How-to design LED signage and LED matrix displays, Part 2

MASASHI NOGAWA - August 20, 2014

LED signage and matrix displays are an integral part of the 21st century landscape. Their quality is becoming so good that it's becoming increasingly difficult to distinguish LED images from traditional billboards.

Part 1 of this three-part series addressed the technical and engineering steps necessary to design an LED display system from individual LED lamps. In Part 2 we'll visit the remaining necessary steps to complete the LED system. This will lay the groundwork for Part 3 which will introduce several important features becoming commonly available on advanced LED display driver ICs, as well as tips on how to make best use of them.

How to transfer display data

To begin, let’s review how data is transferred between the image processing controller and the LED driver ICs. Table 1 lists the design specifications of the example application we'll be working with in Part 2.

Table 1 Example specifications.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame Rate</td>
<td>FR</td>
<td>50 Hz = 20 ms</td>
</tr>
<tr>
<td>Frame Refresh Rate</td>
<td>FRR</td>
<td>200 Hz = 5 ms</td>
</tr>
<tr>
<td>Number of LED Driver IC</td>
<td>N_DIC</td>
<td>8 IC in cascade</td>
</tr>
<tr>
<td>Number of outputs per IC</td>
<td>N_OUT</td>
<td>16 output per IC</td>
</tr>
<tr>
<td>Number of Gray-scale</td>
<td>N_GS</td>
<td>12-bit gray-scale = 4096 steps</td>
</tr>
</tbody>
</table>

System with ON/OFF driver IC

In a basic ON/OFF driver IC system, such as the example in Figure 1 (Originally Figure 11a in Part 1), an image processing controller must generate a pulse-width modulation (PWM) pattern map on its memory.
This pattern map consists of an ON/OFF status (bit) of each output managed by this controller per each gray scale PWM clock (Figure 2a). The ON and OFF patterns of OUT<sub>N</sub> and OUT<sub>M</sub> are rendered for one entire frame period (5 ms), plus a few more clocks. Once the controller finishes rendering the memory map, the controller sends out the resulting pattern of logic-H and logic-L, (along with each red line) to the ON/OFF control of the LED drivers’ shift register.

For this system, the data transfer rate \( f_{\text{DATA(ON/OFF)}} \) is expressed in equation 1.

\[
I_{\text{LED}} = I_S \left( e^{(V_F - R_S \times I_{\text{LED}})/V_T} - 1 \right)
\]

(eq1)

The parameters in Table 1, \( f_{\text{DATA(ON/OFF)}} \), are calculated as 105 MHz. For large outdoor display systems, delivering 105 MHz logic signal on its PCB is neither practical nor realistic. Most LED driver ICs cannot receive 105 MHz logic inputs anyway. In Table 1, the ON/OFF control driver cannot meet the target specification. A realistic frame refresh rate (FRR) needs to be downgraded to 50 Hz. The result is \( f_{\text{DATA(ON/OFF)}} = 26 \) MHz.
PWM driver IC system

In the PWM driver system used in Part 1, there are two logic signal speeds (or frequency parameters) to consider. The first is a gray scale reference clock frequency \( f_{GSCLK(PWM)} \):

\[
f_{GSCLK(PWM)} = FRR \times N_{GS}
\]  

(eq 2)

Using the parameters in Table 1, \( f_{GSCLK(PWM)} \) is calculated as 819 kHz, which is easy to achieve.

The second is a data transfer frequency \( f_{DATA(PWM)} \):

\[
f_{DATA(ON/OFF)} = FR \times N_{GS} \times N_{IC} \times N_{OUT}
\]  

(eq 3)

Based on Table 1, \( f_{DATA(PWM)} \) is calculated as 26 MHz. Note that a PWM control driver IC can repeat the same image data without resending gray scale data and the frame rate (FR) used in equation 3, which is unlike using FRR in equation 1. Figure 2b shows a simplified diagram of how PWM control driver ICs can reduce the data transfer rate.

ON/OFF versus PWM control drivers
There are pros and cons when choosing between ON/OFF and PWM control drivers. The choice depends on your display system’s needs. A generic comparison is listed in Table 2.

