

# 基于 $\mu\text{C}/\text{OSII}$ 操作系统的声音监控设计与实现

卢媛媛<sup>1,2</sup>

(1. 黄冈师范学院 校园网管理中心, 湖北 黄冈 438000;

2. 华中科技大学 软件学院, 湖北 武汉 430074)

**摘 要:** 嵌入式系统处于单独应用的阶段, 以 MCU 为核心, 与一些监测、伺服、指示设备配合实现一定的功能。嵌入式视频监控存在死机、服务无响应等故障, 导致无法长期有效地提供服务, 系统可用性能低。借鉴商业高端服务器领域一些成功的高可用性的建设方案, 结合目前嵌入式领域健壮性技术, 在保证不增加硬件成本的前提下, 设计一个声音监控的高可用实施方案。文中阐述如何选择支持嵌入式的硬件系统及怎样将  $\mu\text{C}/\text{OSII}$  嵌入式操作系统移植到 Intel X86 来开发监控系统。此系统可以定时和实时监控声音, 具有查询声音日志、用户注册等功能。

**关键词:** 声音监控系统; 嵌入式操作系统;  $\mu\text{C}/\text{OSII}$ ; 硬件方案; 软件方案

中图分类号: TP277

文献标识码: A

文章编号: 1673-629X(2006)08-0163-04

## Design and Realization of Audio - Monitor System Based on $\mu\text{C}/\text{OSII}$

LU Yuan-yuan<sup>1,2</sup>

(1. Network Administrative Center of Huanggang Normal College, Huanggang 438000, China;

2. School of Software Engineering, Huazhong University of Science and Technology, Wuhan 430074, China)

**Abstract:** The applications of embedded system are still stay in the separate phases. The MCU is the core and it cooperates with monitoring, servo and indicating equipments to realize some functions. But an embedded audio - monitor system has got the malfunctions of crash and services non - response, thus it's unable to provide services effectively in the long term, and result in a low - performance system. Here the author borrows some successful and highly available ideas and plans from commercial high - end servers' areas, combining the advanced technology in embedded system, ensuring no additional costs for hardware, and designs a highly available sound monitoring program. In this paper, the  $\mu\text{C}/\text{OSII}$  embedded operating system migrated to Intel X86 for developing monitor - control system. This system can monitor audio by timing or real - time, query audio log and users registration.

**Key words:** audio - monitor system; embedded system;  $\mu\text{C}/\text{OSII}$ ; hardware solution; software solution

## 0 引 言

伴随着 20 世纪 90 年代末计算机网络的成熟发展, 到 21 世纪, 人类进入了所谓的后 PC 时代。在这一阶段, 人们开始考虑如何将客户终端设备变得更加智能化、数字化, 从而使得改进后的客户终端设备轻巧便利、易于控制或具有某些特定的功能。为了实现人们在后 PC 时代对客户终端设备提出的新要求, 嵌入式技术 (Embedded Technology) 提供了一种灵活、高效和高性价比的解决方案。目前大多数嵌入式系统还处于单独应用的阶段, 以 MCU 为核心, 与一些监测、伺服、指示设备配合实现一定的功能。Internet 现已成为社会重要的基础信息设施之一, 是信息流通的重要渠道, 将嵌入式系统与 Internet 结合起来的想法其实很早以前就有了, 主要的困难在于, In-

ternet 上面的各种通信协议对于计算机存储器、运算速度等的要求比较高, 而嵌入式系统中除部分 32 位处理器以外, 大量存在的是 8 位和 16 位 MCU, 而支持 TCP/IP 等 Internet 协议将占用大量系统资源, 或根本不可能<sup>[1,2]</sup>。

声音监控系统是一种操作简单、成本低廉的软件产品。它继承了网络和监控的优点, 是一种基于信息网络的、具有交互功能的产品, 可以通过网络与信息中心进行数据双向传递。

## 1 系统设计及其实现方案

### 1.1 系统设计

现今的微机教室不再是以前的机房密闭, 只能靠视频才能监控, 现已经将普通墙壁改为可视的玻璃, 能随时查看情况, 减低了使用成本 (不需配价格昂贵的视频采集卡), 并且每个教室配备了监控管理人员。对于监控人员而言, 能方便地将自己的控制信息实时地传递给所关心的服务端设备, 并以直观、快捷的方式获取所关心设备的实时和定时状态参数及信息是最理想的。

收稿日期: 2005-11-16

基金项目: 黄冈师范学院青年科研基金项目 (05cq96)

