10 command output so as to read the color code data within the VRAM 4 corresponding to the coordinates (Sx, Sy).

Next, CPU 1 calculates the physical addresses within the VRAM 4 based on the coordinates (Dx, Dy) in the destination area to which the data is to be transferred and sets
15 them in the VRAM address register 14 within the display control circuit 3. Here again, CPU 1 outputs a read command to read the color code data within the VRAM 4 corresponding to the coordinates (Dx, Dy), and then obtains the OR of (i.e., ORs) the color code data within the VRAM 4 and the color code
20 data from the above-mentioned coordinates (Sx, Sy). Then, CPU 1 outputs a write command to write the ORed color code data into the VRAM 4, in particular, a location of the VRAM 4 corresponding to the coordinates (Dx, Dy).

Accordingly, the above-mentioned Read/Read/Logical operation/Write sequence is repeated NX times with respect to a horizontal direction and NY times with respect to a vertical direction (that is, NX X NY times) so as to
5 superpose the source area color code data on the destination area color code data.

In order to meet the demands for realization of compact computers and reduced costs, such conventional display control unit for personal computers is designed to reduce the
10 amounts of its hardware, e.g. the number of gates and IC elements used, involving the internal structure of the display control unit and its associated interfaces. This, however, results in increased load on the software accordingly.

15 As can be seen from the above-mentioned example, namely, the color code data transfer/superposition, in the prior art personal computer display control units, all of the processings must be performed by CPU 1 and thus it takes quite a lot of time to execute these processings.

20 In another aspect of the conventional display control unit, CPU 1 and Display Control Circuit 3 are operated independently of each other, and the display timing of the display control circuit 3 has priority over the VRAM timing of

the CPU 1. As a result of this, a wait time arises in access from CPU 1 to VRAM 4, which results in the extremely deteriorated efficiency of the data transfer.

5 In other words, in the above-mentioned prior art, the software must share a heavy load in the display control and thus it takes a very long time to execute its operation. When a computer is upgraded with increased display specifications and with a plurality of display modes, then its address calculation becomes more complicated and thus it will
10 take an extended period of time to perform its running operation.

Also, in the market there are urgent needs for not only reduction of the single block data transfer running time but also for reduction of the transfer run time for a various
15 kinds of block data. Further, there exists a need for new other elements.

For example, the following are necessary

to replace the CPU processing mode during a retrace period by that during a display period;
20 to speed up the updating process of specifying color in the color data; to perform a logical operation for only desired dots on the display screen; and, to transfer at higher speeds a form or an object having a solid body on a

transparent background within a source area. Further, it is desired to be able to display "kanji" patterns fast so as to facilitate accommodation of the kanji.

5 SUMMARY OF THE INVENTION

The present invention aims at eliminating the drawbacks in the above-mentioned prior art display control unit as well as meeting the demands currently requested in the market.

10 Accordingly, it is a primary object of the invention to provide an improved display control unit which, when a run time for a display operation is reduced by reading out data within a source area and writing the read-out data sequentially into a destination area, is able to speed up a command processing during a vertical or horizontal retrace period.

15 It is another object of the invention to provide a block data transfer device which is capable to speed up an updating processing of specifying color in display data.

20 It is another object of the invention to provide a display control unit which is capable to perform a logical operation for only desired dots on a display screen.

It is still another object of the invention to provide a display control unit which is capable to transfer at high speeds a form or an object having a solid body on a trans-

parent background within a source area.

It is yet another object of the invention to provide a memory expansion system in which an expansion memory can be employed as a "kanji" ROM (pattern memory) or as a
5 buffer area.

In accomplishing these objects, according to the invention, there is provided an improved display control unit in which the processing form of its CPU can be modified, that is, when the execution time of its display operation
10 is reduced by reading out data in a source area from within a memory unit and sequentially writing the read-out data into a destination area, a predetermined period of time before the beginning of a vertical retrace

the CPU is interrupted so that the CPU can omit its
15 confirmation as to whether a video CPU has completed its previous processing.

The above and other related objects and features of the invention will be apparent from a reading of the following description with reference to the accompanying drawings and the
20 novelty thereof pointed out in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of a typical example of the prior art color display units;

Fig. 2 is a block diagram of a display control circuit employed in the prior art color display unit of Fig. 1;

Fig. 3 is a block diagram of an example of VRAMs used in the color display unit of Fig. 1, illustrating the transfer operation of a block data;

Fig. 4 is block diagram of an embodiment of the invention;

Figs. 5 and 6 are view to illustrate the contents of the respective registers employed in the above-mentioned embodiment of the invention;

Fig. 7 is a view to illustrate the command codes used in the above embodiment;

Fig. 8 is a view to illustrate a logical operation used in the above embodiment;

Fig. 9 is a timing view of another embodiment of the invention;

Fig. 10 is a circuit diagram of main portions of the embodiment of Fig. 9;

Fig. 11 is a block diagram of still another embodiment of the invention; and,

Fig. 12 is circuit diagram of main portions of the embodiment shown in Fig. 11.

DETAILED DESCRIPTION OF PREFERRED
EMBODIMENTS OF THE INVENTION

In Fig. 4, there is illustrated a block diagram of an embodiment of the invention.

5 In the illustrated embodiment, there are provided a clock generator 31 to generate a display timing clock and a counter 32 which comprises a column counter to generate a CRT screen display timing and a VRAM address in accordance with the display timing clock, a line counter, and a row
10 counter.

 Data bus 41 from CPU 1 is connected via Buffer 42 to Register bus 43. The number of registers within a display control circuit 3 accessed by CPU 1 is held by a register pointer/counter 44 and the outputs of this register pointer/
15 counter 44 are decoded by a register selector decoder 45, so that the registers can be specified individually. In addition to this register function, the register pointer/counter 44 has a count-up function. That is, in setting parameters of the respective registers, after one setting has been
20 completed, it counts up 1. Thus, the registers can be automatically specified successively one by one.

 Command Register 46 holds the command information from CPU 1, and Video CPU 47 processes the display data in accord-

ance with the commands from CPU 1. SR Register 48 holds the status from Video CPU 47 to CPU 1. When CPU 1 specifies the physical address of VRAM 4 to access the VRAM 4, the VRAM address is to be held by VRAM Address Register/Counter 37. Color Code Register 33 holds the read data from VRAM 4 as well as the write data to VRAM 4.

