Ipos copy

A System for Horizontal Scrolling of Display Data

Field of the Invention

This invention relates to a system for horizontal scrolling of display data for a Liquid Crystal Display (LCD), and more particularly, though not exclusively, to a system of horizontally scrolling graphical characters, for example, Chinese characters, on a LCD.

Background of the Invention

Currently, many pagers, for example, Chinese character pagers, display new messages page by page each time the user presses the read-massage button. It would be more user-friendly and desirable to scroll the display content horizontally to output new messages or new information. However, such scrolling is usually controlled by the Microcontroller Unit (MCU), rather than by the LCD driver. Utilising the MCU for this task requires the whole Random Access Memory (RAM) to be rewritten in the LCD driver even when the display is to be scrolled by only one column. The RAM stores bitmap data indicative of which pixels on the LCD panel are to be blank (white) and which are to be nonblank (black). This means that the workload of the MCU is especially heavy since it usually takes many data-write cycles to rewrite the whole RAM, especially when the number of columns of the RAM (and also of the LCD) is large.

To reduce the workload of the MCU, the horizontal scrolling feature can be built into the LCD driver such that it scrolls the display content horizontally during display according to a value provided by the MCU (i.e. the horizontal scrolling value, such value being between 0 and N-1 where N is the number of columns of the LCD). This permits the MCU to go into standby mode (which usually consumes less power) during scrolling.

To build the horizontal scrolling feature into the LCD driver, one can use one of two conventional methods:

1) Using pure combinational logic gates to implement the horizontal scrolling. However in this implementation, the size of the logic block causes a problem since it requires the number of input pins to be

equal to the sum of the number of columns of display (N) and the logarithm of the number of columns with base 2 (i.e. log2(N)), while the number of output pins of the logic block is equal to the number of columns of display (N). For example, if the LCD has 128 columns, then the pure combinational logic block has 128 + log2(128) = 135 input pins and 128 output pins. Such a logic block will consume a large amount of silicon area in the LCD driver, which make it impractical.

2) Using a pure shift register, such that it needs N number of clock cycles to shift 1 row of display content before display to the LCD panel. This method uses minimum circuitry. However, to rewrite the complete bitmap data of the display stored in the RAM for each column scroll would require a large number of clock pulses. Unfortunately, radio frequency interference issues will usually not allow a very high frequency clock signal in the LCD drivers for communication devices such as pagers. So again, this method is impractical to implement.

Brief Summary of the Invention

The present invention therefore seeks to provide a system for horizontal scrolling of display data which overcomes, or at least reduces the above-mentioned problems of the prior art.

Accordingly, the invention provides a system for horizontal scrolling of display data for a Liquid Crystal Display (LCD), the system comprising a display controller, a memory coupled to the display controller and having a plurality of memory rows for storing data representing information to be displayed on LCD, a horizontal scroll shift register coupled to the memory for receiving the data from the memory and for shifting the data horizontally selectively by one of a higher or a lower value to provide horizontally shifted data, a data buffer coupled to the display controller and to the horizontal scroll shift register for receiving the horizontally shifted data and for transferring the horizontally shifted data to a column driver for driving the columns of the LCD to display the information, a horizontal scroll controller for receiving a scroll value indicating the amount by which the information displayed on the LCD is to be scrolled horizontally, the horizontal scroll controller comprising a first counter for receiving the most significant bits (MSB) of the

scroll value and a second counter for receiving the least significant bits (LSB) of the scroll value, the first counter counting down until it reach zero and providing a first output signal to the horizontal scroll shift register while it is counting down indicating the horizontal shift of the data by the higher value and the second counter counting down after the first counter reaches zero until it reaches zero and providing a second output signal to the horizontal scroll shift register while it is counting down indicating the horizontal shift of the data by the lower value.

In a preferred embodiment, the horizontal scroll shift register comprises a row of shift bits, each shift bit comprising a selector and a memory device, each particular shift bit being coupled to receive a first data input from the memory, a second data input from the output of a shift bit previous to the particular shift bit in the row by the lower value and a third data input from the output of a shift bit previous to the particular shift bit in the row by the higher value, the selector of each shift bit having three control inputs coupled respectively to receive a control signal from the display controller and to receive the first and second output signals from the first and second counters for selectively loading data from the memory or from the shift bit previous to the particular shift bit in the row by the lower value or from the shift bit previous to the particular shift bit in the row by the higher value into the memory device according to the received control signal and first and second output signals, respectively, to store the data as the horizontally shifted data.

