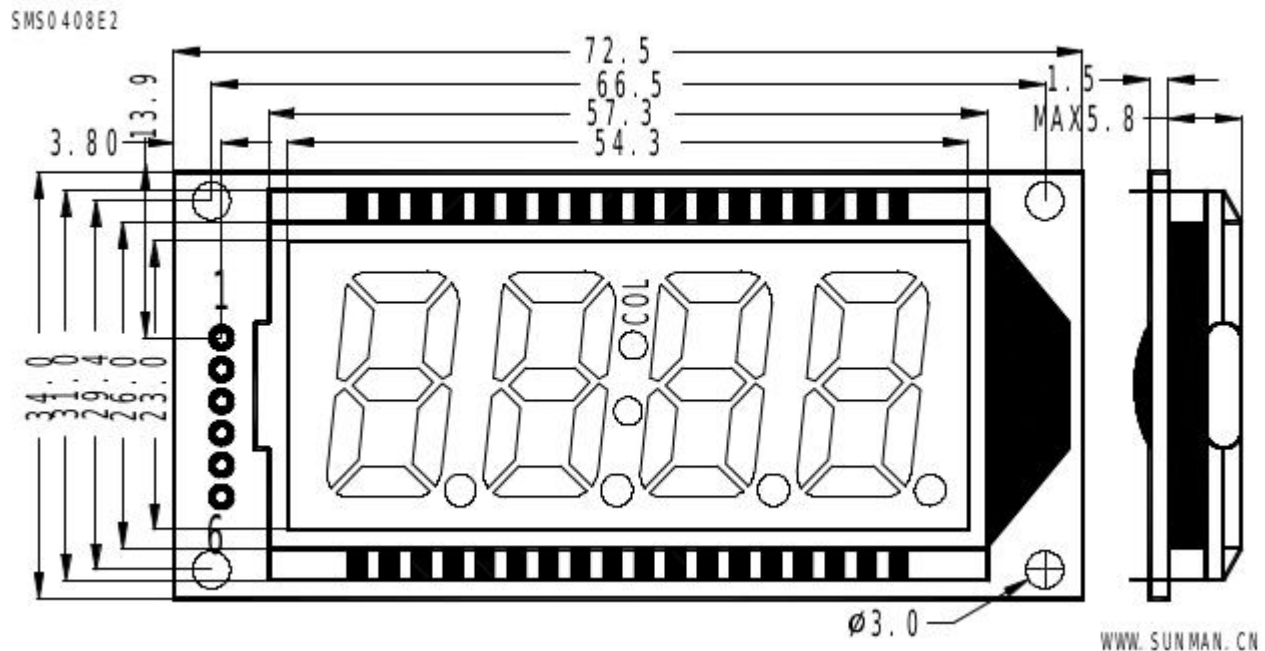


## Application Example for the 4 -Digit 7-segment LCD Display (SUNMAN SMS0408E2) by Filipe César Brandão

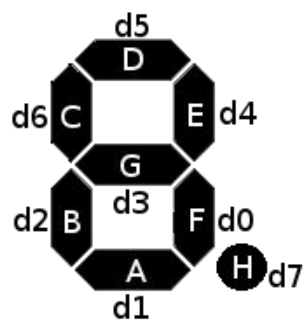
Besides this product does not have an application example on the documentation provided by the manufacturer, this application example was wrote in a simplified way to help you. See official documentation brochures on: <http://www.sunman.com.cn/lcm/product/sms0408e2.html> and <http://download.sunman.com.cn/lcm/1/sms0408.pdf>



This product is very simple to use and the information below can be useful for developers, hobbyists and programmers.

### 1.0 – Digits mapping and layout

Each digit are constructed as shown in Illustration 1:



*Illustration 1: Digits layout*

As possible to see in Illustration 1, the digits does not follow the usual distribution ABCDEFG as seen in many common anode or common cathode 7 segment displays. By this figure, the digit is upside down compared to the usual digits layouts, so if you tried and it did not work, that is one of the good reasons for it not to display what you wanted!

