

硬件实现 RAID 与软件实现 RAID 的比较

陈平仲

(广东外语外贸大学教育技术中心, 广州 510000)

摘要:现在的服务器中存在多种 RAID 实现方式, RAID 系统提供了数据冗余备份功能,能帮助预防信息丢失。本文讨论了硬件 RAID 和软件 RAID 实现的差异和性价比的权衡。并分析了其对系统性能的影响,以便帮助管理员了解哪种 RAID 实现方式最能满足他们的需求。

关键词:RAID ;IOP ;IOC ;RAID 卡

1 RAID 基本知识介绍

RAID 的全称是廉价磁盘冗余阵列(Redundant Array of Inexpensive Disks),于 1987 年由美国 Berkeley 大学的两名工程师提出的。

RAID 出现的最初目的是将多个容量较小的廉价硬盘合并成为一个大容量的“逻辑盘”或磁盘阵列,实现提高硬盘容量和性能的功能。随着 RAID 技术的逐渐普及应用,RAID 技术的各方面得到了很大的发展。现在,RAID 从最初的 RAID0-RAID5,又增加了 RAID0+1 和 RAID0+5 等不同的阵列组合方式,可以根据不同的需要实现不同的功能,扩大硬盘容量,提供数据冗余,或者是大幅度提高硬盘系统的 I/O 吞吐能力。

RAID 技术主要有三个特点:

- ①通过对硬盘上的数据进行条带化,实现对数据成块存取,减少硬盘的机械寻道时间,提高数据存取速度;
- ②通过对一磁盘阵列中的几块硬盘同时读取,减少硬盘的机械寻道时间,提高数据存取速度;
- ③通过镜像或者存储奇偶校验信息的方式,实现对数据的冗余保护。

2 RAID 分类

对不同类型的 RAID 进行分类如图 1 所示。

从图 1 可知,硬件 RAID 可以进一步划分为基

于 I/O 处理器(IOP)和基于 I/O 控制器(IOC)两种类型,而软件 RAID 可以进一步划分为基于驱动程序的和基于 OS 的。

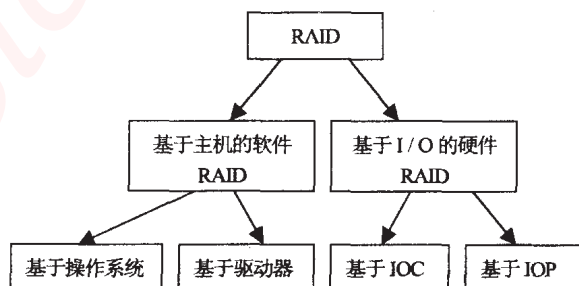


图 1

(1) 基于 I/O 处理器(IOP)的 RAID

这是最为常用的方法,也往往是最为昂贵的,其功能最多、性能最好。基于 IOP 的 RAID 使用专用的处理器、内存子系统和外设,在磁盘上完成 RAID 功能。板内的 IOP 执行 RAID 引擎,处理磁盘阵列的重建、RAID 级别的移植以及错误恢复等其他工作。基于 IOP 的 RAID 要么嵌入在服务器中,作为主板内 RAID(ROMB),要么放置在宿主机的总线适配器上(HBA)。

(2) 基于 I/O 控制器(IOC)的 RAID

这种类型使用 I/O(磁盘接口)控制器的处理器和内存资源来执行 RAID 引擎。尽管它看起来类似于基于 IOP 的 RAID,但基于 IOC 的 RAID 的特

性和功能有限，因为它共享磁盘 I/O 控制器有限的处理器和内存资源。因为它占用的体积较小，而且成本较低，因此基于 IOC 的 RAID 往往嵌入在服务器平台中。

(3) 基于驱动程序的 RAID

这种软件实现方式与特定的磁盘控制器的驱动程序集成在一起。这个驱动程序中包含有在 OS 环境中运行 RAID 引擎的代码，BIOS 在 OS 前（启动）环境中运行 RAID 引擎。在 OS 中，这种方法看起来类似于 IOP 和 IOC 的 RAID，因为它包含 RAID 空间的真正的物理驱动器对于 OS 来说不可见。但是，基于驱动程序的 RAID 完全依赖系统处理器和内存资源来执行 RAID 操作，在 CPU 利用率高的环境中将会影响到系统的性能。

