

摘要¹

目前,数字化和网络化已成为电视台影视播放设备发展的必然趋势。播出系统作为电视台节目面向观众的最后一道关口,直接关系到节目播出的图像质量和可靠性,它是电视台数字化的核心,已成为当前计算机应用技术研究的热点。为了保证播出图像的质量,从数字播放系统的体系结构、关键算法、服务响应时间等方面,进行深入研究,重点是研究播放系统的可靠性、帧精确性及实时性等关键问题。

组建电视台数字视频播放系统最重要的内容是确立其体系结构,而存储播放子系统是整个系统的核心,它的体系结构设计至关重要。为满足高质量存储播放子系统的高可靠性、高数传率、海量存储容量和可扩展性的要求,我们在理论指导下,对存储播放子系统进行了性能平衡设计,并建立了它的可靠性模型,计算出它的可用度,其性能完全满足电视台停播率的要求。该设计与可靠性模型对设计存储播放系统和分析系统性能均具有指导意义。

播出质量是数字视频播放系统一项最重要的技术要求。为了实现无压缩播出和任何制式下的帧精确性。在分析了无压缩数字播放系统的播出工作特性的基础上,提出了一种多线程并行直接 I/O(MPPDIO)存取算法,并建立了计算共享缓冲区大小的模型。实验证明:MPPDIO 存取算法完全能实现无压缩媒体流的不掉帧、无黑场的关键性能,既能保证播放质量,又提高了磁盘 I/O 通道的利用率,降低了 CPU 负载,传统的 PRE_CACHE 顺序存取算法则不能实现系统的这些关键性能。MPPDIO 存取算法也可以在未来用于高清晰度电视的数字播放系统中。

MPEG2 具有压缩效率高的优点,它对电视台业务的大部分应用场合是适用的。我们详细讨论了实现 MPEG2 的数字视频播放的关键技术,提出并实现了冗余双播放缓冲区多线程调度算法,解决了压缩数字播放系统的帧精确和节目间平滑转换的问题。指出:该算法既不影响播放帧的及时处理,又不使预解码数据量受播放缓冲区限制;与现有的压缩播放系统靠双解码卡以硬件实现帧精确与平滑转换相比,具有巧妙性,成本低和可推广性。

电视台需要对某些直播节目进行实时监控,要在对节目进行审阅后,延迟一段时间再播出,以满足电视台的特殊要求,因此有必要研究延迟播放技术。目前的延迟播放系统大多由大规模集成电路实现,价格昂贵,国内也较罕见,针对这种情况,我们在分析比较了四种不同处理方案的特点的基础上,另辟蹊径,提出了利用磁盘缓存共

¹ 本文受国家自然科学基金资助,批准号 69973017、60273074。

华中科技大学博士学位论文

享压缩视频文件来实现延迟播放的新方案，并创建了一种最优算法。使用此算法可以有效地实现延迟播放功能，且不存在延迟时间限制的瓶颈，用软件实现，所需代价最小。

播放系统在播放节目文件时存在实时格式转换的需求，以往的格式转换要么通过底层硬件电路实现，要么通过工具软件实现，两者都不能满足系统的需求。因此我们提出了一种多线程流水调度算法，根据 VOB 文件特点，对文件格式转换的解复用、编码、复用等环节进行优化，较好地实现了 DVD 文件、VCD 文件转换到 MPEG2 文件的混合无掉帧实时播放。实验结果表明有以下三点好处：针对 VOB 文件特点的解复用器比通用解复用器提取音视频流的耗时要小得多；采用编码优化算法使得编码时间远小于播放时间及未优化算法的编码时间；DAT 文件及 VOB 文件到 MPEG2 文件的格式转换及混合播放，可达到帧精确。

在节目的播放过程中存在临时更改节目单的情况。为实现播出时能实时修改节目单，有必要分析播放系统服务响应时间。在分析影响播放系统服务响应时间的因素之后，建立了计算其服务响应时间的数学模型，得出节目单条数、磁盘阵列的磁盘数与分块大小对系统服务响应时间的影响程度。并针对影响系统服务响应时间的因素，提出了四条相应的改进措施，使系统服务响应时间达到可能的最小值。

关键词：数字播放系统，体系结构，帧精确，可靠性，算法，响应时间

ABSTRACT¹

Current, broadcasting movies and TVs in the television station has inevitably developed into digital and network age. Broadcasting systems are the last part before television station program faces to the audience, relating to directly picture quantity and reliability of the program broadcasting. It is a core that television station's digital turn, having become a highlight of research on computer application. For the sake of guarantee the picture quality of program broadcasting, the key problems such as reliability, frame accurateness and real-time performance are studied from the hardware architecture, the key algorithm and the response time etc. aspect of digital broadcasting systems.

The architecture of digital broadcasting systems is the most important issue of building the digital broadcasting systems. Its core is the storage and broadcasting sub-systems, and the design of architecture of the storage and broadcasting sub-system is a key. For satisfy the high reliability, the high data transfer rate, the mass storage capacity and scalability of the high quality storage and broadcasting sub-system, its performance balance design has been done and its reliability model is built based on the theory. Its availability can completely meet the requirement of TV Station. The design method and the reliability model can serve as a guide to the design and performance analysis of the storage and broadcasting system.

The broadcasting quality is one of the most important technology issues in the digital broadcasting systems. To realize frame precision when broadcast, an algorithm called MPPDIO (multi-thread parallel pre-fetch Direct IO) is proposed after the characters of the broadcasting part in uncompressed digital broadcasting system are analyzed. The size of the sharing buffer between multi-threads is modeled and calculated.

Experiment results indicate that the traditional pre-cache sequence algorithm can't meet the need of performance, and the MPPDIO algorithm not only can perfectly satisfy the performance requirements but also can improve the utilization of disk I/O channel and reduce the load of CPU. The MPPDIO algorithm can use in the digital broadcasting systems on HDTV later.

The MPEG2 has the advantage of high efficiency in compression; it is suitable for lot of applications of the television station business. The key techniques of digital broadcasting based on MPEG2 are discussed in detail. A redundancy playing buffer multi-threading scheme is put forward, and it realizes the frame precision and smooth switch between programs. The scheme doesn't affect both processing playing frame and the pre-decoding

¹ This research work was supported by the National Science foundation of china under grant 69973017 and 60273074

on frame. It is smarter and costs lesser compared with the current method to realized the frame precision and smooth switch between programs by hardware.

Some live program must be monitored and delay played to satisfy the special real demand in television station. It's necessary to research the delay playback technology. The current delay playback systems are implemented by LSI and cost huge. We put ingeniously forward a scheme realized by share the compression file in disk buffer, after analyzing and comparing the performance of four kinds different handling methods. This method can realize availablely the function of the delay playback, has no bottleneck in delay time restriction, and costs least.

The digital broadcasting systems demand the real time conversion of file format when broadcasting programs. The past conversion of file format is implemented by hardware or software tool, which can't meet the requirement of the system. A kind of real time conversion method fulfilled by the multithread pipeline scheme algorithm is proposed. According to format characteristics of VOB file in the DVD, each link in conversion the file format—demultiplex, code and multiplex, is improved. Experiment results indicate that the scheme algorithm can convert file formats of VCD and DVD, and can real time broadcast the mixture of DVD, MPEG2 file and VCD, and is more efficient than that of before improve.

The accidentally change of playbill will exist in the process of program broadcasting. To succeed in real time modifying the playbill, it is necessary to study the response time of the digital broadcasting system service. After the influence factors in the digital broadcasting system service are analyzed, the model on the response time of the system service is established. The degree of the influence factors such as the length of playbill, and the number of disks and the number of blocks in RAID affecting the response time of the system service is gotten. And Aiming at the influence factors, four improvements measure are proposed to minimize the response time of system service.

Keywords: Digital Broadcasting System, Architecture, Frame Precision, Reliability, Algorithm, Response Time

独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：王烁

日期：2004年2月3日

学位论文授权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 ， 在 _____ 年解密后适用本授权书。

本论文属于
不保密 。

(请在以上方框内打“√”)

学位论文作者签名：

日期： 年 月 日

指导教师签名：

日期：2004年2月3日

1 电视台视频播放系统概述

1.1 数字化、网络化是电视台发展的必然趋势

当今世界处于飞速发展的信息时代，随着科学技术的日新月异，21世纪的广播电视技术如何发展已经成为业内人士共同关心的问题。由于先进的计算机技术、电子集成技术、通讯技术迅速向电视领域渗透，电视业正迎来一场革命性的变化，这种变化概括地说主要体现在两方面，即电视的数字化和网络化^[1]。

数字电视将传统的模拟电视信号经过抽样、量化和编码成二进制数字信号，然后进行各种功能的处理、传输、存储和记录，用计算机进行处理、监测和控制。数字电视设备不仅可以获得比原有模拟设备更高的技术性能，而且还具有模拟技术达不到的新功能，令电视技术进入崭新的时代。它可以使人们看到、听到接近演播室质量的电视图像和CD质量的声音，并且使节目内容更丰富多彩。不仅为人们提供高质量的视听享受，还提供数据、文字、图形和静止图像等信息。

从技术角度来讲，数字电视技术具有的优点主要体现在以下几个方面：

(1) 数字信号在传输过程中通过再生技术和纠错编解码技术使噪声不会逐步积累，基本不产生新的噪声，能保持信噪比基本不变，且收端图像质量基本保持与发端一致，适合多环节、长距离传输。

(2) 数字电视设备输出信号稳定可靠，能够避免在模拟系统中非线性失真对图像的影响，消除了微分增益和微分相位失真引起的图像畸变。

(3) 数字电视易于实现信号存储和数字处理，如帧存储器、数字特技机、数字时基校正器，以新的特技形式增强了屏幕艺术效果。数字电视信号具有极强的可复制性，用在节目制作上不致降低图像质量。

(4) 数字技术与计算机配合，可以实现电视设备的自动控制并使操作容易。

(5) 采用时分多路数字技术，可以实现信道多路复用。

(6) 利用数字压缩技术使占用的传输信道带宽比模拟电视明显减少，通常为模拟电视的1/4左右，甚至更小，这样可以合理利用各种类型的频谱资源。对地面广播来说，数字电视可以启用模拟电视的“禁用频道”，也可以采用“单频网络”技术进行节目的大面积有效覆盖，如用一个数字电视频道完成一套电视节目的全国覆盖。利用数字压缩技术，在一个卫星频道上转发多套电视节目，可达到节省卫星信道的目的，从而提高信道传输容量。

(7) 采用数字编码方法，便于实现加扰和解扰技术，使收费电视在实际中得以

应用。

(8) 数字电视信号具有可扩展性、可分级性和互操作性, 便于在各类通信信道, 特别是异步转移模式 ATM (Asynchronous Transfer Mode) 网络中传输。随着电视数字设备向多媒体方向发展, 可形成开放性的电视多媒体网络, 方便与各类计算机网络联通, 达到信息共享的目的。

正是由于数字电视技术具有的强大优越性, 以及存储技术、网络技术、压缩技术、影视高新技术的飞速发展, 近年来电视台数字化的进程明显加快, 模拟信号向数字信号过渡全面展开^[2-8]。这不仅体现在单一数字设备的改进和创新上(出现了数字摄像机、录像机、特技机、切换台等一大批运用数字技术的设备), 而且大规模的整体性的数字系统也日趋完善, 出现了全数字电视演播室、全数字电视转播车、数字压缩的卫星新闻采集转播车 DSNG (Digital Satellite Newsgathering) 等; 这种数字化还正向全电视行业的各环节——制作、传输、发射和接收方面纵深发展, 集成多种技术的结果产生了更为新颖的电视数字化系统、数字地面广播 DVB (Digital Video Broadcasting)、数字卫星直播 DTH (Direct to Home) 等。

当从更大范围来考察这场由数字化所引发的电视领域的革命时, 还会看到由于计算机、多媒体技术与数字电视技术相结合而产生了非线性编辑系统和虚拟演播室系统。非线性编辑系统是一种以计算机为操作平台的电视节目后期制作系统。它包括了数字压缩技术、数字存储技术、数字图像处理技术、计算机图形技术、网络技术等相关技术, 把数字化、多媒体化、交互性和网络化带入视频编辑之中, 给电视制作带来重大变革, 使电视工作者过去难以做到的许多美好创意得以实现, 它把人们引入了一个“只有想不到, 没有做不到”的崭新创作空间。虚拟演播室技术, 将计算机与电视技术结合起来, 能提供逼真的虚拟空间, 即将计算机制作出来的背景图像与演播室拍摄的人物完美地结合在一起, 从而制作出传统设备无法表现的节目, 将人物置于千变万化的虚拟三维空间之中, 丰富了电视屏幕, 这是对传统演播室技术的重大挑战。

电视多媒体非线性技术的出现, 为节目制作、播出网络化提供了条件。为了提高工作效率, 达到资源共享, 可以将以单机工作方式的多台非线性编辑系统、虚拟演播室系统、动画工作站、音频工作站等各类以计算机为操作平台的系统组成网络, 成为电视台内部的一个小局域网子系统。还可以将电视台内的各个制作、播出及管理等于子系统, 采用可传输多种信息的 ATM 网或宽带以太网拓扑成一个大的局域网, 建立全台的宽带视频综合业务网络, 实现计算机设备、多媒体设备的互联和信息交流共享, 并支持各虚拟网络之间的信息交换, 使台内与台外的网络之间互相联通, 形成大的广域网。

而且随着电视台的全面数字化,将加强电视最终与通信和计算机一体化,形成一种广义概念上的网络。原来是不同媒体的电视、通信和计算机在全部数字化后,在数字域中均以“0、1”为基本单元,形成“0”“1”符号的比特流。光纤、卫星、数字微波等这些通信传输手段也成了电视的传输手段。电视、通信、计算机这些迄今相互分离的技术将融为一体,使这些业务互相渗透、融合、会聚。这种发展趋势,使电视网络不仅可单向传送节目,还提供多种新形式、程度不同的交互式服务,如视频点播VOD (Video on Demand)、远程教学、电视会议、与因特网联网等。随着电视网络化向深度发展,电视行业最终将会是完全交互式的多媒体^[9-11],但这个过程是逐渐过渡的,从今天的模拟——数字播出——加入交互式数据服务——完全的交互式多媒体环境。这个过程的原动力来自电视台技术的数字化、网络化。

1.2 电视台数字化的国内外现状

从1994年开播卫星数字电视到1998年底开播地面数字电视广播,数字电视实现了全面启动。到1999年底全球约有2500万台卫星数字电视机顶盒、700万台有线网数字机顶盒、40万台地面数字电视机顶盒和接收机,这些主要分布于欧洲、美国和日本^[12]。

美国

1996年12月美国联邦通信委员会FCC (The Federal Communications Commission) 确定ATSC (The Advanced Television Systems Committee) 作为美国数字电视地面广播标准。1997年4月FCC颁布实施数字电视地面广播的时间表和电视频道分配方案,引入数字电视服务的标准,并决定在98年秋季开始启动此服务,为此美国还成立了一个陆地数字电视服务的顾问委员会^[13]。1999年5月,美国通信委员会下达命令,要求美国所有电视台必须在2006年12月31日前完成从模拟到数字化的过渡^[14]。1999年11月1日,ABC (Australian Broadcasting Corporation)、BS、NBC、FOX 全美最大的四家电视网络公司在美国电视用户市场排名前30位的城市的120多家附属电视台,全部播出、发射数字电视DTV (Digital Television) 节目,覆盖全美53%的电视用户。在2002年4月7日下午1:00—4:00,美国拉斯维加斯会议中心N255/N257举行会议,解决电视台数字化过渡的难题,弄清什么是数字化过渡的关键部分以及怎样实施一些具体步骤^[15]。到2003年5月1日全部非商业电视台播出发射DTV节目,到2006年,全部停止NTSC (National Television System Committee) 制式的模拟电视广播,收回全部NTSC占用的频率资源。

欧洲

整个欧洲确定采用以DVB为框架的数字电视标准,发展的侧重点放在标准清晰度数字电视。向用户提供增强型数字电视业务,推动代价小,商业化运作较为成功。1998年秋季欧洲已开始数字电视服务^[16]。在1998年8月24日晚EBU-Eurovision Network开始发射100%数字格式MPEG 4:2:2 的媒体流^[17]。而加拿大联邦政府也要求多伦多和蒙特利尔等大型电视台在99年实现数字发射,全国的陆地广播公司应在2004年实现数字化,技术标准将采用美国通信委员会在97年制定的标准。联邦政府将每年投资伍千万美元用于数字化改造^[15]。到2000年10月为止,英国市场共销售DVB机顶盒896000台,集成数字ITV (Integrated Television)电视机30000台。BskyB (Europe's largest and most successful pay-TV company)有近500万数字电视的英国用户,并正在开拓德国市场。德国电信的有线电视网有80条数字频道。法国电信收购了荷兰的CASEMA和英国的NTL有线电视网,开展CABLE数据和数字电视业务,以卫星和有线电视网为主要手段的数字电视正在抢占更大的份额。

日本

继欧美之后,日本在欧洲DVB技术的基础上研制出世界上第三个拥有独立知识产权的数字电视地面广播标准,但是日本的实施步伐较慢,2000年7月22日,日本也在冲绳岛试验了高清晰度数字电视HDTV (High Definition Television)数字广播,标志着日本数字电视的划时代开端。日本将在2000年12月发射7个频道的数字电视,开展数字广播电视^[18]。正式开播时间定于2003年,服务地区仅限于东京大阪和名古屋三市。

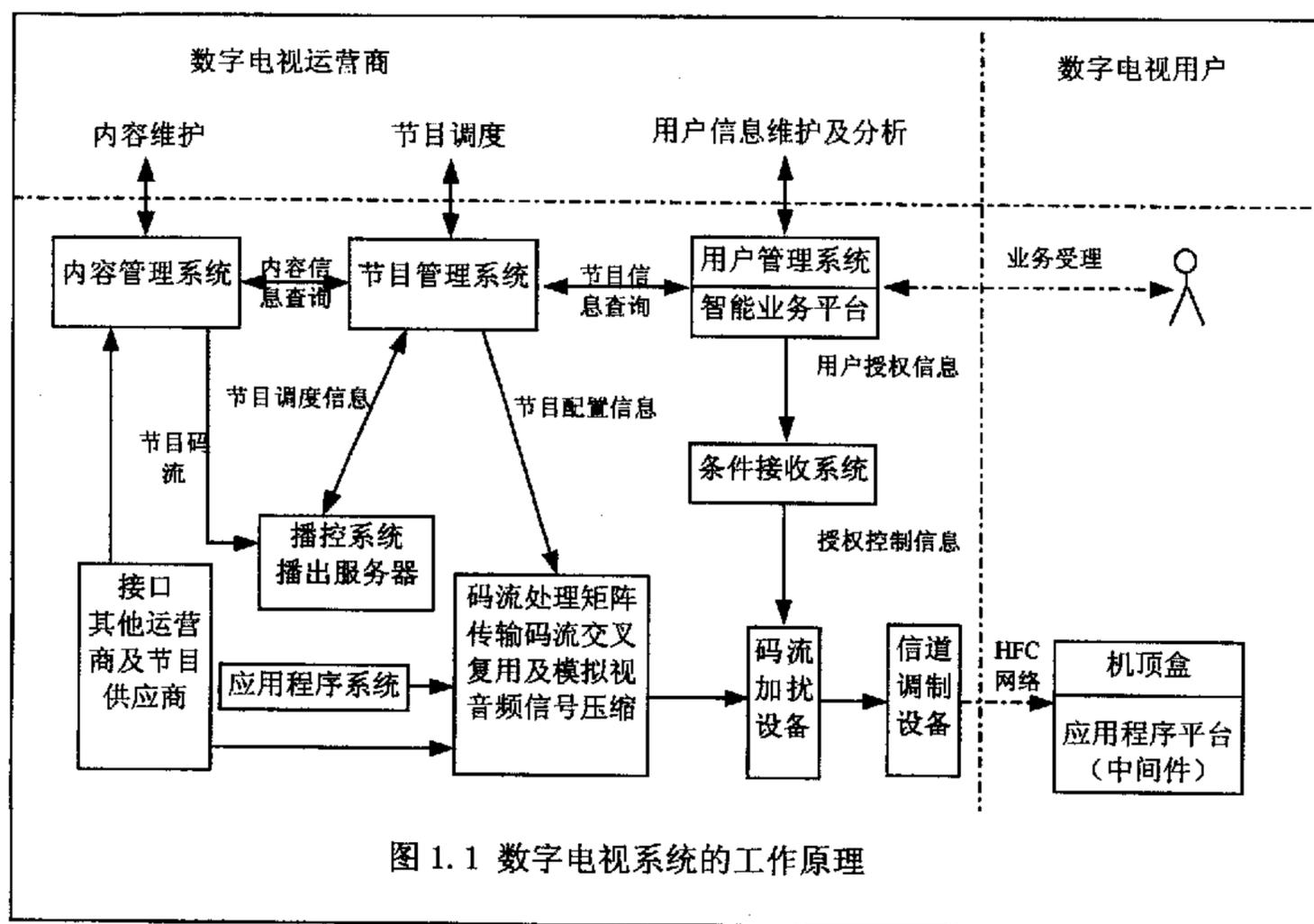
可以说世界各国都在向电视台数字化迈进,数字化过渡进行得如火如荼。我国在多年跟踪研究国际数字电视技术发展的基础上,由国家计委牵头,于1999年确立了《国家数字电视研究开发及产业化专项》,成立了国家数字电视研究开发和产业化领导小组。由国家计委主任曾培炎任组长。2002年我国正式播出有线数字电视,2003年全面推进卫星数字电视,全面推广有线数字电视。2005年全国省级以上电视台基本实现采、编、播全数字化,全国包括各省的上星节目停止模拟传输,全部采用卫星数字电视。有线数字电视接收用户超过3000万户,全国正式播出地面数字电视。

一套完整的数字电视系统由三个部分组成:数字电视前端系统、传输网络 and 用户终端系统^[12]。

(1) 数字前端系统

数字电视前端系统通常划分为信源处理、信息处理和传输处理三部分,从功能上又划分为:用户管理子系统、条件接收CA (Conditional Access)子系统、DVB-C(Digital Video Broadcasting - Cable)前端设备子系统、数字节目的存储和编播子系统、数据广

播和中间件子系统等。原理参见图1.1。



信源

将本地节目编码压缩成DVB-MPEG2 (Motion Pictures Experts Group) 数字信号。适配接收来自其它网络卫星和地面的DVB-MPEG2 信号,送至复用器复用或送至节目库存储、编辑。将IP 数据和其它用于数据广播和交互服务的服务数据转换成DVB 方式,与其它信号一起传送到用户端。准视频点播NVOD (Near Video on Demand) 和视频点播VOD系统可直接连接至复用器,将用户点播的节目或信息送到用户端,对存储的数字节目可进行编辑、管理。

信息处理单元

信息处理单元由节目调度系统、用户管理系统、多路复用器、条件接收系统等组成。节目调度系统是一套完整的系统管理和应用系统平台。多路复用器是系统的核心,相当于交通枢纽中心和货物处理中心,在节目调度系统的控制下,负责将来自各个地方的内容重新进行分配复用,发送到不同的频道,完成内容分配调度。CA系统,根据服务方式和用户需求,通过网络管理系统的控制,将加密信号通过加扰器和复用器注入到不同的节目内容中,使节目内容在不同的时间段和针对不同的用户层进行加密,采用或改变条件接收提供特殊服务如适应当地频率规划的网络信息表、电子节目

表EPG (Electronic program guide) 的产生和插入。

传输处理单元

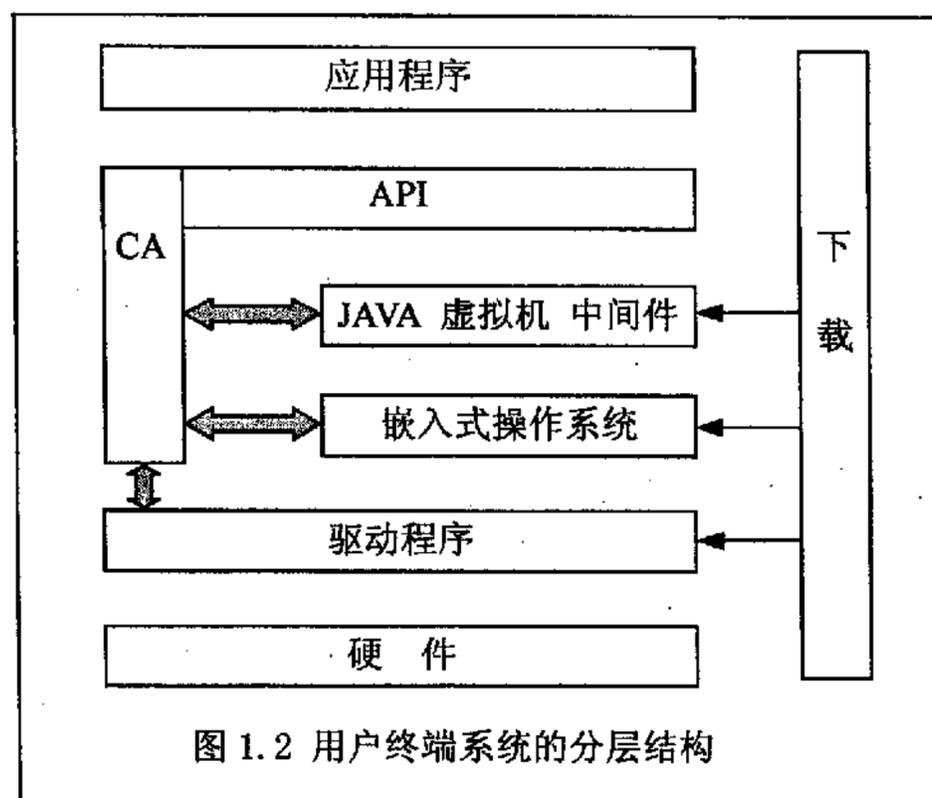
主要是信道编码针对不同的传输媒体, 按照不同的调制方式, 对信号进行调制, 如卫星传输采用QPSK(quadrature phase shift keying) 调制, 有线电视网络采用QAM (Quadrature Amplitude Modulation) 调制, 地面开路传输采用COFDM (Coded Orthogonal Frequency Division Multiplexing) 调制。

(2) 传输网络

数字电视网络的传输手段包括卫星、有线电视网、地面发射、MMDS (Multi-channel Multipoint Distribution Service) 和 XDSL(x-Digital Subscriber Line) 等, 回传通道可选用HFC(Hybrid Fiber Coax)回传通道、PSTN(Public Switched Telephone Network)和 ISDN(Integrated Services Digital Network)网络。

(3) 用户终端系统

用户终端系统就是通常所说的机顶盒STB (Set-top Boxes)。一个完整的数字机顶盒是由硬件平台和软件系统组成的。分为四层, 从底向上依次为硬件、底层软件、中间件、应用软件, 另外还有一个CA 部分, 如图1.2所示。



硬件

硬件层提供机顶盒硬件平台, 主要由数字电视广播接收前端、MPEG 解码、视音频和图形处理、CPU 存储器和各种接口电路组成。数字电视广播接收前端包括调谐器和QAM 解调器, 该部分可从射频信号中解调出MPEG 传输流; MPEG 解码部分包括解复用、解扰引擎和MPEG解压缩, 其输出为MPEG 视音频基本流和数据净荷。

视音频和图形处理部分完成视音频模拟编码和图形处理功能。CPU 与存储器模块用于存储和运行软件系统,并对各模块进行控制;接口电路提供丰富的外部接口,包括通用串行接口USB (Universal Serial Bus)、高速串行接口1394、以太网接口、RS232和视音频接口等。

底层软件

底层软件提供操作系统内核和各种硬件驱动程序。机顶盒操作系统不是非常庞大,但却要求可以在实时环境中工作,并可在较小的内存空间中运行,这种操作系统称为实时操作系统。目前流行的实时操作系统有Wind River System 公司的VxWorks, Integrated Systems Incorporated公司的pSOS、Windows CE 以及专为机顶盒开发的PowerTV 等。这些操作系统各有所长,在机顶盒领域都有应用。其中VxWorks、pSOS等都是通用实时操作系统,在其它嵌入式系统中也有广泛应用。

中间件

中间件将应用软件和依赖于硬件的底层软件分开,使应用不依赖于具体的硬件平台,通常由各种虚拟机构成。目前应用于机顶盒的商用中间件产品有Opentv 的EN2、Liberate 的TV Navigator for DTV、Microsoft 的MSTV、Canel+ 的Mediahighway 等,彼此不兼容。应用软件包括本机存储的应用和可下载的应用。

CA 部分

它是机顶盒为了再生出解扰随机序列,获取相关的条件接收控制信息的模块。

随着数字电视技术的发展,数字电视广播将改变我们的生活习惯,让“用户足不出户,晓天下大事”。通过电视机、互联网终端等推进这种新的服务模式,数字电视将展示一个全新的服务行业。

而现在在我国,相当一部分电视台从复合模拟系统局部向复合分量过渡的过程还没有结束,而向数字化过渡的问题又摆在了面前。而电视台从模拟系统向数字化系统过渡,如果经济条件许可,一次性投入较多资金,整体构建大型数字系统,有利于选择最佳技术方案,但是国内大部分电视台因经济条件并不宽裕,目前普遍采取逐步过渡的方式,其过渡阶段如下^[4]:

(1) 单一数字设备在模拟系统中应用。局部使用单一数字设备,信号接口采用数字设备的模拟接口,如使用数字摄像机、数字录像机、数字特技台等。目前国内部分电视台仍处于这一过渡阶段。

(2) 单一数字系统在模拟环境中应用构成小规模数字系统。如数字录像机+特技台构成的后期制作系统,系统对外接口仍采用模拟方式,以满足模拟播放系统的要求。这是过渡的第二阶段。

(3) 模拟系统与数字系统混合使用。应用大规模数字电视系统,如数字转播车、

数字演播室、各系统之间信号交换数字与模拟共存，模拟系统仍发挥重要作用。这是第三阶段，国内大部分电视台正处于这一阶段。

(4) 数字播放：数字电视台建设的中心。数字播放系统的建设是数字电视台实现的中心和过渡的分水岭。它有两个意义，其一，意味着电视中心的输出信号实现数字化。其二，它要求电视台内各系统信号格式向它所实现的数字化靠拢。

(5) 全面数字采录和制作的实现。经过这一阶段，电视台全盘数字化才能实现。

(6) 电视台全盘数字化。实现全台电视节目采集、编辑、制作、播出、传输和存档的数字化、网络化。

数字化进程在国内刚处于启动阶段，还无多少经验可以借鉴^[19]。因而从整体上如何去把握它就变得更加困难。尽管如此，现在国内各电视台都采取较为积极的态度来推进从模拟系统到数字系统的演变，应尽可能地寻找采用新技术的最佳切入点，充分利用新技术带给他们的实惠，确定最适合其电视台发展的数字化过渡模式。

1997年底，厦门电视台在制定数字化过渡方案时，根据其自身的经济实力和设备情况，作了非常慎重和周密的考虑，制定了“先从前期到后期、再到播出逐步过渡，最终完全实现数字化”的分步实施方案^[20]。首先迈出的第一步是：于1998年1月购买部分SONY Betacam—Sx 摄像机、数模兼容的放像机以及装备数字演播室系统，收到了良好的效果。此后又先后增加了一些 Betacam—Sx 格式的摄像机、编辑机、现场制作系统及便携式编辑机。厦门电视台现有模拟的设备不多，线性数字设备也正好满足目前节目的需求，下一步将根据数字技术发展的趋势，直接采用适合厦门电视台过渡模式的、相对成熟的新技术，循序渐进地从模拟向数字平稳过渡。未来厦门电视台台内的运作模式是：从节目库到节目制作，直至节目播出全面实现数字化、网络化，在网络中，节目以MPEG—2 压缩数据流传送，实现流水线制作工艺流程，实现制播一体、播出无带化。

福建电视台也基于其自身的经济实力和拥有八个频道的情况。将尽可能延长模拟设备的使用寿命，集中重点保证一套(综合频道)和二套(卫星频道)在“十五”期间实现数字化。其它频道以需求为主逐步实现数字化^[21]。福建电视台未来的数字化设想及其相关思路是：第一步，实现以多通道硬盘为主、盘带结合的播出系统改造。第二步，估计2006年前后，建成八频道合一的高度自动化，网络化的全数字播出系统。

安徽电视台认为，目前各级彩电中心都面临要求增加节目播出时间和套数的压力。电视台作为党和政府的宣传机关，它的新闻节目甚至是一般专题节目的时效性要求也愈来愈高。为此台内建立起一套快速节目审改系统是有很高实用价值的^[22]。一旦引入多媒体计算机技术及其网络后，就可以顺理成章地做到编辑和审看直接调用节目了。

湖南电视台生活频道与创智公司合作共同开发集前期制作、后期编辑、硬盘播出、存储检索和互联网发布于一体的“A11—in-One”全程网^[23]。2000年12月27日，全程

网的节目制作、后期编辑部分完成并投入运行。2001 年春天,“A11—in-0ne”全程网将全部完成。A11—in-0ne 网络结构如图 1.3:

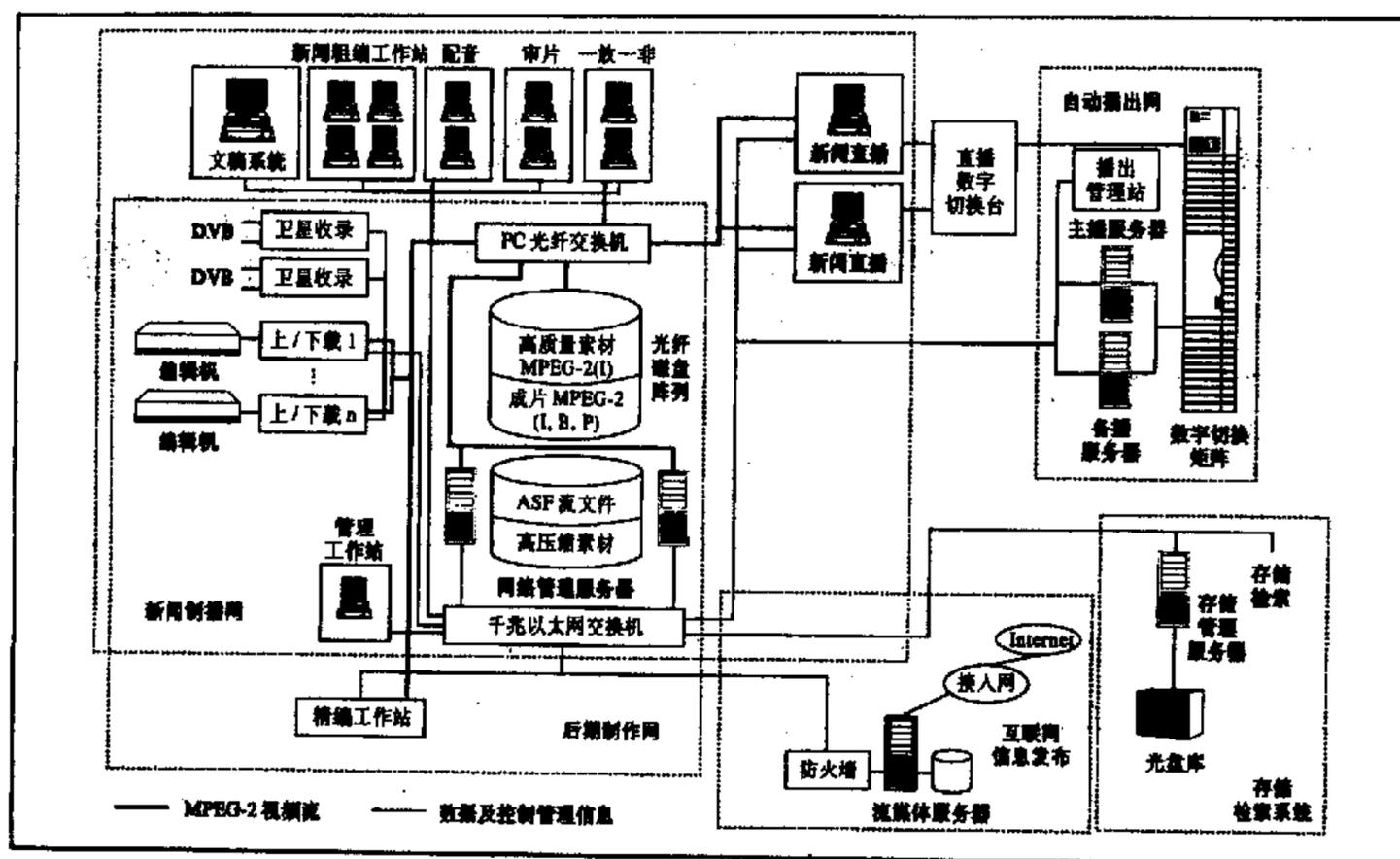


图 1.3 All-in-One 电视台全程网结构

2000 年 8 月,湖南经济电视台与北大方正电子有限公司合作投资了一个大型项目,通过采用视频服务器,构建采编播一体化网络,将网络与视频技术结合起来,解决了实际工作中的许多问题^[24]。此系统采用了 Pinnacle MSS 视频服务器系统,1 台 MSS1600、1 台 MSS700,其配置均为 2 进 5 出,100 小时在线存储,FC 光纤通道进行双机实时备份,构成典型的双机镜像结构(如图 1.4)。

珠海电视台从 2001 年初开始进行调研,经过反复论证,决定构建以视频服务器 Pinnacle 公司的 MSS1600 为中心的网络化多频道数字播放系统^[25]。于 2001 年 12 月 21 日迁入新的电视中心,新的播出系统全面启用。该系统共使用 8 个 MPEG 编码器,16 个 MPEG 解码器。每台 MSS1600 使用 9 块 90G 硬盘,采用 RAID3 技术组成磁盘阵列,编码率采用 10Mb/s,存储时间为 100 小时。两台 MSS1600 之间的光纤网连接,任何一个通道上载素材,都可以通过 FC 在另一台服务器产生镜像备份。同时该系统具有良好的网络信号传送和网络控制功能、兼容 SDTV(Standard Definition Television)和 HDTV,为电视技术的进一步发展打下良好的基础。

综上所述,到目前为止,我国电视台还处于模拟信号的总体技术环境中,数字电视设备都只是在模拟系统中作为单件(黑盒子)设备使用。除了少数几个台已经形成局部的数字化制作岛以外,国内大部分电视台大都处于数字化的较低阶段,都只是构想

了如何数字化，而数字电视台建设的先驱—数字播放系统都还在设想之中；而国外发达国家已比我们领先一步，且它们的电视台数字化也正在进行之中，因此我们要急追猛赶，加快我们的电视台数字化进程。

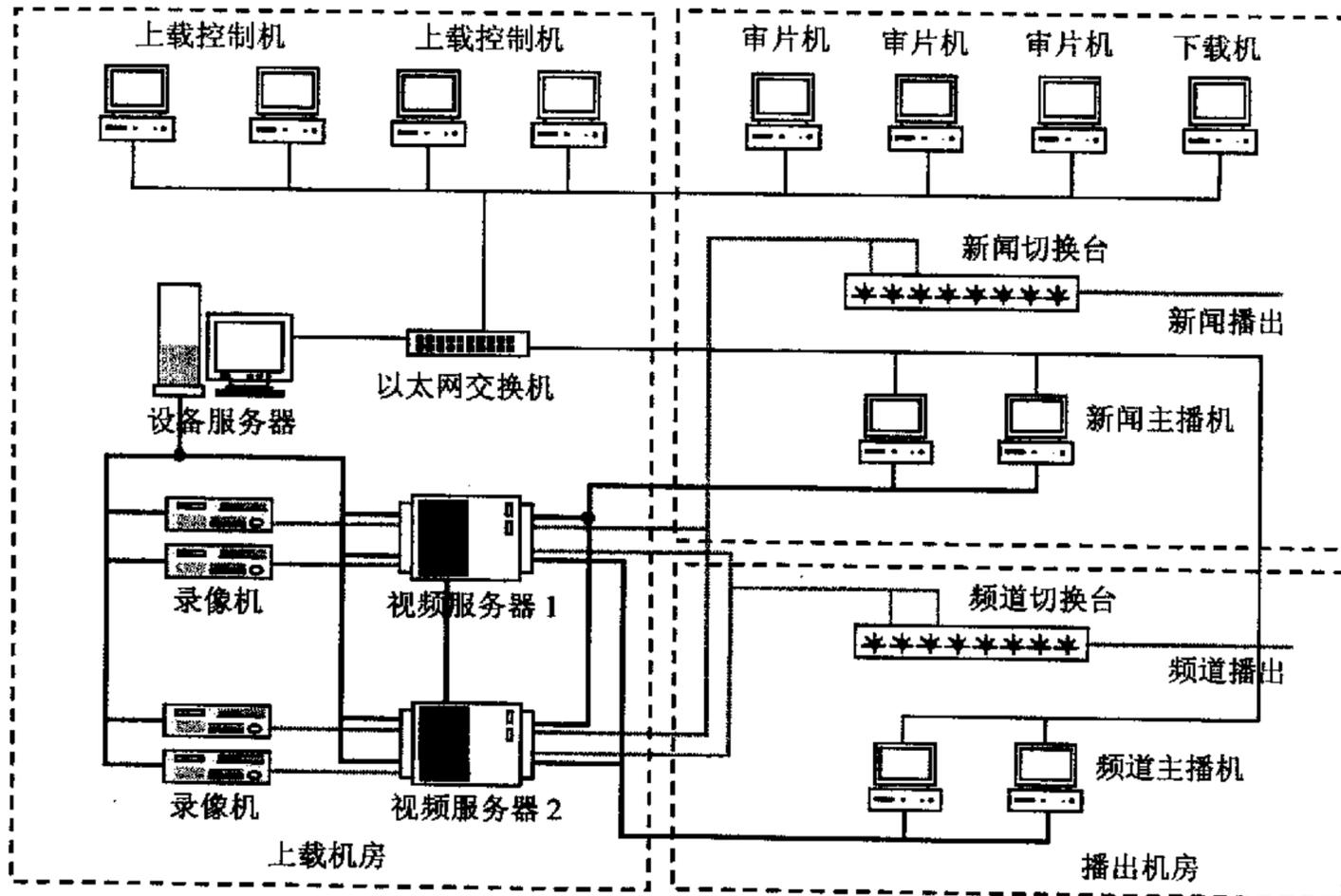


图 1.4 播控系统网络结构

1.3 数字播放系统

当我们确信数字化是电视台未来发展的必由之路时，接下来要做的就是如何完美地实现它。电视台向数字化过渡包括图像采集，节目编辑，演播系统，播出系统等环节。播出系统作为电视台节目编播的最后一道关口，直接关系到节目播出的安全性和质量。该系统的数字化是广播电视数字化进程中的一个至关重要的环节，它是数字电视台建设的中心，对电视台整个数字化体系的建立产生长期决定性的影响。但长期以来，播放系统一直是电视系统数字化改造的瓶颈。因此数字播放系统是当前计算机和广播电视领域研究的热点之一^[26,27]。

1.3.1 数字播放系统的发展历程

从手动播出到自动播出。手动播出系统仅是由录像机和切换开关组成，录像机的启动以及切换开关均由人工完成。其主要缺点是节目切换不准确，播出不准时。将计

了如何数字化，而数字电视台建设的先驱—数字播放系统都还在设想之中；而国外发达国家已比我们领先一步，且它们的电视台数字化也正在进行之中，因此我们要急追猛赶，加快我们的电视台数字化进程。

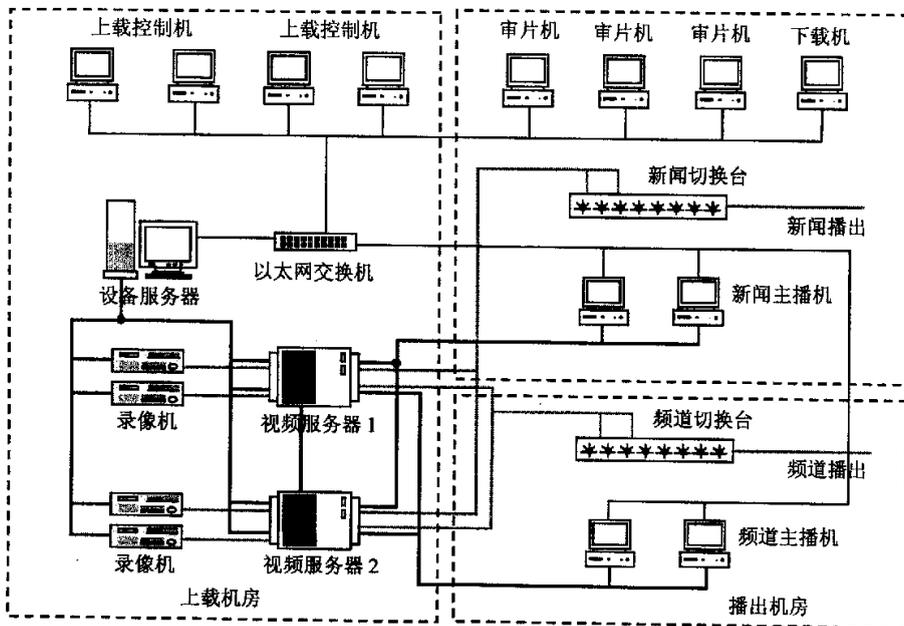


图 1.4 播控系统网络结构

1.3 数字播放系统

当我们确信数字化是电视台未来发展的必由之路时，接下来要做的就是如何完美地实现它。电视台向数字化过渡包括图像采集，节目编辑，演播系统，播出系统等环节。播出系统作为电视台节目编播的最后一道关口，直接关系到节目播出的安全性和质量。该系统的数字化是广播电视数字化进程中的一个至关重要的环节，它是数字电视台建设的中心，对电视台整个数字化体系的建立产生长期决定性的影响。但长期以来，播放系统一直是电视系统数字化改造的瓶颈。因此数字播放系统是当前计算机和广播电视领域研究的热点之一^[26,27]。

1.3.1 数字播放系统的发展历程

从手动播出到自动播出。手动播出系统仅是由录像机和切换开关组成，录像机的启动以及切换开关均由人工完成。其主要缺点是节目切换不准确，播出不准时。将计

算机技术引入播出系统就实现了自动播出。早期的自动播出系统只是在手动播出系统中加入计算机，由计算机控制设备的启动和切换，保证了节目准时播出。后来引进了计算机自动控制网络，自动控制网络除完成自动控制节目自动播出外，还能实现对台内设备的统一管理和集中使用。

从自动播出到硬盘播出。播放系统作为节目制作后播出的重要环节，对安全性、可靠性和播出质量有很高的要求，在经历了手动切换播出、计算机控制切换开关、主控台和录像机实现自动播出以及使用机械手的自动播出系统之后，播出设备的发展趋势是向播出的数字化、网络化，即向硬盘播出系统或制播一体网方向发展。随着电视中心数字化网络化的发展，节目采集、制作逐渐实现了数字化，传输信号采用数字传输，但播出设备的模拟化成为电视系统数字化改造的瓶颈，限制了电视中心网络化的进程，节目质量的提高。目前各电视台正在寻求制播系统的数字化，由自动播出向硬盘播出过渡，电视硬盘自动播出系统是一种特殊的计算机网络系统，它不仅要具备不停地处理大量视音频数据录入和输出的能力，而且要求系统具备高可靠性，包括数据的高可靠性，随之进一步采用制播一体网将播出系统与非线性制作系统相连。该方式在传输时节省了数字与模拟之间的转化过程，提高了信号的传输质量，降低了信号的损失，为实现全系统的数字化、网络化奠定了基础。三种播出方式的广告播出系统的性能比较见表 1-1^[25]。从表 1-1 可看出采用硬盘播出是技术发展的必然，也是国民经济发展的需要。

表 1-1 三种广告播出系统性能比较

	采用 VTR 广告 播出系统	采用机械手广告 播出系统	采用硬盘广告 播出系统
需要 VTR 数量/台	4~5	9	1
维修费用 (PVW 录像机)	高	更高	低
磁带损耗	较大	较小	几乎没有
平均无故障工作时间/h	约 3000	约 2000	约 20000
系统可靠性	较低，但可切换	一般	较高
操作人数	4~5	3~4	1
操作方式	复杂	较简单	很简单
修改广告播出程序	很简单	容易	很容易
临时插入广告	很困难	容易	很容易
产生的经济效益	100%	150%	300%

硬盘播出系统。硬盘播出系统是利用计算机的服务器模式建立起来的一个数字电视节目自动播出系统，是由服务器、硬盘录像机、磁盘阵列、工作站等通过网络构造而成。近几年来，很多电视台广告硬盘自动播出系统的成功应用，特别是大容量磁盘

阵列、视频传输、图像压缩等技术的日益成熟，使得电视节目自动播出的改造可以建立在硬盘技术的基础上，采用硬盘自动播出系统。部分电视台采用的硬盘播出，使其实现了播出的数字化。随着电视中心数字化、网络化的发展，播出系统将成为电视中心网络化发展的重要部分；同时有线电视网络改造的进一步进行，播出系统与传输系统相连，制播网与传输网共同组成广播电视的大网，实现完整的传输系统网络化。因此我们说硬盘播出系统是实现数字化过渡的一条切实可行的途径；数字播放系统的建设是数字电视台实现的中心和过渡的分水岭。

1.3.2 当前典型的数字播放系统

(1) 北大方正、安恒利（香港）、讯德（香港）等公司推介的品尼高（惠普）MediaStream1600 硬盘服务器自动播出系统方案。该网络结构与图 1.2 类似，其制作部分共有 38 个站点，包括 2 台卫星收录工作站、8 台上下载工作站、10 台精编工作站、8 台桌面编辑工作站、2 台配音工作站、8 台审片工作站、以太网服务器、光纤交换机、以太网交换机、SQL 服务器和 1400GB 的 FC 存储中心，其中 FC 站点 21 个。在网络播出部分，采用了力取（Leitch）数字视/音频矩阵和 2 台 MediaStream1600 硬盘播服务器。

(2) 永达（香港）公司推介的 GVG profile XP 硬盘服务器自动播出系统方案。其典型网络结构图如图 1.5，它是由两台视频服务器、4 台播出工作站、4 台上载工作站和一台文件服务器通过集线器构成。文件服务器为以太网内各台工作站服务，也是公共数据库存储的地方。

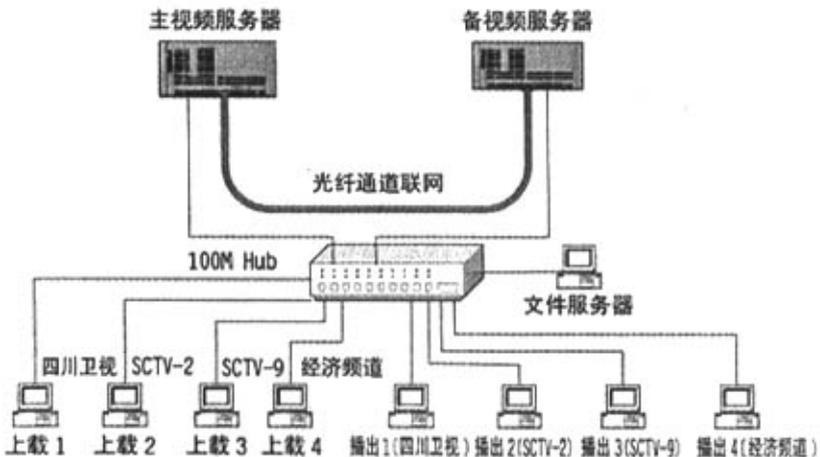


图 1.5 profile XP 多频道硬盘播出系统结构

(3) SoBey 公司的硬盘自动播出系统方案如图 1.6。

采、编、播一体化解决方案

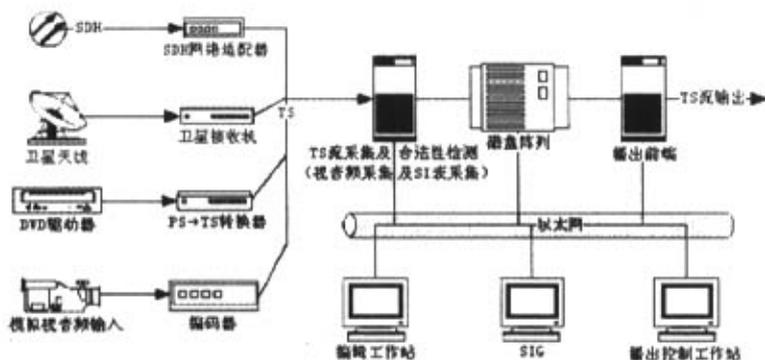


图 1.6 索贝硬盘播出系统结构

1.3.3 数字播放系统研究的必要性

纵观上述各公司系统方案及国内已有的局部数字化系统，不难发现：①其播出系统视频服务器都是与国外公司合作，直接引进国外视频服务器及存储阵列。如方正既用品尼高视频服务器，又与索尼合作，进行上层编播管理、素材管理，资源管理等应用的二次开发。索贝也是如此。②都是以 MPEG2 作存储传输格式。播放系统的一些关键问题如精确到帧的播放，节目间无黑场，无缝切换，即播出系统的可靠性，可扩展性，都会随着国外服务器的引进而解决，而中国作为一个有巨大市场的大国，研究有自主知识产权核心底层技术的数字播放系统是很有必要的。而且国内当前的数字播放系统都是基于 MPEG2 文件格式，没有一个是针对高清晰度电视的，一旦转为高清晰度电视，改动也较大。面临更多频道和大量新兴业务要求时，无法满足需要。尤其是面对互联网。因为互联网要求可提供多种格式的节目源，而节目一旦压缩成特定的压缩格式后，便无法改换成其他压缩格式。因此我们与某影视公司合作研究了能直接过渡到高清晰度电视的数字播放系统。

1.3.4 电视台数字播放系统的关键技术

播出系统是电视台不可缺少的重要部分，一个合理的播出系统应综合考虑系统的安全性、应急能力、播出功能、与电视台其它系统的连接、扩展播出功能等技术指标以及系统造价等问题。数字播放系统必须实现的技术指标有：

(1) 实现广播级视频质量的播出。播出系统的视音频主通道的技术指标应达到

或超过视音频通道技术指标甲级标准。要保证整个通道的指标就要求在整个通道中的每一设备指标都必须达到或超过视音频通道技术指标甲级标准。

(2) 保证每天节目播出时间内正常播出。实现播出的灵活性,即可顺序播、插播、直播、延迟播。

(3) 保证电视节目播出的高可靠性。自动播出系统的可靠性主要取决于直接参与播出的设备的稳定性和可靠性、控制系统的合理性和安全性以及应急处理能力这三个方面。直接参与播出的设备必须经过国家标准测试,控制系统应具备完整的控制逻辑和处理各种应急事件的能力。对于播出系统的安全备份,应遵循一个原则,即,在整个系统中没有“死点”,不会因为一个设备的问题导致整个播出系统瘫痪的原则。

(4) 实现播出与采集的帧精确、无黑场。帧精确的含义是节目录入视频服务器时要根据给定的入点和出点一帧不能多,一帧不能少。节目播出时要根据给定的时间点开始播,一帧不能多,一帧也不能少。播出时节目间的衔接不能出现黑场,播出时视频切换和音频切换完全同步,一帧都不能差。

(5) 要求系统结构可扩展。播出系统在设计时一定要考虑满足技术发展的要求。电视台正从模拟向数字化过渡,从磁带化向无带化发展,并可能朝着高清晰度电视和地面卫星广播等方向发展,因此系统结构有好的扩展性。

(6) 与台内其他系统的无缝连接。网络化和数字化是电视台发展的必然趋势,播出是电视台的重要环节,在建立播出系统时必须考虑与台内其他系统的连接问题:一是,信息的交换如节目播出单、广告播出单、播后信息的返回,这类信息的交换一般通过以太网就能实现;二是节目内容的交换,从其他非线性编辑网络或视频服务器的播出内容如何最快、最安全和方便地传送到播出部,实现全台制播一体化的目标。

1.4 本文研究的主要内容及意义

随着计算机技术向电视领域的渗透,电视业正迎来一场革命性的变化,即电视的数字化和网络化。电视台的数字化包括图像采集、节目编辑、演播和播出等环节,播出系统作为电视台节目面向观众的最后一道关口,直接关系到节目播出的可靠性和图像质量,因而该系统的数字化是广播电视数字化进程中的一个至关重要的环节。由于数字视频播放系统是电视台建设的核心,对整个数字化体系产生长期决定性的影响,因此它是当前计算机和广播电视领域研究的热点之一^{[6-8][28-29]}。

数字视频播放系统是一个重大而复杂的系统工程,它包括:硬件的系统集成以及软件的音视频采集、总控、音视频播放、底层存储控制器等方面的研发。从上节所述

的数字播放系统必须实现的技术指标可看出音视频的播放是系统重之又重的部分，系统总控的命令要由它实现，它输出的音视频数据直接面向观众，它的播放质量直接影响观众的收看效果；而且它是音视频采集的逆过程，它的技术难点解决了，相应地，音视频采集的难点也迎刃而解。而且它与系统体系结构的可靠性有关。作者作为主要研究人员，围绕着音视频播放的这个中心任务，参与了系统的总体设计、主要承担了无压缩播放、压缩播放、以及满足特殊播放需要的直播、延迟播、实时格式转换等的研发工作。为了完成上述研发工作，必须研究相关问题。因此本文研究的主要内容是：

(1) 数字播放系统体系结构的研究。主要从系统可靠性、可扩展性、性能平衡设计等方面研究省级电视台的高质量数字播放系统的核心——存储播放子系统的构成。

(2) 无压缩数字播放及其帧精确的研究。主要从无压缩数字视频播放系统的播出工作特性、共享 buffer 的建模与计算、多线程 I/O 技术等方面研究实现每秒 25 帧平滑无黑场播出。

(3) 压缩数字播放的帧精确与无缝转接的研究。主要从压缩数字播放系统的 I/O 路径、缓冲区设计方法及其性能等方面研究能实现压缩播放系统的帧精确与无缝转接等实时性的播放算法。

(4) 满足特殊播放要求的直播、延迟播、实时格式转换等的技术研究。

(5) 数字播放系统服务响应时间的研究。从数字播放系统的 I/O 路径、影响播放系统服务响应时间的因素，播放系统服务响应时间的数学模型以及相应的改进措施方面研究数字视频播放系统的响应时间。

2 数字视频播放系统的体系结构

数字视频播放系统的软硬件平台是决定播出系统实时性、可靠性和灵活性的关键。实时性与选择的视频设备性能、通道传输能力和存储子系统的速度和容量有关。可靠性则与备份配置有关，而系统调度的灵活性与应急的能力，很大程度决定于视音频设备、切换开关、视频服务器的配置及视频播放服务系统的体系结构^{[28][30-31]}。

