嵌入式操作系统的时钟机制的研究*

田小华　陆少华
（武汉理工大学计算机应用技术系　武汉　430037）

摘　要　在实时操作系统中，无论是实时响应并处理来自各个被控对象的实时信息，还是对任务执行时间的管理、资源的限时等待等，都需要时钟的参与。因此，准确而又具有足够精度的时钟对于实时系统是必不可少的。根据普通嵌入式系统内核中的时钟机制的特点，分析了实时操作系统中的有关时钟机制改进的具体方法与特点，并提出了三种多粒度时钟机制：UTIME 机制、RFRTOS TIMER 机制、RTOS KERNEL TIMER 机制，并且分析了与时钟机制密切相关的中断服务程序。

关键词　嵌入式系统　RTOS　时钟管理　多粒度时钟机制
中图分类号　TP316.1

Research on Timer Machanism of Eebedded Operating System

Tian Xiaohua　Lu Shaohua

（Department of Computer Science and Technology, Wuhan University of Technology, Wuhan　430070）

Abstract　In real-time operating system, the real-timely responding to and processing every real-time massage from the controlled objects or managing the task's excuting time as well as the limited waiting of resource, timer is necessarily to join. Therefore, correct and very accurate timer is requisite. According to the features of timer mechanism of common embedded system kernel, this essay analyzes concrete manners and features concerning reforming the timer mechanism. It proposes three multi-granular timer machanism: UTIME machanism, RFRTOS TIMER machanism, RTOS KERNEL TIMER machanism. Furthermore, it analyzes the interruption service program largely related to timer machanism.

Key words　embedded system, RTOS, timer managing, multi-granular timer mechanism
Class Number　TP316.1

1　引言

目前 Linux 内核在实时性方面有所增强，但它仍然不是实时操作系统，Linux 内核的设计注重应用程序的吞吐量连同内核整体设计的完美性。作为提高吞吐量的必然要求，Linux 的调度器试图提供一种“公平分配”策略来保证所有的进程可以均衡地享有 CPU 的资源。为了提高普通 Linux 系统的实时性，必须对其进行实时性改进。

而对时钟机制的改造，可以提高系统时钟的精度，从而增强了系统的实时性，特别是对外部中断的响应。因为精确的时钟是操作系统进行准确的调度工作的必不可少的条件。

2　时钟粒度

在每个操作系统内部，都有一个计数器，用来作为系统的时钟。每当有时钟中断的到来，该计数器的值就加一。时钟中断到来的频率，即每个时钟中断的时间间隔，就是操作系统时钟的最小计时单位，我们称它为时钟粒度。

* 收稿日期：2008 年 9 月 6 日，修回日期：2008 年 10 月 18 日
作者简介：田小华，副教授，研究方向：嵌入式操作系统、计算机体系结构。陆少华，硕士研究生，研究方向：嵌入式系统，体系结构。

时钟粒度的大小, 直接决定操作系统时间的精度以及对外界的响应速度。因为每个时钟中断可以看做是一个抢占剥夺点。当有时钟中断到来时, 如果有高优先级的任务在等待运行, 则当前运行任务被抢占。但在实时系统中, 大粒度的时钟根本不能满足实时应用的需要, 而简单地提高时钟频率, 又会增加时钟中断的次数, 造成不必要的开销。

在 RTOS Kernel 中, 通过采用可配置的多粒度时钟, 很好地解决了这个矛盾。在系统中, 可以设置了一个定义时钟周期的无符号整型变量: TIMER_PERIOD。在系统时钟初始化阶段, 根据配置文件中用户设定的时钟周期来设置此变量, 从而决定系统时钟中断的频率。这样, 用户可以根据实际应用的需要自行配置系统的时钟粒度, 在需要高精度时可以将中断频率设高点, 在追求更高的系统效率时可以将中断频率设低点, 既满足实时需求, 又减少不必要的系统开销。

但是, 仅仅依靠可配置的时钟粒度是不够的。因为某些实时系统对时钟精度要求非常高, 常常是微秒级、甚至是纳秒级。而如果系统的时钟粒度根据需要也配置到如此高精度的话, CPU 的绝大部分时间都可能在处理时钟中断, 从而严重影响系统正常工作。