Preferably, each selector comprises a multiplexer for receiving the first, second and third data inputs and selectively transferring them to the memory device according to the three control inputs.

Each memory device preferably comprises a flip-flop. In one embodiment, each of the first and second counters includes an OR-gate having a plurality of inputs, each coupled to one bit of the counter and an output, the respective outputs providing the first and second output signals of the first and second counters. Preferably, the lower value is "one" and the higher value is "eight".

Brief Description of the Drawings

One embodiment of the invention will now be more fully described, by way of example, with reference to the drawings, of which:

FIG. 1 shows a functional block diagram of one embodiment of a horizontal scrolling mechanism in an LCD driver;

FIG.2 shows, in more detail, a block diagram of a Horizontal Scrolling Control used in the mechanism of FIG.1; and

FIG.3 shows, in more detail, a block diagram of a Horizontal Scrolling Shift Register used in the mechanism of FIG.1.

Detailed Description of the Drawings

Thus, this embodiment of a horizontal scrolling mechanism for a LCD driver according to the present invention uses the shift register approach to implement the horizontal scrolling feature. However, for each clock cycle, the bits can be shifted in this horizontal shift register by a selectable value. In the present example, the selectable values are chosen to shift by 1 and shift by 8. The particular shift values chosen are actually very flexible and will be discussed further below. By choosing these two particular shift values, the number of clock cycles required to shift N bits horizontally (where N is the number of columns of the LCD and also the length (in bits) of the shift register) is substantially reduced.

In an example, if N is 128, and the horizontal scrolling value HSV is 127 (i.e. we need to shift the 128 bits in the shift register 127 times before displaying on the LCD panel), then:

i) for a traditional shift register, 127 clock cycles are required to finish the horizontal scrolling

ii) for the new shift register, the display content is first shifted by the shift value of 8 fifteen times to scroll 8 x 15 =120 columns, and then shifted by the shift value of 1 seven times to scroll 1 x 7 = 7 columns, thereby providing the 120 + 7 = 127 scrolling value. Thus, the total number of clock cycles required is just 15 + 7 = 22 rather than 127.

To implement this mechanism, as shown in FIG. 1, an external MCU 20 sends a command indicative of the required horizontal scrolling value HSV via a communication link 23 to a command decoder 22, forming part of an LCD driver 21. The horizontal scrolling value HSV is stored in the command decoder 22 ,

where it is transformed into binary form and transferred on parallel lines 78 to a horizontal scrolling controller 24.

A display controller 25 provides a clock signal RAMCLK via line 29 to a RAM 26, which stores the bitmap data indicative of the information to be displayed on the LCD panel 27. A complete row of bitmap data from RAM 26 (which row is N bits in length) is made available on parallel lines 77 to parallel inputs of a horizontal scrolling shift register 28. At the same time the display controller 25 sends an enable signal LOADRAM via line 30 and a clock signal SCLK via line 32 to the horizontal scrolling shift register 28 to enable the N bits of RAM row data to be written to the to the horizontal scrolling shift register 28. The display controller 25 also provides an enable signal LOAD to be sent via line 33 to the horizontal scrolling controller 24 to enable it to load the binary horizontal scrolling value HSV from the command decoder.

As shown in FIG. 2, the horizontal scrolling controller 24 includes a 7-bit counter 34, which is divided into two parts, a Most Significant Bits (MSB) Counter 35, which is 4 bits 46, 47, 48, 49 in length in this embodiment, and a Least Significant Bits (LSB) Counter 36, which is 3 bits 50, 51, 52 in length in this embodiment. Both the MSB counter 35 and the LSB counter 36 are count-down counters, which count down according to received clock signal pulses. The MSB counter 35 receives the clock signal pulses CLKM on line 38 from an AND-gate 39 and the LSB counter 36 receives the clock signal pulses CLKL on line 40 from an AND-gate 41. The AND-gate 39 receives the clock signal SCLK, received by the horizontal scrolling controller 24 on line 37 from display controller 25, via line 42 at one input and its other input is coupled via line 43 to receive a Shift-By-8 signal SHBY8 output from the horizontal scrolling controller 24, as will be more fully explained below. The AND-gate 41 also receives the clock signal SCLK via line 44 at one input and its other input is coupled via line 45 to receive a Shift-By-1 signal SHBY1 output from the horizontal scrolling controller 24, as will also be more fully explained below.

