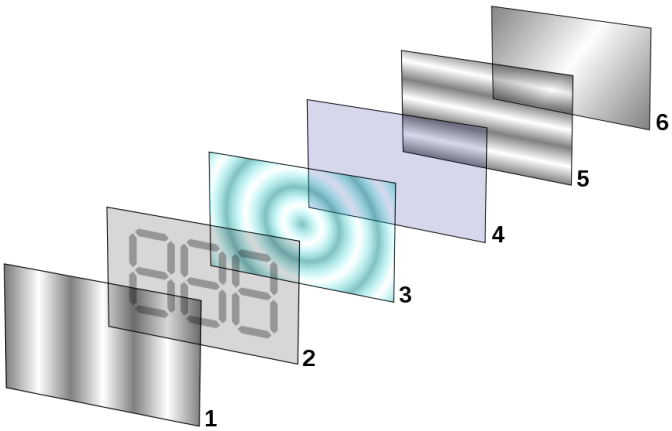
LCD Character Display Primer

Introduction

Liquid Crystal Displays are display devices used to convey information through arrangements of pixels. Graphic and Text LCDs are the most common types available for electronic products. LCDs range from something as simple as the relatively simple 16-120 character displays available for small electronics to large 60+" high definition television sets.

How it works

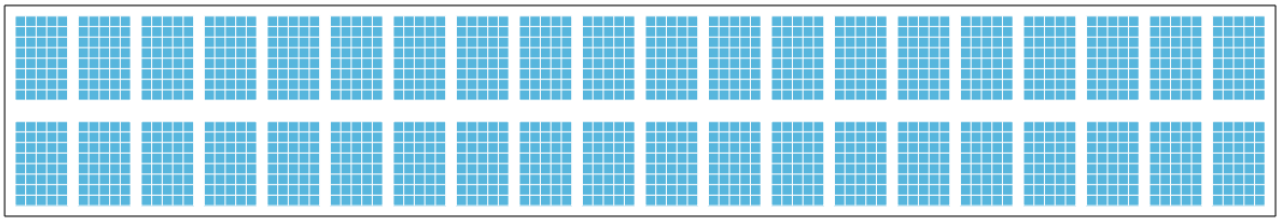
Each pixel of an LCD typically consists of a layer of molecules aligned between two transparent electrodes, and two polarizing filters which are perpendicular to one another. With no actual liquid crystal between the polarizing filters, light passing through the first filter would be blocked by the second (crossed) polarizer. The surfaces of the electrodes in contact with the liquid crystal material are treated in order to align the crystal molecules in a certain direction.



Reflective twisted nematic liquid crystal display. Polarizing filter film with a vertical axis to polarize light as it enters.Glass substrate with Indium tin oxide (ITO) electrodes. The shapes of these electrodes will determine the shapes that will appear when the LCD is turned ON. Vertical ridges etched on the surface are smooth.Twisted nematic liquid crystal.Glass substrate with common electrode film (ITO) with horizontal ridges to line up with the horizontal filter.Polarizing filter film with a horizontal axis to block/pass light.Reflective surface to send light back to viewer. (In a backlit LCD, this layer is replaced with a light source.)

In a twisted nematic device (still the most common liquid crystal device), the surface alignment directions at the two electrodes are perpendicular to each other, and so the molecules arrange themselves in a twist. When light passes through the liquid crystal molecules the rotation of the polarization changes slightly and reduces the effect so the light can pass through mostly unaffected. If a large voltage is applied, the liquid crystal molecules become almost completely untwisted. This causes the light to be heavily polarized again effectively making the pixel appear black. By adjusting the voltage, the polarization can be controlled precisely allowing you to display a full range of grey-scale.

Text LCDs



2X20 LCD Character Arrangement

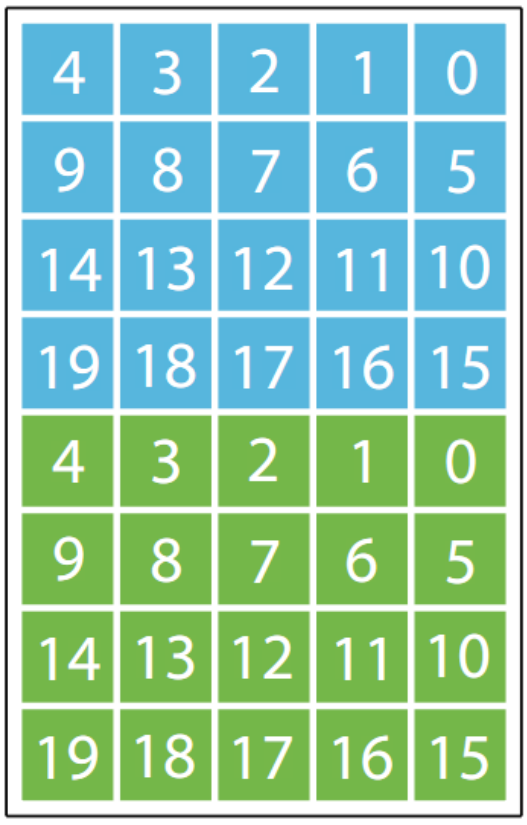
Text LCDs display full text strings set in software. Since text characters are defined from the ASCII standard library, other ASCII standard set characters and glyphs can also be sent to the text LCD. This can be done easily by using unicode characters within your text string. In C#, this may look something like this:

```
tLCD.rows[0].DisplayString = "Apple starts with \u0041";
```