### Table 2 ON/OFF versus PWM control ICs

<table>
<thead>
<tr>
<th></th>
<th>ON/OFF control IC</th>
<th>PWM control IC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pros</strong></td>
<td>● Software provides very flexible control of the PWM output timing</td>
<td>● Less processor power needed</td>
</tr>
<tr>
<td></td>
<td>● Requires lots of software programming</td>
<td>● Less software development time needed</td>
</tr>
<tr>
<td></td>
<td>● Tough to increase frame refresh rate</td>
<td>● Easier to achieve faster frame refresh rate</td>
</tr>
<tr>
<td><strong>Cons</strong></td>
<td>● PWM output timing is hard coded and less flexibility</td>
<td></td>
</tr>
</tbody>
</table>

So far, a PWM control driver might look to be a better choice. Before you decide, however, let’s consider an important benefit of the ON/OFF control driver. The image processing controller has the flexibility to generate PWM patterns for all outputs by its software. For example, the controller can achieve a unique spread-spectrum PWM pattern to avoid potential electromagnetic interference (EMI), or a special gamma-correction.

**Driver ICs improve video image**

Now, let’s review the various LED driver IC functions.

**Dot correction**

Because of various error factors, an LED display must be calibrated in order to present a uniform brightness profile across each of its LEDs. Figure 3a represents a non-calibrated display showing the entire white image data. The inconsistencies may be due to one or more possible error factors such as, but not limited to:

- Current-to-light conversion efficiency difference of each LED lamp
- Forward voltage difference of each LED lamp
- Driving current error of LED driver ICs
- Error of reference current setting resistor connected to LED driver ICs
- Light output loss caused by physical dimension error of LED lamps

A dot correction (DC) function has the ability to adjust driving current of individual output terminals by referring to several bits of digital data. Thus, a dot correction is like a set of current-output digital-to-analog converters (DACs). An example of a dot-corrected calibrated display is shown in Figure 3b.
The schematic in Figure 4 depicts a circuit element which implements the dot correction function using DCx signals (highlighted in red text).

In Part 1, we confirmed that the LED current change shifts its output light wavelength (color). Because the dot correction function adjusts output current amplitude, a calibration attempt with a dot correction causes another smaller color shift. A best practice is to utilize visual inspection equipment which can digitize output color from a RGB lamp. Now, all lamps can be set to have the
same target digitized value.

**Brightness control**

Brightness control (BC) is a similar function to dot correction adjusting output current amplitude. However the brightness control changes all IC outputs simultaneously.

Brightness control is an effective way to adjust whole display brightness. As with adjusting the brightness on your notebook PC, it depends on the ambient brightness. For example, an outdoor display system in bright daylight needs the highest light output to overcome the bright sunlight. However, the same outdoor system doesn’t need near as much light output at night.

![Figure 5](image)

**Figure 5** Example of brightness control.

**Figure 5** shows how brightness settings can change the same image, depending on ambient brightness. **Figure 5a** is in a bright room while **Figure 5b** is in a dark room – but they both look the same to the human eye. Note: The BC signals which implement the brightness control function illustrated here are shown in the previous schematic (Figures 4), highlighted with green text.

**Part 3** explores some of the issues which affect the image quality and reliability of LED displays, and the technologies and design techniques used to deal with them.

**References**

1. How-to design LED signage and LED matrix displays (Part 1), Masashi Nogawa, EDN, July 30, 2014
2. More information about designing LED signage is available at the [LED display home page](http://www.ti.com) on the Texas Instruments web site.
3. Datasheets for the parts mentioned in this story can be downloaded from the following links: [TLC59283](http://www.ti.com), [TLC5958](http://www.ti.com).

**About the Author**

Masashi Nogawa is a product marketing engineer for Texas Instruments’ Power Management group where he is responsible for the SWIFT product line. Masashi received his BSEE and MSEE degrees from the University of Electro-communications, Tokyo, and he holds six US patents. Masashi can be reached at ti_masashinogawa@list.ti.com.