

# 实时操作系统的性能分析和评估

赵立业<sup>1</sup>, 张 激<sup>2</sup>, 游 夏<sup>2</sup>

(1. 国防科学技术大学计算机学院, 长沙 410073; 2. 华东计算技术研究所, 上海 200233)

**摘要:** 针对实时操作系统的基本原理和运行机制, 提出评估系统实时性能的基本参考指标, 并分析各项指标对系统实时性能的影响。采用软件插桩的方法测试不同实时操作系统的性能指标, 根据测试结果对系统实时性能进行综合对比评估。

**关键词:** 实时操作系统; 实时性能指标; 软件插桩

## Analysis and Evaluation of Real-time Operating System Performance

ZHAO Li-ye<sup>1</sup>, ZHANG Ji<sup>2</sup>, YOU Xia<sup>2</sup>

(1. Institute of Computer, National University of Defense Technology, Changsha 410073;

2. East-China Institute of Computer Technology, Shanghai 200233)

**【Abstract】** This paper brings forward basic real-time factors of system evaluation based on the basic principle and operating mechanism of real-time operating system, and analyzes the affection of these factors on system performance. By the method of software stub, it does the factor testing between different systems, also gives a contrastive evaluation result of system real-time performance based on testing data.

**【Key words】** real-time operating system; real-time factor; software stub

### 1 概述

作为操作系统的一个重要分支, 嵌入式实时操作系统为具体实时应用系统提供了基础的运行支撑环境, 其性能在很大程度上决定了具体应用系统的性能。市场上流行的实时操作系统种类繁多、性能特点相似, 许多生产商和相关公司对各种系统进行了性能测试, 但测试方法和测试平台的不同, 导致难以对这些数据进行交叉对比。因此, 需要设计一套简单、有效的测试方案, 对不同系统进行性能对比评估, 从而为具体应用中的系统选取提供客观依据。

实时操作系统是一个可以在有限确定的时间内对异步输入进行处理并输出的信息系统<sup>[1]</sup>。一个高性能的实时操作系统应具备良好的综合性能, 包括系统体系结构、基本系统功能支持(如内存和中断管理)、API支持和稳定性等。而实时性能是系统选取和系统评估最重要的参考指标。

在基于实时操作系统的具体应用中, 很多操作都由系统外部事件引发。实时操作系统主要通过中断和任务方式来处理外部事件。系统对任务的优先级设置, 反映了外部事件的重要程度。实时操作系统的调度策略保证了高优先级的外部事件能得到及时响应。高性能实时系统需要在外部事件复杂频发的环境中, 及时响应并有效处理重要的输入信息, 保证系统高效、稳定的大吞吐量处理能力。评估系统的实时性能, 先要了解系统的工作原理, 对影响实时性能的要害进行分析。

### 2 决定操作系统实时性的重要指标

实时性作为系统的主要特性, 受很多指标的影响<sup>[2]</sup>。由于实时性能指标较多, 因此在具体评估前须提取其中的关键指标。本文根据实时操作系统的特性和一般工作原理, 选取系统调用延时、上下文切换时间、中断延迟时间、中断响应时间和任务切换时间等5项作为系统主要实时性能指标。

(1)系统调用延时, 指系统内核执行常用系统调用所需的平均时间。实时操作系统一般支持多种常用的系统调用, 如任务/文件/信号量的创建/删除、优先级的设置/获取等操作。其中各项系统基本调用在不同情况下的执行时间不尽相同。如任务队列是否为满对任务处理时间的影响, 以及阻塞或非阻塞条件对信号量获取操作时间的影响等。通过对综合样本的有效选取, 可以较为全面的得出系统基本调用的平均时间。

(2)上下文切换时间, 指从当前运行任务的最后一条指令放弃CPU使用权, 到下一个就绪任务的首条指令开始执行的时间, 分为任务分派时间、保存当前任务上下文时间和恢复下一任务上下文时间3个部分。上下文的保存对象为CPU寄存器的内容, 切换时数据会保存在任务自己的堆栈内。上下文恢复的操作正好相反。保存的寄存器数量越多, 上下文切换的工作量越大。因此, 不应以单位时间内可做的上下文切换次数来衡量此项指标。上下文切换时间是系统实时性能的一个重要指标, 通常约为1 μs。

