

基于 ARM9 的触摸屏驱动开发及 LCD 显示程序设计

庄育锋 蔡坤 张墨 郭磊 (北京邮电大学 自动化学院 100876)

摘要:选择 HHARM9 - EDU 多功能教学实验系统作为目标平台,并根据嵌入式系统开发的要求,建立了基于目标平台的交叉编译环境。在研究了嵌入式 Linux 的进程管理与调度,中断响应和处理,嵌入式 Linux 的内核实现机制,驱动程序的编写方法等内容后,开发了触摸屏驱动程序,并编写了两个实现不同功能的 LCD 显示应用程序,最后结合硬件平台调试成功。

关键词:嵌入式系统 驱动程序 触摸屏 ARM9 系统

1 序言

随着当今的世界日益信息化,数字化,计算机和网络已经全面渗透到日常生活中的每一个角落,各种各样的新型嵌入式设备在应用数量上已经远远超过通用计算机,提供良好的图形用户界面可以提高设备的使用效率,这一需求已越来越迫切^[1]。而触摸屏的加入更使得轻松操作成为可能。本课题来源于华恒 HHARM9 - EDU 多功能教学实验系统的触摸屏驱动控制 LCD 显示部分内容的进一步改进。

2 嵌入式 Linux 与 ARM 概述

2.1 嵌入式 Linux 系统

Linux 是模块化的操作系统,用户可以方便地删除不需要的模块,大多数嵌入式系统对操作系统的体积非常敏感,Linux 可以根据自己的需要,选择特定的功能模块,自主地搭建嵌入式操作系统。Linux 支持绝大多数 CPU,包括 Intel, MIPS, ASIC, ALPHA, 68K, POWER PC 等。这使 Linux 几乎可以嵌入到各种硬件设备上^[4]。

2.2 ARM 概述

在嵌入式系统应用领域,ARM (Advanced RISC Machines),既可以认为是一个公司的名字,也可以认为是对一类微处理器的统称,还可以认为是一种技术的名字。ARM 公司自 1990 年正式成立以来,在 32 位的 RISC (Reduced Instruction Set Computer) CPU 开发领域不断取得突破,其结构已经从 V3 发展到 V6。

3 嵌入式 Linux 交叉编译环境的构建

这里逐步介绍常用的安装步骤:

(1) 安装 RedHat LINUX 9.0,在配置 Firewall 时就选择 No Firewall。

(2) 配置网络,TFTP 服务和 NFS (Enable Running)。然后配置宿主机 IP:ifconfig eth0 192.168.1.2

TFTP 服务的配置:

在宿主机上执行 setup,选择 System services,将其中的 tftp 一项选中(出现 [*]表示选中),启动 TFTP 服务:

```
service xinetd restart
```

用如下命令查看 tftp 服务是否开通:netstat -al grep tftp 若 TFTP 服务器没有配置成功,需要按照上述步骤重新检查一遍。

NFS 服务的配置:

按 TFTP 的步骤选中 [*] nfs,然后退出 setup 界面返回到命令提示符下。

```
vim /etc/exports
```

将这个默认的空文件修改为只有如下一行内容:

```
/(rw)
```

然后保存退出(:wq),然后执行如下命令:

```
/etc/rc.d/init.d/nfs restart
```

但是这种方法有个弊端,就是 NFS 不是每次都自动启动的,即无法像 linuxconf 命令那样设置成 Automatic 启动方式,所以每次重启都要执行一下:

```
/etc/rc.d/init.d/nfs restart
```

配置完成后,可用如下办法简单测试一下 NFS 是否配置好了:

在宿主机上自己 mount 自己,看是否成功就可以

判断 NFS 是否配好了。

例如在宿主机/目录下执行:

```
mount 192.168.1.4:/mnt
```

然后到/mnt/目录下看是否可以列出/目录下的所有文件和目录,可以则说明 mount 成功,NFS 配置成功。

?./arminst:执行安装程序。

执行完毕后,会在根目录下生成工作目录:/HHARM2410-R3,内含 Linux 内核、应用程序源代码以及各个工具软件。

4 触摸屏驱动与 LCD 显示

4.1 获取落点坐标 get_handpad()

get_handpad(x,y)的功能是获取触摸在触摸面板上的点的位置。如果成功地获取了落点位置,则函数返回 1,否则返回 0。(x,y)为所检测到落点位置的 AD 转换值。在这里作参数的是指向 x 和 y 的指针。请注意关键字 static 的存在,即 buff 中的字符都是静态存储变量。在函数调用结束后其占用的存储单元并不释放,下一次调用时该变量的初始值就是上一次调用结束后的值。从后面可以发现,这是为了保证采样的有效性。如果两次采样一致,表明该采样数值有效,可以把确认标记发送到 LCD 上去,否则必须重新采样。因此,在程序中设置了一个标志 flag(相当于后述的 X, Y 坐标的 AD 采样成功的 cBuffer.pressure 标志)表示采样是否有效。

```
int get_handpad(unsigned short *x, unsigned short *y)
{
    for(i=0;i<3;) //三次取值;
    {
        TS_RET_HANDPAD cBuffer; //定义结构体 cBuffer;
        read(screen_tp_fd, &cBuffer, sizeof(TS_RET_HANDPAD)); //读取采样值
        if(cBuffer.pressure)
        {
            //前述的 X,Y 坐标的 AD 采样成功的标志
            xa[i] = cBuffer.x;
            ya[i] = cBuffer.y;
            i++;
        }
    }
}
```

```
*x = (get_average_num(xa[0],xa[1],xa[2]));
*y = (get_average_num(ya[0],ya[1],ya[2]));
//取三次采样的平均
return 1;
}
```

4.2 LCD 校准程序

实践发现,在取点函数 get_handpad(&x, &y) 中得到的点的坐标(x,y)值经常超出范围,加了一条语句:

```
printf("x=%d, y=%d\n",x,y);
```

之后再触摸笔任意勾画所得到的(x,y)经常溢出 LCD 显示屏的范围,多为抖动产生,而仔细观察发现 x,y 的最大值分别是 240 和 480。

同时,触摸笔产生的移动轨迹也有漂移,随着 y 值的增大,漂移的距离也越来越大,而 x 值正常显示。

鉴于此,首先规定 x 和 y 的范围分别小于 250 和 500,依次去抖,然后做了一个分段函数,按照差值从小到大给 x,y 依次重新赋值 vx,vy,并以计算后得到的 vx,vy 作为实际点的坐标,同时在终端中显示出来。

```
int main(void)
{
    .....
    while(1)
    {
        get_handpad(&x,&y); //获取落点坐标
        //依据 y 的值,分段,做差值去抖
        if(x < 250)
        {
            //判断 y 值是否越界
            if((y < 30) || (y > 480)) continue;
            if(y <= 80) z = y - 30;
            else if(y < 160) z = y - 40;
            else if(y < 200) z = y - 50;
            else if(y < 240) z = y - 70;
            else if(y < 280) z = y - 90;
            else if(y < 320) z = y - 100;
            else if(y < 360) z = y - 115;
            else if(y < 400) z = y - 130;
            else if(y < 480) z = y - 150;
        }
        //调整后画线
        line(x,z,x+1,z+1);
    }
}
```

```

printf(" x = % d,y = % d\n",x,y)
vx = x;
vy = y;
}
}
|

```

4.3 判断是否连笔

通过判断两个落笔点的间距大小来进行判断。如果间距过大则不连笔,若间距较小则作直线。

```

static unsigned short oldx,oldy,err;
//err 为静态变量,利用起判断是否连笔;
if( ( vx < 250) &&( vy > 90) &&( vy < 260) )
//定义在笔形输入区有效;
{
printf(" vx = % d,vy = % d\n", vx,vy);
err = ( oldx > vx) ? oldx - vx:vx - oldx;
err + = ( oldy > vy) ? oldy - vy:vy - oldy;
if( ( err > 0) &&( err < 20) )

```

//判断两个落点的间距大小,过大则不连笔;较小则作直线。

```

//20 这个参数经几次修改发现最为合适。
lineto( oldx = vx,oldy = vy);
}
else{
moveto( oldx = vx,oldy = vy);
}
}

