

(19)



Europäisches Patentamt

European Patent Office

Office européen des brevets

(11) Publication number:

0 175 499
A2

(12)

EUROPEAN PATENT APPLICATION(21) Application number: **85305996.2**(51) Int. Cl.⁴: **G 09 G 3/36**(22) Date of filing: **22.08.85**

(30) Priority: **22.08.84 US 643132**
22.08.84 US 643133
22.08.84 US 643134
22.08.84 US 643135

(71) Applicant: **DATA GENERAL CORPORATION,**
4400 Computer Drive, Westboro
Massachusetts 01580 (US)

(43) Date of publication of application: **26.03.86**
Bulletin 55/13

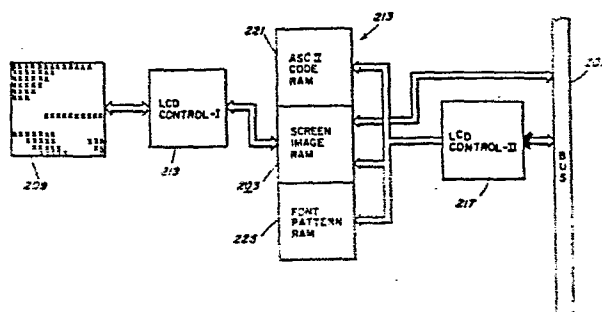
(72) Inventor: **Miyashita, Kazuhiro, 8-17 Fukami Cho Gyodo**
City, Saitama Pref. 361 (JP)
 Inventor: **Katayama, Hideji, 8-4 Ichiryama Cho Gyodo**
City, Saitama Pref. 361 (JP)
 Inventor: **Iijima, Etsuo, 2165 Mochida Gyoda, Saitama**
Pref. (JP)

(84) Designated Contracting States: **DE FR GB NL SE**

(74) Representative: **Pears, David Ashley et al, REDDIE &**
GROSE 16 Theobalds Road, London WC1X 8PL (GB)

(54) **Display system for computers.**

(57) A CPU communicates over a bus (203) with a liquid crystal display (LCD) device (209) by way of a control unit (217), a display memory (213) and another control unit (219). The memory (213) includes a screen image RAM (223) which is a pixel memory cyclically accessed by the latter control unit (219) to keep the display refreshed. The former control unit (217) keeps the contents of the screen image RAM (223) updated in accordance with ASCII codes in a RAM (221), each with an associated attribute code. The control unit (217) converts ASCII character codes into corresponding groups of pixel codes written into the screen image RAM (223) by reference to a font pattern RAM (225). ASCII graphic codes are passed to the screen image RAM (223) without conversion but, in either event, the attribute codes selectively modify what goes into the screen image RAM (223), on a character by character basis. The attribute codes determine whether normal or reverse video is employed, blinking characters are employed, a high speed, imperceptible blinking is employed to provide an intermediate visual tone and to implement a form of colour emulation in which colour codes in the attribute codes select the background tone (and complementary foreground tone) of the display.

**EP 0 175 499 A2**

DISPLAY SYSTEM FOR COMPUTERS

This invention relates to a display system for computers, especially small, portable computers.

In recent years, there has been unprecedented growth in the use of personal computers. When compared to computers of only a decade ago, personal computers are a small fraction of the size and yet have maintained significant processing capabilities.

Many of the personal computers of today available from different manufacturers employ the same or similar central processing units (cpu's) fabricated as one or more integrated circuit chips using large scale integration (LSI) or very large scale integration (VLSI) processes. These cpu's have in some respects become the present standards for the industry. As a natural consequence, one or more sophisticated operating systems has been developed for each brand of cpu and is commercially available to users of the personal computers. When used together the standard cpu's and the operating systems therefor provide sufficient processing speed and flexibility for a large majority of the users of personal computers.

- 2 -

The standardization of cpu's and operating systems has permitted manufacturers of personal computers to focus their attention on other features of the computers in order to differentiate among the available computers and to establish favourable marketing shares. Manufacturers have developed specialized data entry and editing devices, peripheral devices, colour graphics capabilities, and sophisticated application software programs.

With few exceptions, however, personal computers all use cathode ray tube (CRT) displays either through special CRT displays or by connection to a standard television receiver. A CRT is capable of good resolution, colour display, and can display many characters on the screen.

The use of a CRT display, however, limits the extent to which a personal computer can be reduced in size and, since most CRT's are quite large, prevents the computer and display from being truly portable. Some manufacturers have substituted other types of displays including liquid crystal display (LCD) devices. An LCD is much more compact than a CRT and, therefore, contributes to the portability of the personal computers.

Nonetheless, there are several drawbacks related to use of LCD devices. One is that the number of characters displayable on an LCD screen is normally much smaller than the number which

- 3 -

can be displayed on a CRT display. Also, the character cells in LCD devices are typically square whereas character cells for a CRT can be square or rectangular with the longer dimension in either width or height. An LCD does not have the capability to display colour as is possible with a CRT display.

These differences in the operating characteristics of a CRT display and an LCD device have significant ramifications. For example, even if a personal computer with a CRT display is configured identically to a personal computer with an LCD device, i.e. the same cpu, the same operating system, and the same peripherals, the latter will not be able to run application programs written for the former when, as in almost all cases, the program requires the use of a display. This is a serious problem because it either requires modification^a of application software programs or the independent development of programs for use by computers with LCD devices. The lack of compatability between computers with an LCD device and computers with the more popular CRT display places a manufacturer of the former in an unfavourable market position.

Thus, the prior art does not provide a small, easily portable computer with an LCD device which may be used without revising application software programs written for computers with CRT displays.

- 4 -

Thus the objects of the present invention are to make it possible to provide a small, easily portable computer with a compact display, in particular a liquid crystal display which is compatible with software programs prepared for computers with cathode ray tube displays.

Another object of the present invention is to provide a portable computer with a liquid crystal display that displays as many characters as the standard cathode ray tube display.

An additional object of the present invention is to control a liquid crystal display to display half tone images.

The invention in a number of different aspects is defined in the appended claims.

The invention will be described in more detail, by way of example and with reference to the accompanying drawings, in which:

Figures 1(a)-1(c) are views of a portable computer using a display and display control circuitry including the present invention;

Figures 2(a)-2(b) are block diagrams of a computer and a display subsystem, respectively, utilising the present invention;

Figures 3(a)-3(e) schematically illustrate the relationship between the display and display memory within a system utilizing the present invention;

Figure 4 is a detailed block diagram of control circuitry for transferring screen image data to an LCD in a computer system utilising the present invention;

Figure 5 is a high level block diagram of control circuitry for converting ASCII-coded data to screen image data in a computer system utilizing the present invention; and

Figures 6-24 are detailed circuit diagrams illustrating an embodiment of the control circuitry of Figure 5.

Figure 1(a) is a perspective illustration of a computer utilizing the present invention. The computer includes a base portion 11 and a display portion 13. The display portion 13 is hinged to the base portion 11 and is shown in the closed, i.e., folded, position. The compact nature of the computer and display testifies as to its easy portability.

The computer is shown in Figure 1(b) with the display portion 13 in the operative position. An LCD device 15 is included within the display portion 13. The LCD device 15 displays as many characters as a conventional CRT, for example 25 lines of 80 characters each, but can easily be seen to be much smaller than the CRT display alone of most prior art computer. A keyboard 17 is provided beneath and in front of the LCD device 15. The display portion 13 when closed is a protective cover over the keyboard 17.

Figure 1(c) is a side view of the computer with the display portion 13 in the operative position. A support member 19 supports the base portion 11 in such a manner that the keyboard 17 is positioned for optimum operation. Two floppy disc drives 21 are provided within the base portion 11 to provide interchangeable storage capability for the computer.

Figure 2(a) is block diagram illustrating the computer system embodying the present invention. The computer system includes a processor 201 which is connected to an internal bus 203 for the bidirectional transfer of data and control signals therebetween. A suitable processor is the Model 80C88 microprocessor commercially available from Intel. This microprocessor includes both a random access memory (RAM) and a read-only memory (ROM) which are used in the operation of the microprocessor, and may be used in conjunction with the MSDOS and the CP/M-86 operating systems.

Included with the peripherals for the computer system are the keyboard and the disk storage subsystem both of which are individually coupled to the internal bus. A suitable disk subsystem includes one or more compact floppy disk drives 21 as shown in Figure 1(c).

A liquid crystal display (LCD) 209 is connected to display control circuitry 211. The display control circuitry 211 transfers data to be displayed and control signals to the LCD 209. The LCD 209, in turn, sends status signals to the display control 211 along with signals identifying the format of the display.

The display control 211 is connected to the internal bus 203 for bidirectional communication therebetween. A display memory 213 is also connected to the internal bus 203 and to the display control circuitry 211. As will be evident from the description which follows, data to be displayed on the LCD 209 can be stored in and retrieved from the display memory 213 by the display control circuitry 211. The processor 201 can also transfer data to the display memory 213 directly over the internal bus 203.

Figure 2(b) is a more detailed block diagram of the display control circuitry 211 and the display 213. As embodied herein, the display control circuitry 211 includes LCD control-1 219 which transfers data to be displayed from the display memory 213 to the LCD 209. Status signals generated by the LCD 209 are transferred from the LCD 209 to the LCD control-1 219.

LCD-control-2 217 is connected between the internal bus 203 and the display memory 213 and controls the storing of ASCII-

coded display data in the display memory 213 as well as the conversion of the ASCII-coded display data to screen image data suitable for display on the LCD 209.

The display memory 213 comprises a RAM divided into three sections, an ASCII code RAM 221 for storing data to be displayed on the LCD, a screen image RAM 223 for storing all or part of the data also stored in the ASCII code RAM 221 but in a format suitable for display on the LCD 209, and a font pattern RAM 225 for storing conversion data used in converting ASCII-coded data into screen image data.

As will be described later, LCD-control-2 217 includes many internal registers which are accessible by the processor 201 and are used to define and control the LCD 209. One of these registers is an index register (not shown) which is used as a pointer to the memory locations storing the locations of the other registers. The index register is a register which is loaded by the processor 201 by executing an OUT instruction. In order to load any of the other control registers, the index register is first loaded with the appropriate register address, a data register (not shown) is loaded with information to be stored in the selected control register, and an OUT instruction is executed by the processor 201.

The following table identifies the control registers and the values which are stored therein to implement and control different operation modes of the LCD 209.

Name	R/W	Description
R0		*
R1	W	Horizontal displayed (characters)
R2		*
R3		*
R4		*
R5		*
R6	W	Vertical displayed (characters)
R7		*
R8		*
R9	W	Maximum scan line address (scan line)
R10	W	Cursor start scan line address (scan line)
R11	W	Cursor end scan line address (scan line)
R12	R/W	Start address (high)
R13	R/W	Start address (low)
R14	R/W	Cursor address (high)
R15	R/W	Cursor address (low)
R16		*
R17		*
R18	W	Operation mode

R19	W	Scan interval select
R20	W	Blink interval select
R21	W	Underline position (scan line)
R22	W	Font select
R23	W	Background (BG) color table
R24	W	Image memory start address (high)
R25	W	Image memory start address (low)
R26	W	Video RAM address mask
R27	W	Test mode
R28	W	Test status 1
R29	W	Test status 2
R30	W	Data loop back (high)
R31	W	Data loop back (low)

* Not currently utilized.

The function and interpretation of signal values stored within the control registers is explained next.

R1: Horizontal displayed

Bits 7-0 Total horizontal displayed characters. Range is from 2 to OFFH. If the stored value does not match the real screen size (40 at low resolution mode, 80 at high resolution mode), then ERROR.

R6: Vertical displayed

Bits 7-0 Total vertical displayed characters. Range is from 2 to OFFH. If the stored value does not match the proper screen size (25 rows), then ERROR.

R9: Maximum scan line address

Bits 7-4 Ignored

Bits 3-0 Stores a value corresponding to one less than the number of scan lines in a character. Range is from 0 to OFH.

R10: Cursor start scan line

Bits 6-5 Controls the enabling or disabling of the display cursor.

Bit 6	Bit 5	
-----	-----	
0	0	Display the cursor
0	1	Non-display the cursor
1	0	Display the cursor
1	1	Non-display the cursor

Bit 4 Ignored

Bits 3-0 Start scan line of the cursor. Range is from 0 to OFFH. When the stored value is greater than bits 3-0 of R9 (character scan line size), the cursor may not be displayed.

R11: Cursor end scan line (write only)

Bits 7-4 Ignored

Bits 3-0 End scan address of the cursor. Range is from 0 to OFFH. When the stored value is less than bits 3-0 of R10 (cursor start scan address), the cursor may not be displayed.

R12: Start address of character/image buffer high (read/write)

Bits 7-6 Ignored at write, return zeros at read.

Bits 5-0 Most significant 6 bits of relative start address of character/image buffer

R13: Start address of character/image buffer low (read/write)

Bits 7-0 Least significant 8 bits of relative start address of character/image buffer. Relative start address of character/image buffer register is 14 bits in width, therefore, 16K bytes of character/image buffer area are accessible.

R14: Cursor address high (read/write)

Bits 7-6 Ignored at write, return zeros at read.

Bits 5-0 Most significant 6 bits of relative address of the cursor.

R15: Cursor address low (read/write)

Bits 7-0 Least significant 8 bits of relative address of the cursor. Relative address of the cursor register is 14 bits in width. Therefore, 16K bytes of character/image buffer area are accessible. When this cursor address is off screen, the cursor may not be displayed.

R18: Operation mode (write only)

- Bit 7 Controls the recognition of character attributes in character mode. When it is 0, black and white mode is selected. When it is 1, color (emulation) mode is selected.
- Bit 6 Controls the enabling or disabling of the color/image buffer scanning facility. When it is 0, scan function is disabled. When it is 1, scan function is enabled. When main processor is changing the mode registers of the LCD-control-2, the scan function may be disabled to inhibit the destruction of data stored in the screen image RAM 223, i.e., the video buffer.
- Bit 5 When this bit is 0, maximum scan line address in R9 is programmable. When this bit is 1, maximum scan line address is not programmable ("OUT" instruction of R9 is ignored) and is set equal to

7. This bit may be set when selecting monochrome display mode.

Bits 4-3 Ignored

Bits 2-0 Mask bits of scan address of code/image buffer.

Code/image buffer address equals this mask value .AND. scan address register (14 bits).

This facility enables changing code/image buffer address structure.

R19: Scan interval select

Bits 7-4 Ignored

Bits 3-0 Wait time between the scan of each plane of the display 209. If it is 0, then LCD-control-1 219 does not wait between each plane scan.

R20: Blink interval select

Bit 7 Selects base clock for pixel blinking. When it is 0, character blink timing is synchronized with LCD-control-2 217 scan frame clock (not shown). When it is 1, character blink timing is synchronized with character/image buffer scan clock.

Bit 6-4 Ignored

Bits 3-2 Controls the high speed blinking cycle. Blink cycle is specified as follows:

Bit 3	Bit 2	Divisor
0	0	1/2
0	1	1/4
1	0	1/8
1	1	1/16

Bits 1-0 Controls the low speed blinking cycle. Blink cycle is specified as follows:

Bit 1	Bit 0	Divisor
0	0	1/32
0	1	1/64
1	0	1/128
1	1	1/256

R 21: Underline position and over-scan write protection

Bit 7 Ignored

Bits 6-4 Limit address of image memory scanning. By this function, the screen image RAM 209 is protected from being overwritten when mode register is changed.

Bit 3 Raster address of underscore. Range is from 0 to OFH. When the stored value is greater than the maximum scan line address in R9, underscore may disappear.

R22: Font select

Bits 7-4 Address of the font pattern RAM 225. This bit series will be used for bits 12-15 of the address for font RAM 225.

Bit 3 Selects function of highlight mode. When the stored value is 0, bit 11 of address for font addressing is bit 1 of R22 (see below). When the stored value is 1, bit 11 of the address for font addressing is "I" (intensity) bit of character attribute.

Bit 2 Select function of highlight mode. When the stored value is 0, high speed blink is disabled. When the stored value is 1, high speed blink is enabled.

Bit 1 When bit 3 of R22 is 0, bit 11 of the address for font addressing is set to the same value. When bit 3 of R22 is 1, it is ignored.

Bit 0 When the value of the scan line in R23 is equal or less than 7, it is used for bit 10 of the address for font addressing.

R23: Background color table

Bits 7-0 When color mode is selected (bit 7 of R18 is one), background color part of character attributes is decoded by this bit array.

Background attribute			BG color
R	B	G	reference bit
0	0	0	bit 0
0	0	1	bit 1
0	1	0	bit 2
0	1	1	bit 3
1	0	0	bit 4
1	0	1	bit 5
1	1	0	bit 6
1	1	1	bit 7

note: Reference bit = 0 White background
 = 1 Black background (reverse
 video)

R24: Start address of image buffer high (write)

Bit 7 Ignored

Bits 6-0 Most significant bits of start address of image
 buffer.

R25: Start address of image buffer low (write)

Bits 7-0 Least significant bits of start address of image
 buffer.

R26: Display memory 213 address mask (write)

- Bit 7 Enables or disables read/write accessing of display memory 213.
- Bit 6 Ignored
- Bits 5-3 Bits 15-13 of display memory address from processor 201 to display memory through the RAM select #1 is masked by this bit array.
- Bits 2-0 Bits 15-13 of display memory address from processor 201 to display memory through the RAM select #2 is masked by this bit array. By this feature, configuration of memory address can be increased.

R27: Test mode (write)

- Bit 7 Indicates test mode. When the stored value is 0, normal mode is selected. When the stored value is 1, test mode is selected. When in test mode, memory accessing is enabled only by scan control section and memory addressing from other sections is ignored.
- Bits 6-3 Ignored
- Bit 2 Controls read cycle time of video RAM. When it is 0, read memory cycle time is 5 machine clock cycles. When it is 1, read memory cycle time is 4 machine clock cycles.
- Bit 0 Controls write cycle time of display memory 213. When it is 1, write memory cycle time is 5 machine

clock cycles. When it is 1, write memory cycle time is 4 machine clock cycles.

R28: Test status - 1 (read)

Bits 7-0 These bits are used for diagnostic purposes.

R29: Test status - 2 (read)

Bits 7-0 These bits are used for diagnostic purposes.

R30: Data loop back high (read)

Bits 7-0 These bits are used for diagnostic purposes.

R31: Data loop back low (read)

Bits 7-0 These bits are used for diagnostic purposes.

Mode control register

This is a 6 bit register with the I/O address 3D8H. This register controls status of the display control circuitry 211 as follows:

Bits 7-6 Ignored

Bit 5 When the stored value is 1, this bit will change

the character background intensity to the blinking attribute function for alphanumeric modes. When the high order attribute bit is not selected, 16 background colors (or intensified colors) are available. For normal operation, this bit is set to 1 to allow the blinking function.

- Bit 4 When the stored value is 1, high-resolution mode (640 by 200) is selected for black and white graphic mode. One color of 8 can be selected (for emulation) on direct drive sets in this mode by using the mono mode register or graphic mode register.
- Bit 3 When the stored value is 1, the video signals at times of mode changes are enabled.
- Bit 2 When the stored value is 0, the color mode is selected. When the stored value is 1, black and white mode is selected.
- Bit 1 When the stored value is 0, 320 by 200 graphic mode is selected. When the stored value is 1, alphanumeric mode is selected.
- Bit 0 When the stored value is 0, 40 character by 25 line alphanumeric mode (low resolution) is selected. When the stored value is 1, 80 character by 25 line alphanumeric mode (high resolution) is selected.

The following is a listing of the modes selected by this register:

Bits	5	4	3	2	1	0	Selected function
-----	-----	-----	-----	-----	-----	-----	-----
	1	0	1	1	0	0	40 by 25 alphanumeric black & white
	1	0	1	0	0	0	40 by 25 alphanumeric color (emulation)
	1	0	1	1	0	1	80 by 25 alphanumeric black and white
	1	0	0	0	0	1	80 by 25 alphanumeric color (emulation)
	x	0	1	1	1	0	320 by 200 black and white graphics
	x	0	1	0	1	0	320 by 200 color graphics (emulation)
	x	1	1	1	1	0	600 by 200 black and white graphics

Figure 3(a) is a schematic illustration of the LCD 209 which is comprised of an array of pixels 301. In the preferred embodiment, the LCD 209 has a resolution of 640 pixels in the horizontal direction and 256 pixels in the vertical direction. The LCD 209 has a capacity to display 25 rows of characters with 80 characters in each row. This is a common configuration for a CRT display.

The pixels 301 are grouped into respective character cells 303 which, in the preferred embodiment, include an array of pixels 301 8 wide by 10 high. The normal character cell for an LCD device is a square array of pixels such as an 8 by 8 array shown in cell 305. The use of a character cell greater in height than in width

contributes to the readability of an LCD device and is the reason that a rectangularly shaped cell was selected for the preferred embodiment. As will be explained below, however, the size of a character cell is programmable for flexibility in application. The intercharacter and interline spacing in the preferred embodiment is one pixel each.

In the preferred embodiment, the display memory 213 has a capacity of 48K bytes divided in segments of 16K bytes for the ASCII code RAM 221, 24K Bytes for the screen image RAM 223, and 8K Bytes for the font pattern RAM 225. The display control 211 has four major control modes including: character mode in black and white (B & W), character mode in color simulation, graphic mode in color simulation, and direct bit map mode.

In any of the control modes, read or write operations of the display data by the processor 201 are done through the LCD-control-2 217. If the character display mode has been selected, the LCD-control-2 217 scans the ASCII code RAM 221 at every idle cycle and converts ASCII-coded character data, in accordance with associated attribute bits, into screen image data by using the font data stored in the font pattern RAM 225. The converted display data is stored in the screen image RAM 323. If the graphic mode has been selected, the LCD-control-2 217 converts ASCII-coded graphic data stored in the ASCII code RAM 221 into appropriate

pixel image data and writes the pixel image data into the screen image RAM 223.

The LCD-control-1 219 scans the pixel image data stored in the screen image RAM 223 and transfers the image data to the LCD 209 for display in accordance with LCD scan timing. The font pattern RAM 225 is accessible by the processor 201 during idle time.

The processor 201 sends ASCII-coded display data to the LCD-control-2 217 in the form of two bytes, a code (or data) byte and an attribute byte. In a direct bit map mode, the processor 201 sends display data directly to the screen image RAM 223.

When any control mode except the direct bit map mode is selected, and if the mode register indicates the B & W mode, the LCD-control-2 217 converts the ASCII-coded data into pixel data by accessing the font pattern RAM 225 and performing logical operations in accordance with the value of the attribute byte. If the mode register indicates color simulation, the operations performed are almost identical to the B & W mode but processing of the attribute byte is different due to access of the color table register for converting the data associated with the color attribute to a selected pattern of B & W pixels. If graphic mode in color simulation is selected, the data stored in the ASCII-

code RAM 221 is transferred to appropriate locations in the screen image RAM 223 without accesses of the font pattern RAM 225.

If reverse video characters are to be stored in the screen image RAM 223, the LCD Control-2 217 changes the background color into black and the color of the character stroke into white. In the event characters with attributes designating blink are to be stored in the screen image RAM 223, the LCD-control-2 217 will cause selected portions of the display 209 to display character data alternately with an all white character cell, or an all black character cell in the event the character is to be displayed with reverse video.

The LCD-control-2 217 also implements the display of highlighted characters and half-tone images. In the case of highlighted characters, the LCD-control-2 217 accesses a second set of font data stored in the font pattern RAM 215, e.g., a bold font, during the conversion of ASCII-coded data to screen image data. Half-toning is accomplished by alternating the display of a selected character with all white pixels to provide a visual half-tone image.

In the character display mode, character data includes 2 bytes consisting of a character code byte and an attribute byte.

The following shows the definition of the 2 bytes during operation in the B & W control mode.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								BL	R	G	B	I	R	G	B
- - - - CHARACTER CODE - - - -								Background				Foreground			
								- - CHARACTER ATTRIBUTE - -							

BL: Blink attribute causing the alternate display of a character cell and a cell filled by unique data, e.g., all white pixels, defined by background.

I : Intensity attribute to display character with either high-light by selecting an alternate font or dimmer by blinking at a higher rate than the response time of the LCD 209.

Background and foreground attribute portions are coded as follows:

Background			Foreground			
R	G	B	R	G	B	
0	0	0	0	0	0	Non-display, fill by white pixels
0	0	0	0	0	1	Underline
0	0	0	1	1	1	Black character/white background
1	1	1	0	0	0	White character/black background
1	1	1	1	1	1	Non-display, fill by black pixels

The following shows the definition of 2 bytes of character data in the color simulation mode. Because the LCD 209 cannot represent multi-colors naturally, the color attribute will project to specific black and white combinations with programmable threshold.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								BL	R	G	B	I	R	G	B
- - - - CHARACTER CODE - - - -								Background Foreground							
								- - - CHARACTER ATTRIBUTE - - -							

BL: Blink attribute to alternate normal display character cell and a cell filled by unique data, e.g., all white pixels, defined by background.

I: Intensity attribute to display character with either highlight by selecting an alternate font or dimmer by blinking at a higher rate than the response time of the LCD 209.

Background: 3 bit color attribute is used as bit address to access the color table register.

R	G	B	Bit Address in Color Table Register
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

If accessed bit in the color table register is 0 this character cell will be displayed as a black character on a white background, and if it is 1 then the cell will be displayed as a white character on a black background. The correspondence between color attribute and the cell display combination can be programmed by loading appropriate values into the color table register. The foreground bits have no meaning.

Font data stored in the font pattern RAM 225 is always treated as a cell, and a display cell size is programmable with the exception that the cell width is fixed as 8 pixels, but may be elongated to 16 pixels (low resolution mode) with 8 pixel resolution by hardware. Character cell height is programmable from 1 to F pixels.

A character cell is defined as total space including character body, inter-character space and inter-line space. In the font pattern RAM 225, an 8 pixel row in a cell is treated as a byte of display data and the MSB in a byte of display data represents the left most pixel on the screen and the LSB the right most pixel.

The 8K bytes of the font pattern RAM 225 can be managed as four 2048 bytes segments corresponding to 256 character patterns with widths of 8 pixels and heights of 8 pixels to realize multiple font selection. The usage of a character cell with height greater than 8 pixels will enable up to two different fonts to be stored in two 4096 byte segments.

The actual byte address in hexadecimal for a particular row in the display cells is generated as follows:

$$\begin{aligned} \text{Address} = & \text{C6000H} + \{\text{FS } 1 * 1000\text{H}\} + \{\text{FSO} * 800\text{H}\} + \\ & \{\text{ROW2} * 400\text{H}\} + \{\text{ROW1} * 200\text{H}\} + \{\text{CD} * 2\} + \text{ROWO} \end{aligned}$$

FS1: It is the high order bit of the font pattern RAM segment select bits. If highlighted font is enabled, this bit is altered by the intensity bit.

FS0: It is the low order bit of the font pattern RAM segment select bits. If a character height greater than 8 pixels is selected, this bit is altered by ROW3.

ROW3-ROW0: These 4 bits represent a particular row in the character cell. If the character height is less than or equal to 8 pixels, then ROW3 does not have meaning.

CD: It is an 8 bit character code enabling the definition of up to 256 distinct characters, including symbols.

In the graphic display mode, graphic data to be displayed is treated as a byte corresponding to 8 pixels, and is directly transferred to storage locations within the 16K bytes of the screen image RAM 225. The screen image RAM 225 is separated into two segments of 8K bytes each corresponding to an odd row image memory and an even row image memory. The even row image memory stores 80 bytes by 100 rows (row0, row2, row198) of image data, and the odd row image memory stores 80 bytes by 100 rows (row1, row3, ... row199) of image data.

The actual byte address in hexadecimal for a particular group of 8 horizontal pixels on the screen is generated as follows:

$$\text{Address} = \text{MOD}(\text{ROW}/2) * \text{BA000H} + \text{MOD}(\text{ROW}+1)/2 * \text{B8000} + \\ \text{INT}(\text{ROW}/2) * 50\text{H} + \text{COL}$$

MOD: It represents a modulo function

INT: It represents an integer function

ROW: It is the vertical location count of rows on the screen assuming that the topmost row on the screen is row 0

COL: It is horizontal location count in 8 pixel units assuming that the leftmost group of 8 pixels is 0.

Figure 3(b) through 3(d) schematically illustrate a further feature of the present invention wherein the LCD 209 can comprise 1, 2, or 4 segments or planes. If the LCD 209 comprises a single plane, data is transferred to and displayed thereon character by character and line by line. If the LCD 209 is divided into multiple planes, however, a number of characters equal to the number of planes can be transferred in parallel from the screen image RAM 223 to the LCD 209.