(3)中断延迟时间, 是从接收中断信号到操作系统作出响应, 并完成进入到总中断服务体所需的时间。这段时间包括2个部分: 系统最大关中断时间和硬件开始处理中断到执行总中断服务体首条指令前的时间。后者由硬件决定, 一般小于1 μs, 不作为研究的主要对象。因此, 认为中断延迟时间大小主要取决于系统最大关中断时间。硬实时操作系统的系统关中断时间通常为几微秒。因为在关中断时间内, 系统不能对其他任何优先级的中断做出响应, 所以要尽量减少中断处理程序的执行时间, 并避免出现阻塞调用。具体的中断处

**作者简介:** 赵立业(1981-), 男, 硕士研究生, 主研方向: 嵌入式操作系统; 张 激, 研究员; 游 夏, 工程师

**收稿日期:** 2007-05-26 **E-mail:** zhaobigbear@163.com

理工作可以在系统开中断后通过任务方式进行。减小系统的最大关中断时间，可以更快对外部事件做出响应，从而有效提高系统的整体实时性能。

(4)中断响应时间，指从中断触发到中断处理函数中开始执行首条指令的时间。此指标中包含了中断延迟时间，由4个部分组成：最大关中断时间，保护CPU内部寄存器的时间，总中断服务体的执行时间，以及从总中断服务体转到用户中断处理函数入口时间。中断响应时间与中断延迟联系紧密，可作为判定系统实时性能的指标。通常在硬实时操作系统中，中断响应时间比中断延迟多几微秒。

(5)任务切换时间，是从一个事件引起更高优先级的任务就绪，到高优先级任务开始运行前的时间。通常实时操作系统都具有内核可抢占的功能特性，保证了在任切换后，系统总是执行就绪队列中优先级最高的任务。引起切换的事件有很多，例如高优先级任务的等待资源得到释放和信号量的获取成功等。在操作系统高速运行，需要对大量外部事件进行处理，处理中会频繁引起任务切换。因此，任务切换时间也是影响系统实时性能的一个重要指标。

严格来说，不能对影响系统实时性的所有因素进行测试，以上选取的5个主要指标基本可以客观反映系统的实时性能。针对系统实时性能指标，有很多测试方法和程序，如VxWorks的benchmark测试集。通过软件测试手段，可以较方便地得出指标数据，且测试方法较灵活。

### 3 系统实时性指标的评测技术

#### 3.1 技术原理

在进行系统实时性能指标评测时，通常做法是将被测系统移植到目标机上，在系统运行测试用例时动态采集指标参数，并根据采集数据的解析结果进行系统实时性能评估。由于嵌入式实时系统的特点，如实时性强、内存不丰富、I/O通道少以及操作系统的硬件相关性，使得实时性能指标评测时要采用与其他软件测试不同的策略。本文采用软件插桩技术，基本原理是在被测系统代码的关键位置插入探测代码，来获取系统运行的即时信息。根据插桩类型的不同，可以获取系统结构方面的运行信息、时间数据信息以及程序运行轨迹信息等。

对于系统实时性能指标的测试，本文主要关注系统运行的时间数据信息。插桩操作原理如图1所示，在被测系统代码的入口和出口处，分别插入时间戳跟踪代码，从而在系统运行中记录T1和T2的值，两者的差值即作为被测系统代码的执行时间。

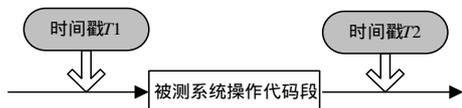


图1 系统软件插桩操作

在系统运行过程中，桩体时间戳的取值为系统时钟的当前值。由于实时操作系统的原子操作时间都在微秒级，因此须选取高精度的系统时钟，以保证获取时间值的精确性和有效性。例如在测试中选取一个300MHz的片载定时器，通过相应的时间戳驱动，可以使其工作时间精度达到0.003μs，基本满足了实时系统性能测试的要求。以上涉及的是有关系统运行的原始时间数据的采集，要得出各项实时性能参数指标，还须对该数据集进行分析提取和计算整合。

#### 3.2 测试要点

通常在实时操作系统运行过程中，外部事件频发，相应的系统处理过程也较为复杂。选取单一的测试用例，不能全面反映系统实时性能的指标参数。因此，要根据影响具体指标的各项因素，来设计不同的用例进行测试，综合测试可以得到更客观的结果。本文以上下文切换时间和中断响应时间2个实时性能指标为例，对测试要点进行分析。

