文章编号:1006-2475(2004)04-0017-02

嵌入式操作系统的设计原理与方法

刘玉兰[1],李湖南[2]

(1.广东工业大学应用数学系,广东 广州 510090;2.华南师范大学数学系,广东 广州 510631)

摘要:基于开发Motorola M68VZ328微处理器的嵌入式系统的实际经验,较为详细地探讨了嵌入式操作系统的设计原则、原理、方法与步骤。

关键词:嵌入式操作系统;嵌入式Linux

中图分类号:TP316.89 文献标识码:A

Exploration of Principle and Method in Design of Embedded Operation System

LIU Yu-lan[1], LI Hu-nan[2]

(1. Department of Applied Mathematics, Guangdong University of Technology, Guangzhou 510090, China;
2. Department of Mathematics, South China Normal University, Guangzhou 510631, China)

Abstract: General principles, methods and steps in the design of embedded operation system are explored mainly by author's development experiences.

Key words: embedded operation system; embedded Linux

0 引 言

随着网络、通信、芯片等技术的发展,全球开始进入后PC时代,其特点是未来信息处理和联网的设备将不仅仅是PC机,形式多种多样的Internet应用设备将形成一个巨大的产业。同PC机的发展带动了整个桌面软件产业的发展一样,Internet应用设备的发展也将带动嵌入式软件产业的发展。其中嵌入式操作系统是该产业的龙头和核心。使用嵌入式操作系统的产品形态丰富多样,如掌上电脑、机顶盒、数字电视、数字音像设备、手机、可视电话、车载智能设备、数控机床、商业终端、教育终端等各种数字化产品,因此未来十几年,嵌入式产品面临着难得的市场机会。

1 嵌入式系统的技术特点

嵌入式系统主要由嵌入式处理器、相关支撑硬件和嵌入式软件系统组成,它是集软硬件于一体的可独立工作的"器件"。嵌入式处理器主要由一个单片机或微控制器(MCU)组成。而这些嵌入式CPU目前多是8位和16位的,与32位或64位的高性能处理器相比,具有很强的经济性和实用性。相关支撑硬件包括显示卡、存储介质(ROM和RAM等)、通信设备、IC卡或信用卡的读取设备等。嵌入式系统有别于一般的计算机处理系统,它不具备像硬盘那样大容量的存储介质,而大多使用闪存(Flash Memory)作为存储介质。嵌入式软件包括与硬件相关的底层软件、操作系统、图形界面、通信协议、数据库系统、标准化浏览器和应用软件等。

总体看来,嵌入式系统具有便利灵活、性能价格比高、嵌入性强等特点,可以嵌入到现有任何信息家电和工业控制系统中。从软件角度来看,嵌入式系统具有不可修改性、系统所需配置要求较低、系统专业性和实时性较强等特点。

2 嵌入式Linux操作系统产生的必然性

信息家电商机引发全球嵌入式操作系统平台大战,全球4大操作系统阵营WinCE、Palm OS、EPOC和Linux展开规格战,各拥有软件及硬件合作厂商逐鹿信息家电市场的份额。

微软窗口操作系统拥有在个人电脑上的操作系统占有率的优势,使WinCE拥有强大的窗口资源支援。不过Palm OS操作系统拥有全球PDA产品70%

收稿日期:2003-06-10

作者简介:刘玉兰(1977-),女,江西吉安人,广东工业大学应用数学系助教,硕士,研究方向:二层多目标最优化;李湖南(1977-),男,江西赣州人,华南师范大学数学系助教,硕士,研究方向:代数。

?1994-2016 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

的市场占有率，同时获得 3COM、IBM 和索尼等跨国公司的支持。EPOC 是发展自欧洲的操作系统、是由世界上最大的 3 家移动电话厂商诺基亚、爱立信和摩托罗拉所共同开发、整合组成新公司开发出来的新操作系统。在 3 大电话厂商的合作下，EPOC 市场潜力很大，且占有率高，但应用功能以手机为主，目前并不开放授权。此外，在 3 大主流操作系统品牌外，Linux 也将是今后一股强劲的力量。

选择 Linux 作为一种嵌入式操作系统，有其外因和内因在起作用。外因在于：嵌入式设备的特点是产品形态各异，难以以标准结构统一，资源有限，因此嵌入式设备对操作系统的个性化有较高要求。而且，目前在嵌入式操作系统领域，尚没有哪一个操作系统能形成垄断局面。这为把以 Linux 为代表的开放源代码软件发展为嵌入式操作系统提供了良好契机。

