基于排队网络 RAID 存储系统的性能评价模型

张　燕[1]，　胡英坚[1]，　姜　涛[2]

(1. 空军航空大学 基础部，吉林 长春　130022；
2. 长春工业大学 机电工程学院，吉林 长春　130012)

摘　要：利用排队网络对软件 RAID 系统建立性能评价模型，并通过 MVA 方法来对软件系统的性能进行分析，理论值与测试值进行了对照，结果表明，建立的模型基本上反映了系统性能在不同负载下的变化趋势，验证了系统的可用性。

关键词：RAID；磁盘阵列；排队网络；性能评价

中图分类号：TP309　　**文献标志码**：A　　**文章编号**：1674-1374(2010)04-0471-05

Performance evaluation model for RAID storage systems based on queueing Network

ZHANG Yan[1]，　HU Ying-jian[1]，　JIANG Tao[2]

(1. Department of Basic，Aviation University of Air Force，Changchun 130022，China
2. School of Mechatronic Engineering，Changchun University of Technology，Changchun 130012，China)

Abstract: A performance evaluation model is built for the RAID software system with queuing network. With MVA method we analyze performance of the software and compare the theoretical value with the measured one. The results show that the model can basically express the trend of the performance with different load, so the usability of the system is verified.

Key words: RAID; disk array; queueing network; performance evaluation.

0　引　言

存储系统是计算机系统的重要组成部分，如何准确地预测和评价存储系统的性能，及时发现系统设计中的瓶颈和主要的性能影响因素，是提高存储系统性能的先决条件。RAID 技术是一种使用非常广泛的存储技术，目前，对于这种存储系统的性能评价模型大多数是基于硬件 RAID 的性能分析[1]，与硬件 RAID 控制器相比，软件 RAID 具有廉价、灵活性高等特点，是低端服务器存储系统的最佳解决方案，但是对软件 RAID 的定量分析工作很少[2]。文中针对磁盘阵列读写流程的特

点,采用闭合排队网络的建模方法对软件 RAID 的性能进行了评价,并通过实验证明了该方法基本可以反映真实系统的性能状况。

1 排队网络的基础知识

排队网络属于排队论中的一个分支,利用排队网络模型,可以分析和评价系统的多种性能指标。

1.1 排队网络的的概念

一个排队网络是一个有向图 $G=(V,E)$,由一组顶点 $V=\{1,2,\cdots,M\}$和一组边 $E\subseteq V\times V$ 组成,每个节点代表一个服务站(Service Center),服务站包括一个队列和多个服务员,E 是边的集合,表示顾客流的可能通路。

一般采用节点的服务时间和节点之间的选道概率来刻画工作负载,最简单的排队网络是所有的顾客具有完全相同的性,称为单一类别排队网络。同一网络中如果存在多种不同类型的顾客,称为多类排队网络,这种情况下,每类顾客具有不同的选道行为。如果网络中的顾客数为常数,所有顾客永远循环流动,这样的排队网格称为闭合排队网络。

1.2 基本运算定律

运算定律抽象出了计算机系统多种性能间的关系,忽略系统中性能因素的随机行为,可以快速分析系统的平均性能。

little 定律:

$$N = X * R$$

式中:N ——队列系统中的平均顾客数;

X ——系统的平均吞吐量;

R ——系统对顾客的平均响应时间。

1)服务节点的类型为下列情况之一:

服务规则为处理器共享节点,服务由队列中所有顾客平等分享;

服务规则为先来先服务;

服务规则为后来先服务,抢占式;

无限服务员节点或者延时节点,为顾客立即提供服务,服务时间服从一定分布。

2)服务节点的服务时间为以下几种类型之一:

单服务员固定速率;

无限服务员,适用于上面提到的延时节点。

1.3 排队网络的分析方法

采用平均值法(MVA)对 RAID 系统进行分析,这种方法可以避免复杂的稳定状态概率。多类顾客闭合排队网络[3]的 MVA 分析方法如下:

假设一个闭合排队网络中有 C 类顾客,用向量 $M=(M_1,M_2,\cdots,M_C)$ 表示,其中 M_c 代表第 c $(0<c<C)$ 类顾客在网络中的个数,假设排队网络中有 K 个服务节点,对于每一个服务节点 k,$V_{c,k}$代表第 c 类顾客对服务节点 k 的访问概率。$S_{c,k}$代表服务节点 k 对 c 类顾客的平均服务时间,则节点 k 对 c 类顾客的服务需求为:$D_{c,k}=V_{c,k}*S_{c,k}$,用 X 表示吞吐量(X_c 代表整个系统对 c 类顾客的吞吐量,X_k 表示节点 k 的吞吐量,其它符号以此类推),R 代表响应时间,Q 代表队列长度,则多类顾客闭合排队网络的 MVA 分析法基于以下 3 个基本公式[4-5]:

