The MM58348/342/341/248/242/241 Directly Drive Vacuum Fluorescent (VF) Displays

National Semiconductor
Application Note 371
David Stewart
August 1984

1.0 INTRODUCTION

National has produced a family of high voltage display drivers which is specially designed for use with vacuum fluorescent (VF) displays. These circuits are fabricated using a standard metal gate CMOS process which has been extended to allow a maximum operating voltage of 60V, thus enabling the design of bright multiplexed displays. In this way, the advantages of CMOS are retained (low power), while the range of applications for this technology is increased. Many of today's high voltage MOS display drivers require the use of one external resistor per display output, and this leads to a considerable increase in component count and board area. National's display drivers, however, incorporate an on-board pull-down resistor structure which removes these disadvantages.

This application note is intended to demonstrate several ways in which these display drivers can be configured to drive and control a wide range of VF displays. Although particular attention will be given to one specific display, a 32-character alphanumeric display, the design is presented in such a way as to enable easy extrapolation to the system designer's specific applications.

2.0 FUNCTIONAL DESCRIPTION

There are six circuits in this new family of high voltage VF drivers and they can be sub-divided according to maximum operating voltage, number of display outputs, data interfacing requirements and ability to be cascaded. Each of the three circuit configurations is available with maximum operating voltages of 35V (MM583XX) or 60V (MM582XX). Due to the nature of the output stage required to attain high voltage operation of CMOS devices, the drive capabilities of the display output decrease as maximum operating voltage increases. Therefore, to maintain the option of trading off display voltage against drive current, each circuit has a high voltage (reduced drive) version and a low voltage (high drive) version. The three circuit configurations can be identified by the number of display outputs they contain (e.g., 20, 32 or 35 outputs). In all cases, data is entered serially into a 5V internal CMOS shift register. This data is latched to the output either by an external enable control signal (MM58241/341/242/342) or automatically by a leading start bit in the data stream (MM58248/348). *Figure 1* shows how the 6 device numbers correspond to the different circuit configurations and operating voltages.

The MM58348/248 devices use a two control line data input format (data in and clock) which enables the 40-pin part to have 35 display outputs. To load data into the controller, a start bit precedes the 35 data bits. The start bit is a logical ''1'' clocked into the IC by the first clock pulse. Next, 35 data bits are clocked into these parts. The start and data bits are shifted in on the rising edge of the clock. As the data is clocked into the IC, the start bit is shifted down the 35-bit register. On the rising edge of the 36th clock pulse, data is transferred to the display register and the start bit is shifted into the control latch. On the negative edge of the clock, the shift register is cleared. The display register feeds the level shifters that translate 5V CMOS levels to the 35V–60V required by the display. The MM58348/248 devices are not cascadable. Typically, these devices would perform the segment refresh drive in a multiplexed multi-digit system. A functional block diagram is shown in *Figure 2*.

The MM58341/241/342/242 devices use a three control line data input format (data in, clock and enable) and have either 32 or 20 display outputs, as given by *Figure 1*. This configuration sacrifices some outputs to enable cascading, enhance control signal flexibility, and provide brightness control. Here again, data is shifted into the shift register on the rising edge of clock, but no start bit is needed. Instead, the enable signal is taken high to input data to the chip. When the enable is taken low, the contents of the shift register are loaded into the display register. Again, the display register feeds the level translator and display driver outputs.

Each of the MM58241/341 and MM58242/342 devices has a serial data output pin which is connected directly to the last stage's output of the shift register. By connecting data out from one device to the data in pin of another device, and by holding each circuit's enable constantly high, the display drivers can be cascaded. The result is a shift register with a variable number of bits, depending on the mix of circuits used.

The MM58341/241/342/242 devices also have a blanking control input. A logic high on this pin turns all outputs off, while still retaining the display data. If a logic ''0'' is then applied, the display data will return unchanged. Consequently, the brightness of the display is proportional to the duty cycle of this blank signal. A functional block diagram of these devices is shown in *Figure 3*.

		Operating Voltage	
		35V	**60V**
Number of Outputs	20	MM58342	MM58242
	32	MM58341	MM58241
	35	MM58348	MM58248

20 and 32 output drivers use envelope enable data format and may be cascaded.

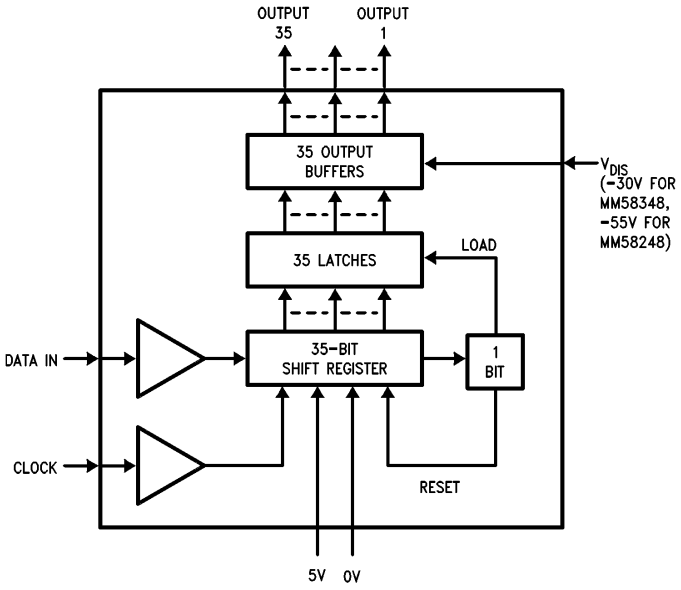
35 output (5 x 7 dot matrix) drivers use start bit data format.