当然，最主要还是 Linux 能为嵌入式系统应用带来许多好处，具体说来：(1)用户和硬件厂商无须交纳巨额运行时间版权费用。提供商收取的是经过打包的全套产品的价格，对运行时间没有另外收取费用。(2)Linux 是开放源码的软件，受 GNU 的 GPI 公约保护，软件源代码可以随意拷贝、散发和使用。随着实时操作系统市场逐渐成熟，开放源代码操作系统将比任何单个厂商提供的操作系统都有优势，当然也就更能适应外界可能发生的变化。(3)Linux 具有体积小巧的特点和丰富的应用程序接口。(4)Linux 的内核是可配置的。我们知道，Linux 的内核是极具弹性的，开发人员可以使用内核配置工具很方便地增加或删去某些功能和服务，像硬件驱动程序、文件系统和网络支持等。这种可配置的特性也可以作为一个独立的模块编译到内核中，运行时内核或者加载这些模块，或者忽略它们。(5)Linux 具有优秀的可扩展性。现有大量的工具、设备和应用软件可供人们定制 Linux 系统功能。各家嵌入式 Linux 提供商都会提供配置工具和占用系统资源很少的有效设备。(6)Linux 的网络功能表现非常出色。(7)应有尽有的驱动程序。随着 Linux 的流行，独立开发商和硬件厂商都开始在新产品中附带面向 Linux 环境的驱动程序，像对 flash 内存和无线网络的支持等。

总之，Linux 以其内核健壮、运行高速、源代码公开等技术优势，使其成为具有生命力的嵌入系统平台。目前，国外不少大学、研究机构和知名公司都加入了嵌入式 Linux 的开发工作，较成熟的嵌入式 Linux 产品不断涌现，如 RT－Linux、Embedix、uClinux 等。

3 嵌入式操作系统的设计原理、开发方法与步骤

鉴于嵌入式 Linux 的以上特点，开发者可以根据开发平台的具体特点，选取合适的版本的嵌入式 Linux 系统，将其移植到开发平台上。具体步骤如下：

(1)选取和分析硬件平台。分析所选的硬件平台的特点，根据它的特点选取某一版本的嵌入式 Linux 作为开发的基础。

(2)精简内核。Linux 的内核是可配置的，开发人员可以根据开发的需要，使用内核配置工具很方便地增加或删去某些功能和服务。

(3)系统启动。修改相应的系统启动文件，使得内核在开发平台上“跑”起来。系统的启动顺序及相关文件在核心源码目录下，看以下几个文件：

./arch/$ARCH/boot/bootsect.s

./arch/$ARCH/boot/setup.s

./init/main.c

boot.s 及 setup.s 这两个程序是 Linux kernel 的第一个程序，包括了 Linux 自身的自启动程序，但是在说明这个程序前，必须先说明一般 IBM PC 开机时的过程(此处开机是指“打开 PC 机的电源”)。

打开电源时，PC 一般是由内存中地址 FFFF:0000(FFFF 为段地址，0000 为偏移量)开始执行，这个地址一定在 ROM BIOS 中，ROM BIOS 一般是在 FE000h 到 FFFFFh 中，而此处的内容则是一个 jump 指令，跳转到另一个位于 ROM BIOS 中的位置，开始执行一系列的动作。

紧接着系统测试码之后，控制权会转移给 ROM 中的启动程序 ROM bootstrap routine，这个程序会将磁盘上的第零扇区读入内存中，至于读到内存的哪里呢？其实是绝对位置 07C0:0000。而位于 Linux 开机磁盘的 boot sector 上的，正是 Linux 的 bootsect 程序。

(4)驱动程序。在 Linux 系统里，设备驱动程序所提供的这组入口点由一个结构来向系统进行说明。设备驱动程序所提供的入口点，在设备驱动程序初始化的时候向系统进行登记，以便系统在适当的进程调用。Linux 系统里，通过调用 register-chrdev 向系统注册字符型设备驱动程序。

在 Linux 中，除了直接修改系统核心的源代码，把设备驱动程序加入核心外，还可以把设备驱动程序作为可加载的模块，由系统管理员动态地加载它，使之成为核心的一部分。也可以由系统管理员把已加载的模块动态地卸载下来。在 Linux 中，模块可以用 C 语言编写，用 gcc 编译成目标文件(不进行链接，作为 *.o 文件存在)。为此，需要在 gcc 命令行加上－c 的参数。成功地向系统注册了设备驱动程序后，就可以用 mknod 命令把设备映射为一个特别文件。其他程序使用这个设备的进程，只要对此特别文件进行操作就行了。

(5)挂上 GUI。在内核的基础上挂上 GUI。大概步骤如下：①分析开发板的配置，选取合适的 Linux GUI；②修改与微处理器有关的代码；③增加设备驱动程序；④GUI 与内核的整合。 (下转第 21 页)

?1994-2016 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

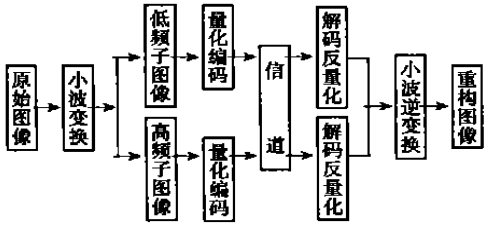

图 5　小波压缩编码原理流程图

(1)对原始图像进行小波变换。选择合适的小波基, 将图像分解为两个部分, 即低频平滑部分(LL)和高频细节部分(LH、HL、HH), 共 4 个子图像。其中, LH、HL、HH 分别对应于图像的水平方向、垂直方向及对角线方向的边缘和细节。这个分解过程可重复地用于分解 LL 部分。