1)对于 c 类顾客,在节点 k 的平均响应时间 $R_{c,k}$等于顾客排队时间和服务时间之和,对于单队列节点,表示为:

$$R_{c,k}(M)=D_{c,k}+D_{c,k}*Q_k(M-I_c) \quad (1)$$

2)对于 c 类顾客在整个系统的吞吐量$X_c(M)$表示为:

$$X_c(M)=\frac{M_c}{\sum_{k}^{K}R_{c,k}(M)} \quad (2)$$

3)对 c 类顾客,在节点 k 的排队长度表示为 $Q_{c,k}(M)=X_c(M)*R_{c,k}(M)$,于是节点 k 的总的队列长度为:

$$Q_k(M)=\sum_{c}^{C}Q_{c,k}(M) \quad (3)$$

式(1)~式(3)对网络中任务数形成一种递归关系,每递归一次,系统中将减少一个顾客。因此,只要给出递归初始条件,该算法即可在有限步之内计算出整个系统的吞吐量和平均响应时间。递归初始条件是显然的,对第 k 个节点,在系统顾客数为 0 时:

$$Q_k(\mathbf{0})=0 \quad (4)$$

2 软件 RAID 系统的性能评价模型

用上述方法对软件 RAID 系统建立性能评价模型,利用评价模型对 RAID 系统进行分析。

2.1 模型的建立

影响软件 RAID 性能主要有 3 个部件:CPU、内存以及磁盘驱动器。CPU 节点处理请求的映射以及校验计算,内存缓冲条纹数据,磁盘驱动器从内存中接收读写请求,完成实际的 I/O 操作,

这 3 个部件就为模型中的服务节点。对 RAID 系统的读写请求抽象为顾客。请求负载采用同步方式进行读写操作,也就是说在稳定状态下当一个请求结束后,CPU 会马上产生出一个类型和大小都相同的请求来代替,这样,在一段时间内,系统中的请求数是一定的。

一个 RAID 系统实际上就是一个多类闭合排队网络。按照读写请求的流程,软件 RAID 的排队网络模型如图 1 所示。

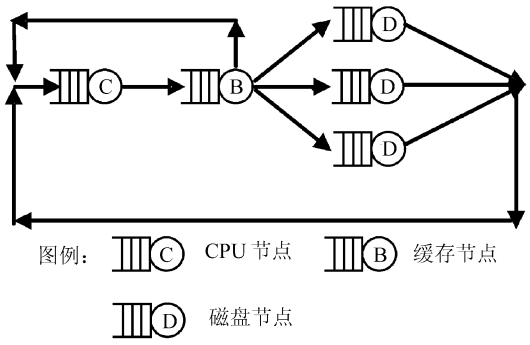

图 1　软件 RAID 系统的排队网络模型

图中,读写请求由 CPU 节点发出,进入缓存节点,当请求的数据在缓存中存在时,读写命中,请求直接返回;否则,要经过 RAID 映射程序,将请求下发到对应的磁盘节点上。

下面以实际的 RAID5 系统为例,讨论其在不同负载条件下的性能。假设请求的大小为 b(kB),各类节点的平均服务时间(用 ST 表示)的计算方式如下:

1) CPU 节点:由于文中的系统完全是由软件实现的,而且所进行的实验对系统来说是一种压力实验,所以,将 CPU 抽象成为 FCFS 的排队节点,其服务时间对所有任务是一个常数。

2) 缓存节点:关键的问题是缓存命中率的问题,Linux RAID 的调度机制存在着条纹预读机制,缓存命中率和请求的顺序程度有关。当对 RAID5 的某个条纹单元进行读写操作的时候,总是会将相关的整个条纹进行预读,这样对于顺序的请求缓存命中率就比较高。文中讨论的缓存命中仅指读请求命中,对于写请求,从系统的可靠性考虑,一般采用同步写方式。预读命中的概率可以采用如下方法计算:

一次实际磁盘操作读取的数据大小 S_{disk_read} 为请求大小 S_{read_req} 和预读大小 S_{read_ahead} 之和,表示为:

$$S_{disk\_read} = S_{read\_req} + S_{read\_ahead} \tag{5}$$

定义一个顺序度的概念,用 P_{seq} 表示。顺序度表示顺序请求在整个请求负载中所占的比率。如果请求负载是完全随机的,则顺序度为 0;如果是完全顺序的请求,则顺序度为 1。有了这一概念,那么一次实际的磁盘读取操作能够满足的读请求命中的个数为:

$$N_{req\_per\_disk\_read} = 1 + P_{seq} * \frac{S_{read\_ahead}}{S_{read\_rep}} \tag{6}$$