Once the binary horizontal scrolling value HSV is loaded to the MSB and LSB counters after the enable signal LOAD is received on line 33, the MSB counter 35 will first start to count down according

to the clock pulses CLKM received on line 38. As will be explained below, the LSB counter 36 does not receive clock pulses CLKL and hence does not count until the MSB counter 35 has finished counting. Taking the example given above of a non-zero horizontal scrolling value HSV of 127, in decimal, or 1111111 in binary, then the four bits 46 - 49 of the MSB counter 35 count down from 1111. Each bit is coupled, via lines 53, 54, 55, 56 respectively, to an input of a four input OR-gate 57. The output of the OR-gate 57 is coupled to an input of an AND-gate 58, whose other input is coupled to receive the enable signal LOAD via an inverter 59. Thus, when the counter 34 is not loading a new HSV and the MSB counter 35 has a non-zero count value, the output of the AND-gate 58 will be high to cause the Shift-By-8 signal SHBY8 output from the horizontal scrolling controller 24 to go high. This Shift-By-8 signal SHBY8 is passed via line 60 to the horizontal scrolling shift register 28 to enable the shift-by-8 operation.

As explained above, this Shift-By-8 signal SHBY8 is also passed via line 43 to the AND-gate 39 to enable the MSB Counter 35 to receive the clock pulses CLKM to count down. During this count down operation, the Shift-By-8 signal SHBY8 remains high until the MSB Counter 35 content reaches zero.

Furthermore, the output of the AND-gate 58 is also passed to an inverting input of a three input AND-gate 61, whose output provides the Shift-By-1 signal SHBY1 output of the horizontal scrolling shift register 28. As will be apparent, the high state of the Shift-By-8 signal SHBY8 will cause the Shift-By-1 signal SHBY1 to go low through the logic operation of the 3-input AND-gate 61. This low state of the Shift-By-1 signal SHBY1 inhibits the operation of the LSB Counter 36 by virtue of the AND-gate 41 whose output provides the clock pulses CLKL to the LSB Counter 36.

When the MSB counter 35 reaches zero, the Shift-By-8 signal SHBY8 turns low, thus stopping the clock pulses CLKM and making the MSB counter 35 stop further counting to stay at zero. The Shift-By-8 signal SHBY8 turning low also stops the shift-by-8 operation of the horizontal scrolling shift register 28.

Similarly, each bit 50, 51, 52 of the LSB counter 36 is coupled, via lines 63, 64, 65 respectively, to an input of a three input OR-gate

62. The output of the OR-gate 62 is coupled to an input of the AND-gate 61, whose other inputs are coupled to receive the enable signal LOAD via the inverter 59 and the inverse of the Shift-By-8 signal SHBY8, as explained above. Thus, if the content of the LSB counter 36 is not zero, the OR-gate 62 will have an output which is high, which in turn causes the AND-gate 61 to output a high state for the Shift-By-1 signal SHBY1. This Shift-By-1 signal SHBY1 is also coupled to the horizontal scrolling shift register 28 via line 66 to enable the shift-by-1 operation. At the same time, by means of the AND-gate 41, clock pulses CLKL will be provided to the LSB counter 36, which will start counting down. Similarly to the MSB counter 35, when the LSB counter 36 reaches zero, the Shift-By-1 signal SHBY1 turns low and stops further counting down of the LSB counter 36 and stops the shift-by-1 operation of the horizontal scrolling shift register 28.

At this time, the horizontally shifted data row is stored in the horizontal scrolling shift register 28 and is latched in display data buffer 74 via parallel lines 31 upon receipt by the display data buffer 74 of a latch signal LATCHCLK provided by display controller 25 via line 75. The display data buffer 74 latches the outputs (N bits in length) of the horizontal scrolling shift register 28 and passes the data to the LCD panel 27 via a LCD Column Driver 76. The above operation is repeated for the next display row.

It will be clear that the two AND-gates 58 and 61 (which output the Shift-By-8 signal SHBY8 and the Shift-By-1 signal SHBY1), each have one input coupled to the inverse of the enable signal LOAD. This is to prevent any count down operation by the two counters 35 and 36 and any shift-by-1 or shift-by-8 operations of the horizontal scrolling shift register 28 while data is from the RAM 26 is being loaded to the horizontal scrolling shift register 28 and while the two counters 35 and 36 are being loaded with the horizontal scroll value HSV from the command decoder 22.