As shown in Figure 3(b) a single plane LCD includes L lines (e.g., 25) with each line including N words (e.g., 80). Each word corresponds to a character to be displayed. Assuming that the first character to be displayed is stored in memory address A

in the screen image RAM 323, then $A+N-1$ is the address of the last character of the first line, $A+(L-1)N$ is the address of the first character of the last line and the last character in the last line is stored at address $A+LN-1$. Thus, the value L is the line number and the value N is the line offset meaning the character position within a line.

Figure 3(c) illustrates an embodiment wherein the LCD 209 is divided into plane-A and plane-B. Lines 1 - L are included in plane-A and lines $L+1 - 2L$ are included in plane-B. Data word 1 in line 1 is stored at plane-offset address A , e.g., 0, and data word 1 in line $L+1$ is stored at plane-offset address B , e.g., $B=A+LN$.

Figure 3(d) illustrates an embodiment wherein the LCD 209 comprises four planes, i.e., plane-A, plane-B, plane-C, and plane-D. In this embodiment, the line offset maximum, N , takes a value of one-half of that with respect to the embodiments of Figures 3(b) and 3(c). The addresses of the first data word in the first line of plane-A, plane-B, plane-C, and plane-D are A , $B=(L+1)2N$, $C=N$, and $D=(L+1)2N+N$, respectively.

Figure 3(e) schematically illustrates the correspondence between addresses in the screen image RAM 223 and the LCD 209 in an embodiment wherein the LCD 209 includes two planes, i.e.,

plane-A and plane-B. Assuming that the plane-offset address A is equal to address 0 within the screen image RAM 223, words are transferred from the screen image RAM 223 to plane-A of the LCD 209 beginning at address 0 and concluding at address $L(N-1)$. Within plane-B, the data words stored at addresses $LN - 2LN-1$ are displayed.

If the LCD 209 is comprised of four planes, the plane-offset addresses for planes A, B, C, and D would be 0, N, $(L+1)N$, and $(L+1)N+N$, respectively.

Figure 4 illustrates a block diagram of LCD-control-1 219 wherein display data is transferred from the screen image RAM 323 to the LCD 209. This block diagram illustrates the circuitry for generating the display location in the LCD 209 for the data to be displayed and for transferring display data from the screen image RAM 223 to the LCD 209.

During initiation of the computer system, e.g., at power-up of the processor 201, certain control values are transferred to the registers 403-411. The control values relate to the number of planes within the LCD 209 and to the number of words included in each plane. For example, Figure 4 illustrates an embodiment of the LCD-control-1 219 wherein the LCD 209 includes planes A through D. Thus, reg A 403 stores the above-discussed plane

offset A, which may take on a zero value when the first storage word in the screen image RAM 323 stores the character to be displayed at location 0 of row zero of the LCD 209.

Similarly, reg B 405, reg C 407, and reg D 409 store the values of plane-offset B, plane-offset C, and plane-offset D, respectively. These values correspond to the location on the LCD 209 for the display of the first character in plane B, plane C, and plane D.

Reg E 411 is loaded with a value equal to the number of characters in each row of a plane. Assuming that the LCD 209 displays 80 characters in a row, the value stored in reg E 411 for the embodiments of the LCD 209 shown in Figs. 3(b) and 3(c) will be 80 and the value stored in reg E 411 for the embodiment of Fig. 3(d) will be 40.

The outputs of the registers 403-409 are supplied as inputs to a multiplexer 413. The signals PLNSELO and PLNSEL1 control the multiplexer 413 to selectively output the values stored in the registers 403-409 to one input of an adder 415. The value outputted by the multiplexer 413 comprises a 16 bit base offset address for one of the planes A through D.

The output of the adder 415 comprises a 16 bit address of a character position on the LCD 209 for one of the planes A through D. The first value outputted by the adder 415 for each of the planes A through D corresponds to the first display location on the LCD 209 within each of the planes A through D and will equal the offset values stored in the registers 403 - 409. For each of the other characters in a row, it is necessary to increment the base offset values. The increment is generated by the adder 417, register 419, and the counter 421.

The value stored in the reg E 411 is supplied to one input of the adder 417. The other input to the adder 417 receives the output of the register 419, LINSAO-15. The output of the register 419 is also loaded into the counter 421 under the control of the line load signal, LINELD. The register 419 stores the output of the adder 417 and is loaded with that value under the control of the line end signal LINEEND. In the embodiment described herein, the values stored in the various registers and adders are expressed in twos-complement. At initiation, the values stored in the register 419 and the counter 421 are set to zero. The value stored in the counter 421 is incremented by one under the control of the data ready signal, DATARDY. The generation of DATARDY will be described hereinafter.

A line word register 423 is loaded at initiation with a value equal to one-half of the number of characters displayed in a row of each plane. Thus, in the embodiments of Figs 3(b) and 3(c) the stored value is 40 based upon the assumption of the LCD 209 displaying 80 characters in a row. In the embodiment of Fig 3(d), each row includes two planes and, therefore, the value stored in the line word register 423 is 20. The stored value takes into account the fact that two 8-bit words or characters are transferred at one time from the screen image RAM 223 as the signals VRAMDO-15.

The two's complement of the value stored in the line word register 423 is loaded into a timing counter 425 upon the occurrence of LINELD which is generated when the counter 425 overflows. The timing counter 425 is incremented under the control of the output of an OR gate 429 which receives as inputs DATARDY and the local clock signal, LOCLK. The value stored in the timing counter corresponds to the number of two-word data transfers from the screen image RAM 223 to the LCD 209.

A line number register 429 is loaded at initiation with the number of lines to be displayed within a plane. In the embodiment of Fig. 3(b), this number is equal to the maximum number of lines that can be displayed on the LCD 209. In the embodiments of Figs. 3(c) and 3(d), the value stored in the

register 429 is one-half the maximum number of lines that can be displayed on the LCD 209 because the LCD 209 is vertically divided into two planes. The value stored in the register 429 is expressed in two's complement and is incremented when all data words within a display line of the LCD 209 have been transferred. Upon overflow, the timing counter 431 is reloaded with the value stored in the line number register.

As will be apparent to one skilled in the art, the display parameters, i.e., the number of lines and characters per line, of the LCD 209 are programmable. This provides flexibility to the computer and ultimately to the user.

The registers 441, 443, 445, and 447 are used in the actual transfer of display data from the screen image RAM 323 to the LCD 209. Each of these registers is associated with a different one of the planes of the LCD 209. Thus, data to be displayed in plane-A is received by register A 441, plane-B by register B 443, plane C by register C 445, and plane D by register D 447.

In the embodiment of Fig. 3(c), only two planes are provided in the LCD 209 and, therefore, only register A 441 and register B 443 would be necessary. Similarly, in the single-plane embodiment of Fig. 3(b), only register A 441 would be utilized.

Since the four data transfer paths shown in Figure 4 are identical and operate in parallel only one data path will be described in detail.

Two words of data, VRAMDO-15, from the screen image RAM 223 are loaded into the register 441 and are shifted to a parallel-to-serial converter 448 under the control of the signal PLANASTB. The shifting of the data from the register A 441 to the parallel-to-serial converter 448 generates the signal DATARDYA which is supplied to an input of an AND gate 430.

The parallel-to-serial converter 448 comprises a shift register 449A and a multiplexer 451A. The 16 bits of data, AD0-15, received as inputs by the shift register 449A, are outputted in groups of four bits each, i.e., NAO0-3, NAO4-7, NAO8-11, and NAO12-15, under the control of the selection signals NBLSEL0 and NBLSEL1. The output of the multiplexer 451A is supplied to plane-A of the LCD 209 as the data PLANAD0-3 and to a buffer 453.

As evident from Figure 4, the registers 443, 445, and 447 generate respective data ready signals DATARDYB, DATARDYC, and DATARDYD. These signals are also supplied as inputs to the AND gate 430 with the result that the signal DATARDY will be generated with a high value capable of incrementing the timing

counter 425 after the display data has been shifted from each of the registers 441-447. In the case of single or dual plane embodiments of the LCD 209 the appropriate values for the inputs signals to.

In operation, the initial values corresponding to the embodiment of the LCD 209 are loaded into the registers 403, 405, 407, 409, 411, 423, and 429, as described above. The values reflect not only the number of planes included in the LCD 209 but also the number of characters in a row and the number of lines in the LCD 209. The adder 415 outputs the display addresses within each plane of the LCD 209 and the multiplexers 451A-451D begin to transfer screen image data to the LCD 209.

Assuming the four plane embodiment of the LCD 209 as shown in Figure 3(d), the data words are transferred to the LCD 209 by the four multiplexers 451A, 451B, 451C, and 451D. Each time two data words, that is 16 data bits, are transferred from the registers 441, 443, 445, and 447 to the shift registers 449A, 449B, 449C, and 449D, respectively, the signal DATARDY is generated with a high value to increment the timing counters 425 and 421. After 20 two word data transfers have been made to the respective shift registers, the timing counters 421 and 425 will overflow. This results in the generation of the signal LINELD which resets the counter 425 to the value stored in the line word

- 39 -

register 423, increments the timing counter 431, sets the counter 421 to the value stored in the register 419, and loads the register 419 with the current output of the adder 417.

As a result, the counter 421 outputs the incremental address which, when added to the plane base address stored in the registers 403-409, generates the screen addresses of the LCD 209 in the second row of each display plane. Subsequent operations transfer the data words to be displayed in the second line of each plane at the screen locations outputted by the adder 415. After the data comprising the second line in each plane of the LCD 209 have been displayed, the third and subsequent lines are displayed. When the last line in each plane has been displayed the timing counter 431 overflows and is reset to the value stored in the line number register 429. This causes the register 419 to be reset to zero. The data in the screen image RAM 223 is again transferred to the LCD 209 to refresh the display and to change the displayed data to reflect changes in the data stored in the screen image RAM 223.

The address provided by the adder 415 on the liquid crystal device address bus LCDA0-15 is also output to the multiplexed video RAM bus VRAMMO-15 via a multiplexer 463 and a buffer 465. Data from the video RAM is returned to a buffer 467 driving a data bus VRAMDO-15 which provides the data for the registers 441, 443, 445 and 447 described above. The CPU supplies the data to the registers 403, 405, 407, 409, 411 via the system bus 203 and a buffer 401 which can also output data and address to the video RAM bus via a register 461 and the multiplexer 463.

Figure 5 is a high level block diagram of the LCD-control-2 217. The LCD-control-2 217 includes the status and control registers previously described, address and data paths to and from the CPU 201, the ASCII code RAM 221, the screen image RAM

223, the font pattern RAM 225, and circuitry for modifying data in accordance with associated attributes.

The status and control section 501 includes the registers R1, R6, R9-R15, and R18-R27. The functions of these registers have been described above, and the embodiment thereof will be discussed hereinafter.

An 8 bit data bus from the CPU 201 provides the signals DBO-7. The data bus to the three RAMs 221, 223, and 225 within the memory 213 is indicated by the signals MBO-15. A memory^{address}/bus shared by the CPU 201, the memory 213, and the LCD-control-2 217 is indicated by the signals AO-15.

Upon initialization, select register logic 503 is controlled by five bits of the address signal AO-7 to enable initial values for the status and control values to be loaded into the appropriate control and status registers from the CPU data bus DBO-7. After initialization, when it becomes necessary to change the values in any of the status and control registers, the select register logic 503 is enabled to reload the appropriate register or registers.

Data supplied from the CPU 201 can be directly transferred to the memory 213 through two write data latches 504 and 505. The data latches enable two 8 bit data transfers from the CPU,

i.e., CPUDO-7, to be transferred in parallel on the 16 bit memory bus as memory data MEMBO-15.

Similarly, a 16 bit data transfer from the memory 213, i.e., MEMIO-15, can be outputted to the CPU 201 as two 8 bit data words, CPUDIO-7, through a pair of read data latches 507, 509. The memory data, MEMIO-15, can also be supplied to an ASCII code latch 511 which receives the high order 8 bits, MEMI8-15, and an ASCII attribute latch 513 which receives the low order 8 bits, MEMIO-7. The ASCII latches 511, 513 are employed to receive data from the ASCII code RAM 221.

The contents of the ASCII code latch 511 are transferred to the memory bus as the low order 8 bits, AO-7, of an address in the font pattern RAM 225. The high order 8 bits, A8-15 are supplied by font select logic 515 which receives inputs from the font select register, R22. The 16 bit address is used to access specific memory locations within the font pattern RAM 225 which store screen image data corresponding to ASCII-coded data. The actual bit representation of the ASCII-coded data is used as part of the address within the font pattern RAM 225.

The two words of data returned from the font pattern RAM 225 are received by the font data latches 517, 519. The low order 8 bits are received by the latch 519 which supplies them to the

attribute processing circuitry 521 and to the memory bus as the low order 8-bit word A0-7. The font data latch 517 supplies the high order 8 bits to the attribute processing circuitry 521 and to the memory bus as the high order 8-bit word A8-15. The data bits transferred directly from the font data latches 517, 519 are transferred by the memory bus to the screen image RAM 223.

The attribute processing circuitry 521 receives the data from the font data latches 517, 519 eight bits at a time and modifies the received data under the control of the attribute control circuitry 523. The modified data is supplied to the memory bus through the output regular mode font data latches 525, 527 or through the bold mode font data latches 529, 531. In the bold (low resolution) mode, each displayed character is two character cells wide.

The addresses within the screen image RAM 223 for storing screen image data are generated by a counter 541 which receives the image start address, ISA0-15 from the image start address registers R24, R25. An offset value can be added to the output of the counter 541 by an adder circuit 543. The offset is generated by an adder 545 and is stored in a latch 547 coupled thereto. The output of the adder 543 comprises a 16 bit image plan^e address, IMPA0-15, which is supplied to the memory bus. A pair of data latches 549, 551 is provided to transfer the image

plane address from the memory bus to the CPU 201 as loop back data LBD0-15. A next line address latch 553, 555, is provided to receive an address from the memory bus and to supply the received address to the counter 541 through an adder 557 or to the CPU 201 as loop back data.

Figures 6- 24 comprise detailed logic diagrams of a preferred embodiment of the LCD-control-2 217 as shown in Figure 5 and generally described above. As shown in Figure 6, the memory bus includes a pair of 8 bit latches 601, 603 which receive the memory bus signals AO-15 from the CPU 201 and the memory 213. The signals AO-7 are transferred to a latch 605 and are subsequently designated MEMB0-7. The bits A8-11 are transferred to a latch 607 to become the bits MEMB8-11. The four high order bits, MEMB12-15 outputted by the latch 607 are supplied by the display memory address mask register R26.

The eight bit data bus, D0-7, coupled with the CPU 201, is received by an input latch 609 which outputs the CPU data, CPUD00-07. Data is transferred to the CPU 201 via an output latch 611. Clock signals employed within the LCD-control-2 are generated by an oscillator 613 and outputted by a latch 615.

The select register logic 503 (Figure 5) is embodied by decoders 621, 623 and NAND gates 625-645. The inputs to the

decoders 617-623 include the signals REGSEL0 - REGSEL4 which comprise the CPU data bits CPUD10 - CPUD14 outputted by a latch 647. The latches 621, 623 output enable signals UR18WR-UR27WR for controlling writing of corresponding status and control registers. The decoder 623 also outputs the signals UR28RD-UR31RD which are read enable signals.

The NAND gates 625-629 and 639-645 output LCD control register write enable signals UR9WR-UR25WR, respectively. The NAND gates 631-637 output LCD control register read control signals UR12RD-UR15RD, respectively.

Figure 7 illustrates circuitry 701 for generating CPU access request signals. While this circuitry plays a role in the operation of the LCD-control-2 217, a detailed description thereof will not be set forth because such description of the circuitry 701 is not necessary for an understanding of the present invention.

Figure 7 also illustrates embodiments of the write data latches 504, 505 which receive as inputs CPU data, CPUD00-07, and respectively output to the memory bus MEMB8-15 and MEMB0-7. Similarly, the read data latches 507, 509 are shown as receiving as inputs MEMB8-15 and MEMB0-7, respectively, and generating as outputs CPUDI0-7 which are transmitted to the CPU 201.

Figures 8-10 illustrate embodiments of the status and control registers discussed above. The mode control register is embodied as an edge-triggered flip-flop 801 which selectively outputs the blink enable signal (BLKENB), the video enable signal (VIDENB), the graphic mode signal (GRAPHIC), and the high resolution signal (HIRES) in accordance with the CPU data, CPUD00-07. The horizontal displayed register R1 and the vertical displayed register R6 are similarly embodied as edge-triggered flip-flops 803, 805, respectively, which receive as inputs CPUD00-07.

The register R12 storing the start address of the character/image buffer (high) is embodied as an edge-triggered flip-flop 807 receiving the input signals CPUD00-07 and a transceiver 809 receiving the input signals CPUDI0-7. Data to be written into the register R12 is supplied by the flip-flop 807 whereas a read out of the register R12 is transferred through the transceiver 809.

The register R13, which stores the start address of the character/image buffer (low), is embodied as an edge-triggered flip-flop 811 and a transceiver 813. The data to be written into the register R13 is supplied through the edge-triggered flip-flop 811 and data to be read from the register R13 is transferred through the transceiver 813.

The cursor address (high) register, which is the register R14, is supplied by an edge-triggered flip-flop 815. A transceiver 817 is employed to transfer the values stored in the register R14 to the CPU 201. Similarly, the register R15 storing the cursor address (low) is embodied as an edge-triggered flip-flop 819 and a transceiver 821.

The register R9 storing the maximum scan line address is embodied as an edge-triggered flip-flop 901. The register R10 controlling the cursor start scan line is embodied as an edge-triggered flip-flop 903. An AND gate 905 coupled to the flip flop 903 through an inverter 906 generates the cursor inhibit signal CSRINH. The register R11 controlling the cursor end scan line is embodied as an edge-triggered flip-flop 907. The flip flops 901, 903, 907 all receive the CPU data, CPUD00-07, as inputs.

The operation mode register R18, the scan interval select register R19, and the blink interval select register R20 are embodied as edge-triggered flip-flops 909, 911, and 913, respectively.

The underline position and over-scan protection register R21, the font select register R22, and the background color

register R23 are embodied by the edge-triggered flip-flops 915, 917, and 919, respectively.

Figure 10 illustrates the image buffer start address (high) register R24, the image buffer start address (low) register R25, the display memory mask register R26, and the test mode register R27 as being embodied by edge-triggered flip-flops 1001, 1003, 1005, and 1007. A multiplexer 1009 receives the outputs of the flip-flop 1005 and is coupled to the AND gates 1011, 1013, and 1015 to generate the three high order address mask bits. The test-status-1 register R28, test-status-2 register R29, data loop back (high) register R30, and the data loop back (low) register R31 are embodied as transceivers 1017, 1019, 1021, and 1023, respectively, to selectively output the data CPUDI0-7.

Figures 11 and 12 illustrate circuitry for generating timing and control signals used within the LCD-control-2 217, including timing signals for accessing the memory 213. Also shown in Figure 11 is a priority encoder 1101 including a D flip-flop 1103 which outputs the signals LCDSEL, CPUSEL, and SCNSEL to control access to the LCD 209, CPU 201, and memory 213, respectively. The function of the timing circuitry will not be described in detail because such description is not necessary for an understanding of the present invention.

Figure 13 illustrates circuitry for generating additional timing and control signals. A scan control sequencer 1301 generates timing and control signals for reading the ASCII code RAM 221, accessing the font pattern RAM 225, and writing screen image data into the screen image RAM 223.

Figure 14 illustrates an embodiment of the counters and comparators (shown in Figure 5) connected to the horizontal displayed register R1, the vertical displayed register R6, and the maximum scan line address register R9. As embodied herein, a horizontal character counter 1401 includes a comparator 1403 having one set of inputs HDISP0-7 corresponding to the contents of the register R1 and a second set of inputs generated by counters 1405, 1407. The counters 1405, 1407 are incremented each time an ASCII code character is converted to a screen image character by the LCD-control-2 217. The count stored in the counters 1405, 1407 is equal to the present value stored in register R1. An end of line signal, LINEND, is generated by a D flip-flop 1409.

A vertical line counter 1411 includes a comparator 1413 having one set of inputs, VDISP0-7, supplied by the vertical displayed register R6 and a second set of inputs supplied by the counters 1415, 1417. The counters 1415, 1417 are incremented by the LINEND signal and store an indication of the line number

being stored within the screen image RAM 223. When the line number stored in the counters 1415, 1417 is equal to the preset value in the vertical displayed register R6, a D flip-flop 1419 generates a frame end signal, UFRAMEND.

A character row address counter 1421 comprises a comparator 1423 having one set of inputs receiving the contents of the maximum scan line address register R9 and a second set of inputs receiving the value stored in a counter 1425. The output of the counter 1423 is the maximum row control signal, MAXROW.

Figure 15 illustrates an embodiment of the font data latch 517, 519, the ASCII code latch 511, and the attribute latch 513. The values of the attributes associated with an ASCII code word are outputted by the attribute latch and include the control signals for the blink bit (BLBIT), background red (BGRED), background green (BGGRN), background blue (BGBLU), intensity bit (IBIT), foreground red (FGRED), foreground green (FGGREN), and foreground blue (FGBLU).

The signals BGRED, BGGRN, and BGBLU are employed as the selector signals of a color emulator multiplexer 1501. The inputs to the multiplexer 1501 comprise the outputs of the edge triggered flip-flop 919 which is an embodiment of the background color table register R23. Since the LCD 209 is not capable of

displaying the colors red, green, and blue, the color emulator multiplexer 1501 is provided to select either a light or dark background on the display 209 depending upon the values of BGRED, BGGRN, and BGBLU.

Figure 15 further illustrates an embodiment of a cursor timing circuit 1503 which includes comparators 1505 and 1507 which compare the output, ROW0-3, of the counter 1425 to the counters of the cursor start scan line register R10 and the cursor end scan line register R11, respectively. The inputs of an AND gate 1509 are connected to the outputs of the comparators 1505, 1507. The output, CSRPOS, of the AND gate 1509 controls display of the cursor on the LCD 209.

A black and white mode attribute decoder 1511 generates the control signals no-display white, UNDWHT, which causes all pixels in a character cell to be displayed as white and no-display black, NDBLK, which causes all pixels of a character cell to be displayed as black. The decoder 1511 further generates the reverse video control signal, RVVID, which causes all pixels in an associated character cell to display reverse values.

An underline timing generator 1513 includes a counter 1515 which at one set of inputs receives the outputs of the counter 1425 and at another set of inputs receives the value stored in

the underline position register R21. An AND gate 1517 coupled to the outputs of the counter 1513 and the decoder 1511 generates the underline control signal, LNUNDER.

Figure 16 is an embodiment of the attribute circuit 521 of Figure 5. A pair of transceivers processing 1601, 1603 receive the two data words (16 bits) in parallel from the font data latches 517, 519 and converts them into two serial data words of eight bits each. The eight bits of each data word are then processed in parallel in accordance with the values of the attribute bits to generate an eight bit font data, word, FWRD0-7. Because each bit is identically processed, only the processing of bit 0 will be discussed in detail.

Bit zero is supplied as one input of a two input OR gate 1605. The other input to the OR gate 1605 is the output of an OR gate 1607 having as inputs the signals LNUNDER and NDBLK. If either LNUNDER or NDBLK is set equal to 1 (TRUE) then the output of the OR gate 1605 will be 1 indicating that the pixel in the display 209 corresponding to bit 0 should be displayed as black.

The output of the OR gate 1605 is supplied to one input of an AND gate 1609. The AND gate 1609 implements several functions which are based upon the controlled blinking of pixels of the LCD 209 at selected positions. One function is half-toning which

entails blinking of pixels at a very high rate so that the visual appearance is of a continuous tone between white and black. This function is embodied by a NAND gate 1611 which receives the dim enable signal, DIMEN, and a dim blink clock signal, DIMBLK, of high frequency. The AND gate 1611 functions as a switch to turn on and off the pixel corresponding to bit zero at a high rate. This switching is done through the second input of the AND gate 1609.

A pixel can also be blinked at rate discernible visually. This is accomplished by an AND gate 1613 and a NAND gate 1615. Blinking of the pixel is controlled by the blink bit, BLBIT, which is one input to the NAND gate 1615. The other input of the NAND gate 1615 is the output of the AND gate 1613. The AND gate 1613 outputs the blink frequency signal in accordance with the blink clock signal CHARBLK.

The output of the AND gate 1609 is supplied to one input of an OR gate 1617. The other input to the OR gate 1617 is the output of a cursor blink AND gate 1619. If the cursor position includes the pixel associated with bit 0, the pixel will be blinked at a rate determined by the cursor blink clock signal, CSRBLK. The frequency of the cursor blink clock signal, CSRBLK, is preferably different from, e.g., twice the rate of, the

frequency of the character blink clock signal, CHARBLK, to enable visual discrimination between the two.

The output of the OR gate 1617 is supplied to one input of a NOR gate 1621. The other input to the NOR gate 1621 is the output of an OR gate which receives the color emulator signals BGDARK and RVVID. The OR gate 1613 enables the value of bit 0 to be changed to emulate color display through the selection of the background color, i.e., black or white.

Figure 17 illustrates the circuitry for implementing the high resolution mode of the LCD 209 and also the embodiments of the latches 525-531. In low resolution mode each character includes twice as many character cells as in low resolution mode. This is accomplished by generating data words wherein each data word includes values for controlling four pairs of pixels with the pixels in a pair being identical.

The output of the attribute processing circuitry 521, i.e., the font data signals FWRD0-7 are supplied to the data latches 527, 525 (Figure 17) and outputted as the signals MEMB8-15 and MEMB0-7, respectively, in high resolution mode. In normal (low) resolution mode, bits FWRD4-7 of the font data word are supplied to the latch 529 and bits FWRD0-3 of the font data word are supplied to the latch 531. The data bit FWRD7 is supplied to

inputs 6 and 7 of the latch 529 so that bits MEMB14 and MEMB15 will have equal values. Similarly, inputs 4 and 5 of the latch 529 are set to the value of FWRD6 so that bits MEMB12 and MEMB13 will have equal values. Each of the bits FWRD 0-5 is similarly duplicated in the output bits MEMB0-11 of the latches 529, 531. Selection of the high resolution or the low resolution mode is accomplished by the attribute bit HIRES. Graphic mode font data inputs, FD0-15, are transmitted to the memory bus MEMB 0-15 through the latches 1701, 1703.

Figure 18 illustrates the circuitry for generating addresses in the ASCII code RAM 221 to enable access of ASCII code words stored therein so that conversion to display image data may be accomplished by reference to the font data. The ASCII code RAM start address is supplied by the registers R12, R13 as inputs to code buffer address counters 1801, 1803, 1805, and 1807. The value stored in the counters 1801, 1803, 1805, and 1807 is incremented to output in sequence the addresses of ASCII code words stored in the ASCII code RAM 221.

The comparators 1809, 1811 compare the output of the ASCII code buffer address to the present cursor address and sets the signals CURSOR and UCURSOR accordingly. As previously described, the CURSOR signal is used to control blinking of selected pixels to identify the location of the cursor.

Figure 19 illustrates circuitry embodying the image data address generator for generating addresses within the display image RAM 223 for storing image data produced by converting ASCII coded data through accesses to the font data stored in the font data RAM 225. The next line address latch 553, 555 receives the memory address MEMB0-15 from the memory bus and transfers the address to the latches 549, 551 as loop back data LBD0-15 for test purposes. The outputs of the latches are also supplied to incrementers 1901-1907 where the value of MEMB0-15 is incremented by +1. The incremented address is supplied to transceivers 1909, 1911.

A second pair of transceivers 1913, 1915 receives the image start address from the registers R24, R25. The contents of the transceivers 1909, 1911 or of the transceivers 1913, 1915 are selectively supplied to an image plane base address counter comprising the counters 1917 - 1923. The counters 1917 - 1923 are incremented by +1 every time all of the pixels within a character cell have been converted from ASCII code data to screen image data. Thus, the counters 1917 - 1923 store an address corresponding to the top row of pixels in a character cell. The counters 1917 - 1923 are set to the value stored in the transceiver 1913, 1915 whenever the last character in the last row of the ASCII code RAM 221 has been processed.

Figure 20 illustrates the circuitry for generating addresses within the screen image RAM 223 which correspond to the beginning of a line of characters to be displayed on the LCD 209.