FIGURE 1. The Complete VF Display Driver Family

COPS™ and NSC800™ are trademarks of National Semiconductor Corporation.

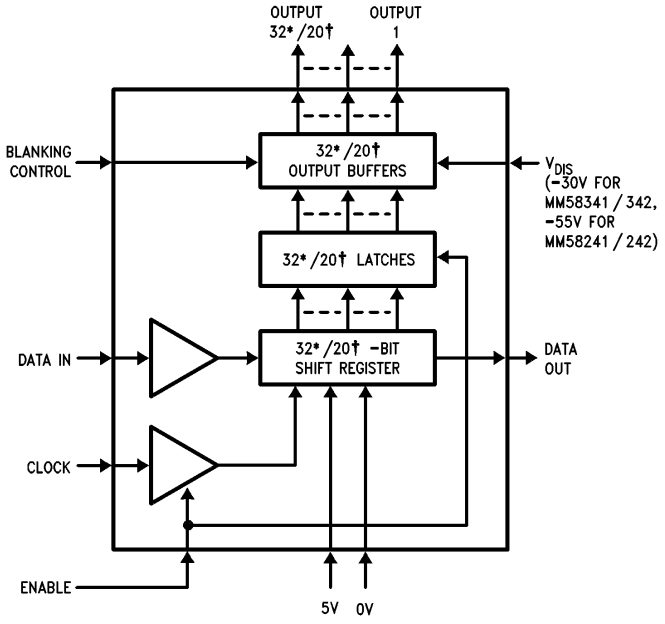
RRD-B30M105/Printed in U. S. A.

BLOCK DIAGRAMS



TL/F/7394–1

FIGURE 2. MM58348/248

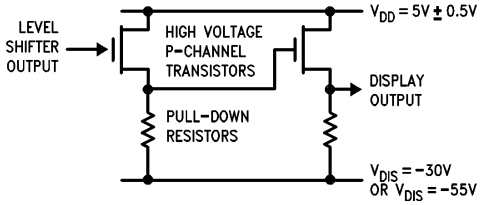


TL/F/7394–2

FIGURE 3. MM58341/241* and MM58342/242†

2.0 FUNCTIONAL DESCRIPTION (Continued)

Referring to the functional block diagrams shown in *Figures 2* and *3*, it is clear that all the internal logic is implemented in standard 5V CMOS. Such signals do not possess sufficient drive for the high voltage output stage, so the data passes through a bank of 15V level shifters to the output section. A schematic of the output stage is shown in *Figure 4*. It can be seen that all these display drivers use a two-stage high voltage structure with active pull-up transistors and passive pull-down resistors to the display voltage. Because resistor pull-downs are used, it is the output switching ''off'' time which is critical for the system design, and this is typically 20 μs for a rail-to-rail voltage swing.



TL/F/7394–3

FIGURE 4. High Voltage Output Structure

3.0 DESIGN CONSIDERATIONS

3.1 The VF Display Configuration

The operation of a VF display is merely an extension of the valve principle, i.e., it is a voltage controlled device. An AC waveform is applied across the filament of the display, and this excitation causes electrons to be emitted. If both the grid and the anode are at a high positive voltage with respect to the cathode, the electrons reach the anode area, which is coated with a fluorescent material. When bombarded by electrons, this material emits light, hence one segment of the display is turned on.

This particular family of display drivers can drive a wide range of VF displays. The simplest case is where each display segment can be directly driven by wiring each output to the display anode. This normally occurs on displays with a small number of digits and segments (e.g., 4 characters of 7 segments) and this can be driven by cascading the drivers until sufficient data bits are available. This display configuration has the advantage of not requiring any refresh (which would be required if a multiplexed configuration were used) but has the disadvantage of needing one wire per segment.

As the size of the display increases, the number of available segments also rises, thus a multiplexing scheme which will reduce the number of display connections is desirable. This is normally achieved by hard wiring all the segments (anodes) of each digit together, then using the grids to select each digit in turn. The correct segment data can then be displayed. Using these techniques necessitates that the display be continually refreshed with each digit of data, even when that data has not changed.

To see the advantage of multiplexing, if a 32 character 5 x 7 dot matrix display is used, a total of 1120 segments is available. For this reason, the display is multiplexed and has 32 grid inputs and 35 segment inputs. The required refreshing task must be accomplished without the detection of flickering by the human eye, i.e., at a rate greater than 50 Hz. (Refresh timing is discussed later.)

Given the aforementioned display pinout and control logic, it is desirable in multiplexed displays to use the MM58341 to control the display grids (digits) and one MM58348 to control the display anodes (segments).

3.2 The Display-Driver Interface

When using the MM58XXX series, no buffering is required between the driver output pins and the VF display. It is necessary only that the driver charge and discharge the display in such a time that the refresh rate outlined in the previous section can be achieved. All the CF drivers have LSTTL compatible inputs, and, as the data source is generally a microprocessor, no special interface requirements exist.