2.1 典型的播放系统的可靠性、可扩展性比较

(1) 泰克公司的 Profile 300 硬盘播出系统

Profile，对中国的电视台来说是比较熟悉的，应该说它是最早被大家所接受的硬盘录像机。早期的 Profile100 和 Profile 200 是以 M-JPEG (Motion-Joint Picture Expert Group) 方式存储的，而 Profile 300 是在原来 profile 200 的基础上做了改动，增加了 MPEG2 的编码板和解码板。Profile 网络按主备方式工作，主机和备机有 2 进 2 出 (最多 4 出) 数字通道，均可自带硬盘 (9G×8 或 18G×8)，可再通过 SCSI (Small Computer System Interface) 口外接硬盘箱。主备机通过光纤通道连接，控制部分通过以太网联接^[32,33]。泰克的视频公司在 90 年底在 GVG 公司收购后，推出新一代视频服务器 Profile-XP。该机型采用 MPGE2 存储，支持帧精度剪辑和流传送，通道数扩展到 8 个 (最多 6 入)，总体性能比 Profile 300 有很大提高。目前 Profile 网络在国内的用户有南京有线台、福建电视台、河北省台、厦门电视台、山西省台。该系统的 OEM (Original Equipment Manufacturer) 开发商主要是合肥高视和安徽现代两家公司。福建电视台的系统主、备机各配置 16 个 18G 硬盘，共计使用 2 组 288G 的硬盘。在 8Mb/s 码率时，可存储播放 80 小时的节目，价格在 30 万美元左右。

(2) 惠普 MediaStream 硬盘播出网络 (见图 2.1)

该系统采用惠普视频服务器、全数字影视节目的存储和重放，广泛用在体育插播、多通道广播、节目重放、网络延迟、NVOD、节目分配和 DTV 播出。该系统采用高可靠性和可扩展性设计。主机 HP E2524C 共有 6 个通

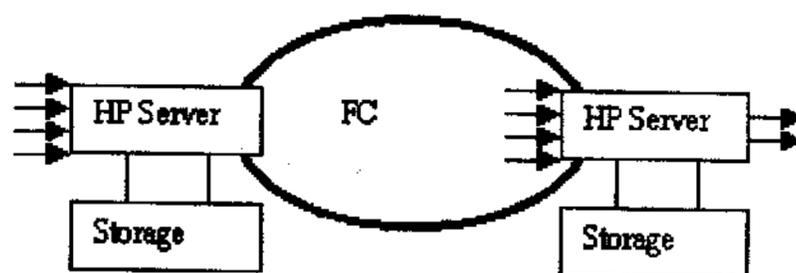


图 2.1 HP 公司 Media Stream 硬盘播出网络

道，其中最多 4 个输入，每通道有两块专利解码芯片，利用“CleanCut”技术进行剪辑。可做到任意帧播放、变速慢进和往复搜索，以及帧精度标记入/标记出。采用 6—15Mb/s

或者 6—30Mb/s 格式储存。在 8Mb/s 速率下, E2524C 可存储 36 小时节目, 加上选件 044 可储到 108 小时节目。该播出系统同样采用主备机独立存储、光纤通道联网方式, 安全可靠。它的国内 OEM 开发商, 北大方正, 已在香港亚视做了一个系统用于广告、新闻播出。

(3) DRM 公司的 Doreme 硬盘播出网络 (见图 2.2)

该系统采用光纤通道联网, 与前述几款不同的是除了共享存储之外, 其服务器有多个视音频接口 (VID)。视频输入在通过该接口进行 A/D 变换、图像压缩编码后, 再通过 SCSI 总线传至服务器, 最后由服务器的光纤通道转发存储于硬盘塔。

此种硬盘存储网络也是主、备机热备份方式工作, 主备机的切换自动进行。为便于编辑、上字幕等, 该系统采用 M-JPEG 存储。在 5:1 压缩比时, 码率为 $166\text{Mb/s} \div 5 = 33.2\text{ Mb/s}$, 可达 Betacam sp 质量, 若连续播放 100 小时, 则需用硬盘容量为 $100 \times 3600 \times 33.2 \div 8 = 1500\text{GB}$, 需配备容量为 18G 硬盘达 83 个! 该系统采用 M-JPEG 存储, 通过以太网可准确地寻找硬盘存储节目的编辑点进行编辑, 也可通过高压缩比以低数据率将图象传送到终端进行审查。灵活方便, 只是系统构造复杂, 硬盘数量巨大。从可靠性和经济性考虑, 似不宜作为播出设备的范例。南京

电视台为购置这套系统价格为 600 万人民币。

(4) 力取公司的 ASC VR300 硬盘播出网络 (见图 2.3)

力取 (Leitch) 公司的 VR300 光纤通道共享存储视频服务器, 利用 ASC 独特的 FibreDrive 技术, 结合光纤通道环, 使所有工作站同时分享所有在线的视频资源。一个 VR300 可连接 5 个以上带光纤接口

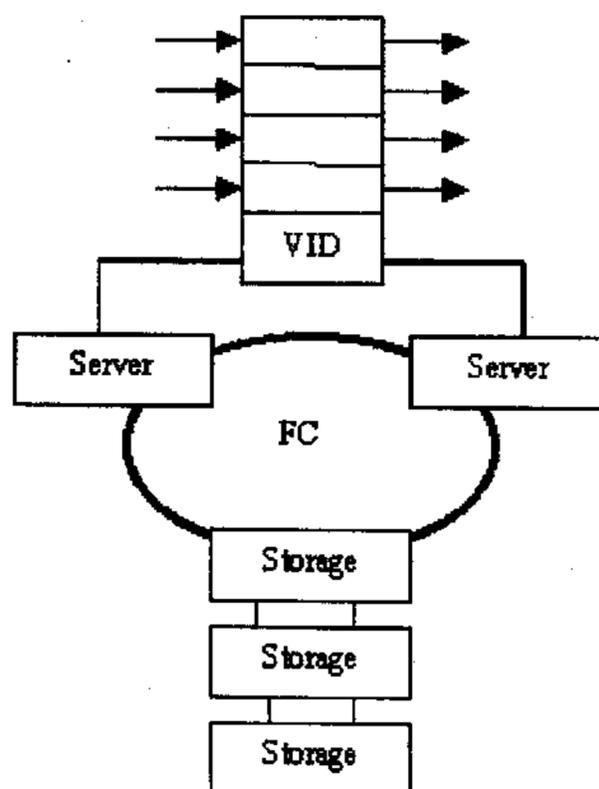


图 2.2 DRM 公司 DoReMe 硬盘播出网络

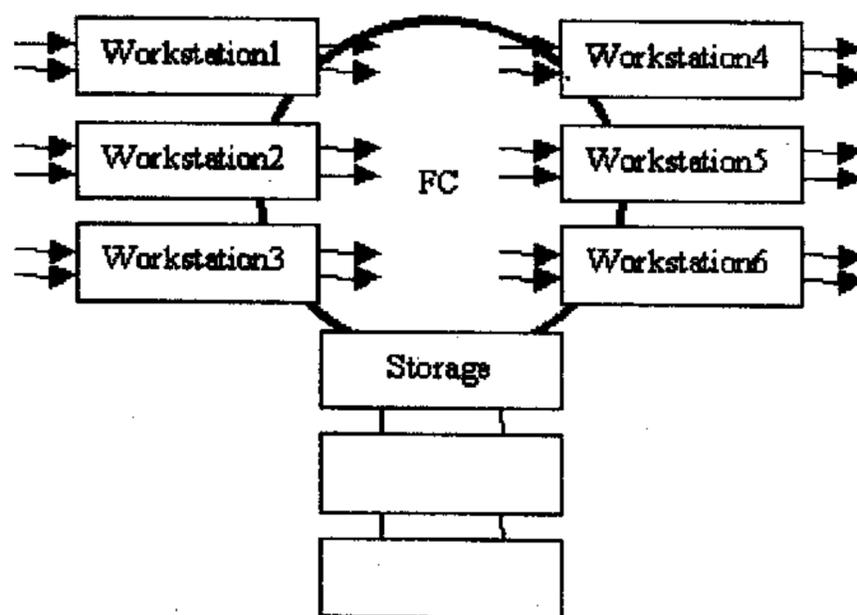


图 2.3 力取公司 ASC VR300 硬盘播出网络

的硬盘塔，拥有 20 个以上 I/O 通道的系统可联接超过 100 个光纤接口的硬盘塔。此方案与我国众多的非线性编辑系统的联网方式类似。目前它应用 M-JPEG 压缩存储，今后可升级到 DVCPRO 和 MPEG-2 压缩标准。

(5) SeaChange 公司的 Media Cluster 硬盘播出网络 (见图 2.4)

与前述几款不同，该系统采用服务器堆的工作方式，运用其自己独创的先进技术方案— Media Cluster，由 4 个核心部分综合而成^[34-35]。它具有多级故障自恢复功能、易于扩充的网络结构、多输入输出接口，多 UPS 电源共享等特点。多级故障自恢复功能没有采用传统的主备方式来实现，这样可以减少冗余。它的实现方法是：将任意一个输入口的输入数据通过算法分配至每个节点服务器，再由每个节点服务器分配至 RAID 系统。当任意一个 RAID 中的磁盘驱动器发生故障时，同组的磁盘驱动器能够通过 RAID 算法维持播出，并自行恢复数据到换上的新磁盘驱动器中。而任意一台服务器发生故障也同样由其它服务器维持播出并恢复数据，因此有两级数据保护。

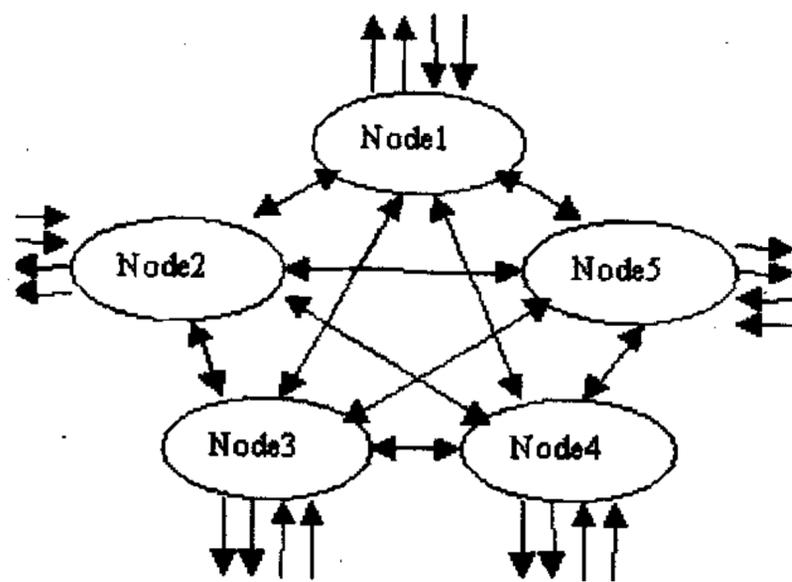


图 2.4 SeaChange 公司 MediaCluster 硬盘播出网络

该网络由最少三台最多九台服务器（节点）组成。为提高可靠性，每节点只配置 10000 rpm 的 SCSI 硬盘驱动器 12 个(4.3GB、9GB、18GB 或 36GB)。若采用 5 节点，18G 硬盘驱动器，硬盘总容量为： $12 \times 18 \times 5 = 1080\text{GB}$ 。若采用 MPEG2 格式存储，按 8Mb/s 的码率，可存储 192 小时节目。该系统采用 100Mb/s 的快速以太网联网，通过硬件结构使每个节点均与其它节点互联，并采用了 Media cluster 专利技术的 SeaNet 视频网络传输协议。该协议可满足高带宽传输通讯的要求，并允许全部输入、输出同时存取共享的视频节目素材，其传输效率高达 80% 以上。

比较以上五款网络，泰克公司的 Profile 和 HP 公司的 Media Steam 采用可靠性较高的专用硬盘录像机和专用服务器，采用高效的 MPEG2 压缩方式，通过 SCSI 接口自带硬盘，并通过成熟的 FC 网联成主备网络。作为广告、体育、新闻多频道插播以及专业频道播出，其实用性和可靠性都是令人信服的，只是冗余过高，多频道大容量播出时效率相对较低。DRM 公司的 DoReMe 和力取公司的 ASC VR300，采用低效的 M-JPEG 压缩方式共享硬盘的网络结构，适合于节目的后期制作。而

SeaChange 公司的 Media Cluster 硬盘播出网络, 其新颖的设计思想、高效率的压缩方式和存储方式、大容量多输入输出端口, 安全的二级保护措施, 都有理由让我们继续倍加关注。

2.2 高质量数字播放系统的体系结构

通过对以上硬盘播放系统的分析我们也可以看到: 多数系统均采用了专用视频服务器, 一方面造价极其昂贵, 不适于在经济实力不足的中小型电视台推广; 另一方面, 正是由于专用视频服务器昂贵的价格以及设计上的专门性, 系统硬件结构不易改动。从系统总体结构上来分析, 多数硬盘播放系统编辑网和播控网分离, 对整个系统网络缺乏统一的管理; 素材的录入一般通过录像机进行人为控制, 录入过程不可打断, 没有实现素材上载的全自动化控制; 同时帧精确和延迟播放等关键技术的实现基本都是建立在增加冗余硬件设备基础上的。

正是基于以上的分析结果, 我们研制了一套高质量数字播放系统 (High Quality Digital Broadcasting System, HQDB)。同其他系统相比, 该系统面向中大型电视台, 全部构建在千兆光纤网络平台上, 实现编辑网和播控网的融合, 实现整个电视台系统的高度集中自动化管理; 采用专门的上下载工作站实现节目素材上、下载的全自动化控制。而且以普通的服务器加视频卡代替专门的视频服务器, 降低系统的整体价格; 同时对帧精确、延迟播放等关键技术在该系统上进行深入研究, 实现低成本、高性能, 提高了产品的性价比, 价格是国外同等功能系统的五分之一。

视频播放的播出子系统及其品质与总的系统体系结构有关, 因此作者参与了系统的总体设计及系统的统调。为了以下章节阐述的方便, 下面将先对该系统的总体方案与软硬件平台作一简单描述。

2.2.1 HQDB 的硬件结构

该系统硬件主要由以下几部分组成: ①存储播放服务器, 负责素材的存储与播放; ②网管工作站, 负责整个播放网络的管理, 也包括故障检测与故障恢复; ③编目工作站, 负责节目素材的编辑和节目单的编制; ④上下载工作站, 负责素材的采集与上下载; ⑤可编程切换矩阵及千兆交换机。系统硬件体系结构如图 2.5 所示。

(1) 存储播放服务器组

存储播放服务器组由 4 台存储播放服务器组成, 每台存储播放服务器由 HP 服务器和通过 SCSI 总线挂接的 2 台磁盘阵列组成。4 台存储播放服务器组共分为 2 组, 每组包括一主一备。存储播放服务器安装有视频采集与播放卡, 主要负责完成各种播放与存储节目的功能。任意时刻, 主备机同步播放相同的节目, 播出通道通过切换矩

阵接到同一视频输出通道，通过通道冗余实现节目的安全播放。另外，存储播放服务器还负责素材的存储和管理，存储播放服务器通过机内配备的千兆光纤网卡和千兆交换机连接，编辑好的素材都由上载工作站通过千兆局域网上载到存储播放服务器的磁盘阵列上，存储播放服务器根据素材的不同予以分类存储，同时实现素材的网络共享，方便对素材进行网上浏览。存储播放服务器还有一定的故障检测功能，能实时地向网络管理工作站报告当前的运行状态。

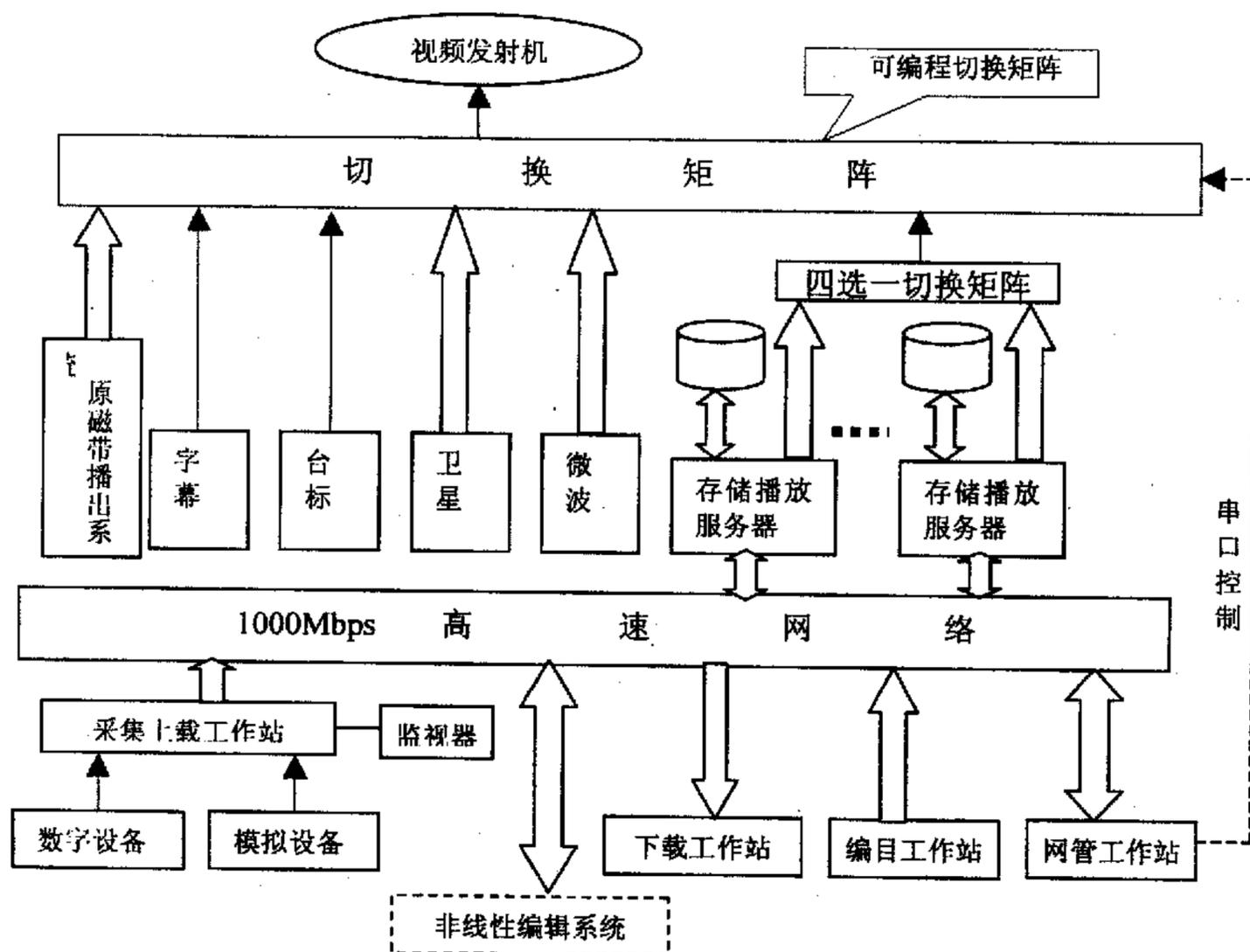


图 2.5 系统硬件体系结构

(2) 网络管理工作站

网管工作站由一台 HP 工作站组成，它也通过千兆网卡和千兆交换机相连，是整个播放网络的控制中心，主要负责整个播放系统的管理和协调工作。它实时监视系统的运行状态，当任何计算机节点发生故障或状态变化时都要实时报告，当存储播放服务器故障时，它必须负责完成故障的仲裁并通过串口线实现对切换矩阵的控制。它还拥有节目素材的列表，向编目工作站提供素材列表信息，以便进行节目单的编制。

(3) 上、下载工作站

上下载工作站由两台 HP 工作站组成，工作站配有视频采集卡和大容量的 SCSI

硬盘，主要负责节目素材的采集和上载。它提供多种采集接口，可以实现对磁带机、VCD 机、卫星接收机等设备视频信号的采集。其配备的采集软件可以使采集精度达到“帧精确”，并能对采集效果进行实时监控和预览。采集的素材先存放在本地大容量硬盘上，经分类整理，通过千兆局域网上载到存储播放服务器的素材库中。上载方式采用全自动方式，操作人员只需要提交上载作业，上载软件会在合适的时刻自动将素材上载到视频服务器上，不再需要人工干预。同时，他还要提供下载功能，可以将视频服务器上的素材下载到本地硬盘，便于实现素材的收藏和刻录。

(4) 编目工作站

编目工作站也由一台 HP 工作站组成，它主要负责完成节目单的编制和发送工作，它通过网管工作站获取存储播放服务器上的素材列表，可以通过网络对素材进行浏览，并根据现有的素材进行节目单的编制，同时它也是播放命令控制中心，负责 HOLD 及 TAKE 播放命令的发送。

(5) 千兆光纤交换机

在本播出系统中，各计算机节点以 Cisco 光纤交换机为中心构成一个千兆光纤局域网。视频素材可以作为数据在高速光纤网络上共享，从而达到视频围绕设备运动的能力。光纤网提供 1Gbps 的环路传送速度，能高速传输大量的数据，传送后无需担心素材损失，网络协议保证在每次传送过程中文件的完整性；保证每台存储播放服务器都拥有足够的传送带宽，在一个光纤通道环路上可连接多台存储播放服务器，有利于系统的扩展。它在内部支持与各种类型的物理接口，能区别物理接口和逻辑协议，如许多协议能在一个物理接口运行。采用光纤通道后，光纤通道数据流使文件在传送进行的同时能够回放、编辑、复制或查看文件，这为在网络环境中的工作提供非常有效的操作方式。

千兆以太网标准 1000BaseX 的目标是要以 1000Mbps 的速率在多模、单模光纤或屏蔽平衡型铜缆上实现半双工 / 全双工操作，使用 802.3 以太帧格式，或在半双工模式下使用 CSMA/CD 介质访问控制方法。每个冲突域支持一个 repeater hub，并与 10BaseT 和 100BaseT 技术相兼容。在 MAC 层，1000BaseX 与它的前驱 10BaseT 和 100BaseT 采用的技术相仿，只是由于传输速率的提高，需要作出一些调整和协调，重要和关键的工作是在物理层。

本系统选用 12 端口千兆交换机 Cisco Catalyst c4003 组成高速网络平台，以满足无压缩视频信息的传输需求。

(6) 可编程切换矩阵

台标信号、字幕信号、存储播放服务器播出的媒体流信号、老的模拟播放系统的播出信号、微波与卫星信号等都通过可编程切换矩阵输送到发射台。网管工作站通过一个小的四选一矩阵控制存储播放服务器播出信号的切出。

2.2.2 HQDB 硬件配置

1) **存储播放服务器**: 采用 **HP LH3000/r**, 内插 64 位 Targa3000PCI 视频卡。将配置的磁盘阵列与服务器等设备形成一个总的机柜, 体现出自己的品牌。所有型号都包括: PC-133MHz ECC SDRAM, 具有内存清除功能, 集成双通道 I₂O HP NetRAID, 控制器, 集成的 10/100TX 网卡, CD-ROM 驱动器, 软盘驱动器, 热插拔电源和风扇, 集成惠普远程助理, HP TopTools for Servers, 及 HP OpenView ManageX/Event Manager。

2) **工作站**: 选用 **HP P-Class**, 按照所承担角色不同, 各工作站在配置上稍有不同, 主要计有上载工作站 2 台, 编目工作站 1 台, 网络管理工作站 1 台, 下载工作站 1 台。其中, 考虑到上载节目的方便性和安全性, 上载工作站配置了一个 72GB 的本地磁盘作为缓存。其它工作站仅限于完成本身工作, 无需太大的存储空间, 为避免造成浪费, 只配置系统盘。

3) **交换机**: 采用 **Cisco Catalyst 4003**, 主要是具有足够的冗余端口。本系统配置的 4003 提供 16 个 GBIC。所配置的模块有:

- (1) WS-C4003-S1 Cat4000 Chassis(3-slot), Supervisor, 1 AC PS, Fan Tray
- (2) WS-X4418-GB Cat4000 GE Module, Server Switching 18-Ports (GBIC)
- (3) WS-G5484 1000BASE-SX "Short Wavelength" GBIC (Multimode only) X 16

4) **数字切换台**: 要求能够使用 RS422A 串口进行遥控。选择的型号是 Tektronix 的产品。参照主流配置, 采用 DVS2000-C (包括 BKDS-2530 RS-422A PC 串行接口板, 连接线, 遥控软件开发工具等)。

其它: 包括光缆、光缆接头、千兆网络适配器, 电源及相关配件。

硬件设备清单如下所示:

- (1) Targa 3000 X 8
- (2) HP Net Server LH3000/r X 4
- (3) HP P-Class Workstation X 6
- (4) Cisco Catalyst 4003 X 1
- (5) Sunny49160 SCSI Card X 8
- (6) RAID X 8
- (7) 1000BaseX NetCard 3COM 3C985-SX X 10
- (8) Clock GDB-5sb & SPD106 X 1

2.2.3 HQDB 的软件组成

软件系统主要包括 5 个子系统: 系统控制子系统、上/下载子系统、数据存储与播

放子系统、网络故障检测与诊断子系统、时钟同步子系统。

1) 系统控制子系统

为了有效地协调系统的运行，在系统总控中特别嵌入 3 个控制模块（子系统）：存储播放控制子系统、故障监测控制子系统、时钟同步控制子系统。

系统总控与存储播放控制子系统间的控制策略：

对于上下载子系统向存储播放子系统提出的上下载请求，系统总控首先根据网络运行状态（如网络的流量状态、节点设备就绪状态、功能子系统的运行状态等）进行适当的预处理（如上下载任务的过滤），再以适当的方式提交存储播放控制子系统，存储播放控制子系统进一步根据存储播放服务器、磁盘阵列的工作与负载状态作出响应；对于编目子系统提出的节目单编辑/更新请求，经系统总控许可后，交由存储播放控制子系统负责节目单的编辑/更新；对于存储播放控制子系统向上下载子系统提出的有关上/下载请求（如因存储播放控制子系统提出上/下载暂停后的上/下载恢复请求），同样经由系统总控许可后，交由上下载子系统作出响应。

系统总控与故障监测控制子系统间的控制策略：

系统总控根据故障监测控制子系统给出的设备就绪状态、功能子系统的运行状态等信息，采用文字或图表形式实时地进行系统运行状态指示；当故障监测控制子系统报告系统发生故障时，系统总控进行及时报警（如特殊效果信息显示或声音报警），以便工作人员及时采取措施解除系统故障。

系统总控与时钟同步控制子系统间的控制策略：

系统总控按照系统存储播放对节点的同步要求，合理地确定同步过程的时间间隔参数，时钟同步控制子系统根据同步时间间隔适时地对系统主要节点进行同步校时。

2) 上/下载子系统

在不影响系统播出工作的情况下，完成规定的上/下载任务。在播出过程中，可以根据实际情况暂停、停止或恢复以前未完成的任務。保证上载的速度和连续性，系统由双工作站轮流上载。根据需要制定上载任务，根据制定的上/下载任务向总控发出上/下载请求，接到上/下载确认之后建立与存储服务器之间的数传连接。

数传过程中，可以根据任务的优先等级进行动态的处理：处理紧急任务时，暂停某一任务；取消某任务，并通知总控删除已上载的内容；收取总控信息，恢复先前被中断的传输。

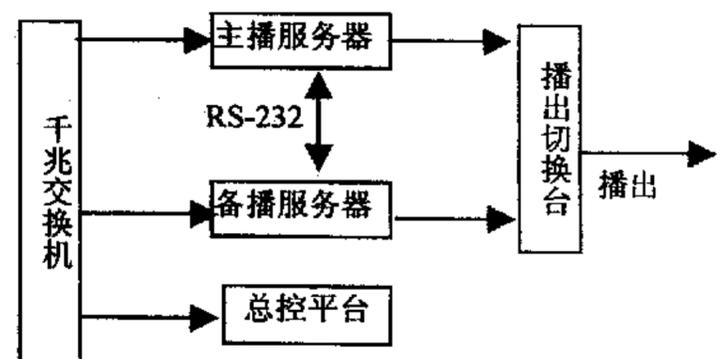


图 2.6 主备播同步连结实现

3) 存储播放子系统

数据存储与节目播放子系统由存储播放控制子系统与存储播放服务子系统组成,存储播放控制子系统嵌入系统总控,存储播放服务子系统驻留在存储播放服务器中。系统正常运行时,备播服务器启动一个同步进程,等待主播服务器的同步信号,主播同步进程在同步点通过串口将同步信息传送到备播,备播根据同步信号调整并修改相应的状态,使得主备播保证帧输出同步。当主播发生故障时,系统切换到备播服务器播出,保证帧连续播出。主备播间的连结形式如图 2.6 所示,两台存储播放服务器之间通过 RS232 串口连接,传递同步控制信息。

存储播放服务子系统接收、控制网络传输的音视频数据流并将其可靠地存储到磁盘阵列。从磁盘阵列中读取音视频流,通过 T3K 经由切换矩阵连续播出。存储播放服务子系统给编目工作站提供文件数据表单,并提取现在要播放的节目单和明天要播放的节目单,以便预取和存储。存储播放服务子系统接收从网管工作站发来的启动命令后开始存储与播放工作。