也就是说,经历 $N_{req\_per\_disk\_read}$ 次读请求就需要进行一次磁盘操作,也就是缓存不命中,于是缓存不命中的概率为 $1/N_{req\_per\_disk\_read}$,所以预读的缓存命中率为:

$$P_{hit} = 1 - \frac{1}{N_{req\_per\_disk\_read}} \tag{7}$$

则缓存节点平均服务时间表示为:

$$ST_{buf}(b) = \frac{b}{V_{RAM}} \tag{8}$$

3) 磁盘节点:一次磁盘操作包括磁盘寻道时间、磁盘旋转时间和数据传输时间,前两者统称为定位时间。磁盘定位时间是磁盘队列长度的函数,二者的关系为:

$$T_{disk\_pos} = a + \frac{b}{1 + Q_{disk}} \tag{9}$$

而磁盘传输时间为数据量的线性函数表示为:

$$T_{disk\_transfer}(b) = \frac{b}{V_{disk}} \tag{10}$$

式中:b ——传输数据量大小;

V_{disk} ——磁盘的传输速率。

为了简化计算,我们忽略磁盘的定位时间,所以:

$$ST_{disk}(b) \approx \frac{b}{V_{disk}} \tag{11}$$

2.2　读写请求服务的需求分析

对 RAID 系统的应用主要包括单用户的大数据访问和多用户的随机访问,而 RAID5 对大数据写和小数据写的处理方式完全不同,所以将请求负载分为读数据、大数据写和小数据写 3 类,它们对各节点的服务需求是不同的。

2.2.1　读和大数据写请求服务需求分析

大数据写请求是指请求的数据远远大于 RAID5 条纹的长度,这样在 RAID 层为完全条纹写的方式。这种方式和读请求的数据流程基本一致,只是有以下几点不同:

1) 讨论的为同步方式,所以大数据写的缓存命中率为 0。

2) 大数据写增加了为计算校验单元而进行异或运算的时间,所以,CPU 节点的平均响应时间要高于读操作。

3) 由于 RAID5 的驱动程序需要生成新的校验冗余信息,所以,大数据写的实际写的数据大小为:

$$S_{req\_w} = S_{req} * \frac{C_{stripe\_len}}{C_{stripe\_len} - 1} \quad (12)$$

式中:S_{req} ——请求大小;

C_{stripe_len} ——RAID5 条纹长度。

以上 3 点在计算服务需求时表现为输入参数的不同,下面分别讨论在读(或大写)请求下各个节点服务需求计算方法。

CPU 节点的服务需求表示为:

$$D_{cpu} = CPU\_Delay \quad (13)$$

缓存节点的服务需求表示为:

$$D_{buf} = ST_{buf}(S_{req}) \quad (14)$$

注意,此时的 S_{req} 为实际请求大小(对于写请求进行相应的转换)。

磁盘节点的服务需求表示为:

$$D_{disk} = ST_{disk}(S_{sub\_req}) * (1 - P_{hit}) \quad (15)$$

式中:S_{sub_req} ——每个磁盘的子请求的大小。

2.2.2 小数据写的服务需求分析

小数据写的过程是如果缓存中没有请求的条纹就读出整个条纹的数据,然后改写需要更新的条纹单元并计算校验单元,再将改写后的数据和校验单元写回到磁盘上,所以与大数据写是不同的。

对于小数据写请求,回写磁盘包含校验单元和修改后的数据单元两个数据块,所以访问概率为 2/ C_{stripe_len},因此小写方式下磁盘服务的需求为:

$$D'_{disk} = (1 - P_{hit}) * ST_{disk}(S_{stripe\_unit}) + \frac{2}{C_{stripe\_len}} * ST_{disk}(S_{req}) \quad (16)$$

式中:S_{stripe_unit} ——整个条纹单元的大小;

C_{stripe_len} ——条纹长度,也就是磁盘个数。

2.3 模型分析

前面给出了 RAID5 在不同负载条件下的服务需求,用 MVA 分析方法可以计算出 RAID5 的吞吐量和平均响应时间。除 CPU 节点外,磁盘节点和缓存节点在系统任务数为 m 时的平均响应时间表示为:

$$R_{buf}(m) = D_{buf} * (1 + Q_{buf}(m - 1)) \quad (17a)$$

$$R_{disk}(m) = D_{disk} * (1 + Q_{disk}(m - 1)) \quad (17b)$$

$$R_{cpu}(m) = D_{cpu} * (1 + Q_{cpu}(m - 1)) \quad (17c)$$

整个系统的平均响应时间为:

$$R_{system}(m) = R_{cpu} + R_{buf}(m) + R_{disk}(m) \quad (18)$$

系统的吞吐量为:

$$X_{system}(m) = \frac{m}{R_{system}(m)} \quad (19)$$

由 little 定律,各节点的队列长度可计算为:

$$Q_{disk}(m) = R_{disk}(m) * T_{system}(m) \quad (20a)$$

$$Q_{buf}(m) = R_{buf}(m) * T_{system}(m) \quad (20b)$$

$$Q_{cpu}(m) = R_{cpu}(m) * T_{system}(m) \quad (20c)$$

以上公式构成了递归关系,当 $m = 1$ 时,显然 $Q_*(m - 1) = Q_*(0) = 0$($*$ 代表磁盘或者缓存节点),算法即可结束。

3 实 验

通过实验比较模型的理论值和实际系统的测试值,实验参数见表 1。

表 1 实验参数表

参数	参数值
CPU_Delay	8 ms
V_{buf}	82 MB/ s
V_{disk}	35 MB/ s
P_{hit}	0.21

测试对象为 linux RAID5 的磁盘阵列,实验数据分为 3 组,分别对应了读请求、大数据写请求和小数据写请求的理论值和实测值。

RAID5 读请求性能曲线如图 2 所示。

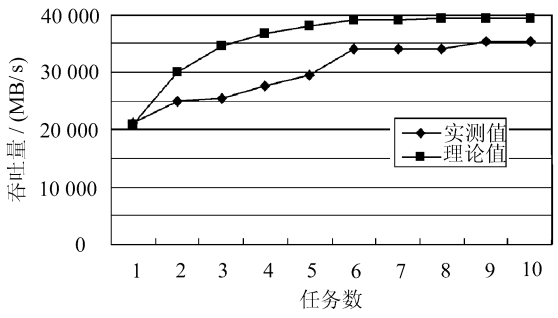

图 2 RAID5 读请求性能曲线

RAID5 大数据写请求性能曲线如图 3 所示。

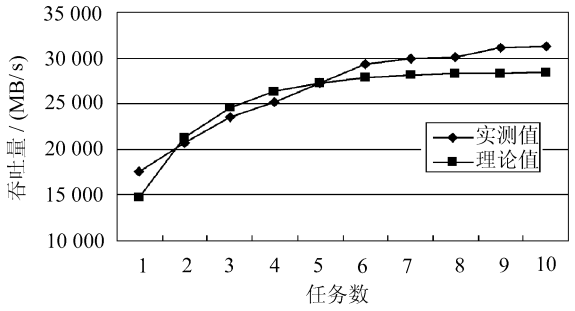

图 3　RAID5 大数据写请求性能曲线

RAID5 小数据写请求性能曲线如图 4 所示。

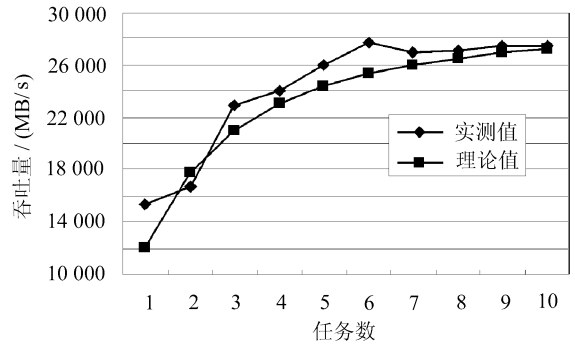

图 4　RAID5 小数据写请求性能曲线

从以上 3 个曲线可以看出,系统的吞吐量在低负载时增加非常明显,随着负载的增加,系统中某些部件已经饱和,也就是总有任务在该节点排队等待,此时系统的吞吐量很难再增加。由于各个服务节点的服务需求是不同的,在系统负载达到极限时,整个系统的吞吐量取决于服务需求最低的那个节点,理论值同样反映了这一趋势,虽然二者有些差距,但表现是一样的。

4　结　语

通过以上的分析和实验得出,多类顾客闭合排队网络模型可以反映真实 RAID 系统的性能状况,采用这种模型对系统进行评价,可以在多种可行性方案中选择性价比高的设计方案;可以发现影响系统性能的瓶颈部件,提出改进方法(可以预测现有系统中那些部件的性能及提高的程度)。另外,它的工作量和费用比其它评价方法要小很多,因此,这种方法具有一定的实用性。

参考文献:

[1] M Uysal, G Alvarez A Merchant. A modular, analytical throughput model for modern disk arrays [C]// Proc. of the 9th Intl. Symp. on Modeling, Analysis and Simulation on Computer and Telecommunications Systems (MASCOTS), 2001:183-192.

[2] Dan Feng, Hong Jiang, Yi-feng Zhu. I/O performance of an RAID10 style parallel file system[C]// Journal Computer. Sci. and Technology Nov, 2004: 965-972.

[3] Anand Kuratti, William H. Sanders performance analysis of RAID5 disk array[C]// Hawaii USA: Proceedings of the IEEE International Computer Performance and Dependability Symposium, 1995: 236-246.

