

# 基于 uClinux 触摸屏的设计

殷惠莉<sup>1</sup>, 刘少君<sup>2</sup>, 黄道平<sup>2</sup>

(1. 华南农业大学工程学院, 广东省广州市 510640;

2. 华南理工大学自动化科学与工程学院, 广东省广州市 510640)

**【摘要】** 介绍了 AD7843 芯片及其在硬件电路中的配置, 系统 CPU 选用 Motorola 公司 ColdFire 系列中的 MCF5272, 根据 MCF5272 芯片的 QSPI 口和 AD7843 芯片的读写特性, 设计了触摸屏的硬件电路, 基于差分模式的优点, 在硬件电路中, 将 AD7843 配置为差分模式; 在软件方面给出了基于 uClinux 操作系统的触摸屏驱动程序的整体设计, 并重点介绍了驱动程序中的中断处理程序。

**关键词:** uClinux, 触摸屏, 驱动程序, 中断

**中图分类号:** TM571.6

## 0 引言

触摸屏技术的应用大大简化了嵌入式系统的输入模式, 使用者不必事先接受训练和对系统有相当了解, 仅需用手指触摸显示器屏幕, 即可启动系统、查询资料、分析数据。因此, 带有触摸屏的嵌入式系统以其方便实用、简单灵活深受广大用户的喜爱, 有着广阔的前景。Linux 操作系统功能强大, 具有极高的性价比, 当前在嵌入式系统中的应用最为广泛。

本文结合当前嵌入式系统的热点, 介绍一种基于 uClinux 操作系统的触摸屏设计。

## 1 触摸屏硬件的实现

目前触摸屏技术大致可分为红外线式、电阻式、表面声波式和电容式 4 种。本嵌入式系统使用的是四线电阻式触摸屏, 它是由两个透明电阻膜构成的, 在它的水平和垂直电阻网上施加电压, 就可通过 A/D 转换面板在触摸点测量出电压, 从而得到对应的触摸屏水平和垂直方向上的坐标值。本文将触摸屏水平方向称为 X 方向, 其垂直方向称为 Y 方向。

### 1.1 AD7843 芯片及其在硬件电路中的配置

AD7843 是电阻式触摸屏的转换接口芯片, 是一款具有同步串行接口的 12 位取样 A/D 转换器, 在 125 kHz 吞吐速率和 2.7 V 电压下的功耗为 750  $\mu$ W, 而在关闭模式下的功耗仅为 0.5  $\mu$ W。因此, AD7843 具有低功耗和高速率等特性。

当按下触摸屏时, 有两种情况可影响接触点的电压: 一种是当触摸到显示屏时, 会导致触摸屏外层振动; 另一种是由触摸屏顶层和低层之间的寄生电容引

起的电流振荡以及在 AD7843 输入引脚上引起的电压振荡。这两种情况都可导致触摸屏上的电压发生振荡以及使达到稳定的时间变长。AD7843 有差分 and 单端两种工作模式, 对转换后的精度和可靠性有一些影响。如果将 A/D 转换器配置为读绝对电压方式(单端模式), 那么驱动电压的下降将导致转换输入数据的错误。如果配置为差分模式, 则可以避免上述错误。差分模式具有以下两个优点:

a) 能够用很长的设置时间处理触摸屏, 即触摸屏电压可以有足够的时间稳定下来;

b) AD7843 通过快速时钟可以进入低功耗模式, 从而可以节约能量。

基于差分模式的优点, 在硬件电路中将 AD7843 配置为差分模式。

### 1.2 触摸屏硬件电路的实现

本嵌入式系统选用的 CPU 是 Motorola 公司 Cold-Fire 系列中的 MCF5272, 其集成度高, 带有序列串行外围设备接口(QSPI)。MCF5272 的串行外围设备接口(SPI)多达 16 个, 形成了一个 SPI 序列组。SPI 总线技术是 Motorola 公司推出的一种同步串行接口, 是一种三线同步总线, 因其硬件功能很强, 所以, 与 SPI 有关的软件就相对简单, 使 CPU 有更多的时间处理其他事务。SPI 用时钟(SPI-CK)、数据输入(SPI-IN)、数据输出(SPI-OUT)3 根线进行数据传输, 片选信号(CS#)用于控制器件的选通。

AD7843 操作时序如图 1 所示。从波形图可以看出, 其时序也是一种三线同步方式: 同步时钟 DCLK, DIN 和 DOUT 都是以 DCLK 为基准的, 分别为 AD7843 的数字信号值输入和输出, 这种操作时序正好与 QSPI 的操作时序匹配, 所以在此系统中选用 QSPI 作为 AD7843 的控制接口。