3 时钟机制的改进

3.1 UTIME 机制的提出

为此, 有人提出了 UTIME 机制, 将原本周期性的时钟中断改为非周期性的, 在每次时钟中断处理程序中, 计算好下次产生中断的具体时间, 然后重新设定定时器中断值, 从而达到产生微秒级中断的效果。但是这种方法需要在时钟中断中增加额外的计算, 以及根据计算结果重新写芯片, 对系统的性能产生了一定影响。

通过试验比较, 在没有 UTIME 的 Linux 中, 多数时钟中断处理的时间约为 1μs, 最多不过 5μs。但加入了 UTIME 后, 多数时钟中断处理时间约为 15μs, 如果按 10ms 产生一个中断的话, 系统性能下降了 0.15%。

3.2 RFRTOS TIMER 机制

在 RFRTOS 中则采用了另外一种方法来提高时钟精度。与 UTIME 机制不同, 它提供了一个与核心时钟并行的具有精密时钟刻度的实时时钟, 高精度的时钟粒度由实时时钟来提供, 核心时钟的周期一般不用修改。实时时钟的设计不但提高了系统的稳定性, 而且独立的实时时钟更加容易维护改进。但是, 实时时钟还是与核心时钟共用同一个时钟中断服务程序, 只是服务程序中添加了对实时时钟中断的处理。通过实验比较, 加入了实时时钟的时间时钟中断处理大都控制在 9μs 到 10μs 之间, 如果按 10ms 产生一个中断的话, 这种方案的系统整体性能下降约 0.1%。

3.3 RTOS KERNEL TIMER 机制

在 RTOS Kernel 中, 也采用了两个时钟, 一个普通时钟, 一个高精度时钟。但是与 RFRTOS 不同, 对于不同的时钟有不同的中断处理程序。普通时钟用于正常的周期性计时, 而高精度时钟可以根据实际需要进行配置。如果任务是有周期性的中断要求, 则可以将它设为周期性高精度时钟; 否则设为 one-shot 模式, 在每次需要的时刻才进行设定。当系统普通时钟能够满足任务对于时钟精度的需求时, 还可以完全关闭高精度时钟。因为高精度时钟的中断处理程序是独立的, 所以在关闭高精度时钟的情况下, 系统就和只有一个时钟的系统完全一样, 不会有任何额外的开销。

4 三种机制的比较

三种不同的时钟机制如图 1 所示。

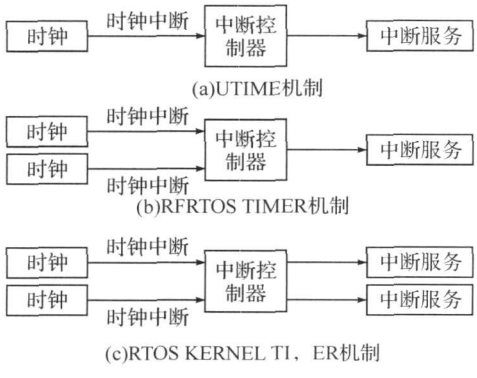

图 1 三种时钟机制比较

RTOS Kernel 机制具有如下优点:

1)实现简单: 它从最底层将两种时钟完全区分, 普通时钟和高精度时钟完全独立, 互不影响, 实现起来相对简单;

2)由于一般在高精度时钟中断中只是做非常简洁的处理, 如定时器管理等, 并不涉及系统计时等操作, 所以它的执行速度相当快, 对系统性能影响小;

3)可以产生周期性的紧急时钟中断, 避免了 UTIME 等类似机制的计算中断时间以及对定时器重新编程的开销。

(下转第 114 页)

