

嵌入式系统的动态功耗管理结构

王力生, 康 珊

(同济大学 计算机科学与工程系, 上海 200433)

摘 要:近几年来功耗问题成为在嵌入式系统领域中普遍关注的热点问题。其中动态功耗管理便是一种重要的减少系统范围的能量的方式。在近年的处理机设计技术中引入了支持基于动态电压与频率缩放的功耗管理的系统, 主要提出一种基于这种技术的动态功耗管理的层次体系结构, 它是基于“策略”的一种抽象定义。最后文章还用一实例说明该结构不但能很好地完成复杂系统的功耗管理功能, 而且具有较好的可扩展性, 并且可以大幅度地降低系统功耗。

关键词:嵌入式系统; 动态功耗管理; 层次体系结构; 策略

中图分类号: TP39

文献标识码: A

文章编号: 1673-629X(2008)08-0188-04

A DPM Architecture of Embedded Systems

WANG Li-sheng, KANG Shan

(Computer Science and Engineering Department, Tongji University, Shanghai 200433, China)

Abstract: Nowadays the energy consumption problem has become a main challenge in the embedded systems design. The DPM (dynamic power management) is an important solution of reducing energy consumption in the system level. In recent years, the dynamic voltage scaling techniques have been also used in the processors design. And aims at presenting a hierarchy of DPM for the embedded systems which is based on those techniques. This architecture is an abstract definition on the basis of “policy”. Lastly an example is used to describe that the architecture not only can accomplish the complex system power management functions very well, but also has better scalability, and can significantly reduce the system power consumption.

Key words: embedded systems; dynamic power management; hierarchy; policy

0 引 言

在嵌入式系统的设计中, 低功耗设计是许多设计人员必须面临的问题。嵌入式系统可被看作一些组件的集合, 其中一些组件可能会有机械的部分, 例如硬盘驱动器(HDD), 或者光学的部分, 例如显示器。比如, 手机就有一个超大规模集成电路组件, 一个模拟无线电组件以及一个显示器。在不同的时间这些组件可能会是运作的, 并相应地消耗不同量的手机能源。同样, 笔记本电脑的主要组件有超大规模集成电路芯片, HDD以及显示器, 而且 HDD 与显示器是最经常出现能源饥饿的组件^[1]。然而系统组件并不是总要处于活动状态的, 那么动态调节组件工作状态的能力便是达到节能设计的一种关键手段了。动态功耗管理(DPM)^[2,3]是这样一种方法, 用最少数量的活动组件或尽可能少的能源来提供所需的服务及性能^[4], 在文献[5]中对DPM在节能方面的可行性作了详细的分析与解释。

传统上针对计算机系统的功耗管理主要集中于控制诸如 sleep 与 suspend 这类静态模式的能源消耗。这些都是不活动状态, 经常需要用户再唤醒系统。而且进入和退出这些状态都会有严重的延迟与开销。

DPM 涉及到在程序运行时的功耗管理方案^[6]。许多体系结构提供像 halt 之类的指令来降低空闲期间的 CPU 功耗。操作系统与设备驱动器也可以管理外围设备的功耗, 例如在不活动期间停止转动光盘。对于具有板上外围设备的高度集成的处理机来说, 它经常包含由软件控制的时钟管理来减少不活动外围设备及外围设备控制器的功耗的性能。

在近年的处理机设计技术中引入了支持基于动态电压与频率缩放^[7]的 DPM 的系统。由于 CPU 功耗的减少与电压的立方有关, 而频率又与电压呈线性比例关系, 那么可以调节功耗-性能的平衡来满足应用程序的需要。诸如 Transmeta™, Crusoe™, Intel® StrongARM™, XScale™ 与 IBM® PowerPC™ 405LP 这些处理机允许动态电压与频率缩放并对 DPM 方案给予支持。这些处理机中除了 Transmeta 系统外, 上述的所有处理机都是针对嵌入式系统应用的高集成的片上系