The features of the invention include the below-mentioned component elements:

At first, there are included SX Register/Counter 38 to hold the values on X coordinates existing in a horizontal direction within a source area, SY Register/Counter 39 to hold the values on Y coordinates in a vertical direction, and SX Address Composing Circuit 40 to create the physical address of VRAM 4 in accordance with the respective outputs of SX, SY Register/Counter 38, 39.

Also, there are provided DX Register/Counter 58 to hold the values on the horizontal X coordinates of a destination area, DY Register/Counter 59 to hold the values on the vertical Y coordinates of the same, and DX Address Composing circuit 57 to create the physical address of VRAM 4 in accordance with the respective outputs of DX, DY Register/Counters 58, 59.

The above-mentioned SX, SY, DX, DY Register/Counters 38,

- 11 -

39, 58, 59 have not only a register function but also an up/down counter function, respectively.

Further, VRAM Address Bus 36, which is contained within the display control circuit 3, is connected through Buffer 55 to Address Line 56 of VRAM 4. VRAM Data Bus 35, which is also contained in the display control circuit 3, is connected through Buffer 53 to VRAM Data Line 54.

NX Register 61 holds the transfer data number in a horizontal direction (that is, a direction of the X coordinates), while NY Register 63 holds the transfer data number in a vertical direction (that is, a direction of the Y coordinates). There is provided a horizontal direction flag 60 which points to a positive direction (or, right direction) when it is "0" and points to a negative direction (or, left direction) for "0". A vertical direction flag 62 indicates a positive direction (or, downward direction) when it is "0", and a negative direction (or, upward direction) for "1". S Register 34 holds the read data from the source area, and D Register 52 holds the read data from the destination area. ALU (arithmetic and logic unit) 51 performs logical operations such as IMP, AND, OR, EOR, NOT operations of the outputs of S Register 34, Color Code Register 33 and D Register 52 in accordance with the control from Video CPU 47.

Interrupt line (IL) Register 70 is used to previously set the number of columns, lines, or rows for IL interruptions, while Comparator 71 is dedicated to detecting the correspondence of the number of the columns, lines, or rows set by IL Register 70.

Referring now to Fig. 10, Source Data Bit Selector 101 selects higher 4 bits or lower 4 bits from the source data and then uses the selected 4 bits to form higher 4 bits or lower 4 bits. Byte Data Selector 102 selects the data which has passed through Source Data Bit Selector 101 or the source data from S Register 34.

Transparence Detection Circuit 104 serves to detect a color code (that is, transparency) for a portion of the source area in which no object exists.

Parallel Bit Selector 103 causes omission of logical operations of some of the color codes within the destination area which corresponds to the color code within the source area, provided that the very source area color code happens to be transparent.

Referring to Fig. 11, Expansion Memory 111 is used as "kanji" ROM(Pattern Memory) or a buffer area.

Referring to Fig. 12, Slot Switch 121 is dedicated to switching a video request or a process request.

ARGR Switch 123 is used to switch a request to a video request or a process request in accordance with the respective bits of an argument register.

5 The foregoing elements are the characteristic ones of the present invention. Although there exist other elements than the foregoing in the display control circuit 3, an explanation is omitted here regarding such elements as not necessary in describing the operation of the invention.

10 Next, we will describe the operation of the above-mentioned embodiment.

First, the operation of the display control circuit 3 is described, by way of example, in connection with the block data transfer/superposition based on X, Y coordinates.

15 It is necessary that CPU 1 has previously set information necessary for the logical operation and block data transfer in the respective registers of the display control circuit 3. When accessing the respective registers shown in Figs. 5 and 6, CPU 1 sets the register number of a register to be accessed in a register pointer/counter 44 and thereafter performs its read/write operation.

20

When the source area color code data within VRAM 4 and the destination area color code area are ORed and superposed on each other, "10010010" is to be set in Register #45

(Command Register). Its higher 4 bits "1001" represents a command code shown in Fig. 7 (block data transfer from VRAM 4 to VRAM 4 with a logical operation), while its lower 4 bits "0010" expresses an OR shown in Fig. 8.

5 Also, when such block data as shown in Fig. 3 is to be processed, it is necessary to set up the following parameters: The start coordinates (SX, SY) of the source area are set in SX Register/Counter 38 and SY Register/Counter 39, respectively. SX Register/Counter 38 is composed of SXL
10 (Register #32) and SXH (Register #33), while SY Register/Counter 39 is composed of SYL (Register #34) and SYH (Register #35). Therefore, CPU 1 sets a 4-byte parameter on the transfer starting point or the starting coordinates (SX, SY).

By the way, Fig. 5 illustrates the contents of Registers
15 #32 ~ 42, and Fig. 6 illustrates the contents of Registers #43 ~ 46 as well as the contents of Registers #2, #8.

Then, the destination area start coordinates (DX, SY) are set in SX Register/Counter 58 and DY Register/Counter 59. DX Register/Counter 58 is composed of DXL(Register #36) and
20 DXH(Register #37), while DY Register/Counter 59 is composed of DYL(Register #38) and DYH(Register #39).

Next, the number NX of data to be transferred to a horizontal direction (a direction of the X coordinates) is

set in NX Register 61 and the number NY of data to be transferred to a vertical direction (a direction of the Y coordinates) is set in NY Register 63. NX Register 61 is composed of NXL (Register #40) and NXH (Register #41), while NY Register 63 is composed of NYL (Register #42) and NYH (Register #43).

Since the block data to be transferred to both of the X, Y directions when viewed from the start coordinates (SX,SY) are in a positive direction, "0" is set in both of Direction Flag 60 and Direction Flag 62. Direction Flag 60 corresponds to the bit 3 of Argument Register ARGR (Register #45), and Direction Flag 62 to the bit 2 of Argument Register ARGR (Register #45). When all of the foregoing settings have been effected, it is considered that the setting of parameters necessary for transfer of the block data has been completed. These parameter settings are to be performed successively from Register #32 to Register #45. First, "32" is set in Register Pointer/Counter 44. Then, it is necessary only to write the parameters successively so that the associated registers can be set up sequentially. After then, Register Pointer/Counter 44 is caused to point to Register #46 and is in a position to wait for setting of a command code.