收稿日期: 2003-09-18; 修回日期: 2004-02-04

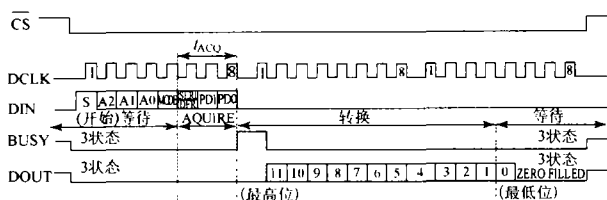


图1 AD7843 操作时序

这样,根据 SPI 和 AD7843 的读写特性设计出了 MCF5272 与 AD7843 的电路。触摸屏硬件电路如图 2 所示。

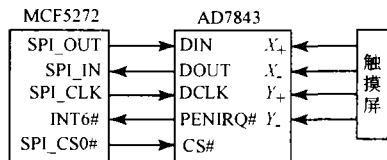


图2 触摸屏驱动电路

在图 2 中, MCF5272 通过 SPI 与 AD7843 相连, AD7843 外接一个四线的电阻式触摸屏。触摸动作完成的过程为:当手指接触触摸屏时, PENIRQ# 置为低电平,从而引发 INT6# 中断,通知 MCF5272 有触摸动作到达, MCF5272 随后将 CS# 置为低电平,开始对 AD7843 进行操作, SPI\_CLK 便有振荡频率输出(此频率由 CPU 总线频率分频得到),与 SPI\_CLK 同步,用 MCF5272 的 SPI\_OUT 向 AD7843 的 DIN 脚发出一串 8 位的命令;接着, SPI\_IN 便可以从 DOUT 脚接收 12 位的 A/D 转换信号,也就是 X 或 Y 方向的坐标值,参照图 1 的 CS#、DIN 和 DOUT,从而精确得到触摸点的位置。在系统中,触摸屏对应的液晶屏像素的大小为 640 × 480,触摸屏的 A/D 转换精度为 12 位,亦即其 A/D 转换值范围为 0 ~ 4 096。对于 X 方向,假设液晶屏 0 点对应 0x160, 640 处对应于 A/D 转换值的 0xfe0,当手指按在触摸屏上时,假定测得 X 方向的 A/D 转换值为 0x870,这样通过下式便可得到触摸点 X 的位置为 372。

$$x = \frac{0x870}{0xfe0 - 0x160} \times 640 = 372 \quad (1)$$

同理也可以计算出 Y 的坐标值。

## 2 触摸屏驱动程序的设计

### 2.1 触摸屏驱动程序的整体设计

uClinux 操作系统下的设备驱动程序分为 3 种方式:字符设备、块设备和网络设备。其中字符设备以字节为单位,只能按顺序访问,不用缓存。这里,触摸屏驱动程序采用字符设备方式,其结构如图 3 所示。其

中:中断处理程序完成对触摸屏触摸点的位置读取;Ts\_open() 给触摸屏驱动程序分配中断源;Ts\_init() 对 MCF5272 的 SPI 功能寄存器进行初始化;Ts\_ioctl() 函数的功能是把从中断处理程序中获得的 X、Y 方向的 A/D 转换值赋给用户空间的程序。

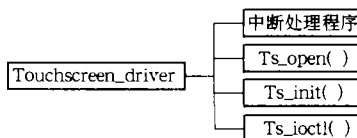


图3 字符设备结构

在本系统中,对触摸屏操作过程为:当启动 uClinux 操作系统时,完成其触摸屏驱动程序的初始化,即完成 Ts\_init() 函数中的功能;当 uClinux 操作系统用户空间的程序打开触摸屏驱动程序时,即执行驱动程序中 Ts\_open() 函数,完成对中断的分配;当有触摸动作时,立即产生硬件中断,中断处理程序工作,读出触摸屏相应位置对应的 A/D 转换值。用户空间的程序调用驱动程序的 IO 控制程序 Ts\_ioctl() 函数时,读取 A/D 转换值到用户空间。在用户空间经过转换,将 A/D 转换值转换为具体的 X 和 Y 方向的坐标值,赋予图形库中的鼠标处理程序,借助于图形库的鼠标功能程序便可在图形界面上实现触摸功能。

### 2.2 触摸屏的中断处理程序

触摸屏的驱动程序核心是用中断处理程序。响应中断并提供相应服务的程序称为中断服务程序。在中断服务程序结束之前,来自同一个设备的后续中断都将丢失。因此,中断服务程序应该尽快完成执行,否则就有可能丢失来自设备的另一个中断。因整个触摸屏的驱动程序处理比较复杂,而且耗时较长(大约几十毫秒),因而触摸屏驱动程序不可能在中断服务程序中完成。