Transceivers 2001, 2003 store values which correspond to the number of characters in a display line of the LCD 209 in high and low resolution modes. This value is preset by the register R1.

If high resolution mode is indicated by HIRES, then the value HDISP0-7, which is stored in the transceiver 2001, is transferred to the row offset latch 2005. If low resolution mode is indicated by the signal UHIRES, then HDISP1-HDISP7, i.e., a value equal to one half of the value of HDISP0-7, which is stored in the transceiver 2003, is supplied to the row offset latch 2005.

The row offset latch comprises adders 2007 - 2011 and latches 2013 - 2017 which generate the image offset signals IMOFF0-11 corresponding to N times the value initially received as inputs by the adders 2007, 2009. The value N is equal to the line number currently being stored in the screen image RAM 223. Thus, IMOFF0-11 is a value which always corresponds to the beginning of a line to be displayed on the LCD 209.

The effective address, i.e., the physical address IMPA0-15, within the screen image RAM 223 for storing the screen image data currently being processed is generated by adders 2019 - 2025.

The adders 2019 -2025 add the line address represented by the signals IMOFF0-11 to the character address IMA0-15.

Figure 21 illustrates circuitry 2101 embodying a scan address control signal generator which generates a signal, UCBRENB, which controls whether an address in the ASCII code RAM 221 or in the screen image RAM 223 is being processed. A font selector circuit 2103 is provided to generate bits MEMB10 and MEMB11 of a memory address MEMB0-15 outputted by a pair of transceivers 2105, 2107. In this manner, one of two sets of font data stored in the font data RAM 225 can be selectively addressed.

Figure 22 illustrates a scan interval selector circuit 2201 which includes a comparator 2203 having a first set of inputs receiving the signals SCNIV0-7 stored in the scan interval register R19 and a second set of inputs receiving the outputs of a pair of counters 2205, 2207. The signal UCHIST is generated from the output of the comparator 203 to initiate a scan of the memory 213 and the conversion of ASCII code data to screen image data. The scan interval is selectable through the counters of the register R19. An advantage can be obtained in reduction of power consumption by initiating a refresh of the memory at the longest acceptable intervals.

A clock divider circuit 2209 generates the clock signals RATE0-7 having different frequencies. This enables a blink interval selector circuit 2211 to generate the previously-discussed blink control signals CHARBLK, CSRBLK, and DIMBLK with different frequencies.

Figure 23 and 24 illustrate an embodiment of the memory 213 and circuitry for memory and input-output decoding. The manner of operation of the memory 213 and the various circuitry shown in Figures 23 and 24 will be apparent to one of ordinary skill in the art from the description of the LCD-control-2 217 set forth herein and, therefore, additional description of Figures 23 and 24 is not necessary.

While the salient features of the invention have been described with reference to the drawings, it should be understood that the described embodiment is susceptible of modification and substitution without departing from the scope of the following claims.

- 59 -
CLAIMS

1. A method of operating a liquid crystal display to provide at least one intermediate tone, characterised in that the display is blinked at a rate too high to be visually perceptible.

2. A control system for a liquid crystal display device (209) capable of presenting first and second visual tones, comprising means (517, 519) for receiving data to be displayed, the data having first and second values for causing the display device to display the first and second tones respectively, and means (525, 527, 223, 219) for supplying signals to the display device in accordance with the data, characterised by switch means (1609) between the receiving means (517, 519) and the signal supplying means (525, 527, 223, 219) for selecting between the said data and the said second value, and means (1611, DIMBLK) for operating the switch means (1609) at a rate sufficient to cause data with the first value to be displayed with the appearance of a constant tone intermediate the first and second tones.

3. A control system according to claim 2, wherein the display device (209) has an array of display pixels which are individually turned on and off to display the first and second tones respectively in response to the first and second data values respectively, characterised in that the switch means comprise an AND gate (1609) having a first input connected to the receiving means (517, 519, 1601, 1603), and an output connected to the supplying means (525, 527, 223, 219), and in that the operating means comprise a source (2209, 2211) of clock pulses (DIMBLK) at the said rate, and a gating circuit (1611) for selectively applying these pulses to a second input of the AND gate (1609).

4. A control system according to claim 3, characterised in that the gating circuit (1609) is a second AND gate having inputs to which are applied the clock pulses (DIMBLK) and a control signal (DIMEN) for enabling and disabling the intermediate tone display.

5. A control system according to claim 2, 3 or 4, characterised by means (511, 513) for receiving character codes accompanied by attribute codes, means (225) for generating the said data from the character codes, and means (523) responsive to the attribute codes to determine on a character by character basis whether a character is to be displayed with the first and second tones or with the first and intermediate tones, and to provide a control signal (DIMEN) accordingly, which control signal determines whether the operating means (1611, DIMBLK) function or not.

6. A system for generating data to be displayed by a display device (209) with an array of rows and columns of display pixels, characterised by the combination of a code memory (221) for storing data words identifying characters to be displayed by the display device (209), a font translation memory (225), addressable with addresses formed from said data words, for outputting pixel data corresponding to the data words, a display image memory (223) for storing the pixel data, means (219) responsive to the stored pixel data for driving the array of display pixels to display the characters, and a control circuit (217) for accessing the data words stored in the code memory (221), for forming the said addresses and accessing the corresponding pixel data in said font translation memory (225), and for storing the output pixel data in the display image memory (223).

7. A system according to claim 6, characterised in that each of the data words stored in the code memory (221) has an associated attribute word stored therewith, by a plurality of registers (R1-R31) for controlling the control circuit in accordance with the values of the attribute words associated with the data words.

8. A system according to claim 7, characterised by a register initiation circuit (503) for selectively supplying control data to the control registers (R1-R31).

9. A system according to claim 7 or 8, characterised by an address generation circuit (1801, 1803, 1805, 1807) for generating addresses

- 61 -

in the code memory (221) of the stored data words and associated attribute words, the control circuit (217) including a code memory read latch (511, 513) for receiving the data words and associated attribute words stored in the code memory (221) at the addresses generated by the address generator.

10. A system according to claim 9, characterised in that the control registers (R1-R31) include a font selected register (R22) for supplying selected bits of an address in the font translation memory (225) and in that the control circuit (217) includes a font address circuit (515) for generating the addresses in the font translation memory by combining the selected bits from the font select register with the data words received by the code memory read latch (511).

11. A system according to claim 9 or 10, characterised by a font data latch (517, 519) for receiving the pixel data output by the font translation memory (225), and an attribute processing path (521) for selectively altering the values of the pixel data in accordance with the attribute words associated with the data words corresponding to the pixel data.

12. A system according to claim 11, characterised in that the attribute processing path (521) includes a first pixel data output latch (525, 527) for outputting characters to be displayed with a first width corresponding to a first number of pixels, and a second pixel data output latch (529, 531) for outputting characters to be displayed with a second width corresponding to a second number of pixels greater than the first number of pixels.

13. A system according to any of claims 6 to 12, characterised in that the font translation memory (225) includes a plurality of sections for outputting different pixel data corresponding to the data words, for displaying characters having different visual appearances.

14. A system according to any of claims 6 to 13, characterised in

- 62 -

that code memory (221), the font translation memory (225) and the display image memory (223) comprise separate segments of a random access memory (213).

15. A system according to claim 9, characterised by an image data path for transferring the data words received by the code memory read latch (511) directly to the display image memory (223).

16. A system for controlling a monochrome display to emulate a colour display, comprising means (517, 519, 513) for receiving display signals including a data portion for display and an associated control portion indicating foreground and background colours with which the data portion is to be displayed on a colour display; a colour emulation circuit (1501) responsive to the control portion of the display signals for translating each combination of foreground and background colours to selected monochromatic display control signals (BGDARK), and a data conversion circuit (1621, 1623) for setting the value of the data portion of the display signals in accordance with the corresponding control signals from the colour emulation circuit (1501).

17. A system according to claim 16, in combination with a monochrome liquid crystal display (209) having an array of display pixels capable of first and second visual states, and wherein the data portion of the display signals has first and second values to cause display pixels to be in the first and second visual states respectively.

18. A system according to claim 17, characterised in that the colour emulation circuit comprises a background colour decoder (1501) for outputting a second state control signal (BGDARK) responsive to a selected combination of the background colours indicated by the control portion of the display signals; and a foreground colour decoder (1511) for outputting a reverse video signal (RVVID) in response to a selected combination of the foreground and background colours indicated by the control portion of the display signals.

19. A system according to claim 18, characterised in that the data conversion circuit comprises a NOR gate circuit (1621) having a first input for receiving the data portion of the display signals, a second input, and an output; and an OR gate circuit (1623) having a first input for receiving the reverse video signal (RVVID), a second input for receiving the second state control signal (BGDARK) and an output connected to the second input of the NOR gate circuit (1621), the NOR gate circuit outputting the data portion having a value to cause display pixels in the display to assume the second visual state in response to the occurrence of the second state control signal (BGDARK) and to assume an inverse value with respect to the value thereof supplied to the first input terminal of the NOR gate circuit in response to the occurrence of the reverse video signal (RVVID).

20. A system according to any of claims 16 to 19, characterised by means (R23, (919)) for selecting between a plurality of different translation modes of the colour emulation circuit (1501).

21. A system for controlling a liquid crystal display device (209) capable of displaying first and second visual tones to emulate the colour display of a cathode ray tube, comprising means for receiving display signals (517, 519, 513) which would control the cathode ray tube to produce a colour display, the display signals including a data portion for display and an associated control portion indicating foreground and background colours with which the associated data portion would be displayed on the cathode ray tube; a colour decoder circuit (1501, 1511) for generating display control signals responsive to the foreground and background colours indicated in the control portion; and a display data translation circuit (1621, 1623) for selectively changing the value of the display portion in accordance with the display control signals, whereby the value of the display portion is arithmetically complemented, is set to a value that controls the display to display the second visual tone, or is not changed whereby the liquid crystal display assumes selected display states in response to the indication of colours to be displayed on the cathode ray tube.

- 64 -

22. A system for controlling a liquid crystal display, comprising a display memory for addressably storing at associated addresses data to be displayed by the display; means for supplying addresses of data in the memory and for accessing the data in accordance with said addresses; means for converting said addresses into corresponding display position signals for the display; and means for transferring to the display said display position signals and the associated display data, whereby the display data is displayed at display positions of the display corresponding to the associated display position signals.

23. A system according to claim 22, characterised in that the display includes a numbered sequence of display pixels arranged in a matrix of rows and columns for displaying the display data and wherein the converting means comprises an offset counter for generating pixel offset values indicating the display pixels in the sequence of display pixels wherein the transferred display data is to be displayed.

24. A system according to claim 23, characterised by a pixel column counter, incremented in response to each transfer of display data, for generating a row overflow signal when the count stored therein is equal to the maximum number of pixels in a row of the matrix and for being reset responsive to the generation of the row overflow signal; and a pixel row counter, incremented in response to the generation of the row overflow signal, for generating a reset signal when the count stored therein is equal to the maximum number of rows of pixels of the matrix and for being reset responsive to the generation of the reset signal.

25. A system according to claim 24, characterised in that the offset counter comprises a pixel counter incremented in response to each transfer of display data to the display; an offset adder incremented by a value equal to the number of pixels in a row of the matrix in response to the row overflow signal; and a register for transferring the contents of the offset adder to the pixel counter in response to the row overflow signal, the pixel counter and offset

adder being reset to zero in response to the reset signal.

26. A system for controlling a liquid crystal display which includes a plurality of display segments comprising a display memory for storing data to be displayed by the display, the memory including a plurality of storage locations for storing the display data with each of the storage locations having an associated address; means for supplying addresses of the storage locations in the memory and for accessing, in response to each supplied address, data to be displayed in each of the display segments; means for converting each of the supplied addresses into a corresponding display position signal within each of the segments of the display; and means for transferring in parallel the accessed data to be displayed in each segment of the display and the display position signal whereby data is simultaneously displayed at selected display positions of each segment of the display responsive to the supplying of a single address in the display memory.

27. A system according to claim 26, characterised in that the display includes four display segments and the transferring means comprises four multiplexers, each of the multiplexers having an output connected to a different one of the display segments and an input connected to the display memory for receiving the display data therefrom.

28. A system according to claim 27, characterised in that the storage locations of the display memory are divided into four memory segments, each of the memory segments being associated with a different one of the multiplexers and display segments.

29. A system according to claim 28, characterised in that the display includes a numbered sequence of display pixels arranged in a matrix of rows and columns and in that each of the segments of the display includes an initial display pixel associated with a segment offset indicating the position of the initial display pixel within the numbered sequence of display pixels and in that the converting means comprises an offset counter for generating in association with

- 66 -

each transfer of display data a pixel offset value indicating the display pixel in the sequence of display pixels wherein the associated transferred display data is to be displayed.

30. A system according to claim 29, characterised by a pixel counter, incremented in response to each transfer of display data, for generating a row overflow signal when the count stored therein is equal to the maximum number of pixels in a row of the matrix and for being reset responsive to the generation of the row overflow signal; and a pixel row counter, incremented in response to the generation of the row overflow signal, for generating a reset signal when the count stored therein is equal to the maximum number of rows of pixels of the matrix and for being reset responsive to the generation of the reset signal.

31. A system according to claim 30, characterised in that the offset counter comprises a pixel counter incremented in response to each transfer of display data to the display; an offset adder incremented by a value equal to the number of pixels in a row of the matrix in response to the row overflow signal; a register for transferring the contents of the offset adder to said pixel counter in response to the row overflow signal, the pixel counter and offset adder being reset to zero in response to the reset signal; and a segment offset adder for adding the contents of the pixel counter and the segment offset associated with a selected one of the display segments to generate the pixel position associated with display data being transferred to the selected segment of the display.

1/55

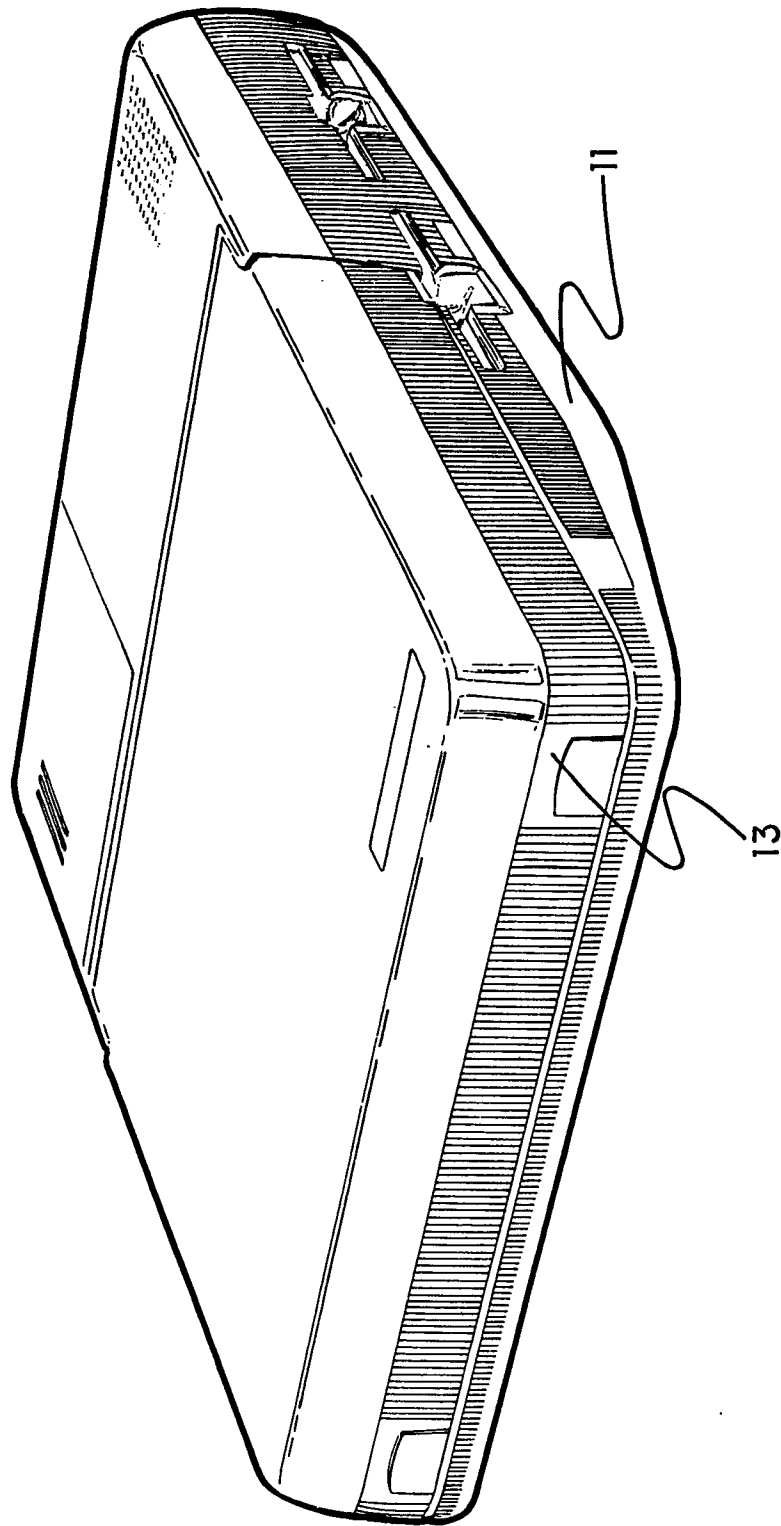
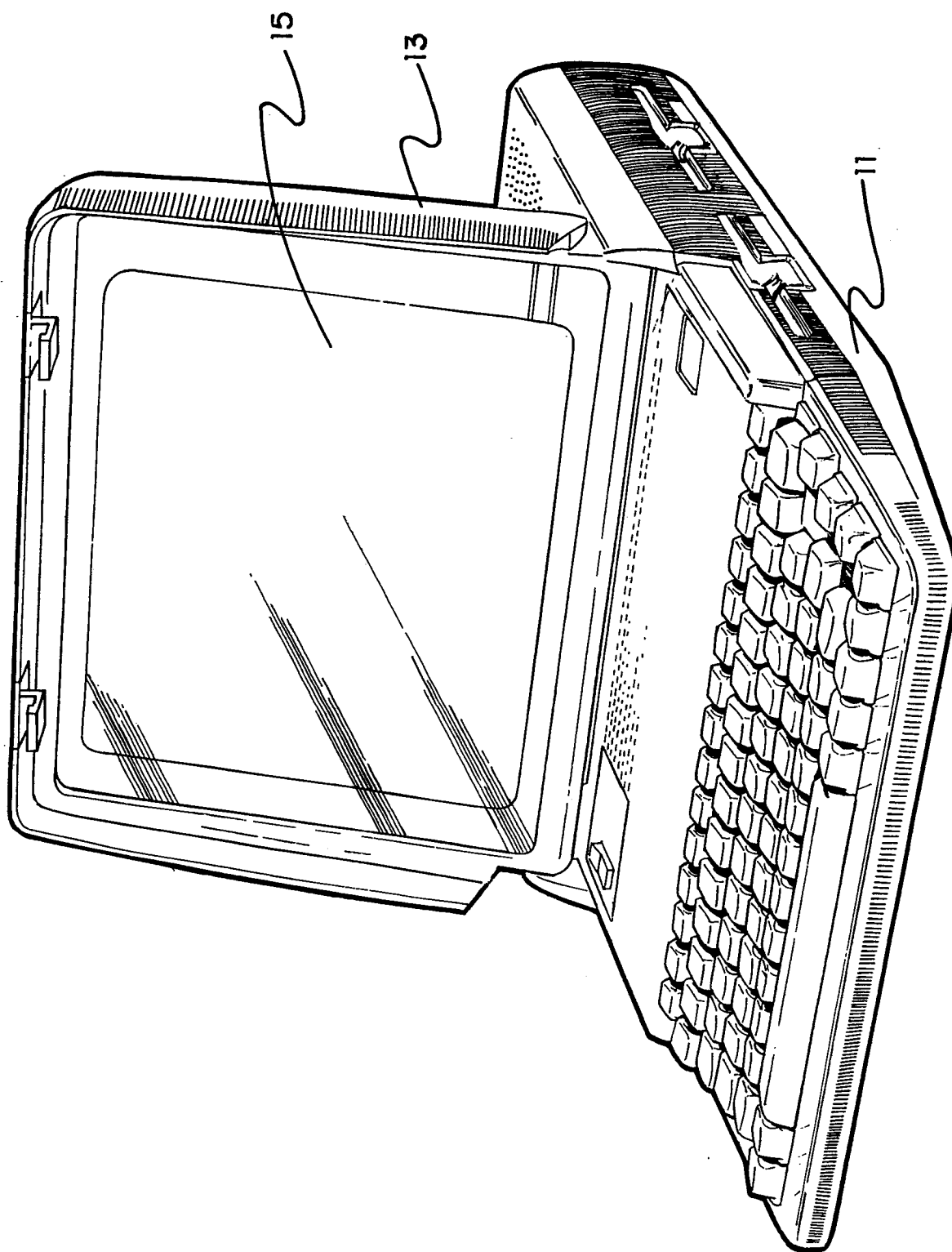
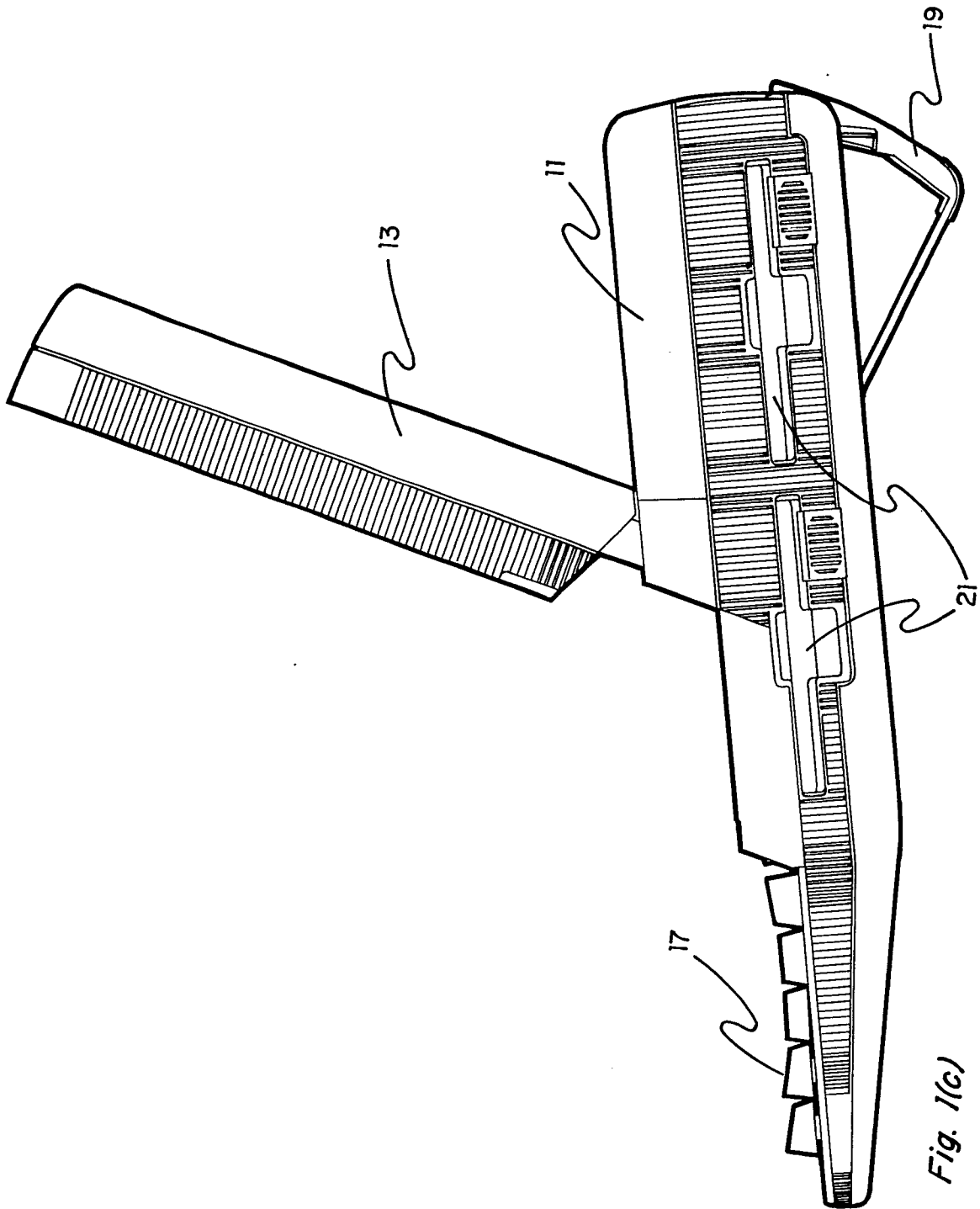


Fig 1(a)

2/55

*Fig. 1(b)*

3/55



4/55

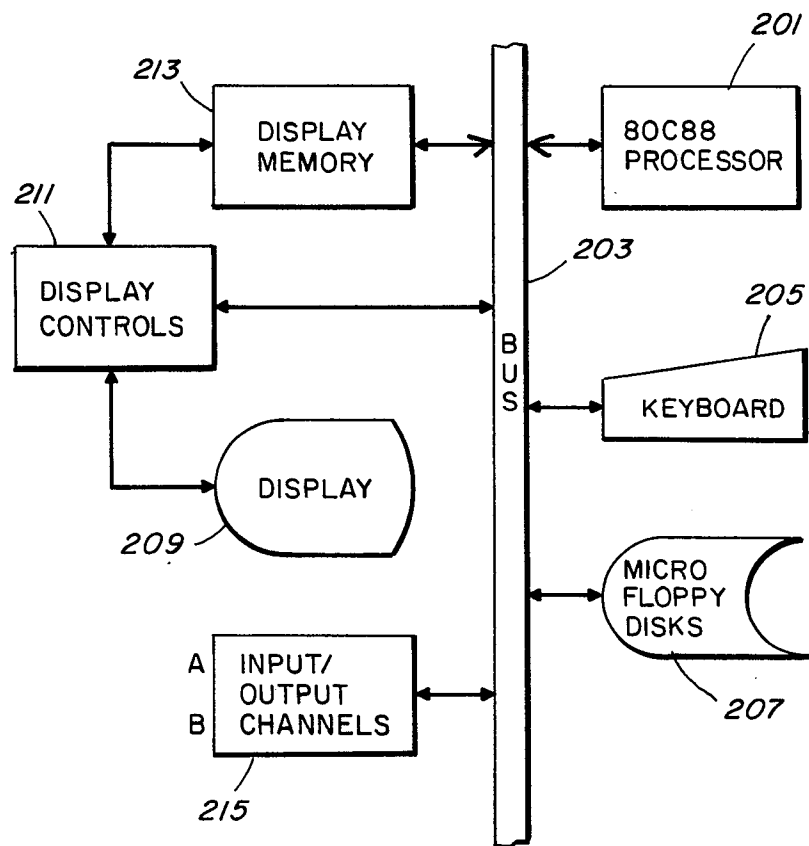


FIG. 2(a)