##### (1)上下文切换时间

上下文切换时间分为3个部分，如图2所示。

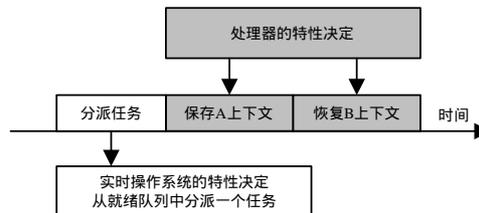


图2 上下文切换时间

有关寄存器的保存和恢复操作主要由硬件决定，因此，要重点分析不同情况下的分派任务时间长度。任务的调度和分派由操作系统内核调度机制和算法决定。在发生任务切换时，系统会查询就绪任务队列，从中选取优先级最高的就绪任务来执行。

测试中先以队列长度为2的情况进行分析：任务1和任务2优先级分别为150和200，都为就绪状态；任务1执行自我挂起的操作，任务2执行恢复任务1的操作；系统循环执行任务切换操作( $1 \times 10^4$ 次)，通过软件插桩获取每次上下文切换的执行时间，并记录此指标的最小值。

然后以任务队列长度为500的情况进行分析：任务1和任务2优先级分别为200和250，都为就绪状态；任务3~任务500的优先级为[20,249]间的随机整数，都为挂起状态；与上一个测试类似，通过执行任务1和任务2间的循环切换操作( $1 \times 10^4$ 次)，记录每次上下文切换的执行时间，并记录该指标在此情况下的最大值。

通过对以上测试结果进行综合分析，可知在一定范围内任务队列长度的增加会延长任务查询时间，进而影响上下文切换时间。通过在不同队列长度情况下对上下文切换指标进行综合测试，分析结果可以更接近指标参数的客观值。

##### (2)中断响应时间

影响中断响应时间的主要因素为中断延迟时间、中断向量表的查询时间和系统的最大关中断时间。其中，中断延迟时间主要由硬件系统决定；由于系统使用了固定的数据结构和运算操作，在系统内核不变的情况下，中断向量表的查询时间是确定的，重点测试变量是系统的最大关中断时间，要获取这个时间值，须对系统代码的每组开/关中断操作(intLock, intUnlock)计时，通过计算获取最大关中断时间。这种做法在实际测试中难以实现，因此，采用大量循环测试的方法(如 $5 \times 10^5$ 次)，用频繁的中断信号使系统在高负载状态下运行，通过记录大量的中断响应时间，得出其中的最大值、最小值及平均关中断时间。

测试过程中，使用辅助时钟作为中断源，定时产生中断信号。一般实时操作系统采用基于优先级的可抢占的调度策略，较低优先级的中断响应可能会被中断，产生中断嵌套；而较高优先级中断的服务程序不确定，执行时间难以计算。在本文中，此指标专指最高优先级中断的中断响应时间。

### 3.3 具体实现

当前测试平台的环境参数如下：

主机端 Pentium-IV 3.00 GHz 处理器 ,1 GB 内存 ,160 GB 硬盘 , Windows XP Professional 操作系统；

目标机端 :Pentium-II 300 MHz 处理器 ,64 MB SDRAM。

基本的测试结构原理如图 3 所示。

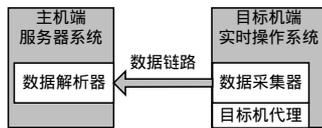


图 3 系统实时性能测试环境结构

对实时操作系统源文件进行代码插桩，然后编译生成目标系统文件，并下载到目标机内存中运行。测试过程中，在目标机实时操作系统运行的同时，会创建一个 BufferManager 任务，负责对采集的测试数据进行传输管理。随着目标机的运行，实时系统会处理一系列的事件操作，类似系统调用和外部响应过程中软件插桩会将即时数据采集，并将数据存放在指定的 Buffer 空间内。当数据采集到一定程度(如 Buffer 满)时，系统就会调用 BufferManager 将缓冲区内的采集数据通过链路发回给主机端。主机端的数据采集器负责接收测试数据，并对数据进行解析处理，从而得出量化的测试数据结果。通过查看测试结果，可以客观地了解系统各项实时性指标参数，对系统实时性进行整体评估。

关于有针对性的具体测试用例，主要参考一些已有的 Benchmark 测试集。通过对用例的大量测试，可得到性能指标的运行极值；通过求取平均值，可以得出系统实时性指标参数。对指标参数进行解析，在量化基础上对系统实时性进行总体评估。

### 4 系统实时性能的对标

在复杂的测控应用中，须使用对实时性要求很高的实时操作系统。关键领域中，系统若不能对事件及时响应，就可能导致巨大损失。因此，在选择操作系统时，须对系统实时性进行深入的测试分析。根据具体指标参数和测试方法，可以对被测系统的实时性有具体了解。另外，分别对不同实时操作系统进行测试分析及对比评估<sup>[3]</sup>，有助于正确选择实时系统，以适应具体的应用需要。