(4) 基于 OS 的 RAID

这种类型的软件 RAID，通常被实施为一个过滤驱动程序，与 OS 打包在一起，使用主机系统的处理器和内存资源来执行 RAID 引擎。基于 OS 的 RAID 与磁盘控制器的类型无关，使用磁盘控制器的驱动程序进行磁盘 I/O。

3 硬件 RAID 与软件 RAID 的比较

下面就对每种 RAID 方法的优劣进行了逐项比较。

(1) 基于 IOP 的 RAID

●特性：高性能，高成本。

●特点：对 CPU 密集型应用将会因为 RAID 引擎运行在专用的 RAID 处理器上，减轻了系统处理器的负担，从而提高性能。硬件 RAID 通常使用大型的电池支持的缓存内存，提高了性能和可靠性，专用的 I/O 处理器处理所有的 RAID 功能并完成重建、错误恢复以及系统自己所具有的特定功能。因为 I/O 处理器可以独立于 OS 完成操作，所以它可以在不影响主机 CPU 的性能之前运行很多 RAID 任务。

●优势：在通常情况下，其性能最优，不消耗主机服务器的 CPU 时钟周期；提供了最大的技术特性；为多种总线 and 操作系统提供灵活的方法。

●劣势：其成本最高，比其他方法要求更多的组件空间。

(2) 基于 IOC 的 RAID

●特性：占用空间小，费用低。

●特点：它是由系统共享磁盘控制器中的处理器和内存来执行 RAID 引擎，但这些系统的特性要比基于 IOP 的硬件 RAID 少，I/O 控制器中的 RAID 固件处理错误恢复和重建算法。基于 IOC 的 RAID 在理论上都与 RAID 的等级类型没有限制，但最常用的等级是 RAID-0 和 RAID-1，因为在 IOC 中为 RAID 引擎提供的处理器和内存资源有限。

●优势：要求的组件空间最少；成本与基于 IOP 的 RAID 相比极低；在后台运行任务，与服务器操作系统无关。

●劣势：性能和提供的技术特性都有限。

(3) 基于驱动程序的 RAID

●特性：低成本，高管理费用。

●特点：基于设备驱动程序的 RAID 实现方法使用系统处理器和内存来处理 RAID 功能。基于驱动程序的 RAID 往往在 ATA 实现中使用，但也可以在 SCSI 系统中使用。IOC 仅作为磁盘接口，但基于驱动程序的 RAID 可以利用提供商在 IOC 中特有的功能。RAID 配置数据往往保存在磁盘上。基于驱动程序的 RAID 系统往往限制于使用 RAID-0、RAID-1 和 RAID-10，但实际上没有物理限制。

●优势：成本非常低，实现比较容易。

●劣势：依靠主机 CPU 时钟周期执行任务；不适合高 CPU 使用环境（特别是数据库中）；使用为特定磁盘控制器专门优化的复杂驱动程序；与服务器操作系统相关；会受到主机操作系统崩溃或内存损坏的影响。

(4) 基于操作系统的 RAID

●特性：没有附加费用，管理费用高。

●特点：基于 OS 的 RAID 系统包含在宿主机的 OS 中。RAID 引擎位于 OS 中，作为设备驱动程序层之上的过滤驱动程序层。软件 RAID 选件现在在 Microsoft(r) Windows(r)、Novell(r) NetWare(r) 和 Linux(r) 操作系统中都提供，这些选件的设置都与控制器的类型和磁盘的类型无关。

●优势：系统免费，与操作系统集成；无需特殊的驱动程序或硬件；优于基于驱动程序的解决方案；与磁盘控制器无关，并且兼容多种控制器。

●劣势：要求手工干预以启动设备；在某些操作系统中无法使用镜像方式启动驱动器；必须依靠

主机 CPU 时钟周期执行任务 ;不适合高 CPU 使用环境(特别是在数据库系统中);后台无法运行操作系统之外的任务 ;会受到主机操作系统崩溃或内存损坏的影响。

结 语

软件 RAID 不需要额外成本就可以提供容错磁盘管理系统 ,允许所有硬盘驱动器作为一个单一的驱动器而且越过硬盘故障继续执行。在确定的情况下 ,软件 RAID 超过非 RAID 磁盘阵列取得性能的增加 ,因为操作系统可以同时写和读多个磁盘。但是硬件 RAID 提供的 RAID 性能好 ,因为它有专门的 RAID 控制芯片来实现 RAID。此外 ,硬件 RAID 的设置不依赖于操作系统 ,因为 RAID 在操作系统启动前就已经启动了。当然 ,要实现硬件 RAID 需要购置额外的硬件设备 ,比如 RAID 卡(主板本身集成了 RAID 适配器的用户除外)、硬盘等。