收稿日期: 2007-11-15

作者简介: 王力生(1956-), 男, 副教授, 硕士生导师, 研究方向为嵌入式系统及其应用。

统处理机。它们没有传统的 BIOS,所以对系统的动态功耗状态的控制必须在操作系统中实施。

在接下来的篇幅中将会详细介绍基于上述技术的DPM 体系结构。

1 DPM 体系结构

文中所提到的 DPM 体系结构是一个层次式结构,共分为三层:策略层、操作状态层以及操作点层。其中策略层是最高层,它将每个操作状态映射到一组操作点上。

如图 1 所示,在操作系统内核中具体实现低层,而功耗管理策略来自于系统的外界。注意 DPM 不是一个设备齐全的设备驱动器,需要在操作系统中的一些关键地方增加低层的 DPM 执行。希望熟悉嵌入式系统特征及其特点和需求的系统设计师能为每个应用程序都预先定义一个完整动态功耗管理策略。策略可以用两种方式传达给 DPM:一种是预先定义的一套策略,另一种则是用专门的策略管理器来对策略进行管理。

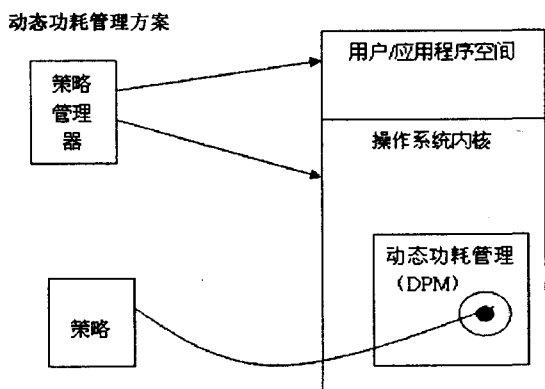


图 1 DPM 方案概况

接下来将按照由低层向高层的顺序对三个层次分别进行介绍。

1.1 操作点

在任何给定的时间点,系统都被称作在一个独特的操作点上工作。操作点可以由核心电压,CPU 和总线频率以及外围设备状态这样的参数来描述。一个动态功耗管理系统可以恰当地定义成一系列规则和程序,这些规则与程序在事件发生时把系统从一个操作点移到另一个操作点。当前那些可以缩放电压及频率的系统都支持多个操作点,而且只要对操作点作适当的选择便会对节能起到关键的影响作用。

操作点是 DPM 系统层次中最低层次的对象。一个操作点对象封装了最小的相互依赖的、物理的并且离散参数集合,它定义了一个明确的系统性能及功耗级别。一个相互依赖数据的例子是 CPU 内核电压

与频率之间的关系,内核电压限制了 CPU 的最大操作频率,而处理机的频率又不能撇开电压而单独地考虑。刚才提到的离散参数是像 SDRAM 中“CAS2”这样的参数,它们对正确性有关键的作用并且受到操作点中其他参数(例如 SDRAM 中的接口频率)的约束。

先进处理机的操作点会依赖于处理机与系统,因此为了应用程序的功耗管理的需要,系统设计师应定义尽可能多的操作点。下面参考 IBM®PowerPC™405LP 给一个操作点的简单例子,在这个设计例子中,包括一个 VGA LCD 板,一个用来安全管理外部的安全芯片。LCD 控制器在 405LP 的主板上,接受一个变速的像素时钟信号,像素时钟频率决定 LCD 的刷新速度以及处理机局部总线 (ProcessorLocalBus, PLB) 的带宽,PLB 把 PowerPC405 内核与存储器控制器以及所有板上外围设备连接了起来。安全新片需要一个外部时钟端口开启一个 33MHz 的时钟。具体见表 1。

表1 一个简单设计实例的操作点

参数	操作点 1	操作点 2	操作点 3	操作点 4	操作点 5	操作点 6
内核电压	1.8V	1.8V	1.8V	1.0V	1.0V	1.0V
CPU 频率	266MHz	266MHz	266MHz	8MHz	33MHz	33MHz
PLB 频率	133MHz	133MHz	133MHz	8MHz	33MHz	33MHz
像素时钟频率	4MHz	22MHz	22MHz	0	17MHz	17MHz
外部时钟频率	0	0	33MHz	0	0	33MHz