3.3 The Microprocessor-Driver Interface

Typically, the system utilizing these display drivers will have some sort of microprocessor or single chip computer controlling the display. Thus, this processor will control one or more of the display drivers. The drivers have relatively little intelligence, therefore the host processor will be in charge of updating the display drivers and generating refresh timing if needed. The advantage of having minimal intelligence on the drivers themselves is flexibility. Virtually any display size or type can be used with equal ease, from small 7-segment, to British flag types, to larger 5 x 7, 7 x 9 or 5 x 12 displays.

The drivers can be directly interfaced to the microcontroller, COPS™4XX, or 80C48/9. This would normally be accomplished by connecting the driver's data and clock lines to control ports on the microprocessor. The MM58248/348 series is capable of accepting clock rates up to 1 MHz, and the MM58241/341 800 kHz. This is far faster than the control port bit manipulation rates for these controllers and will ensure compatibility with most low end microprocessors. 1 MHz input clock rates will also ensure that the desired display refreshing rate is attained.

In higher end systems using NSC800™ or 6800 8-bit microprocessors, the 1 MHz clock rate, coupled with a 300 ns minimum pulse width, simplified direct interfacing of these drivers to a μP bus. In the simple case, some logic for address decoding would set aside an I/O port for communiction to each driver, then several bits of the data bus could be gated to create the clock, data and enable signals.

4.0 TYPICAL DESIGN IMPLEMENTATION

4.1 Simple Direct Drive Application

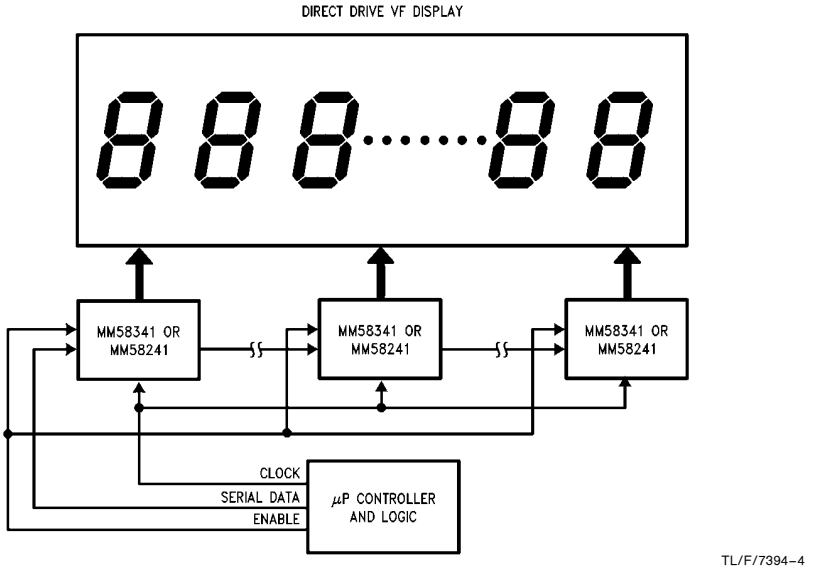
Figure 5 illustrates a simple cascaded direct drive application where MM58241s are cascaded to drive a 7-segment (plus decimal point) display. The MM58241s were chosen because of the ease with which they can be cascaded. The MM58248s can also be used and provide a few more outputs per package, but cannot be cascaded.

In this application, the controlling μP need only update the display whenever the data changes. When updating the display, the data is assembled, enable is raised, and the data is clocked serially to the driver. Once all the data is loaded into the shift registers, the enable is taken low. This action updates the display.

4.2 A 32-Digit 5 x 7 Dot Matrix Application

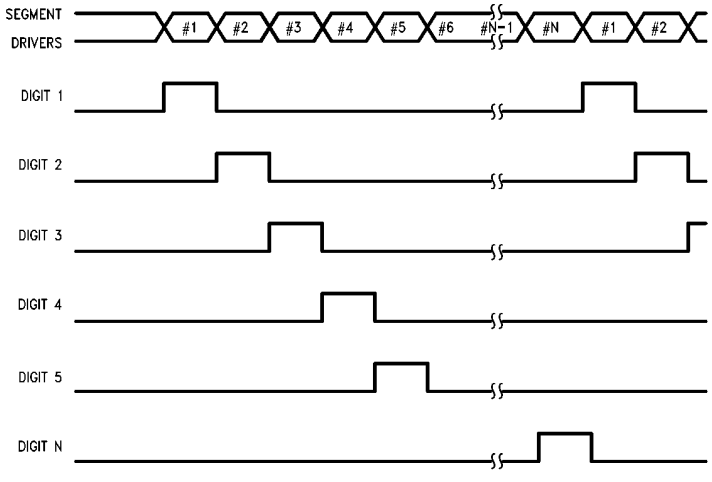
In this application, the obvious choice is to implement some sort of multiplexing scheme to drive the display with fewer lines. This application usually requires that a dedicated controller be used to generate all the timing signals.

General multiplex timing of a VF display is usually similar to LED multiplexing. First, the segment data for one character is output to the display. Next, the digit strobe for that digit is raised, enabling the character. Then the digit strobe is brought low while the segment data is changed to the next character on the display. The next character is enabled by raising the digit strobe. This action continues until each character is turned on sequentially. *Figure 6* shows the basic timing for a simple display.