4) 网络故障检测与诊断子系统

网络故障检测与诊断子系统由故障检测与诊断控制子系统、故障检测与诊断节点子系统组成。故障检测与诊断控制子系统嵌入系统总控,故障检测与诊断节点子系统驻留在网络系统各节点上。

故障检测与诊断节点子系统收集本地的硬件和软件两方面的信息,通过判断,非正常时发送消息给故障检测与诊断控制子系统。

另外,故障检测与诊断节点子系统通过定时给故障检测与诊断控制子系统发送活动消息表明自己的工作状态。若故障检测与诊断控制子系统在一定的时间间隔内未收到这个消息,应认为故障检测与诊断节点子系统有问题。

5) 时钟同步子系统

系统同步子系统包括:同步控制子系统、同步节点子系统。同步控制子系统嵌入系统总控运行在网管工作站/控制台上,同步节点子系统运行在系统网络的其它节点上。

系统时钟同步策略,对整个系统业务而言,工作需要连续、有序的进行,这就要求系统中所有的设备、组件按照统一的时间基准运行。

系统采用标准授时设备 Clock GDB-5sb & SPD106。Clock GDB-5sb 数字母钟:可外接 CCTV 信号,解码标准北京时间。并且集驱动、显示、守时、去干扰、自动校时、自动亮度适应等功能于一体。SPD106 232/422 接口分配器可将 1 个标准 232 通讯接口分配为 6 个接口。因此本系统用它向网络节点的 4 台存储播放服务器、上下载工作站定时发送时间信号。它首先从 CCTV 信号中提取标准北京时间,由母钟输出时间

码给时间脉冲信号分配器，信号经分配放大后再驱动计算机，并以此为基准通过 SPD106 232/422 接口分配器交由各同步节点子系统进行节点系统校时。

系统总控中的同步控制子系统每间隔 40ms 通过时钟同步端口向网络各节点发起轮询，得到各节点运行环境时间后，加上网络延迟平均值（试验得出的统计数据）与标准时间（总控通过自己的串口线获得的时间）进行对比，若有超出 40ms 范围的情况就发送系统重新校时命令，各节点的同步节点子系统收到命令后运行其校时程序，重新对时。

播出质量和播出可靠性是数字视频播放系统最重要的两个技术指标，存储播放子系统的设计水平直接决定了整个系统的性能。而且从系统的整体构成来看，每个子系统的设计都是为保证存储播放子系统的正常运行而设计，因此下面将详细讨论满足实时性、可靠性的存储播放子系统的性能平衡设计。

2.3 HQDB 的存储播放子系统

电视台数字视频播放系统是一个多媒体计算机系统。传统的模拟播放系统因画面质量欠佳，节目管理、拷贝和保存都极不方便，必须进行技术改造，而使用数字化技术改造传统的电视播放，最核心与最关键的位置是存储播放子系统。

现行的数字化播放系统大多为支持 MPEG-2 MP@ML 的视频流的播放系统^[36-39]，这种播放系统虽然造价相对较低，但因内容修改导致多次压缩与解压缩，图像质量受损较大，不能满足对画面质量的要求^[40]。而兼容无压缩视频流的 HQDB 就不存在此缺点，但对其存储播放子系统的性能则提出了极高的要求，与之相应的节目编排、管理、传输、保存等复杂的工作都将通过网络在存储播放子系统中完成。因此本节将对这种满足高质量数字播放的存储播放子系统进行系统的分析研究。

HQDB 因要兼容无压缩媒体流的播放，因此对其存储播放子系统有相应的性能要求。

(1) 高数传率。无压缩媒体流的数传率特性，以 SDTV 标准清晰度电视为例，视频分辨率是：水平方向 720 象素，垂直方向 576 象素。若采样格式是 4: 2: 2，样本采样精度为 8bit，则 PAL 制式数字电视的无压缩视频流的数传率为：

$$720*576*25*8*2/8=20.736(\text{MB/S})$$

对音频数据，若采样频率为 48KHZ，样本精度为 16bit，通道数为 2，则无压缩音频流的数传率为：

$$48000*16*2/8=192(\text{KB/S})$$

因此 PAL 制式标准清晰度数字电视无压缩媒体流的总数传率为：

$$20.736+0.192=20.928\text{MB/s}$$

要实现播出时的不掉帧、无黑场,就必须保证从存储子系统到视频处理器之间的 I/O 路径上至少有 21MB/S 持续恒定的媒体流。对于 HDTV 高清晰度电视 1920/1080/60frames/s MP@HL^[40],其水平方向 1920 像素,垂直方向 1080 像素,其无压缩媒体流的总的数据传输率约为 238MB/S。压缩 14 倍后其数传率为 17MB/S,而一般的基于 MPEG2 MP@ML 标准清晰度电视的数字播放系统在同样的压缩比下则只需具有 1.5MB/S 的数据传输率即可满足其媒体流要求。

(2) 海量存储容量。电视台要求能预先存储容量为一星期的节目。若每天要求连续播放 17 小时,除去重播节目,每天的节目容量为 8 小时即可,则须预先存储容量为 $8*7=56$ 小时的节目。因此其存储容量要求至少为:

$$56\text{h} \times 3600\text{S/h} \times 21\text{MB/s} = 4134.375\text{GB} = 4.037\text{TB} \quad (2.1)$$

若每天连续播放 24 小时,并且考虑存储系统的冗余、镜像,以及磁盘存储必然会存在的碎片等,则所需存储容量会更大。

(3) 高可靠性、可扩展性。可靠性是电视台数字播放系统的一个重要性能指标,是电视台的生命,因此相应地要求其核心存储播放子系统也具有高可靠性。

电视台数字播放系统的可靠性是指电视播出的高质量不间断的可靠程度,或者定义为不中断播出的概率。在电视台的管理中,通常有一个“停播率”指标,那么系统可靠性就定义为:可靠性 = $1 - \text{停播率}$ 。目前一般省级电视台停播率指标定为 ≤ 20 秒/100 小时^[41],则可靠性指标是:

$$\text{可靠性} \geq 1 - 20\text{s}/100\text{h} = 99.9944\% \quad (2.2)$$

可见这是一个极其高的要求,存储播放子系统作为数字播放系统媒体流的必经的核心环节之一,其可靠性也必须达到 99.9944%。为了实现此要求,除了用软件算法(将另文描述)进行精确控制外,存储播放子系统也必须在硬件结构上提供可靠性基础。因为存储播放子系统是可修系统,因此我们可度量其可靠性如(2.3)式所示:

$$A = \text{MTBF}/(\text{MTTR} + \text{MTBF}) \quad (2.3)$$

其中 MTTR (Mean Time To Repair) 是系统排除故障(或更换新设备)的平均时间; MTBF (Mean Time Between Failure) 是系统平均无故障时间。

电视台因经济实力,和对数字化改造的投入不同,相应地对其播放频道要求、存储容量要求也不同,要求存储播放子系统具有高可扩展性。

由图 2.5 可看出:存储播放子系统由 4 台存储播放服务器组成,每台存储播放服务器由 HP 服务器和通过 SCSI 总线挂接的 2 台磁盘阵列组成。而且存储播放子系统所具有的功能都通过存储播放服务器实现,因此下面先讨论存储播放服务器的配置。

2.3.1 存储播放服务器的配置

视频卡有很多，达广播级质量的有 Optibase movemake200、Pinnacle TARGA3000、Apollo Expert Plus 广播级 DVD.SVCD.VCD 实时压缩卡等，因为 HQDB 系统要兼容无压缩与压缩媒体流，因此选用既有无压缩功能又有压缩功能的 TARGA3000 卡。

T3K 的硬件结构都使用以存储器为中心的结构^[22]。由 HUB 视频处理器来管理，存储器中的数据是可以被别的程序处理的。T3K 把输入到卡上的基带信号立即转换成数据存储到存储器中。操作在 T3K 的存储器中进行，在主机的存储器和 T3K 的存储器之间可以进行很方便的数据交换。输出也会被转换成基带信号。视、音频的编、解码器都可以访问 T3K 的存储器。通过板卡上的一个 VIP PORT，一个输出的视频流可以被直接送到一个独立的 VGA 卡上，这样可以使第三方的 VGA 卡发挥他们的作用。另外，板卡上还提供一个 3D 效果循环连接器，可以连接第三方的 3D 处理器。T3K 卡所带存储为 128MB，服务器内存至少为 128MB。

存储播放服务器必须接收消息、节目流，进行节目的存储、读取和回送。对磁盘阵列来说，有存和取两种数据流。据实验测出相邻 50G 阵列的切换时间最小为 10 毫秒，最大为 110 毫秒，100MB 阵列间切换为：1 毫秒与 2 毫秒。61MB (win2000 用) + 16MB (VC 用) + 2*20.48MB (缓存帧用) + 128MB = 246MB。因此当服务器中插一块 T3K 时，服务器内存至少为 256MB。

又据 T3K 的 64 位 PCI 槽的要求，选定 HP Netserver LH30000 为存储服务器，其性能指标如表 2.1。软件在 WIN2000 平台下用标准 C 或 VC 实现。

为了满足播放的可靠性，存储播放子系统采用了四个存储播放服务器，每个服务器下挂两个 1080GB SCSI RAID0 级磁盘阵列。存储播放子系统的体系结构如图 2.7 所示。图 2.7 中，A 组内的存储服务器 A1 系统与存储服务器 A2 系统互为镜像，B 组内的系统也互为镜像。A1 与 A2，B1 与 B2 使用 RS232 的串口相互通信，传递同步信息来保持二者之间播出进程的帧同步。在这种结构下，镜像服务器间周期性地交换信息，并利用回应信息推断其镜像成员是否正常运行。一旦检测到故障时，立即通知总控进行故障切换。这种技术称之为心跳线技术。总控也通过网络监控和故障检测及时进行故障切换。

存储播放子系统首选的关键是高可靠、支持长期播出的高速大容量存储子系统。况且随着网络技术和影视高新技术的飞速发展，影视的网络化、数字化已成为世界影视业发展的必然趋势，因此将导致存储子系统在影视行业的广泛应用。下面将对其存储子系统进行详细地性能平衡设计。

2.3.2 存储系统的性能平衡

主机通道适配器和串控制器，是影响整个系统性能的主要因素。当然，CPU 对系统性能也有一定的影响。在其他资源不变的情况下，选择高档的串控制器，会提高系统的性能。但是，到达某一程度后，再高档的串控制器对系统性能也不会有多大的影响。一般来说，串内磁盘的数量增加会降低系统的性能，因此，串内磁盘增加到一定程度后就不能再增加了。串控制器的性能也直接影响串内磁盘的最佳数量，即串控制器性能与串中磁盘驱动器数必须要匹配。

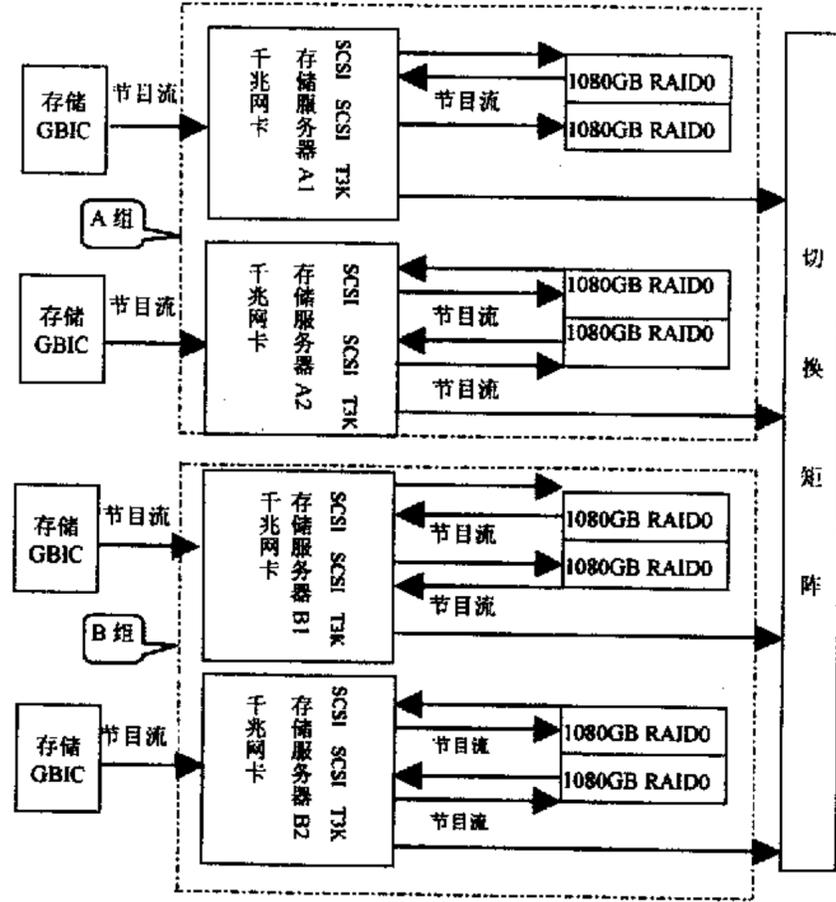


图 2.7 存储播放子系统结构图

(1) 串控制器与串中磁盘驱动器的性能平衡。为满足高数传率、海量存储的要求，存储子系统选用磁盘驱动器性能参数如表 2.2 所示的磁盘构成的 SCSI 磁盘阵列，阵列容量为 1080GB。其阵列结构是 3 串，每串三个盘。

表 2.2 磁盘驱动器性能参数表

磁盘容量 (C)	120GB
Cache	4096KB
主轴转速	10000rpm
平均寻道时间	5.2ms
平均旋转等待时间	2.99ms
平均通道时间	0.75ms
数据传输率 (M_d)	160MB/s

设 B_s 为 SCSI 总线选择时间； R 为磁盘驱动器平均寻道时间， L 为磁盘驱动器平均旋转延迟， T 为从磁盘驱动器传输数据到磁盘 Cache 的时间。假设每个磁盘驱动器的 I/O 数据大小一样为 0.5MB，并且服务时间也完全一样，那么要使串控制器连接的 n 个磁盘驱动器能连续不断地被启动，SCSI 总线选择延时必须满足下列不等式^[42]：

$$B_s \leq (R+L+T)/n,$$

此处 $R=5.2ms$, $L=2.99ms$, $T=0.5/160*1000=3.125ms$, $n=3$, 代入上式，则 $B_s \leq 3.77ms$ ，现在的快速 SCSI 适配器的总线设备选择时间均小于 200ns，因此要从 SCSI 适配器的 I/O 带宽的角度配置合适的 SCSI 串控制器。考虑到存储子系统到视频处理器之间的 I/O 路径上必须有 21MB/S 持续恒定的媒体流。因此要选择带宽至少为 40MB/S 的 SCSI 串控制器。

(2) 主机通道适配器与串控制器的性能匹配及阵列配置。为保证正常的读操作, SCSI 卡的 I/O 带宽必须大于阵列控制器从其主存储器读取数据的速度 D_R 。若阵列由 N 串组成, 则主机通道 SCSI 适配卡的数传率 D_{ar} 应满足:

$$D_{ar} \geq X \cdot N \cdot D_R$$

式中, X 为考虑串间并行性的系数, $X = 1-0.5$, 取 $X=0.5$, $D_R=21 \text{ MB/S}$, 则 D_{ar} 至少为 $0.5 \cdot 21 \cdot 3 = 25.5 \text{ MB/S}$, 考虑到 SCSI 通道利用率为 40%, 因此选取 80MB/S 的 SCSI 适配卡作主机通道适配器。

(3) 且因为 RAID0 与其他级别阵列如 RAID1、RAID3、RAID5 相比, 具有最好的读写性能和最大的存储容量^[42], 适合于具有可靠备份的高数传率环境。而视频播放服务器存储子系统在体系结构上具有镜像, 热备份功能, 并且它存储的数据并非要永久保存, 而是一星期左右更新一次, 因此数据存储子系统选用 RAID0 级阵列。

(4) 磁盘阵列中 PCI 总线与外围设备适配器的性能平衡。SCSI 磁盘阵列的串控制器是直接插在 PCI 总线上的, 多个串控制器连接在 PCI 总线上并发地进行传输, 假设 PCI 总线的利用率是 100%, 则阵列控制器中 PCI 总线带宽 D_{pci} 应满足:

$$D_{pci} \geq \sum_{i=1}^N D_i$$

其中 D_i 为外围设备 (磁盘串控制器, 网络接口) 的数传率, 则 HDDP-MSS 中 D_{pci} 至少为 $40 \cdot 3 = 120 \text{ MB/S}$, 考虑串控制器的并行开销, 及系统将来的可扩展性, 因此取 $D_{pci} = 264 \text{ MB/S}$ 。

2.4 存储和播放的可靠性分析

首先, 由以上存储播放子系统的性能配置过程可知, 存储播放子系统完全能满足其高数据传输率 21MB/S 的要求。

其次, 存储播放子系统由 8 个 1080GB SCSI RAID0 磁盘阵列构成的 A1 存储服务器, A2 存储服务器, B1 存储服务器, B2 存储服务器组成, 其中 A1 与 A2, B1 与 B2 互相镜像, 这样 $1080 \text{ GB} \cdot 4 = 4320 \text{ GB} = 4.21875 \text{ TB} > 4.037 \text{ TB}$, 即系统存储容量满足 (1) 式要求, 可预先存储电视台播放 7 天, 每天连续播放 17 小时的节目。满足了一般电视台用户的高容量要求。

并且, 因 A1 与 A2, B1 与 B2 互为热备份, 互有心跳线技术, 且它们分别下挂两个 RAID0 级阵列, 其系统可靠性模型结构如图 2.8:

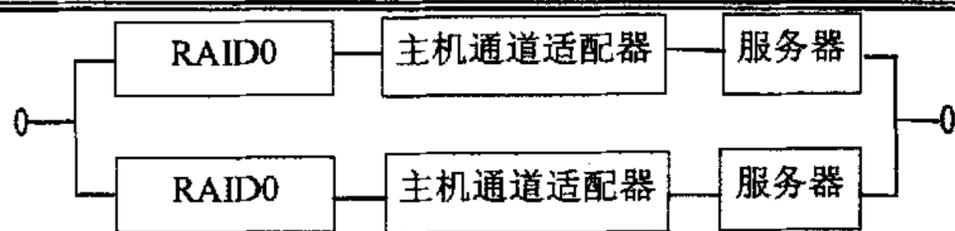


图 2.8 存储播放子系统可靠性模型

显然存储子系统是属于 n ($n=4$) 个不同单元组成的串联系统再并联的可修系统^[43-48], 假设各部件的失效率 λ_i 和修复率 μ_i 都是常数, 用 $x(t)$ 表示 t 时刻系统状态,

$$x(t) = \begin{cases} 0, & \text{在时刻 } t, n \text{ 个单元均正常} \\ i, & \text{在时刻 } t, \text{第 } i \text{ 个单元故障, 其余单元} \\ & \text{正常, 系统故障, 一修理工修理。} \end{cases}$$

设 $P_j(t) = P(x(t) = j) \quad j = 0, 1, 2, \dots, n$

则可得方程组 (2.4), 求解方程组, 可得串联系统的有效度为 (2.5) 式。

$$[P_0'(t) \quad P_1'(t) \quad \dots \quad P_n'(t)] = [P_0(t) \quad P_1(t) \quad \dots \quad P_n(t)] \times \begin{bmatrix} -\sum_{i=1}^n \lambda_i & \lambda_1 & \lambda_2 & \dots & \dots & \lambda_n \\ \mu_1 & -\mu_1 & 0 & \dots & \dots & 0 \\ \mu_2 & 0 & -\mu_2 & 0 & \dots & \vdots \\ \vdots & \vdots & 0 & \dots & \dots & \vdots \\ \vdots & \vdots & \dots & \dots & \dots & \dots \\ \mu_n & 0 & \dots & \dots & \dots & -\mu_n \end{bmatrix} \quad (2.4)$$

$$A_{\text{串联}}(t) = P_0(t) \quad (2.5)$$

当 $t \rightarrow \infty, P_j(t) \rightarrow P_j, P_j'(t) \rightarrow 0$ 时

串联系统处于平稳状态下的解为:

$$P_0 = (1 + \frac{\lambda_1}{\mu_1} + \dots + \frac{\lambda_n}{\mu_n})^{-1}$$

$$P_i = \frac{\lambda_i}{\mu_i} P_0 \quad i = 1, 2, \dots, n$$

串联可修系统的稳态可用度如式 (2.6) 所示:

$$A_{\text{串联}} = P_0 = \left[1 + \sum_{i=1}^n (\lambda_i / u_i) \right]^{-1} \quad (2.6)$$

因为服务器 A1 与 A2, B1 与 B2 互为热备份, 且相互独立, 并假定在很小时间间隔 ΔT 内 A1 与 A2, B1 与 B2 不会同时失效, 失效率 $\lambda_{\text{串联}}$ 和修复率 $\mu_{\text{串联}}$ 都是常数, 同样可得方程组 (2.7)。对上述方程进行拉氏变换, 并代入初始条件, 得当 $t \rightarrow \infty$ 时, 此并联可修热储备系统的稳态可用度为 (2.8) 式。

$$\begin{bmatrix} P_0'(t) & P_1'(t) & P_2'(t) \end{bmatrix} = \begin{bmatrix} P_0(t) & P_1(t) & P_2(t) \end{bmatrix} \times \begin{bmatrix} -2\lambda & 2\lambda & 0 \\ \mu & -(\mu + \lambda) & \lambda \\ 0 & u & -u \end{bmatrix} \quad (2.7)$$

$$P_0(0) = 1 \quad P_1(0) = 0 \quad P_2(0) = 0$$

$$A_{\text{并联}} = (2\lambda_{\text{串联}}\mu_{\text{串联}} + \mu_{\text{串联}}^2) / (2\lambda_{\text{串联}}^2 + 2\lambda_{\text{串联}}\mu_{\text{串联}} + \mu_{\text{串联}}^2) \quad (2.8)$$

而存储播放子系统各部件 (根据目前各厂商数据) 的参数如表 2.3 所示, 又

$$\lambda = 1/MTBF \quad (2.9)$$

$$u = 1/MTTR \quad (2.10)$$

由于 $MTBF_{\text{RAID0}} = \frac{MTBF_{\text{磁盘驱动器}}}{\text{RAID0 中的磁盘数 } n}$ [45], 将系统中磁盘驱动器的 MTBF 除以 9,

其余部件的 MTBF 与 MTTR 数据代入 (2.6)、(2.9)、(2.10) 可得串联可修系统:

$$A_{\text{串联}} = 0.9987873 \quad (2.11)$$

$$MTTR_{\text{串联}} = (4.5 + 0.5 + 2) / 3 = 2.3333 \quad (2.12)$$

表 2.3 存储播放子系统部件的 MTBF 与 MTTR

部件 参数	磁盘驱 动器	主机通道 适配器	服务器
MTBF (小时)	50,000	120, 000	5000
MTTR (小时)	0.5	0.5	2

据(2.3)、(2.9)、(2.10)、(2.11)、(2.12)有

$$\lambda_{\text{串联}} = 5.203572856 \times e^{-4}$$

$$\mu_{\text{串联}} = 0.42857142857$$

将 $\lambda_{\text{串联}}$ 、 $\mu_{\text{串联}}$ 代入 (2.8) 则有数据存储子系统的可用度:

$$A=0.9999970587=99.9997\%$$

此值完全满足 (2.2) 式的可靠性指标。

最后, 存储播放子系统体系结构与性能配置也可使存储容量随经济实力而定, 可大可小, 具有高可扩展性。如若向下扩展, 则或者每组去掉一台阵列, 或者将阵列容量减少, 或者去掉一个镜像服务器组, 随具体情况而定。如若向上扩展, 则其一, 可扩大阵列容量; 其二, 可每组增加磁盘阵列; 其三, 可再增加镜像服务器组; 均可满足要求。

2.5 运行效果

使用 Iometer 测试存储播放子系统读写的数传率, 平均可达 25.85MB/S。含有此存储播放子系统的数字播放系统, 运行标准清晰度电视的无压缩媒体流实现一路播出, 已经过用户近两年的实际运行, 运行结果表明系统满足了无压缩媒体流每秒 21MB 的数传率的要求, 实现了每秒 25 帧的无黑场播放; 运行标准清晰度电视的基于 mpeg2 压缩 50Mb/S 的媒体流, 同时进行一路采集两路播出, 也经过用户一年多的实际运行, 运行结果表明此存储播放子系统也能向下兼容压缩数字播放系统的要求, 实现每秒 25 帧的不掉帧播放。

对于 HDTV 高清晰度电视^[40]1920/1080/60frames/s MP@HL, 其水平方向 1920 像素, 垂直方向 1080 像素, 其无压缩媒体流的总的数据传输率约为 238MB/S, 压缩 12 倍后其数传率为 19.83MB/S。存储播放子系统满足了 21MB/S 的高数传率要求, 因此一旦电视转为高清晰度电视, 它也能很好地工作。

小结

高质量数字视频播放系统是针对当前的数字播放网络存在价格昂贵、编辑网与播放控制网分离、帧精确与延迟播放等关键技术需依靠硬件实现等缺点而提出的。该系统构建在千兆光纤网络平台上, 实现了编辑网和播放控制网的融合, 实现了整个系统的高度自动化管理, 实现了专门的上下载工作站录入节目, 实现了数据传输的高速化和全自动化。该系统既支持标准清晰度的无压缩视频流的播放, 又能支持基于 MPEG2

的压缩媒体流的多路采、编、播工作。而且用普通的服务器加视频卡组建视频服务器，降低了系统的整体价格；同时，将帧精确、延迟播放等关键技术应用于该系统，提高了产品的性价比。在相同性能条件下，它的价格仅是国外同类系统的五分之一，应该说这个系统是目前率先研制成功的一个有自己独立版权的无压缩视频播放系统。

体系结构是电视台数字视频播放系统最重要的内容之一，而存储播放子系统是整个系统的核心，它的体系结构设计至关重要。为满足高质量存储播放子系统的高可靠性、高数传率、海量存储容量和可扩展性的要求，本章在理论指导下，对存储播放子系统进行了性能平衡设计，并建立了它的可靠性模型，计算出它的可用度，其性能完全满足电视台停播率的要求。该设计方法与可靠性模型对设计存储播放系统和分析系统性能均具有指导意义。

3 视频播放系统中的并行调度策略

3.1 无压缩数字视频播放系统的关键技术

目前国内有多家影视科技公司如大洋、索贝、方正、索尼、北航、安徽现代、碧玺、安恒利、广迪威、品尼高、影佳科技等的数字化播放系统，它们都是以 MPEG2 作存储传输格式^[38, 39]。但是 MPEG 中的压缩方法，在源图像-压缩编码-传输-解码接收的链条中，后 3 个过程都是有损的。这将导致图像细节信息的丢失而产生压缩失真，使压缩图像的质量有所下降，或者说最好的结果是基本保持源图像的质量，即，不管压缩编码是多么的高级，所用的码率有多大，经压缩再解码后的图像的质量绝对不可能超过源图像质量。一般情况下，国内的电视制片要经过多级行政主管的审片，反复地压缩与解压缩必然引起节目画质的受损，因此研究基于无压缩媒体流的编辑与播放系统对提高电视台播放质量很有必要。而且电视一旦从标准清晰度电视转为高清晰度电视，现有的基于 MPEG2 压缩媒体流的数字播放系统的软硬件都得改变。无压缩数字视频播放系统中媒体文件存放格式是无压缩的，正好可以避免反复压缩与解压缩引起的图像质量损耗。

如第二章所述数字视频播放系统是一个千兆以太网系统。存储播放子系统是其核心部分。它存储节目到磁盘阵列，并从磁盘阵列中读取媒体流，送入视频卡，根据节目单准时播放节目。它除了要实现一系列自动化功能之外，其关键性能是必须实现节目播出时不掉帧、画面无黑场、无闪烁，且能按帧切换。传统的 PRE-CACHE 顺序存取策略不能满足其性能要求，因此本文提出了一种 MPPDIO(多线程并行预取直接 IO)存取控制策略，用以有效地控制媒体流的存取，实现连续播放的性能要求。下面将对这一策略进行详细描述。

3.1.1 多线程技术

多线程并不是一个全新的概念，它是数据流计算机研究的继续^[49]。传统上，计算模型被分成数据流和控制流两种模型，而多线程计算模型实际上是这两种模型的综合。换句话说，多线程模型是数据流模型和控制流模型这两极向中间发展的结果。对线程尚没有一个严格而统一的定义。从程序的角度来看，线程(thread)是一段顺序指令序列。线程分为就绪态、运行态和挂起态，与进程状态空间相同。一旦某个线程产生一个长时延操作，如访问存储器、处理机间通讯或长浮点运算等，该线程即被挂起，随即由调度器从线程池中选择一个就绪线程进入 CPU。这样，时延被隐藏

起来了。线程作为 CPU 调度的基本单位，子线程共享父线程的资源。进程可看作是由线程组成的，一个含有多线程的进程中，多个线程共享同一地址空间，所不同的只是私有栈和代码栈。线程的创建和调度成本大大低于进程，所以，线程有时也被称作轻进程 (light-process)。