[4] Shenze Chen, Don Towsley. A performance evalution of RAID Architectures[J]. IEEE Transaction on Computers, 1996, 45(6):1116-1130.

[5] R Onvural. Survey of closed queueingnetworks with blocking[J]. ACM Computing Surveys, 1990, 22(2):83-21.

[6] E Varki, A Merchant, J Xu, et al. Issues and challenges in the performance analysis of real disk arrays [J]. IEEE Transactions on Parallel and Distributed Systems, 2004, 15(6):21-50.

[7] M Reiser, S S Lavenberg. Mean-value analysis of closed multichain queuing networks[J]. Journal of the Association for Computing Machinery, 1980, 27(2):313-322.

[8] 谢斌,蒋宁,姜涛. 某型航空发动机温度限制系统检测仪的硬件设计[J]. 长春工业大学报:自然科学版, 2010, 31(2):171-175.

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
40. 基于 CPCI 总线的红外实时信号处理系统
41. 硬件实现 RAID 与软件实现 RAID 的比较
42. 基于 PCI Express 总线系统的热插拔设计
43. 基于 RAID5 的磁盘阵列 Cache 的研究与实现
44. 基于 PCI 总线的 MPEG2 码流播放卡驱动程序开发
45. 基于磁盘异或引擎的 RAID5 小写性能优化
46. 基于 IEEE1588 的时钟同步技术研究
47. 基于 Davinci 平台的 SD 卡读写优化
48. 基于 PCI 总线的图像处理及传输系统的设计
49. 串口和以太网通信技术在油液在线监测系统中的应用
50. USB30 数据传输协议分析及实现
51. IEEE 1588 协议在工业以太网中的实现
52. 基于 USB30 的设备自定义请求实现方法
53. IEEE1588 协议在网络测控系统中的应用
54. USB30 物理层中弹性缓冲的设计与实现
55. USB30 的高速信息传输瓶颈研究
56. 基于 IPv6 的 UDP 通信的实现
57. 一种基于 IPv6 的流媒体传送方案研究与实现
58. 基于 IPv4-IPv6 双栈的 MODBUS-TCP 协议实现
59. RS485CAN 网关设计与实现
60. MVB 周期信息的实时调度
61. RS485 和 PROFINET 网关设计
62. 基于 IPv6 的 Socket 通信的实现
63. MVB 网络重复器的设计
64. 一种新型 MVB 通信板的探究
65. 具有 MVB 接口的输入输出设备的分析
66. 基于 STM32 的 GSM 模块综合应用
67. 基于 ARM7 的 MVB CAN 网关设计
68. 机车车辆的 MVB CAN 总线网关设计
69. 智能变电站冗余网络中 IEEE1588 协议的应用
70. CAN 总线的浅析 CANopen 协议
71. 基于 CANopen 协议实现多电机系统实时控制
72. 以太网时钟同步协议的研究
73. 基于 CANopen 的列车通信网络实现研究
74. 基于 SJA1000 的 CAN 总线智能控制系统设计
75. 基于 CANopen 的运动控制单元的设计
76. 基于 STM32F107VC 的 IEEE 1588 精密时钟同步分析与实现

77. 分布式控制系统精确时钟同步技术
78. 基于 IEEE 1588 的时钟同步技术在分布式系统中应用
79. 基于 SJA1000 的 CAN 总线通讯模块的实现
80. 嵌入式设备的精确时钟同步技术的研究与实现
81. 基于 SJA1000 的 CAN 网桥设计
82. 基于 CAN 总线分布式温室监控系统的设计与实现
83. 基于 DSP 的 CANopen 通讯协议的实现
84. 基于 PCI9656 控制芯片的高速网卡 DMA 设计
85. 基于以太网及串口的数据采集模块设计
86. MVB1 类设备控制器的 FPGA 设计
87. MVB 接口彩色液晶显示诊断单元的显示应用软件设计
88. IPv6 新型套接字的网络编程剖析
89. 基于规则的 IPv4 源程序到 IPv6 源程序的移植方法
90. MVB 网络接口单元的 SOC 解决方案
91. 基于 IPSec 协议的 IPv6 安全研究
92. 具有 VME 总线的车载安全计算机 MVB 通信板卡
93. SD 卡的传输协议和读写程序
94. 基于 SCTP 的 TLS 应用
95. 基于 IPv6 的静态路由实验设计
96. 基于 MVB 的地铁列车司机显示系统研究
97. 基于参数优化批处理的 TLS 协议
98. SSD 数据结构与算法综述
99. 大容量 NAND Flash 文件系统中的地址映射算法研究
100. 基于 MVB 总线的动车组门控系统的设计与仿真研究
101. 多功能车辆总线 MVB 控制
102. 基于 LabVIEW 的 MVB 和 WTB 帧解码方法
103. 基于双 FPGA 的 MVB 通用接口研制
104. RAID 中 Cache 的设计与实现
105. RAID 及并行预取技术分析
106. RAID 系统中 RAID 级别的具体实现算法
107. 一种基于超大容量 Cache 的 VOD 系统
108. 一种面向视频播放系统的 RAID 并行预取技术及实现

