

分布式实时操作系统消息机制的设计与实现

舒红霞, 王继红

(武汉数字工程研究所, 湖北 武汉 430074)

摘要: 随着数字信号处理技术的迅猛发展, 针对并行数字信号处理(DSP)应用自主开发了一个满足用户需要的高性能分布式实时操作系统——腾飞分布式实时操作系统(TF-RTOS)。消息机制用于线程间的通信, 是操作系统中的重要部分。在开发 TF-RTOS 过程中, 从消息命令包、消息队列、消息传递过程和消息原语这 4 个方面设计并实现了一种直接消息传递的消息机制, 该消息机制具有简化线程间通信、增强系统功能、提高系统性能的特点。

关键词: 分布式; 实时操作系统; 数字信号处理; 并行; 消息机制

中图分类号: TP316.4 **文献标识码:** A **文章编号:** 1000-7024 (2008) 09-2239-03

Design and implementation of message passing for distributed real-time operating system

SHU Hong-xia, WANG Ji-hong

(Wuhan Digital Engineering Institute, Wuhan 430074, China)

Abstract: With the development of digital signal processing, a distributed real-time operating system named TengFei distributed real-time operating system is independently developed based on parallel DSP systems. Message passing is a mechanism, which is applied in communication between threads. It is an important part in real-time operating system. According to message package, message queue, message passing procedure, and message passing API, the design and implementation of a message passing mechanism in TF-RTOS are described. The message mechanism has facilitated communication between threads and improved system performance.

Key words: distributed; RTOS; DSP; parallel; message mechanism

0 引言

随着数字信号处理技术的迅猛发展, 各种高性能的数字信号处理(DSP)芯片应运而生, 在科学研究、军事及民用电子技术领域发挥越来越重要的作用, 其应用广度和深度正不断地扩展和深化^[1-3]。为了适应 DSP 应用的复杂性和多样性, 缩短其开发周期, 自主开发了腾飞分布式实时操作系统, 简称 TF-RTOS。消息机制是操作系统中的重要部分^[4-5], 在 TF-RTOS 中, 设计并实现了一种简单可行的直接消息传递的消息机制。

1 腾飞分布式实时操作系统 TF-RTOS

TF-RTOS 是针对并行 DSP 应用开发的操作系统, 它采用了虚拟单处理器模式, 支持多线程, 调度方式是可抢占式调度。如图 1 所示, TF-RTOS 操作系统由 4 层结构组成。最底层是硬件层, 即数字信号处理器 DSP 层。在硬件层上面是内核

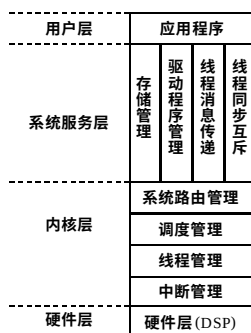


图 1 TF-RTOS 结构

层, 包括系统必不可少的与硬件相关的管理功能, 如中断管理、线程管理、调度管理和系统路由管理。在内核层上面是系统服务层, 包括线程同步与互斥、线程消息传递、驱动程序管理和存储管理, 这一层独立于硬件系统, 与硬件相关的操作全部通过对内核的调用实现。最上层是用户层, 即应用层, 包括各种应用程序。

2 消息机制

目前, 众多的实时操作系统中采用的消息机制有两种^[4-7]: 一种是传统实时操作系统采用的消息传递机制, 简称传统消息传递机制; 另一种是直接消息传递机制。

传统消息传递机制借助于邮箱、管道等通信介质进行消息传递, 并且, 传统消息传递机制有过多的消息拷贝开销, 大大影响了实时系统性能。直接消息传递机制允许一个线程直接发送消息到另一个线程, 无需邮箱或管道等的介入, 并且, 直接消息传递机制通常不进行消息的拷贝, 而直接将消息指针从发送线程传送到接收线程, 即使在具有多个处理器的分布式系统上, 也只在转发消息时才进行一次消息拷贝。因此, 直接消息传递机制具有很高的通信性能。

直接消息传递机制较传统消息传递机制有许多优点, 基

收稿日期: 2007-05-05 E-mail: shirleyshu@ustc.edu

作者简介: 舒红霞 (1980—), 女, 湖北孝感人, 硕士, 研究方向为分布式实时操作系统、并行算法、并行体系结构; 王继红 (1960—), 男, 研究员, 研究方向为分布式实时操作系统、并行处理系统软件开发环境。