每种 RAID 实现方法都具有一定的优势和劣势 ,IT 管理员可以从这些方法中选择 ,均衡考虑各项需求 ,例如性能、运行后台任务的能力、使用的简易性 (例如费用和所需的部件空间)等。根据组织特定的 IT 需要定制一种 RAID 方法 ,可以帮助管理员获得经济高效的数据保护。

参考文献

- [1] Hogge, Steve and Bo Zhou. Understanding Server-based RAID Using Ultra ATA Technology. Dell Power Solutions , March 2002. [http ://www.dell.com/us/en/esg/topics/power_ps1q02-hogge.htm](http://www.dell.com/us/en/esg/topics/power_ps1q02-hogge.htm)
- [2] Luse, Paul. Understanding RAID-5 and I/O Processors. Dell Power Solutions, May 2003. [http ://www.dell.com/us/en/esg/topics/power_ps2q03-luse.htm](http://www.dell.com/us/en/esg/topics/power_ps2q03-luse.htm)
- [3] [http ://www1.ap.dell.com/content/topics/topic.aspx/ap/topics/power/zhcn/power?c=cn&l=zh&s=bsd](http://www1.ap.dell.com/content/topics/topic.aspx/ap/topics/power/zhcn/power?c=cn&l=zh&s=bsd)

(收稿日期 2004-07-12)

A Comparative Study of Hardware and Software RAID

CHEN Ping-zhong

Abstract : There are many techniques to implement a RAID system in a server system. A RAID system is capable of data redundancy and backup. Therefore, it can prevent the loss of information. In this paper, the differences between hardware and software RAID, balancing performance and price of these two RAID systems are discussed. The effects of these two RAID systems to the performance of a server system are analyzed. This analysis helps system administrators understand what technique of RAID is most suitable to their requirements.

Key words : RAID ; IOP ; IOC ; RAID CARD

嵌入式资源免费下载

总线协议:

1. [基于 PCIe 驱动程序的数据传输卡 DMA 传输](#)
2. [基于 PCIe 总线协议的设备驱动开发](#)
3. [CANopen 协议介绍](#)
4. [基于 PXI 总线 RS422 数据通信卡 WDM 驱动程序设计](#)
5. [FPGA 实现 PCIe 总线 DMA 设计](#)
6. [PCI Express 协议实现与验证](#)
7. [VPX 总线技术及其实现](#)
8. [基于 Xilinx FPGA 的 PCIE 接口实现](#)
9. [基于 PCI 总线的 GPS 授时卡设计](#)
10. [基于 CPCI 标准的 6U 信号处理平台的设计](#)
11. [USB30 电路保护](#)
12. [USB30 协议分析与框架设计](#)
13. [USB 30 中的 CRC 校验原理及实现](#)
14. [基于 CPLD 的 UART 设计](#)
15. [IPMI 在 VPX 系统中的应用与设计](#)
16. [基于 CPCI 总线的 PMC 载板设计](#)
17. [基于 VPX 总线的工件台运动控制系统研究与开发](#)
18. [PCI Express 流控机制的研究与实现](#)
19. [UART16C554 的设计](#)
20. [基于 VPX 的高性能计算机设计](#)
21. [基于 CAN 总线技术的嵌入式网关设计](#)
22. [Visual C 串行通讯控件使用方法与技巧的研究](#)
23. [IEEE1588 精密时钟同步关键技术研究](#)
24. [GPS 信号发生器射频模块的一种实现方案](#)
25. [基于 CPCI 接口的视频采集卡的设计](#)
26. [基于 VPX 的 3U 信号处理平台的设计](#)
27. [基于 PCI Express 总线 1394b 网络传输系统 WDM 驱动设计](#)
28. [AT89C52 单片机与 ARINC429 航空总线接口设计](#)
29. [基于 CPCI 总线多 DSP 系统的高速主机接口设计](#)
30. [总线协议中的 CRC 及其在 SATA 通信技术中的应用](#)
31. [基于 FPGA 的 SATA 硬盘加解密控制器设计](#)
32. [Modbus 协议在串口通讯中的研究及应用](#)
33. [高可用的磁盘阵列 Cache 的设计和实现](#)
34. [RAID 阵列中高速 Cache 管理的优化](#)