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
23. Vxworks 下的冗余 CAN 通讯模块设计
24. WindML 工业平台下开发 S1d13506 驱动及显示功能的实现
25. WindML 中 Mesa 的应用
26. VxWorks 下图形用户界面开发中双缓冲技术应用
27. VxWorks 上的一种 GUI 系统的设计与实现
28. VxWorks 环境下 socket 的实现
29. VxWorks 的 WindML 图形界面程序的框架分析
30. VxWorks 实时操作系统及其在 PC104 下以太网编程的应用
31. 实时操作系统任务调度策略的研究与设计
32. 军事指挥系统中 VxWorks 下汉字显示技术
33. 基于 VxWorks 实时控制系统中文交互界面开发平台
34. 基于 VxWorks 操作系统的 WindML 图形操控界面实现方法
35. 基于 GPU FPGA 芯片原型的 VxWorks 下驱动软件开发
36. VxWorks 下的多串口卡设计
37. VxWorks 内存管理机制的研究
38. T9 输入法在 Tilcon 下的实现
39. 基于 VxWorks 的 WindML 图形界面开发方法
40. 基于 Tilcon 的 IO 控制板可视化测试软件的设计和实现
41. 基于 VxWorks 的通信服务器实时多任务软件设计
42. 基于 VXWORKS 的 RS485MVB 网关的设计与实现
43. 实时操作系统 VxWorks 在微机保护中的应用
44. 基于 VxWorks 的多任务程序设计及通信管理
45. 基于 Tilcon 的 VxWorks 图形界面开发技术

46. 嵌入式图形系统 Tilcon 及应用研究
47. 基于 VxWorks 的数据采集与重演软件的图形界面的设计与实现
48. 基于嵌入式的 Tilcon 用户图形界面设计与开发
49. 基于 Tilcon 的交互式多页面的设计
50. 基于 Tilcon 的嵌入式系统人机界面开发技术
51. 基于 Tilcon 的指控系统多任务人机交互软件设计
52. 基于 Tilcon 航海标绘台界面设计
53. 基于 Tornado 和 Tilcon 的嵌入式 GIS 图形编辑软件的开发
54. VxWorks 环境下内存文件系统的应用
55. VxWorks 下的多重定时器设计
56. Freescale 的 MPC8641D 的 VxWorks BSP
57. VxWorks 实验五[时间片轮转调度]
58. 解决 VmWare 下下载大型工程.out 出现 WTX Error 0x100de 的问题
59. 基于 VxWorks 系统的 MiniGUI 图形界面开发
60. VxWorks BSP 开发中的 PCI 配置方法
61. VxWorks 在 S3C2410 上的 BSP 设计
62. VxWorks 操作系统中 PCI 总线驱动程序的设计与实现
63. VxWorks 概述
64. 基于 AT91RM9200 的 VxWorks END 网络驱动开发
65. 基于 EBD9200 的 VxWorks BSP 设计和实现
66. 基于 VxWorks 的 BSP 技术分析
67. ARM LPC2210 的 VxWorks BSP 源码
68. 基于 LPC2210 的 VxWorks BSP 移植
69. 基于 VxWorks 平台的 SCTP 协议软件设计实现
70. VxWorks 快速启动的实现方法[上电到应用程序 1 秒]
71. VxWorks 下 SL811HS 的 Host 驱动源码

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
33. LINUX ARM 下的 USB 驱动开发
34. Linux 下基于 I2C 协议的 RTC 驱动开发
35. 嵌入式下 Linux 系统设备驱动程序的开发
36. 基于嵌入式 Linux 的 SD 卡驱动程序的设计与实现
37. Linux 系统中进程调度策略
38. 嵌入式 Linux 实时性方法
39. 基于实时 Linux 计算机联锁系统实时性分析与改进
40. 基于嵌入式 Linux 下的 USB30 驱动程序开发方法研究
41. Android 手机应用开发之音乐资源播放器
42. Linux 下以太网的 IPv6 隧道技术的实现
43. Research and design of mobile learning platform based on Android
44. 基于 linux 和 Qt 的串口通信调试器调的设计及应用
45. 在 Linux 平台上基于 QT 的动态图像采集系统的设计
46. 基于 Android 平台的医护查房系统的研究与设计
47. 基于 Android 平台的软件自动化监控工具的设计开发
48. 基于 Android 的视频软硬解码及渲染的对比研究与实现
49. 基于 Android 移动设备的加速度传感器技术研究
50. 基于 Android 系统振动测试仪研究
51. 基于缓存竞争优化的 Linux 进程调度策略
52. Linux 基于 W83697 和 W83977 的 UART 串口驱动开发文档