参 考 文 献

[1] 郭宏瑞, 谭吉春. 印刷品防伪新技术研究[D]. 长沙:国防科学技术大学研究生院, 2006, 11

[2] 廖丹, 谭吉春. 印刷二维条码防伪新技术[D]. 长沙:国防科学技术大学研究生院, 2005, 11

[3] 二维条码加密技术在检验报告防伪系统中的应用研究[D]. 长沙: 中南大学, 2006, 11

[4] 刘磊, 袁国桃等. 基于二维条码的身份证防伪认证系统[J]. 西安理工大学学报

[5] 付利莉. DES 算法在二维条码数据加密中的应用[J]. 石油化工高等学校学报, 2005, 18(2): 2~4

[6] 刘晓星, 胡畅霞, 刘明生. 安全加密算法 DES 的分析与改进[J]. 微计算机与信息, 2006, 22(4): 32~33

[7] 柴井坤. RSA 加密算法的实现及其细节问题的研究[J]. 黑龙江科技信息, 2007, 12: 75

[8] 吴宇新, 余松煜. 对 LZW 算法的改进及其在图像无损压缩中的应用[J]. 上海交通大学学报, 1998, 32(9): 10~11

(上接第 70 页)

5 时钟中断服务程序的实现

每个时钟中断到来,时钟中断服务程序都会被执行。它主要完成系统计时、任务已占用 CPU 时间计时以及定时器队列的管理等工作。因为时钟中断服务程序执行的频率相当高,每个系统周期都会被执行,这就要求程序的处理必须非常高效、简洁。分别对应于普通时钟和高精度时钟,中断处理程序分为普通时钟中断处理和高精度时钟中断处理。

在普通时钟中断处理程序中,首先完成系统计时,将当前时间加上一个时间片。然后对当前任务进行计费—统计占用 CPU 的时间,最后对各种定时器进行处理。在定时器的管理上,采用一个双向链表管理所有的定时器,将各个定时器按触发的系统绝对时间由近到远进行排序。在每次时钟中断中,将定时器的时间与当前时间进行比较,如果定时器时间小于或等于当前时间,则说明定时器已经到期,将其从定时器队列移出,并执行该定时器的回调函数。依次处理,直到遇到第一个没有到期的定时器为止。定时器事件队列的数据结构及主要代码如下:

```
typedef struct timer_event_block{
  QUEUE queue; /*双向队列*/
  LSYSTIM time; /*事件时间*/
  CBACK callback; /*回调函数*/
  VP arg; /*回调函数参数*/
}TMEB;
//定时器管理
while(!isQueEmpty(&timer_queue)){ /*判定时器队列是否为空*/
  event=(TMEB*)timer_queue.next; /*取出下一个定时器*/
  if(ll_cmp(event->time, current_time)>0)break; /*定时器是否已经到期*/
  QueRemove(&event->queue); /*到期定时器出队*/
  if(event->callback!=NULL){ /*执行回调函数*/
  (*event->callback)(event->arg);
  }
}
```

高精度时钟及其中断服务程序只是作为普通时钟的一种补充,只有在需要的时候才被激活。而且因为其中断产生的频率可能非常高,所以在高精度时钟的中断处理程序中除了最简单的定时器管理之外,不作任何其他处理,尽可能地减小执行开销。因此,它的中断处理代价是非常小的。在中断处理程序中对于定时器的管理与普通时钟中断处理中的管理完全相同。

6 结语

本文提出了三种多粒度的时钟机制,其中 RTOS KERNEL TIMER 机制的特点为独立的、可配置的。在用户自主配置时钟粒度的基础之上,采用双精度时钟机制,完全地区分普通时钟中断和高精度时钟中断,既满足实时需求,又减少不必要的系统开销,较好地解决实时性与系统效率之间的矛盾。

参 考 文 献

[1] 苏中义, 杨宇. 嵌入式系统[J]. 上海电机技术高等专科学校学报, 2004, (3)

[2] 罗炜. 嵌入式实时操作系统关键技术的研究[D]. 湘潭大学, 2006

[3] 沈胜庆. 嵌入式操作系统的内核研究[J]. 微计算机信息, 2006, (5)

[4] 高晓超. 嵌入式操作系统实时机制研究[D]. 南京航空航天大学, 2003

[5] 孔军. 嵌入式操作系统实时性研究与改进[J]. 黑龙江科技信息, 2008, (4)