Fig. 7 is a table to illustrate command codes.

In this figure, "VDC" stands for the display control circuit 3.

5 Fig. 8 is a table to illustrate logical operations. In this figure, SC represents a source color code and DC stands for a destination color code.

CPU 1 creates command codes such as "10010010" in accordance with the above-mentioned command codes and logical operation codes and then sets them in Command Register 45 (Register #45).

10

The higher 4 bits of the above-mentioned command code express an instruction to transfer a block data within VRAM 4 provided that the source area and the destination area are both present within the VRAM 4. On the other hand, the lower 4 bits of the above example represent a logical operation code, that is, "0010" means that the OR of the source color code data and the destination color code data before transferred is to be written into the destination.

15

On receiving a command code and a logical operation code from CPU 1, Video CPU 47 sets the command executing (CE) of the bit 7 of SR Register 48 to initiate the execution and processing of the command.

20

SXY Address Composing Circuit 40 is operated to create

a physical address of VRAM 4 from SX Register/Counter 38 and SY Register/Counter 39 respectively holding the source area coordinates under control of Video CPU 47. According to this physical address, a color code data is read out from VRAM 4, which read-out data is in turn set in S Register 34 through Data Line 54, Buffer 53, and VRAM Data Bus 35.

Next, DXY Address Composing Circuit 57 is operated to create a physical address of VRAM 4 from the outputs of DX Register/Counter 58 and DY Register/Counter 59 respectively holding the destination area coordinates. According to this physical address, a color code data is read from VRAM 4, which data is in turn set in D Register 52.

On the other hand, a color code data within S Register 34 read out from the source side and a color code data in D Register 52 read out from the destination side are ORed by ALU (Arithmetic and Logic Unit) 51 to produce a superposed color code data.

The newly operated and produced color code data is output on VRAM Data Line 54 via VRAM Data Bus 35 and Buffer 53, and is then written into VRAM 4 in accordance with a physical address on the destination side produced by DXY Address Composing Circuit 57.

The above operations complete the logical operation and

data transfer of a color code data of 1 dot.

In the same manner as with the block data transfer based on X, Y coordinates, logical operations (OR) and block data transfer operations are performed on the sum ($NX \times NY$) of X-direction NX color code data and Y-direction NY color code data.

When NX Register 61 coincides with NX Counter 64 and NY Register 63 coincides with NY Counter 65, then Video CPU 47 decides that the logical operation (OR)/ block data transfer has been completed, clears the command executing (CE) bit within SR Register 48, and informs CPU 1 of the end of the command.

In the above discussion, the logical operation (OR)/ block data transfer based on the X, Y coordinates only within VRAM 4 has been explained. However, it is similarly possible to perform other logical operation/block data transfer processings in accordance with commands which specify other combinations. Such cases will be discussed from now on.

(1) Logical operation/block data transfer from CPU 1 to VRAM 4 (Command code CM 3~0 "1011"):

In this case, since the source is CPU 1, SX Register/counter 38, SY Register/counter 39 and S Register 34 are not used, but instead Color Code Register 33 is used.

Specifically, when CPU 1 sets a transfer data in Color Code Register 33 and Video CPU 47 writes the transfer data of Color Code Register 33 into VRAM 4 according to DX Register/counter 58 and DY Register/counter 59, the transfer ready (TR) bit of SR Register 48 is set to inform CPU 1 that one data transfer is completed and that Color Code Register 33 is now ready to receive a next data.

After confirming that this TR bit is "1", CPU 1 sets a next transfer data in Color Code Register 33. This resets the TR bit and returns it to its original status. Other operations are the same as with the above-mentioned block data transfer within VRAM 4.

(2) Logical operation/block data transfer from VRAM4 to CPU 1 (Command code CM 3-0 "1011"):

In this instance, since the destination is CPU 1, a color code data from CPU 1 (that is, a fixed data) is set into D Register 52 via Color Code Register 33. The color code data, after operated, is set in Color Code Register 33 and is then read by CPU 1.

Video CPU 47 reads out a transfer data from VRAM 4 in accordance with SX Register/counter 38 and SY Register/counter 39 and sets the read-out transfer data in Color Code Register 33, and at the same time sets the TR bit of SR Register 48 for "1".

CPU 1 checks this TR bit, and, if it is "1", then CPU 1 reads out the transfer data from Color Code Register 33. As a result of this, the TR bit is reset and returns to its initial status. Other operations are performed in the same way as in the above-mentioned data transfer within VRAM 4.

(3) Logical operation/block data transfer from a single register (Color Code Register 33) within the display control circuit 3 to VRAM 4 (Command Code CM3-0 "1010"):

In this case, the data written into Color Code Register 33 is transferred to the destination area of VRAM 4. This is an effective way in writing the same data. The procedure of this operation is the same as in the above-mentioned block data transfer from CPU 1 to VRAM 4. However, in this instance, CPU 1 is required to write the data into Color Code Register 33 only once, and the data is to be transferred under control of Video CPU 47.

(4) Various logical operations including not only the OR but also the AND, XOR, complement or the like of the source area color code data and the destination area color code data can also be performed at high speeds by ALU 51 (in accordance with the instructions from Command Register L $\bar{0}$ 2~0).

The operations concerning the above-mentioned articles (2) and (3) are performed by CPU 1 in cooperation with Display Control Circuit 3, which means both of them must wait while their partners are in execution mutually. Such mutual execution wait is controlled by setting and resetting the TR bit of SR Register 48.

Conditions of the above mentioned execution wait are different between in the display period and in the retrace period. Namely, during the return line period, since

all of the memory accesses can be used for command processing, the command processing can be executed at high speeds so that the waiting of CPU 1 is eliminated. In particular, a vertical return line period is longer than a horizontal retrace

period and thus it takes longer time to process commands in the vertical retrace period than in the horizontal retrace line period. Accordingly, if it is possible to employ such a command processing system as can avoid the waiting of CPU 1 in the vertical period, then the performance of the display control unit can be enhanced substantially. To this end, when the vertical retrace period comes near, an interrupt (or, IL interrupt) is caused to occur to inform CPU 1 of the effect.