本文依据已经计算整合过的指标参数进行系统实时性能评估。测试系统对象分别为 WindRiver 的 VxWorks 5.4 版本<sup>[4]</sup>，和 RTLinux 3.0 版本<sup>[5]</sup>。这是 2 种适用于苛刻实时性要求的操作系统。主要实时性能指标的测试数据均值对比如表 1 所示。

(上接第 282 页)

采集器在通信前，将通过比较节点总数得出最优路径为 R0, R2, R4, R6, ..., R30, R28, R26, ..., R4, R2, R0，节点总数为 15，因此，按此路径进行访问。

静态路由算法原理与动态路由相同，不同的是静态路由算法在出厂时确定了每块电能表的路由表，或通过软件对其路由表进行了设置。这样就不必每次由采集器动态维护节点路径。但静态路由一旦须新增电表或更换坏表都要在现场重新设置路由表。

### 3 结束语

本文指出自动中继、动态路由等功能的实现是基于 433M 无线技术的远程抄表系统的实施难点，并设计了一套无线自

表 1 系统实时性指标的参数值对比

	VxWorks	RTLinux
上下文切换	0.40	0.49
中断响应时间	3.20	4.30
二进制信号量获取(不可用)	0.31	0.34
二进制信号量获取(可用)	0.30	0.31
TaskLock	0.05	0.07
TaskUnlock	0.10	0.08
消息队列发送(非挂起)	1.01	0.96
消息队列发送(挂起)	1.70	1.61
消息队列发送(队列满)	0.63	0.57

表 1 数据显示，RTLinux 的上下文切换时间指标参数比 VxWorks 高，但两者指标参数值都在 1 μs 以下，此项性能都比较高；中断响应方面，VxWorks 操作系统的性能指标参数比 RTLinux 的低 34%，表现更优秀；信号量方面，测试中把二进制信号量 SemB 操作作为主要研究对象；关于 SemB 的 Take 操作，在信号量不可用的情况下，VxWorks 指标表现较好，而在信号量不可用的情况下，两者性能表现相当。通常任务加/解锁操作在应用中会成对出现，所以要通过参数和的形式来综合衡量此指标。由表 1 可见，2 种系统在任务加/解锁方面各有优势，综合指标基本持平。在消息队列的发送方面，综合数据显示，RTLinux 的性能指标更优。2 种实时操作系统对于进程间通信的不同实现，使它们在消息队列操作性能方面有所不同。

综合各项实时性关键指标，笔者认为 VxWorks 和 RTLinux 在上下文切换、中断响应、信号量和消息队列操作方面各有优势，都体现了高性能硬实时操作系统的指标要求。

### 5 结束语

通过对实时系统的实时性进行指标测试和评估，可以识别出不同系统的细微差别。目前流行的很多系统都是优秀的强实时操作系统，各具特色。充分理解并掌握实时操作系统的实现技术、性能指标和欠缺，有助于在实时性应用中做出科学的评价和合理的选择。

#### 参考文献

- [1] Timmerman M. Embedded Systems Definition Taxonomies Field [EB/OL]. (2007-05-20). <http://www.dedicated-systems.com>.
- [2] WindRiver Systems Inc. VxWorks Programmer's Guide[Z]. 1999.
- [3] Omimo Inc. Evaluation Report Definition[EB/OL]. (2007-05-16). <http://www.dedicated-systems.com>.
- [4] Benjamin I. Performance Analysis of VxWorks and RTLinux[D]. NY, USA: Columbia University, 2004.
- [5] Yodaiken V. The RTLinux Approach to Hard Real Time[EB/OL]. (2007-10-07). <http://rtlinux.org/documents/papars/whitepaper.html>.

适应路由算法，在节约方案中通信设备成本的同时提高了系统通信的可靠性。本文算法也适用于需要中继节点传输数据的场合。

#### 参考文献

- [1] 彭远强. 无线抄表系统的设计与实现[J]. 现代电子技术, 2006, 24(9): 110-112.
- [2] 崔 玮, 张其善. 基于 GPRS 的无线抄表系统的设计与实现[J]. 无线电工程, 2006, 26(2): 54-56.
- [3] 李正友, 茅忠明. 实用“三表”一体化电脑网络与管理系统的的设计[J]. 计算机自动测量与控制, 2001, 9(6): 51-52.