5/55

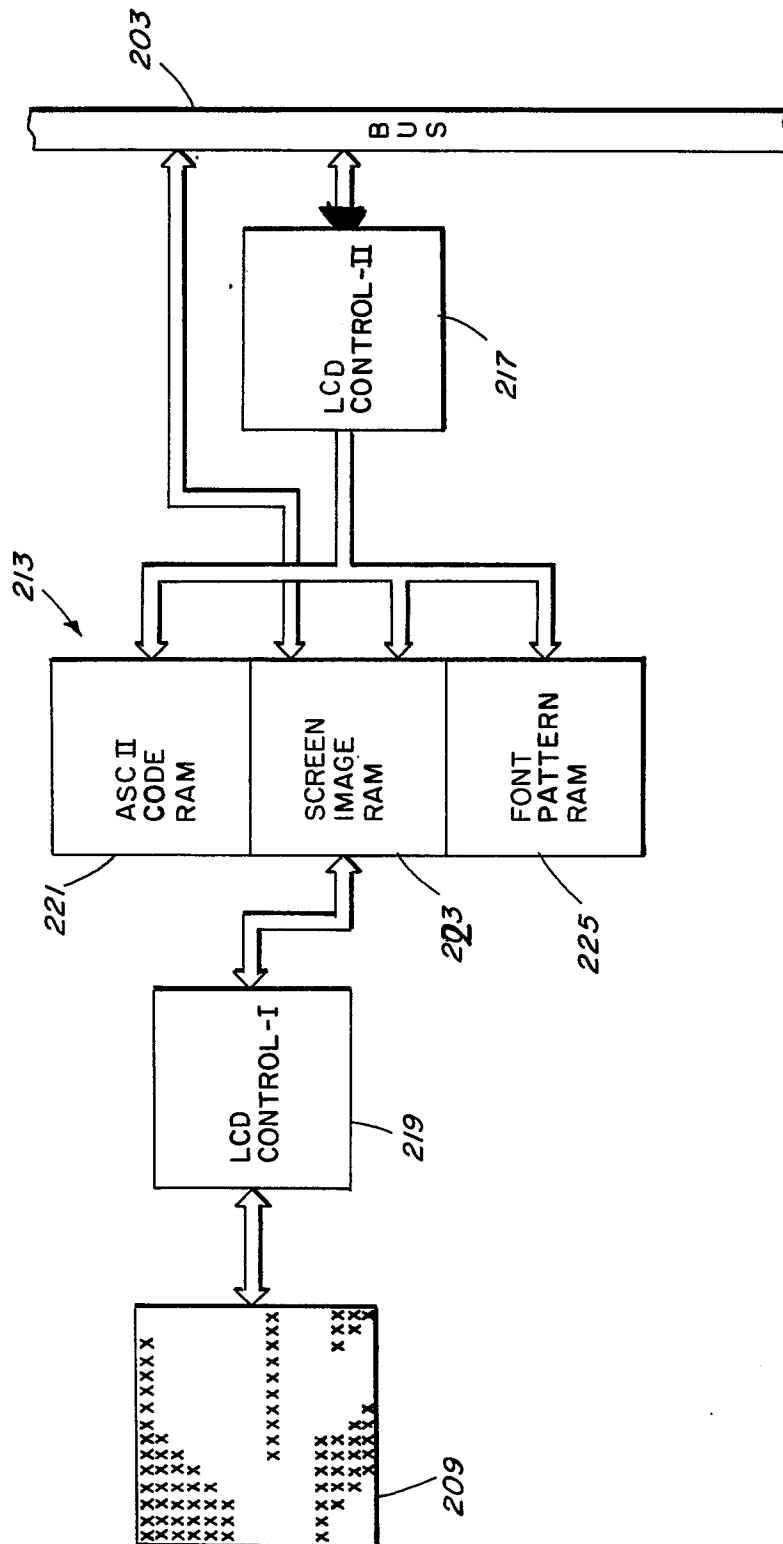


FIG. 2(b)

6/55

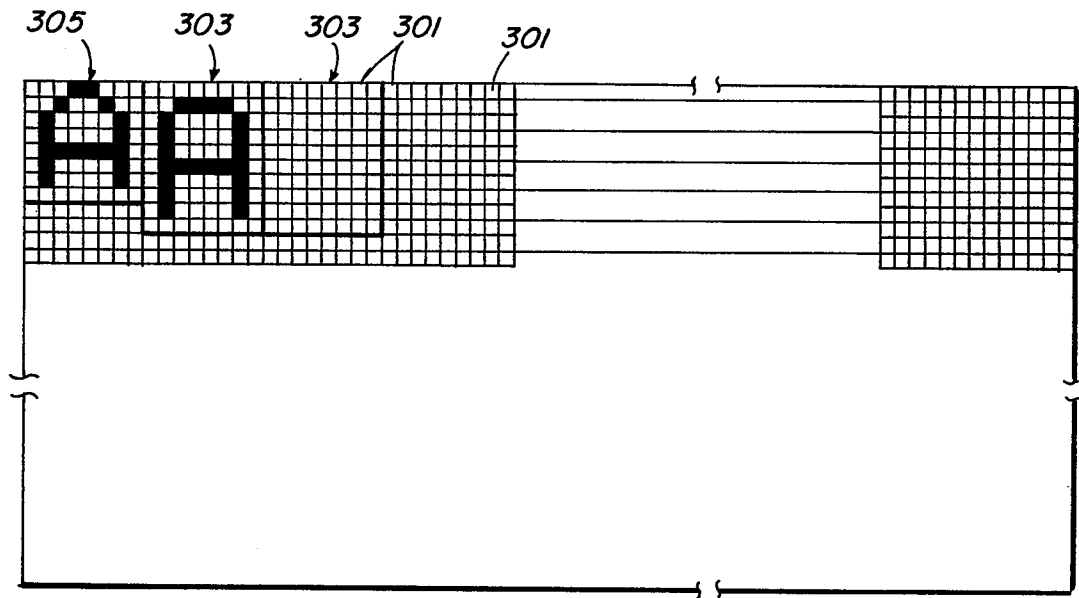


FIG. 3(a)

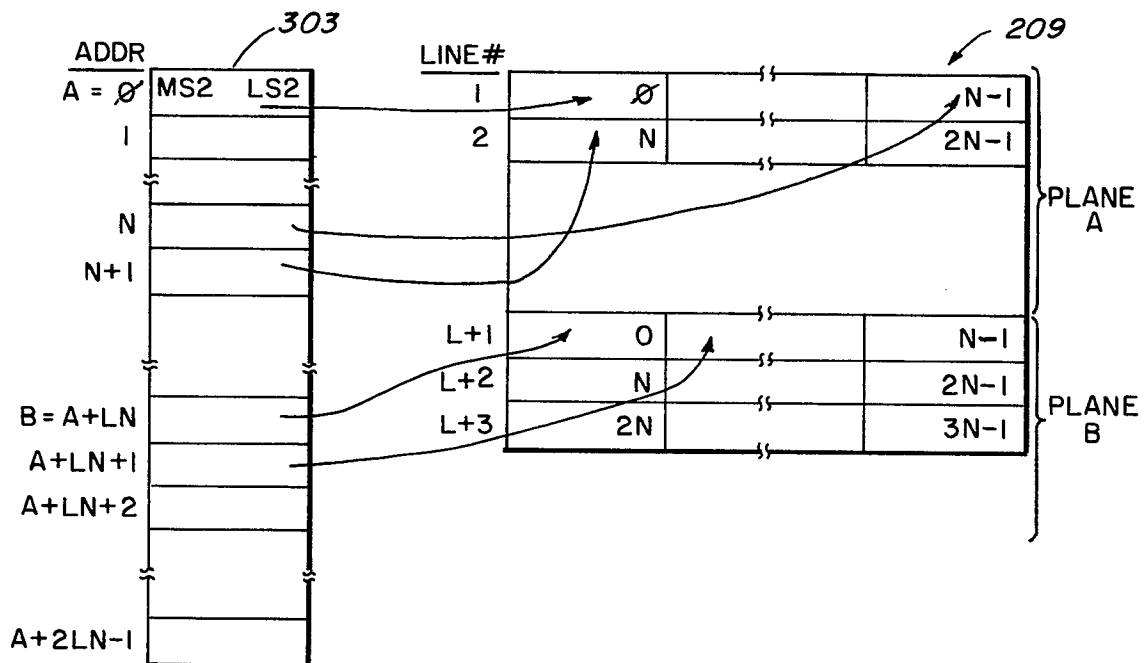


FIG. 3(e)

7/55

LINE WORD#		1	2	N-1		N
LINE #		1	2	N-1		N
1		A	A+1			A+N-1
2		A+N	A+N+1			A+2N-1
		VRAM ADDR PLANE A				
L-1						
L		A+(L-1)N				A+LN-1

FIG. 3(b)

LINE WORD #		1	2	N-1		N
LINE #		1	2	N-1		N
1		A	A+1			A+N-1
2		A+N	A+N+1			A+2N-1
		PLANE A				
L-1		VRAM ADDR				
L		A+(L-1)N				A+LN-1
L+1		B	B+N+1			B+N-1
		B+N	B+N+1			B+2N-1
		PLANE B				
2L		B+(L-1)N				B+LN-1

FIG. 3(c)

LINE WORD#												
LINE #		1	2			N-1	N	N+1			2N-1	2N
1		A	A+1				A+N-1	C	C+1			C+N-1
2		A+2N	A+2N+1				A+3N-1	C+2N	C+2N+1			C+3N-1
		PLANE A					PLANE C					
L-1		VRAM ADDR										
L		A+2(L-1)N					A+(2L-1)N-1	C+2(L-1)N			C+(2L-1)N-1	
L+1		B	B+1				B+N-1	D	D+1			D+N-1
		B+2N	B+2N+1				B+3N-1	D+2N	D+2N+1			D+3N-1
		PLANE B					PLANE D					
2L		B+2(L-1)N					B+(2L-1)N-1	D+2(L-1)N			D+(2L-1)N-1	

FIG. 3(d)

44/55

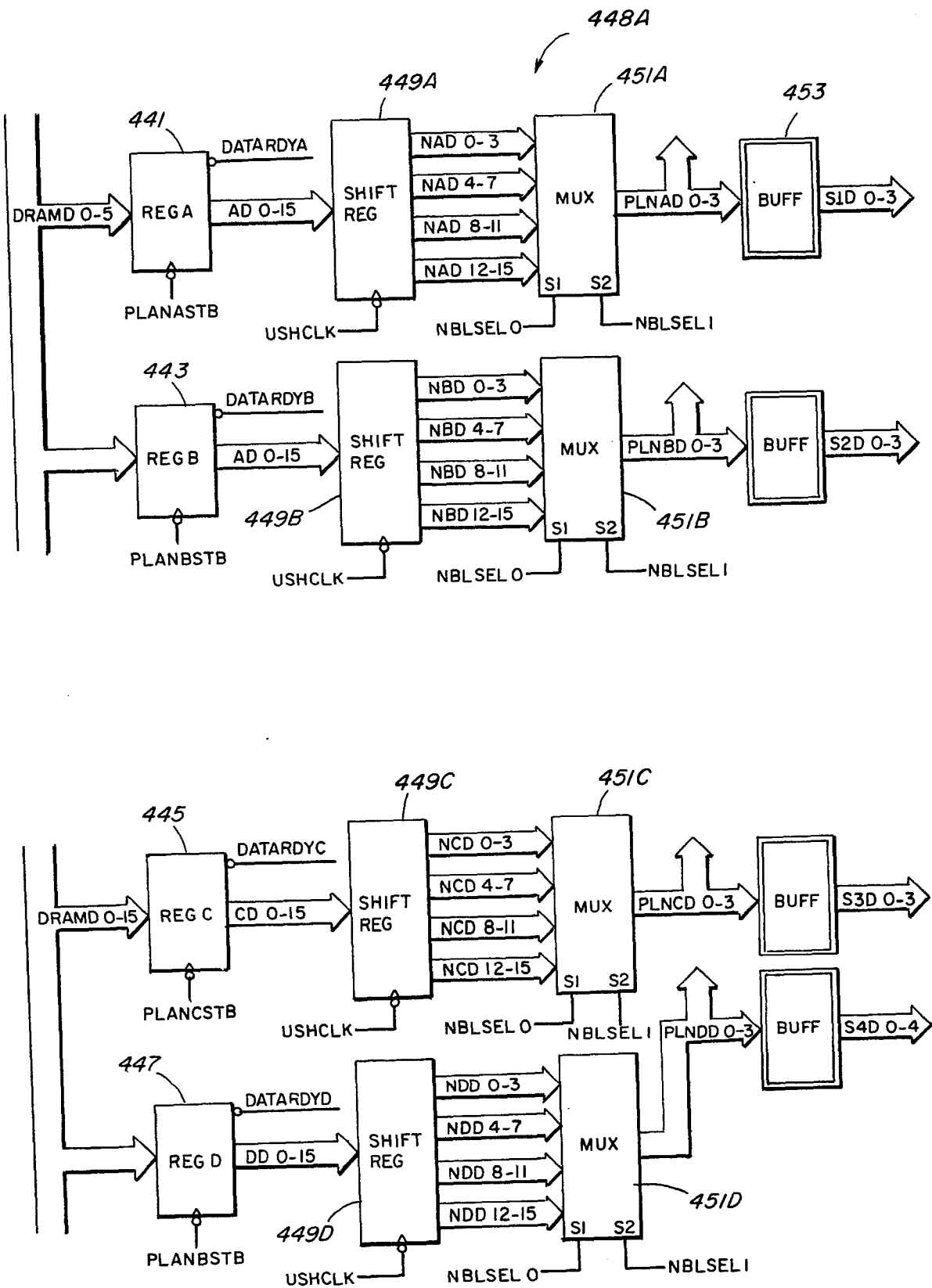


FIG. 4 SHEET 1 OF 3

9/55

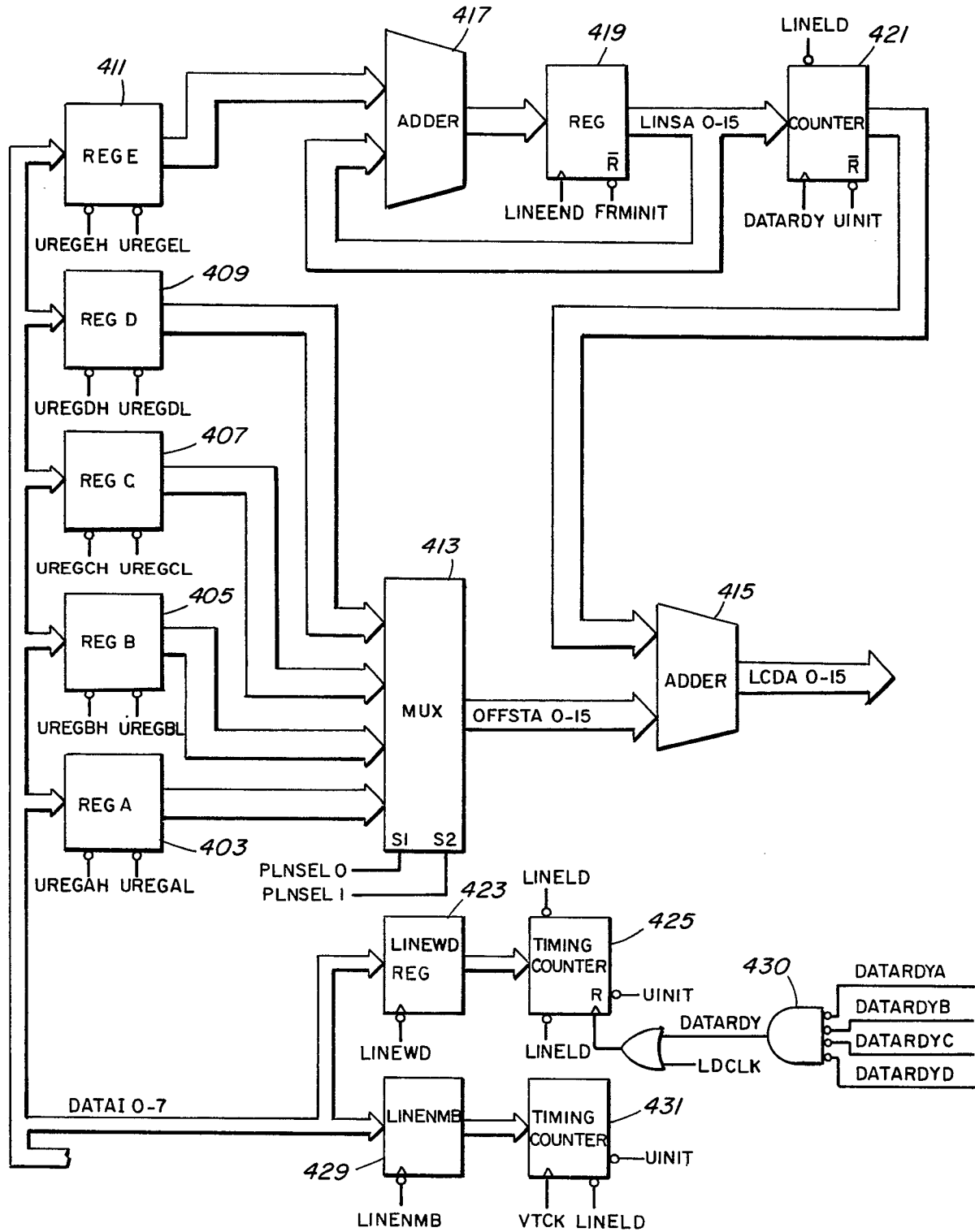
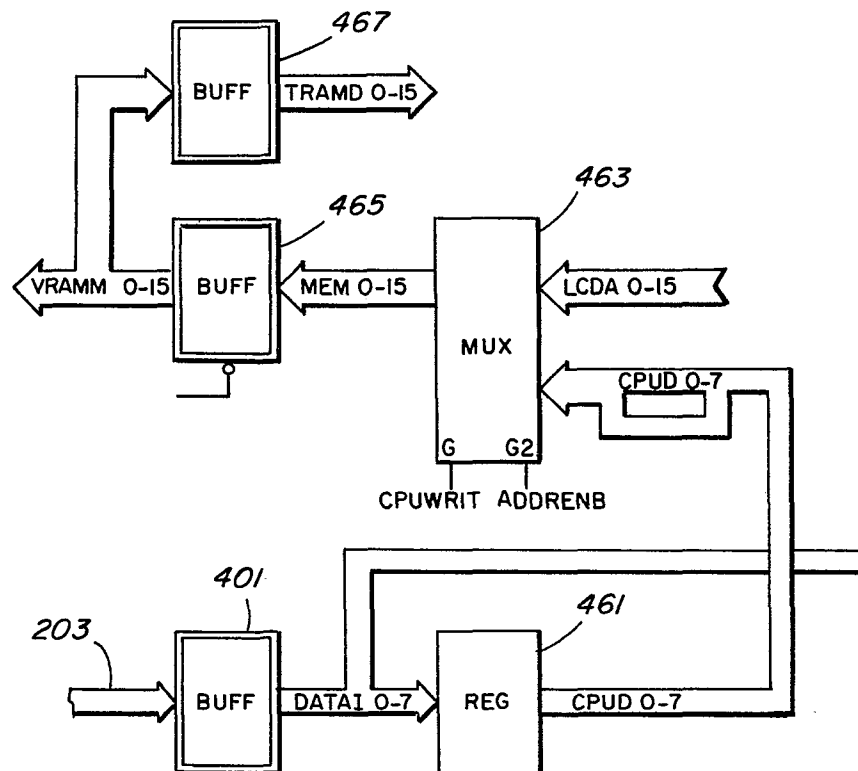


FIG. 4 SHEET 2 OF 3

10/55

**FIG. 4** SHEET 3 OF 3

11/55

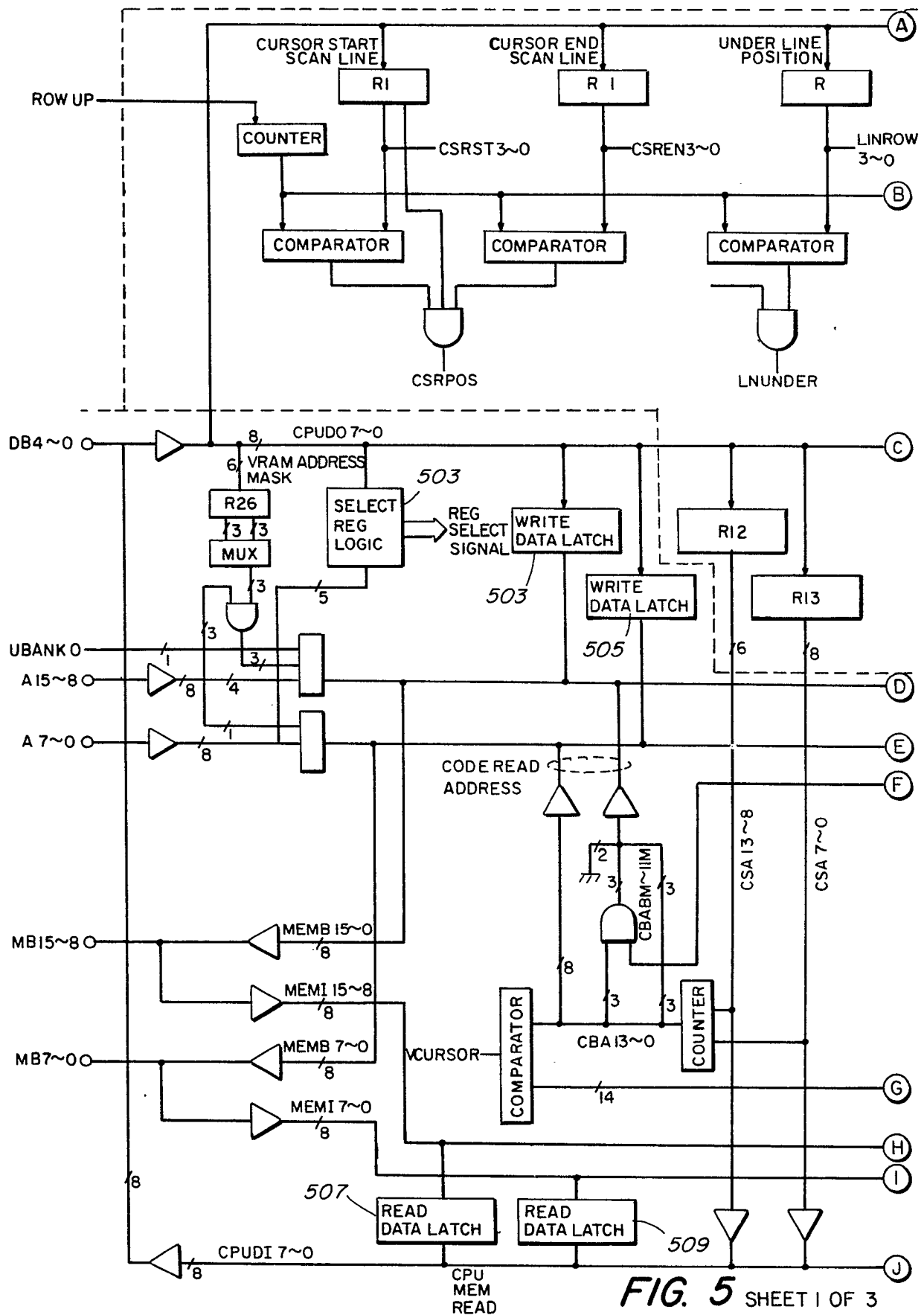
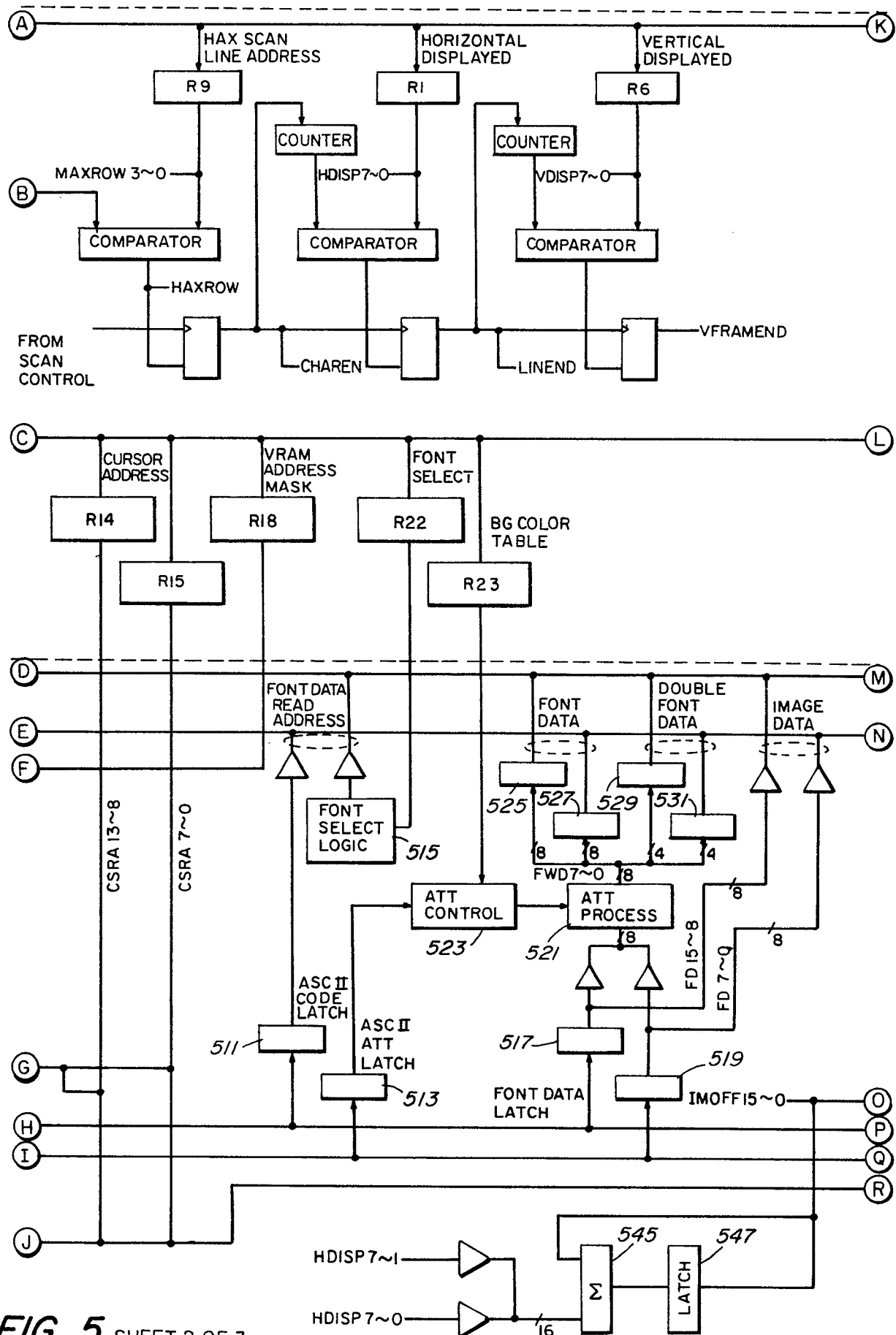