Such IL interrupt is enabled by previously having set

- 22 -

the value of Vertical Counter (Line, Row) 32 in IL Register 70 shown in Fig. 4 (an interrupt line register, #8 Register shown in Fig. 6).

5 The value to be set for this purpose may be the start line number of the vertical retrace or, in case when the overhead time for interrupt processing is long, such a value may be set as causes an interrupt to take place earlier by such long time. In either event, it is possible to improve the efficiency of the display control unit.

10 During the vertical retrace, SR Register 48 shown in Fig. 4 (a status register, or #2 Register shown in Fig. 6) is sometimes checked for its VR bit (that is, the output of the VR status signal produced by decoding the output of Counter 32 is read out) to decide whether a processing
15 in the very vertical retrace is to be continued or not.

20 The above-mentioned VR bit is produced by decoding the out-put of Counter 32 such that it becomes "0" a predetermined time before the termination of the vertical retrace line period. It is necessary to set up the predetermined time duration even when a processing happens to be longest during the vertical retrace so as to prevent such longest processing from being carried over into the display

period.

Similarly as with the above-mentioned IL interrupt occurred when the vertical retrace approaches, the status of the horizontal retrace can also be shifted forwardly in time to increase its processing efficiency.

When CPU 1 performs a processing aiming at the horizontal retrace, it is necessary during such processing to check SR Register 48 for its HR bit. In this case, a timing to generate the HR bit can also be staggered in the following manner to enhance the processing efficiency. That is, in the repetitive processings during the horizontal retrace,

the leading edge of the timing may be shifted before a minimum time duration required to issue a VRAM access after detection of the HR bit, and the trailing edge thereof may be staggered before more than the maximum time duration.

Fig. 9 illustrates the timing mentioned above.

The predetermined time duration from the beginning of the vertical or horizontal retrace necessary to cause an interrupt to CPU 1 may be changed depending upon the time periods from the generation of an interrupt signal to CPU 1 to the initiation of the interrupt processing. Also, this predetermined time duration must be changed in accordance with the length of the execution time of programs.

Also, during the vertical and horizontal retrace,
the processing mode of CPU 1 is changed by causing
CPU 1 to omit the confirmation as to whether Video CPU 47
has completed its previous processing, and at the same time
5 a status signal to indicate the then processing is present
within the retrace period is monitored so as to continue
to change the processing mode of CPU 1. The status signal
has been previously extinguished within a predetermined time
after the termination of the retrace. Then, the
10 processing mode of CPU 1 is returned to its original way,
in which CPU 1 confirms whether Video CPU 47 has completed
its previous processing.

Fig. 10 illustrates a block diagram of another embodi-
ment of the invention, showing an example in which an up-
15 dating processing of color specification in the display data
can be performed at increased speeds.

The foregoing discussion is concerned with the memory
contents of one memory address and is limited to the 1 dot
display system. However, in general, a memory interface
20 consists of a byte(8 bits) or a word (16 ~ 32 bits) and thus
contains information on display of a plurality of dots. In
this case, when the dots are to be processed one by one, the
remaining bits must be masked.

Next, we will describe the operation to be performed when a byte interface has 4-bit color information (2 dots/byte). As it contains the information about 2 dots/byte, the source data and destination data are respectively selected for each
5 bit.

Source Data Selector 101 selects higher 4 bits when 0 bit of SXY is "0", and lower 4 bits when this 0 bit is "1". The data is then transferred via Byte Data Selector 102 to ALU 51. In ALU 51, a logical operation of the data and the
10 value of D Register 52 is executed for each bit. After then, Parallel Bit Selector 103 outputs as VRAM data either 4 bits (that is, the upper 4 bits for "0", the lower 4 bits for "1") to be specified in accordance with the value of the bit 0 of DXY.

15 If the value of the source data bus is "0" and $L\bar{0} 3 = 1$, Transparency Detection Circuit 104 decides the source data as transparent and thus Parallel Bit Selector 103 allows the value of D Register 52 as it is to pass through.

In this manner, the bit select/mask function and trans-
20 parency processing can be realized.

In other words, the color code data within the source area and the color code data within the destination area are logically operated, while a portion belonging to the source

area and having no object existing therein, more particularly the color code (transparent color) of such portion is detected by Transparency Detection Circuit 104 and the logical operation of the transparent portion is omitted, so that only the form stored in the source area and having a solid body can be transferred at high speeds.

The above-mentioned operations can be realized similarly even if the number of the color information bits or the number of bits / word may be varied.

The above discussion relates to a bit-by-bit processing. According to cases, however, it may be necessary to process data in bytes for purposes of higher speeds. In this instance, the command code "1111 ~ 1100" is used. Source Data Bit Selector 101 is not used for this purpose, but the value of S Register 34 is directly guided to ALU 51 by Byte Data Selector 102 (CM2 = 1), and then the output of ALU 51 is forcibly guided to VRAM Data Bus 35, so that a higher speed processing can be executed.

In other words, a part of the display data read out to the destination register is modified and the modified display data is written into VRAM 4, so that it is possible to speed up the updating processing of the color specification in the display data.

Source data specified by a source address is divided into a plurality of portions and one of the data portions is then selected, while destination data specified by a destination address is also divided into a plurality of data portions and one of the data portions is selected. After logical operations are performed on the thus selected data portions, the logical operation result or the destination data is selected for each of the portions of the respective data. As a result of this, it is possible to perform logical operations only on a desired dot on the display screen.

Fig. 11 is a block diagram of another embodiment of the invention, illustrating an example using an expansion memory as a "kanji"(chinese-character) accomodation or as a buffer area.

In Fig. 11, Expansion Memory 111 is extended in parallel with VRAM 4. For example, when this expansion memory 111 is extended in parallel with VRAM 4 as a chinese-character pattern ROM, chinese characters ("kanji") can be accommodated. This is because a high-speed display can be realized by transferring a chinese-character pattern to VRAM 4 by means of area movement. Also, it is convenient since there is no need to load any external pattern data for this purpose. Further, since the read speed of the chinese-character pattern ROM, or

the cycle time of the area movement in this case may be slower than the display memory access, low-speed and large-capacity memory elements can be used. To achieve this, an address register may be located within the expansion memory and, at the time when the previous access has been completed, an address may be updated to initiate the next reading.