53. 基于 AT91RM9200 的嵌入式 Linux 系统的移植与实现
54. 路由信息协议在 Linux 平台上的实现
55. Linux 下 IPv6 高级路由器的实现
56. 基于 Android 平台的嵌入式视频监控系统设计
57. Redhat linux 教程
58. 一种 Linux 平台下校园网服务器集群实现方案
59. 基于知识库的 Unix 主机配置安全审计软件的设计与实现

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
19. 基于 WinCE 的嵌入式图像采集系统设计
20. 基于 ARM 与 WinCE 的掌纹鉴别系统
21. DCOM 协议在网络冗余环境下的应用
22. Windows XP Embedded 在变电站通信管理机中的应用
23. XPE 在多功能显控台上的开发与应用
24. 基于 Windows XP Embedded 的 LKJ2000 仿真系统设计与实现
25. 虚拟仪器的 Windows XP Embedded 操作系统开发
26. 基于 EVC 的嵌入式导航电子地图设计
27. 基于 XPEmbedded 的警务区 SMS 指挥平台的设计与实现
28. 基于 XPE 的数字残币兑换工具开发
29. Windows CENET 下 ADC 驱动开发设计

30. Windows CE 下 USB 设备流驱动开发与设计
31. Windows 驱动程序设计
32. 基于 Windows CE 的 GPS 应用
33. 基于 Windows CE 下大像素图像分块显示算法的研究
34. 基于 Windows CE 的数控软件开发与实现
35. NAND FLASH 在 WINCENET 系统中的应用设计

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
13. 基于 PowerPC 的雷达通用处理机设计
14. PowerPC 平台引导加载程序的移植
15. 基于 PowerPC 嵌入式内核的多串口通信扩展设计
16. 基于 PowerPC 的多网口系统抗干扰设计
17. 基于 MPC860T 与 VxWorks 的图形界面设计
18. 基于 MPC8260 处理器的 PPMC 系统
19. 基于 PowerPC 的控制器研究与设计
20. 基于 PowerPC 的模拟量输入接口扩展
21. 基于 PowerPC 的车载通信系统设计
22. 基于 PowerPC 的嵌入式系统中通用 IO 口的扩展方法
23. 基于 PowerPC440GP 型微控制器的嵌入式系统设计与研究
24. 基于双 PowerPC 7447A 处理器的嵌入式系统硬件设计
25. 基于 PowerPC603e 通用处理模块的设计与实现
26. 嵌入式微机 MPC555 驻留片内监控器的开发与实现
27. 基于 PowerPC 和 DSP 的电能质量在线监测装置的研制
28. 基于 PowerPC 架构多核处理器嵌入式系统硬件设计
29. 基于 PowerPC 的多屏系统设计

30. 基于 PowerPC 的嵌入式 SMP 系统设计
31. 基于 MPC850 的多功能通信管理器
32. 基于 MPC8640D 处理系统的技术研究
33. 基于双核 MPC8641D 处理器的计算机模块设计
34. 基于 MPC8641D 处理器的对称多处理技术研究
35. PowerPC 处理器原理
36. Freescale 的 P1020 参考设计原理图
37. Freescale T1040 参考设计板原理图

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 μC-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
22. ARM CortexM3 处理器故障的分析与处理
23. ARM 微处理器启动和调试浅析
24. 基于 ARM 系统下映像文件的执行与中断运行机制的实现
25. 中断调用方式的 ARM 二次开发接口设计
26. ARM11 嵌入式系统 Linux 下 LCD 的驱动设计
27. Uboot 在 S3C2440 上的移植

28. 基于 ARM11 的嵌入式无线视频终端的设计
29. 基于 S3C6410 的 Uboot 分析与移植
30. 基于 ARM 嵌入式系统的高保真无损音乐播放器设计
31. UBoot 在 Mini6410 上的移植
32. 基于 ARM11 的嵌入式 Linux NAND FLASH 模拟 U 盘挂载分析与实现
33. 基于 ARM11 的电源完整性分析
34. 基于 ARM S3C6410 的 uboot 分析与移植
35. 基于 S5PC100 移动视频监控终端的设计与实现
36. UBoot 在 AT91RM9200 上的移植简析
37. 基于工控级 AT91RM9200 开发板的 UBoot 移植分析
38. 基于 ARM11 和 Zigbee 的人员定位防丢器
39. 基于 NAND FLASH 的嵌入式系统启动速度的研究
40. μCOSⅡ在 ARM7 上的移植
41. 基于 ARM11 的嵌入式视频采集系统设计
42. 基于 ARM11 的视频监控系统设计
43. μC-OSII 在 LPC2210 上的移植研究