TL/F/7394–4

FIGURE 5. Typical Direct Drive System with μP



TL/F/7394–5

FIGURE 6. Simplified Timing for Multiplexed VF Display

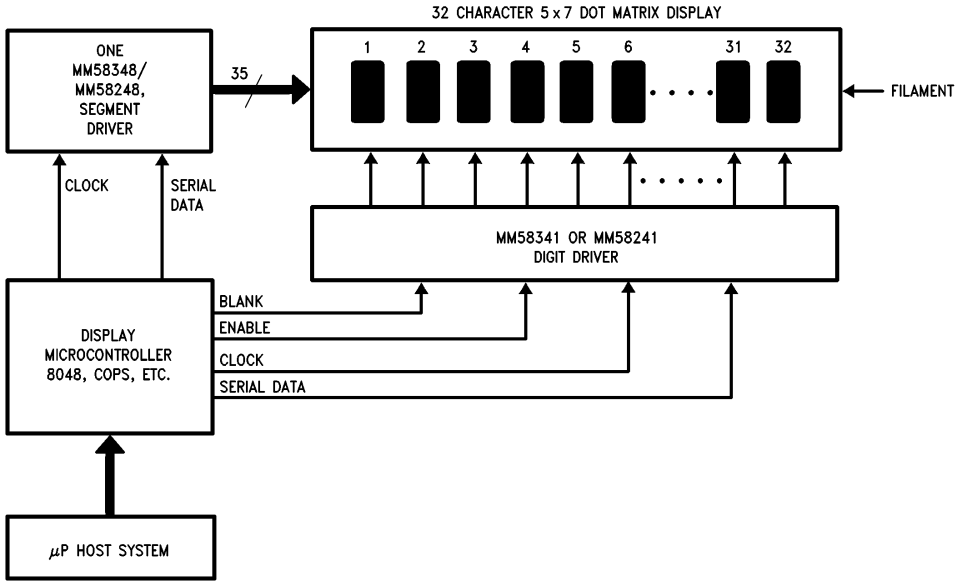
4.2 A 32-Digit 5 x 7 Dot Matrix Application (Continued)

In this design, it is logical to use one MM58341 to control the display's digits. As will be seen, this driver can be easily used to shift a single high level bit which will be used to sequentially enable each character. One MM58348 can be used to drive the segments. A 5 x 7 matrix has 35 segments, which is ideal for the MM58348. Therefore, this configuration has a total of 6 connecting lines to interface the microprocessor to the display drivers. The connection diagram is shown in *Figure 7*. Because both of the drivers accept data only when the clock is active, it would be possible to couple both data lines together. However, although this saves one interface line, there is a disproportionate increase in the software burden.

The choice of which driver to use for segments and which for digits is dependent only on which configuration is the simplest to implement in hardware or software. The MM58241/242/248 devices are all equally capable of driving the digits or segments of a display.

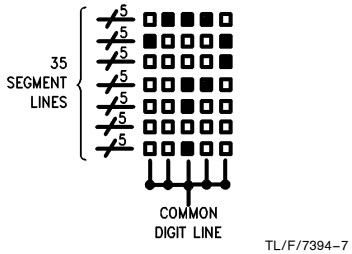
4.3 Multiplexed Display Refresh Timing: The Controllers

Considering first the digit driver (MM58341), it is clear that the digits must be enabled sequentially and that this process must be continuous, even when the display data has not changed. To this end, the data for the MM58341 is simply a one followed by 31 zeroes, where the one is shifted along the internal register. As each digit is enabled, the corresponding segment data is displayed. To ensure that no ghosting effects are seen during the transition between digits, the blank signal is activated for a short time before and after the segment data is changed. *Figure 8* shows the microprocessor waveforms and the resultant display waveforms for the 32-character design. Thus, one can see how the blank is used to mask the display while the digit enable signal goes low and the segment data is latched.



TL/F/7394–6

FIGURE 7a. Typical Architecture for Higher End System Utilizing Dedicated Display μP



TL/F/7394–7

FIGURE 7b. Detail of Typical Dot Matrix Digit

4.4 Multiplexed Display Refresh Timing: Display Brightness (Continued)

The refresh rate of the display is defined as the frequency at which each digit is enabled (i.e., the reciprocal of the time taken to display all digits). It is generally accepted that in order to avoid visible flickering, a refresh rate in excess of 50 Hz is required. Typically, VF manufacturers recommend 100 Hz–200 Hz. Some sample calculations follow and assume a refresh rate of 100 Hz. Therefore, the time given to display all digits is 1/(100 Hz) = 10 ms, and is 10 ms/number of digits) for any one digit. For this example, 10 ms/32 = 312.5 μs. This is defined as the total digit multiplex time and will be made up of a digit ''on'' time and an inter-digit blanking time. The inter-digit blanking is required to prevent display ghosting when digit information changes.