The segments are “**active low level**” wich means that, for a specific digit (buffer) the segment will be activated by a low level bit. Table 1 shows the construction of some common figures. Note that the **bit d7** is used to activate segment H and sometimes, to represent decimal point, you will need to enable this by setting this bit to **level logic 0**, on the buffer corresponding to the decimal digit.

## 2. Serial communication

On figure 2, there is an example of the synchronous serial communication implemented to communicate with the SMS0408E2, when 4 digits are sent. In this example, the number 801.3 is displayed on the 7 segments LCD if the described serial communication is implemented. Data is transmitted via DI input and synchronized via CLK clock input. All the tests were done in 5V supply.

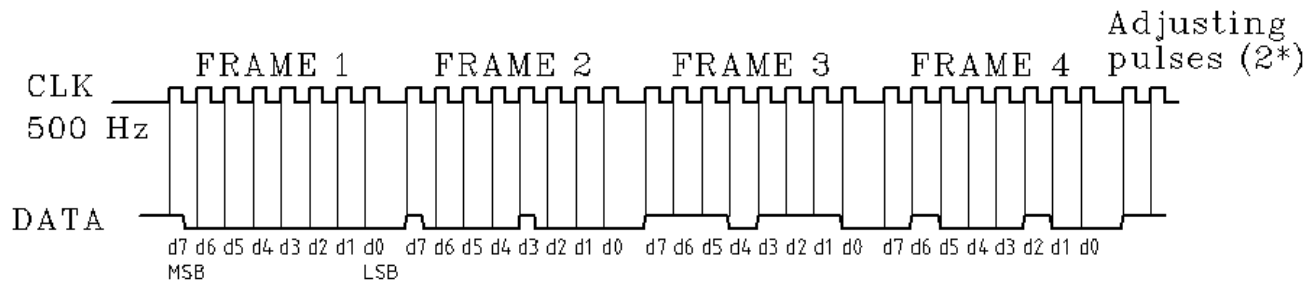
It is possible to observe that each frame is composed by 8 bits, and the transmission starts with the most significant bit (MSB) of the buffer 1 (most significant digit) and finishes with the least significant bit of the last digit. In the end, put data pin DI in high level and then, two clock cycles to adjust the display. This adjusting clock cycles are necessary and there is nothing on the documentation giving this tip. It was just discovered during LCD tests!

As reported before, the segments will be activated by “0” or low logic level bits, according to table 1. The column (buffer 0 or “:” sign used for time format display) was not covered in this example. To display this column, just send a first “zero” frame, basically **0h** or **0 0 0 0 0 0 0 0** **0b** which means, to send a series of 8 “zeros” via serial, send the four digits (HHMM) on the subsequent frames and finalize with the adjusting 2 clock cycles keeping data line high.

Figure	d7* H	d6 C	d5 D	d4 E	d3 G	d2 B	d1 A	d0 F
<b>0</b>	1	<b>0</b>	<b>0</b>	<b>0</b>	1	<b>0</b>	<b>0</b>	<b>0</b>
<b>1</b>	1	1	1	<b>0</b>	1	1	1	<b>0</b>
<b>2</b>	1	1	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	1
<b>3</b>	1	1	<b>0</b>	<b>0</b>	<b>0</b>	1	<b>0</b>	<b>0</b>
<b>4</b>	1	<b>0</b>	1	<b>0</b>	<b>0</b>	1	<b>0</b>	<b>0</b>
<b>5</b>	1	1	1	1	1	1	1	1
<b>6</b>	1	<b>0</b>	<b>0</b>	1	<b>0</b>	1	<b>0</b>	<b>0</b>
<b>7</b>	1	1	<b>0</b>	<b>0</b>	1	1	1	<b>0</b>
<b>8</b>	1	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>9</b>	1	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	1	<b>0</b>	<b>0</b>

Table 1 – Common figures according to the SMS0408E2 documentation. Digit d7 is used to represent the decimal point and is enabled by low level.