作者简介: 卢媛媛 (1981-), 女, 湖北荆门人, 助教, 硕士研究生, 主要从事软件工程和校园网管理研究。

基于上述考虑,研究并设计了一种基于嵌入式 Web 服务器的声音监控系统,本系统具有如下功能<sup>[3]</sup>:

上局域网的用户通过客户端界面对存储在教室内嵌入式 Web 服务器上串行 EEPROM 中的网页进行访问,实现对挂接在嵌入式 Web 服务器上的设备进行控制,并且能将设备的状态信息经由嵌入式 Web 服务器返回给用户的浏览器,从而达到监控的目的。另外,系统应具有根据口令访问功能,只有被授权的用户才能对本系统进行访问,以保障系统的安全运行。系统采用浏览器作为通用的程序界面,客户端只要有通用的浏览器就可以显示程序的界面,而且客户端的数量基本上是没有限制的,所以程序的维护与修改只与服务器有关,而与客户端无关,这样极大地便利了程序的维护。同时采用 TCP/IP 协议,既可以在计算机之间构成局域网,也可便于以后接入 Internet 实现远程监控。

## 1.2 硬件方案

### 1.2.1 系统设计

本系统选用嵌入式  $\mu C/OSII$  系统作为一般的声音采集系统,也可以加上网络功能成为远程监控系统。

根据功能需求,系统主要分成 3 个部分:

#### (1) 声音采集部分。

这个部分主要是麦克风,由它来传输声音,再把声音传输到服务端的数据卡上,进入嵌入式  $\mu C/OSII$  系统中进行处理。

#### (2) 嵌入式 $\mu C/OSII$ 系统部分(Server)。

这个部分就是嵌入式系统,用来处理数据和存储处理好的声音数据,并且通过 TCP/IP 协议,还需把处理好的数据传输到监视器上。

#### (3) 监控系统部分(Monitor)。

把从嵌入式  $\mu C/OSII$  系统传输过来的数据显示在监视器上,可以在监控系统的用户界面上进行实时操作。

### 1.2.2 硬件平台

#### (1) 微处理器。

$\mu C/OSII$  可以运行于多种处理器上,如 ARM, AXP, Intel X86 等等。但 Intel X86 是最合适的,它是一个非常容易学习的平台,可以在 PC 上学习软件的开发,学习硬件结构。因此,采用 Intel X86 微处理器作为本系统的微处理器,可以利用大量现成的开发工具和 BIOS,从而能在最短的时间内完成产品的开发。最终,笔者选择了 Pentium 133 CPU。 $\mu C/OSII$  的移植性非常好,系统平台之间可以相互移植,将来可以考虑向其它的系统平台移植。

#### (2) 声音数据采集卡。

随着计算机接口技术和数字信号处理技术的发展,可以利用高速数据采集技术和数字信号处理技术,将其工作时的声音和信号实时或定时高速采集。那么,数据采集通

常是基于计算机的设备。内置的采集卡通过计算机总线和计算机连接,如 PCI, PCMCIA 总线等等。外置的数据采集设备通常通过串口、IEEE-488 和 USB 等与计算机连接。通过认真考证,本系统选择 30MS/s 同步数据采集多功能卡 PCI-1714。最为重要的是 PCI-1714 提供板载自动校正功能,使用者只要通过软件指令,就可以启动 PCI-1714 上的自动校正功能,完成模拟输入信道的校正工作,不需要任何繁杂手工操作。PCI-1714 在触发数据采集方面,提供多种选择方式:触发的来源包括软件触发、内部时钟同步触发以及外部触发等,模拟触发功能提供多种触发条件的选择,数字触发则提供上升沿触发与下降沿触发两种选择。在多卡同步功能方面,PCI-1714 可以接受外部输入时钟进行同步数据采集,所以使用者可以使用同一外部输入时钟达成多卡同步采集功能。

#### (3) 存储方式。

网络 BIOS 方式为无盘系统准备的调用操作系统的方式,在 TCP/IP 协议中开发了 BOOTP 协议。该协议利用客户机/服务器方式,将本地待启动的机器视为客户端,远端启动一个服务器设备,通过网卡将软件系统从远端服务器传递过来。该方式可以在系统启动时动态加载操作系统和应用程序,软件配置可以相当的灵活。

#### (4) 主板。

上面提到,系统的主板要支持 Pentium 133 的 CPU, BIOS, PCI 和 IDE 总线,还要支持网卡设备。另外,嵌入式系统的主板的体积一定要小,因此考虑采用 Embedded PC 主板,它体积小(5.25 英寸)、性能好(Socket 370, 支持 Intel Pentium 处理器)、功能强(256 内存, 自带网卡, 支持 IDE、PCI 接口、LCD 接口),可以满足嵌入式系统要求。