In general, then, the total digit time (t_D) is the inverse of the refresh rate (fr) divided by the number of digits in the display (nd), i.e.,

$$t_D = 1/(\text{fr x nd}) \text{ seconds.}$$

Since each digit time is composed of the ''on'' time, t_{DON} and the blanking time, t_{DOFF}, the total digit time is:

$$t_D = t_{DON} + t_{DOFF} \text{ (seconds).}$$

A useful measure of the brightness of a multiplexed display can be obtained by comparing it to the direct drive (100% brightness) case. In the direct drive application each digit is ''on'' permanently, while in the multiplexed mode each digit is ''on'' only for a portion of the time taken to refresh the display. Therefore, the measure of multiplexed brightness is given by the ratio of an individual digit ''on'' time to the total refresh time. Noting that the refresh time is a function of the total digit time (t_D) and the number of display digits (nd), a percentage figure for the brightness compared to the direct drive case can easily be calculated.

$$\text{percent muxed brightness} = \frac{t_{DON}}{t_D \times \text{nd}} \times 100$$

$$\text{percent muxed brightness} = \frac{t_{DON}}{(t_{DON} + t_{DOFF})\text{nd}} \times 100$$

Thus, regardless of the display logic's refresh speed, the display brightness will obviously depend on the amount of multiplexing and the amount of inter-digit blanking time. This is one constraint limiting the multiplexing scheme, and display manufacturers' data sheets should be consulted to determine a display's limits. This will, to a large degree, determine whether a design should use 32-digit multiplexing or perhaps two separate 16-digit multiplexed displays.

There are also limitations on the refresh rate based on the speeds of the hardware. In this design, the ''on'' time has a minimum value given by the time required to load the start bit and the first 34 data bits of the MM58348/248, i.e., the time required for 35 clock pulses of the MM58348. The MM58348 has a maximum clock frequency of 1 MHz, so the minimum time for 35 clock pulses is 35/(1 MHz) = 35 μs.

The digit ''off'' time is constrained by the time required to clock the digit driver and to load the final segment data bit, or the time for the display outputs to switch off then on, whichever is greater. The MM58341 has a maximum clock frequency of 800 kHz, so the minimum digit ''off'' time due to driver limitations is related to 1/(800 kHz) + 1/(1 MHz) = 2.25 μs. The blanking signal must be active during this time. Also, though the display outputs take typically 20 μs to switch, the display itself limits the minimum digit ''off'' time and is actually 20 μs.

Thus, the minimum total digit time per digit is:

$$t_D = 35\ \mu\text{s}, + 20\ \mu\text{s} = 55\ \mu\text{s}.$$

At a refresh rate of 10 ms, 10 ms/55 μs equals the theoretical maximum number of digits that can be multiplexed, or about 180 35-segment characters. This is unrealizable since current display ''on'' times must be greater than 35 μs, and total digit duty cycles (or percent brightness) must be much higher.

$$\text{percent brightness} = \frac{25\ \mu\text{s}}{55\ \mu\text{s} \times 180} \times 100 = 0.25\%$$

For the 32-digit case, the percent brightness is more realistic:

$$\text{percent brightness} = \frac{0.29\ \text{ms}}{0.31\ \text{ms} \times 32} \times 100 = 2.9\%$$

These times are the limits of the drivers. If the time required to load them is limited by the speed of the controlling processor, the update times are calculated from the clock rates of the controlling μP. However, as one can see, the limitations are more likely due to the display.

4.5 VF Display Brightness Control

Generally, to control or vary the brightness of a display, one can either vary the display drive voltage or vary the ''on'' time duty cycle by applying a signal to the blanking control. The duty cycle of the blanking signal will determine the brightness. This latter technique is preferred since more predictable behavior results.

In the simple direct drive case the MM58241/341/242/342 must be used. A periodic waveform is applied to the blanking pin. Its frequency should be greater than 100 Hz–200 Hz. As the duty cycle is varied, the percentage of time that the digits are ''on'' is changed and the perceived brightness changes.

In a multiplexed application, the brightness can be altered by merely modifying the relative length of the inter-digit blanking signal. This is easily accomplished in the software of the controlling μP by adding a delay while the blanking signal is active and subtracting the same delay from the time the blanking signal is inactive.

The relative brightness is the percentage of time that any one character is ''on'' divided by the sum of the character's ''on'' and ''off'' times. The latter term was previously defined as the total digit time. Thus:

$$\text{relative brightness} = t_{DON}/t_D.$$

Due to hardware refresh update speed limitations, 100% and 0% brightness cannot be achieved and also maintain proper refreshing, although 0% can be achieved just by stopping refresh and blanking continuously. (Note that this percentage is relative to the theoretical minimum and maximum brightness for a given multiplexed display, not the brightness relative to a direct drive display as was done previously.)

In the above 32-digit example, the maximum brightness is (assuming 10 ms refresh rate and 20 μs minimum inter-digit blanking):

$$\text{max. percent brightness} = (0.29\ \text{ms}/0.31\ \text{ms}) \times 100 = 93.6\%.$$

The minimum brightness, assuming 35 μs minimum ''on'' time is:

$$\text{min. percent brightness} = (0.035\ \text{ms}/0.31\ \text{ms}) \times 100 = 11.2\%.$$

Clearly there is a large range of available display brightness levels which are easily software controllable by altering the duty cycle of the blanking signal.

Again, the above analysis assumes that the microprocessor unit is interfacing with the display drivers at their maximum data rates. If this is not the case, some part of the brightness range will be lost.