8 0 1 . 3



*Illustration 2- Description of the serial communication with the LCD*

### 3. LCD Testing with an Arduino Nano V3.0

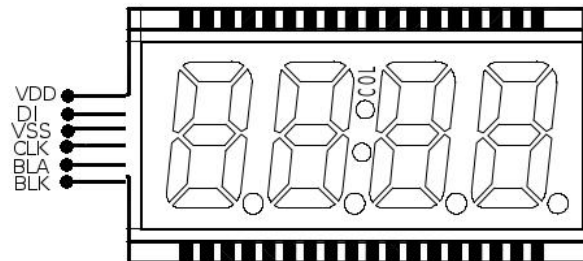
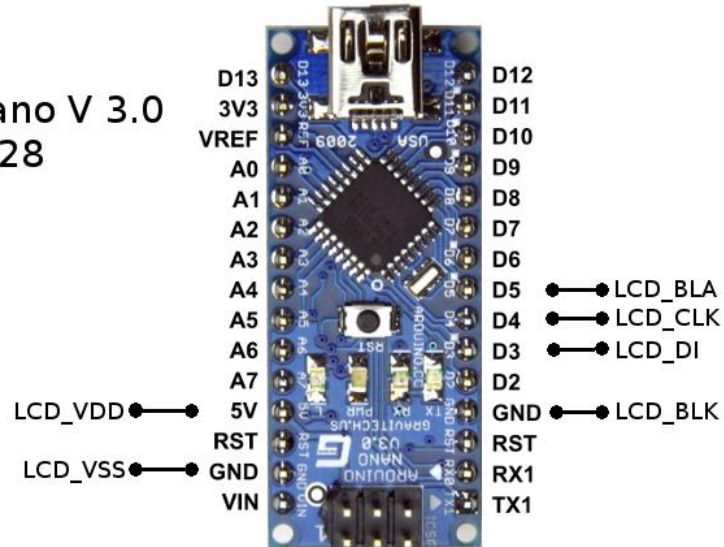
To test the LCD, the circuit on Illustration 3 has been implemented. An Arduino Nano V.3 (Atmel AT Mega 328 micro processor based module) was used to test the LCD and all programming was made with the Arduino IDE.

The connections on pins BLA and BLK are optional (to activate back light). It is recommended to add a 330Ω resistor between the Arduino D5 pin and the LCD\_BLA pin to limit the current on the back light led.

On Illustrations 4, 5 and 6, it's possible to see the LCD driven by the Arduino Nano V3.

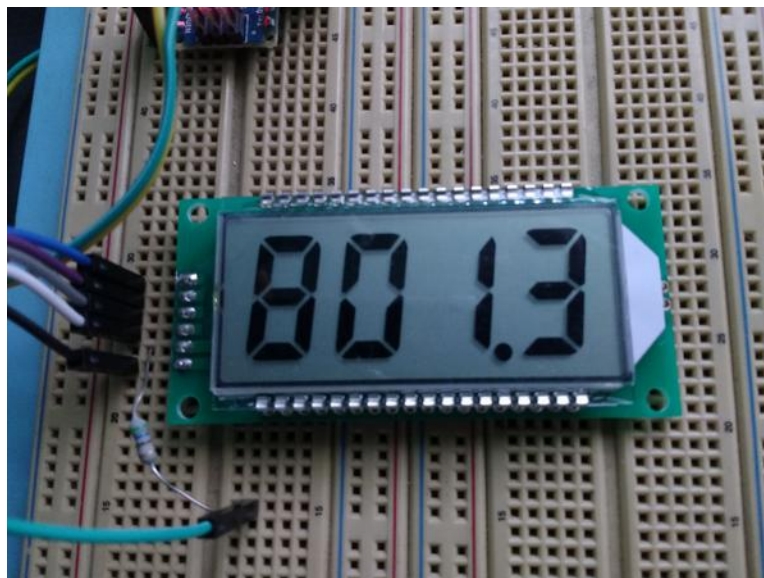
To have access to the Arduino code (library and examples), just send an e-mail to [budegarower@gmail.com](mailto:budegarower@gmail.com) with the subject "SMS0804-Arduino code". Many improvements should be done in the code, but as a starting point, this code can help a lot.