FIG. 5 SHEET 1 OF 3

12/55



13/55

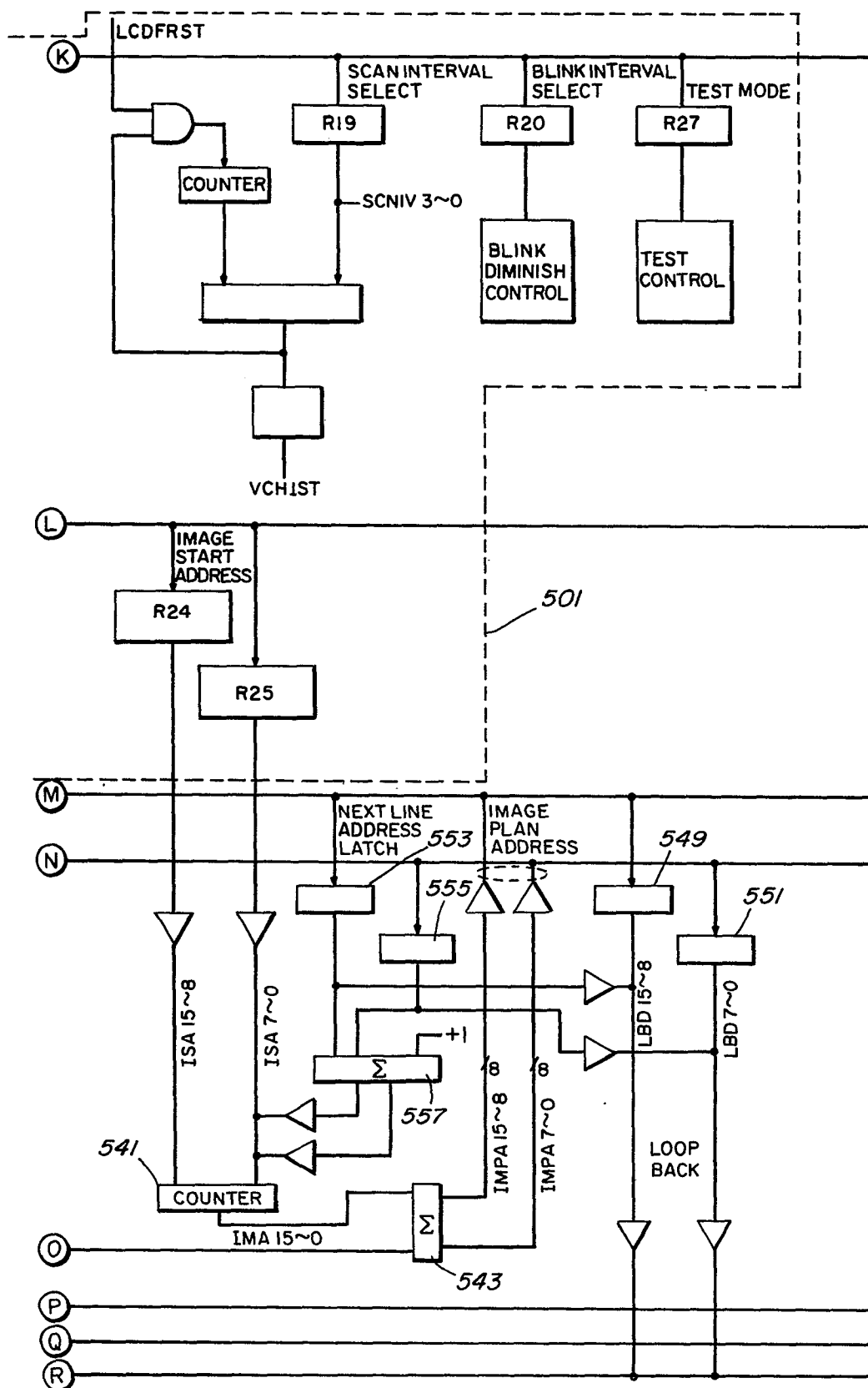


FIG. 5 SHEET 3 OF 3

14/55



15/55

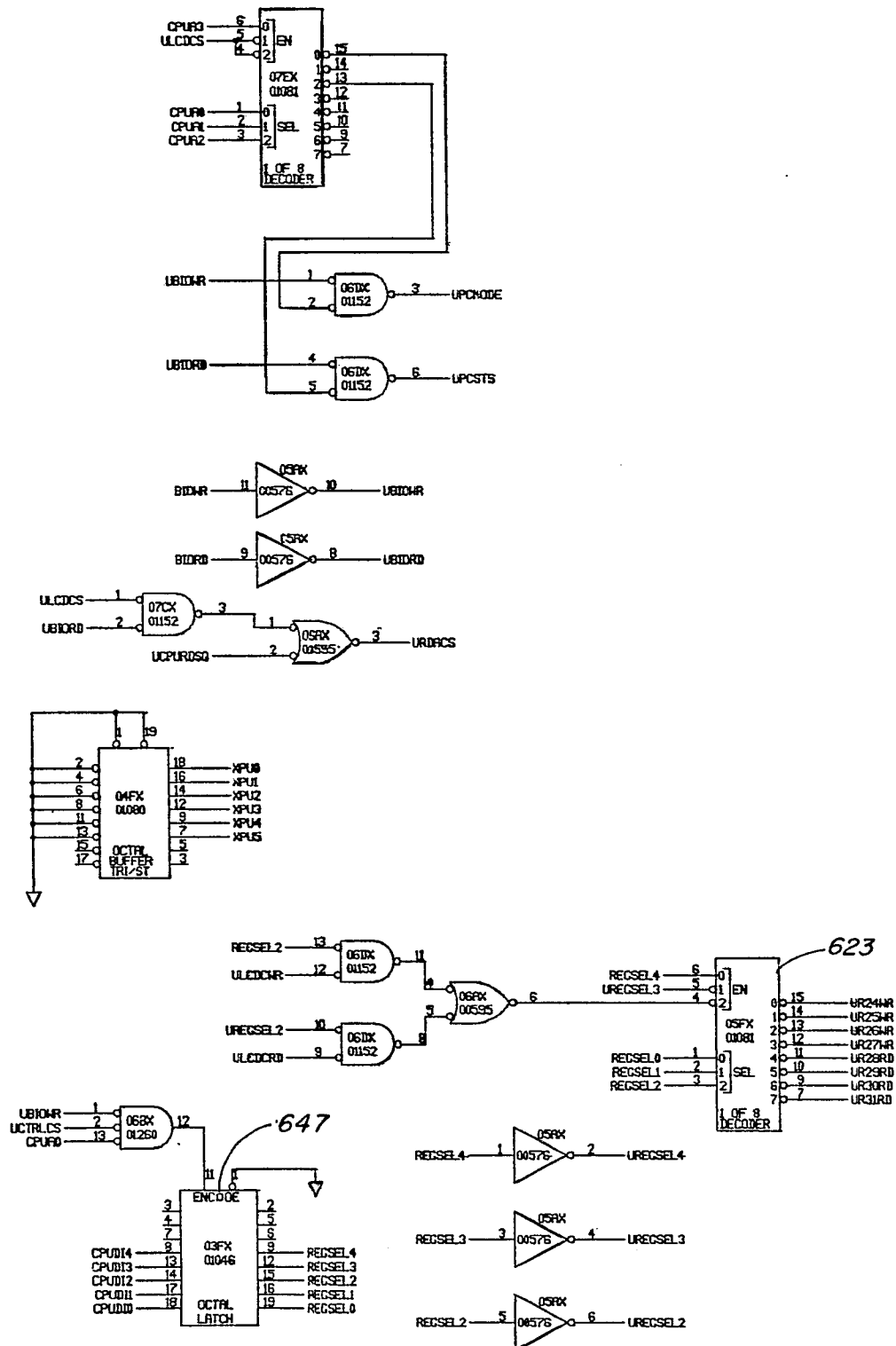


FIG. 6 SHEET 2 OF 3

16/55

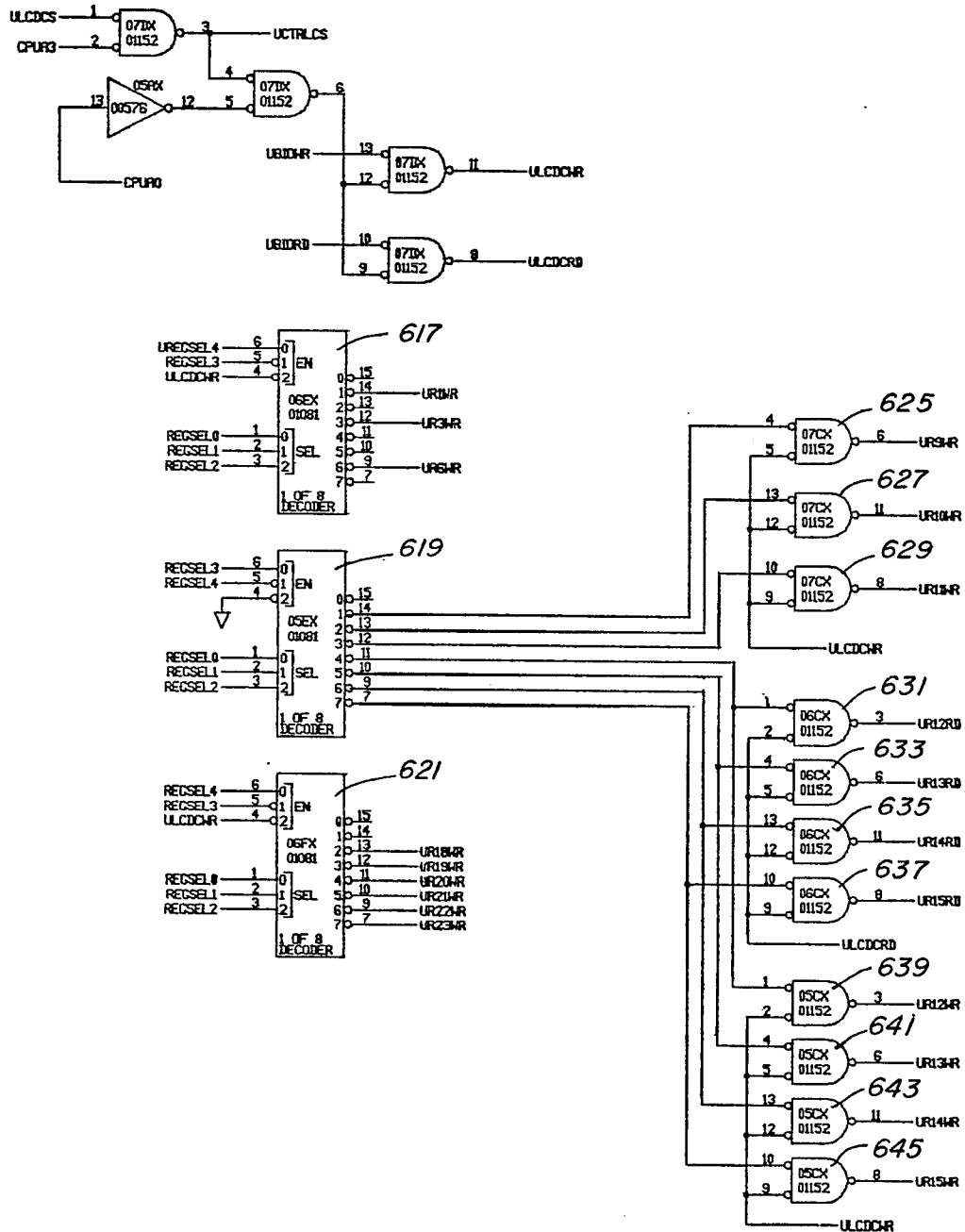


FIG. 6 SHEET 3 OF 3

17/55



18/55

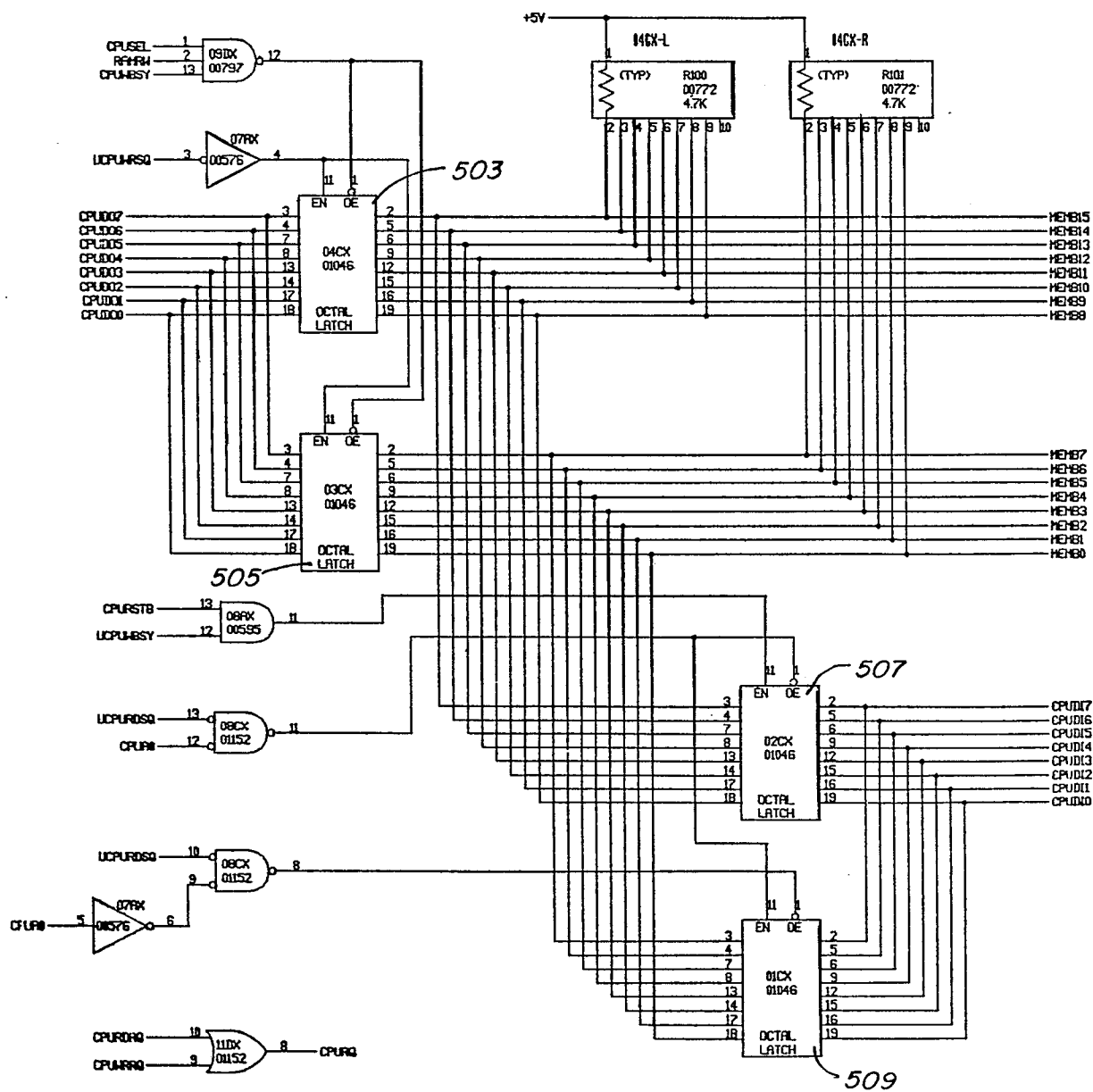


FIG. 7 SHEET 2 OF 2

19/55

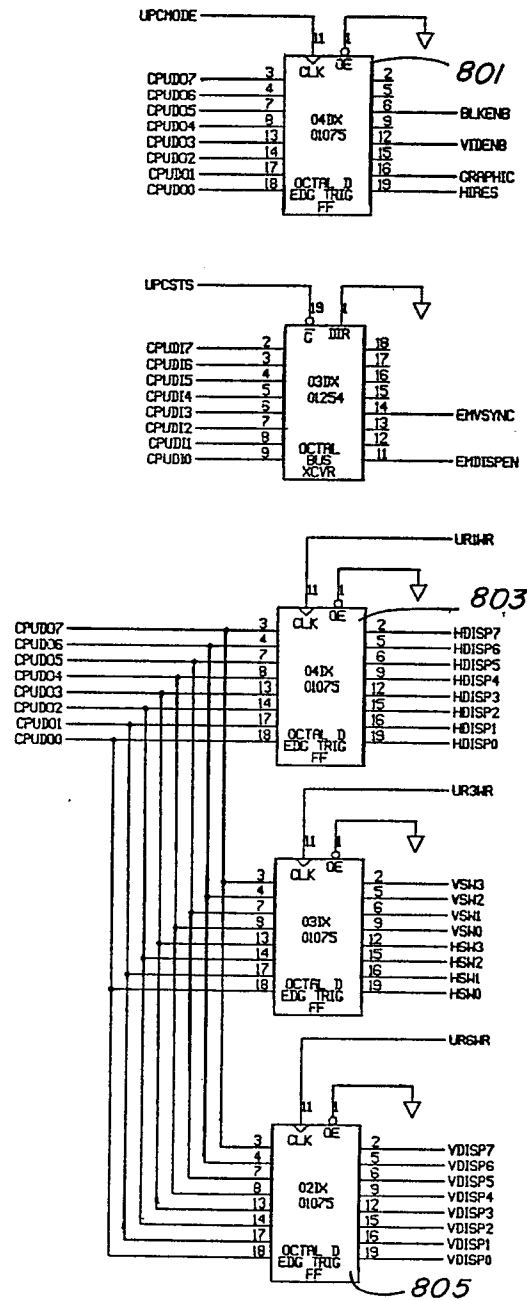


FIG. 8 SHEET 1 OF 3

20/55

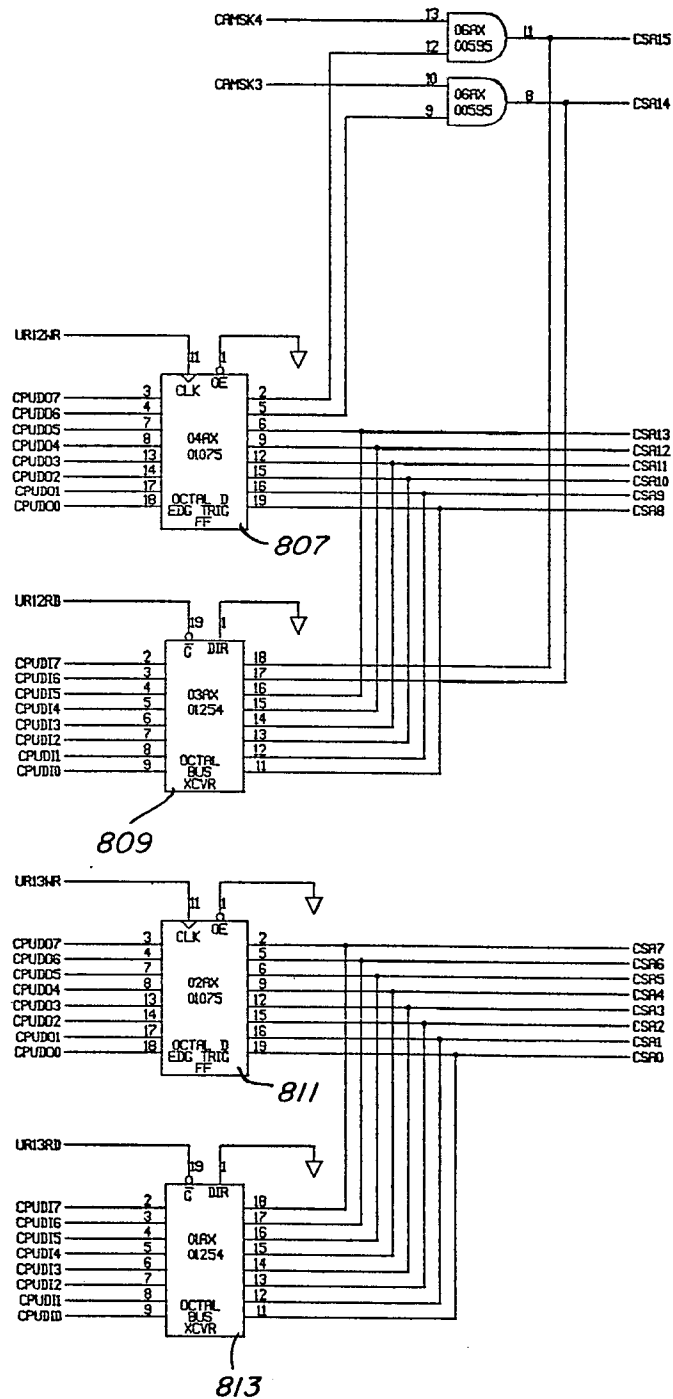


FIG. 8 SHEET 2 OF 3

21/55

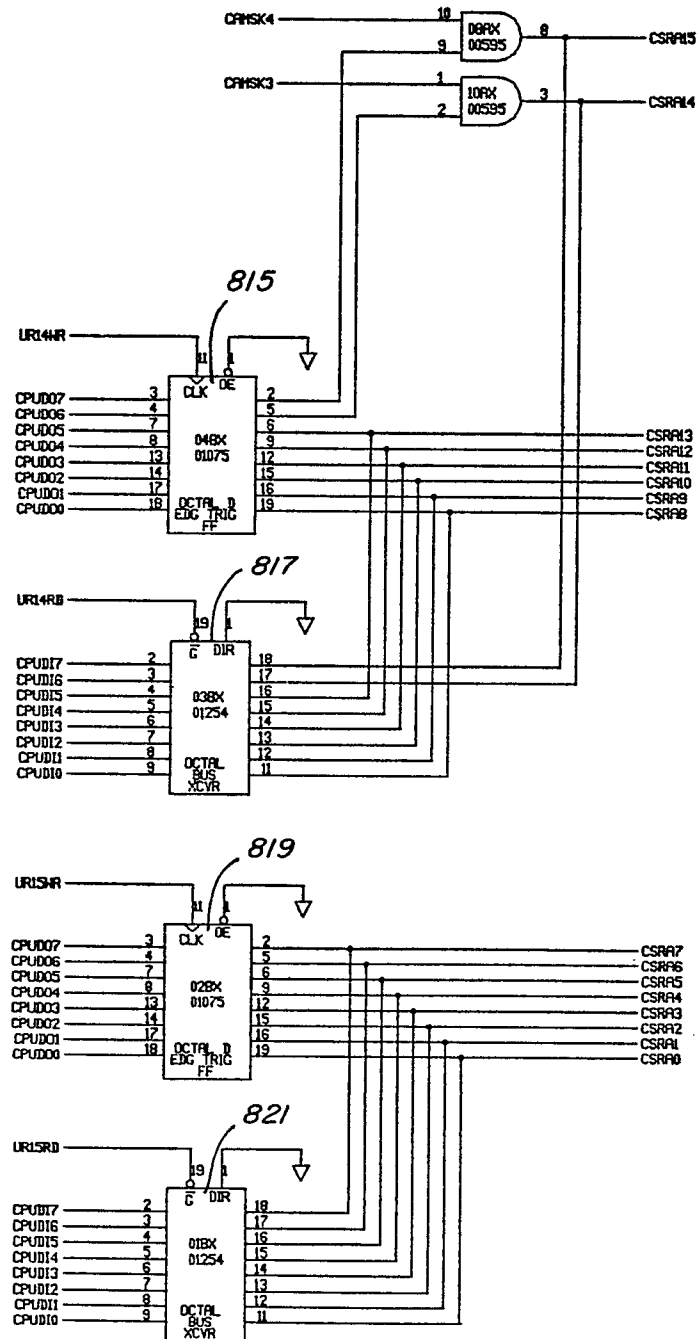


FIG. 8 SHEET 3 OF 3

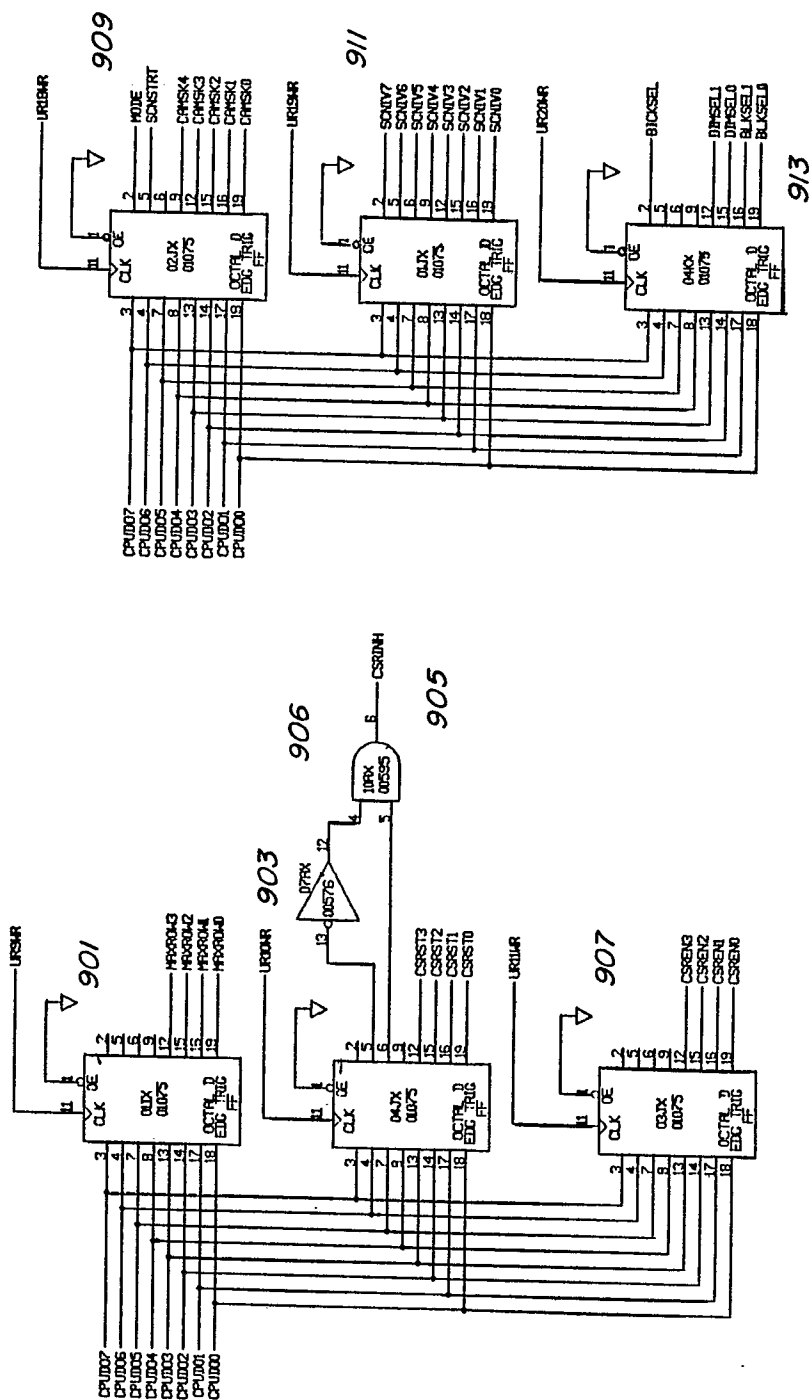
$$22 \overline{) 55}$$


FIG. 9 SHEET 1 OF 2

23/55

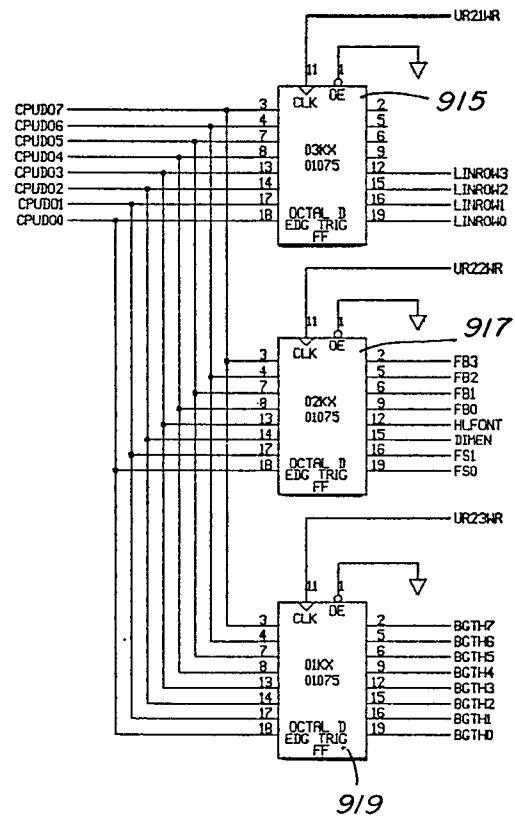


FIG. 9 SHEET 2 OF 2

24/55

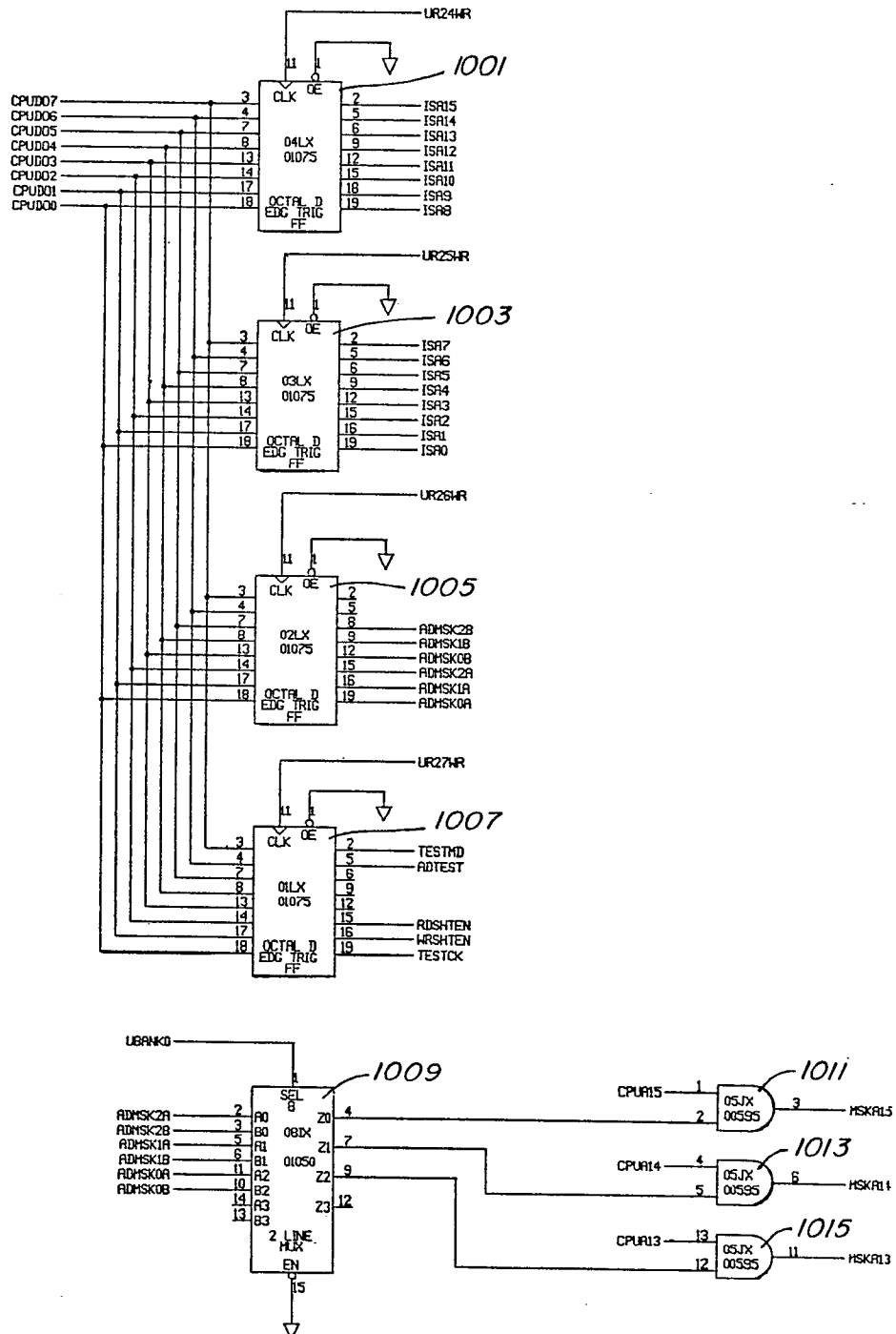