35. [一种新的基于 RAID 的 CACHE 技术研究与实现](#)
36. [基于 PCIE-104 总线的高速数据接口设计](#)
37. [基于 VPX 标准的 RapidIO 交换和 Flash 存储模块设计](#)
38. [北斗卫星系统在海洋工程中的应用](#)
39. [北斗卫星系统在远洋船舶上应用的研究](#)

VxWorks:

1. [基于 VxWorks 的多任务程序设计](#)
2. [基于 VxWorks 的数据采集存储装置设计](#)
3. [Flash 文件系统分析及其在 VxWorks 中的实现](#)
4. [VxWorks 多任务编程中的异常研究](#)
5. [VxWorks 应用技巧两例](#)
6. [一种基于 VxWorks 的飞行仿真实时管理系统](#)
7. [在 VxWorks 系统中使用 TrueType 字库](#)
8. [基于 FreeType 的 VxWorks 中文显示方案](#)
9. [基于 Tilcon 的 VxWorks 简单动画开发](#)
10. [基于 Tilcon 的某武器显控系统界面设计](#)
11. [基于 Tilcon 的综合导航信息处理装置界面设计](#)
12. [VxWorks 的内存配置和管理](#)
13. [基于 VxWorks 系统的 PCI 配置与应用](#)
14. [基于 MPC8270 的 VxWorks BSP 的移植](#)
15. [Bootrom 功能改进经验谈](#)
16. [基于 VxWorks 嵌入式系统的中文平台研究与实现](#)
17. [VxBus 的 A429 接口驱动](#)
18. [基于 VxBus 和 MPC8569E 千兆网驱动开发和实现](#)
19. [一种基于 vxBus 的 PPC 与 FPGA 高速互联的驱动设计方法](#)
20. [基于 VxBus 的设备驱动开发](#)
21. [基于 VxBus 的驱动程序架构分析](#)
22. [基于 VxBus 的高速数据采集卡驱动程序开发](#)

Linux:

1. [Linux 程序设计第三版及源代码](#)
2. [NAND FLASH 文件系统的设计与实现](#)
3. [多通道串行通信设备的 Linux 驱动程序实现](#)
4. [Zsh 开发指南-数组](#)

5. [常用 GDB 命令中文速览](#)
6. [嵌入式 C 进阶之道](#)
7. [Linux 串口编程实例](#)
8. [基于 Yocto Project 的嵌入式应用设计](#)
9. [Android 应用的反编译](#)
10. [基于 Android 行为的加密应用系统研究](#)
11. [嵌入式 Linux 系统移植步步通](#)
12. [嵌入式 CC++语言精华文章集锦](#)
13. [基于 Linux 的高性能服务器端的设计与研究](#)
14. [S3C6410 移植 Android 内核](#)
15. [Android 开发指南中文版](#)
16. [图解 Linux 操作系统架构设计与实现原理（第二版）](#)
17. [如何在 Ubuntu 和 Linux Mint 下轻松升级 Linux 内核](#)
18. [Android 简单 mp3 播放器源码](#)
19. [嵌入式 Linux 系统实时性的研究](#)
20. [Android 嵌入式系统架构及内核浅析](#)
21. [基于嵌入式 Linux 操作系统内核实时性的改进方法研究](#)
22. [Linux TCP IP 协议详解](#)
23. [Linux 桌面环境下内存去重技术的研究与实现](#)
24. [掌握 Android 7.0 新增特性 Quick Settings](#)
25. [Android 应用逆向分析方法研究](#)
26. [Android 操作系统的课程教学](#)
27. [Android 智能手机操作系统的研究](#)
28. [Android 英文朗读功能的实现](#)
29. [基于 Yocto 订制嵌入式 Linux 发行版](#)
30. [基于嵌入式 Linux 的网络设备驱动设计与实现](#)
31. [如何高效学习嵌入式](#)
32. [基于 Android 平台的 GPS 定位系统的设计与实现](#)

Windows CE:

1. [Windows CE.NET 下 YAFFS 文件系统 NAND Flash 驱动程序设计](#)
2. [Windows CE 的 CAN 总线驱动程序设计](#)
3. [基于 Windows CE.NET 的 ADC 驱动程序实现与应用的研究](#)
4. [基于 Windows CE.NET 平台的串行通信实现](#)
5. [基于 Windows CE.NET 下的 GPRS 模块的研究与开发](#)
6. [win2k 下 NTFS 分区用 ntldr 加载进 dos 源代码](#)
7. [Windows 下的 USB 设备驱动程序开发](#)
8. [WinCE 的大容量程控数据传输解决方案设计](#)