Arduino Nano V 3.0  
AT Mega 328

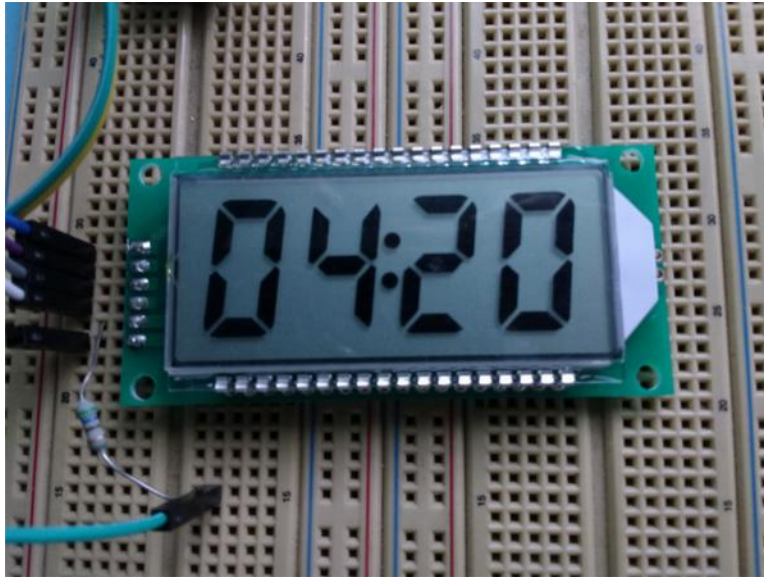


LCD - SUMAN  
SMS0408E2

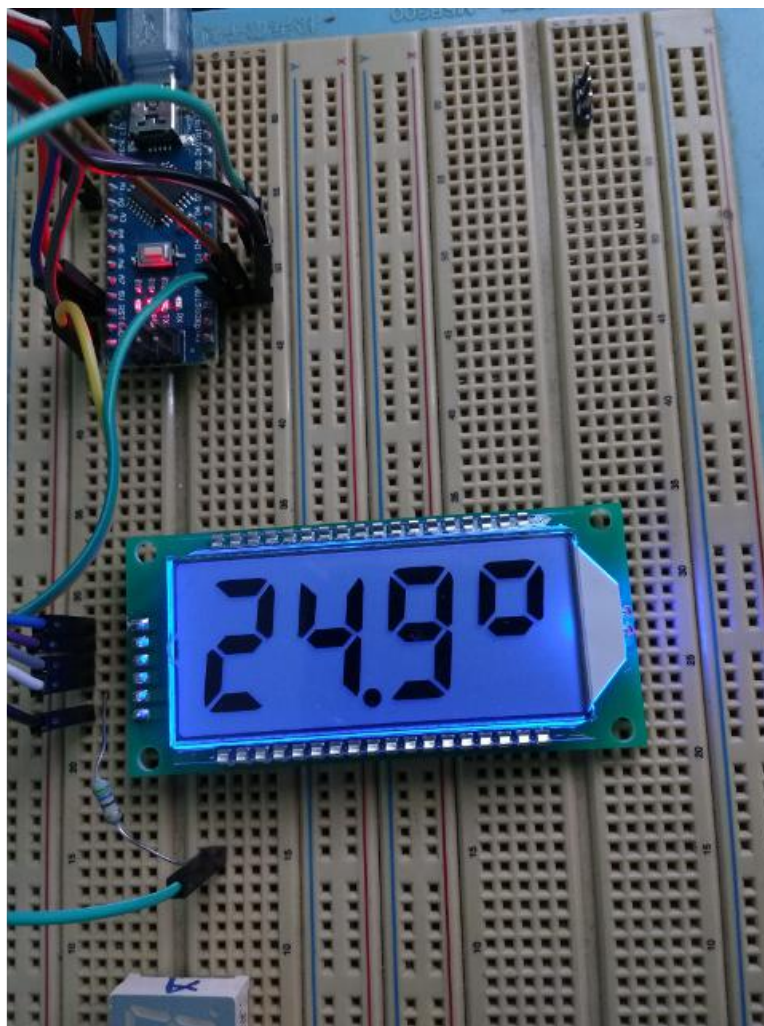
*Illustration 3: Test circuit*



*Illustration 4: LCD displays number 801.3 as described in section 2- Serial communication*



*Illustration 5: Time format on display.*



*Illustration 6: Implementation of a thermometer, back light is on.*