As an example to demonstrate the operation of the horizontal scrolling controller 24, assume (again) that the HSV is 127 (i.e. 1111111 in binary) and that N (number of columns) is 128. Then the values of the various counters ans signals will be:

MSB Counter	LSB Counter	SHBY8	SHBY1	LOAD	CLOCK CYCLES
1 1 1 1	1 1 1	0	0	1	1
1 1 1 0	1 1 1	1	0	0	2
1 1 0 1	1 1 1	1	0	0	3
1 1 0 0	1 1 1	1	0	0	4
1 0 1 1	1 1 1	1	0	0	5
1 0 1 0	1 1 1	1	0	0	6
1 0 0 1	1 1 1	1	0	0	7
1 0 0 0	1 1 1	1	0	0	8
0 1 1 1	1 1 1	1	0	0	9
0 1 1 0	1 1 1	1	0	0	10
0 1 0 1	1 1 1	1	0	0	11
0 1 0 0	1 1 1	1	0	0	12
0 0 1 1	1 1 1	1	0	0	13
0 0 1 0	1 1 1	1	0	0	14
0 0 0 1	1 1 1	1	0	0	15
0 0 0 0	1 1 1	0	1	0	16
0 0 0 0	1 1 0	0	1	0	17
0 0 0 0	1 0 1	0	1	0	18
0 0 0 0	1 0 0	0	1	0	19
0 0 0 0	0 1 1	0	1	0	20
0 0 0 0	0 1 0	0	1	0	21
0 0 0 0	0 0 1	0	1	0	22
0 0 0 0	0 0 0	0	0	0	23

NEXT ROW REPEAT THE OPERATION

Turning now to FIG. 3, there is shown in more detail the horizontal scrolling shift register 28. The horizontal scrolling shift register 28 consists of N bits, of which only 4 bits, that is the bit (i) 68, bit (i-1) 67, bit (i+1) 69 and bit (i+8) 70 are shown. Each bit consists of a 3-way selector 71 having three data inputs A, B and C and three select inputs X, Y and Z and an output O, which is coupled

to a D-input D of a D-type flip-flop 72, which also has a clock input CK and an output Q.

The first data input A of each 3-way selector 71 is connected to the output Q of the D-type flip-flop 72 of the previous bit. The second data input B of each 3-way selector 71 is connected to the output Q of the D-type flip-flop 72 of the eighth previous bit. For example, the output Q of the flip-flop 72 of bit (i) 69 is connected to data input B of bit (i+8) 70, as shown in FIG. 3. The third data input C of each 3-way selector 71 is coupled to the respective parallel input of the horizontal scrolling shift register 28 to receive the respective bit of the bitmap row being output from the RAM 26.

The first select input X of each 3-way selector 71 is connected to receive the Shift-By-1 signal SHBY1 from line 66. The second select input Y of each 3-way selector 71 is connected to receive the Shift-By-8 signal SHBY8 from line 60. The third select input Z of each 3-way selector 71 is connected to receive the enable signal LOADRAM from the display controller 25 via line 30.

Each selector 71 selectively transfers the signal on one of the data inputs A, B and C to the output O depending on the signals on the select inputs X, Y and Z. Thus, the signal on the first data input A is selected when the Shift-By-1 signal SHBY1, which corresponds to the shift-by-1 operation, is high; the signal on the second data input B is connected to the output O when the Shift-By-8 signal SHBY8, which corresponds to the shift-by-8 operation, is high; the third data input C of the selector 71 is selected when the enable signal LOADRAM, which corresponds to the load-data-from-RAM operation, is high.

The Shift-By-1 signal SHBY1 from line 66, the Shift-By-8 signal SHBY8 from line 60 and the enable signal LOADRAM from line 30 are each respectively coupled to one input of a four-input OR-gate 73, whose other input is coupled to receive the clock signal SCLK via line 32. The output of the OR-gate 73 is coupled to the clock input CK of each flip-flop 72. The operation of the OR-gate 73 is such as to make sure there will be clock pulses (derived from the clock signal SCLK) provided to the N flip-flops 72 whenever there is load-data-from-RAM operation, shift-by-8 operation or shift-by-1 operation.