Refer to Appendix for general system considerations.

5.0 THE SOFTWARE

Having outlined the general method by which the data can be displayed, it now remains to demonstrate how this can be achieved at the microprocessor level. It was thought best to use a familiar microprocessor for this task, so the implementation will use a 6502 and 6522 VIA circuit. The procedure which will be described is merely one example of how these display drivers can be applied, and it is hoped that by concentrating on the arranging and loading of the data, a more general benefit will be gained.

The application to be described here is that of an alpha-numeric display where characters are entered from a keyboard onto a VF display, feeding in from the left. The program will also accept control codes such as line feed, delete, etc. The general flowchart for this routine is shown in *Figure 9*. When the character is collected from the keyboard buffer it must first be established that it is a valid ASCII code; if not, it is ignored and the present data will continue to be displayed. The next thing to check is whether it is a control code. If it is a control code, the function represented must be executed on the existing data and the resulting data displayed. Assuming the ASCII code is not identified as a control code, it must correspond to a display character, and hence will be entered at the start of the display data buffer. Following this, the 32 characters denoted by the contents of the display data buffer are displayed, and after the last digit is enabled the keyboard buffer is checked for new data. As the display refresh rate far exceeds the speed of the human typist, each set of data is displayed several times before it changes.

Looking at the routine for displaying the 32 digits in more detail, a flowchart can be drawn up, as shown in *Figure 10*.

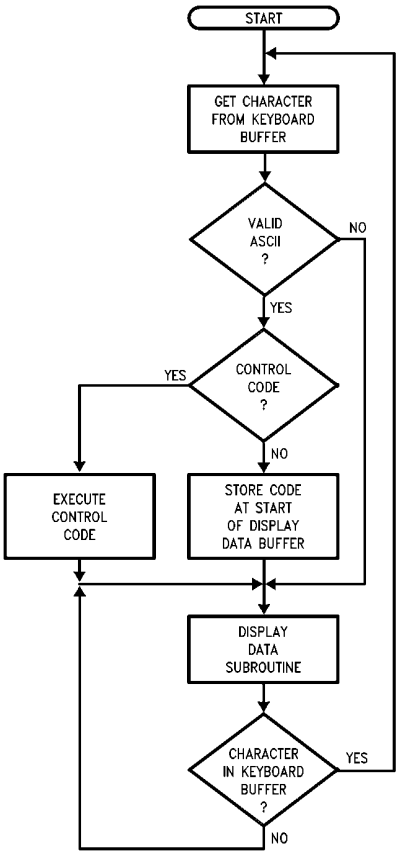


FIGURE 9. General IC Flowchart

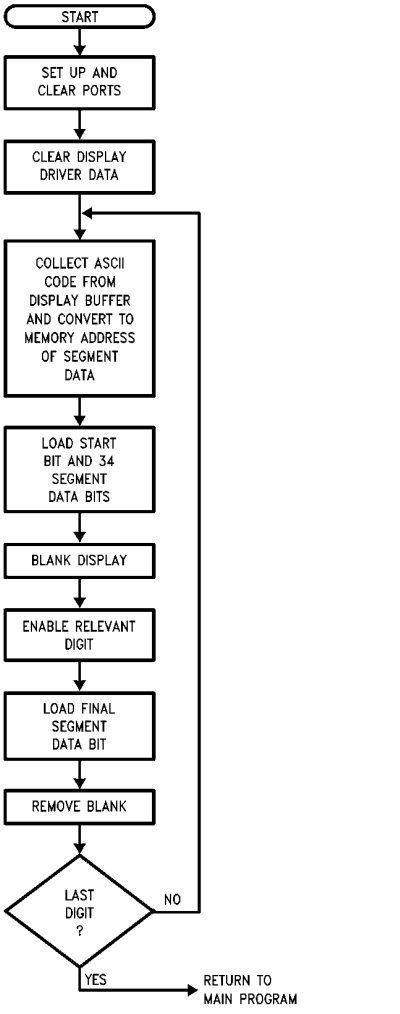


FIGURE 10. Display Data Subroutine Flowchart

5.0 THE SOFTWARE (Continued)

After first setting up and clearing the port lines and both the display drivers, a routine is performed for each of the 32 display characters. The ASCII code is collected and decoded to reveal memory locations where the corresponding 35 segment bits are stored. The start bit and first 34 data bits are loaded into the MM58348. Then the MM58341 blank signal is activated while the relevant digit is enabled and the final segment is loaded. After blank is removed, the next ASCII code is collected and decoded, etc. When the last digit has been loaded, control returns to the main program for the updating of the display data.

The machine code routine for the setting up of each digit of display data is shown in *Figure 11*. The relevant addresses of ports and digit codes are included in *Figure 12* to make the program comprehensible. The address of the segment data is stored in locations 00E2 and 00E3, and it is assumed that this is updated outside the display subroutine. The ports can be addressed as 8-bit memory locations or as individual lines. When considered as individual bits, the fifth address bit either sets or clears that bit, i.e., STA 0915 sets PA5 and 090B clears PB3.

The segment data for each character is stored as 7 consecutive memory locations, each containing 5 data bits.