Also, when an RAM is extended as an expansion memory, this can be used as a work memory of VRAM 4 to expand the address space up to the same capacity as that of VRAM 4. Specifically, bits MXC, MXD and MXS of ARGR are defined. MXC, which serves to switch and control the VRAM access from CPU 1, makes it possible to directly read and write an expansion memory from CPU 1. MXD specifies the destination area to the expansion memory and makes it possible for the expansion memory to be read and written as the data memory. MXS specifies the source area to the expansion memory and permits reading of a fixed pattern or reading from a buffer memory.

Fig. 12 is a circuit diagram of the main portions of the embodiment shown in Fig. 11.

Next, the operation of the embodiment in Fig. 11 will be described with reference to Fig. 12.

Access requests for a memory are normally classified into two main categories; namely, video requests (VRQ) and

process requests (PRQ). The video requests VRQ are requests for reading of data for CRT display and are issued based on the counts of Counter 32. The process requests PRQ are VRAM accesses that are issued by Video CPU. This issue results from the controls of CPU such as setting of parameters from CPU 1, starting of commands, VRAM access and the like.

The video requests VRQ and process requests PRQ are controlled by timing control signals, and are respectively permitted by the allocated time slots. These operations are processed by Slot Switch 121 shown in Fig. 12. That is, at a timing when the video request VRQ is issued, Slot Switch 121 is always connected to the video request VRQ side, and in other cases it is connected to the process request PRQ side. Thus, PRQ is permitted only when Slot Switch 121 is connected to the PRQ side.

Next, we will describe the operation and function of the bits MXC, MXD, MXS of ARGR.

The contents of the process request PRQ are classified into CRQ to be issued when CPU 1 accesses VRAM 4 directly, DRQ issued when the destination data is accessed while Video CPU 122 is executing a command, and SRQ issued when the source data is accessed. These requests are normally connected to the process request PRQ side by ARGR Switch 123.

This ARGR Switch 123 is adapted to connect the requests, CRQ, DRQ and SRQ to an XRQ side in accordance with each of the bits MXC, MXD and MXS. This XRQ is a memory request (MX request) to the expansion memory, and when the XRQ appears
5 the expansion memory is to be accessed instead of VRAM 4. Thus, by distributing each of the process request or PRQ requests independently to the VRAM 4 expansion memory in this way, the expansion memory can be used as a buffer memory or a pattern memory. When the expansion memory is specified by
10 MXD and VRAM 4 by MXS to specify the area movement, then it is possible to save the data in an area where VRAM 4 is contained. On the other hand, when VRAM 4 and the expansion memory are specified by MXD and MXS respectively, then it is possible to return the saved data to its original state, or
15 to move a fixed pattern (or, a chinese-character pattern) to VRAM 4 for display.

The foregoing description has been made on condition that a color code or color data is to be handled, but this can also apply to a monochrome system. In that case, the
20 color data may be replaced by the byte data.

The present invention is useful in performing display control of not only the color CRT but also other display units such as a monochrome CRT, LCD, plasma, and EL.

As can be clearly understood from the foregoing description, the present invention provides several effects: namely, when reducing the execution time for display operation by reading out the source area data from a memory unit and then writing the read-out source area data sequentially into the destination area, the command processing speed can be increased during a vertical or horizontal retrace line period; it is possible to increase the speed of an update processing on color specification in display data, and at the same time only a desired dot on the display screen can be logically operated; and, it is possible to transfer a form or an object existing within the source area and having a solid body at high speeds, and an expansion memory can be used as a "kanji" (chinese-character) ROM (or, a pattern memory) or a buffer area.

Claims:

1. A display control unit for providing screen image information and a control signal to a display unit (5), said display control unit (3) comprising a column/line/row counter (32) for generating a control timing, characterized by an IL register (70) for storing an interrupt issuing count value, and an IL interrupting means (71) for comparing said count value with the value of said IL register (70) and for outputting an interrupt signal (INT) on detecting the coincidence of said two values, whereby said display control unit (3) is able to generate said interrupt signal at an arbitrary position simultaneously with display scanning.
2. A display control unit as claimed in claim 1, characterized in that said IL interrupt means (71) is initiated to change the processing mode of a central processing unit (CPU) so as to enhance system performance.
3. A display control unit as claimed in claim 1, characterized in that said interrupt means (71) is set a predetermined time before the start of a retrace period and said interrupt signal is generated earlier by the time of the overhead of an interrupt routine so as to make effective use of processings during said retrace period.
4. A display control unit for executing a command process function under control of a central processing unit (CPU), characterized in that the wait time processing of said CPU is changed by initiation of IL interrupting means (71), in that during a display period a command is initiated after confirmation of completion of its previous command, and in that during a retrace a next command is initiated without confirmation of completion of

its previous command, whereby the speed of a command processing during said retrace period is increased.

5. A display control unit as claimed in claim 4, characterized in that said IL interrupting means (71) is set a predetermined time before the start of a retrace period and an interrupt signal is generated earlier by the time of the overhead of an interrupt routine so as to make effective use of processings during said retrace period.

6. A display control unit for providing screen image information and a control signal to a display unit, characterized by means for generating a status signal to indicate that it is within a retrace period based on the value of a column/line/row counter (32) for producing a control timing.

7. A display control unit as claimed in claim 6, characterized in that said status signal is outputted a predetermined time before the actual start of said retrace period.

8. A display control unit as claimed in claim 6, characterized in that said predetermined time is set as a minimum time necessary for transfer to execution of a processing in said retrace period on detection of said status signal.

9. A display control unit as claimed in claim 6, characterized in that said status signal is turned off a predetermined time before the actual termination of said retrace period.

10. A display control unit as claimed in claim 9, characterized in that said predetermined time is set as a

maximum time necessary to execute said processing in said retrace period on detection of said status signal.

5 11. A display control unit for providing screen image information and a control signal to a display unit (5), said display control unit (3) comprising a destination register (52) for reading and storing display data at a destination address to be written by a central processing unit (CPU), and a write register for holding data
10 to be written, characterized in that a portion of said display data is combined with a portion of said data to be written to modify said display data, and said modified display data is written into said destination address so as to be able to change a portion of a word in
15 a display memory.