1.2 操作状态

若某个系统支持多操作点,根据一些规则和机制,系统会从一个操作点移动到另一个操作点。现有的动态机制会在活动改变或在关键程序请求时移动操作点。事实上,像 IBMPowerPC405LP 这样先进的处理机可以在几微妙的延迟后缩放频率,在几十微妙的延迟后缩放电压,与此同时,这些举动都不会中断系统的操作。图 2 描述了事件发生时操作系统是怎样在不同状态之间进行移动的。

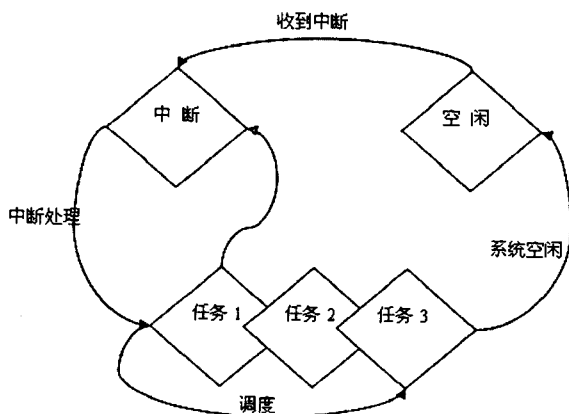


图2 操作状态与状态转移

在 DPM 策略中,每个操作状态会与一个满足其

专门需求的操作点关联在一起。由于在系统空闲时可以通过减少 CPU 和总线频率以及内核电压的方法来达到系统范围的节能目的,那么一个机制应该会区分程序执行期间不同的操作点,这便自然而然地引出了“活动”与“空闲”这两个状态,潜在地区分了不同的操作点。通过操作系统中的 DPM,便可以高效地在“活动”状态中的操作点与“空闲”状态中的操作点之间进行转移。在文献[6]中对操作点的这种形式的细粒度控制的可行性进行了探索。

操作状态的概念也为功率监控的任务提供了特殊服务操作点。DPM 体系结构顾及了任何数量的任务状态。默认的系统状态被认为是多数任务使用的状态,如同大多数任务所使用的操作系统的默认调度策略一样。有特殊需求的任务会设有专门的任务操作状态,并且关联一个不同的操作点。注意任务从不明确地指定一个操作点,它是被任务状态以及当前策略所暗示的。

操作状态也是 DPM 体系结构的一层,DPM 策略简单地就把操作点与系统的每个操作状态关联了起来,若要转到一个新 DPM 策略的话只要简单改变关联就可以了。

1.3 策略与策略管理器

策略被命名为一些数据结构,并且被定义为对系统动态状态所实施的细粒度控制。因此为了高效,策略必须安装在系统内核中。策略详细说明了组件与设备状态的变迁,以确保符合功耗管理方案的可靠操作。

在 DPM 体系结构中策略是最高层的抽象,它将每个操作状态映射到一个操作点同余类(每个同余类是一个操作点的集合,类中的任何操作点在已给状态下都是系统可接受的,考虑到功耗,DPM 策略会从中选中一个最合适的操作点)。一个功耗管理方案至少会指定一个策略,并且为了适应不同的场合,会指定尽可能多的策略。在任何给定时间点,生效的策略会控制在每个操作状态下的操作点。完整的 DPM 系统层次如图 3 所示。

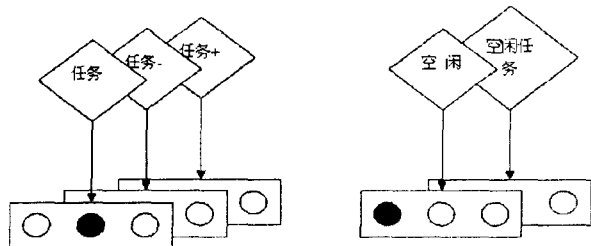


图 3 DPM 系统层次图

在上图中,为每个操作状态都配备了一个操作点同余类,其中灰色的操作点是系统在那个状态时被选

中的操作点。

值得注意的是,除了所有平台共有的单一“任务”状态之外,DPM 体系结构并没有硬性规定存在任何其他操作状态。任务状态的数量会随着平台的不同而不同。然而在所有平台上,都会通过策略以及策略管理器给予任务状态实际意义。