The contents of each location are loaded into port A via the accumulator, starting with the highest memory address. This is achieved by indexing the lowest memory address by the contents of the Y register (starting as 06) and decrementing this register as every 5 bits are loaded. The least significant bit of port A is used for the segment data (PA0) and the 5 data bits are loaded by storing the code to port A, logically shifting it to the right, and storing it to port A again. This procedure is repeated 5 times for each memory location. The code given in *Figure 11* is for the brightest case, i.e., where the blank signal is disabled, as soon as the data has been latched to the display outputs. Clearly, the brightness can be altered by delaying this action.

The data is held in memory in the form of 7 locations of 5 bits each. This format was chosen because each location can be equated to one row of 5 x 7 dot matrix, where the lowest memory location corresponds to the bottom row. For example, if it is desired that a ''5'' be displayed in the form shown in *Figure 12*, then the 36-bit data stream is as demonstrated. Assuming that the data is stored at the 7 locations starting at address 2120, then the location contents are as denoted in *Figure 13*.

```
DISPLAY   STA 0910
          STA 0918
          STA 0908
          STA 0900        \ Load start bit.
          LDY #06         \ Load 30 segment
          JSR LOAD5       \ bits, 5 at a time
          LDA (I)E2       \ Load lowest addr.
          LDX #04
LOOP1     STA 0920
          STA 0918
          STA 0909
          LSRA
          DEX
          BNE LOOP1       \ Load 4 seg. bits.
          STA 091B        \ Enable high.
          STA 0919
          STA 0909        \ Digit select clock.
          STA 091C        \ Blank high
          STA 090B        \ Enable low.
          STA 0920
          STA 0918        \ Load 35th segment
          STA 0908        \ bit.
          STA 090C        \ Blank low.
          RTS             \ Ret. to main prog.
LOAD5     LDA (I), Y E2   \ Load mem. contents.
          LDX #05         \ Count for 5 bits.
LOOP2     STA 0920        \ Push data to port A.
          STA 0918        \ Segment clock high.
          STA 0908        \ Segment clock low.
          LSRA            \ Shift seg. data
          DEX             \ and dec. bit count.
          BNE LOOP2
          DEY             \ Set up address of
          BNE LOAD5       \ next 5 seg. bits.
          RTS             \ Return to display
                          \ subroutine.
```

FIGURE 11. Display Data Load Subroutine

Port A—Address 0920

Port Bit	Function	Address
PA0	Data 8	09-0
PA1	Not Used	09-1
PA2	Not Used	09-2
PA3	Not Used	09-3
PA4	Not Used	09-4
PA5	Not Used	09-5
PA6	Not Used	09-6
PA7	Not Used	09-7

Port B—Address 0921

Port Bit	Function	Address
PB0	Clock 8	09-8
PB1	Clock 1	09-9
PB2	Data 1	09-A
PB3	Enable	09-B
PB4	Blank	09-C
PB5	Not Used	09-D
PB6	Not Used	09-E
PB7	Not Used	09-F

Lowest memory address giving location of segment display data: stored in locations 00E2 and 00E3.

FIGURE 12. Port and Relevant Memory Addresses

5.0 THE SOFTWARE (Continued)

Desired Display Character (Numeric 5)

XXXXX	row 1
X	row 2
X	row 3
XXXX	row 4
X	row 5
X X	row 6
XXX	row 7

Desired Data Stream

01111000010000111110100001000011111 | 1 | start bit

direction of data entry →

Memory Contents (Assuming Lowest Address = 2120)

Address	Contents	
2120	0E	row 7
2121	11	row 6
2122	01	row 5
2123	1E	row 4
2124	10	row 3
2125	10	row 2
2126	1F	row 1

FIGURE 13. Example of Segment Data Arrangement

6.0 CONCLUSIONS

This design example is merely one of the many possible applications for the MM58348/341/248/241 family of high voltage display drivers. For other applications it should be noted that the other 2 circuits in this series, namely the MM58342 and the MM58242, can provide a much wider range of possible connections. For example, larger displays such as the 2-line by 40-digit 5 x 7 dot matrix formats can be driven with two MM58348s and two MM58342s cascaded to form a 40-bit shift register.

The method by which the segment data is shown to be stored and accessed in memory in convenient in the above example, although again it is only one of the many methods. An alternative using 5 locations of 7-segment data bits is possible, where the software would be faster but the data formatting more difficult. There are many trade-offs to be found with so versatile a series of circuits.

The code used to demonstrate this example is that of the 6502 microprocessor, and it would be a simple task to convert the instructions for another device (e.g., National's CMOS NSC800). The versatility of display formats available is a major feature, and the fact that these display drivers are CMOS devices guarantees their low power consumption. In addition, the outputs incorporate internal pull-down resistors which greatly reduce the external component count. This cuts the required board area; consequently a considerable saving in system cost can be made.

APPENDIX: SYSTEM CONSIDERATIONS FOR VF DISPLAY DRIVING

The purpose of the following text is to show how a designer can make decisions on displays he can drive or ranges of brightness he can achieve with a given system. Alternatively, it can be used as a method of designing a system to meet a desired display specification.