于这种消息机制,在开发TF-RTOS过程中,提出了一种简单可行的消息传递机制,简称TF-IPC。TF-IPC允许一个线程直接向另一个线程发送消息,并提供了异步通信和同步通信两种通信方式。在分布式系统中使用TF-IPC,应用程序只需知道消息是从哪个线程传递到哪个线程即可,而不必了解分布式系统的结构,因此,TF-IPC使线程间的通信变得非常简单。此外,TF-IPC还能增强系统功能,提高系统性能。

3 TF-IPC的设计与实现

本节将从消息命令包、消息队列、消息传递过程和消息原语这4个方面来介绍TF-IPC。

3.1 消息命令包

TF-IPC采用了直接消息传递方式,线程也不必立即处理接收的消息,为了有效地管理消息,设置了一种称为消息命令包的数据结构,该结构包括消息管理区域和一个指针。其中,指针指向实际消息内容,消息管理区域包括发送线程标识符(ID),接收线程ID,消息类型,消息长度等。在发送或者接收消息前,TF-IPC会自动为每个消息设置相应的消息命令包结构,并在传递过程中更新消息命令包结构。事实上,TF-IPC将线程之间传递的消息内容封装到了消息命令包中,线程之间传递的是消息命令包。

3.2 消息队列

传统消息传递机制中使用的消息队列是一个特定的对象,是操作系统中一个独立的部件,一个线程向消息队列发送消息,然后另一个线程检查并从消息队列中接收消息。在使用传统消息传递机制的操作系统中,消息队列与线程数目不一定相等,往往消息队列数目大于线程数目。

TF-IPC中使用的消息队列与传统消息传递机制中使用的消息队列对象不是同一个概念。TF-IPC中使用的消息队列不是一个独立的对象,而是线程控制块结构体的一部分,它用来存放需要发送的消息或者是已经接收但还没来得及处理的消息。显然,这种消息队列与线程的数目是相等的,在创建线程的时候产生。

TF-IPC中,消息队列由消息命令包构成。该消息队列结构包括接收消息命令包链表、请求消息命令包链表、等待发送消息命令包链表、等待线程链表和已经发送消息计数器。当线程来不及处理接收的消息时,直接将消息挂到接收消息命令包链表上;当线程没有等到需要接收的消息时,将该请求消息挂到请求消息命令包链表上;当线程发送的消息个数已经超过了最大允许发送个数时,将要发送的消息挂到等待发送消息命令包链表上,若该最大允许发送个数置1,线程之间实现同步通信,否则线程之间实现异步通信;当线程等待的消息没有到来,或者不能再次发送消息时,将线程自身挂到等待线程链表上,即线程由运行态转化为挂起态。

3.3 消息传递过程

TF-RTOS是建立在多个DSP上的实时操作系统,那么通信线程可能分布在不同的DSP上,将位于同一个DSP上的线程之间的消息传递称为本地消息传递;将位于不同DSP上的线程之间的消息传递称为异地消息传递。

3.3.1 本地消息传递

假定某DSP上有线程A和线程B。当线程A和线程B进行本地消息传递时,线程A向线程B发送消息,线程B从线程A接收消息。

对于发送线程A,TF-RTOS内核只需要为线程A即将发送的消息设置一个消息命令包,该命令包中包括线程A的ID,线程B的ID,消息类型,消息长度,以及用来链接实际消息内容的指针。TF-RTOS内核根据该命令包中的接收线程标识符(即线程B的标识符)找到线程B的消息队列,并将该命令包插入到线程B消息队列的接收消息命令包链表,等待线程B空闲时处理。

对于接收线程B,可以不必立即处理已接收的来自线程A的消息,等到空闲时再对该消息进行处理。

对于本地消息传递,TF-IPC不进行消息拷贝,直接将消息指针从发送线程传送到接收线程。在整个消息传递过程中,发送消息一直位于开始的内存位置,没有做任何移动或拷贝。TF-IPC避免了多次拷贝,具有很高的通信性能。

3.3.2 异地消息传递

假定线程A和线程B在不同的DSP上。当线程A和线程B进行异地消息传递时,线程A向线程B发送消息,线程B从线程A接收消息。

当线程A向线程B发送消息时,与本地消息传递类似,TF-RTOS内核为线程A设置消息命令包。TF-RTOS内核检测到该命令包中的接收线程B与发送线程A不在同一个DSP上,TF-RTOS内核通过系统路由管理调用相应的系统线程,将该消息命令包及其指针所指向的实际消息内容,通过DSP硬件系统发送到接收线程B所在的DSP上。