9. [WinCE6.0 安装开发详解](#)
10. [DOS 下仿 Windows 的自带计算器程序 C 源码](#)
11. [G726 局域网语音通话程序和源代码](#)
12. [WinCE 主板加载第三方驱动程序的方法](#)
13. [WinCE 下的注册表编辑程序和源代码](#)
14. [WinCE 串口通信源代码](#)
15. [WINCE 的 SD 卡程序\[可实现读写的源码\]](#)
16. [基于 WinCE 的 BootLoader 研究](#)
17. [Windows CE 环境下无线网卡的自动安装](#)
18. [基于 Windows CE 的可视电话的研究与实现](#)

PowerPC:

1. [Freescale MPC8536 开发板原理图](#)
2. [基于 MPC8548E 的固件设计](#)
3. [基于 MPC8548E 的嵌入式数据处理系统设计](#)
4. [基于 PowerPC 嵌入式网络通信平台的实现](#)
5. [PowerPC 在车辆显控系统中的应用](#)
6. [基于 PowerPC 的单板计算机的设计](#)
7. [用 PowerPC860 实现 FPGA 配置](#)
8. [基于 MPC8247 嵌入式电力交换系统的设计与实现](#)
9. [基于设备树的 MPC8247 嵌入式 Linux 系统开发](#)
10. [基于 MPC8313E 嵌入式系统 UBoot 的移植](#)
11. [基于 PowerPC 处理器 SMP 系统的 UBoot 移植](#)
12. [基于 PowerPC 双核处理器嵌入式系统 UBoot 移植](#)

ARM:

1. [基于 DiskOnChip 2000 的驱动程序设计及应用](#)
2. [基于 ARM 体系的 PC-104 总线设计](#)
3. [基于 ARM 的嵌入式系统中断处理机制研究](#)
4. [设计 ARM 的中断处理](#)
5. [基于 ARM 的数据采集系统并行总线的驱动设计](#)
6. [S3C2410 下的 TFT LCD 驱动源码](#)

7. [STM32 SD 卡移植 FATFS 文件系统源码](#)
8. [STM32 ADC 多通道源码](#)
9. [ARM Linux 在 EP7312 上的移植](#)
10. [ARM 经典 300 问](#)
11. [基于 S5PV210 的频谱监测设备嵌入式系统设计与实现](#)
12. [Uboot 中 start.S 源码的指令级的详尽解析](#)
13. [基于 ARM9 的嵌入式 Zigbee 网关设计与实现](#)
14. [基于 S3C6410 处理器的嵌入式 Linux 系统移植](#)
15. [CortexA8 平台的 \$\mu\$ C-OS II 及 LwIP 协议栈的移植与实现](#)
16. [基于 ARM 的嵌入式 Linux 无线网卡设备驱动设计](#)
17. [ARM S3C2440 Linux ADC 驱动](#)
18. [ARM S3C2440 Linux 触摸屏驱动](#)
19. [Linux 和 Cortex-A8 的视频处理及数字微波传输系统设计](#)
20. [Nand Flash 启动模式下的 Uboot 移植](#)
21. [基于 ARM 处理器的 UART 设计](#)

Hardware:

1. [DSP 电源的典型设计](#)
2. [高频脉冲电源设计](#)
3. [电源的综合保护设计](#)
4. [任意波形电源的设计](#)
5. [高速 PCB 信号完整性分析及应用](#)
6. [DM642 高速图像采集系统的电磁干扰设计](#)
7. [使用 COMExpress Nano 工控板实现 IP 调度设备](#)
8. [基于 COM Express 架构的数据记录仪的设计与实现](#)
9. [基于 COM Express 的信号系统逻辑运算单元设计](#)
10. [基于 COM Express 的回波预处理模块设计](#)
11. [基于 X86 平台的简单多任务内核的分析与实现](#)
12. [基于 UEFI Shell 的 PreOS Application 的开发与研究](#)
13. [基于 UEFI 固件的恶意代码防范技术研究](#)
14. [MIPS 架构计算机平台的支持固件研究](#)
15. [基于 UEFI 固件的攻击验证技术研究](#)

Programming:

RT Embedded <http://www.kontronn.com>

1. [计算机软件基础数据结构 - 算法](#)