在 uClinux 操作系统中,中断处理程序可以分为上半部(top half)和下半部(bottom half)。上半部是一般的中断服务程序,由硬件中断触发,应当尽可能短小,处理尽可能快;而下半部是单独的一段处理程序,一般将其放入立即队列中以便快速执行。在此系统中,按触摸屏时,从 AD7843 输出的数值有一个抖动的过程,即从 AD7843 输出的 A/D 转换值有一个不稳定时期,这个过程大约为 10 ms。所以中断处理程序的下半部处理没有采用立即队列,而是采用定时器,使下半部在中断发生 10 ms 后再作处理,这样有效地避开 AD7843 输出值的不稳定时期,使中断服务程序和中断处理任务串行化,达到了处理时间较长的触摸屏动作的目的。

定时器是 uClinux 操作系统内核中最终的计时资源。定时器调度函数(定时器处理程序)可在某个特定的时间执行。与任务队列不同,可以指定定时器函数何时被调用,但不能确定任务队列中的任务何时执行;与任务队列相似,内核定时器注册的处理函数只执行一次。

定时器列表的数据结构为<sup>[2]</sup>:

```
struct timer_list{
    struct time_list * next,prev;
    unsigned long expires;
    //timeout 超时值,以 jiffies 为单位
    unsigned long data;
    //传递给定时器处理程序的参数
    void (* function)(unsigned long);
    //超时调用的定时器处理程序
};
```

定时器的 timeout 值是个 jiffies 值, jiffies 在 uClinux 操作系统中以 10 ms 为单位, 1 个 jiffies 为 10 ms。当 jiffies 值  $\geq$  timer  $\rightarrow$  expires 时,就运行定时器处理程序 timer  $\rightarrow$  function 函数。在触摸屏驱动程序中,中断服务程序主要功能就是加载定时器处理程序和指定其执行的时间点。定时器处理程序即中断处理程序下半部在中断服务程序响应大约 10 ms 后执行,亦即在中断服务程序响应 1 个 jiffies 后执行。

定时器处理程序是触摸屏驱动程序的核心部分。它的主要任务为:通过 AD7843 的 X<sub>+</sub>、X<sub>-</sub>、Y<sub>+</sub>、Y<sub>-</sub> 引脚分别读取触摸屏 X 轴和 Y 轴的 A/D 转换坐标值,并对其进行判断,获得真实结果,达到对触点定位的目的。定时器处理程序流程如图 4 所示。

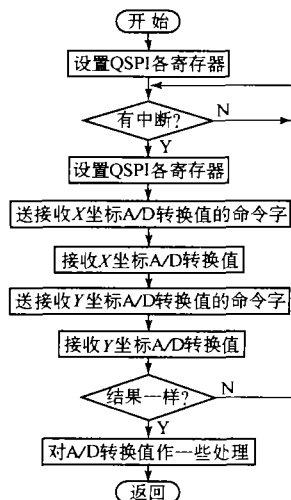


图 4 定时器处理程序流程

定时器处理程序的过程为:首先, MCF5272 根据具体需要对 SPI 的功能寄存器进行配置,再通过 SPI

的 SPI\_OUT 向 AD7843 的 DI 脚发送一个接收 A/D 转换值的 8 位命令字,然后,通过 SPI 的 SPI\_IN 从 AD7843 的 DOUT 脚接收 12 位 X(Y) 方向的 A/D 转换值,依次循环 4 次,接收 2 组 X(Y) 方向的 A/D 转换值,进行比较,判断该值是否是其真实值。

### 3 结束语

在触摸屏的设计过程中,作者对其做了大量的实验调试工作。触摸屏的抗干扰设计是整个设计工作中的难点和重点,它直接关系到触摸屏的工作性能。在硬件设计方面,实践证明差分模式比单端模式效果好,输出的结果要准确;软件方面,针对中断处理程序的下半部,开始采用的是 Linux 驱动程序中常用的立即队列,发现有时从 AD7842 输出的 A/D 转换值不准确,也就是在触摸屏电压抖动的过程中读取了电压 A/D 转换值,没有避开触摸时 A/D 转换输出值的不稳定时期。后来,根据实际情况,采用了定时器作为中断处理程序的下半部,经过反复实验,发现在触摸中断到达 10 ms 后,再读取电压 A/D 转换值,效果最好。另外,在整个调试过程中,作者始终运用示波器跟踪 AD7843 的操作时序,对调试结果进行分析,加速了调试进度。该触摸屏设计已用于实际的嵌入式产品中,该产品运行效果好,性能稳定,具有很好的市场前景。

#### 参 考 文 献

- [1] Analog Device, Inc. Touch Screen Digitizer AD7843. <http://www.analog.com>, 2002
- [2] Alessandro Rubini. LINUX 设备驱动程序. 北京:中国电力出版社, 2000

## 电子工程师

月刊

欢 迎 订 阅

全国各地邮局均可订阅

邮发代号:28-251

编辑部电话:025-83772588

E-mail: radarnet @ public1. ptt. js. cn