```

5 结论

校准程序的完成,使得可以得到比较准确的坐标点,接下来只是如何去运用了。最后我做了两个 LCD 显示应用程序。

第一个应用程序设计出了软键盘界面,整个界面分为三部分,尺寸是 240 * 320。上面 120 * 240 部分是软键盘区,排列有 10 个阿拉伯数字和 26 个英文字母。用触摸笔点击对应的字符,则在下面 20 * 240 的数据输出区会显示相应字符。中间的部分是绘图区,可以实时捕捉触摸笔的移动,并在终端里实时显示点的坐标值,如图 1 所示。

第二个应用程序是将一个 ppt 整理成 240 * 320

大小的位图文件,并采用 16 位真彩模式显示在 LCD 上,可以触摸笔来代替鼠标,点击相应的链接处,再由触摸屏捕捉点击产生的命令,按照程序设计的逻辑顺序来切换位文件,如图 2 所示。此程序的功能就很接近工业上的实际应用了,通过人手指或触摸笔点击屏幕上特定位置图标产生控制命令,轻松而方便,具有一定的实际意义。

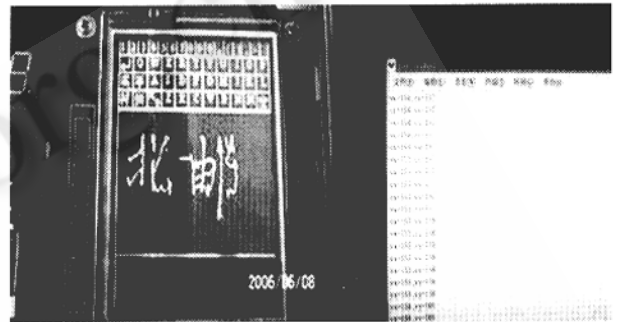


图 1 输入汉字“北邮”及终端显示

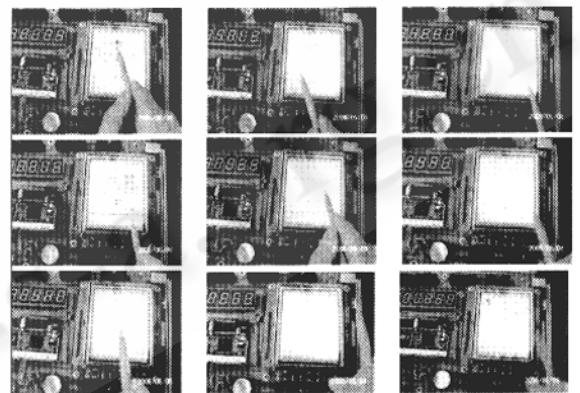


图 2 触摸屏版 ppt

参考文献

- 1 张娟、张雪兰,嵌入式 Linux 的 GUI 应用程序的实现. 计算机应用. 2003;115 ~ 117.
- 2 李驹光、聂雪媛、江泽明等,ARM 应用系统开发详解—基于 S3C4510B 的系统设计,清华大学出版社, 2003,12:3 ~ 11.
- 3 <http://www.hhcn.com/chinese/overview.html>, 华恒公司主页.
- 4 张超,基于 ARM 和 uClinux 的嵌入式系统的构建研究,西安电子科技大学硕士学位论文,2005;3 ~ 5.

(下转第 89 页)

(上接第 98 页)

- 5 李善平、刘文峰等编著, Linux 与嵌入式系统, 清华大学出版社, 2002: 6 ~ 14.
- 6 胡希明、毛德操, 嵌入式系统——采用公开源码和 StrongARM/Xscale 处理器, 浙江大学出版社, 2003, 10: 34 ~ 75.
- 7 HHARM2410 tech manual - v1. 3. pdf, 华恒公司技术文档, 2003: 13 ~ 21.
- 8 Alessandro Rubini 著, Lisoletg 译, Linux 设备驱动程序, 中国电力出版社, 2000: 105 ~ 139.
- 9 李玉波、朱自强郭军, Linux C 编程, 第 2 版, 清华大学出版社, 2006: 130 ~ 148.
- 10 EXP19 - touchpanel - v1. 0. pdf, 华恒公司技术文档, 2003: 1 ~ 6.
- 11 s3c2410 处理器手册, pdf. 2003: 480 ~ 489.