多线程的思想最早在 B. Smith 设计的 Deneclor HEP 机器中率先采用，更早一点的历史可以追溯到六十年代 CDC660 0 机器中的多功能部件计分牌。一般认为，多线程的发展遵循两条途径：一是在冯诺依曼控制流结构上扩充对多线程并发的支持；二是在数据流结构上扩充顺序执行的线程。这种结构被称为混合结构 (Dataflow/ Von Neumann Hybrid)。MIT 研制的多线程系统如 P- RISC 和 Empire 等等大多属于这种类型。从体系结构分类上看，多线程计算机属于多指令流多数据流计算机(MIMD)。多线程内部按控制流驱动方式顺序执行，而线程间则是按数据流方式调度和执行。因此，在多线程内部和多线程间都能开发大量的并行性，最重要的是对一些长时延操作对系统性能的影响基本消失。多线程与其他减少时延的方案相比，优势十分明显。多线程技术通过隐藏时延，提供完全用户级软件兼容，以此来达到整体性能和效率的提高，为高性能和高可用性这一对矛盾的解决展现了希望。

在 UNIX 进程里，能够被分割成独立处理的单元叫线程 (Thread)，它是进程内的一连串指令的执行，一个进程里可以有多个线程，它们之间共享全部的进程地址空间和进程资源。用户可以按需要编写拥有多个线程的程序，从理论上说，用户进程可以创建任意多个用户线程。这些线程可以在不同的处理机上同时执行。但从资源分配、处理机调度、线程切换、效率等诸多问题出发，并不是每创建一个线程，就作为一个独立的单位，交给核心管理^[40]。在多线程结构的操作系统中，多线程结构大都分成两个层次来实现，一层是用户层，在用户级多线程库中实现，一层是核心层，在操作系统内核中实现。为了区别，处于不同层次中的线程分别叫做用户线程和核心线程。用户线程只是一个代表对应线程的数据结构，它只占用用户空间的资源，是用户级的对象，对核心是透明的。而核心线程才是线程执行的实体，是

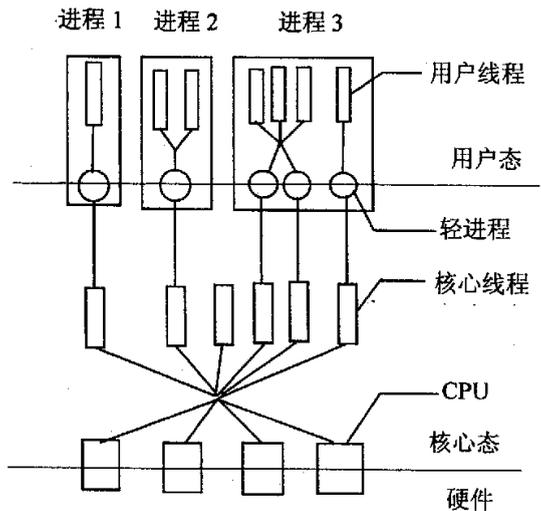


图 3.1 多线程结构系统示例

核心线程才是线程执行的实体，是

核心调度的单位。用户线程和核心线程并不是一一对应的，它们之间要通过一个新的对象来联系，在 Solaris 操作系统中，这个新的对象叫 LWP(Light Weight Process)，即轻进程。LWP 如同用户线程的虚拟处理机，与用户线程一样共享进程中的所有资源。LWP 总是按一定的策略，从用户线程中选择一个执行，而 LWP 对核心是可见的，是核心级的对象，它与一个核心线程一一对应，核心按一定的策略，选择一个核心线程运行。因此，一个线程要经过核心和 LWP 的两级调度后才能真正占有 CPU，开始运行。LWP 的引入使得用户级的多线程能浮在核心之上实现。

每个进程可以创建几千个线程，而不需要占用核心资源，只占用用户空间的资源。线程由于共享地址空间，当 LWP 在一个进程的不同用户线程之间进行切换时，时间开销远远低于两个核心线程之间的切换时间，因为如图 3.1 多线程结构系统例程的用户空间是共享的，仅仅是一种数据结构的转换，并且这种切换完全在用户级的线程库中完成，对核心是透明的。线程间的同步也可以独立于核心的同步对象，在用户空间中独立实现，而不用陷入内核，在核心对象上实现同步。用户线程对核心来说是不可见的，核心能看见的只是 LWP。因此，多线程结构有两层模式，第一层为线程接口，提供给用户，使用户能编写多线程的应用程序。这一层由动态链接库利用轻进程 LWP 来实现，线程库在进程的 LWP 池上调度线程，线程之间的同步也在线程库中实现。第二层为 LWP 接口，提供给线程库，用于管理 LWP，这一层由核心通过核心线程实现，LWP 通过这些接口访问核心资源。典型的多线程结构的系统如图 3.1 所示。

图 3.1 中，进程 1 为传统的单线程结构的进程，进程 2 和进程 3 都是多线程结构的进程。每个用户进程有一个对应的 LWP 池。用户级的线程在用户级调度，被线程库指派到其进程的任意 LWP 中。用户线程之间的切换在用户级线程库中进行，不用进入内核。用户线程也能与某一个 LWP 捆绑在一起，占有对应的 LWP，在此线程退出之前，对应的 LWP 不能调度其它的用户线程运行。每个 LWP 与一个核心级的线程相对应。除此之外，还有一些系统核心线程，完成系统功能，不与任何 LWP 相关联，核心按照 LWP 的优先级，调度核心级线程到某一个 CPU 上运行，也可以把某个核心线程与某个 CPU 捆绑在一起，使它不运行别的核心线程。用户级线程之间的同步在线程库中实现，不用进入内核，核心线程之间的同步在内核中实现。用户线程调度在 LWP 池中实现。每个线程都有一个从 0 到无穷大的一个调度优先级，LWP 总是从最高优先级的就绪队列中调度一个用户线程运行。当某一个线程就绪时，它插入到对应的就绪队列中，并唤醒 LWP 池中的一个空闲 LWP，去选择一个最高优先级的线程运行。如果 LWP 池空则在就绪队列中等待。线程调度允许抢占，如果就绪线程的优先级比活动队列中任一线程的优先级高，则要实施抢占。从活动队列

中找到一个优先级最低的线程，向所对应的 LWP 发信号，LWP 接收到信号后，重新调度优先级最高的线程，则原来的线程被抢占。当正运行的线程被某一个同步变量阻塞，或退出，或被暂停时，要释放对应的 LWP，则该 LWP 从就绪队列中调度新的线程去运行，如果没有可运行的线程，则等待在 idle 变量上，放入 LWP 池中睡眠。

用户线程同步机制有以下几种：①互斥锁：一次只有一个线程能获得锁，常用于临界区的互斥执行；②条件变量：用于使一个线程等待某一个条件为真，必须与互斥锁一起使用；③信号灯：信号灯是一个非负的整数计数器，当一个线程进入时其值减少，当一个线程退出时其值增加；④多读者，单写者锁：允许多个线程同时读某个共享对象，但当一个线程要写该对象时，要阻塞读线程。同步机制可用于同一进程的各个线程之间，此时同步变量在进程的常规内存空间中分配，对核心是透明的，每个同步类型都支持几种实现方法，程序员在变量初始化时选择实现方法。同步机制也可用于不同进程的各个线程之间，在变量初始化时标志为共享，在共享的映射文件中分配。当线程被阻塞时，核心对其处理，挂在核心的睡眠队列中，此时线程同执行系统调用一样，与 LWP 捆绑在一起，LWP 不能去执行其他的线程。

3.1.2 MPPDIO 存取控制策略

3.1.2.1 无压缩数字播放系统的播出工作特性分析

如图 3.2 为无压缩播放系统播出部分系统结构图，其播出的 I/O 路径是：CPU 接到播放命令及播放节目单后，通过系统内存总线、PCI 桥、PCI 总线、SCSI 适配器等从磁盘读取节目文件到系统内存，然后传送到视频卡内存，由视频处理器处理后经外部接口箱输入到电视机中进行节目播放。其媒体流特性，如 2.3 节所述，PAL 制式标准清晰度数字电视无压缩媒体流

的总数据率为 20.928MB/s，要实现播出时的不掉帧^[50-53]、无黑场，就必须保证从磁盘到视频处理器之间的 I/O 路径上至少有 21MB/S 持续的媒体流。

分布式多机系统中存在着严重的网络延迟^[54]。(1) 在多机系统中要完成一个计算任务，必须把一定数量的数据传输给其它结点，因此要频繁进行消息传递和

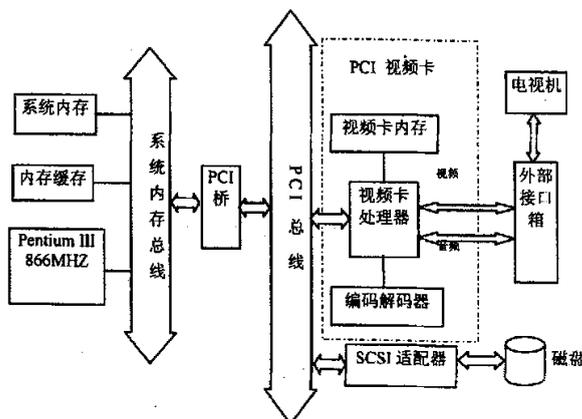


图 3.2 无压缩播放系统播出部分系统结构图

同步操作。(2) 另外, 由于在多机系统中, 各节点是分布式的, 所以在 Cache 发生远程失效时也要访问网络, 这就进一步增大了网络通信量。(3) 所有这些都使得网络变得拥挤, 从而导致了很长的通信延迟。由于这些通信操作发生得相当频繁, 因此处理器 (经常还包括网络) 不得不经常忍受很长的空闲时间以等待所请求的数据的到达, 这就在很大程度上降低了对处理器资源的有效利用, 进而导致多机系统性能的大幅度下降, 有研究表明, 网络延迟在严重时将会使系统的整体性能下降 60% 以上。为了尽可能地提高处理器和网络的利用率、提高应用程序的实际执行效率, 因此非常需要有合适的延迟容许机制来减少或隐藏这些延迟所带来的损失。多线程技术是解决上述问题的一个有效的方法。(4) 多线程技术就是允许在一个处理机上同时运行两个或两个以上的线程, 此时, 当一个线程向远地结点发出数据请求时, 处理器就可以切换到另一个线程去执行, 不必再等待前一线程需要的数据的到达, 这不但大大提高了处理器的利用率、更充分地利用了互连网络硬件本身所提供的高带宽通信能力, 而且还通过计算与通信的重叠使通信的延迟得以掩盖, 从而使多机系统的整体性能得到了很大的提高。而无压缩数字视频播放系统实质是一个分布式多机系统。因此我们也用多线程 IO 技术来实现无压缩数字视频播放系统的关键性能。

3.1.2.2 MPPDIO 算法实现

MPPDIO 算法的逻辑示意图如图 3.3 所示。为了保持从磁盘阵列到视频处理器之间 I/O 路径上持续 21MB/S 的无压缩视频流, 为了实现播放的无帧丢失, 无黑场, 算法采用 Direct IO^[55] 内存映射文件方式、共享缓冲池及多线程并行预取策略。其中 Direct IO 直接将磁盘文

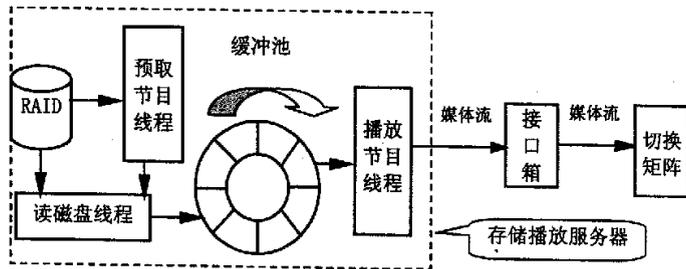


图 3.3 MPPDIO 算法逻辑示意图

件数据从主机内存映像到视频卡内存, 缩短了媒体流从系统内存到视频卡内存的 I/O 路径所需时间, 提高了系统性能; 这允许以无文件系统缓冲方式访问文件数据, 最大化了磁盘到主机内存之间的数据流速。

(1) 多线程并行预取

将 I/O 路径上从磁盘阵列预读文件句柄与计算起始位置、读取音视频数据到主机内存及从主机内存读取音视频数据播放的三过程分别以预读线程、读磁盘线程与播放线程实现, 使它们并行工作, 当播放时, 同时进行预读文件句柄、读音视频数据到主

机内存, 这样可隐藏预读文件句柄、读音视频数据的时延, 进一步缩短系统总的 I/O 路径长度以实现播放的无黑场、无失帧^[56-62]。如图 3.4 所示的多线程并行预取策略被采用在 MPPDIO 算法中。

设信号量 A、B、Is-next, 分别表示共享缓冲区中可用的空位数、可供播放的音视频帧数及系统是否播下一节目; 初始时, $A=n$ (假设缓冲区最多能存放 n 帧音视频), $B=0$, $Is-next=0$ 。

这里读磁盘线程相当于生产者进程, 它首先从磁盘中读取一帧音视频, 做 P 操作使 $A=A-1$, 若 $A>=0$, 表示缓冲区有空位, 则存放此帧音视频到缓存, 否则等待, 直到有空位。接着做 V 操作使 $B=B+1$, 通知播放线程, 已有音视频数据可播。播放线程相当于消费者进程, 它首先做 P 操作使 $B=B-1$, 若 $B>=0$, 表示缓存中有音视频数据, 则从中取出一帧音视频, 否则等待, 直到 $B>=0$ 。接着做 V 操作使 $A=A+1$, 通知读磁盘线程已有一空位腾出, 然后播放此帧音视频。预取线程当 Is-next 为真时, 读文件句柄计算播放起始位置并传送给读线程。

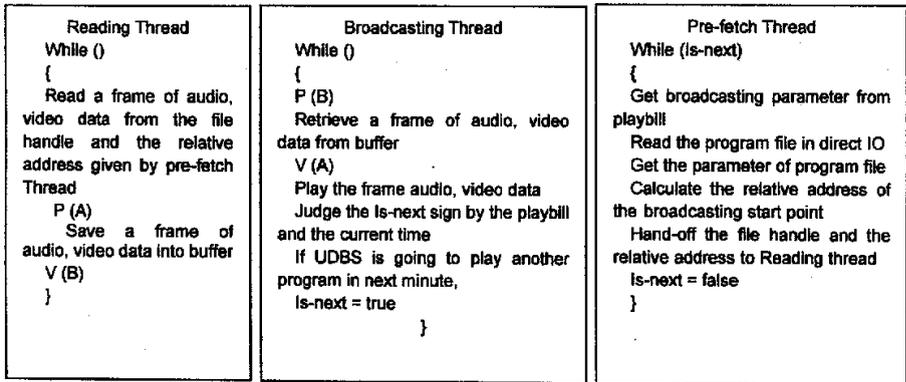


图 3.4 多线程并行预取策略

从图 3.4 可看出, 在 MPPDIO 算法中有两个层次上的预取。第一层是从磁盘阵列预取 m 帧媒体流到主机 buffer 的预取, 以缓冲读磁盘线程与播放线程之间的速度差异, 保证稳定输出, 它是普通的。第二层预取是预取线程读文件句柄、计算播放起始地址并只传送句柄与起始地址给读磁盘线程。它不是预读一段节目、将节目内容传给读磁盘线程, 而是只取句柄与播放起始地址给读磁盘线程。这节省了读一段文件时间、传送一段文件时间、而且因为避免了线程间文件拷贝而节省了内存空间。

(2) Buffer 的建模与计算

如图 3.3 所示, 由于磁盘数传率是随机的, 因此 MPPDIO 算法采用共享 buffer 来补偿读磁盘线程与播放线程之间的速度差异。那缓冲池应设置多大, 至少要预取多少

帧媒体流才能满足系统无黑场、无掉帧的播放性能要求，本文将就多媒体同步与表现的有关理论^[63-71]，分析其量化关系。为分析方便先引入两个定义。

同步时间单位 (τ): 用来确定媒体流间同步的最小时间单位。同步时间单位不仅决定了媒体流同步粒度的大小，而且决定了系统同步的精度。因此同步时间单位的取值应在同步质量与实现难易之间进行权衡。无压缩数字播放系统要实现精确到帧的播放质量，因此取播放一帧的时间为同步时间单位，即 $1/25=0.04$ 秒。

媒体流同步粒度 (Media Stream Synchronization Granularity, MSSG): 指同步时间单位内包含的媒体流信息。本系统取大小为一帧的音视频流为媒体流同步粒度。

又由于 MPPDIO 存取控制策略采用 Direct IO^[55] 内存映射文件方式，直接将磁盘文件数据从主机内存映像到视频卡内存由视频处理器进行处理，数据从缓冲池到视频处理器的时间与磁盘的存取延时相比可忽略不计，因此图 3.5 所示的接收方与处理部分可合并看作是接收方，以便分析。

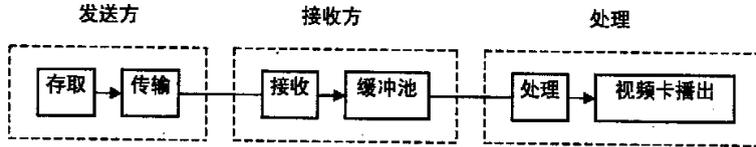


图 3.5 播出部分 IO 路径处理过程的抽象模式

从发送方到接收方的 MSSG 同步示意图如图 3.6 所示，其中 t_1, t_n 为发送方发送 MSSG₁ 与 MSSG_n 的时间， T_i, T_i' 为接收方收到 MSSG_i 的最早时间与最晚时间， ΔT_{\min} ， ΔT_{\max} 为一个同步粒度 MSSG 的最小存取延时与最大存取延时时间。从图 3.6 可知，

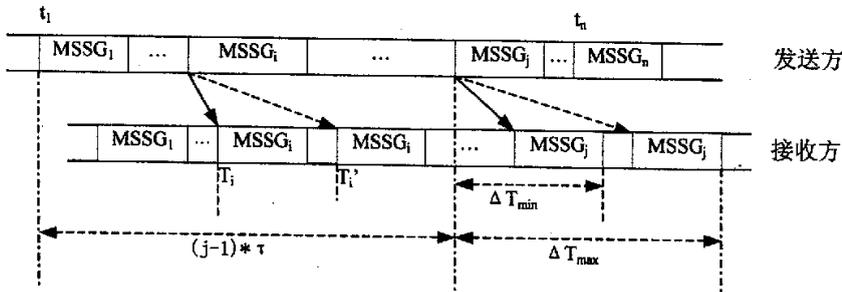


图 3.6 考虑存取延时的 MSSG 同步示意图

同步粒度 MSSG_i 的最早播放时间为：

$$T_{pl}_{first}^i = t_1 + (i-1) \times \tau + \Delta T_{\min}$$

最晚播放时间为：

$$T_{pl}_{latest}^i = t_1 + (i-1) \times \tau + \Delta T_{\max}$$

同步粒度 MSSG_i 最迟到达接收方的时间为

$$T_{latest}^i = t_i + \Delta T_{max}$$

若要保证无黑场播放则必须满足

$$T_{latest}^i \leq Tpl_{first}^i \tag{3.1}$$

$$\text{即 } t_i + \Delta T_{max} \leq t_1 + (i-1) \times \tau + \Delta T_{min} \tag{3.2}$$

整理 (3.2) 式可得:

$$t_i \leq t_1 + (i-1) \times \tau - (\Delta T_{max} - \Delta T_{min}) \tag{3.3}$$

(3.3) 式表明在系统没有缓冲预取的情况下, 为满足连续播放, 则从磁盘读取第 i 个同步粒度的时间不得大于

$$t_1 + (i-1) \times \tau - (\Delta T_{max} - \Delta T_{min})$$

如果系统采用预取缓冲技术, 在开始播放前预取 m 帧音视频流, 则 (3.3) 式应修改为:

$$t_i \leq t_m + (i-1) \times \tau - (\Delta T_{max} - \Delta T_{min}) \tag{3.4}$$

假设 $i=m+1$, 代入 (3.4) 式, 又由于 t_m 是单调递增的, 则有

$$m \geq (\Delta T_{max} - \Delta T_{min}) / \tau \tag{3.5}$$

即, 为保证连续平稳播放, 则在系统中至少预取缓冲 $[(\Delta T_{max} - \Delta T_{min}) / \tau]$ 帧音视频。

3.1.3 MPPDIO 算法与 PRE-CACHE 顺序存取算法的 I/O 性能比较

如图 3.7 所示, 假设一帧音视频数据从磁盘到系统内存总线 I/O 路径上所花费时间为 T_1 , 从内存总线到文件缓存所花费时间为 T_2 , 从内存总线到进程地址空间所花费时间为 T_3 , 从进程地址空间到视频卡内存所花费时间为 T_4 , 从视频卡内存到外部接口箱的输出花费时间 T_5 ^[82-85]。若用 PRE-CACHE 顺序算法^[86], 因大块数据分布在整个磁盘的各地地方, 而 PRE-CACHE 顺序算法总是读出连续的数据块, 所以经常错误地读出一些不需要的数据, 使缓存不命中, 这样使得读取过程既要在缓存中搜寻, 若不命中, 从磁盘读, 又要往缓存中写缓存数据, 假设在缓存中搜寻的时间为 T_s , 则采用 PRE-CACHE 顺序算法系统读取 N 帧音视频数据所耗费时间可近似为

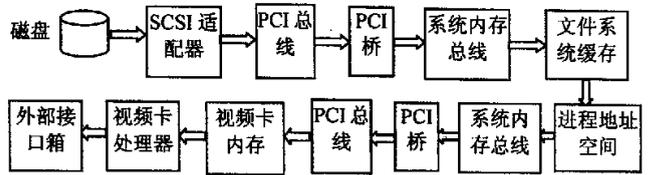


图 3.7 采用 PRE-CACHE 顺序存取策略的 I/O 路径

若用 PRE-CACHE 顺序算法^[86], 因大块数据分布在整个磁盘的各地地方, 而 PRE-CACHE 顺序算法总是读出连续的数据块, 所以经常错误地读出一些不需要的数据, 使缓存不命中, 这样使得读取过程既要在缓存中搜寻, 若不命中, 从磁盘读, 又要往缓存中写缓存数据, 假设在缓存中搜寻的时间为 T_s , 则采用 PRE-CACHE 顺序算法系统读取 N 帧音视频数据所耗费时间可近似为

$$T_r = n * (T_1 + T_s + T_2 + T_3 + T_4) = n * T_{RB} \tag{3.6}$$

播放 N 帧音视频数据所耗费时间为 $T_p = n * T_5$ ，这样采用 PRE-CACHE 顺序算法时，播放 N 帧媒体流总共花费的时间为

$$T = T_R + T_p = n * (T_{RB} + T_5) \quad (3.7)$$

系统连续播放两个有 N 帧媒体流节目所花费的时间为 $2T$ ，系统 I/O 服务时序图见图 3.8。

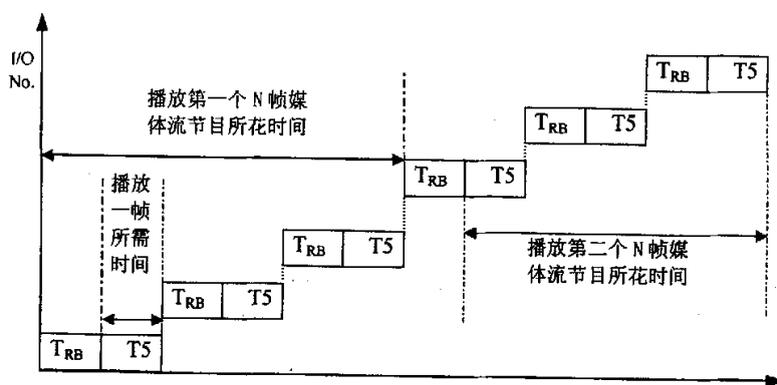


图 3.8 采用传统 pre-cache 顺序存取算法时播放两个 N 帧媒体流节目的系统 I/O 时序图

从图 3.8 可看出，采用传统的 PRE-CACHE 顺序存取算法播放节目，在一个节目内与两个节目间都有明显的黑场，不能满足播放性能要求。

而采用 MPPDIO 存取算法后，系统以 NO BUFFER 方式访问磁盘文件数据，这使得系统不必花费时间 T_2 去写文件缓存，花费 T_3 时间去搜寻文件缓存；以内存映射文件直接将媒体流数据从系统内存映射到视频卡内存，设此过程所花时间为 T_4' ，这里 $T_4' \ll T_4$ 可忽略不计；假设 T_{OP} 是打开一个文件获得文件句柄的时间， T_{TP} 是传输文件句柄与相对地知道读线程的时间， T_{CA} 是计算播放起始点相对地址的时间，因此预取下一节目总共花费的时间为：

$$T_{PF} = T_{OP} + T_{CA} + T_{TP}$$

读取 N 帧媒体流的时间为：

$$T'_R = n * (T_1 + T_3) = n * T'_{RB}$$

很明显 $T_{PF} < T'_{RB} < T_{RB}$ 。系统将读磁盘文件、播放节目与预取节目以线程的形式并行执行，因此则 T_{PF} 能被 T'_{RB} 隐藏， T'_{RB} 能被 T_5 隐藏。假设 t_m 表示预取 m 帧媒体流到共享缓冲池的时间，因此连续播放两个 N 帧媒体流节目所耗费的总的的时间为 $t_m + 2n * T_5$ 。

系统 I/O 服务时序图见图 3.9。从图 3.9 可以看出在一个节目内与两个节目间均无

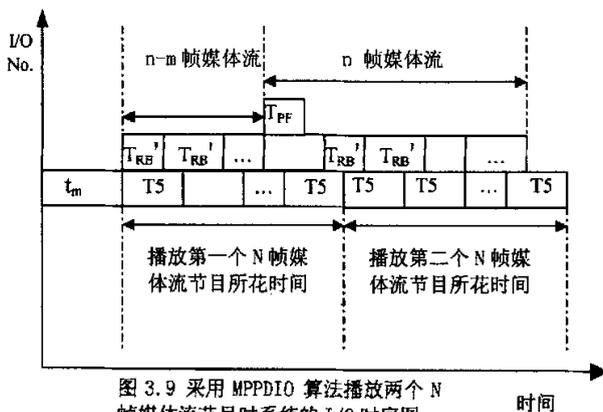


图 3.9 采用 MPPDIO 算法播放两个 N 帧媒体流节目时系统的 I/O 时序图

播放黑场。

播放 K 个 N 帧媒体流节目，采用传统 PRE-CACHE 顺序存取算法时，系统吞吐率为

$$TP_{PRE-CACHE} = kn/kT = 1/(T_{RB} + T5) \quad (3.8)$$

采用传统 MPPDIO 算法时，系统吞吐率

$$TP_{MPPDIO} = kn/(t_m + kn * T5) \quad (3.9)$$

因为 T5 是播放一帧 PAL 制式媒体流的时间，T5=40MS，因此从 (3.6) 式与 (3.8) 式可看出：

$$TP_{PRE-CACHE} < 25 \text{ frames / s}$$

它表明采用传统 PRE-CACHE 顺序存取算法时，系统有播放黑场有掉帧。而从 (3.9) 式可看出，当 $n \rightarrow \infty$ 时，

$$TP_{MPPDIO} = 25 \text{ frames / s}$$

即经过初始预取时间 t_m 后，系统吞吐率为 25 帧/秒。采用 MPPDIO 算法后系统加速比为：

$$speedup = (kn * (T_{RB} + T5)) / (t_m + kn * T5)$$

这里 $n \gg m$ 且 $n * T_{RB} \gg t_m$ 所以加速比

$$speedup \gg 1.$$

从以上分析可看出，在磁盘物理特性相同的情况下，采用 MPPDIO 存取算法使得总的 I/O 路径、I/O 时间缩短，能实现系统每秒 25 帧平滑无黑场播出；而传统的 PRE_CACHE 顺序存取算法却不能实现无黑场播放。与传统的 PRE_CACHE 顺序存取

华中科技大学博士学位论文

算法相比, 采用 MPPDIO 存取算法可使系统吞吐率提高, 且系统加速比远大于 1, 这是因为系统不必花费时间 T2 去写文件缓存, 且只要共享缓冲区有空位, 读磁盘线程就不停地发出读请求, 以最充分地利用磁盘的 I/O 通道。使磁盘 I/O 与播放过程并行, 能充分利用 CPU 资源。当缓冲区满时, 读磁盘线程又 Sleep, 这样又减轻了 CPU 负载的缘故。