FIG. 10 SHEET 1 OF 2

0175499

25/55

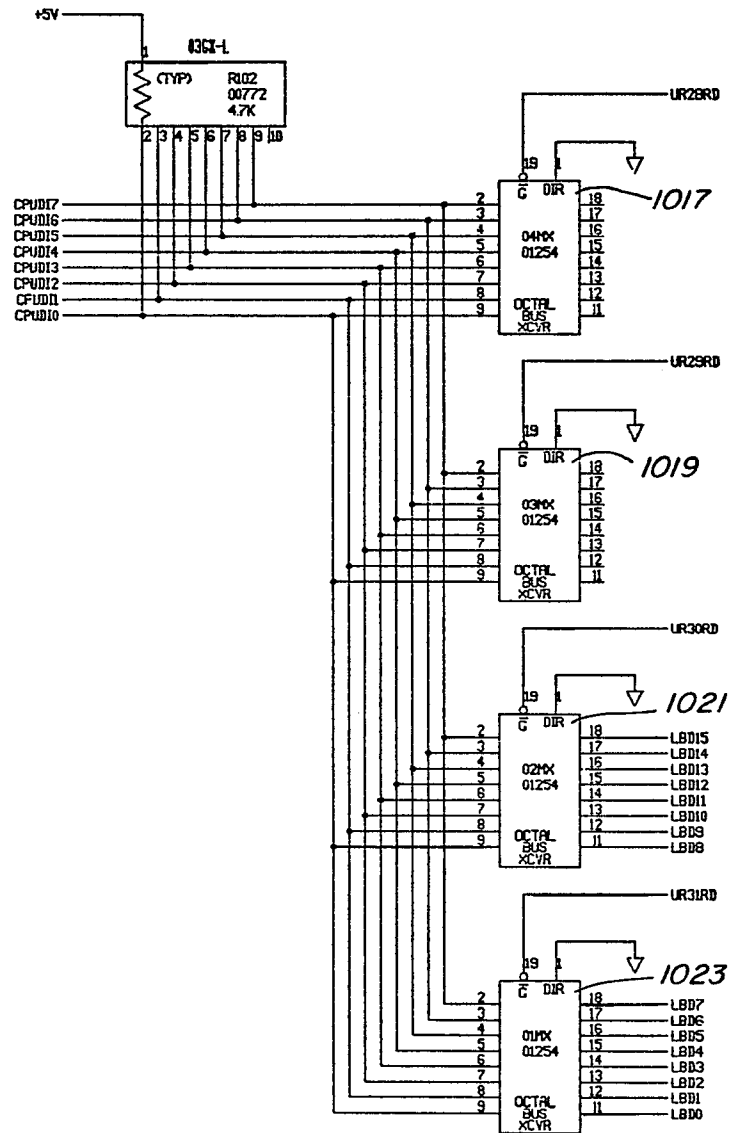


FIG. 10 SHEET 2 OF 2

26/55

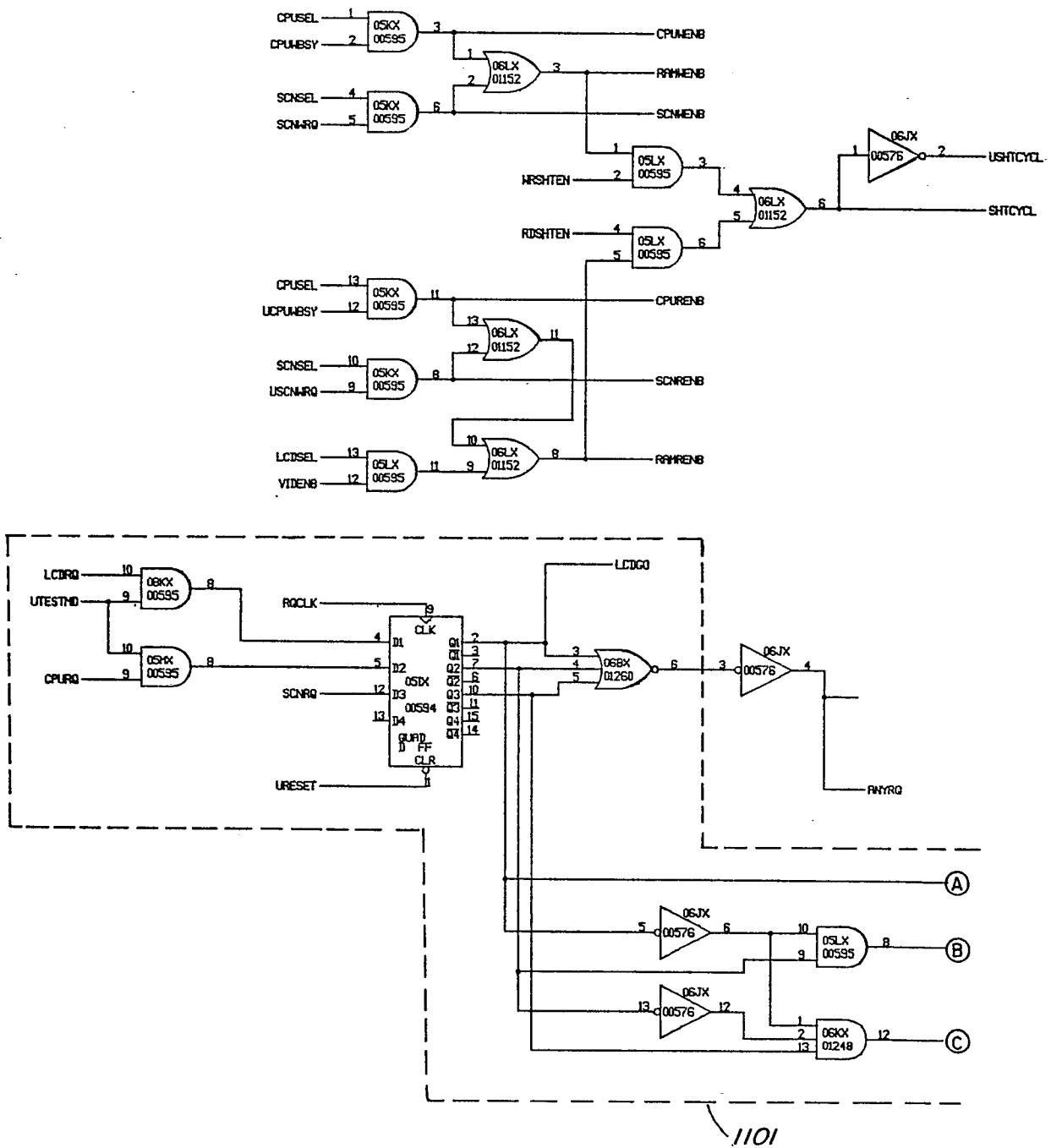


FIG. 11 SHEET 1 OF 2

27/55

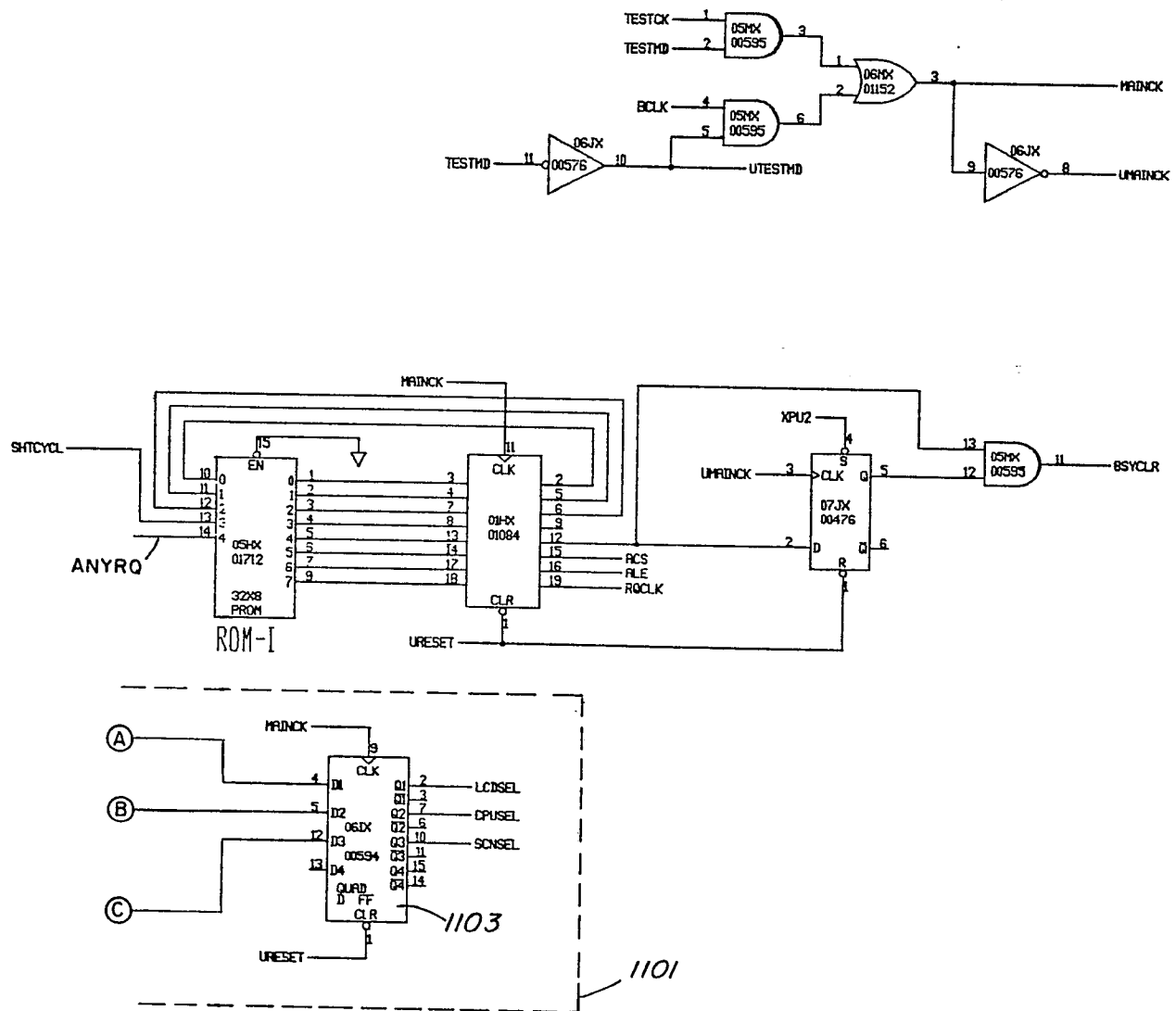


FIG. 11 SHEET 2 OF 2

28/55

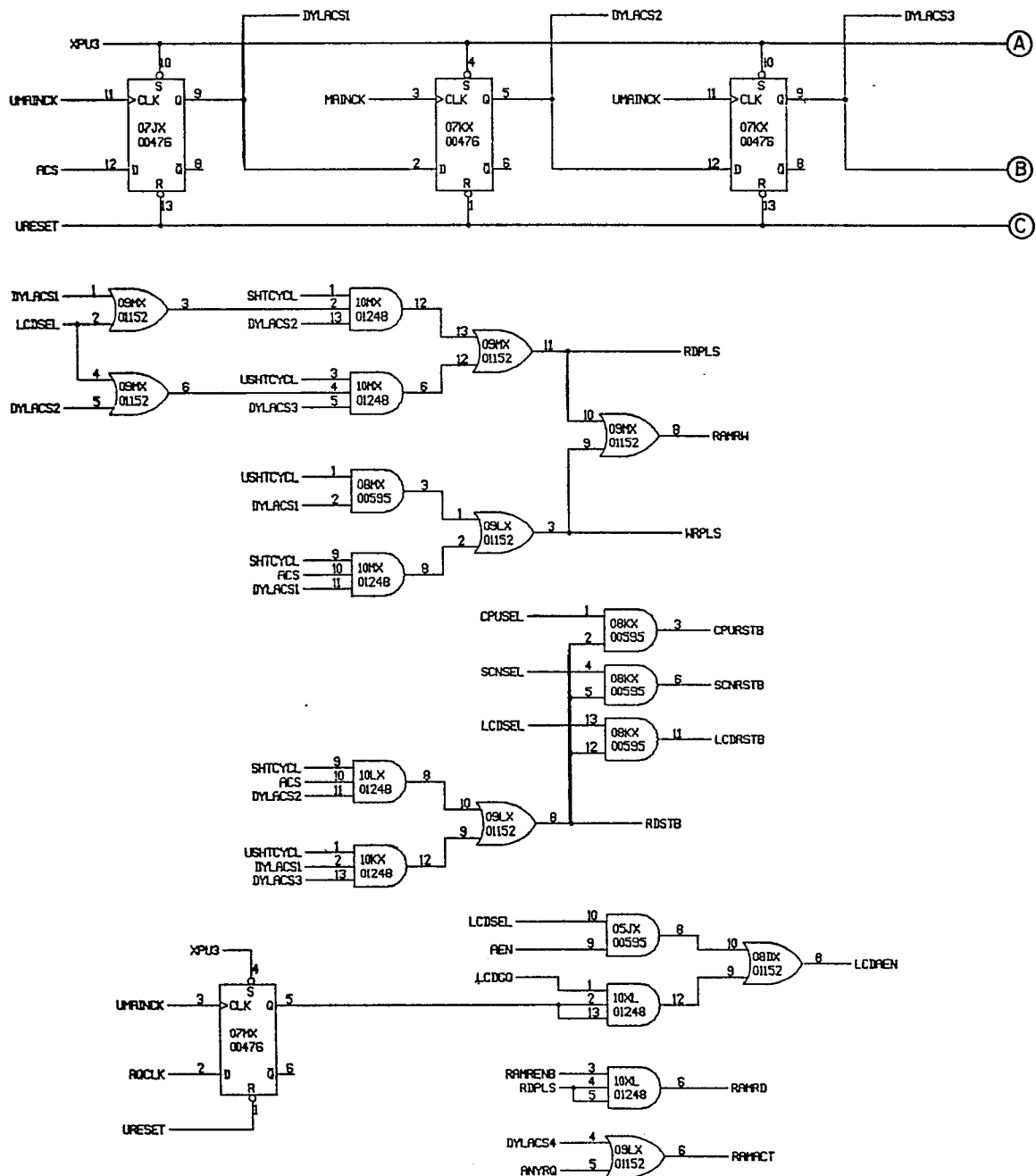


FIG. 12 SHEET 1 OF 2

29/53

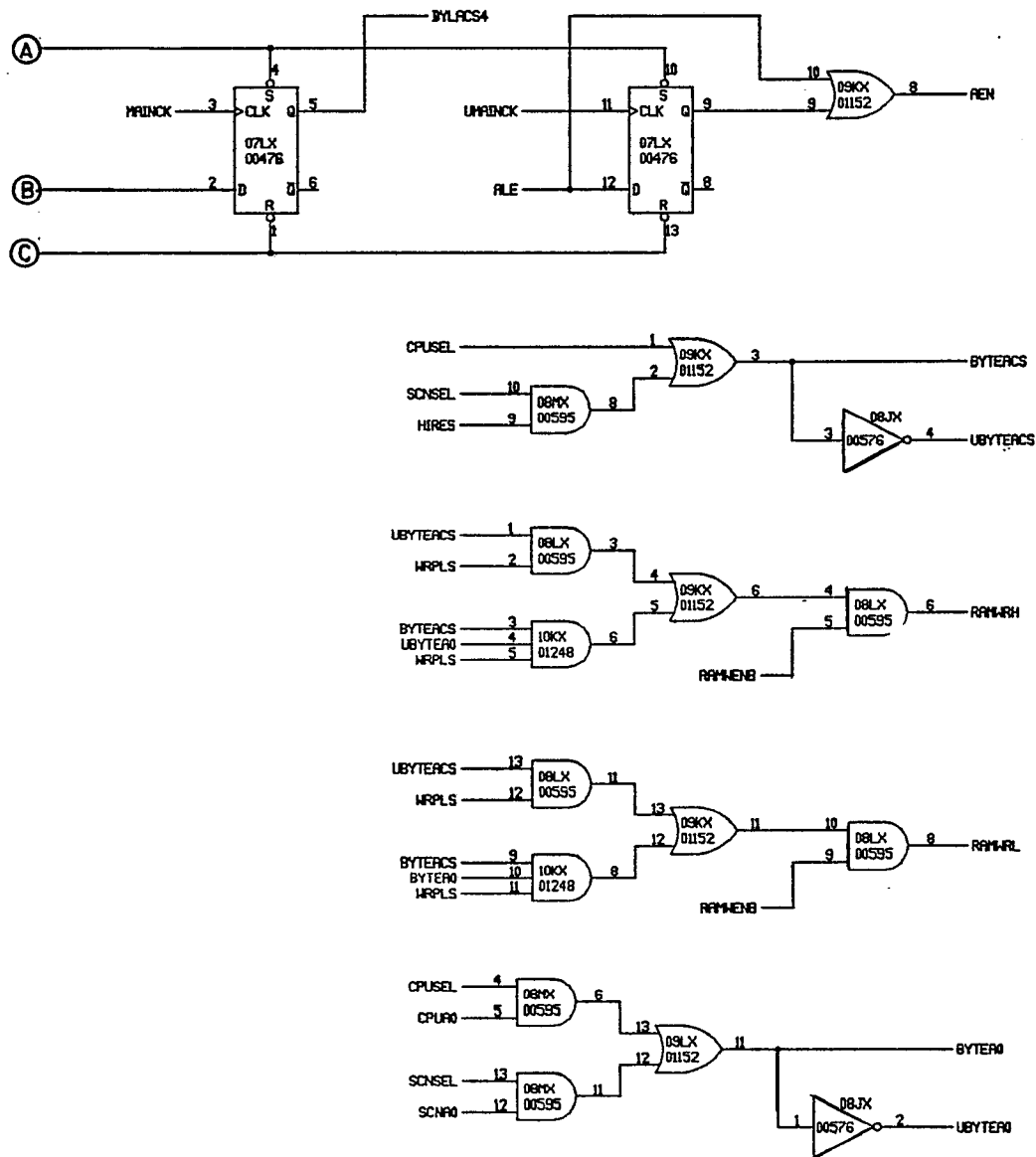


FIG. 12 SHEET 2 OF 2

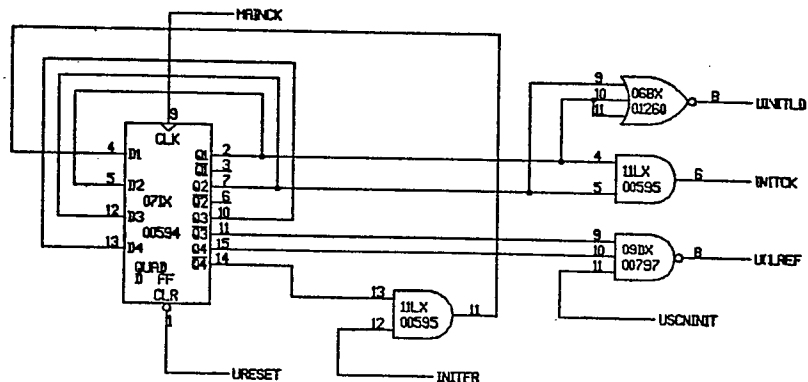


FIG. 13 SHEET 1 OF 2

31/55

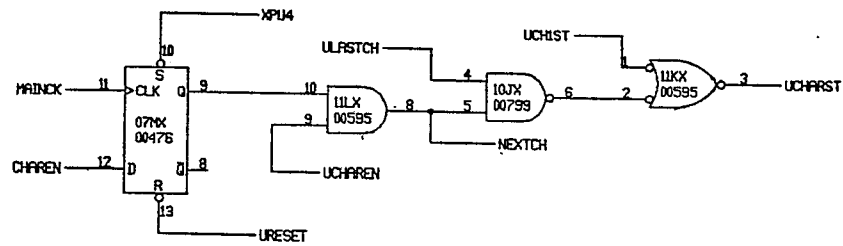
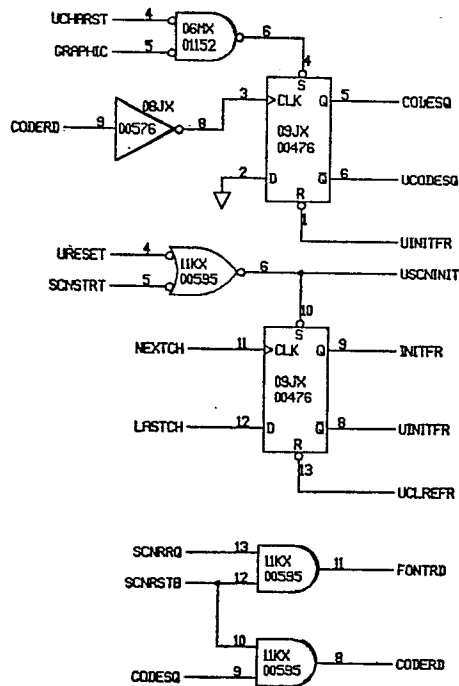


FIG. 13 SHEET 2 OF 2

32/55

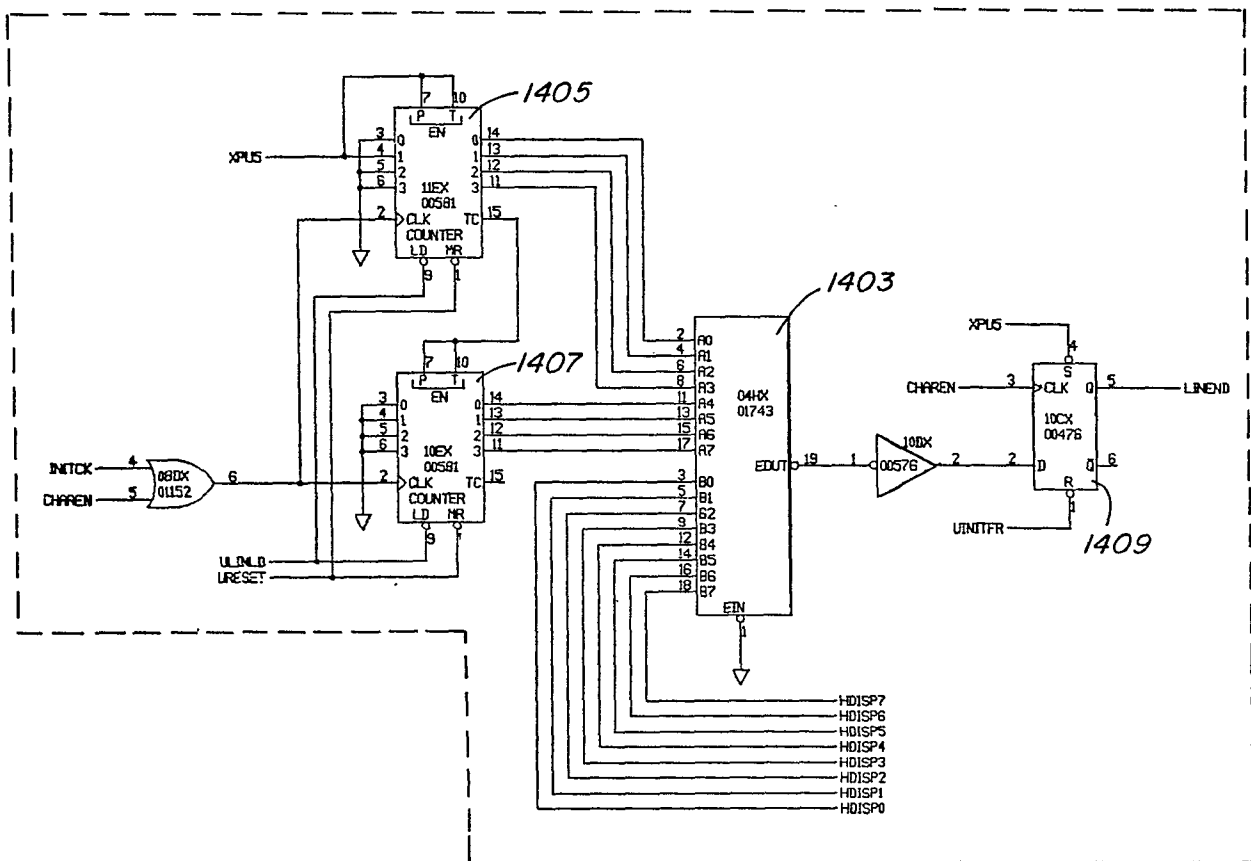


FIG. 14 SHEET 1 OF 3

33/55

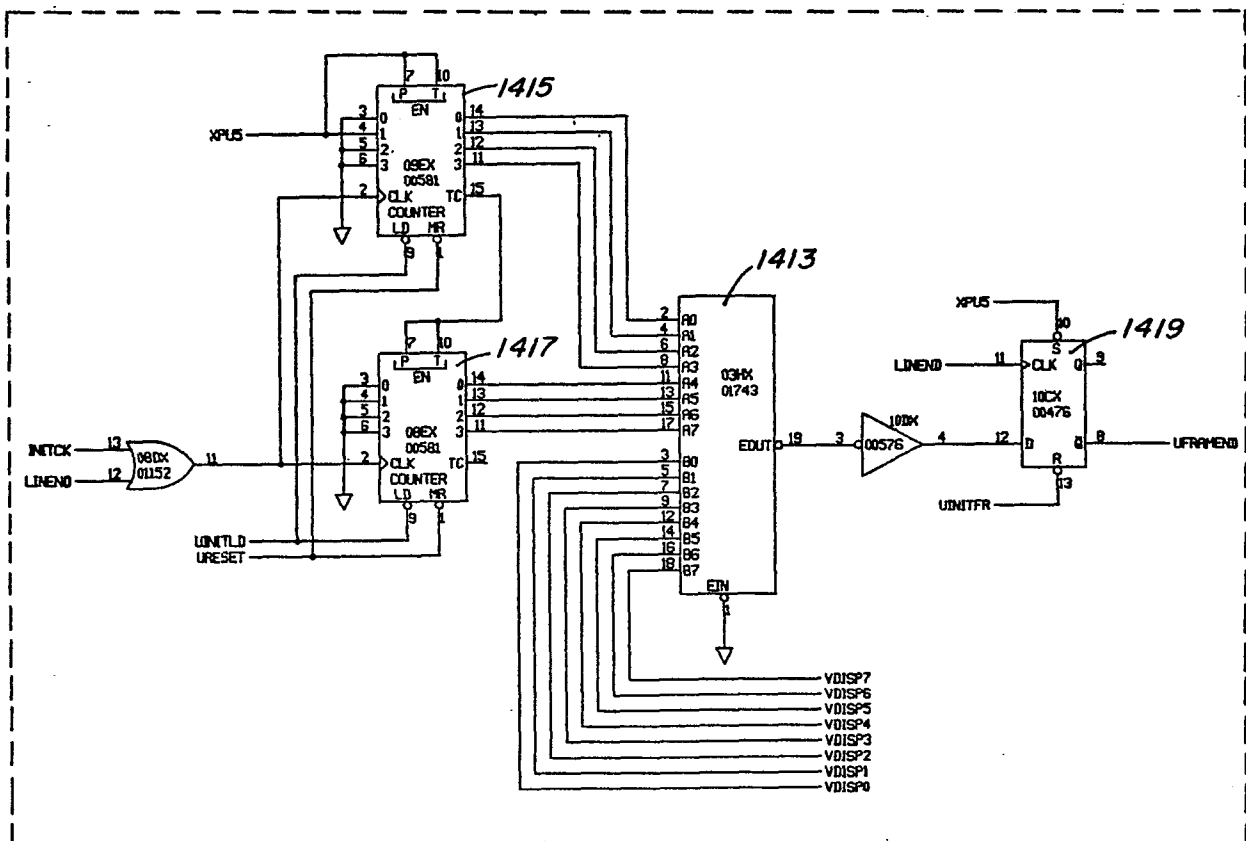
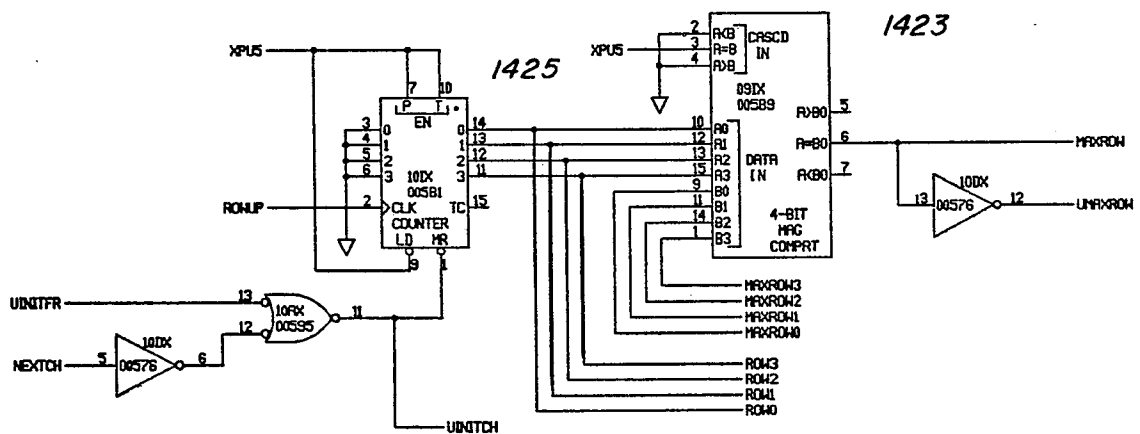


