一种超微内核嵌入式实时操作系统的设计

杨小明[1],陈晓华[1],邹晓[2]

(1.湖州师范学院 信息工程学院,浙江 湖州 313000;2.华东师范大学 教育科学学院,上海 200062)

摘要:本文提出了一种基于对象的超微内核嵌入式实时操作系统的设计方法。采用面向对象的分析和设计技术,结合了微内核和层次式操作系统的特点,设计了一种超微内核的嵌入式实时操作系统,有效地解决了实时操作系统的可伸缩性、实时性、可移植性等问题,并具有信息隐藏、代码可重用等优点,使得嵌入式实时系统的软件开发方便和快捷。

关键词:超微内核;嵌入式;实时操作系统;面向对象

中图分类号:TP316　　**文献标识码**:A　　**文章编号**:1009-3044(2007)01-10171-01

The Design of a Nanokernel Embedded Real-Time Operating System

YANG Xiao-ming[1],CHEN Xiao-hua[1],ZOU Xiao[2]

(1.School of Information & Engineering,Huzhou Teachers,Huzhou 313000,China;2.School of Education Science,East China Normal University,Shanghai 200062,China)

Abstract:In this paper, we present a design method of an object-based nanokernel embedded real-time operating system .We use object-oriented analysis and design technology, and combine microkernel and layered operating system characteristics .The design can efficiently resolve these problems of scalability, real-time behaviors and porting behaviors in the real-time operating system, and has many advantages, such as information hiding, coding reusing.It also makes software development more convenient and rapid in the embedded real-time system.

Key words: nanokernel;embedded;real-time operating system;object-oriented

1 引言

嵌入式实时系统,一般指计算机以片级、板级或箱级埋藏在智能设备中的实时系统。这种系统中,内存空间一般较小,且没有可用的外存,程序放在有限的内存中(一般是固化在ROM或FLASH等)。随着实时应用领域的扩大,实时应用覆盖的范围也不断加大,从低端到高端都有应用的需求,这给现代实时操作系统(RTOS)的研究提出了新的要求与挑战。

传统嵌入式实时操作系统设计采用一体化、高效大内核结构的方法。系统功能集中在内核实现,内核向应用程序提供高效率的系统调用及良好的系统响应时间。传统设计方法设计的嵌入式实时操作系统固然有其好的一面,但与现在嵌入式实时操作系统需求相比仍有较大差距,主要表现在操作系统的可伸缩性、可移植性、对新型结构的硬件平台的支持及接口标准的开放性上。而且大内核结构的系统设计对功能的扩展往往造成内核修改牵一发而动全身。为满足现在实时操作系统的需求,同时兼顾嵌入式应用的特点,本文提出了一种基于对象的超微内核嵌入式实时操作系统的设计方法。我们采用面向对象的分析和设计技术,结合了微内核和层次式操作系统的特点,设计了一种超微内核的嵌入式实时操作系统,有效地解决了实时操作系统的可伸缩性、实时性、可移植性等问题,并具有信息隐藏、代码可重用等优点,使得嵌入式实时系统的软件开发方便和快捷。

2 可伸缩性设计

现代RTOS应能支持多种实时应用的需求,用户希望开发的RTOS可方便配置,满足各种实时应用的需求,无论它们是高端的实时分布式应用(如银行业务),还是低端的强实时、深嵌入应用(如智能武器系统和工业控制系统等)。要求RTOS有良好的可伸缩性、可裁剪性和重用性。

在用面向对象的分析和设计技术来设计超微内核嵌入式实时操作系统,内核的可伸缩性设计以下三个方面来设计:

(1)从操作系统中抽象出超微内核,且内核的大小可以根据用户的需要和具体运行环境来配置的。超微内核抽象层次高、高内聚、代码小,以对象的形式提供给内核其他的部分使用,内核对象高内聚、松耦合,使得内核的扩充变得非常容易,扩充新的类和对象,并不影响超微内核代码本身,从而为系统功能的扩展提供条件。超微内核中的就绪任务表结构设计如图1。超微内核维护了用于系统管理的数据结构,当系统创建新的任务时,调用超微内核的方法动态创建任务控制块并挂接到超微内核中的就绪任务控制块;而且系统中任务的最大优先级和任务的优先级都是由用户指定,从而实现内核的大小可以由用户根据具体情况设定。

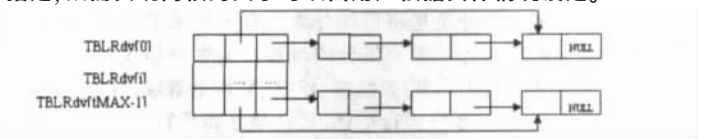

图1　就绪任务表结构

(2)通过在编译时进行重新配置来实现静态扩展系统功能。在编译时进行重新配置,通过省略一些用不到的系统成分,或者在特定接口下对提供不同权衡的实现做出选择,来针对特定工作负载进行定制。