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
16. 基于 UEFI 的 Application 和 Driver 的分析与开发
17. 基于 UEFI 的可信 BIOS 研究与实现
18. 基于 UEFI 的国产计算机平台 BIOS 研究
19. 基于 UEFI 的安全模块设计分析
20. 基于 FPGA NiosⅡ的等精度频率计设计

21. 基于 FPGA 的 SOPC 设计
22. 基于 SOPC 基本信号产生器的设计与实现
23. 基于龙芯平台的 PMON 研究与开发
24. 基于 X86 平台的嵌入式 BIOS 可配置设计
25. 基于龙芯 2F 架构的 PMON 分析与优化
26. CPU 与 GPU 之间接口电路的设计与实现
27. 基于龙芯 1A 平台的 PMON 源码编译和启动分析
28. 基于 PC104 工控机的嵌入式直流监控装置的设计
29. GPGPU 技术研究与发展
30. GPU 实现的高速 FIR 数字滤波算法
31. 一种基于 CPUGPU 异构计算的混合编程模型
32. 面向 OpenCL 模型的 GPU 性能优化
33. 基于 GPU 的 FDTD 算法
34. 基于 GPU 的瑕疵检测
35. 基于 GPU 通用计算的分析与研究
36. 面向 OpenCL 架构的 GPGPU 量化性能模型
37. 基于 OpenCL 的图像积分图算法优化研究
38. 基于 OpenCL 的均值平移算法在多个众核平台的性能优化研究
39. 基于 OpenCL 的异构系统并行编程
40. 嵌入式系统中热备份双机切换技术研究
41. EFI-Tiano 环境下的 AES 算法应用模型
42. EFI 及其安全性研究
43. 基于 UEFI Shell 的 PreOS Application 的开发与研究
44. UEFI Bootkit 模型与分析
45. UEFI 计算机系统快速调试方法的实现
46. 基于 EFI 系统的多文件系统解决方案
47. 基于 UEFI 的可信 Tiano 设计与研究

Programming:

1. 计算机软件基础数据结构 - 算法
2. 高级数据结构对算法的优化
3. 零基础学算法
4. Linux 环境下基于 TCP 的 Socket 编程浅析
5. Linux 环境下基于 UDP 的 socket 编程浅析
6. 基于 Socket 的网络编程技术及其实现
7. 数据结构考题 - 第 1 章 绪论
8. 数据结构考题 - 第 2 章 线性表

9. 数据结构考题 - 第 2 章　线性表 - 答案
10. 基于小波变换与偏微分方程的图像分解及边缘检测
11. 基于图像能量的布匹瑕疵检测方法
12. 基于 OpenCL 的拉普拉斯图像增强算法优化研究
13. 异构平台上基于 OpenCL 的 FFT 实现与优化
14. 数据结构考题 - 第 4 章　串
15. 数据结构考题 - 第 4 章 串答案
16. 用 IPv6 编程接口实现有连接通信的方法
17. 一种战棋游戏的 AI 算法设计与实现浅析
18. 基于 TLS 协议的 ECC 扩展研究

FPGA / CPLD:

1. 一种基于并行处理器的快速车道线检测系统及 FPGA 实现
2. 基于 FPGA 和 DSP 的 DBF 实现
3. 高速浮点运算单元的 FPGA 实现
4. DLMS 算法的脉动阵结构设计及 FPGA 实现
5. 一种基于 FPGA 的 3DES 加密算法实现
6. 可编程 FIR 滤波器的 FPGA 实现
7. 基于 FPGA 的 AES 加密算法的高速实现
8. 基于 FPGA 的精确时钟同步方法
9. 应用分布式算法在 FPGA 平台实现 FIR 低通滤波器
10. 流水线技术在用 FPGA 实现高速 DSP 运算中的应用
11. 基于 FPGA 的 CAN 总线通信节点设计
12. 基于 FPGA 的高速时钟数据恢复电路的实现
13. 基于 FPGA 的高阶高速 FIR 滤波器设计与实现
14. 基于 FPGA 高效实现 FIR 滤波器的研究
15. FPGA 的 VHDL 设计策略
16. 用 FPGA 实现串口通信的设计
17. GPIB 接口的 FPGA 实现
18. 一种基于 FPGA 的 FFT 阵列处理器
19. 基于 FPGA 的 FFT 信号处理器的硬件实现
20. CPLD 在 CAN 通讯卡中的应用
21. 用 CPLD 实现同步串口与异步串口的转换
22. CPLD 在 LED 网络控制器中的应用
23. 基于 CPLD 的双口 RAM 设计与应用
24. CPLD 在有源电力滤波器中的应用