In this example, the string \u indicates that a unicode character follows, and the unicode character 0041 (which references the hexadecimal character code 0x41) represents the capital letter A. After the LCD converts the unicode character, the above example would cause the LCD screen to read Apple starts with A. A chart of all ASCII standard set character codes is available here.

Custom Characters

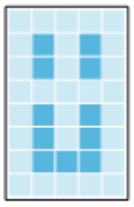
The normal characters in the TextLCD are hard-wired and inaccessible to the user. However, custom characters can be generated for the PhidgetTextLCD. A custom character can be any arrangement of pixels within the space allotted for a single character. Single characters are made up of pixels arranged in a grid 5 pixels wide by 8 pixels high. Once generated, custom characters can be stored in any one of eight volatile memory locations on the PhidgetTextLCD, and can be recalled with a simple API command from software. When custom characters are designed, a formula is used to change the pixel design into a pair of numerical values. The first value relates to the design of the top 4 rows of the character, and the second value relates to the design of the bottom 4 rows of the character. Unlike the unicode characters used in the Special Characters section above, the calculated number is not in hexadecimal format but is an integer value up to six characters in length. The calculation for custom characters can be done by hand, or can be completed for you by using the form available here [1]. Done by hand, each integer value represents the sum of two to the power of each individual on-pixel's location within that integer-value's half of the character. Pixels not turned on are valued at zero. For example, a custom



5X8 LCD Character Pixel Arrangement

character happy-face with pixels 6, 8, 11 and 13 in the upper half turned on, pixels 1, 3, 6, 8, 11, 12 and 13 in the lower half turned on, and all other pixels turned off, would result in the following integer values.

$$VAL_{UPPER} = 2^6 + 2^8 + 2^{11} + 2^{13} = 10560$$
$$VAL_{LOWER} = 2^1 + 2^3 + 2^6 + 2^8 + 2^{11} + 2^{12} + 2^{13} = 14666$$



These two values are then stored in one of eight memory locations (CG-RAM 0 to 7) on the PhidgetTextLCD by using the Set Custom Character method in software. In C#, this may look something like this:

```
tLCD.customCharacters[0].setCustomCharacter(10560, 14666)
```

Once stored, characters can be recalled into a text string by either using the unicode value for the location as referenced in the ASCII chart (Appendix A) or by using the String Code method from the API. Examples in C# of both methods are shown below:

```
tLCD.rows[0].DisplayString = "I am happy \u0008";
```

```
tLCD.rows[0].DisplayString = "I am happy " +
tLCD.customCharacters[0].StringCode;
```

ASCII Chart

References

[1] http://www.phidgets.com/documentation/customchar.html

Article Sources and Contributors

LCD Character Display Primer *Source*: http://www.phidgets.com/wiki/index.php?title=LCD_Character_Display_Primer *Contributors*: Burley, Mparadis

Image Sources, Licenses and Contributors

Image:3650.jpg *Source*: http://www.phidgets.com/wiki/index.php?title=File:3650.jpg *License*: unknown *Contributors*: Burley
image: lcdlayers.png *Source*: http://www.phidgets.com/wiki/index.php?title=File:Lcdlayers.png *License*: unknown *Contributors*: Burley
image:lcd display.png *Source*: http://www.phidgets.com/wiki/index.php?title=File:Lcd_display.png *License*: unknown *Contributors*: Burley
image:Lcdpixelarrangment.png *Source*: http://www.phidgets.com/wiki/index.php?title=File:Lcdpixelarrangment.png *License*: unknown *Contributors*: Burley
image:Lcdcustomcharmap.png *Source*: http://www.phidgets.com/wiki/index.php?title=File:Lcdcustomcharmap.png *License*: unknown *Contributors*: Burley
image:asciichart.png *Source*: http://www.phidgets.com/wiki/index.php?title=File:Asciichart.png *License*: unknown *Contributors*: Burley