(3)使用微内核的消息传递机制来实现动态扩展系统功能。实时应用的多样性,要求实时操作系统为应用提供多种服务,甚至一些现在没有,但将来可能需要的应用提供扩展支持。现代新型操作系统采用基于消息传送的微内核设计来实现系统的可伸缩性,这一方法同样适合实时嵌入式操作系统设计。微内核设计思想是将操作系统中不常用的系统调用移到核外服务器中实现,并且支持核外服务器的动态配置(加入/退出),微内核与核外服务器之间的联系通过消息传送来实现。微内核可使操作系统的功能扩展靠加入服务器来方便完成。

3 实时性能设计

对于实时系统,一个重要的条件就是延迟有确定的上界,这样的系统属于确定性系统,是从系统论的观点讲的一个功能完整的设计,能够独立和外部世界交互,实现预期功能,包括实时硬件系统设计、实时操作系统设计、实时多任务设计三个部分。

对于实时操作系统来说,应满足的条件有如下几点:可强占式内核、可强占优先级、中断优先级、中断可嵌套、系统服务的优先级由请求该服务的任务的优先级来确定、优先级保护(优先级翻转保护)、中断延迟时间有固定上界、任务切换时间有固定上界、系统响应时间有固定上界等。

(下转第231页)

收稿日期:2006-10-31

作者简介:杨小明(1978-),男,助教,研究方向:嵌入式系统;陈晓华(1977-),男,助教,硕士,主要研究方向:嵌入式系统,数据库技术;邹晓(1981-),女,硕士研究生,研究方向:数据库技术,远程网络教育。

败之地,各行各业的竞争归根到底是人才的竞争。而教育工作则是培养人的工作,只有通过信息化教学手段,采用优资的多媒体教学资源,再加上师生、社会及家庭的共同努力,才能达到较为理想的教学效果,才能实现即定的教学目标,才能培养出较为合格的人才。

参考文献:

[1]祝智庭. 现代教育技术——走进信息化教育[M].高等教育出版社, 2001.09.

[2]张舒予. 视觉文化概论[M].江苏人民出版社, 2003.12.

[3] 齐治昌, 等. 软件工程 (第 1 版)[M]. 高等教育出版社, 1997.07.

[4]张森, 宗绪锋. 多媒体 CAI 课件基本原理与制作技术[J].北京航空航天大学出版社, 2000.06.

[5]罗竞. 现代远程教育的理论和方法[J].电化教育研究, 2000,(1).

(上接第 171 页)

满足上述条件,内核内部具体的实现机制决定了其实时性的优劣。采用基于对象的超微内核操作系统,实时性从以下三点来加强:

第一,不区分核心态和用户态,系统调用是对对象方法的调用。系统调用时不需要在核心态和用户态之间切换,相当于直接的系统调用。设计支持对象的操作系统支持采用对象结构的软件,但是它们内部的结构并不需要基于对象;而基于对象的操作系统使用对象作为它们内部的结构化机制。采用基于对象的操作系统,对象被用来表示类似文件、内存、任务和设备这样的资源。一个类型管理器用来管理各种资源类型,而对象用来表示某种资源类型的一个实例,与资源相关的系统调用实际上就是对对象方法的调用。内核分离成多个独立的模块,每个模块有自己的接口,应用程序按类似的方法来实现,省掉了单一硬件实施保护边界。

第二,为了提高内核效率,采用无保护的单一实地址空间模式,所有任务在同一地址空间运行。优点是任务切换不需要进行虚拟地址空间切换;任务可以直接共享变量,不需要通过内核在不同的地址空间复制数据。在"单地址空间的操作系统"(SASOS),没有必要把内核当作单独的实体来保护,嵌入式计算机系统主要运行静态定义的软件,同样也适合。

第三,提高内核与核外服务器的通信效率,并且对实时性要求高的任务采用直接调用系统接口, 比如设备驱动程序的接口等。典型微内核设计优点主要是为加强操作系统的多模块化设计,即通过简化内核的结构和功能,组成微内核,其中只保留最基本的功能,而将操作系统的大部分服务放在微内核外的各类服务器上实现。如 QNX 公司的 QNX4 RTOS,其微内核只含 17 个最基本的系统调用,处理 IPC、基本调度、底层中断处理等功能,而系统其他功能都在微内核外,以进程管理服务器、文件服务器、设备管理服务器和网络管理服务器等形式实现。典型微内核结构设计对嵌入式强实时系统是不利的,几乎所有系统调用都是一个 RPC 过程,它包含应用到微内核、微内核到核外服务器、服务器工作完成返回微内核以及唤醒请求应用等过程,这主要是过分强调内核功能外移到服务器所至。采用基于对象的超微内核操作系统通过超微内核来优化消息的通信传递来提高超微内核与核外服务器的通信效率。

