

# **Micrium, Inc.**

© Copyright 2001, Micrium, Inc.  
All Rights reserved

## **Software simplifies LCD board layout**

**Application Note**  
AN-4001

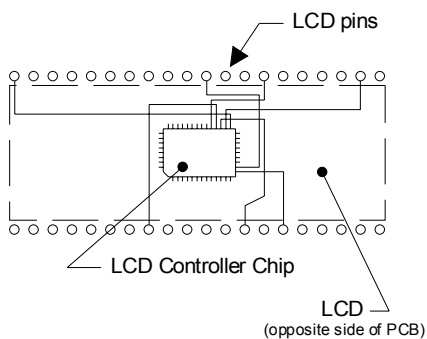
**Jean J. Labrosse**  
[Jean.Labrosse@Micrium.com](mailto:Jean.Labrosse@Micrium.com)  
[www.Micrium.com](http://www.Micrium.com)

## Summary

This application note shows you how you can use software to simplify the PCB (Printed Circuit Board) layout of discrete LCD segments. This application note is a reprint from an article I published in Electronic Design Magazine, April 15, 1996 issue (page 92).

## Introduction

Laying out a Liquid Crystal Display (LCD) and its controller chip on a Printed Circuit Board (PCB) can become tricky as shown in figure 1. The LCD is mounted on the opposite side of the PCB for sake of discussions and is shown with dotted lines. The LCD controller chip is connected to the LCD as required by the mapping of LCD segments which, in this case, is dictated by software requirements. The C code shown in figure 1 assumes that the desired segment patterns are stored in `DispBuf[]`. The LCD controller is assumed to be a serial access device and LCD segment information is shifted out to the LCD controller one bit at a time, least significant bit of `DispBuf[]` first.



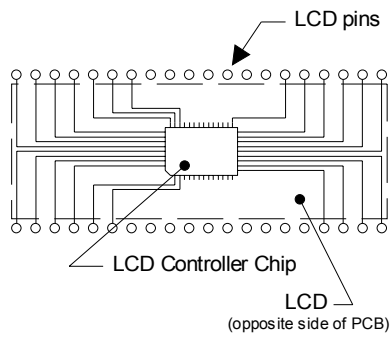
```
unsigned char DispBuf[SIZE];

void DispSend (void)
{
    unsigned char i, j, bit, data, *pbuf;

    pbuf = &DispBuf[0];
    for (i = 0; i < SIZE; i++) {
        data = *pbuf;
        for (j = 0; j < 8; j++) {
            bit = data & 0x01;
            output 'bit' to LCD controller;
            data >>= 1;
        }
        pbuf++;
    }
}
```

Figure 1, a) Conventional method of connecting a LCD controller to a LCD.  
b) C code used to output LCD segment information to LCD controller(s).

To simplify the layout of the PCB, a simple software mapping technique can be used to allow the PCB layout engineer to connect the LCD controller to the LCD as shown in figure 2. In this case, the connections are direct which reduces the number of feedthrough holes and line crossings. The software routine is almost as efficient as the one shown in figure 1. The new code requires the definition of a 'mapping' table called `DispMapTbl[]`. `DispMapTbl[]` simply defines the order in which bits in `DispBuf[]` are shifted out to the LCD controller. The mapping table would reside in ROM and would be defined once the PCB layout is complete. `DispMapTbl[]` assumes that bits in `DispBuf[]` are numbered as shown in figure 3. Custom LCDs having many segments may require more than one LCD controller chip. In this case, this mapping technique will considerably simplify the board layout and prevent lines from going from one end of the PCB to the other.



```

unsigned char DispBuf[SIZE];

void DispSend (void)
{
    unsigned char i, x, bit, msk, *pmap;

    pmap = &DispMapTbl[0];
    for (i = 0; i < MAX_BITS; i++) {
        x = *pmap >> 3;
        msk = 1 << (*pmap & 0x07);
        if (DispBuf[x] & msk) {
            bit = 1;
        } else {
            bit = 0;
        }
        output 'bit' to LCD controller;
        pmap++;
    }
}

```

Figure 2, a) LCD controller with direct connections to the LCD.  
b) C code used to perform the mapping.

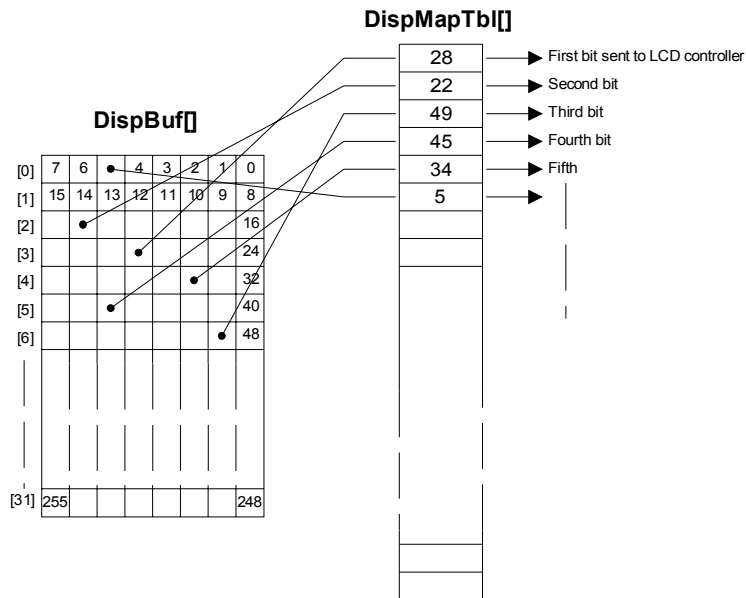


Figure 3, Flow diagram of LCD segment mapping.

## Contact Information

**Micrium, Inc.**

949 Crestview Circle

Weston, FL 33327

954-217-2036

954-217-2037 (FAX)

e-mail: [Jean.Labrosse@Micrium.com](mailto:Jean.Labrosse@Micrium.com)

WEB: [www.Micrium.com](http://www.Micrium.com)