12. A display control unit as claimed in claim 11, characterized in that said modify function is composed of data for each 1-dot information when a command is
20 executed.

13. A display control unit as claimed in claim 11, characterized in that said modify function is composed of a parallel bit selector.
25

14. A display control unit for providing screen image information and a control signal to a display unit (5), said display control unit being characterized by:

30 means (38,39) for specifying a transfer start point of a source area;

means for specifying a transfer start point of a destination area (58,59);

means (61) for holding the amount of transfer data in a horizontal direction;

35 means (63) for holding the amount of transfer data in a vertical direction;

means (60,62) for holding the moving directions of respective horizontal and vertical transfer points;

first select means (40) for dividing source data specified by a source address into a plurality of data sections and selecting one of said data sections;

second select means (57) for dividing destination data specified by a destination address into a plurality of data sections and selecting one of said data sections;

means (51) for performing logical operations on said data sections selected by said first and second select means; and

third select means (47) for selecting said logical operation result or destination data for each of said sections of said data,

wherein data in said source area specified by said means is read out from a memory unit (4) and said read-out data is sequentially written into said destination area, thereby performing data transfer between said source and destination areas, and wherein said logical operations are performed only on a desired dot on a display screen (5).

15. A display control unit for providing screen image information and a control signal to a display unit, said display control unit being characterized by:

means (38,39) for specifying a transfer start point of a source area;

means for specifying a transfer start point of a destination area (58,59);

means (61) for holding the amount of transfer data in a horizontal direction;

means (63) for holding the amount of transfer data in a vertical direction;

means (60,62) for holding the moving directions of the respective horizontal and vertical transfer points;

transparency detection means (104) for detecting a transparent color that is a color code for a portion of said source area with no object existing therein; and

5 logical operation means (51) for performing logical operations on color code data within said source area as well as on color code data within said destination area and also omitting logical operations on said transparent color portion,

10 wherein data in said source area specified by said means is read out from a memory unit (4) and said read-out data is sequentially written into said destination area, thereby performing data transfer between said source and destination areas, and wherein a form having a solid body within said source area is transferred at
15 high speeds.

16. A memory expansion system characterized in that an extension memory (111) is provided in parallel with VRAM (4) for holding display data, and in that there is provided
20 specification means for specifying either said VRAM or said extension memory for the command processing memory access of CPU or a display control unit, whereby said extension memory (111) is used as a data area of said VRAM (4) to be able to expand the address space of
25 said VRAM equivalently.

17. A memory expansion system as claimed in claim 16, characterized in that said extension memory (111) is used as a buffer area.
30

18. A memory expansion system as claimed in claim 16, characterized in that said expansion memory (111) is used as a pattern memory.