在接收线程B所在的DSP上,首先,TF-RTOS内核通过系统路由管理调用相应的系统线程接收该消息命令包;其次,根据该消息命令包中接收线程标识符(即线程B的标识符)找到线程B的消息队列;最后,将该命令包插入到线程B的消息队列的接收消息命令包链表,等待线程B空闲时来处理该消息。

3.4 消息原语

TF-RTOS提供了4种消息原语:①发送消息原语MsgSend,向由用户自行指定的线程发送消息;②无等待接收消息原语MsgReceive,若没有接收到需要的消息,则直接返回;③无限等待接收消息原语MsgReceiveW,若没有接收到需要的消息,那么将线程挂起,直到等到该消息到来;④超时等待接收消息原语MsgReceiveWT,增加一个超时控制时间,若在该时间内没有接收到需要的消息,则直接返回。

目前,为了保证消息传递的可靠性,实现线程之间的同步或异步通信,设置了应答消息,即:对于接收线程,每接收一个消息,就必须回发给发送线程一个应答消息,表示已经接收到消息;对于发送线程,每接收一个应答消息,就将已经发送消息计数器减1。由于3种接收原语类似,下面仅介绍发送消息原语和无限等待接收消息原语。

3.4.1 发送消息原语

发送消息原语型: void MsgSend(unsigned ThreadId, void* MsgBody, unsigned MsgType, unsigned MsgLength)。第1个参数表示接收线程标识符,第2个参数表示发送的消息内容,第3个参数表示发送消息的类型,第4个参数表示发送消息的长度。

如图 2 所示,发送消息原语完成的功能:首先,为需要发送的消息分配并设置消息命令包;然后,检查该线程已经发送消息计数器是否到达最大允许发送个数,若到达,则将该消息命令包和发送线程挂起到自身的消息队列上,若没有到达,则将该计数器加 1,并判断发送线程和接收线程是否位于同一个 DSP,若在一个 DSP 上,则进行本地发送消息操作,否则,进行异地发送消息操作。

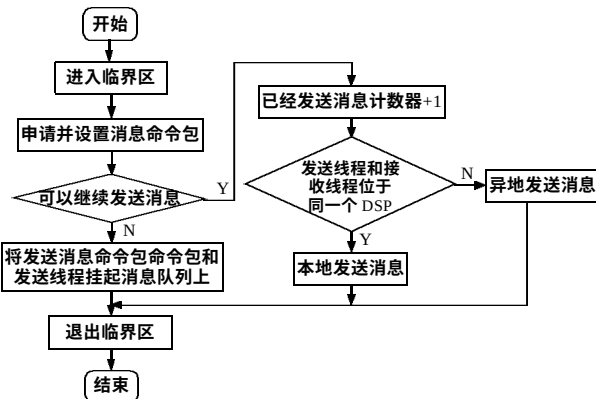


图 2 发送消息原语

本地发送消息操作完成的主要功能是:根据消息命令包中接收线程标识符找到接收线程的消息队列,若接收线程此时正忙,没有时间处理该消息,则直接将该消息命令包挂到其消息队列上;如果接收线程此时正在等待该消息,那么就on接将消息命令包中的实际消息内容交由接收线程处理。

异地发送消息操作完成的主要功能是:根据消息命令包中接收线程所在的 DSP 号,TF-RTOS 系统路由管理将消息命令包转发到相应的系统线程,然后由该系统线程将消息命令包及其指向的实际消息内容通过 DSP 硬件系统转发给相应的接收线程所在的 DSP。

3.4.2 无限等待接收消息原语

无限等待接收消息原语原型 void MsgReceiveW(unsigned ThreadId, void *MsgReceive, unsigned MsgType, unsigned length)。第 1 个参数表示发送线程标识符,第 2 个参数表示存放接收消息内容的变量的地址,第 3 个参数表示接收消息的类型,第 4 个参数表示接收消息的长度。

如图 3 所示,无限等待接收消息原语 MsgReceiveW 完成的功能:首先,判断接收线程消息队列上接收消息命令包链表是否为空,如果是,则为接收线程分配并设置一个消息命令包,并将该消息命令包和接收线程同时挂起;如果不是,则将接收线程需要接收的消息与接收消息命令包链表上的消息进行匹配,匹配成功,则处理匹配成功的消息,并向发送线程回发应答消息,否则与接收消息命令包链表为空时的操作类似。