The outputs Q of each of the N flip-flops 72 are coupled to the display data buffer 74 which will latch the data after all the shifting operations have been done. The input of the first bit is connected to the output of the last bit such that the shifting operation will be wrapped-around, i.e. rotating. As described above, the display data buffer 74 then passes the data on parallel lines 79 to the LCD Column Driver 76, which drives the LCD panel 27 via parallel lines 80.

It can therefore be seen that the provision of the horizontal scrolling controller 24 and the horizontal scrolling shift register 28, together with the other parts of the LCD driver 21 implement the horizontal scrolling mechanism in an efficient way (both in silicon area consumption and in number of clock cycles required). It should be pointed out that the choice of the particular values to shift by is very flexible. For example, if N (number of columns of display) is very large, say 256, but the number of clock cycles given for operating the horizontal scrolling is still very limited, the shift-by-8 can be changed to shift-by-16, so that the maximum number of clock cycles required (which occurs when the Horizontal Scroll Value is 255) is:

15 clock cycles for shift-by-16 operation
+ 15 clock cycles for shift-by-1 operation
= 30 clock cycles,

whereas if shift-by-8 operation instead of shift-by-16 operation were still used, then the number of clock cycles required would be:

31 clock cycles for shift-by-9 operation
+ 7 clock cycles for shift-by-1 operation
= 38 clock cycles.

Thus, we can see the advantage in choosing shift-by-16 in the case of a larger value of N. However, the drawback in increasing the value of the shift operation is that the routing area for the horizontal scrolling shift register is increased. Therefore, there will always need to be a compromise between the amount of silicon area required and the speed (clock cycles) for this approach and will need to be carefully chosen according to the number of clock cycles available.

It will be appreciated that although only one particular embodiment of the invention has been described in detail, various modifications and improvements can be made by a person skilled in the art without departing from the scope of the present invention.

Claims

1. A system for horizontal scrolling of display data for a Liquid Crystal Display (LCD), the system comprising:

a display controller;

a memory coupled to the display controller and having a plurality of memory rows for storing data representing information to be displayed on LCD;

a horizontal scroll shift register coupled to the memory for receiving the data from the memory and for shifting the data horizontally selectively by one of a higher or a lower value to provide horizontally shifted data;

a data buffer coupled to the display controller and to the horizontal scroll shift register for receiving the horizontally shifted data and for transferring the horizontally shifted data to a column driver for driving the columns of the LCD to display the information;

a horizontal scroll controller for receiving a scroll value indicating the amount by which the information displayed on the LCD is to be scrolled horizontally, the horizontal scroll controller comprising a first counter for receiving the most significant bits (MSB) of the scroll value and a second counter for receiving the least significant bits (LSB) of the scroll value, the first counter counting down until it reach zero and providing a first output signal to the horizontal scroll shift register while it is counting down indicating the horizontal shift of the data by the higher value and the second counter counting down after the first counter reaches zero until it reaches zero and providing a second output signal to the horizontal scroll shift register while it is counting down indicating the horizontal shift of the data by the lower value.

2. A system according to claim 1, wherein the horizontal scroll shift register comprises a row of shift bits, each shift bit comprising a selector and a memory device, each particular shift bit being coupled to receive a first data input from the memory, a second data input from the output of a shift bit previous to the particular shift bit in the row by the lower value and a third data input from the output of a shift bit previous to the particular shift bit in the row by

the higher value, the selector of each shift bit having three control inputs coupled respectively to receive a control signal from the display controller and to receive the first and second output signals from the first and second counters for selectively loading data from the memory or from the shift bit previous to the particular shift bit in the row by the lower value or from the shift bit previous to the particular shift bit in the row by the higher value into the memory device according to the received control signal and first and second output signals, respectively, to store the data as the horizontally shifted data.

3. A system according to claim 2, wherein each selector comprises a multiplexer for receiving the first, second and third data inputs and selectively transferring them to the memory device according to the three control inputs.

4. A system according to either claim 2 or claim 3, wherein each memory device comprises a flip-flop.

5. A system according to any preceding claim, wherein each of the first and second counters includes an OR-gate having a plurality of inputs, each coupled to one bit of the counter and an output, the respective outputs providing the first and second output signals of the first and second counters.

6. A system according to any preceding claim, wherein the lower value is "one".

7. A system according to any preceding claim, wherein the higher value is "eight".

8. A system for horizontal scrolling of display data for a Liquid Crystal Display (LCD) substantially as hereinbefore described with reference to the drawings.