FIG. 1

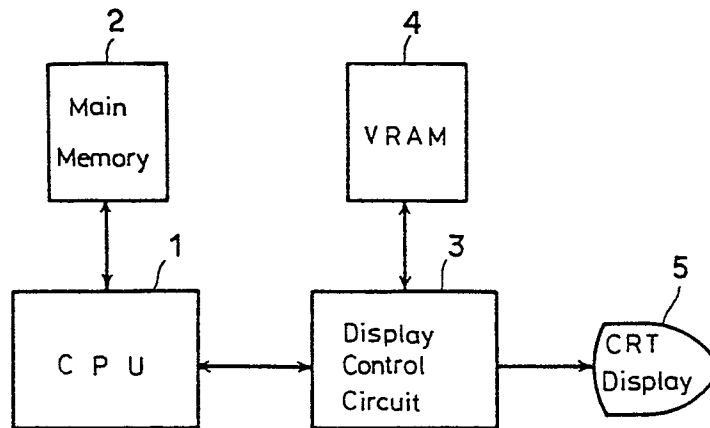


FIG. 3

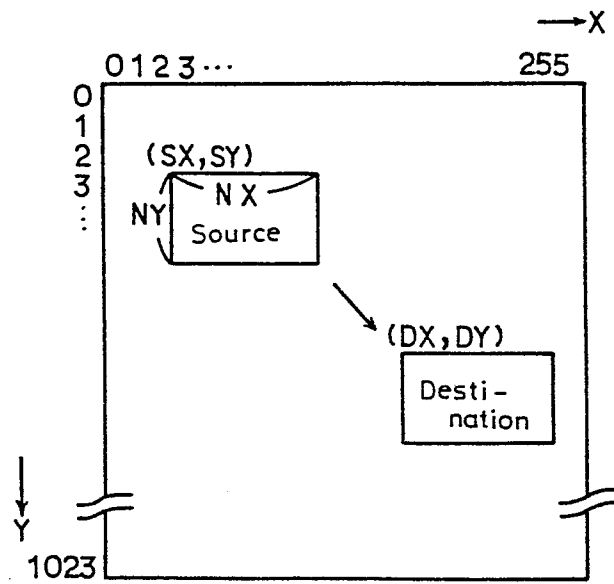


FIG. 2

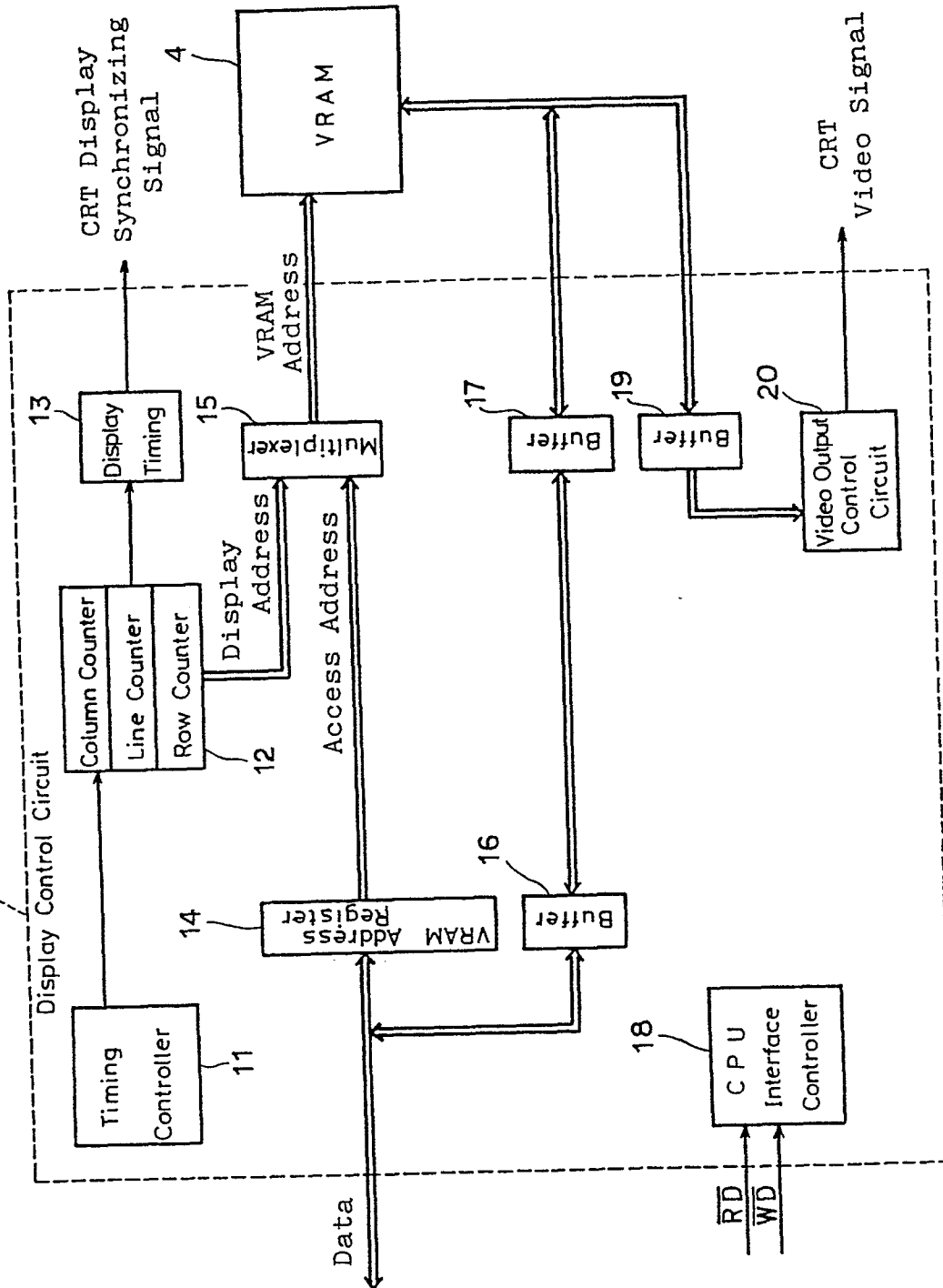


FIG. 5

Register No.

#32	7 0 $X_7, X_6, X_5, X_4, X_3, X_2, X_1, X_0$ (MSB) (LSB)	SXL (Source	X Low)
#33	0, 0, 0, 0, 0, 0, 0 X_8	SXH (Source	X High)
#34	$Y_7, Y_6, Y_5, Y_4, Y_3, Y_2, Y_1, Y_0$	SYL (Source	Y Low)
#35	0, 0, 0, 0, 0, 0 Y_9, Y_8	SYH (Source	Y High)
#36	7 0 $X_7, X_6, X_5, X_4, X_3, X_2, X_1, X_0$ (MSB) (LSB)	DXL (Destination X Low)	
#37	0, 0, 0, 0, 0, 0, 0 X_8	DXH (Destination X High)	
#38	$Y_7, Y_6, Y_5, Y_4, Y_3, Y_2, Y_1, Y_0$	DYL (Destination Y Low)	
#39	* * * * * * Y_9, Y_8	DYH (Destination Y High)	
#40	7 0 $NX_7, NX_6, NX_5, NX_4, NX_3, NX_2, NX_1, NX_0$ (MSB) (LSB)	NXL (Dot Number X Low)	
#41	0, 0, 0, 0, 0, 0 NX_9, NX_8	NXH (Dot Number X High)	
#42	$NY_7, NY_6, NY_5, NY_4, NY_3, NY_2, NY_1, NY_0$	NYL (Dot Number Y Low)	

Register NO.

FIG. 6

#43	7 0 0 0 0 0 0 0 0 (MSB) 0 NY₉ NY₈ (LSB)	NYH (Dot Number Y High)
#44	7 CH₃ CH₂ CH₁ CH₀ LO₃ LO₂ LO₁ LO₀ (MSB) 0 (LSB)	CLR (Color Register)
#45	* MXC MXS MXD DIR DIR EQ MAJ Y X NEQ MIN	ARGR (Argument Register)
#46	CM₃ CM₂ CM₁ CM₀ CL₃ CL₂ CL₁ CL₀	CMR (Command Register)
#2	CE TR HR VR * * * * *	SR (Status Register)
#8	IL₇ IL₆ IL₅ IL₄ IL₃ IL₂ IL₁ IL₀	IL (Interrupt Line Register)

0155499

FIG. 7

CMR CM ₃ CM ₂ CM ₁ CM ₀	Mnemonic Code	Desti- nation	Source	Transfer Unit	
1 1 1 1	HMMC	VRAM	C P U	Byte	High-speed Transfer
1 1 1 0	HMCM	C P U	VRAM	Byte	"
1 1 0 1	HMMM	VRAM	VRAM	Byte	"
1 1 0 0	HMMV	VRAM	V D C	Byte	"
1 0 1 1	LMMC	VRAM	C P U	Dot	Logical Transfer
1 0 1 0	LMCM	C P U	VRAM	Dot	"
1 0 0 1	LMMM	VRAM	VRAM	Dot	"
1 0 0 0	LMMV	VRAM	V D C	Dot	"
0 1 1 1	LINE	VRAM	V D C	Dot	Line Command
0 1 1 0	SRCH	V D C	VRAM	Dot	Search Command
0 1 0 1	PSET	VRAM	V D C	Dot	P Set Command
0 1 0 0	PINT	V D C	VRAM	Dot	Point Command