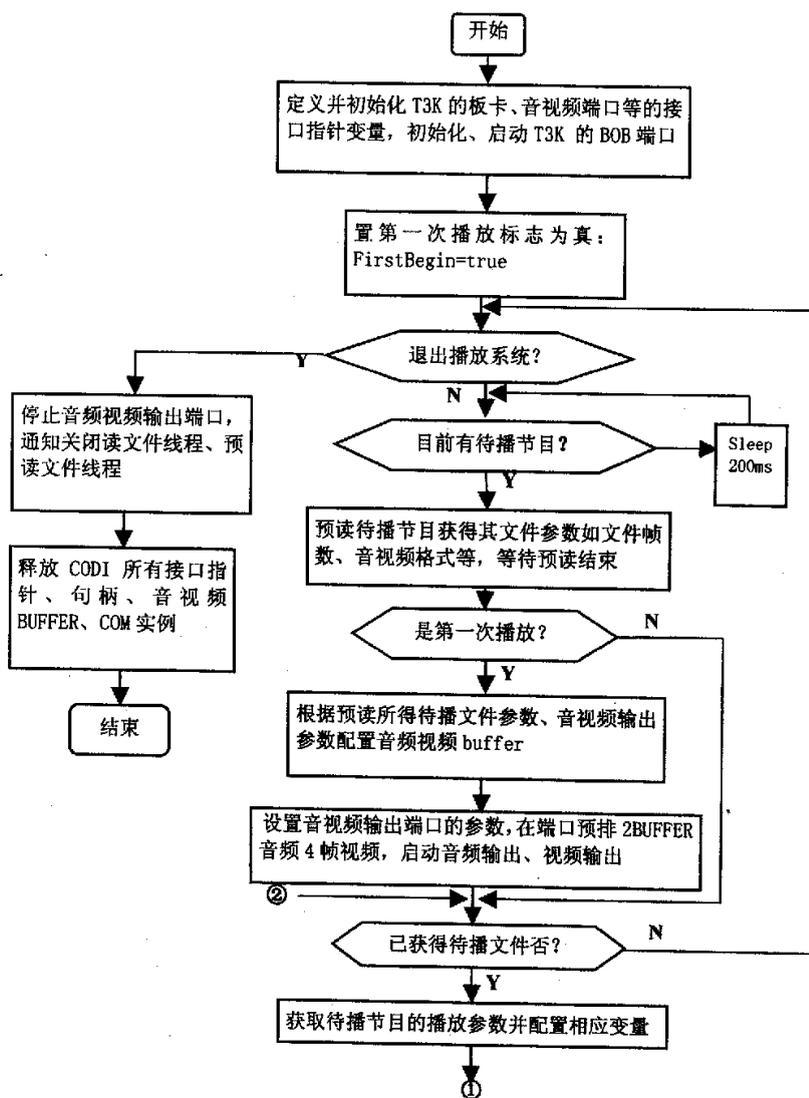
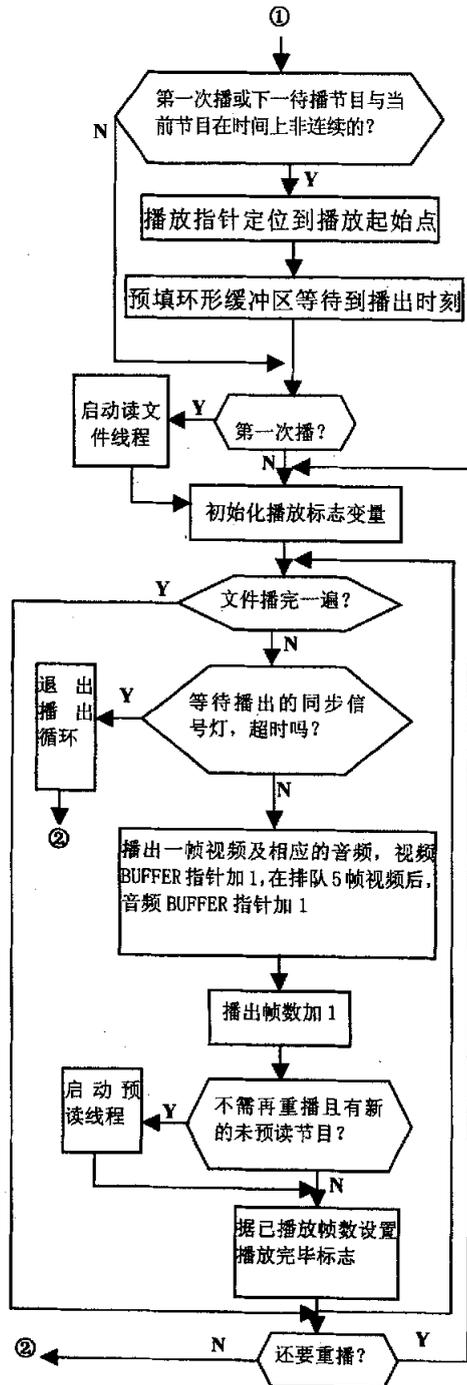


图 3.10 按节目单播放多节目的播放线程



3.1.4 实验验证

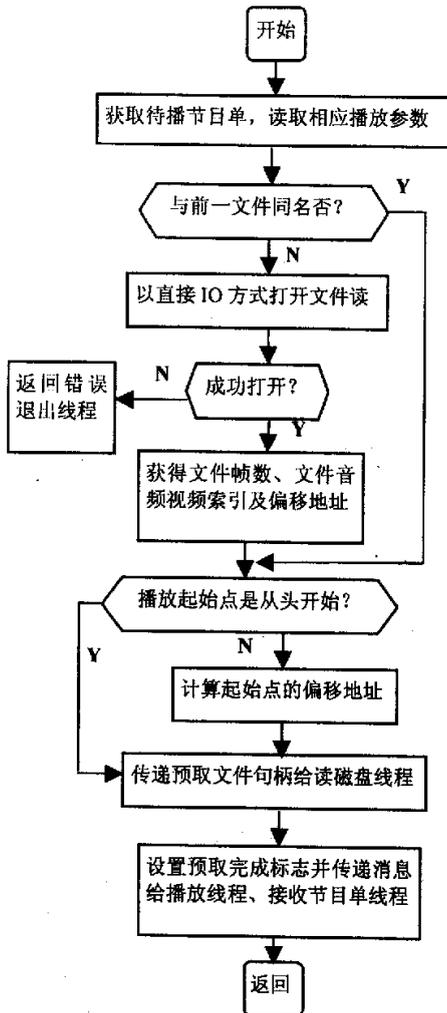


图 3.11 预读线程

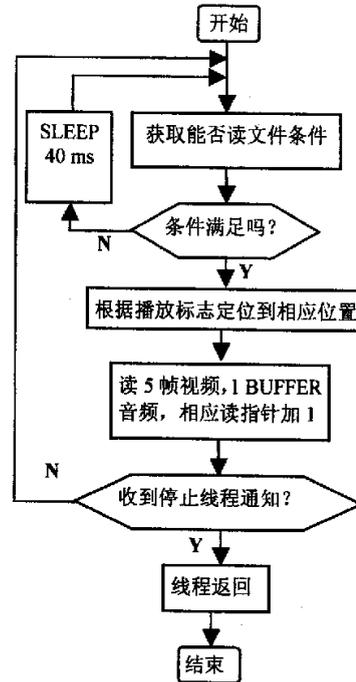


图 3.12 读磁盘线程

3.1.4 实验验证

实验平台如图 3.3 所示, 其中播放服务器配置为 PIII866MHz/256M RAM/18G HDD/AGP/15" Monitor 的 HP 服务器及视频卡构成, 所用磁盘的磁盘参数如表 3.1 所示。播放服务器通过千兆交换机接收来自主控工作站的命令消息; 当它收到主控工作站发出的待播节目单时, 就读取磁盘中的媒体文件, 通过视频卡严格按节目单播出。采用了 MPPDIO 存取控制策略的播放服务程序运行在播放服务器上; 读磁盘文件采用

华中科技大学博士学位论文

NO BUFFER 的 Direct I/O 方式；多线程并行执行的程序流程图如图 3.10、3.11、3.12 所示。

(1) 根据同步时间 τ 的定义，我们取 $\tau=40\text{ms}$ 。而如前所述，媒体流同步粒度指同步时间单位内包含的媒体流信息，本系统取一帧的音视频信息为媒体流同步粒度。而磁盘驱动器 IO 响应时间 T 是指从 IO 请求到 IO 服务结束为止的时间，它由排队等待时间，磁盘驱动器服务时间，和磁盘适配器接口传输时间等部分组成，分别用 T_{Que} 、 T_{D} 、 T_{I} 表示。其中磁盘驱动器服务时间又由寻道时间、旋转延迟、数据传输时间组成，分别用 T_{s} 、 T_{r} 、 T_{t} 表示。并且磁盘数据块大小用 B (字节) 表示，一次 IO 请求的长度用 L (字节) 表示，因此

表 3.1 Quantum ALAS10K2-TY184L 磁盘性能参数

磁盘容量 (C)	18.2GB
主轴转速	10000rpm
最大寻道时间	10ms
最大旋转等待时间	6ms
内部传输率	215Mb/s to 315Mb/s
磁盘 cache	2MB

$$T = T_{\text{Que}} + T_{\text{D}} + T_{\text{I}}$$

$$\text{即 } T = T_{\text{Que}} + (L/B) * (T_{\text{s}} + T_{\text{r}} + T_{\text{t}}) + T_{\text{I}} \quad (3.10)$$

这里 L 为一个 MSSG 同步粒度大小即一帧的音视频信息，即

$$L = (20.928\text{MB/S}) / (25\text{frame/S}) * (1\text{frame}) = 0.83712\text{MB} = 857.211\text{KB}$$

而 ΔT_{min} 、 ΔT_{max} 为一个同步粒度 MSSG 的最小存取延时与最大存取延时时间。若 $B=32\text{KB}$ ，则据以上 L 的取值及表 3.1 数据，由 (3.10) 式可推出 (3.11) 式、(3.12) 式。

$$\Delta T_{\text{max}} = T_{\text{Que}} + 27 * (16 + 31.15) + T_{\text{I}} \quad (3.11)$$

$$\Delta T_{\text{min}} = T_{\text{Que}} + 27 * 21.26 + T_{\text{I}} \quad (3.12)$$

为简化分析，设每次 IO 请求的 T_{Que} 、 T_{I} 相等。则由 (3.5) 式、(3.11) 式、(3.12) 式可得出：

$$[699/40] = 18 \leq n$$

即缓冲区最少应设置 18 帧。

表 3.2 不同 buffer 大小时，播放一个长度为一千帧的节目时的掉帧率

Buffer 大小 (帧)	4	8	16	18	32	36	64
掉帧率 (%)	8.07	0.75	0.094	0	0	0	0

我们在采用了 MPPDIO 存取控制策略的播放服务程序中改变 buffer 大小，并在播放节目的过程中记录相应的掉帧情况，测试结果如表 3.2 所示。从表 3.2 可以看出，

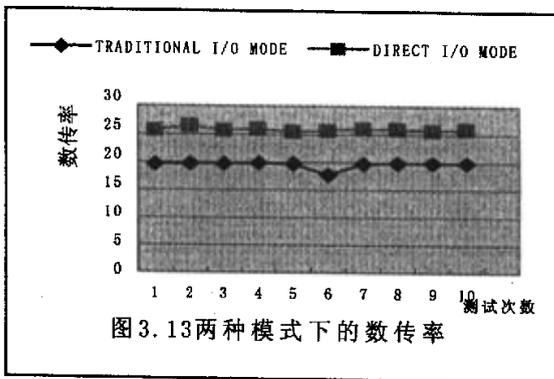
华中科技大学博士学位论文

当 buffer 大小取大于等于 18 帧时, 播放效果良好, 掉帧率为零, 与前面分析相符。

(2) 运行 IOMETER 测试磁盘 100%读操作时的数传率; 然后在播放服务器上运行 Direct I/O 方式读磁盘服务程序, 用 Windows2000 管理工具中的性能测试工具, 测试 Disk Read Bytes/Sec 以获得在 Direct I/O 方式下的实际数传率。测试结果如表 3.3 及图 3.13 所示: Direct I/O 方式下的实际数传率比传统方式下的数传率有显著提高, 平均数传率增加了 32.55%。这是因为 Direct I/O 的特点是读写磁盘时不用文件系统 cache, 直接从磁盘读入主机内存或从主机内存直接写入磁盘; 且本系统中磁盘上的媒体文件是以 Direct I/O 方式写入的, 所以在本系统中采用 Direct I/O 方式存取媒体文件比传统 I/O 方式更合适。

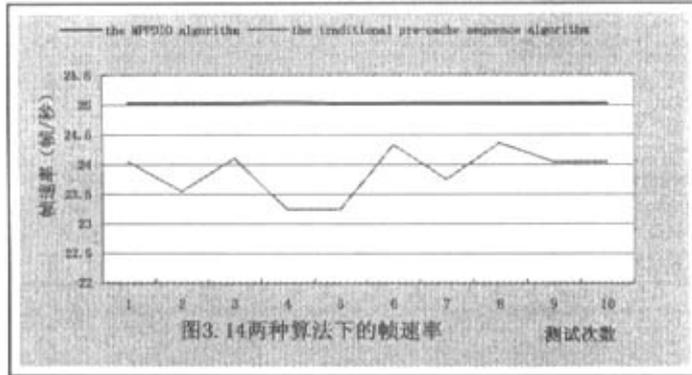
表 3.3 两种 I/O 方式的数传率测试结果

IO 方式 顺序	传统 I/O 方式的 数传率 (MB/S)	DIRECT I/O 方式的 数传率 (MB/S)
1	19.56	25.71
2	19.65	26.40
3	19.70	25.71
4	19.73	25.90
5	19.65	25.43
6	17.71	25.71
7	19.69	26.01
8	19.74	25.88
9	19.73	25.71
10	19.76	25.90
mean	19.492	25.836



(3) 在采用 MPPDIO 存取算法与传统的 PRE_CACHE 顺序存取算法的播放服务程序中记录播放开始到结束播放的时间 T (浮点型) 及播放帧数 FrameNum (整型),

用 $\text{FrameNum}/T$, 以获得两种方式下实际的帧播放速率。测试结果如图 3.14。由图 3.14 可知, 采用 MPPDIO 存取策略后的媒体输出流更平稳, 播放质量更好, 不掉帧。播放的平均帧速率为 25.0120 帧/秒, 为标准帧速率 25 帧/秒提供了保证, 而传统的 PRE_CACHE 顺序存取方式则由掉帧不能满足要求。实验结果与前面的分析相符。



(4) 用 Windows2000 管理工具中的性能测试工具, 测试 MPPDIO 存取算法与传统的 PRE_CACHE 顺序存取算法下的程序所占 CPU 时间的百分比 %Processor Time 及磁盘驱动器忙于为读请求提供服务所用时间的百分比 %Disk Read Time。测试结果如表 3.4、图 3.15 及 3.16 所示。

表 3.4 并行与串行方式下服务器性能指标测试结果

方式 顺序	并行方式			串行方式		
	I/O 通道 利用率 (%)	磁盘读速 率 (MB/S)	CPU 占 用率(%)	I/O 通道利 用率(%)	磁盘读速 率 (MB/S)	CPU 占用 率(%)
1	58.371	20.73	1.563	56.405	19.90	4.688
2	58.371	20.73	1.563	55.605	20.73	3.125
3	53.915	20.73	1.563	52.651	20.73	3.125
4	53.707	20.73	3.125	44.613	20.73	6.25
5	77.923	21.56	3.125	63.05	20.73	1.563
6	55.289	21.56	1.563	60.32	21.56	4.688
7	67.477	19.90	1.563	53.173	20.73	3.125
8	50.75	21.56	3.125	46.22	20.73	6.25
9	87.463	24.05	6.255	53.803	18.24	1.563
10	89.952	22.39	3.125	57.384	20.73	3.125
mean	65.322	21.394	2.657	54.324	20.481	3.906

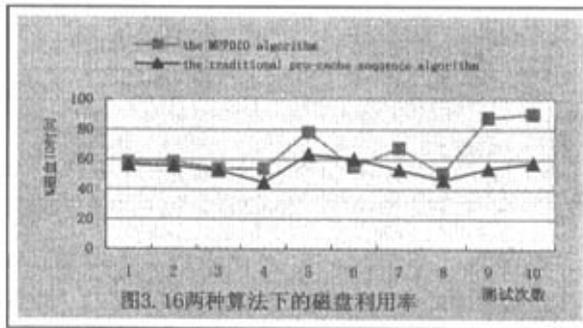
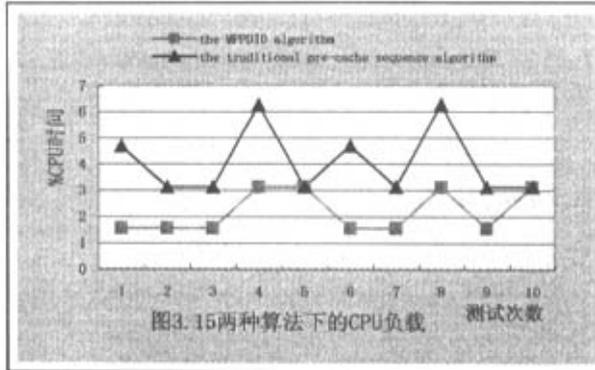


表 3.4、图 3.15、3.16 表明，与传统的 PRE_CACHE 顺序存取算法相比，MPPDIO 存取算法下程序所占 CPU 时间的百分比明显减少，CPU 负载降低了 31.98%；磁盘驱动器忙于为读请求提供服务所用时间的百分比明显增加，磁盘 I/O 通道的利用率提高了 20.24%。这正是 MPPDIO 存取策略的优势所在，也与前面的分析相符。

3.2 压缩数字播放系统的关键技术

数字化的电视台播出系统中，电视节目以音频、视频信息为主体，数据量大与实时性是多媒体应用的两大特点。从 MPEG2 技术标准出台以来，基于 MPEG2 技术的电视台业务解决方案和新设备不断推出，吸引了广播电视行业的众多目光。在中国，MPEG2 也越来越成为焦点，视频厂商也在热炒 MPEG2 技术，许多电视台正在思考自己需要使用哪些 MPEG2 产品。目前用于广播级视频的压缩编码一般为 MPEG2。采用 MPEG2 压缩得到的视频文件在一定的质量要求下尽量提高压缩比，有利于存储与传输的需要，但是不利于编辑（帧精确与多代后质量迅速下降的问题）。若采用无压缩的视频数据，可以解决编辑的问题，但同时带来的问题是数据量极其巨大，如采用 4:2:2 的 YUV 视频编码，1 小时的电视节目要占用 72GB 的存储空间。PAL 制的电视节

目的是 25frames/s 的速率, 其数据传输率达到 21MB/S, 意味着对磁盘 I/O 性能的要求很高, 因此现行的数字化播放系统大多为支持 MPEG-2 MP@ML 的视频流的播放系统。为了与现行压缩数字播放系统兼容, 我们的数字视频播放系统也实现了支持 MPEG-2 MP@ML 视频流的播放。本节也从软件实现方面讨论压缩视频播放系统的关键技术。下面首先介绍 MPEG 压缩标准。

3.2.1 MPEG 压缩标准

3.2.1.1 MPEG 压缩标准类型及其发展

MPEG, 译为运动图像专家组, 它是在国际标准化组织 ISO (International Standards Organization) 的召集下, 为数字视频和音频制定压缩标准的专家组。它同时也表示 ISO / IEC (International Electrotechnical Commission) 制定的视听信息(电影、录象、音乐等)数字压缩格式的编码标准。MPEG 使用了非常先进的压缩技术。与其它视频、音频编码格式相比, 在同样的质量下, MPEG 压缩文件要小得多。目前最通用的 MPEG 标准有: MPEG—1, MPEG—2, MPEG—4 和 MPEG—7。其中, MPEG—1 已经用于 VCD, MPEG—2 用于 DVD 和 DTV。MPEG 能够把数字视频数据压缩到原始大小的 1 / 40 或 1 / 50^[72-74]。

·MPEG—1^[75]

1992 年 ISO 和 IEC 之间的会议产生视音频编码标准, 即 MPEG—1。MPEG—1 支持数据交换, 但不限制创建或解码比特流的创新。比特流以数据层为基础, 数据层能够自我解释。比特流结构以一系列图片为基础, 每个图片包含需要创建成能单一显示图像的数据。有 3 种不同图片, 分类取决于每种图片是如何解码的。I——图片: 内部编码, 它们独立于其它图片编码; P——图片: 预测图片, 通过前面的 I 图片或 P 图片的运动补偿编码; B——图片: 内插图片, 通过前面和将来的 I 或 P 图片内插编码。这个过程叫做双向预测。D——图片: 图片是空间格式, 用来实现序列查询。一组视频可分成若干组图片(GOP), 在一组图片中, 第 1 个图片一定是 I 图片, 随机播放时也是从 I 图片开始。接下来是几个 B 图片, 然后是 P 图片, 之后再是 B 图片和 P 图片。编码时, 先编码 I 图片, 然后根据 I 图片, 编码 P 图片, 再根据前面已编码的 I(P)和 P 图片编码 B 图片。每一图片在编码中可分为若干片, 它包含若干宏块(16x16), 宏块又分为一些块。编码过程中, 运动估计、预测和量化以宏块为单位, DCT 变换以块为单位进行。

·MPEG—2

1994 年推出的 MPEG—2 压缩技术, 目标是高级工业标准的图像质量以及更高传输率。MPEG—2 把一个或多个视频、音频基本流, 包括其它数据合并成适于存储或

传输的单一流或复合流。可以用 2 种方式来描述: 程序流 (Program Stream) 和传输流 (Transport Stream)。每种流都被完善为不同的应用集合。程序流来自于把一个或多个打包的基于相同时间的元素流合并而成的单一流, 程序流被设计用于相对来讲无差错的环境, 适合用于软件处理。传输流把基于一个或多个独立时间的打包元素流合并成单一流。共享相同时间基础的元素流形成程序流。传输流被设计用于可能会出错的环境, 如在耗损或噪声媒体中存储或传输。传输流包有 188 字节。

MPEG—2 采用可变速率 (VBR——Variable Bit Rate) 技术, 能够根据动态画面的复杂程度, 适时改变数据传输率获得较好的编码效果, 目前使用的 DVD 就是采用了这种技术。MPEG—2 支持伸缩性编码, 同一数据流中可以包含不同分辨率的图像。不同的像素可以具有不同的分辨率, 可以低速或高速回放图像数据。与 MPEG—1 相比, MPEG—2 中没有 D 图片, 利用 I 图片来进行快速回放。MPEG—2 采用分层编码 (Hierarchical coding)。由于不同的像素可以具有不同分辨率, 利用分层编码设置不同的量化步长来处理 DCT 系数。当图像中活动或变化部分较多时, 先传输低分辨率的图像, 然后根据信道空间, 相应补充图像细节。这样, 低分辨率图像在低编码层, 高分辨率图像在高编码层, 对于运动视频, 分层编码快速高效, 编码后的图像效果很好。MPEG—1 没有考虑基于场的表示, MPEG—2 则允许对交错数据进行更有效的压缩, 对于每个宏块都允许基于场或帧的压缩。MPEG—2 主要应用于视音频压缩和传输。

·MPEG—4^[76-77]

MPEG—4 于 1999 年年初正式成为国际标准 (标准号为 ISO/IEC 14496), 在 1999 年 12 月的后继版本中增加了可变性, 半透明视频对象及其先进工具, 它进一步提高了编码效率并且与第一版兼容。它支持用于通信、访问和数字视听数据处理的新方法 (特别是基于内容的)。

MPEG—4 提供灵活的框架和开放的工具集, 这些工具支持一些新型的和常规的功能。由于快速发展的技术使得工具软件的下载极为便利, 因此这种方式极具吸引力。MPEG—4 使 CD—ROM 上的交互式视频和数字电视成为可能。MPEG—4 标准与 MPEG—1 和 MPEG—2 标准最根本的区别在于 MPEG—4 是基于内容的压缩方法, 而 MPEG—1 和 MPEG—2 是将整帧图像划分成固定大小的矩形子块处理图像, MPEG—4 根据内容, 把一幅图像分成各种尺寸的子块, 由于基于内容分成的子块内部信息相关性强, 能够产生高压缩比效果。

MPEG—4 主要针对低比特率, 特别是 64kbps 下的音频视频压缩。MPEG—4 运用运动补偿 DCT 算法来实现低比特率编码, 允许更多的运动补偿方式如标准的 8x 8 或 16x16 基于块的运动估计, 基于 8 个运动参数的全局运动补偿, 或子图的运动补偿。

MPEG—4 提供基于对象的分级功能, 有两种分级工具: 时域分级和空域分级。时域分级降低了原视频序列的帧率, 空域分级降低了原视频序列的分辨率。与 MPEG—2 不同, MPEG—4 提供了视频对象的可伸缩性, MPEG—4 称之为“视频对象面”, 它代表了视频对象平面的基本层或增强层。

MPEG—4 不仅支持自然声音和视觉图像, 还支持合成声音和合成的视觉对象, MPEG-4 的音频部分将音频的合成编码和自然声音编码结合, 视频部分把自然编码和合成视觉对象编码合成在一起。MPEG—4 提供的场景描述和用户交互工具用来描述场景中的元素以便组合产生待显示的图像。

MPEG—4 主要应用于数字电视, 交互式图像应用, 交互式多媒体(网络, 根据内容来分布和存储), 和基于计算机网络的可视化实验室场景应用等。

·MPEG—7^[78-79]

随着多媒体信息的增多, 多媒体信息在数据库中的存储和基于内容的查询变得越来越重要。当前多媒体信息的存储和查询只能是间接的, 即通过文件名或文件描述信息来进行。缺少灵活性和智能性。从 1996 年 10 月起, ISO / IEC 下的 MPEG 开始制定多媒体信息内容描述标准, 以便能快速有效地搜索所需的各种信息。MPEG-7 的标准名称是“多媒体内容描述接口”, 目标是: 指定一组描述不同多媒体信息的标准描述符, 并且这些描述要与信息内容相关, 以便能用来快速有效地查询各种多媒体信息。MPEG—7 仅是表达描述有关内容的信息, 但并不表示内容本身, 所以它也被称做“the bit about the bits”——元比特。MPEG—7 可以独立于其它 MPEG 标准, 但它仍支持 MPEG—4 的视频、音频标准。由于 MPEG—7 使我们可以对压缩后的信息用相关内容来查询。如查找罪犯资料时, 可以根据目击者画出的嫌疑犯画像在数据库中查找, 匹配相应颜色信息或轮廓等信息; 若查找一首曲子, 可以哼一段相似曲调, 在音频库中查找具有相似波段的曲目。

MPEG—7 目前的应用主要有 3 类: (1)第 1 类是索引和检索类应用。使人们能够像检索文字一样对视音频进行快速有效的检索。(2)第 2 类是选择和过滤类应用, 帮助人们仅接受符合需要的广播发送信息服务数据的应用。(3)第 3 类是生物医学应用和基于视觉的控制等专业应用, MPEG—7 基于内容的多媒体信息检索非常重要的一部分是提取特征, 但特征提取和搜索的算法并不包含在制定的标准中, 主要原因是促进竞争, 以得到各种好的算法实现。

MPEG—7 使多媒体信息查询更加直接, 更加智能化。MPEG—7 主要应用于数字图书馆、广播媒体选择, 多媒体编辑, 地理信息系统, 购物等方面, 在教育、新闻、生物医学、公安调查、视频存档等方面也有潜在的应用。

表 3.5 MPEG 的各个标准的比较

标准	数据率	分辨率和帧速	伸缩性	基于对象	信噪比
MPEG-1	中	较高	较好	否	高
MPEG-2	高	当前最高的分辨率和帧速	基于空间、数据率和幅值	否	最高
MPEG-4	低		基于内容	是	高
MPEG-7	低		基于内容	是	高

·MPEG—21^[80]

MPEG—21 的正式名称是“多媒体框架”(Multimedia Framework), 目标是“将标准集成起来支持和谐 / 协调的技术以管理多媒体商务”。目前人们已有许多技术发送和使用多媒体内容, 但一直没有一个标准将各种技术联系在一起, 而基于互联网的数字经济市场的发展, 使传统的交易发展为电子商务。各种媒体内容(视音频、文本等)的界限越来越模糊, 所以需要新的综合性标准来协调管理不同的内容。在 1999 年 12 月的 MPEG 会上, 确定了 MPEG—21 主要为 2 种关键技术的结合: 用户如何能查询和获得满意的内容、如何将内容根据与内容结合的使用权送给用户。MPEG—21 与传统的自下而上分析技术标准需求的工作方式不同, 它是从商业内容和提供与内容相关服务的前景出发来开展工作。2000 年 6 月, 正式批准制订 MPEG—21 标准, 并确定 MPEG—21 框架包括 6 部分: 用户需求、IPMP 相关技术、终端和网络、内容辨识和描述、内容管理和使用、内容表达。

MPEG 的各个标准的比较见表 3.5。从表 3.5 可看出: MPEG 系列标准 (MPEG—1、MPEG—2、MPEG—4) 都采用一些基本技术实现高品质的画面传输和压缩。但 MPEG—7 及以后的标准重点已经不在压缩上, 而是更侧重于多媒体数据基于内容的存储、检索和交互。即 MPEG—1 使得 VCD 取代了传统的录像带, MPEG—2 使高质量的 DVD 取代 VCD, MPEG—4 提高了多媒体的交互式和灵活性, MPEG—7 使基于内容的多媒体存储和查询成为可能, MPEG—21 综合各种标准的优点。MPEG 标准能够高效压缩多媒体数据, 使大量多媒体数据的传输及存储查询技术向着规范化、灵活性、智能性发展。

3.2.1.2 压缩标准的比较

表 3.6 反映了 MPEG 体系中着重压缩与传输性能的标准的部分技术指标。从表 3.6 可看出: MPEG-4 是高位比特率有损压缩, 其图像质量始终无法与 MPEG-2 相比。当 MPEG4 与 MPEG-2 码率输出相同时, 其质量仍稍逊于 MPEG-2; 同时 MPEG-4 对硬

件要求也较高,事实上,我们注意到 MPEG-4 在保证令人满意的图像质量的情况下,更侧重于较低数据率和灵活的交互功能。因此对于 MPEG 体系中着重压缩与传输性能的标准来说, MPEG-2 是最适宜于广播电视行业的压缩标准,而目前的数字图像压缩格式还是很多的,并不只有大家关注的 MPEG-2 这一种。其他的格式还有几种,象 Apple 的 QuickTime, Intel 的 Indeo, Radius 的 Cinepak, 低速视频传输用的 H261、H263 等,以及 DV 格式和目前在中国数字视频领域应用广泛的 M-JPEG。在这么多的压缩格式中,广播电视行业根据什么来选择使用不同的压缩标准?