Abstract

A System for Horizontal Scrolling of Display Data

To enable horizontal scrolling of a message on a LCD panel (27), without requiring the microcontroller (20) to rewrite the whole RAM (26), a horizontal scrolling controller (24) and a horizontal scrolling shift register (28) are provided to allow data to be shifted by a Horizontal Shift Value (HSV) in a number of clock cycles less than that HSV. This is enabled by providing MSB (35) and LSB (36) counters in the horizontal scrolling controller (24) to count down from the HSV, with the MSB counter (35) providing Shift-by-8 signals to the horizontal scrolling shift register (28) while it is counting down and the LSB counter (36) providing Shift-by-1 signals to the horizontal scrolling shift register (28) while it is counting down, so that the data in the horizontal scrolling shift register (28) can be shifted in blocks of 8, and in blocks of 1, to reduce the clock cycles required.

FIG. 1

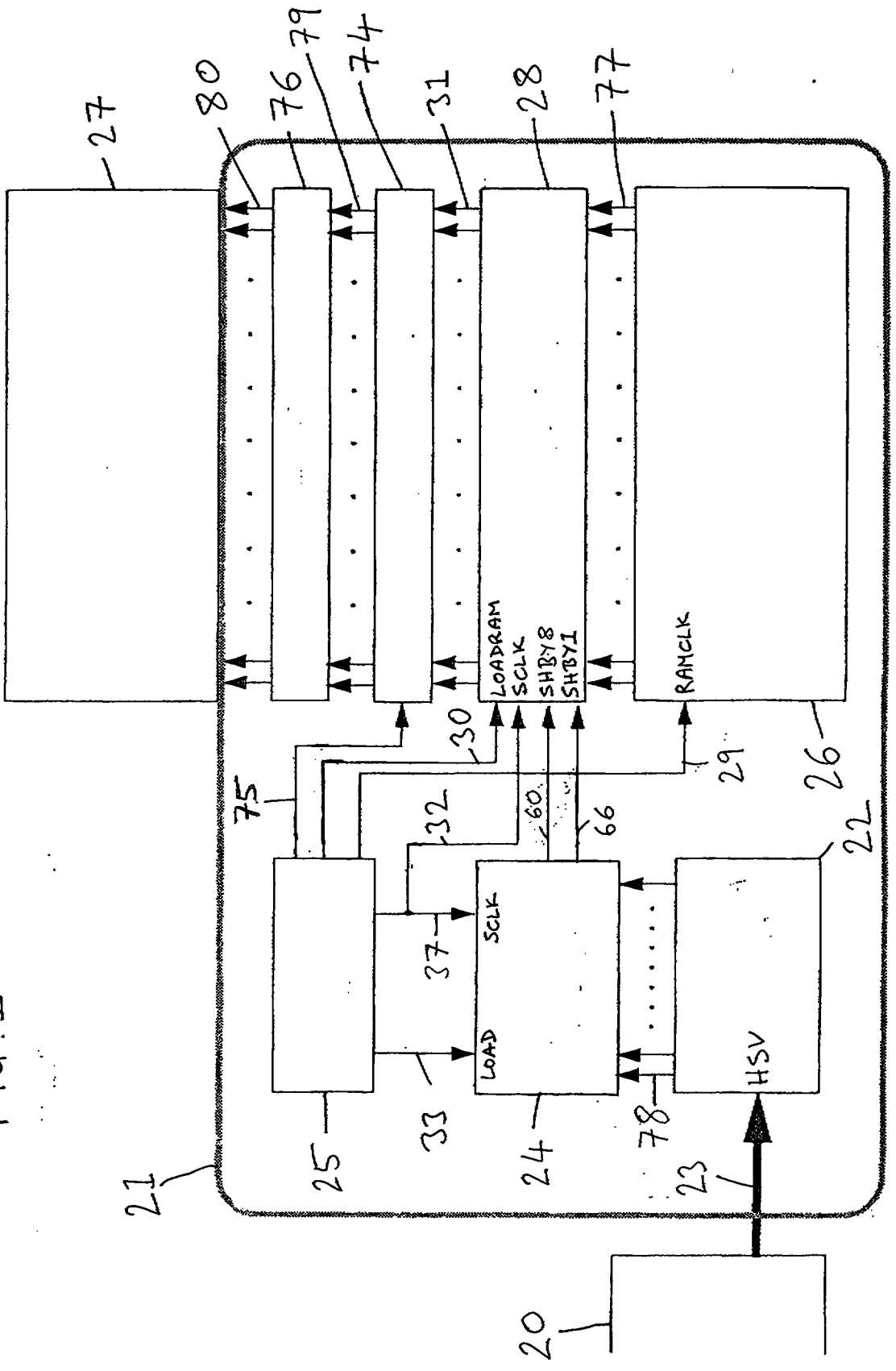

27
80
76
79
74
31
28
77
LOADRAM
SCLK
SHBY8
SHBY1
RAMCLK
30
29
26
75
32
60
66
22
37
SCLK
LOAD
HSV
25
33
24
78
23
21
20

FIG. 1

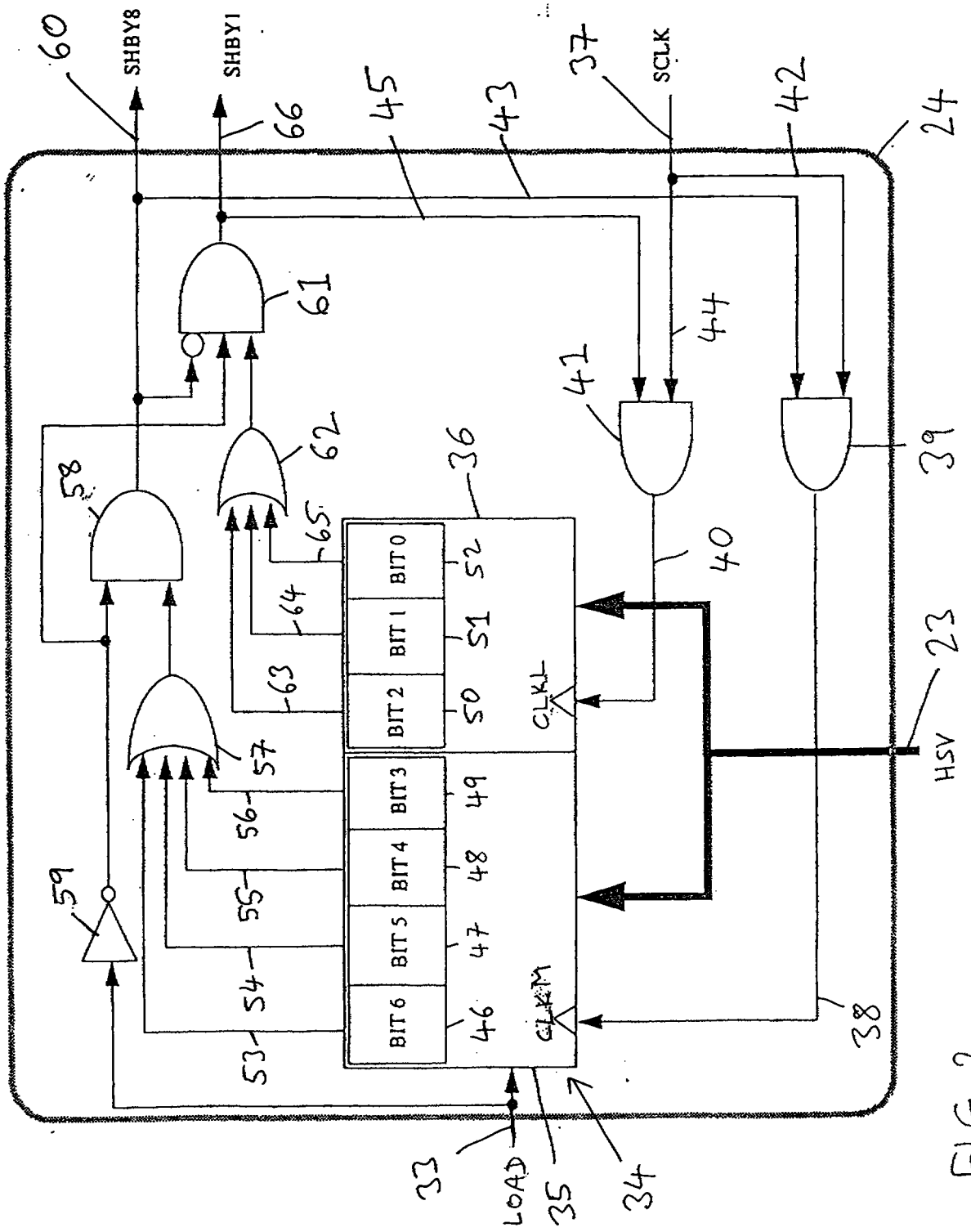

SHBY8
SHBY1
SCLK
BIT 0
BIT 1
BIT 2
BIT 3
BIT 4
BIT 5
BIT 6
CLKL
CLKM
HSV
LOAD
60
66
45
43
37
42
24
61
44
41
62
36
39
58
40
64
63
65
52
51
50
23
57
56
55
54
53
49
48
47
46
59
38
33
35
34