(2)对变换系数进行量化。量化方法主要有标量量化、矢量量化和零树量化等。

标量量化能保持较高的重建图像质量, 但压缩比较低, 在实际标量量化过程中, 结合人眼视觉特征得到的标量量化公式为:

$$\begin{cases} q_1 = q_{max} \\ q_k = \max\left\{1, \mathrm{int}\left(\dfrac{q_k - 1}{4}\right)\right\} \end{cases}$$

式中, q_{max}是最大量化台阶, 实际运用时可取 128、64、32 等。

矢量量化是对矢量集合的量化, 由于变换后子图像不同的分布特性和矩特征, 不适宜采用统一的矢量码书进行矢量化, 应对各子图分别进行矢量化, 能得到较高的压缩比, 但压缩时间较长且重建图像的质量较差; 零树量化则充分利用了小波变换的优秀的时域—频域局部化特性, 及各方向上分解系数间的相关性, 取得了很好的压缩效果。该方法已被应用于许多图像压缩算法之中。

(3)对量化后的系数进行编码。将量化后的系数转化为字符流, 使所得字符流的熵尽可能地小。由图4可以看出, 若原始图像的大小为 N×N, 经过一级小波变换之后, 得到的 4 个子图像的总和仍为 N×N。亦即小波变换之后的数据量与原始图像的数据量是相同的。但变换之后的数据能量更集中, 更有利于图像的压缩。实际的压缩工作是在量化过程中完成的。量化是基于小波变换的图像压缩方法的关键所在。Shapiro 等人提出的嵌入式零树小波算法(EZW), 是目前公认的静态图像变换压缩编码的最好的方法之一。近年来, 小波包及多维小波在图像压缩中的应用也在逐渐增加。

4　结束语

基于小波变换的图像压缩方法, 是一种与图像压缩自身特点相适应的高效的图像压缩方法。利用小波变换的图像多分辨率分解和良好的时频分析特性, 将小波变换用于图像数据的压缩处理, 经过小波分解后图像的大部分能量主要集中在低频信号中, 图像边缘附近的大多数能量主要集中在高频信号中, 这样在视觉允许的情况下, 可以适当选取小波分解的低频信号成分来表示原图像, 实现图像压缩。利用小波变换实现图像数据压缩处理可以获得高的压缩比, 压缩速度快, 图像数据量小, 压缩后能够保持信号与图像特征基本不变, 具有很好的压缩效果。

参考文献:

[1] Mallat S G. Multifrequency channel decompositions of images and wavelet nodels[J]. IEEE Trans. on ASSP, 1989, 37(12): 2091-2110.

[2] 秦前清, 杨宗凯. 实用小波分析[M]. 北京: 科学出版社, 1992.

[3] Kenneth R Castleman. 数字图像处理[M]. 朱志刚, 等译. 北京: 电子工业出版社, 1998.

[4] 李在铭, 等. 数字图像处理、压缩与识别技术[M]. 成都: 电子科技大学出版社, 2000.

(上接第 18 页)

4　结束语

嵌入式 CPU 是嵌入式系统的核心。拥有自己的嵌入式 CPU 和支撑硬件是发展自主产权嵌入式系统的前提条件和基础。操作系统是软件产业的基础和龙头, 它能左右软件产业发展的方向, 是世界软件产业最大的利润来源。研制自主版权的嵌入式操作系统意义重大, 它可以摆脱 PC 时代我们被外国牵着鼻子走的局势, 使中国的 IT 行业真正走向成熟和自立。

参考文献:

[1] 汤子瀛, 等. 操作系统[M]. 西安: 西安电子科技大学出版社, 1996.

[2] 马季兰, 等. 操作系统原理与 Linux 系统[M]. 北京: 人民邮电出版社, 1999.

[3] 陈亮, 等. 跟我学 Linux 操作系统[M]. 北京: 航空工业出版社, 1999.

[4] 赵敏哲. 64 位 Linux 操作系统[M]. 北京: 机械工业出版社, 1999.

[5] 陈亮, 郑敬云. Linux 操作系统[M]. 北京: 航空工业出版社, 1999.

[6] 毛德操, 胡希明. Linux 内核源代码情景分析[M]. 杭州: 浙江大学出版社, 2001.

[7] K Wall, M Watson, M Whitis. Linux Programming Unleashed [M]. Sams, 1999.

[8] John Goerzen. Linux Programming Bible[M]. House of Electronic Industry, 1999.

[9] [美] Maxwell S. Linux 内核源代码分析[M]. 冯锐, 等译. 北京: 机械工业出版社, 2000.

[10] 邹思轶. 嵌入式 Linux 设计与应用[M]. 北京: 清华大学出版社, 2002.

?1994-2016 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net