表 3.6 MPEG 体系的部分技术指标

	MPEG-1	MPEG-2	MPEG-4
最大图像输出	352*288	1920*1152	720*576
默认图像输出 (PAL)	352*288	720*576	720*576
最大音频取样	48KHz	96 KHz	96 KHz
最大音频声道	2	8	8
最大数据率	3Mbit/s	80Mbit/s	5 to 10Mbit/s
常用数据率	1.38Mbit/s	6.5Mbit/s	880Kbit/s
图像质量满意度	一般	非常好	好
编码硬件要求	Low	High	Very high
解码硬件要求	Very low	Medium	High

首先,广播电视行业需要的是高质量、兼容性好的压缩算法。QuickTime、Indeo 和 Cinepak 都没有提供对声音的压缩,基本都是以软件压缩/解压缩为主,包括 H261、H263 在内,都存在帧频低、图像质量差或需专门的平台支持等问题,因此不适用于在广电领域使用。相反, M-JPEG、DV (Digital Video) 和 MPEG2 在硬件的支持下,可以保证在支持正常的视频帧频率情况下,提供相当好的图像质量。这就决定了只能是 M-JPEG、DV 和 MPEG2 可以在电视台占一席之地。

其次,在同样的视频质量下,压缩算法的输出码率直接决定其性能的优劣,或者说,输出码率相同的条件下,图像质量的优劣反映了压缩算法的性能。由于 M-JPEG 和 DV 都是采用帧内压缩方式,它们的压缩效率当然要比 MPEG2 低。当然,这是一个平均情况,在低码率的时候, MPEG 可以比 M-JPEG 的压缩比高很多而保持较好的图像质量;而在要求图像质量很好的时候(比如演播室节目编辑和后期制作), MPEG2 与 M-JPEG 的输出码率差别不是很大。

第三,由于电视台业务的多样性,要求一个压缩标准能提供多种码率。比如说节目传播环节,在传播不同类型的节目时,所需的带宽是不同的,体育节目要比新闻节

目的码率高,而新闻节目又要比电影节目高。再比如,考虑到进行过程中复制次数的不同,节目制作环节和节目传播环节所需的码率也不同,传播环节可以使用低码率。压缩算法的可变码率 VBR (Variable Bit Rate) 特性对电视台有效利用资源有非常重要的作用。MPEG2 可以通过改变 GOP (Group of Picture) 结构和离散余弦变换 (DCT) 及 Huffman 编码的参数等手段来调整输出的码率, DV 格式基于它的应用特点,没有提供 VBR 技术。M-JPEG 虽没有帧间编码,但是也可以通过改变 DCT 及 Huffman 编码参数调节压缩比。

通过以上讨论,我们可以看出, MPEG2 标准在压缩性能上,确实具有比其他标准优越的地方。但是由于 M-JPEG 发展较早,经过多年的在非线性视频编辑上的应用,软、硬件技术成熟,成本低廉,以其目前的硬件平台的价格来看,平均比 MPEG2 平台便宜 5000 美元左右。DV 的优势是与磁带的紧密结合。另外 M-JPEG 和 DV 都是帧内压缩方式,可以提供精确到帧的随机存取访问,不附带任何的访问延迟,这与 MPEG2 有相当大的不同。因此,在目前的电视业务中,基于 M-JPEG、DV 和 MPEG2 三种压缩标准的设备都在被使用。但 MPEG2 标准的好处是其压缩效率高。在不强调图像质量的情况下,对电视台业务的大部分应用场合都适用,因此 MPEG 2 成为电视行业的热点,下面将介绍 MPEG2 的优势。

3.2.1.3 MPEG-2 压缩格式特点与优势

数字电视的视频压缩编码基本采用 MPEG-2 的标准, MPEG-2 涵盖了从标准清晰度电视到高清晰度电视范围内的视频技术标准,用于 HDTV, DVD 和 DTV。MPEG-2 标准的目标是高级工业标准的图像质量以及更高传输率。和 MPEG-1 格式相比 MPEG-2 具有以下特点^[73-74]:

MPEG-2 处理 4:2:2 颜色分辨率和高比特率的图片。比较测试证实 MPEG-2 视频性能接近甚至在许多情况下超过为高比特率或演播室应用研制的标准或规范。

MPEG-2 采用可变速率(VBR—VariableBitRate)技术,能够根据动态画面的复杂程度,适时改变数据传输率获得较好的编码效果,目前使用的 DVD 就是采用了这种技术。

MPEG-2 支持伸缩性编码,同一数据流中可以包含不同分辨率的图像,不同的像素可以具有不同的分辨率,可以低速或高速回放图像数据。

MPEG-2 采用分层编码(Hierarchicalcoding)。对于运动视频,分层编码快速高效,编码后的图像效果很好。MPEG-1 没有考虑基于场的表示, MPEG-2 则允许对交错数据进行更有效的压缩,对于每个宏块都允许基于场或帧的压缩。

MPEG-2 标准的传输速率比 MPEG-1 高 1.5~2 倍。图象处理有隔行扫描式和顺序

扫描式。

随着 MPEG-2 在广播电视领域的应用,各种针对 MPEG-2 的软硬件技术也得到较快发展,在编辑方面,SONY 和 NEC 等公司已有进行 I 编辑, I、B 编辑和 I、B、P 编辑的软硬件产品问世,编辑效果接近 M-JPEG。在编解码的切换方面,信息总线 info-bus 和 mole 技术为之铺平了道路。在广播电视领域,采用 MPEG-2 压缩方式正形成主流,正是考虑到这些优势和系统的兼容性,本系统既支持标准清晰度电视的无压缩媒体流的一路播放,又能支持基于 MPEG-2 压缩格式的媒体流的多路采编播,下面介绍压缩数字播放系统的关键技术。

3.2.2 实现帧精确与无缝转接的方法

电视台用户对节目的播出质量有严格的要求,特别是对于广告节目,播放过程中要求播放时间和位置精确到帧(可以是 B、P 帧),同时要保证顺序播放的两个节目之间保持视频音频同步,转换平滑,不能出现抖动、停滞或黑场^[10,29,81]。现有的基于 mpeg2 的压缩播放系统一般通过双解码卡来实现这一要求,本节将讨论一种更经济、更合适的方法。

若采用传统的顺序打开文件播放的方式,由于打开节目文件及读取视频流到播放输出缓冲区的延时,往往很难满足无缝转接的要求。而采用多线程预读的方式,在前一个节目播放完之前就打开下一个节目文件,预读视频数据到播放输出缓冲区,并且和精确到帧的播放缓冲区控制相配合,就可以很好地解决这一问题。针对 Pinnacle 视频卡,我们通过采用双播放缓冲区多线程调度算法实现了上述功能。

1) 媒体播出数据路径

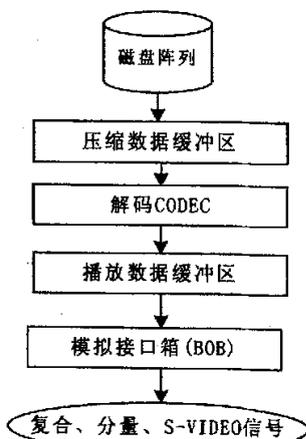


图 3.17 播出部分数据路径

节目视频数据以 MPEG-2 的格式存储在磁盘阵列上。在播放时，播放程序首先对视频卡和接口箱（BOB）及其缓冲区进行初始化，然后根据播放要求，以帧为单位从磁盘阵列读取视频压缩数据，送入压缩数据缓冲区中，并驱动解码 CODEC 进行解码，解码后的无压缩视频数据先送入播放数据缓冲区，然后经接口箱（BOB）进行 D/A 转换后变为模拟视频信号输出。通过配置接口箱，输出的模拟信号可以是复合信号或 YUV 分量信号。

2) 冗余双缓冲区设计

由播出部分数据路径可以看到，传统的播放控制程序在解码前后使用了两个缓冲区分别用于解码和播出，即压缩数据缓冲区和播放数据缓冲区，压缩数据缓冲区用于存放节目的压缩视频数据，供 CODEC 解码用，播放数据缓冲区用于存放 CODEC 解压后的视频数据，用于播放输出用，两个缓冲区一般设计为环形结构，并采用典型的生产者、消费者工作模式，播放缓冲区使用两个指针分别指示视频数据起始点和结束点，当有视频数据时就取起始指针所指数据进行播出。工作机制如图 3.18 所示^[87]。

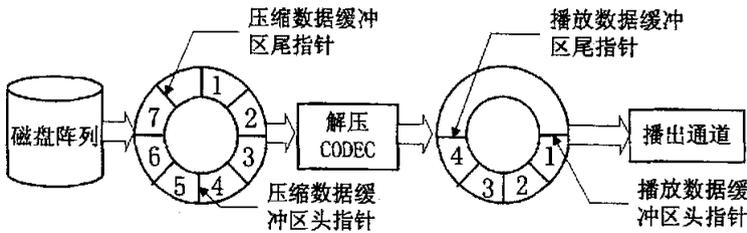


图 3.18 传统缓冲区的播放模式

两个缓冲区的同步通过几个共享变量来实现。下面是同步控制的设计：

```
#define COMBUFFERNUM 40 //压缩数据缓冲区缓冲块数目
#define PLAYBUFFERNUM 20 //播放数据缓冲区缓冲块数目
int ComBuffer_Full = 0; //压缩数据缓冲区的视频数据所占缓冲块数目
int ComBuffer_Free = COMBUFFERNUM; //压缩数据缓冲区空闲块数目
int PlayBuffer_Full = 0; //播放缓冲区中解码后的视频数据所占缓冲块数目
int PlayBuffer_Free = PLAYBUFFERNUM //播放缓冲区空闲块数目
```

```
DiskReadThread { //磁盘读线程
```

```
    if(ComBuffer_Free) { //如果压缩数据缓冲区中有空闲缓冲块可用
        读磁盘数据入压缩数据缓冲区;
```

```
ComBuffer_Free--; //压缩数据缓冲区空闲缓冲块数目减一
ComBuffer_Full++; //压缩数据缓冲区可用缓冲块数目加一
}else ...
}

DecodeThread(){ //解码线程
    if(ComBuffer_Full&&PlayBuffer_Free) { //有压缩视频数据可用
        //并且有空闲播放缓冲块可用, 则进行解码
        解压压缩视频数据到播放缓冲区;
        ComBuffer_Full--; //压缩数据缓冲区可用缓冲块数目减一
        ComBuffer_Free++; //压缩数据缓冲区空闲缓冲块数目加一
        PlayBuffer_Free--; //播放数据缓冲区空闲缓冲块数目减一
        PlayBuffer_Full++; //播放数据缓冲区可用缓冲块数目加一
    }else ...
}
```

```
PlayThread(){ //播放线程
    if(PlayBuffer_Full){ //如果播放数据缓冲区有未播放数据, 则予以播放
        取无压缩视频数据播出
        PlayBuffer_Free++; //播放数据缓冲区空闲缓冲块数目加一
    } else ...
}
```

在播放单个节目、非连续节目的时候, 采用这种工作方式不存在任何问题, 但是在多个节目顺序播放的情况下, 特别是节目的起始帧为 B、P 帧时, 这种机制就不是最有效的处理机制。

当节目是以 B、P 帧为起点时, 在单个解码卡的情况下, 为能对起始帧进行解码, 必须先获取该帧所在 GOP 组中该帧前的所有帧信息, 然后从该 GOP 组的 I 帧开始解码直到要播放的帧。为实现精确到帧的播放, 播放帧前的帧不予播出, 只是辅助进行播放帧解码用, 可称之为“无效帧”, 对“无效帧”的处理可以采用以下三种处理方式:

第一种处理方式采用播放指针直接跳转的方式, 适用于以上所示的两个缓冲区的播放控制模式, 在该工作模式下, 我们要为每帧设立专门的有效标志, 有效代表正常播出帧, 无效代表无效帧, 解码后的视频数据均直接送入播放数据缓冲区中, 在循环播放过程中, 当播放指针指向无效帧时, 将直接跳转到下一帧进行处理。

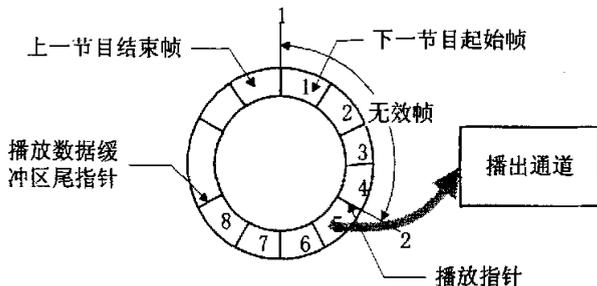


图 3.19 播放指针跳转的播放缓冲区操作

图 3.19 所示为该处理方式下对播放缓冲区数据的操作，假设下一节目的解码结果从标号为 1 的帧开始，并假设编号 1~4 的帧为无效帧，则当前一个节目播放完毕后，通过检测有效位，播放指针将依次跳过 1~4 帧，直接从第 5 帧开始播放。

该种处理方式存在的问题是：当视频节目的 GOP 组很大而播放起始帧恰好在 GOP 的最后几帧时，需要一次解码缓存较多的“无效帧”，在这种情况下，播放指针要进行多次比较和跳转，增加了播出延时，特别是在播放缓冲区分配的不够大时，由于无法一次预读较多帧，会影响到播放帧的及时处理。

第二种处理方式是设立一个较小的“无效帧”暂存缓冲区，专门用于存放“无效帧”数据，有效的播放视频数据均直接送入播放数据缓冲区进行循环播放，而避免将“无效帧”送入播放缓冲区。为节省内存空间，暂存缓冲区可以设计的很小(只需要能容纳一帧无压缩视频数据量大小)，并且可以循环使用，工作机制如图 3.20 所示，假设无效帧为 1、2、3 帧，则 1、2、3 帧解压缩后的视频数据将直接送入“无效帧”暂存缓冲区丢掉，而不进播放缓冲区。

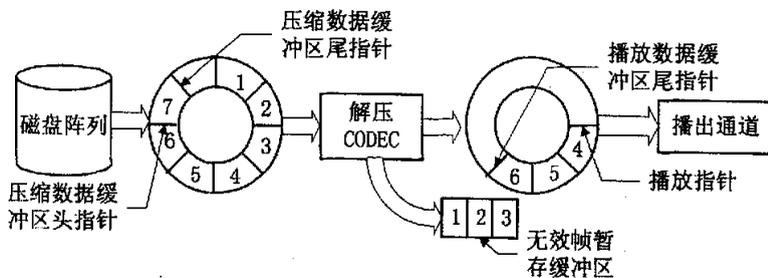


图 3.20 使用暂存缓冲区的播放模式

在该工作方式下，对缓冲区操作需要进行以下修改：

增加如下布尔变量数组 `pIs_Valid[]` 对压缩数据缓冲区的帧作是否“有效”标记。

```
    BOOL pIs_Valide[COMBUFFERNUM];
    解码程序作如下修改:
DecodeThread(){
    if(ComBuffer_Full&&PlayBuffer_Free){ //压缩数据缓冲区有未解压数据, 且播放//缓
        冲区有空闲缓冲块可用
        if(*pIs_Valide){ //如果为有效帧则进行解压, 压缩数据缓冲区
            解压压缩视频数据数据到播放缓冲区;
            ComBuffer_Full--; //修改同步变量
            ComBuffer_Free++;
            PlayBuffer_Free--;
            PlayBuffer_Full++;
        }
        else{
            解压无效压缩视频数据数据到暂存缓冲区;
            ComBuffer_Full--; //修改同步变量
            ComBuffer_Free++;
        }
    }
    else ...
}
```

该种方式存在的问题是: 暂存缓冲区不能用于播放, 存在内存空间的浪费; 预取数据量受播放缓冲区限制, 一次不能对下一节目较多视频的数据预解码。

第三种处理方式是采用两个播放缓冲区: 正播数据缓冲区和预播数据缓冲区。正在播放节目的压缩视频数据经解码后直接送入正播数据缓冲区中, 而下一播放节目在前一节目结束前就可以预解压, 并将解压缩后的视频数据直接送入待播节目播放缓冲区中, 两个缓冲区共用一个播放指针, 一旦前一个节目播放完毕(正播数据缓冲区数据全部送入播放通道), 就将两个缓冲区的首尾指针互换, 从而实现正播数据缓冲区和预播数据缓冲区的角色互换。在接下来的播放中, 原来的正播数据缓冲区变为预播数据缓冲区, 用于解压下一节目用, 原来的预播数据缓冲区转为正播数据缓冲区用于播放输出。

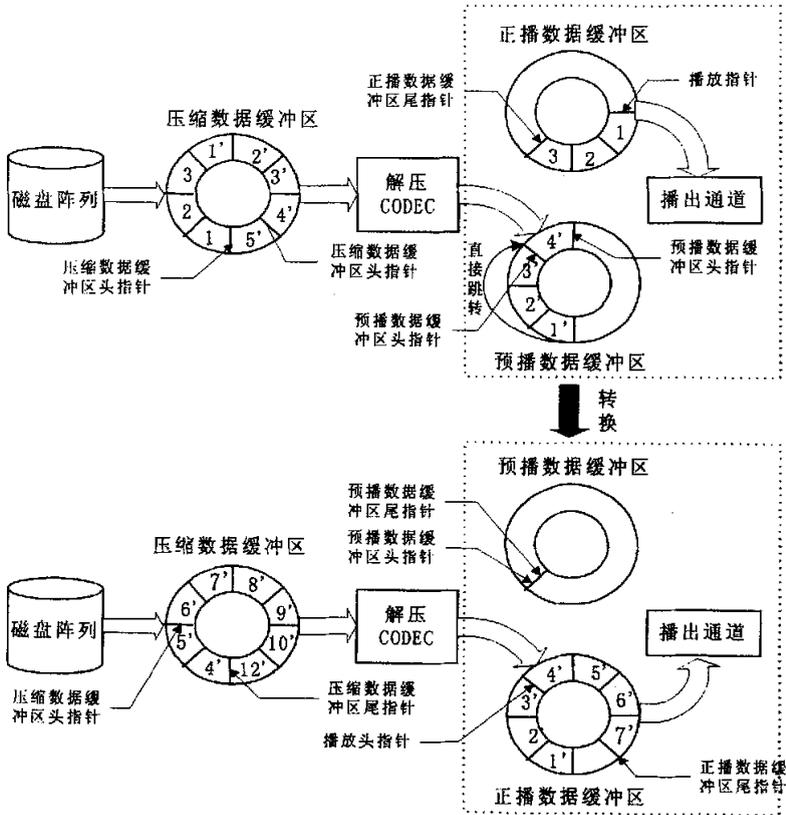


图 3.21 冗余双缓冲区的播放模式

在这种工作模式下，如果是从 B、P 帧开始播放，其第一个有效帧前面的无效帧也可以解码后存入预播数据缓冲区中。在角色互换前，只需要将预播数据缓冲区头指针直接指向起始有效帧。如图 3.21 所示，假定前一节目各帧编号为 1、2、3……，后一节目各帧编号为 1'、2'、3'……，并假定后一节目 1'-3' 帧为无效帧，由转换前的图可知：前一节目尚未播放完毕，下一节目已经开始解码，并将头指针定位到“起始有效帧”（在图中为 4' 帧）；由转换后的图可知：一旦前一节目播放完毕，立即完成缓冲区的角色互换，原来的播数据缓冲区用于播出节目，而原来的正播数据缓冲区则用于解压接下来的下一节目。

采用双播放缓冲区模式，一方面可以实现 B、P 帧的帧精确，另一方面，由于两个播出节目分别使用两个分离的播放缓冲区，可以避免节目之间因解码造成的相互干扰，容易实现播放控制。

我们可以看到，在单个播放卡情况下，无论采用上述的那种工作模式，由于只使用一个解码 CODEC，都必须在前一节目的帧全部解码完成的情况以后，才能进行下一节目的预解码，解码芯片的解码速度一般远远大于播放速度，假定解码速度为

V_d (Frame/s)，节目素材文件 GOP 最大长度为 L_g (Frame)，则为实现迅速定位到任意帧的目的，每个节目的预解码时间 T_d 应该满足：

$$T_d \geq L_g / V_d$$

而每个播放缓冲区所含缓冲块数目 B_b 应该满足：

$$B_b \geq T_d * V_d \geq L_g$$

每个播放缓冲块的大小必须以一帧无压缩视频数据的数据量为单位进行分配。其最小缓冲容量应该满足 3.1.2.2 节所分析的缓冲区最小值的表达式，在内存容量允许的情况下越大越好。

(3) 多线程调度

为了提高系统的效率，我们采用几个线程并行运行，协调工作，来实现从视频数据的读取到解码，一直到最后的播出。下面我们就对播出过程中各线程的工作过程作一下详细的分析。我们在具体实现过程中共采用了四个工作者线程，其具体的工作流程如图 3.22 所示。

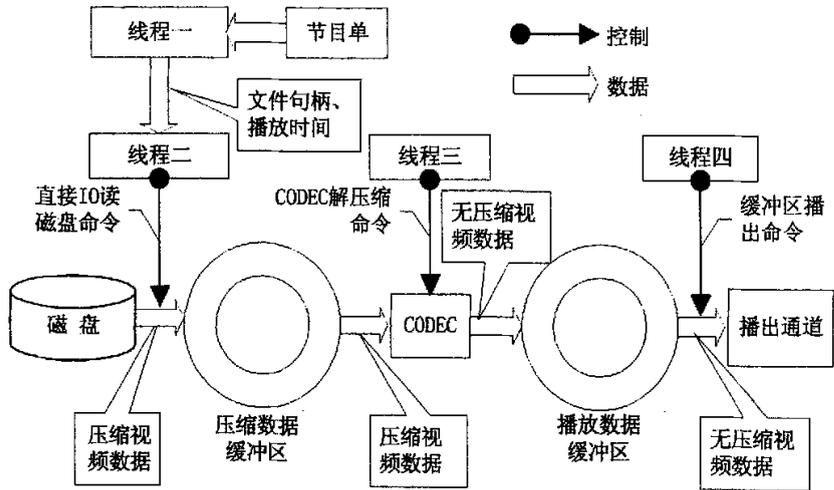


图 3.22 多线程调度

其中，线程一负责监视当前节目的播放进度，并在当前节目播放结束前预打开下一个节目文件，提取下一节目文件的信息，向线程二传送下一个节目文件的文件句柄、播放起始位置和结束位置。

线程二则负责根据当前节目文件句柄和节目的播放位置，在节目播放之前，从磁盘预读视频数据到压缩数据缓冲区中的空缓冲块中。

线程三则负责控制 CODEC 预解码压缩数据到播放数据缓冲区的空缓冲块中。

线程四则是最终的播出控制线程，负责根据节目单将播放缓冲区中的视频数据发送到视频服务器的播出通道。

线程三（解压缩线程）与具体的播放节目无关，只负责将压缩数据缓冲区中的待解压缩数据解压缩到播放数据缓冲区中的空缓冲块中。这样，当前一个节目还没有完全播完时，后一个播放节目的视频数据就已经解压缩到播放数据缓冲区等待播出，从而做到前一个播放节目的视频数据与后一个播放节目的视频数据紧密衔接，线程四（播放线程）只需要根据节目的开始位置、结束位置、播出时刻以及已播出的帧数决定是否将当前播放数据缓冲区的视频数据送到播放输出通道。这样，四个线程协调工作就可以真正实现不同节目之间播放的连续性，从而达到播放连续无黑场的要求。

小 结

播出质量是数字视频播放系统另一个最重要的技术指标。无压缩数字播放系统必须实现节目播出时不掉帧、画面无黑场、无闪烁，且能按帧切换。传统的 PRE-CACHE 顺序存取策略不能满足其性能要求，因此本章提出了一种 MPPDIO 算法，并建立了其共享缓冲区的大小配置的计算模型。

分析与实验结果均表明：MPPDIO 存取算法完全能实现无压缩媒体流的不掉帧、无黑场的关键性能，既能保证播放质量，又提高了磁盘 I/O 通道的利用率，降低了 CPU 负载，传统的 PRE_CACHE 顺序存取算法则不能实现系统的这些关键性能。MPPDIO 存取算法已成功应用于电视台无压缩数字播放系统中，经过用户两年的实际运行，效果良好。MPPDIO 存取算法也可以在未来用于高清晰度电视的数字播放系统中。

MPEG2 标准的压缩效率高，只要不用它进行演播室节目编辑和后期制作，它对电视台业务的大部分应用场合是适用的。为了解决基于 MPEG2 的压缩数字视频播放系统的播出能精确到帧及两个节目之间平滑转换的问题，本章分别对不同的缓冲区设计方法及其性能进行了分析和比较，提出并实现了冗余双播放缓冲区多线程调度算法，很好地解决了这一问题。指出：该算法既不影响播放帧的及时处理，又不使预解码数据量受播放缓冲区限制。与现有的压缩播放系统靠双解码卡实现帧精确与平滑转换相比，具有巧妙性，和可推广性。

4 数字视频播放系统的关键技术

电视台除了要求视频播放系统能实现无压缩视频数据和压缩视频数据节目的采编播的普通功能外，还要求系统能实现一些特殊的功能，使系统能满足电视台实际运行时的一些特殊需要。如：直播功能、延迟播、take、hold、格式转换等。下面就其实现与关键技术一一进行论述。

4.1 直播

传统的电视节目播出方式主要是录播，录播的节目生产周期长、成本高、效率低。对于新闻类节目和谈话节目，由于生产周期长和效率低，往往失去了新闻价值。电视直播能把人们身边即将发生或正在发生的事情及时、准确、真实、迅速地传播给广大观众，这不仅能极大地提高电视的现实性，而且还能邀请嘉宾来参与，观众也可通过热线电话参与；从而提高了观众的收看兴趣，提高了收视率。因此，电视直播技术被越来越多的电视台所利用^[93]。

通用的电视直播系统，不仅能够用于新闻直播，而且可以用于谈话类和综艺类节目的直播。目前大多数电视台的电视直播系统都是由以前的演播室系统发展而成，一般可分为4个部分：视频部分、音频部分、时钟系统、热线电话系统，而且这四部分都是由大量相应的硬件系统实现的，如图4.1，4.2，4.3，4.4所示。