FIG. 8

CMR 3 2 1 0 LO ₃ LO ₂ LO ₁ LO ₀	Mnemonic Code	Operation
0 0 0 0	I M P	SC $\longrightarrow$ DC
0 0 0 1	A N D	SC * DC $\longrightarrow$ DC
0 0 1 0	O R	SC + DC $\longrightarrow$ DC
0 0 1 1	E O R	SC $\oplus$ DC $\longrightarrow$ DC
0 1 0 1	N O T	$\overline{SC}$ $\longrightarrow$ DC
1 0 0 0	T I M P	If SC=0, then DC $\longrightarrow$ DC, otherwise I M P
1 0 0 1	T A N D	" " A N D
1 0 1 0	T O R	" " O R
1 0 1 1	T E O R	" " E O R
1 1 0 0	T N O T	" " N O T

FIG. 9

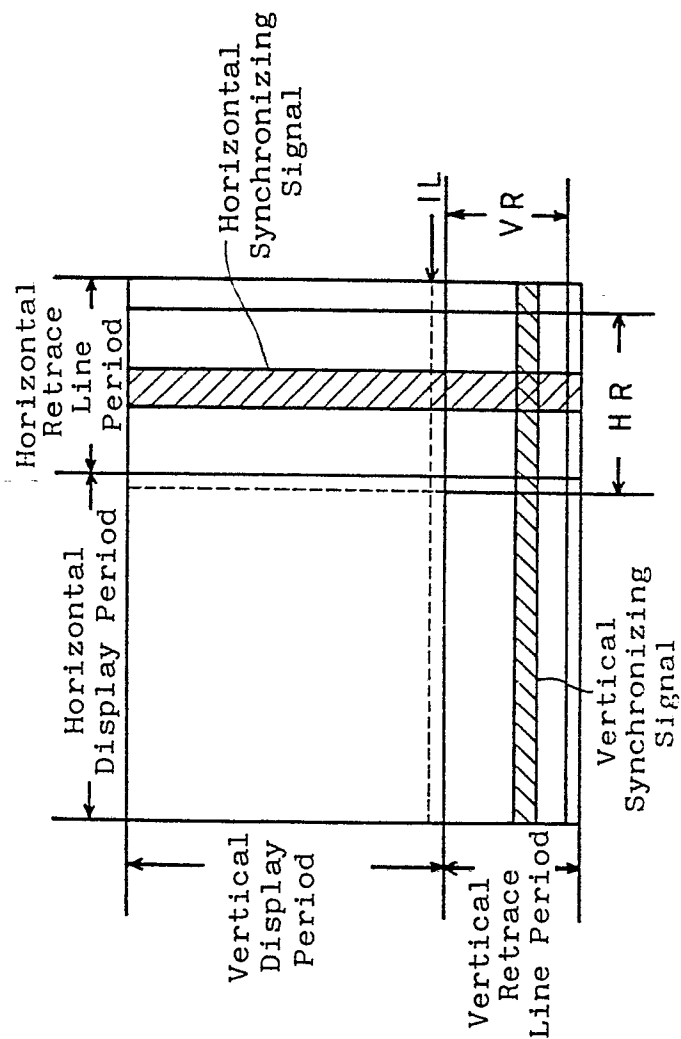


FIG. 10

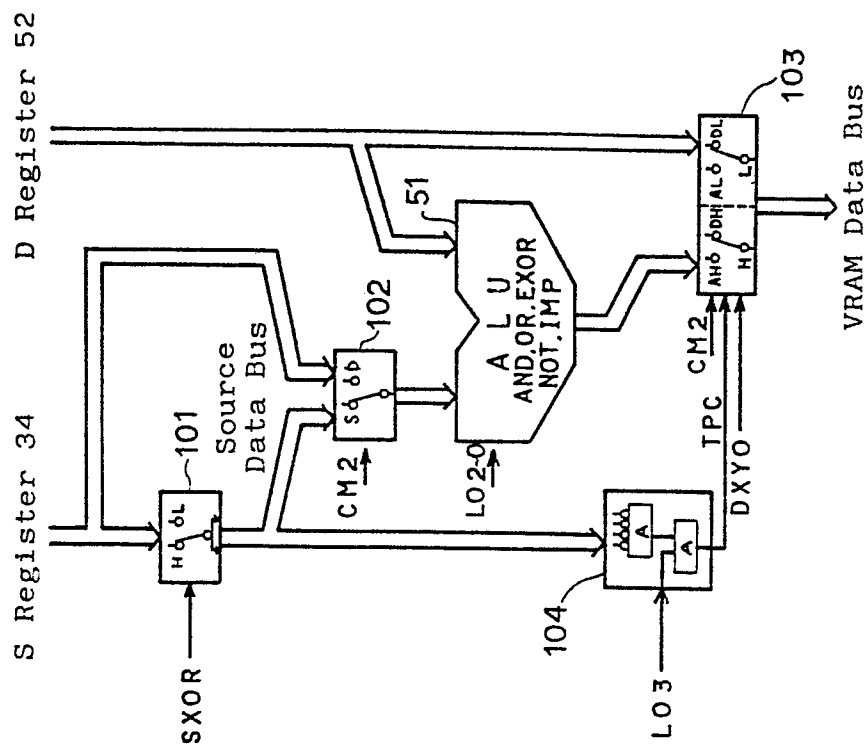


FIG. 11

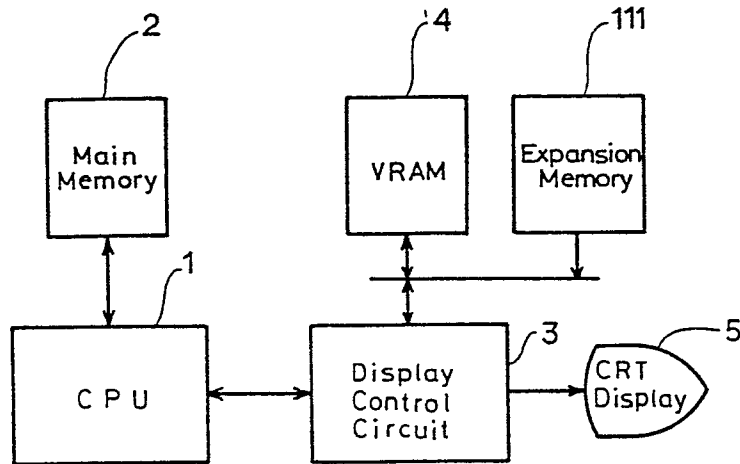


FIG. 12