### 1.2.3 软件方案

#### 1.2.3.1 嵌入式系统体系

整个嵌入式系统体系结构如图 1 所示。

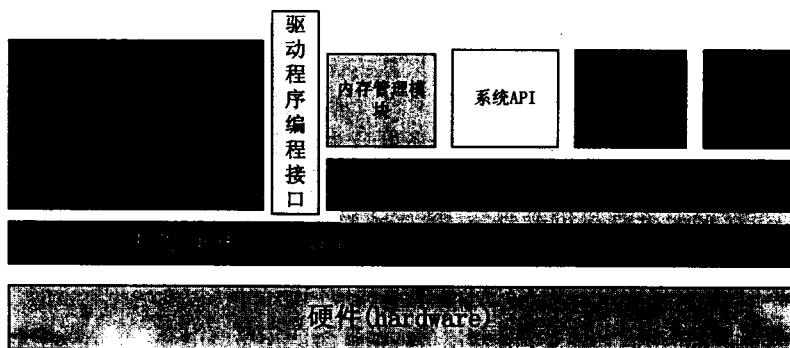


图 1 系统体系结构图

硬件抽象层(HAL)包含了所有和硬件平台相关的代码,如上下文切换和 I/O 寄存器访问等等。它存在于整个软件体系结构的最底层,直接访问和控制硬件,对其上的系统软件层提供机器无关代码和提供访问和控制服务。这部分的代码一般由硬件设备厂家提供。这样极大地缩短了驱动开发时间。 $\mu C/OSII$  是开放的设计,很多软件包比如网络模块可以采用现成通用的 TCP/IP 协议栈(如图

外的 LWIP,国内的 Skyeye)。这样只需要着重实现声音监控系统的功能即可。

### 1.2.3.2 软件方案概述

本系统采用  $\mu\text{C}/\text{OSII}$  操作系统,在 X86 平台上用 C 语言开发,采用 BC++3.1 编译器编译。

在 main 中建立一个起始任务

StartTask。

```
int main(void)
```

```
{
    OSInit();
    OSTaskCreate((void *)StartTask,
        (void *) 0x00, (OS_STK)
        &StartTaskStk[START_TASK_STK-
        SIZE-1],0);
    return 0;
}
```

起始任务中,首先建立一系列的信号量和邮箱<sup>[4]</sup>:

```
Sem_Int_Sound = OSSemCreate(0); /* 唤醒声音采集任务,每
一路创建一个 */
Sem_Int_Network = OSSemCreate(0); /* 唤醒网络通信任务
*/
Sem_Int_ClockTask = OSSemCreate(0); /* 唤醒时钟中断 */
Sem_Int_Store = OSSemCreate((void *)0); /* 唤醒数据存储
任务 */
Mbox_Any_UI = OSMboxCreate((void *)0); /* 唤醒 UI 任务
*/
Mbox_Any_SMTask = OSMboxCreate(0); /* 唤醒系统配置管理
任务 */
```

然后,用 OSTaskCreate() 函数建立 6 个任务,分别为:

- 1) SoundTask 多路声音采集任务:负责轮询多个声音采集线路,打开或者关上;
- 2) StoreTask 数据存储任务:将来自线路上的声音编码数据存贮到缓冲区中;
- 3) CommuTask 网络通信任务:将数据通过 TCP/IP 协议传送到局域网上;
- 4) SmTask 系统配置管理任务:根据来自用户的输入命令来改变系统配置;
- 5) UITask 用户交互任务:通过 HTTP 协议和用户交互,显示系统状态,接受用户输入;
- 6) ClockTask 时钟任务:定时器,可以由用户设定时间进行某种动作,例如开关某个线路,最后,在开始任务中将自身删除。

### 1.2.3.3 任务分解和系统设计

图 2 为系统内部软件任务分解图(图中,灰色箭头为数据流,黑色箭头为控制流)。

各个任务实现详解:

#### 1) SoundTask 多路声音采集任务。

SoundTask 负责轮询多个声音采集线路,对其进行控制。它接受 SmTask 任务的控制命令,同时将采集来的数

据送到 StoreTask 进行处理。SoundTask 直接控制底层的数据采集卡,它包含了采集卡的硬件驱动程序。当数据流达到存储条件时,调用 OSSemPost(Sem\_Int\_Store)唤醒 StoreTask 数据存储任务。

