**http://blog.csdn.net/drivermonkey/article/details/7181313**

[**LCD之调色板**](http://blog.csdn.net/drivermonkey/article/details/7181313)

 对于一个显示设备，数据的更新率正比于画面的像素数和色彩深度的乘积。在**嵌入式Linux**系统中，受处理器资源配置和运算能力的制约，当使用大分辨率显示时(如在一些屏幕尺寸较大的终端上，往往需要640×480以上)，需要降低显示的色彩深度。否则，由于数据处理负担过重会造成画面的抖动和不连贯。这时，调色板技术将发挥重要作用。**ARM9**内核的S3C2410在国内的嵌入式领域有着广泛的应用，芯片中带有**LCD控制器**，可支持多种分辨率、多种颜色深度的LCD显示输出。在此，将S3C2410的调色板技术，以及**嵌入式Linux**系统下调色板显示的实现方法进行分析。   
    
**1 S3C2410调色板技术概述   
　　1．1 调色板的概念**   
　　在计算机图像技术中，一个像素的颜色是由它的R，G，B分量表示的，每个分量又经过量化，一个像素总的量化级数就是这个显示系统的颜色深度。量化级数越高，可以表示的颜色也就越多，最终的图像也就越逼真。当量化级数达到16位以上时，被称为真彩色。但是，量化级数越高，就需要越高的数据宽度，给处理器带来的负担也就越重；量化级数在8位以下时，所能表达的颜色又太少，不能够满足用户特定的需求。   
　　为了解决这个问题，可以采取调色板技术。所谓调色板，就是在低颜色深度的模式下，在有限的像素值与RGB颜色之间建立对应关系的一个线性表。比如说，从所有的16位彩色中抽取一定数量的颜色，编制索引。当需要使用某种彩色时，不需要对这种颜色的RGB分量进行描述，只需要引用它的索引号，就可以使用户选取自己需要的颜色。索引号的编码长度远远小于RGB分量的编码长度，因此在彩色显示的同时，也大大减轻了系统的负担。   
　　以256色调色板为例，调色板中存储256种颜色的RGB值，每种颜色的RGB值是16位。用这256种颜色编制索引时，从OOH～FFH只需要8位数据宽度，而每个索引所对应的颜色却是16位宽度的颜色信息。在一些对色彩种类要求不高的场合，如仪表终端、信息终端等，调色板技术便巧妙地解决了数据宽度与颜色深度之间的矛盾。   
    
**1．2 S3C2410中的调色板**   
　　**ARM9**核的S3C2410芯片可通过内置的LCD控制器来实现对LCD显示的控制。以TFT LCD为例，S3C2410芯片的LCD控制器可以对TFT LCD提供1位、2位、4位、8位调色板彩色显示和16位、24位真彩色显示，并支持多种不同的屏幕尺寸。   
    
　　**S3C2410**的调色板其实是256个16位的存储单元，每个单元中存储有16位的颜色值。根据16位颜色数据中，RGB分量所占位数的不同，调色板还可以采取 5：6：5(R：G：B)和5：5：5：1(R：G：B：1)两种格式。当采用5：6：5(R：G：B)格式时，它的调色板如表1所示。   
  
　　表1中，第一列为颜色索引，中间三列是R，G，B三个颜色分量对应的数据位，分别是5位、6位和5位，最后一列是对应颜色条目的物理地址。当采用5：5：5：1(R：G：B：1)格式时，R，G，B三个颜色分量的数据位长度都是5位，最低位为1。   
    
　　用户编程时，应首先对调色板进行初始化处理(可由操作系统提供的驱动程序来完成)，赋予256色调色板相应的颜色值；在进行图像编程时，可以将图像对象赋予所需的颜色索引值。程序运行时，由芯片的**LCD控制器**查找调色板，按相应的值进行输出。S3C2410芯片图像数据输出端口VD[23：O]有24位，当使用不同的色彩深度时，这24位数据可以表示一个或多个点的颜色信息。   
    
**1．3 调色板颜色的选择**   
　　调色板中颜色的选择可以由用户任意定义，但为了编程方便，颜色的选取应遵循一定的规律。例如在Windows编程中，系统保留了20种颜色。另外，在Web编程中，也定义了216种Web安全色，这些颜色可以尽量保留。2S3C2410调色板在嵌入式Linux系统下的使用ARM实现图像显示时，由LCD控制器将存储系统中的视频缓冲内容以及各种控制信号传送到外部LCD驱动器，然后由LCD驱动器实现图像数据的显示。实际应用中，常通过驱动程序由操作系统对寄存器、调色板进行配置。以Linux 2．4内核为例，对调色板的配置是在驱动程序S3C2410fb．c中完成的。   
    
　　在一些公司Linux源码包的S3C2410fb．c文件中，并没有对调色板进行配置，因此在8位以下的显示设置下。LCD不能正常工作。若需要使用调色板，必须对此文件进行修改。   
 **2．1 驱动程序的修改**   
　　查S3C2410数据手册，调色板的物理起始地址为0x4d000400，应先将调色板的物理地址映射到内核中的虚拟地址，然后对其进行赋值。具体步骤如下：   
    
　　(1)在S3C2410．h文件中添加：   
    
　　#define MYPAL(Nb)\_\_REG(Ox4d000400+(Nb)\*4)   
    
　   其作用是实现物理地址到虚拟地址的映射。   
    
　　(2)在S3C24lOfb．h文件，通过下列语句定义256种颜色。   
    
　　static const u\_short my\_color[256]={0x0000，0x8000，…}：   
    
　　数组中的每个16位二进制数表示一种颜色，RGB分量采用的是5：6：5格式。   
    
　　(3)在S3C2410fb．c文件的S3C2410fb-activate\_var(…)函数中，通过下列语句对这256个调色板进行赋值。   
    
　　(4)另外，注意改变LCD控制寄存器LCDCON1的BPPMODE值，设定为需要的颜色深度。   
    
　　(5)重新编译内核，烧写内核。   
    
**2．2 应用程序的编写**   
　　当**S3C2410**用于嵌入式Linux操作系统时，其图形功能一般是依靠帧缓存(Frame buffer)实现的。屏幕上的每个点都被映射成一段线性内存空间，通过应用程序改变这段内存的值，就可以改变屏幕的颜色。当色深在16位以上时，用户直接指定颜色的RGB分量；当色深在8位以下时，用户应当指定颜色在调色板中的索引值。   
    
　　当使用MiniGUI等嵌入式图形系统时，只需要将界面元素的颜色值设为所需颜色的索引值即可。例如：   
    
　　WinElementColors[i]=142；   
    
　　就是将WinElementColors[i]的颜色设置为索引号为142的调色板颜色。   
    
**3 结语**   
  　　在笔者开发的某型指挥车仿真终端中，其显示分辨率设置为640×480。如果色深设置为16 b／p，在系统使用时，画面将会出现明显的抖动、不连贯，这是由于芯片的运算负荷过重造成的。如果按本文中提到的方法对显示驱动加以修改，采用8位色深显示，颜色的选取可以满足需要，画面的显示将明显稳定。这说明，在显示分辨率较高，色彩种类要求比较简单的嵌入式应用中，调色板技术是一个非常值得重视的选择