4 可移植性设计

硬件平台多样化为嵌入式实时操作系统提供了多种选择,而且用户也希望所选购的操作系统能适合在各类微处理器上快速移植。此外,具有相同微处理器的硬件系统也可能会有多种不同的外围电路(如串行通信控制器、定时器等),对于这类不同的硬件平台,操作系统也要求容易移植。对于基于对象的超微内核操作系统,系统的可移植性主要体现在超微内核和驱动管理的可移植性上。

第一,超微内核的可移植性设计:超微内核主要包括任务调度、任务的初始化、中断管理、消息管理和用于系统管理的数据结构。

第二,驱动管理的可移植性设计:对深嵌入、强实时应用,系统硬件多样性不仅表现在微处理器的差异,也可能体现在外围电路(如定时器、串/并行通信控制等)的不同。外围硬件特性容微处理器的特性一样,也不宜在操作系统内固定不变。为此,引入硬件抽象层来支持操作系统在各类 OEM 板或用户定制硬件系统板上的移植。硬件抽象层提供中断调用服务程序。

本文以任务管理调用超微内核为例,讨论可移植性的设计。如图 2,任务管理包括任务的删除、任务挂起与恢复、改变优先级等,这些管理功能操作与硬件有关的部分如任务调度,通过超微内核对象方法的调用来实现的, 与处理器特性有关的部分最终由超微内核调用硬件抽象层来实现。可见任务管理在不同微处理器上的移植只需要修改超微内核和硬件抽象层部分。

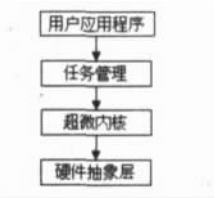

图 2 任务管理示意图

5 应用编程接口的设计

基于对象的超微内核嵌入式实时操作系统中,超微内核并不直接实现实时及线程服务,服务原语通过其调用他对象方法来实现的,而超微内核为这些方法提供支持,如实时任务调度。为满足开放性的需要,服务原语在定义及设计时,尽量向已有的国际标准或工业标准靠拢。在设计时,遵循 POSIX1003 .1c(线程服务)、1003.1/a/b(时钟服务)、1003.1b(定时器服务)、1003.1d draft standard(中断服务)、1003.1b(调度管理)及 1003.1c(解决优先级反转的互斥信号量)。

6 超微内核嵌入式实时操作系统 (MKERTOS)的结构

MKERTOS 设计结合了微内核结构和层次结构的特点, 利用微内核固有的特点加强系统的可伸缩性、可裁剪性、代码的可重用性,采用面向对象的分析和设计技术,独立的超微内核可移植界面以及硬件抽象层来提高 MKERTOS 的可移植性,优化系统内中断响应方式和超微内核中的任务调度算法来提高实时性。MKERTOS 的总体结构框架如图 3 所示。

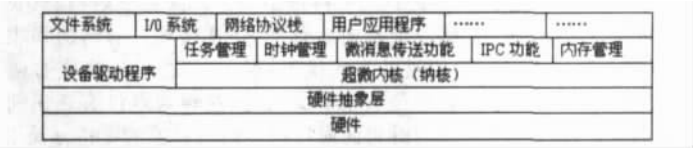

图 3 总体结构框图

7 结束语

嵌入式操作系统随着硬件的发展而有了新的要求与挑战。基于对象的超微内核嵌入式实时操作系统 MKERTOS 是嵌入式实时操作系统的一次新的尝试,目前已完成详细设计,并推出系统原型,在 ADS1.2 下编译,并在基于 ARM7 微处理器 LPC2210 的实验板上测试通过,基本达到前述设计目标。

参考文献:

[1]JEAN- PHILIPPE BABAU.Object Oriented Design for Real-Time Systems—Response to C. E. Pereira' s Contribution[J]. Boston, The International Journal of Time - Critical Computing Systems.18, 95- 99,2000.

[2]Eric Verhulst.The Rationale for Distributed Semantics as a Topology Independent Embedded Systems Design Methodology and its Implementation in the Virtuoso RTOS [J].Boston,Design Automation for Embedded Systems.6,277- 294,2002.

[3]Alan C. Bomberger,A.Peri Frantz,William S.Frantz,Ann C. Hardy,Norman Hardy,Charles R.Landau,Jonathan S.Shapiro .Proceedings of the USENIX Workshop on Micro - Kernels and Other Kernel Architectures [J].USENIX Association,April 1992. pp 95- 112.

[4]Jean J .Labrosse .著.邵贝贝.等.译.嵌入式实时操作系统 uC/OS- II(第二版) [M].北京航空航天大学出版社,2003.5.