FIG. 2

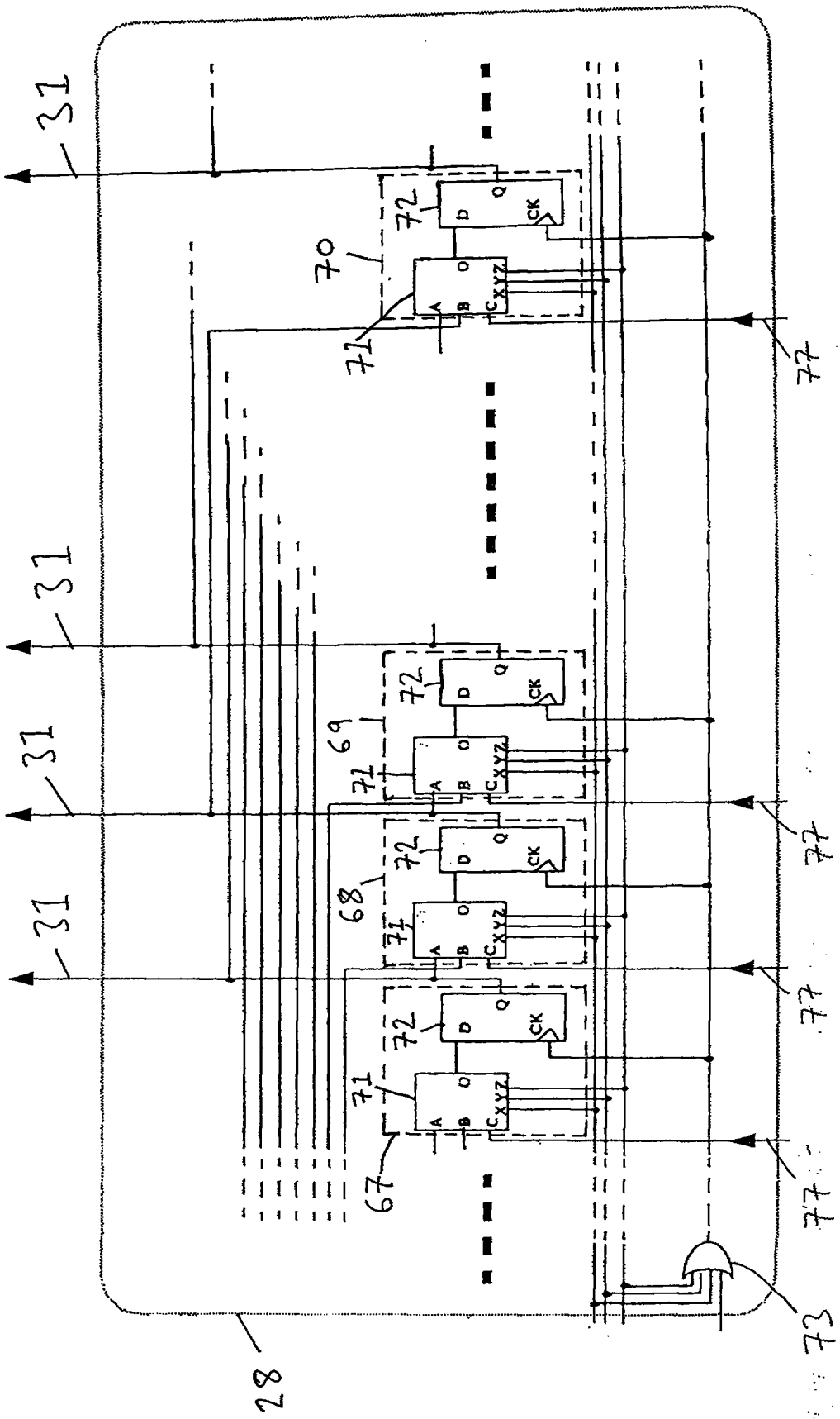


FIG. 3