4 实例

ADSP-TS201 数字信号处理器^[8-9]是美国 ADI 公司的 TigerSHARC 系列 DSP 中一款高性能的静态超标量数字信号处理器,它整合了 I/O 接口、外部 SDRAM 控制器、24 Mbits 的片内 DRAM、主处理器接口、以及适合多处理器的无缝共享总线连

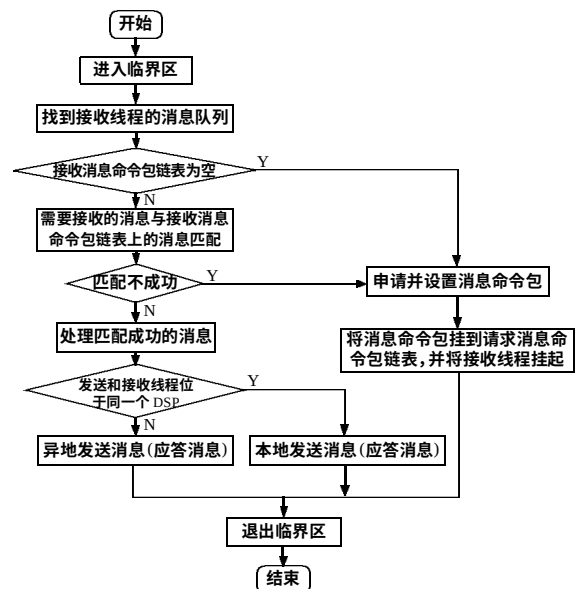


图 3 无限等待接收消息原语

接。它的数据传输系统包括 4 个链路通道对和由 DMA 控制器支持的 14 个 DMA 通道,可完成片内与片外存储器、存储器映射外设、链路口、主处理器和其它 DSP(多 DSP 系统)间的数据传输。它的内部核时钟高达 600 MHz,指令周期为 1.67ns, TigerSHARC 系列的静态超标量结构使 DSP 每周期能够执行多达 4 条指令,每秒可进行 48 亿次 16bit 定点全累加操作,或者每秒可进行 12 亿次 32 bit 浮点全累加操作。

ADSP-TS201 数字信号处理器具有很强的数据处理能力和很高的运算速度,在具有两个 ADSP-TS201S 处理器的 ADSP-TS201S EZ-KIT Lite Rev1.1 评估板上,基于 TF-RTOS 操作系统,设计了消息传递实验。

编写如下应用程序:

```

#define MSGSIZE 64 //每次传递的消息大小
#define MSGTYPE 1//传递整型消息

void task1 {
    int *send;//消息内容
    while(1) {
        send=TFMalloc(MSGSIZE);//调用 TF-RTOS 原语分配一块内存存放消息
        初始化消息内容;
        MsgSend(TASK2,send,MSGTYPE,MSGSIZE);//向线程 TASK2 发送消息
    }
}

void task2 {
    int *receive;//存放接收消息内容
    int RecL;//接收消息长度
    while(1) {
        MsgReceiveW (TASK1,&receive, MSGTYPE,&RecL);//接收 TASK1 发来的消息
        TFFree(receive, MSGSIZE);//不再使用 receive 指向内容,释放之
    }
}
  
```

(下转第 2245 页)

了OMAP平台在技术战术指标方面时候满足各种便携式测试仪器的设计要求。

5 结束语

嵌入式系统技术每天都在飞速发展,各种先进新异的控制芯片不断涌现,操作系统领域也在蓬勃发展,随着Linux和WIN-CE全面进入嵌入式领域,操作系统将产生变革,最终促进整个嵌入式的发展,便携式仪器具有高集成度、低功耗以及使用灵活的优点,因此广泛运用于各种维修测试领域,尤其适合部队战场装备检测和没有交流电源的机动性强的场合。目前国内也有便携式仪器的应用,但是有些仪器功能单一、技术落后、无法满足部队各种各样的战场需求。为了研制出一系列满足武器装备野外维护测试需求的便携式仪器设备,提高武器维护保障能力,开展了本课题的研究。本课题在功能上满足了用户的需求,更重要的是为下一步研制出一系列满足武器装备野外维护测试需求的高性能便携式仪器系统奠定了坚实的基础。

参考文献:

[1] 马忠梅,马广云,徐英慧,等.ARM 嵌入式处理器结构与应用基

础[M]. 北京:北京航空航天大学出版社,2002.