THE THEORY

1. System Decisions

System Constraints:

a. Refresh rate (f_r)

b. Number of display digits (nd)

c. Rate at which drivers are clocked by system (f_{CLK})

Associated Parameters:

a. Total time available to diplay all digits (t_r)

b. Total time allocated to each digit (t_D)

c. Total time each digit is on (t_{DON})

d. Total time each digit is off (t_{DOFF})

e. Number of display segments (ns)

f. Number of system clocks required to display one digit (nc_{ON})

g. Number of system clocks required to load segment bits (nc)

h. Number of system clocks required to latch both segment and digit data (nc_{OFF})

From the above definitions, the following equations can be stated:

$t_r = 1/f_r$ (seconds)

$t_D = t_r/nd = 1/(f_r \times nd)$ (seconds)

t_{DON} = time to load segment bits for next digit
= ns system clocks
= nc_{ON} system clocks
= nc_{ON}/f_{CLK} (seconds)

t_{DOFF} = time to latch segment bits and to enable relevant digit
= nc_{OFF} system clocks
= nc_{OFF}/f_{CLK} (seconds)

Hence:

$t_D = t_{DON} + t_{DOFF}$
$= (nc_{ON}/f_{CLK}) + (nc_{OFF}/f_{CLK})$
$= (nc_{ON} + nc_{OFF})/f_{CLK}$
$= nc/f_{CLK}$ (seconds)

And:

$f_{CLK} = nc/t_D$
$= nc \times f_r, \times nd$ (Hertz)

2. Brightness Variation Considerations

The brightness of the display is proportional to the duty cycle of the blank signal, and the range of intensities available depends on the size of t_{DON} and ultimately the refresh rate, fr.

Bd = brightness of the display
= duty cycle of blank signal
= t_{DON}/t_D

In the above example, the least bright case is where the blank signal is low for only one system clock per digit.

Bd (min) = 1/nc

And the brightest case is where blank is low only for the time required to latch the segment data and enable the digit.

Bd (max) = $(nc - nc_{OFF})/nc$
= $1 - [(nc_{OFF})/nc]$

The range of available brightness level, Br, is:

Br = Bd (max)/Bd (min)
= $[(nc - nc_{OFF})/nc]/(1/nc)$
= $nc - nc_{OFF}$

It should be noted that this is the minimum range of available brightness levels because t_D was minimized to maximize fr. If the system clock were fast enough to allow the maximum refresh rate to be in excess of the desired fr, then t_D could be increased from its minimum value. This would, in turn, produce a wider range of brightness levels.

It should also be noted that most manufacturers quote a minimum duty cycle for each digit. The system designer should ensure that neither end of the brightness specification exceeds this value.

THE APPLICATION

For the purposes of doing some sample calculations using the above theory, we will assume use of the system previously described, i.e., the driving of a 32-digit 5 x 7 dot matrix display by one MM58341/241 and one MM58348/248.

1. System Decisions

The number of display digits is fixed, i.e., nd = 32 (nc_{ON} = 35 and nc_{OFF} = 2, so nc = 37 system clocks). Assume system has a 125 kHz clock rate.

Therefore, the resulting refresh rate is:

$f_r = f_{CLK}/(nc \times nd)$
$= 125000/(37 \times 32)$
$= 105$ Hz

Also, the system clock rate needed for a given refresh rate can be calculated, e.g., f_r = 200 Hz.

$f_{CLK} = nc \times f_r \times nd$
$= 37 \times 32 \times 200$
$= 237$ kHz
= approximately 250 kHz

There are many other examples of how this theory can be used to evaluate the possibilities for VF systems.

2. Brightness Variation Considerations

We can now calculate range of brightness intensities available with the above system, i.e., where f_{CLK} = 125 kHz, f_r = 105 Hz.

Bd (min) = 1/nc
= 1/37

Bd (max) = $1 - (nc_{OFF}/nc)$
= $1 - (2/37)$
= 35/37

Br = Bd (max)/Bd (min)
= 35

So the brightness can vary from its lowest value to its maximum value, which is 35 times the minimum level.

Also note that the minimum duty cycle for this display is given as 1/40 (manufacturer's specification), so there is no problem in this application.

Let us now take the example of driving the same display with a system where $f_{CLK} = 500$ kHz, at a desired refresh rate of 200 Hz.

$$nc = f_{CLK}/(f_r \times nd)$$
$$= 500000/(200 \times 32)$$

nc_{OFF} is 2 as before, and although we require only 35 clocks to load the segment data,

$$nc_{ON} = nc - nc_{OFF}$$
$$= 78 - 2$$
$$= 76 \text{ system clocks}$$

In general, the higher the system clock rate, the wider the brightness control range.

Therefore,

$$Bd\ (min) = 1/nc$$
$$= 1/78$$

But, remembering that Bd (min) must be less than the stated duty cycle (1/40):

$$Bd\ (min) = 2/78$$
$$= 1/39$$
$$Bd\ (max) = 1 - (nc_{OFF}/nc)$$
$$= 1 - (2/78)$$
$$= 76/78$$
$$= 38/39$$
$$Br = Bd\ (max)/Bd\ (min)$$
$$= (38/39)/(1/39)$$
$$= 38$$

So, we can see that by manipulation of the system constraints, a wider range of brightness levels can be attained, although this is ultimately limited by the stated duty cycle of the display.

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-180-530 85 86
Email: cnjwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180-532 78 32
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
Tel: 81-043-299-2309
Fax: 81-043-299-2408