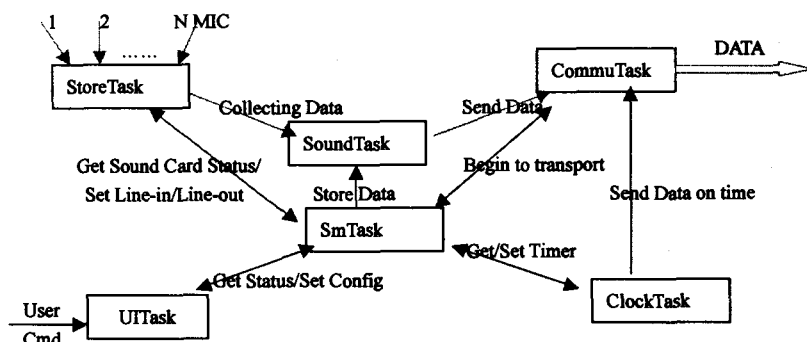


图 2 系统内部软件任务分解图

#### 2) StoreTask 数据存储任务。

StoreTask 将来自线路上的声音编码数据存贮到缓冲区中,它和 SoundTask 一样,直接控制采集卡硬件,因此也包含了一部分采集卡的硬件驱动程序。当它被唤醒时,读取采集卡数据通道的数据,并调用 OSSemPost(Sem\_Int\_Network)唤醒 CommuTask 网络通信任务。

#### 3) CommuTask 网络通信任务。

CommuTask 将数据通过 TCP/IP 协议传送到局域网上。它负责将 StoreTask 传来的数据流打包,并通过网卡传到局域网上。它建立在一个 TCP/IP 的协议栈之上,可以通过 TFTP/FTP/HTTP 等协议,将采集处理后的数据存到上位机(主机)上。同时,它受到 SmTask 和 ClockTask 的控制。SmTask 发送命令,通知它何时开始发送,何时中止;而 ClockTask 则启动定时器来控制它。

#### 4) SmTask 系统配置管理任务。

由系统内部软件任务分解图可以看到,SmTask 是整个系统的控制核心。它处理一切的系统控制命令,并负责协调控制其他各个任务的工作状态。它接受 UITask 的用户命令,同时将系统状态传递给 UITask,调用 OSMboxPost(Mbox\_Any\_UI, Msg\_Any\_UI)唤醒用户界面任务 UITask,并将系统状态数据传递到 UITask 的邮箱中。

#### 5) UITask 用户交互任务。

用户交互采用 B/S 模式,用户端使用网络浏览器如 IE 登陆监控下位机上的嵌入式 Web Server,通过 HTTP 协议来进行配置管理和状态监控。用户交互任务处理以下几类事件<sup>[5]</sup>:

- a. 1-N 线路的状态显示和故障告警。
- b. 1-N 线路的开关设置。
- c. 1-N 线路的定时开/关设置。
- d. 1-N 线路的数据采集和发送。
- e. 1-N 线路的定时数据采集和发送。

每个事件都有特定的编号的数据结构。当用户确定下达命令后,UITask 调用 OSMboxPost(Mbox\_Any\_SMTask, Msg\_Any\_SMTask)唤醒系统配置管理任务 Sm-

Task, 并将用户命令数据传递到 SmTask 的邮箱中。

#### 6) ClockTask 时钟任务。

X86 每 10ms 产生一个中断, 在这个中断服务子程序中, 对 clock\_count 减 1, 当减为 0 时, 就调用 OSSemPost (Sem\_Int - ClockTask) 唤醒时钟任务; 同时, clock\_count 的值重新回到 100。这样每 1s, 就会调用一次 OSSemPost (Sem\_Int - ClockTask)。时钟任务调用 OSSemPend (Sem\_Int - ClockTask, 0, &err) 获得信号量。当获得信号量时, 对设置的定时器进行对比。在定时器时间到时, 触发预先设计好的系统动作。它可以控制 CommuTask 定时发送数据, 也可以通过 SmTask 来控制多路声音采集线路的打开和关闭。

#### 1.2.3.4 用户交互方式设计

本系统采用基于 Browser 的交互方式, 在监控下位机上有嵌入式的 Web Server, 而上位机采用 IE 浏览器访问下位机的 IP 地址, 通过 HTTP 协议对下位机的状态进行监控和监控数据采集 (如图 3 所示)。