- [2] 周立功.ARM 微处理器基础与实践[M]. 北京:北京航空航天大学出版社,2003.
- [3] 马忠梅,李善平,慷慨,等.ARM&Linux 嵌入式系统编程[M]. 北京:北京航空航天大学出版社,2004.
- [4] 王学龙.嵌入式 Linux 系统设计与应用[M]. 北京:清华大学出版社,2001.
- [5] Michaelj Pont. C 语言嵌入式系统开发[M]. 陈继辉,译.北京:中国电力出版社,2003.
- [6] 谭浩强,张基温,唐永炎.C 语言程序设计教程[M].北京:国防工业出版社,2003.
- [7] 杨宗德.嵌入式 ARM 系统原理与实例开发[M]. 北京:北京大学出版社,2007.
- [8] 吴旭光,何军红.嵌入式操作系统原理与应用[M].北京:化学工业出版社,2007.
- [9] 杨恒.ARM 嵌入式系统设计及实践[M].西安:西安电子科技大学出版社,2005.
- [10] 黄飞.OMAP 平台的开发及其在便携式仪器中的应用[D]. 长沙:国防科学技术大学,2004.

(上接第 2241 页)

设计两个实验:实验 1,将应用程序 task1 和 task2 置于同一个 DSP;实验 2,将应用程序 task1 和 task2 置于不同 DSP。TF-RTOS 内核在 task1 所在的 DSP 上创建线程 TASK1,在 task2 所在的 DSP 上创建线程 TASK2。由于使用 TF-IPC 只需要指定消息是从哪个线程发送到哪个线程,因此,当改变应用程序 task1 和应用程序 task2 所处 DSP 位置时,不需要对其代码作任何修改,设计非常简单方便。传递 40K 个字的消息,当每次传递的消息大小分别为 64、128、512、1024 和 4096 个字,实验 1 和实验 2 的消息传递时间如表 1 所示。

表 1 消息传递时间/ms

线程位置	消息大小/字					
	64	128	256	512	1024	4096
不同 DSP	18.3	11.2	8.3	6.7	5.9	4.9
相同 DSP	10.0	6.9	5.3	4.5	4.1	3.7

从表 1 可知,对于同一个实验,当每次传递的消息大小不同时,传递时间不同,并且,每次传递的消息大小越大,消息传递时间越短。对于不同的实验,若每次传递的消息大小相同,实验一消息传递时间要比实验二的要短。这是因为两个应用程序位于不同的 DSP 上还需要借助链路口进行 DSP 之间的消息传递,所以消息传递时间要比它们位于相同 DSP 长,但是两种情况下消息传递都非常迅速,能够满足实时性要求。

5 结束语

本文在自主开发的分布式实时操作系统 TF-RTOS 上,设计并实现了 TF-IPC 消息传递机制,该消息传递机制是基于直

接消息传递机制的,它提供了线程同步与异步通信两种通信方式,简化了线程间的通信,增强了系统功能,提高了系统性能,并能满足实时性要求。目前,在需要大量数据通信的 Mandelbrot Set 图形测试实例中应用 TF-RTOS 操作系统,运行效果良好。

参考文献:

- [1] Eyre J, Bier J. The evolution of DSP processors from early architectures to the latest developments [J]. IEEE Signal Processing Magazine, 2000,17(2):43-51.
- [2] 魏晓云,陈杰,曾云. DSP 技术的最新发展及其应用现状[J]. 半导体技术,2003,28(9):18-21.
- [3] 周俊华,李正明,钱小荣. 基于 TMS320F2812 的微型实时内核的实现[J]. 计算机工程与设计,2006,27(23):4543-4546.
- [4] 郑宗汉. 实时系统软件基础[M].北京:清华大学出版社, 2003.
- [5] 安波. pSOS 实时操作系统 IPC 机制的分析与实现[J]. 黑龙江工程学院学报,2001,15(2):33-35.
- [6] Wind River Systems Inc. VSPWorks programmer's guide [M]. Alameda, CA, USA: Wind River Systems Inc, 2002.
- [7] Enea Embedded Technology. OSE_κ kernel user's guide [M]. Kista, Sweden: Enea Embedded Technology, 2004.
- [8] Analog Devices Inc. ADSP-TS201 TigerSHARC processor hardware reference [M]. Norwood, MA, USA: Analog Devices Inc, 2004.
- [9] Analog Devices Inc. ADSP-TS201 TigerSHARC processor programming reference [M]. Norwood, MA, USA: Analog Devices Inc, 2005.