FIG. 14 SHEET 2 OF 3

0175499

34/55



1421

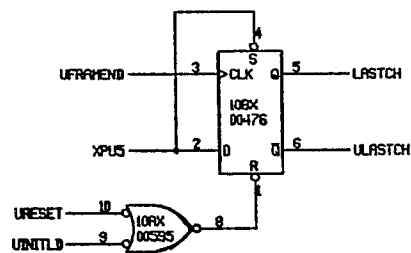


FIG. 14 SHEET 3 OF 3

35/55

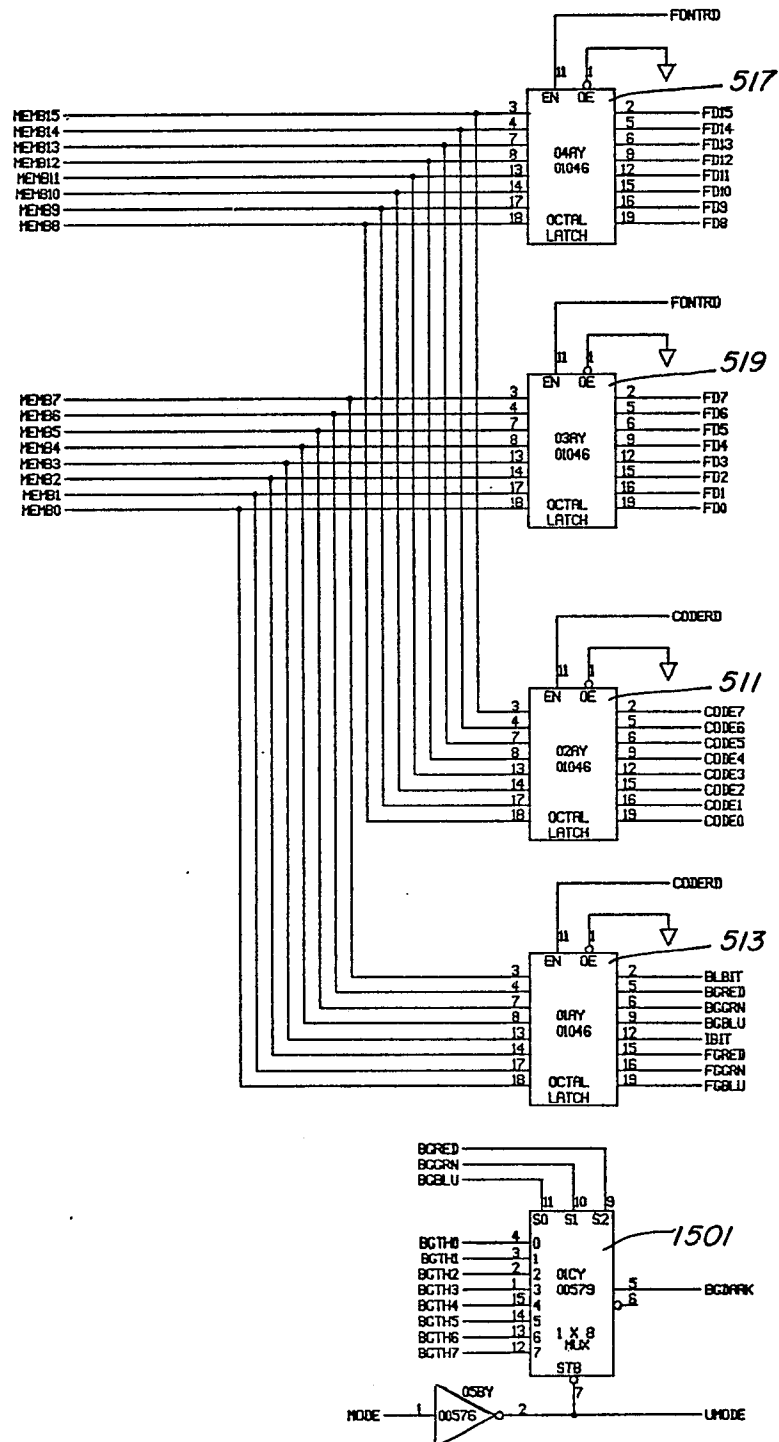


FIG. 15 SHEET 1 OF 3

36/55

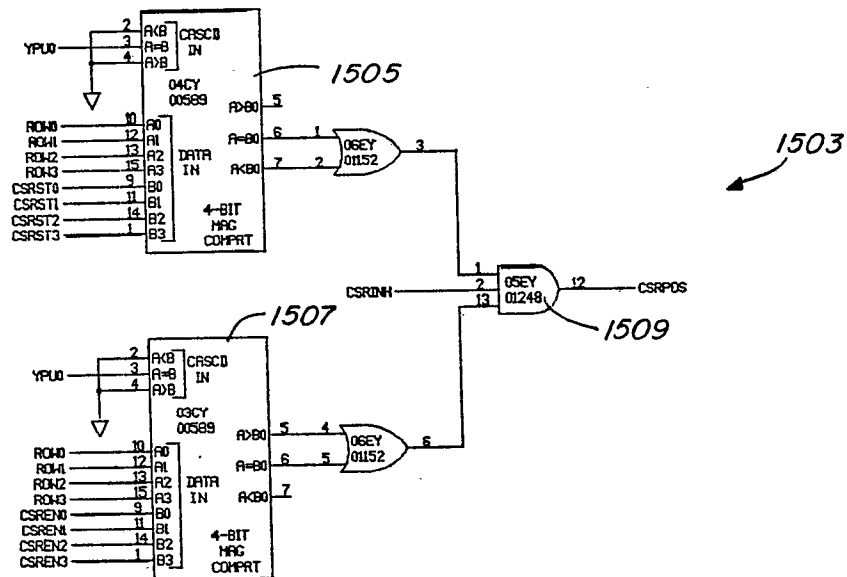


FIG. 15 SHEET 2 OF 3

0175499

37/55

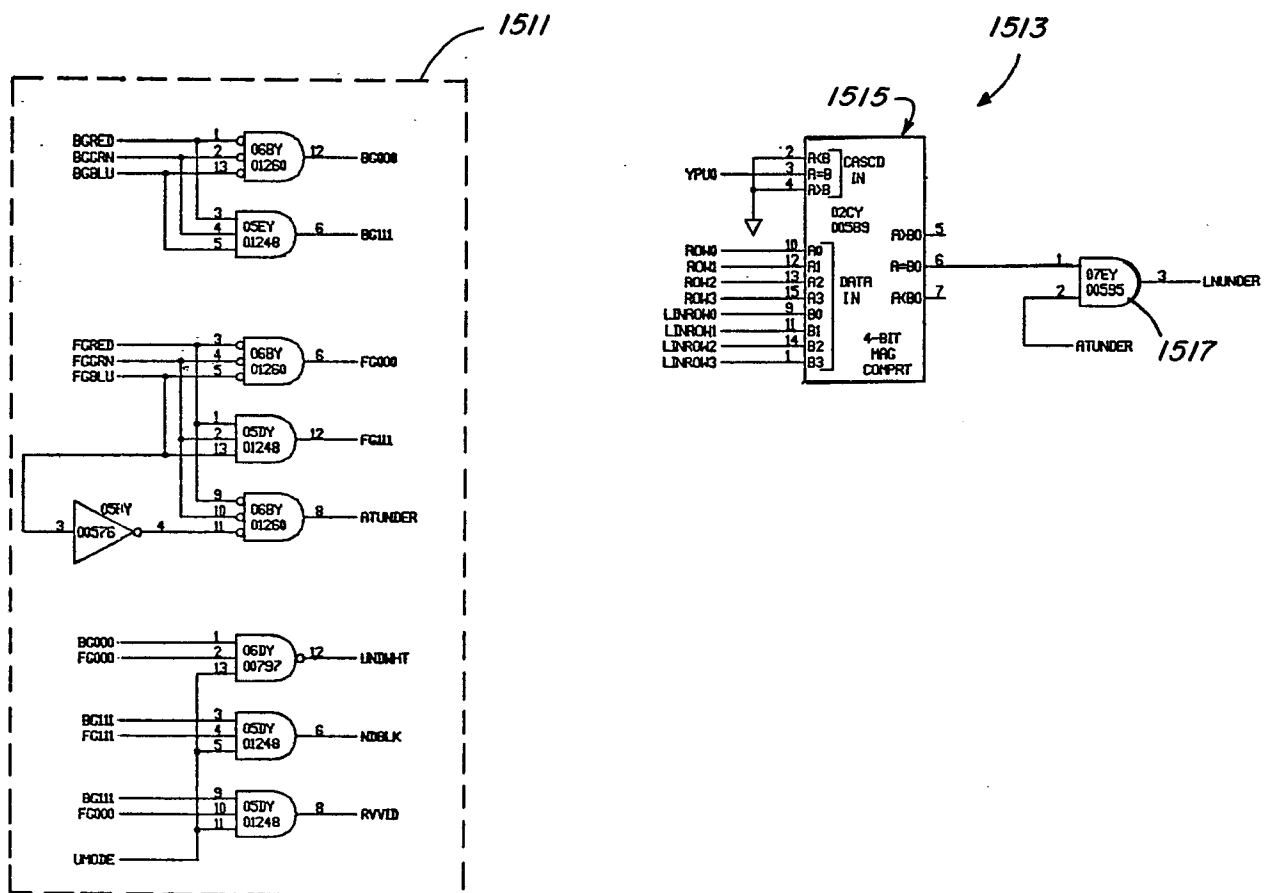


FIG. 15 SHEET 3 OF 3

38/55

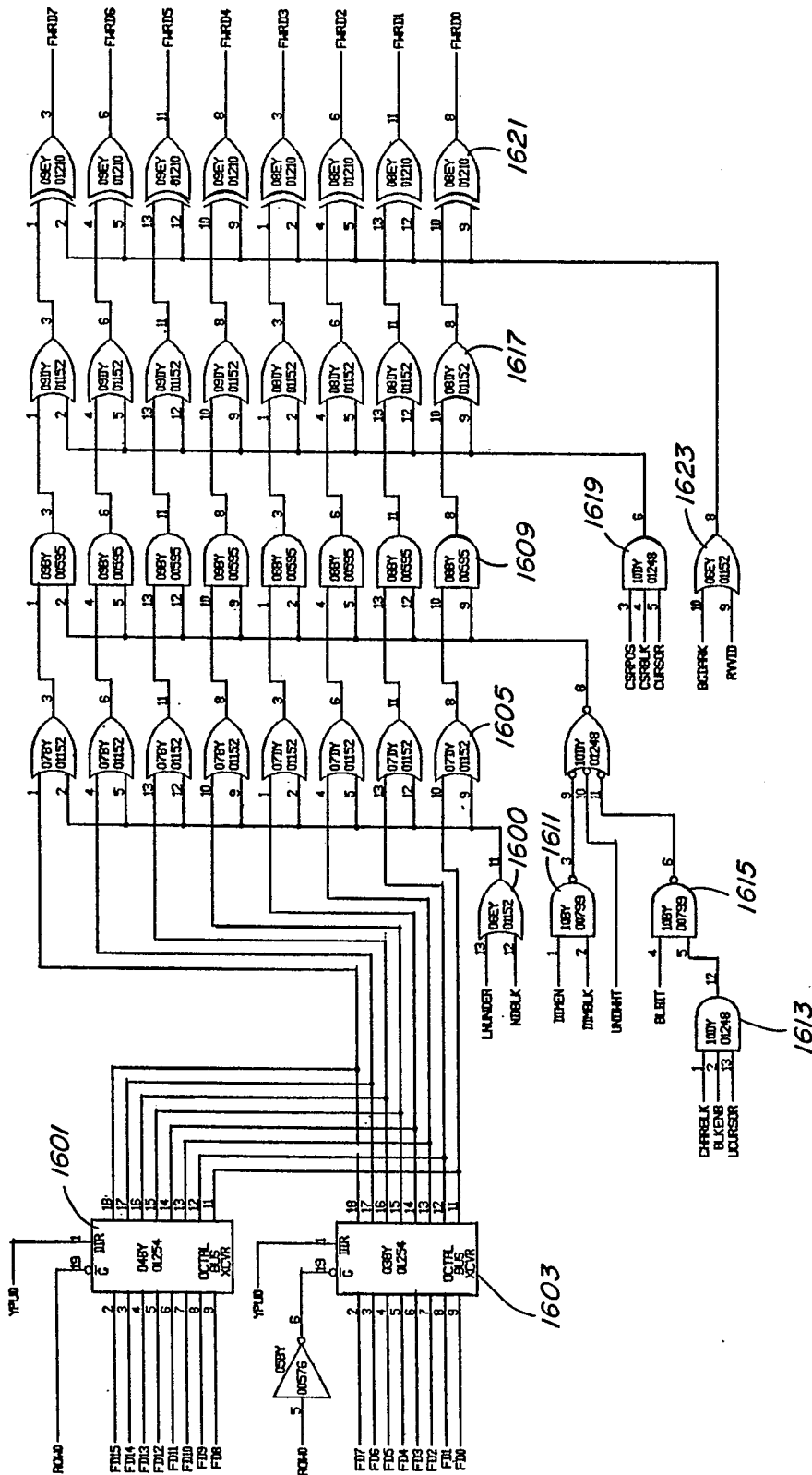


FIG. 16

39/55

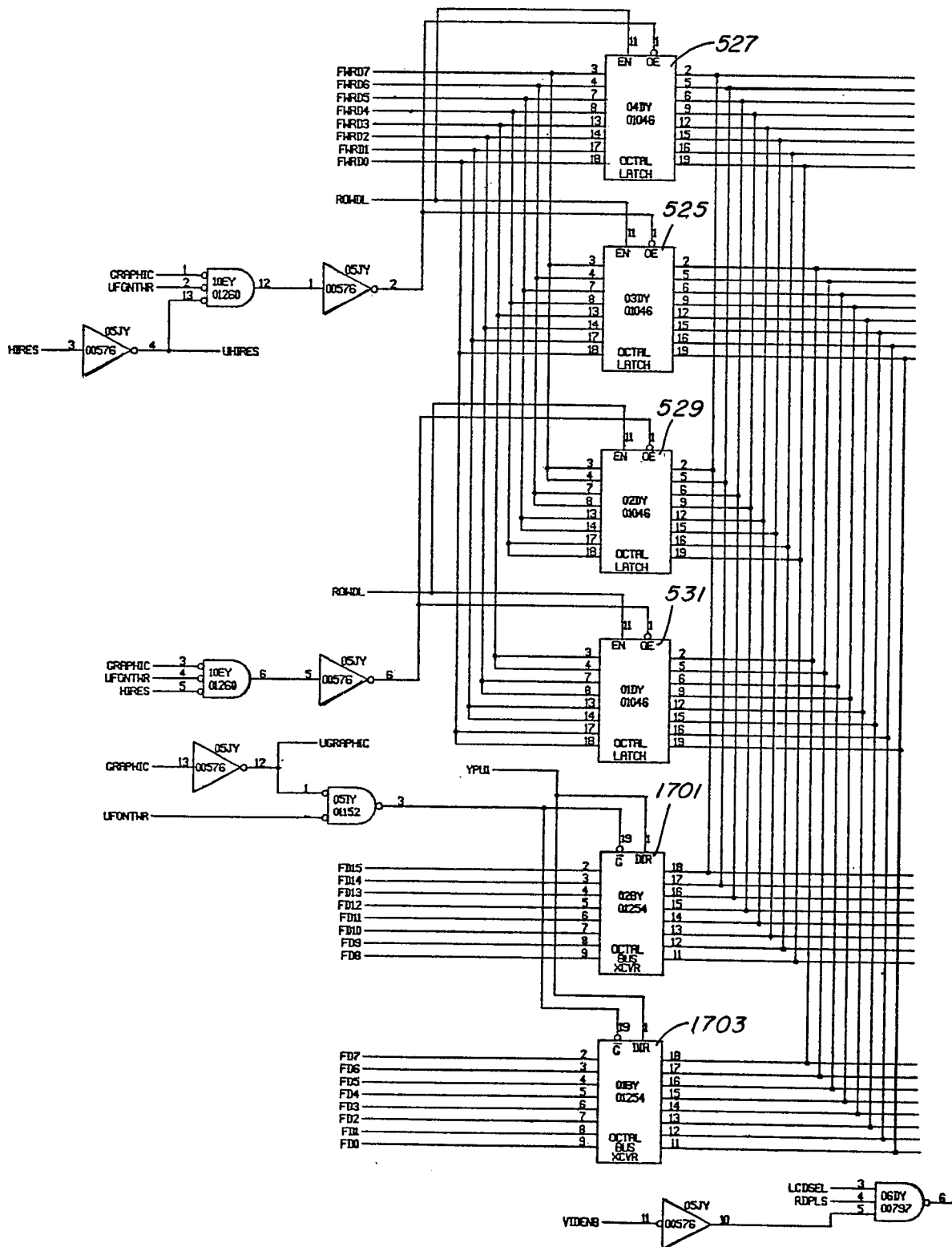


FIG. 17 SHEET 1 OF 2

0175499

40/55

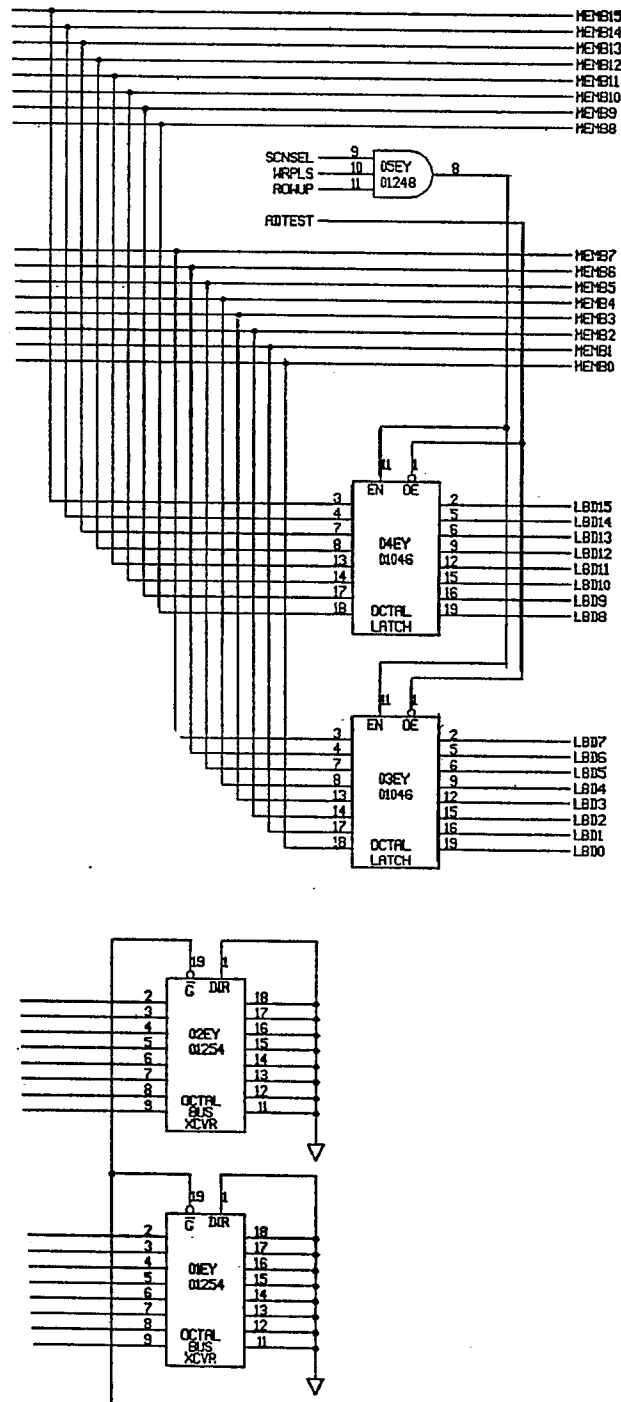


FIG. 17 SHEET 2 OF 2

44/55

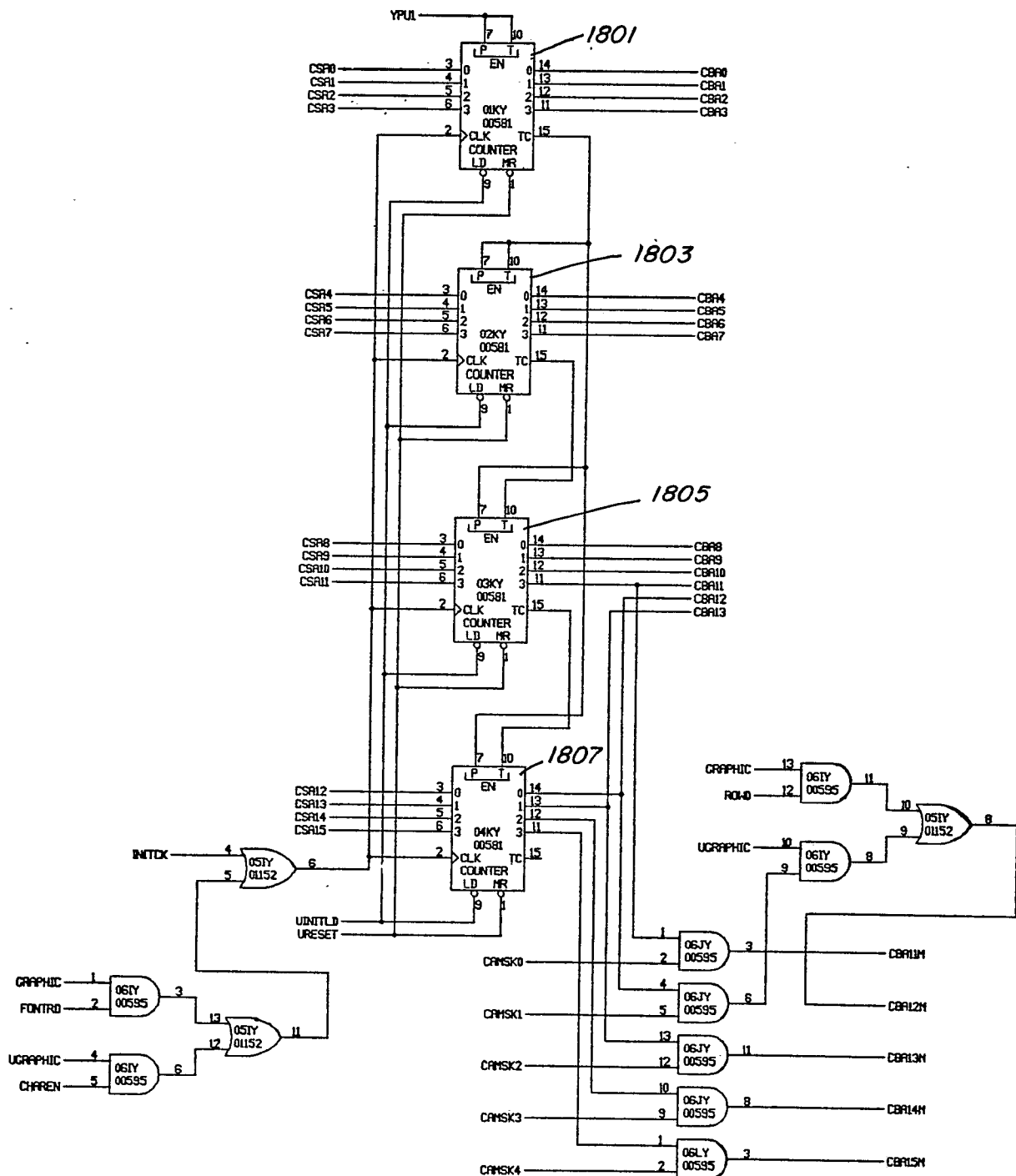


FIG. 18 SHEET 1 OF 2

42/55

CURSOR POSITION DETECTOR

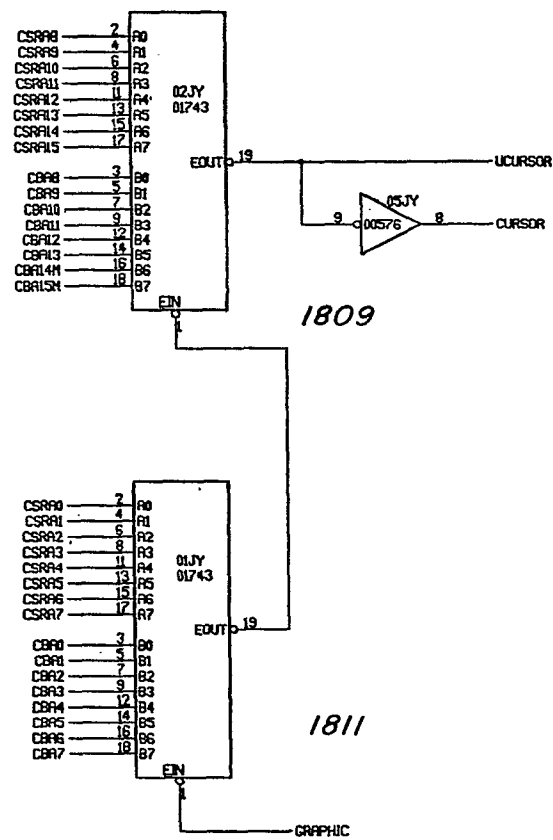


FIG. 18 SHEET 2 OF 2

43/55

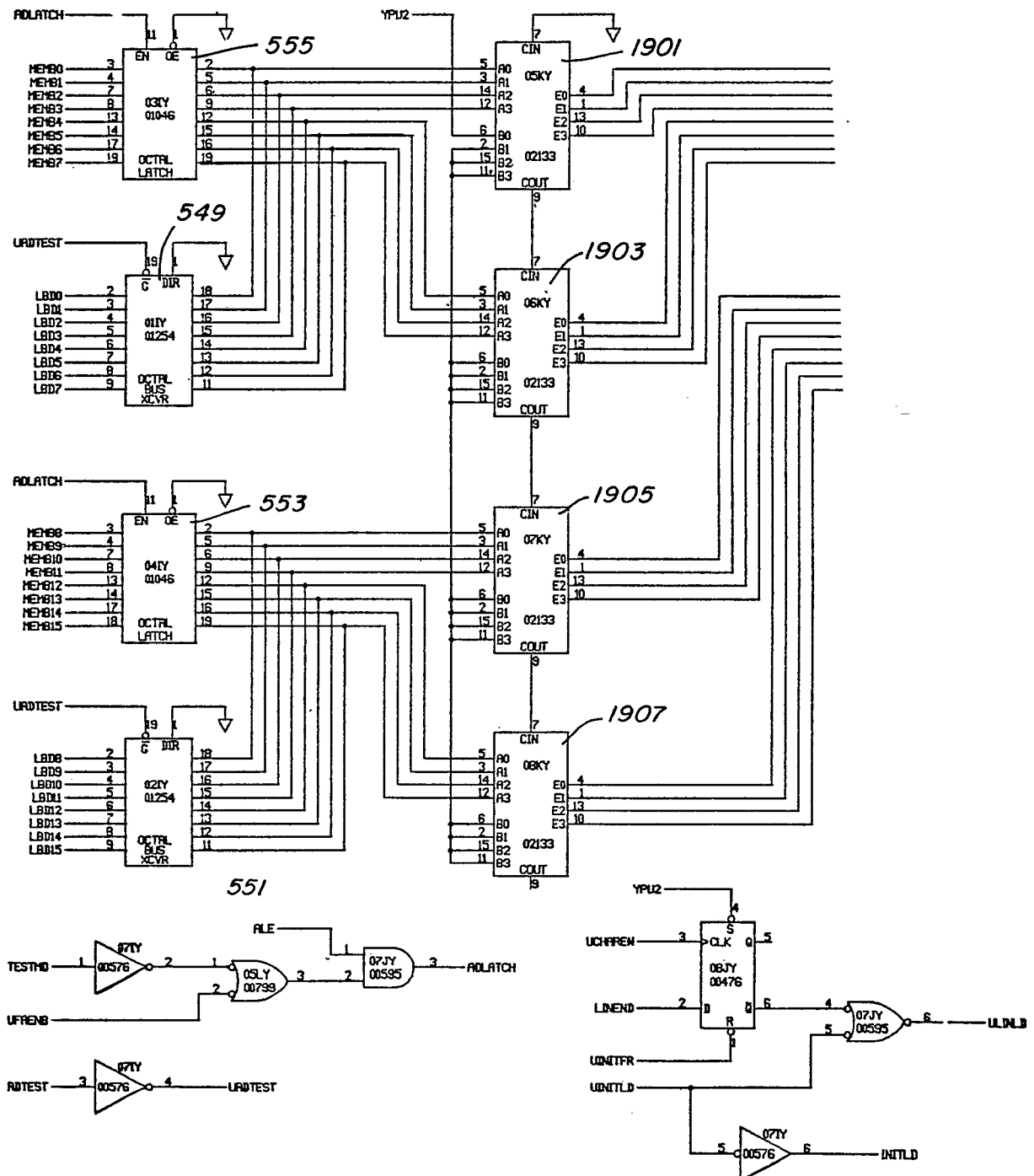


FIG. 19 SHEET 1 OF 3

45/55