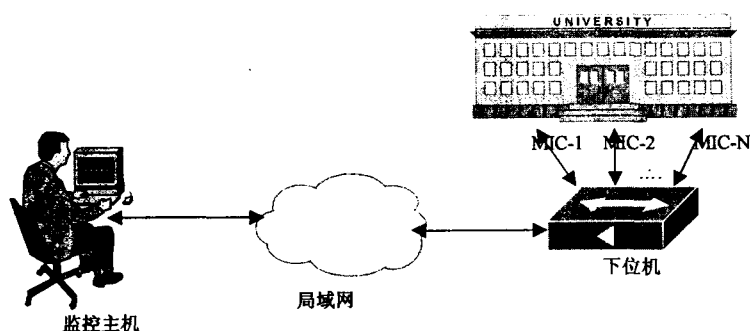


图 3 系统交互方式设计图

## 2 结 论

软件方案用 C 编译器, 将嵌入式实时多任务操作系统  $\mu\text{C}/\text{OSII}$  移植到单片机上, 用 Intel X86 在线仿真器进行仿真。在仿真器仿真完毕后, 再脱机运行, 实验结果令人满意, 达到了预期的结果。经实践证明, 基于嵌入式  $\mu\text{C}/\text{OSII}$  的声音监控的设计, 是具有一定可用性 & 可靠性的。整个系统设计体现了嵌入式  $\mu\text{C}/\text{OSII}$  系统适应性强、体积小、成本低、开放源代码、开发使用容易等特点, 由于使用了  $\mu\text{C}/\text{OSII}$ , 系统的控制逻辑结构清晰, 与普通的单片机系统相比, 在对功能的进一步扩展、移植及接入网络等方面都有着极大的优势。

#### 参考文献:

- [1] 田 泽. 嵌入式开发与应用实验教程[M]. 北京: 北京航空航天大学出版社, 2004.
- [2] Li Qing, Yao C. 嵌入式系统的实时概念[M]. 北京: 北京航空航天大学出版社, 2004.
- [3] 郑家莉, 黄 伟. 无线远程监控系统的核心技术研究[J]. 单片机与嵌入式系统应用, 2004 (6): 89-92.
- [4] Labrosse J J. 嵌入式实时操作系统  $\mu\text{C}/\text{OSII}$  (第 2 版)[M]. 邵贝贝, 等译. 北京: 北京航空航天大学出版社, 2005.
- [5] 刘 奇. 基于  $\mu\text{C}/\text{OS-II}$  接入网接口的研究与设计[J]. 武汉理工大学学报(信息与管理工程版), 2004, 26(2): 31-34.

(上接第 161 页)

```
feedsFeed f = (feedsFeed) feedManager. FeedsTable
[feedNode. Tag];
if(readItems == list. Count){ feedNode. NodeFont =
treeView1. Font;
f. containsNewMessages = false; }else{
feedNode. NodeFont = new Font(treeView1. Font, tree
View1. Font. Style | FontStyle. Bold); f. containsNewMessages =
true; }
RssReaderlistView. EndUpdate(); }
.....
```

## 5 小 结

现代信息技术是影响图书馆发展最深刻的环境因素, 图书馆应随着信息技术的飞速发展, 将一些先进的信息技术结合应用到图书馆中, 虽然 RSS 在中国才刚刚起步, 但 RSS 是新一代互联网的必然发展趋势。图书馆专家和管理人员已经重视 RSS 在图书馆方面的应用, 许多图书馆网站已经或者准备提供 RSS, 更有效地为广大读者提供更方便、更及时的服务。图书馆利用 RSS, 有利于把主动性、人性化服务做到实处, 最终成为开放的、全方位服务的数

字图书馆。

#### 参考文献:

- [1] Cohen S M. A Weblog[EB/OL]. <http://www.librarystuff.net>, 2005-10-02.
- [2] 厦门大学图书馆主页[EB/OL]. <http://library.xmu.edu.cn>, 2005-09-12.
- [3] Darlene F. Why and How to Use Blogs to Promote Your Library's Services [EB/OL]. Marketing Library Services. <http://www.infotoday.com/mls/nov03/fichter.shtml>, 2003-11/12.
- [4] Roddy M. RSS - Not Just For Techies[EB/OL]. Informed Librarian Online. <http://www.informedlibrarian.com>, 2004-06.
- [5] 吴振新. RSS 元数据在门户网站建设中的应用[J]. 现代图书情报技术[J], 2004, 10: 62-66.
- [6] 魏 英, 林怀忠, 陈 纯. RSS 标准在 Internet 环境下新闻发布系统中的应用研究[J]. 计算机工程与应用, 2005 (15): 194-196.