若需要多种策略,那么系统中必须存在策略管理器来调整激活不同的策略。DPM 策略管理器是通过名字激活策略的可执行程序,它执行用户定义的及专用的功耗管理方案。策略管理器可以从操作系统、用户偏好设置、运行程序、配置文件及(或)物理设备中收集信息来作其策略决定。它的“位置”(内核空间或用户空间),需要信息的类型,以及作为回应所采取的所有行动都没有被指定。这个体系结构的意图仅仅是定义一种相容的方式,它可以帮助策略开发者描述被策略管理器控制的以及由 DPM 执行的策略。

DPM 的中心功能是在设备状态改变时自动选择操作点。接下来用一个简单的例子来描述 DPM 体系结构关于动态改变来自外围设备请求的操作。这里还是准备用表 1 的那个设计实例,各个组件的具体情况在此便不再累述。

如图 4 所示,这个例子中有两个状态,“任务”与“空闲”,并且每个状态都给了 3 个操作点。每个操作点中的元素集合由内核电压,CPU 频率,PLB 频率,像素时钟频率以及外部时钟频率组成,正如表 1 中所示。

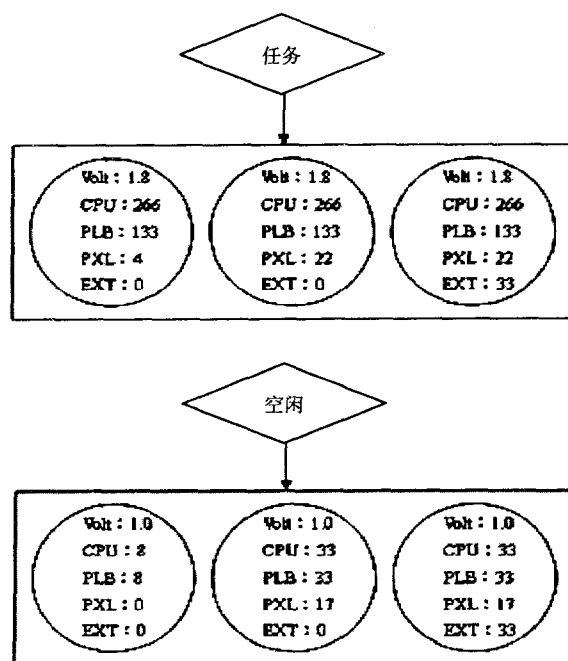


图 4 一个简单的策略实例

这是一个高性能的策略,它为“任务”状态指定了 1.8V,266MHz 的操作点,而为“空闲”状态给了低压和

低频的操作点。两种状态下,最左边的操作点是集合中最低耗的,其中像素时钟是几乎不工作的(因为 VGA 显示器的显示功能需要的像素时钟频率最小为 17MHz),并且外部时钟是完全不工作的。其余两个操作点的区别仅仅在于是否开启了外部时钟而已。

现假设 LCD 控制器工作而安全芯片脱机的情形。在这种状态下,每种状态的第一个操作点都是无效的,因为没有足够的像素时钟频率。在 LCD 控制器改变状态的时候,LCD 控制器的像素时钟请求会传达给系统,因此当 LCD 控制器被激活时,DPM 系统会使指示的操作点无效。每个操作点同余类中的下一个操作点是合法的,所以这是一个合法的策略。这里假设 DPM 的具体实现使用了预定义的排序,它对类中的操作点按照优先权排了顺序(图中从左至右优先权依次递减),用这种方法来选择合法的操作点。

再设想另外一种安全芯片联机的情形。在这种情况下,每个状态下只有一个操作点是合法的,即那个以 33MHz 开启外部时钟端口的那个操作点。一旦安全芯片设备驱动器决定该芯片可以脱机时,这个决定会传达给功耗管理系统然后策略将会把两个状态都关联到图 4 中各自的最右边的操作点。