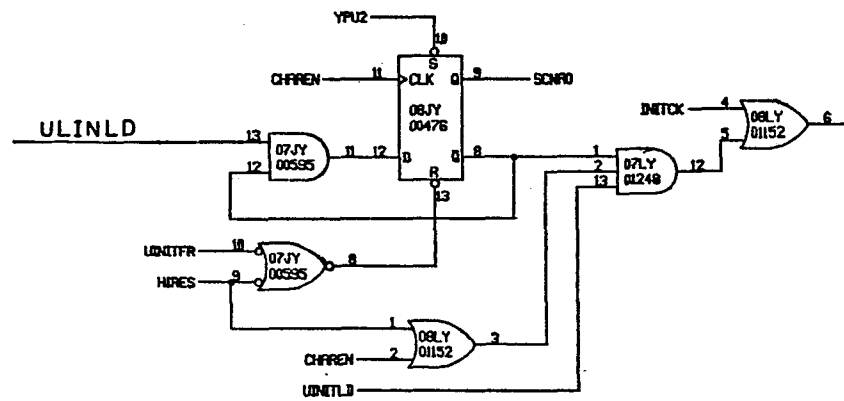


FIG. 19 SHEET 3 OF 3

47/55

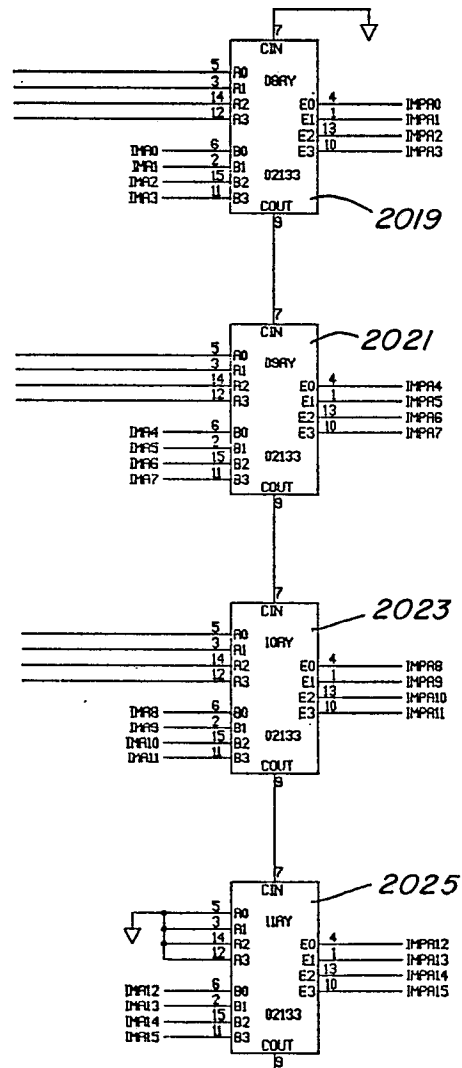


FIG. 20 SHEET 2 OF 2

48/55

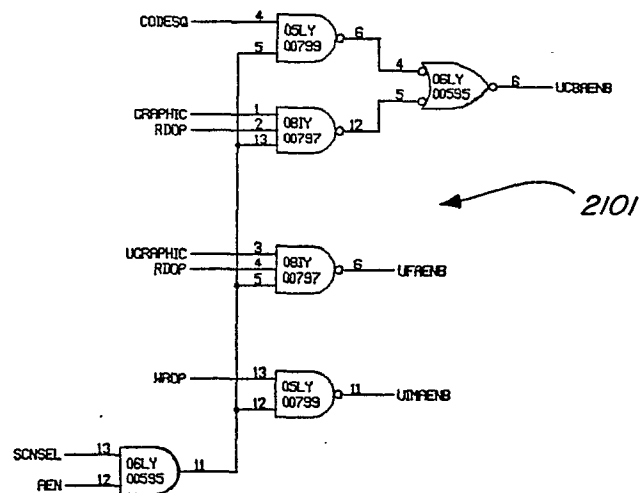


FIG. 21 SHEET 1 OF 2

49/55

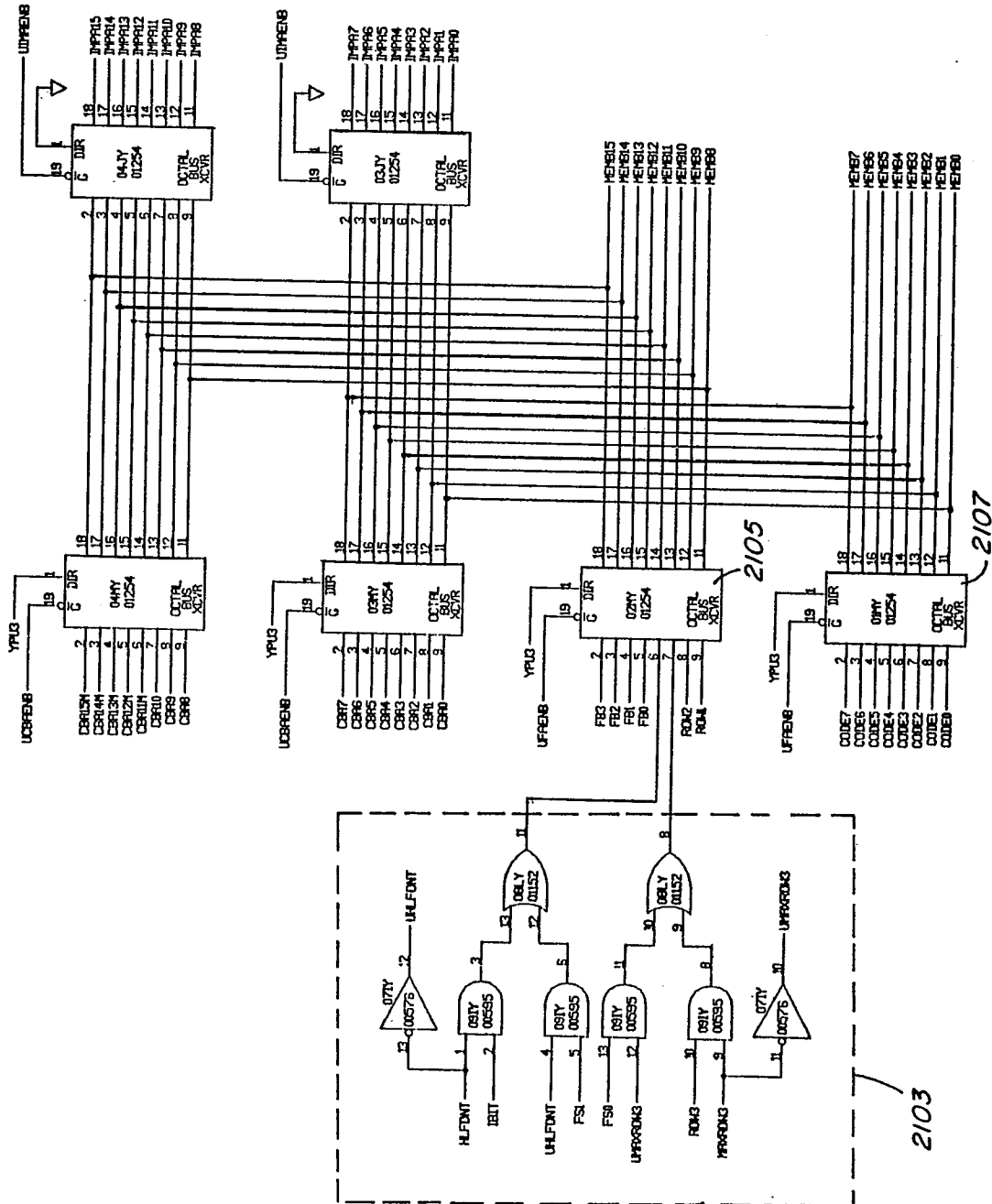


FIG. 21 SHEET 2 OF 2

50/55

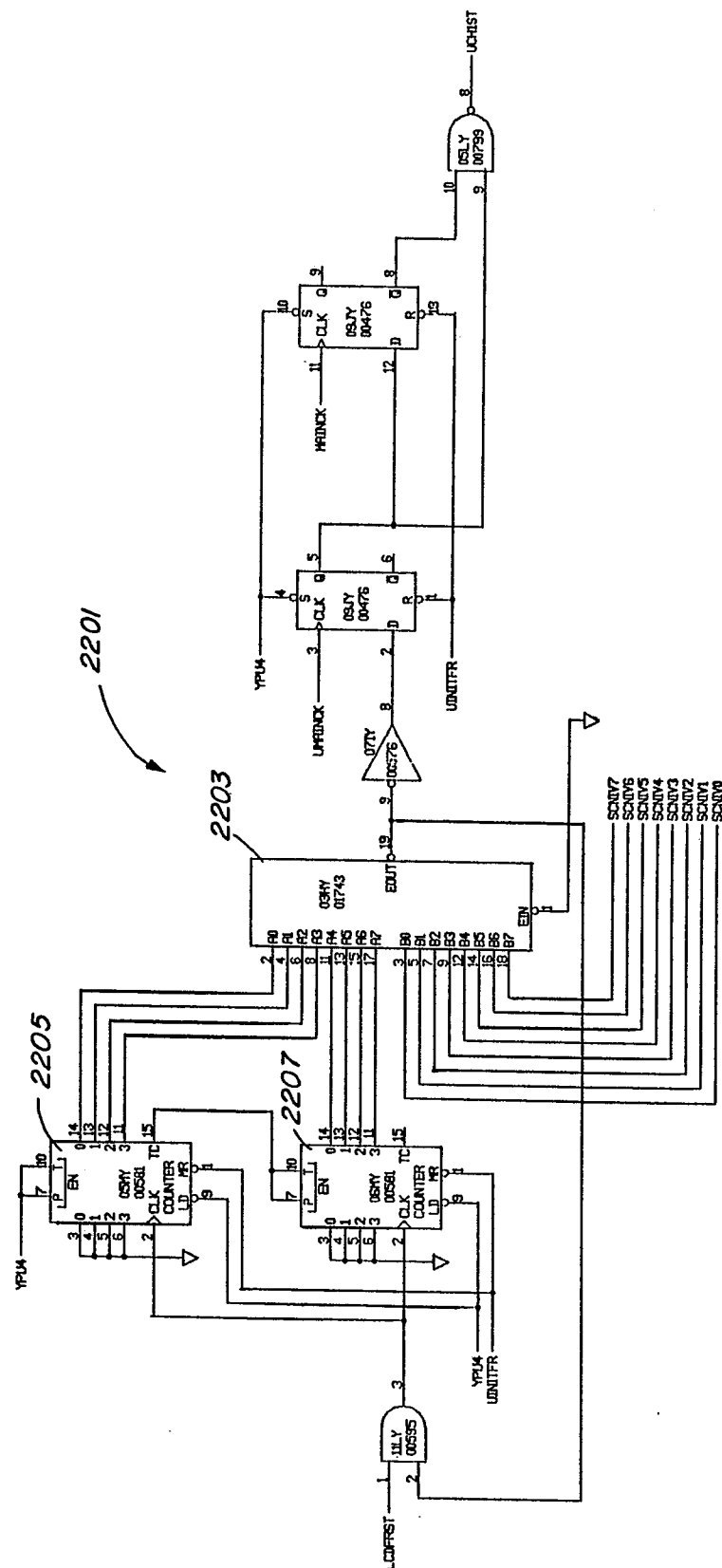


FIG. 22 SHEET 1 OF 2

51/55

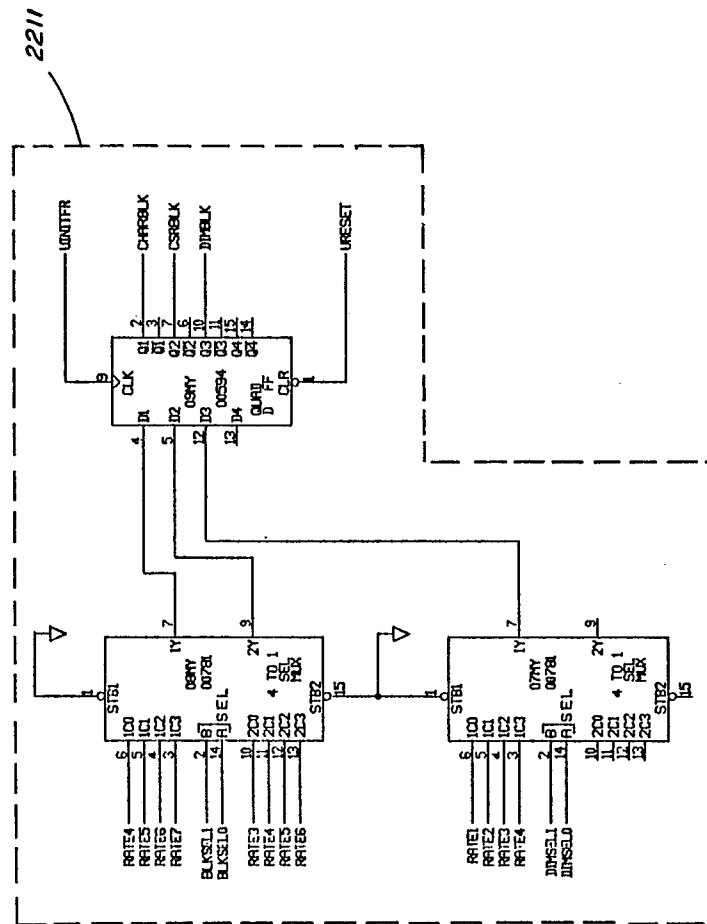
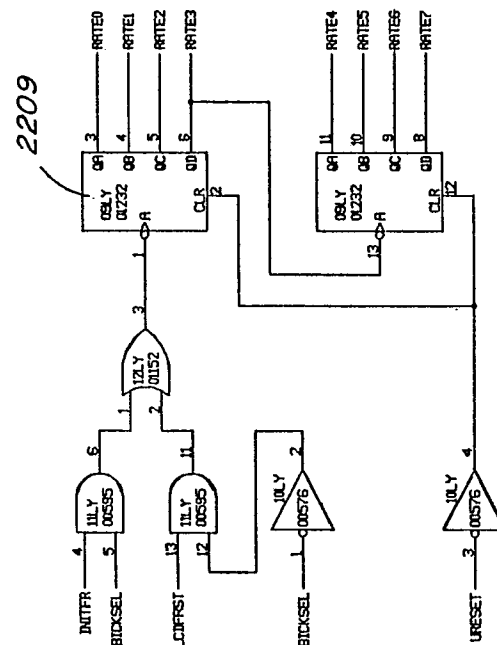


FIG. 22

52/55

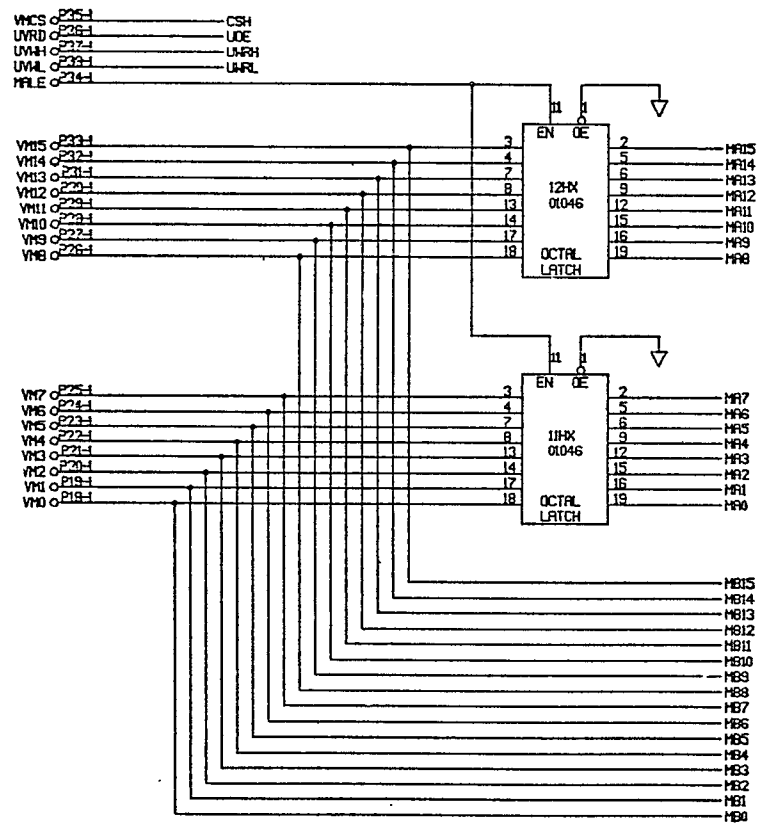
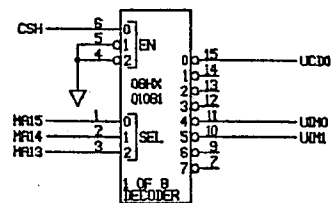


FIG. 23 SHEET 1 OF 2

53/55

213

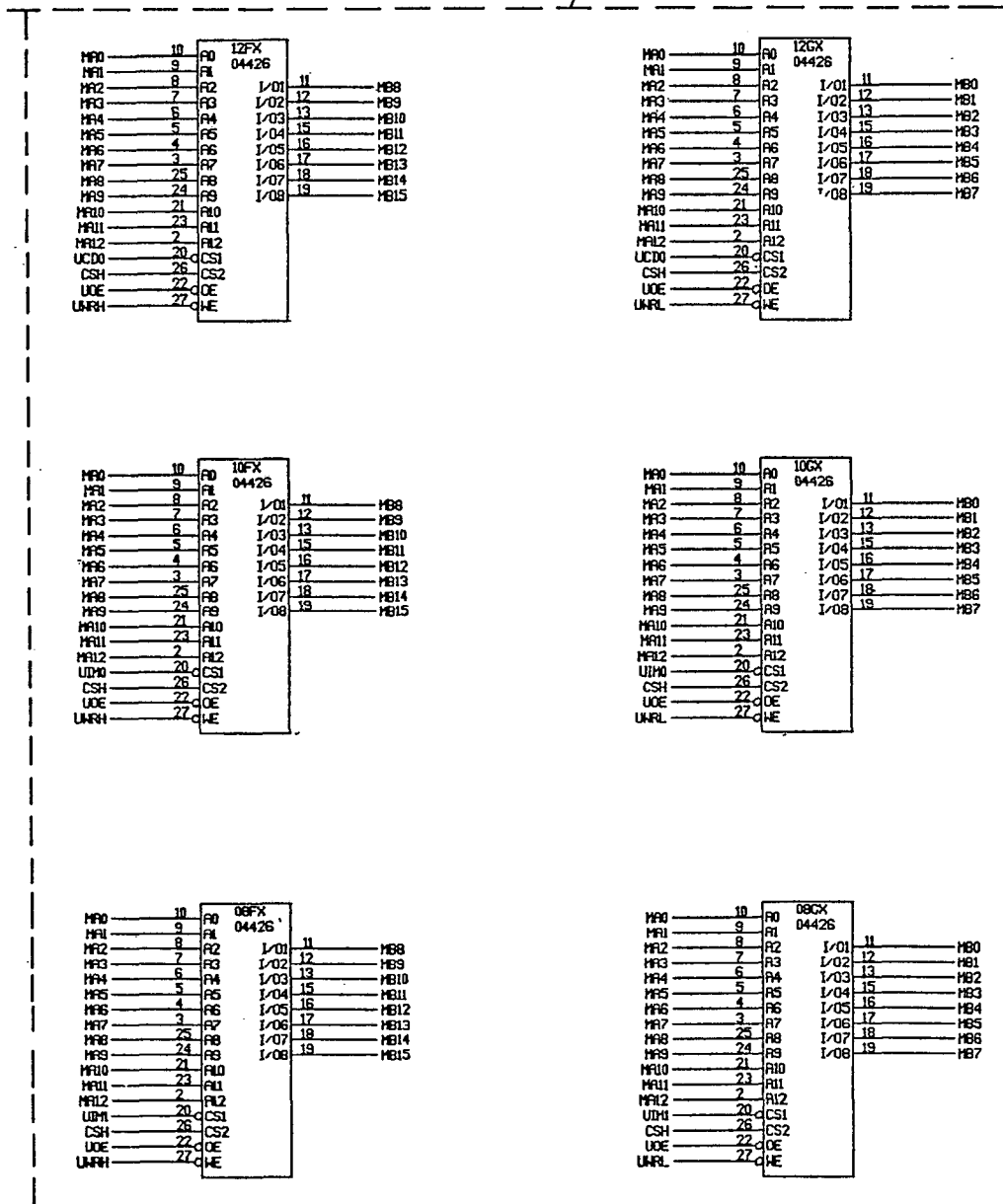


FIG. 23 SHEET 2 OF 2

54/55

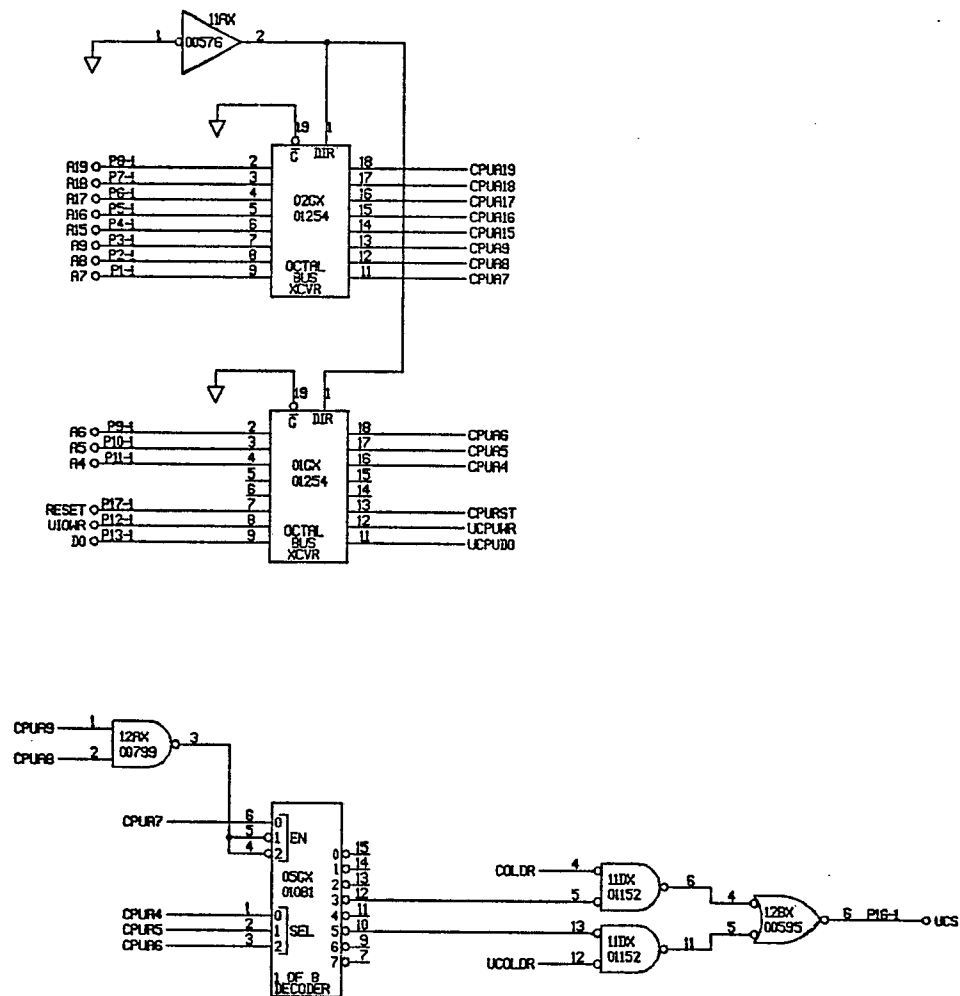


FIG. 24 SHEET 1 OF 2

55/55

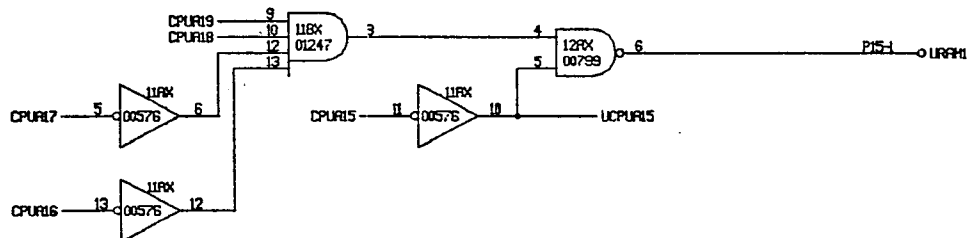
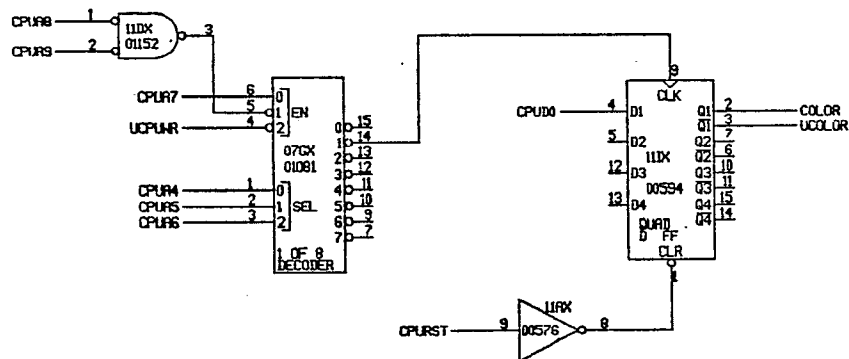


FIG. 24 SHEET 2 OF 2