在用户仅仅把例中这个设备用作音频播放器的时候,LCD 控制器与安全芯片可能都不工作。在这种情况下,会选中每个操作点同余类中的最低耗操作点。由于 LCD 控制器不工作,那么对 PLB 和存储器带宽的需求量也很有限,所以“空闲”状态操作点指定了最低耗的状态,这个状态的功耗仅仅只够让系统退出“空闲”状态并回到“任务”状态而已。

(上接第 187 页)

力强。尤其在网络规模较大,备选路径较多时,本算法搜索速度快,减少了路由计算时间,解决了实时通信中动态拓扑结构下的多约束 QoS 组播路由问题,算法有效性强。同时,通过并行进化模型,解决了主从并行遗传算法中主、从节点负荷不均匀,运行效率不高;细粒度并行遗传算法中需要大量处理器;粗粒度并行遗传算法中通信开销大、算法收敛速度慢等问题。

参考文献:

- [1] 王征应,石冰心.基于启发式遗传算法的 QoS 组播路由问题求解[J].计算机学报,2001,24(1):55-61.
- [2] 曹元大,蔡 烈.组播路由 QoS 路由的遗传算法研究[J].计算机工程,2004,30(7):80-82.
- [3] Han K-H. Genetic Quantum Algorithm and its Application to Combinatorial Optimization Problems[C]//In: IEEE Proc. of

2 结束语

介绍了一个针对嵌入式系统的支持动态功耗管理的体系结构。这个结构是以当代处理机以及其应用需求为基础的,它是基于“策略”的一种抽象定义,并定义为“操作点”到系统中的“操作状态”的一种映射。最后用了一个简单设计实例说明 DPM 体系结构关于动态改变来自外围设备请求的操作。可以看出该结构不但能很好地完成复杂系统的功耗管理功能,而且具有较好的可扩展性,并且可以大幅度地降低系统功耗。

参考文献:

- [1] Lorch J, Smith A. Software strategies for portable computer energy management[J]. IEEE Personal Commun, 1998, 5: 60-73.
- [2] 陈静华,陈迪平,徐勇军,等.面向芯核设计的功耗层次化管理策略[J].计算机辅助设计与图形学学报,2005,17(5):1079-1084.
- [3] 车 健.嵌入式系统中低功耗设计[J].电子测量技术,2005(3):16-18.
- [4] Benini L, De Micheli G. Dynamic Power Management: Design Techniques and CAD Tools[M]. Norwell, MA: Kluwer, 1998.
- [5] Benini L, Bogliolo A, De Micheli G. A survey of design techniques for system-level dynamic power management[J]. IEEE Transactions on VLSI Systems, 2000, 8(3): 299-315.
- [6] Shaffer S, Beyda W J, Gold C. Apparatus and Method for Automatic CPU Speed Control Based on Application-Specific Criteria[P]. United States Patent: 6,298,448, 2001-10.
- [7] 卢春鹏.动态电压与频率调节在降低功耗中的作用[J].单片机与嵌入式系统应用,2007(5):12-17.

the 2000 Congress on Evolutionary Computation. San Diego, USA: IEEE Press, 2000: 1354-1360.

- [4] Narayanan A, Moore M. Quantum Inspired Genetic Algorithm [C]//In: Proc. of the 1996 IEEE Intl. Conf. on Evolutionary Computation (ICEC96). Nogyaya, Japan: IEEE press, 1996: 41-46.
- [5] Yang Junan, Zhuang Zhenquan. Research of Quantum Genetic Algorithm and Its Application in Blind Source Separation[J]. Journal of Electronics(China), 2003, 20(1): 62-68.
- [6] 杨俊安,庄镇泉.量子遗传算法研究现状[J].计算机科学,2003,30(11):13-15.
- [7] 陈国良,王熙法.遗传算法及其应用[M].北京:人民邮电出版社,1996.
- [8] Bernard M, Waxman. Routing of Multipoint Connections[J]. IEEE Journal on Selected Areas in Communications, 1988, 6(9): 1617-1622.