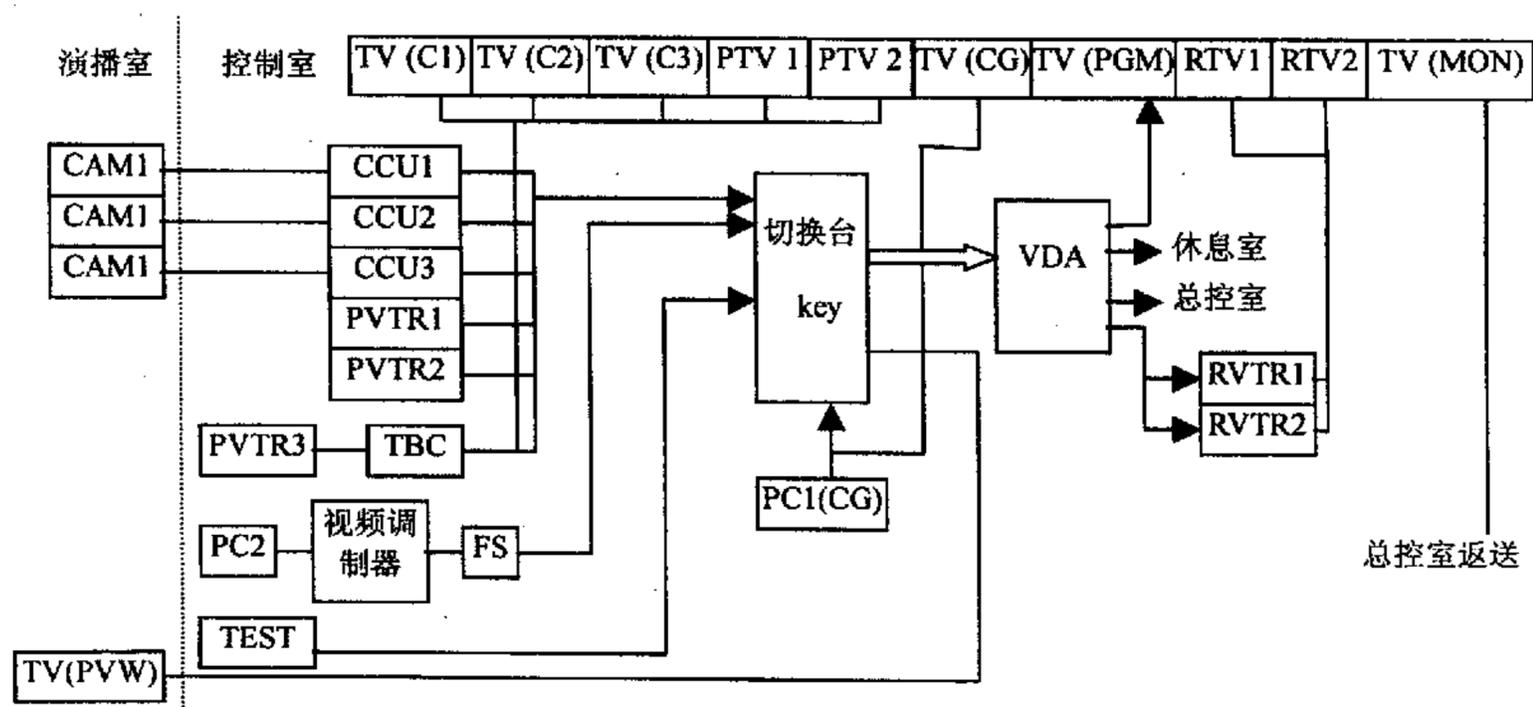


图 4.1 目前视频直播系统的视频部分的简图

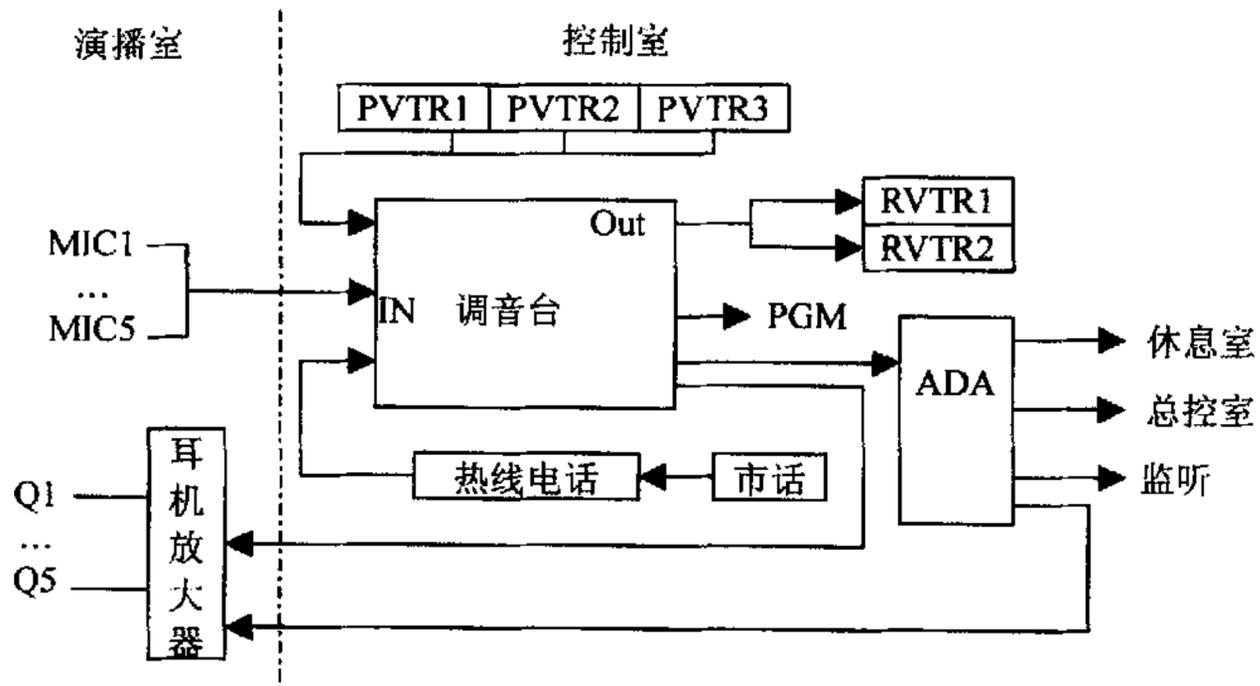


图 4.2 目前视频直播系统的音频部分的简图

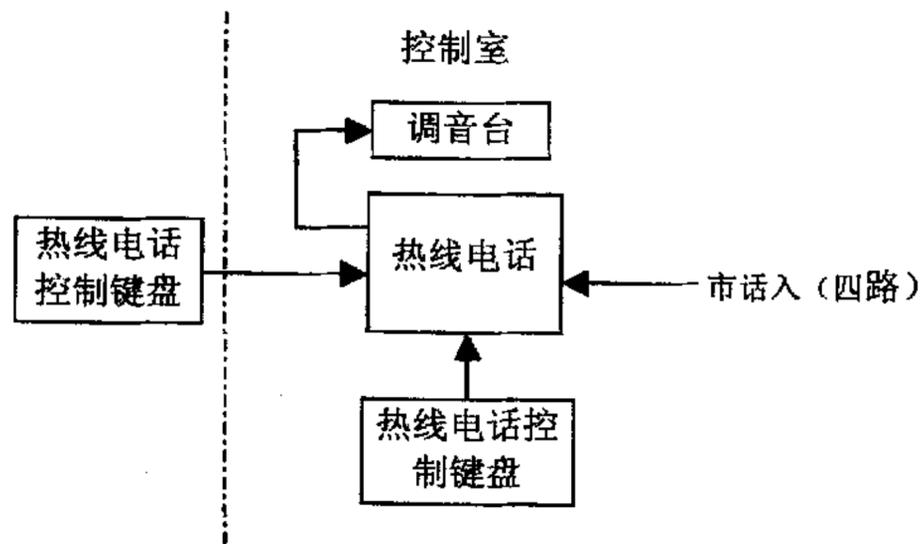


图 4.4 目前视频直播系统的热线电话部分的简图

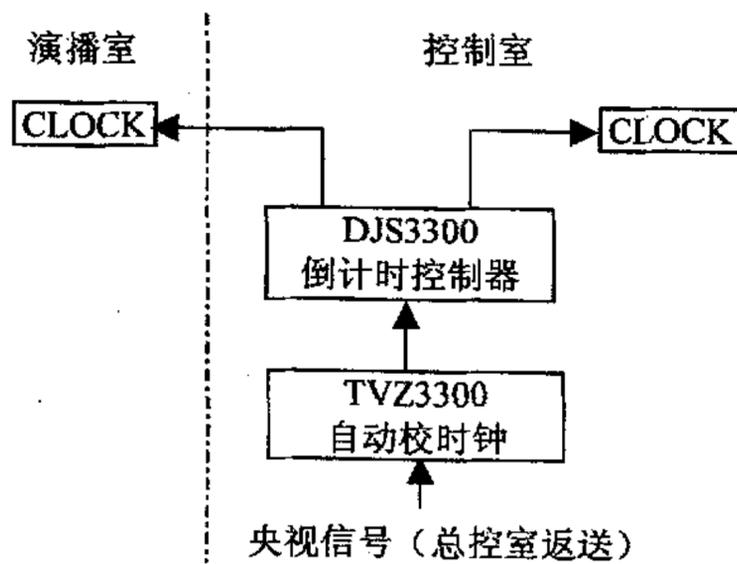


图 4.3 目前视频直播系统的时钟部分的简图

这样的直播系统，虽然功能较全，具有热线电话等，但硬件繁多而且操作复杂，要至少四人同时工作，才能做好直播节目。由于人多，存在协调同步问题；由于硬件繁杂，存在容易出现硬件问题与价格昂贵的问题；由于音频与视频分开，存在音视频同步的问题。而我们的视频播放系统的音视频直播处理是在 TARGA3000 的基础上主要由软件来实现的。

其数据流图如图 4.5 所示。模拟视频信号经接口箱(BOB)进行 A/D 转换，转为数字信号以后，以无压缩视频数据的形式送入该共享缓冲区，在供播放 DSP 进行回放的同时，由压缩 CODEC 压缩为 MPEG-2 数据写入磁盘文件中。在具体实现时，在主机内存为播放与捕获分配一个共享视频数据缓冲区。这样视频服务器在直播处理过程中完成了两件事情，一是将直播节目通过视频卡进行实时直播，另一方面完成了节目的实时捕获，并将捕获的节目文件作为素材存到本地磁盘存储器上。

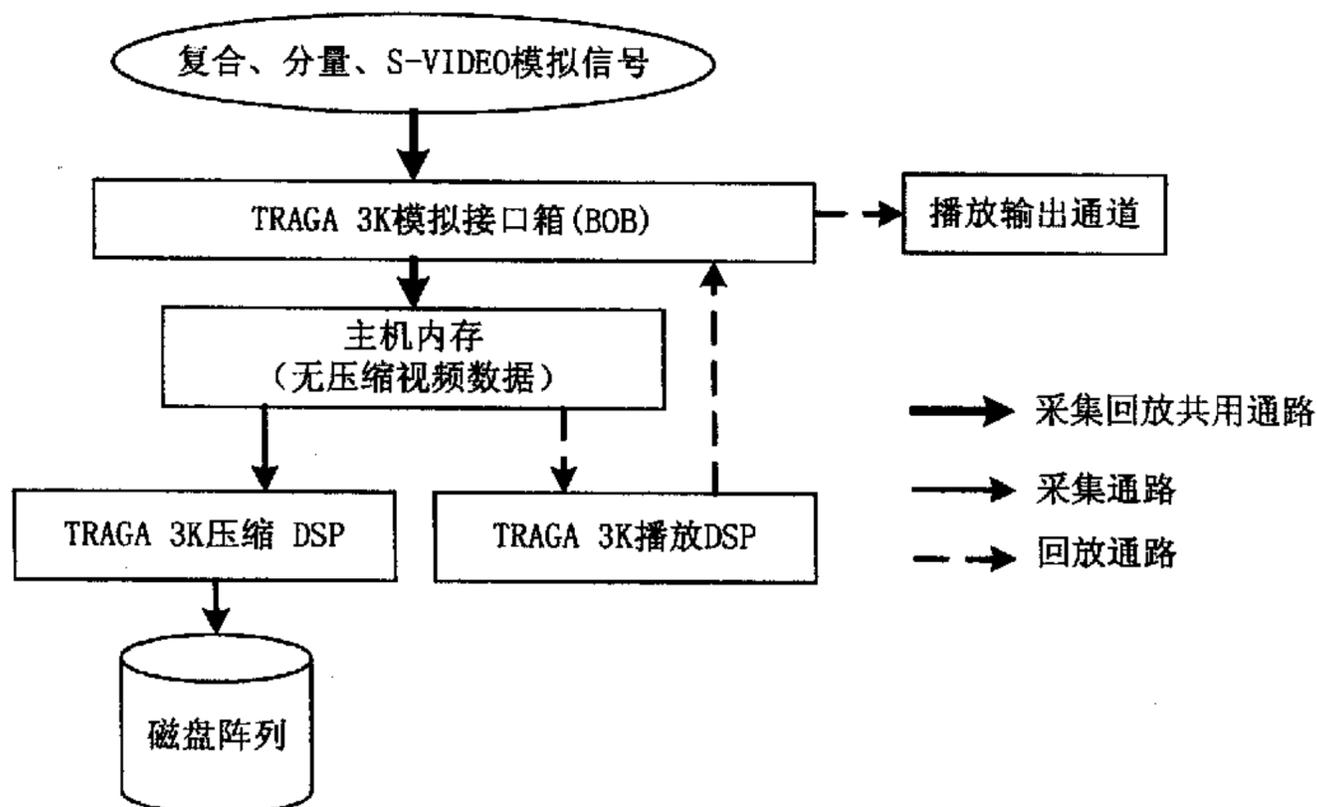


图 4.5 直播的数据流图

由于它不需要另外的硬件，是我们研制的高质量数字视频播放系统的功能的一部分，因此与通用的直播系统相比，它的实现更简单且更可靠。由于它的音视频同时实现，因此不存在音视频同步的问题，它的实时性、同步性更好。只需一人操作，不存在人员协调同步问题，在直播的同时也将现场的实况存到本地磁盘存储器上，因此性价比更高。

4.2 延迟播放

延迟播放是相对直播提出的，由于电视台需要对某些直播节目进行实时监控，在转播直播节目时不是立即播出，而是在对节目进行审阅后，延迟一段时间再播出，以满足电视台的特殊要求。延迟的时间一般在几秒到几十分钟之间。目前的延迟播放系统大多由大规模集成电路实现，价格昂贵，国内也较罕见^[94]。而且随着延迟时间的增长其价格大幅增长，一个延迟 30 秒的延迟播系统价格就要 30 万左右。而我们所研制的延迟播放系统是在一块 TARGA3000 视频卡的基础上用软件实现的，相比之下，性价比要高得多，而且也解决了延迟时间长度限制的瓶颈。

1) 延迟播放方案的比较

延迟播放，一般需要将延迟播出的数据缓存在某一地方，到播出的时间后再播出。

(1) 以无压缩的音视频数据缓存在内存的方式实现延迟播。如果采用与直播一样地实时回放，实时采集的方式，将音视频数据缓存在内存，在回放前，系统需要在主存中缓存大量的原始无压缩视频数据，以 PAL 制式为例，一帧数据为 8294400bit，以 1 分钟延迟计算，一次缓存的无压缩视频数据量为：

$$8294400(\text{bit/Frame}) \times 25(\text{Frame/s}) \times 60(\text{s}) = 12441600000(\text{bit}) \approx 1.45\text{GB}$$

如此大的数据量不可能在内存中一次缓存，这样会出现延迟时间瓶颈，使延迟时间不得超过 30 秒。即使 30 秒这样短时间的延迟对系统的要求也很高，需要系统配置 1G 以上的内存。显然性价比低，这种方法不可取。

(2) 以无压缩的音视频数据缓存在磁盘的方式实现延迟播。我们将原始的无压缩音视频数据缓存在磁盘上，在只有一块采集卡的情况下采用如图 4.6 所示的方式。

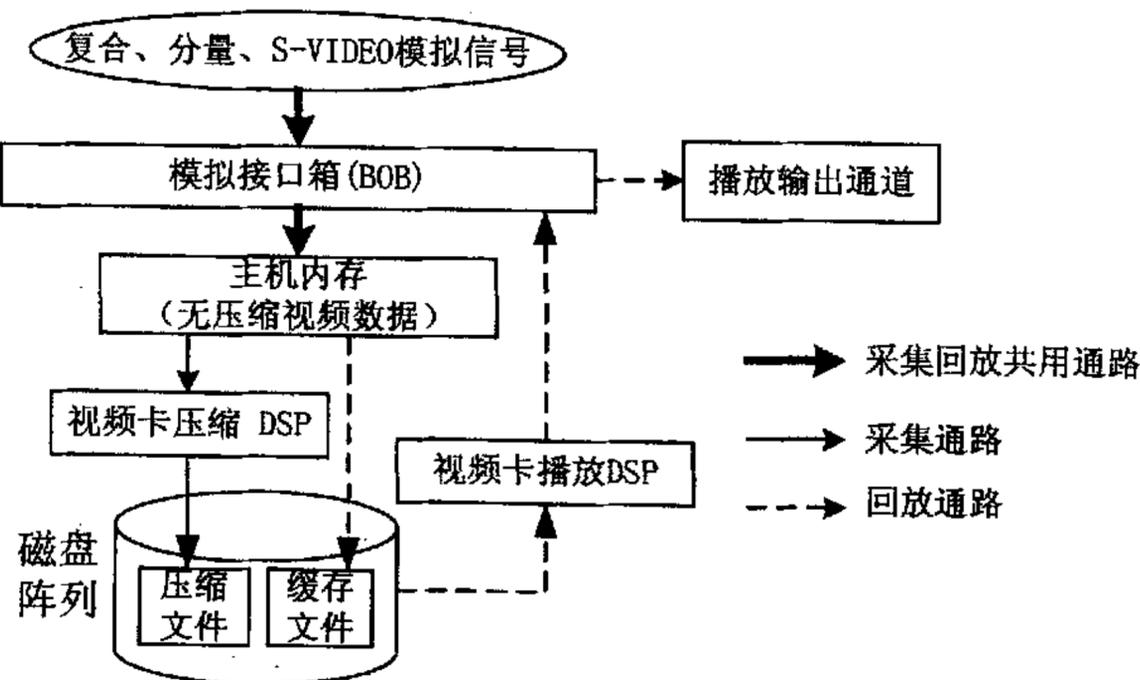


图 4.6 磁盘缓存无压缩音视频数据实现延迟播放

在这种方式下，采集的过程不变，原始无压缩数据直接经采集卡压缩成 MPEG-2 数据存入文件，而延迟播放则需经过内存缓存->磁盘缓存->播出的 IO 路径。由于缓存音视频数据和采集要同时进行磁盘写入操作，特别是在边延迟播放边采集的情况下，还要对缓存在磁盘上的视频数据进行读取回放，磁盘的 I/O 速度将成为严重的瓶颈，以压缩比 10:1 计算，在不需延迟播放的情况下对磁盘写速度的要求为：

$$8294400(\text{bit}/\text{Frame})/10 \times 25(\text{Frame}/\text{s}) = 1036800 \approx 2.48\text{MB}/\text{s}$$

在利用磁盘缓存进行延迟播放的情况下，磁盘缓存写入和读取回放的速度要求为：

$$8294400(\text{bit}/\text{Frame}) \times 25(\text{Frame}/\text{s}) / 8 = 24.72\text{MB}/\text{s}$$

缓存回放要求同时支持 $24.72 + 2.48 = 27.20\text{MB}/\text{s}$ 的磁盘写和 $24.72\text{MB}/\text{s}$ 的磁盘读速度，而磁盘的读写速度一般为 $30\text{MB}/\text{s}$ 左右，在 I/O 速率不足时，播放和采集都会出现严重的丢帧现象，因此采用这种方式虽然没有延迟时间限制，但是会出现严重的丢帧现象，实际中不宜采用。

(3) 以压缩的音视频数据缓存在磁盘的方式实现延迟播。通过以上的分析可知，对无压缩数据处理时间占总处理时间的比率约为： $49.44 / 51.92 \approx 95.2\%$ 。可见出现 I/O 瓶颈的主要原因是因为无压缩音视频数据的磁盘缓存和回放对磁盘的 IO 处理速度要求过高。如何处理无压缩音视频数据成为解决问题的关键，为此我们采用了采集和播放共享压缩视频数据的方式，在这种方式下，不再对无压缩视频数据进行磁盘缓存，而是对压缩后的音视频数据进行磁盘缓存，延迟播放和采集共享同一个音视频压缩文件，工作方式如图 4.7 所示。

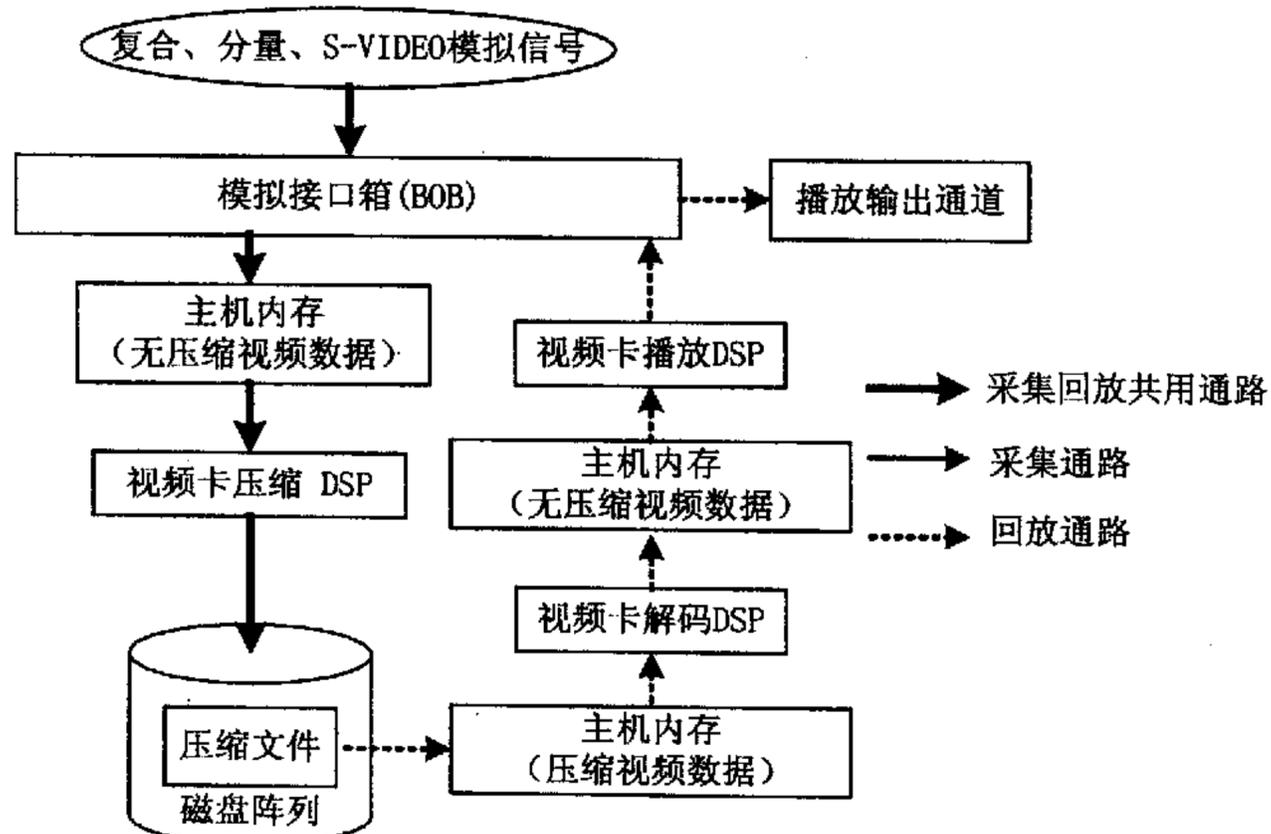


图 4.7 磁盘缓存压缩音视频数据实现延迟播放

在这种工作模式下,采集线程与延迟播放线程互斥地共用一个文件指针,采集线程将实时采集的视频数据压缩后直接存到视频文件中,延迟播放线程负责将写入视频文件的压缩数据读入解压缓冲区,压缩数据经解压后送入播放缓冲区进行播放.可见在这种工作模式下,对磁盘的操作只限于压缩数据的写入和读取,磁盘只需要同时支持 2.48MB/s 的写速度和读速度,远远低于一般磁盘的最大 I/O 速度.该工作方式需要系统硬件同时提供解码和编码功能,并且要为播放另外分配两个缓冲区:压缩缓冲区和播放缓冲区,由于两个缓冲区的缓存的数据量不大,系统可以满足这一要求.由于磁盘容量大,缓存的又是压缩的音视频数据,因此这种方式不存在延迟时间限制的瓶颈,延迟时间可由 5 秒到 30 分钟以上,最长时间由磁盘容量确定.而且这种方式的实现对系统要求不高,具有较好的性价比.

(4) 以压缩的音视频数据缓存在内存的方式实现延迟播.此种方式将压缩的音视频数据缓存在内存里,也会因延迟时间的长度增长而增加对系统内存配置的要求.同样以压缩比 1:10 计算,延迟播放 1 分钟,则

$$8294400(\text{bit}/\text{Frame})/10 \times 25(\text{Frame}/\text{s}) \times 60(\text{s}) = 1244160000(\text{bit}) \approx 145\text{MB}$$

而系统还要配置播放缓存与解压缩缓存,以缓存 40 帧为例,需配置内存

$$8294400(\text{bit}/\text{Frame})/8 \times (40+4) = 43.5\text{MB}$$

即延迟 1 分钟,系统内存容量需要增加 189MB,若延迟 10 分钟,则系统内存容量需增加

$$145\text{MB} \times 10 + 43.5\text{MB} = 1494\text{MB} = 1.45\text{GB}$$

而系统本身运行的内存开销还没计算在内,因此这种方式也存在延迟时间限制的瓶颈.对系统要求也较高,性价比显然不如第三种方式.

2) 磁盘缓存的延迟播放

通过以上的分析比较,系统采用第三种方式实现延迟播放.其程序流程图如图 4.8 所示.当采集开始后,采集线程负责在磁盘上创建共享压缩文件,并返回共享的文件指针.当播放开始后,播放线程和采集线程互斥地使用磁盘缓存文件,分别进行数据写入和数据读取操作.由于写入和读取的文件位置不同,播放线程与采集线程分别有各自的访问文件指针.延迟播放程序播放完毕后,负责完成文件的关闭操作.

```
CCriticalSection FileOperationLock; //临界区互斥锁
void PlayBack(){ //播放程序
    FileOperationLock.Lock();//进入临界区,加锁
    从磁盘读取一帧视频数据;
    FileOperationLock.Unlock();//退出临界区,开锁
}
```

```
void CaptureVideo{ //采集程序
    FileOperationLock.Lock();//进入临界区, 加锁
    写一帧压缩视频数据到磁盘文件;
    FileOperationLock.Unlock();//退出临界区, 开锁
}
```

4.3 HOLD 与 TAKE 的调度

播放控制的另外一项功能是实现 HOLD 与 TAKE 功能, HOLD 命令负责保持当前播放节目的播放状态直到发生 TAKE 命令, TAKE 命令负责立即停止当前节目的播放, 并以尽可能快的速度调度下一节目的播出。由于控制的特殊性, 对这两个命令必须定义特殊的命令格式, 并加以特殊处理。

```
/*端口定义*/
#define PORT 0x4000
/*命令定义*/
#define HOLD 4001
#define TAKE 4002
```

1) 通过共享控制变量实现 HOLD 控制功能

HOLD 和 TAKE 命令是配合使用的, HOLD 功能主要是保持当前节目的播放状态, 或者在节目播放完毕的时候保持住最后一帧, 保证在播放下一节目前不出现黑场。实现方法 (1): 由于允许节目单实时修改, 我们可以通过修改当前节目播放长度来实现 HOLD 功能。如对于以下节目单:

序号	播放时间	节目名称	入点(h:m:s:ms)	出点(h:m:s:ms)
1	12:00:00	综艺大观	00:00:00:00	00:30:00:00
2	12:30:00	广告一	00:00:00:00	00:05:00:00

要保持(HOLD)播放第一条节目, 只需要修改该节目的播放长度。

序号	播放时间	节目名称	入点(h:m:s:ms)	出点(h:m:s:ms)
1	12:00:00	综艺大观	00:00:00:00	03:00:00:00

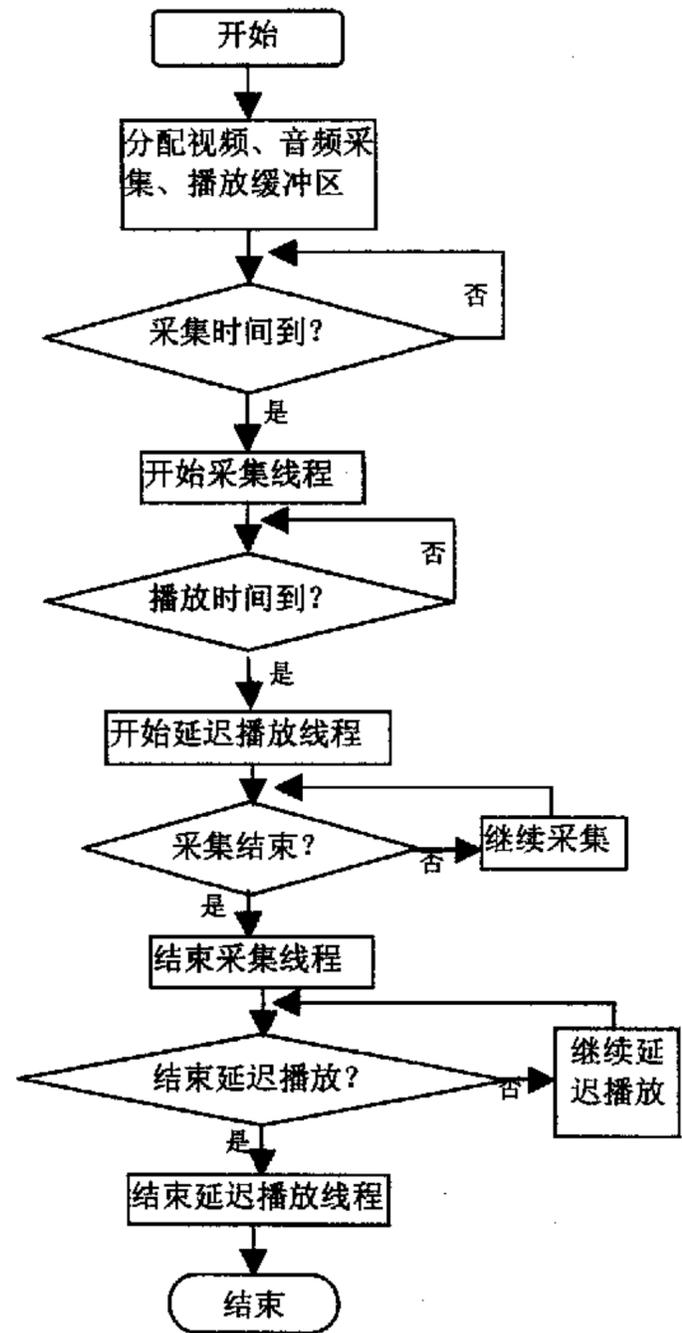


图 4.8 延迟播程序流程图

```
void CaptureVideo{ //采集程序
    FileOperationLock.Lock();//进入临界区,加锁
    写一帧压缩视频数据到磁盘文件;
    FileOperationLock.Unlock();//退出临界区,开锁
}
```

4.3 HOLD 与 TAKE 的调度

播放控制的另外一项功能是实现 HOLD 与 TAKE 功能, HOLD 命令负责保持当前播放节目的播放状态直到发生 TAKE 命令, TAKE 命令负责立即停止当前节目的播放,并以尽可能快的速度调度下一节目的播出。由于控制的特殊性,对这两个命令必须定义特殊的命令格式,并加以特殊处理。

```
/*端口定义*/
#define PORT 0x4000
/*命令定义*/
#define HOLD 4001
#define TAKE 4002
```

1) 通过共享控制变量实现 HOLD 控制功能

HOLD 和 TAKE 命令是配合使用的, HOLD 功能主要是保持当前节目的播放状态,或者在节目播放完毕的时候保持住最后一帧,保证在播放下一节目前不出现黑场。实现方法(1): 由于允许节目单实时修改,我们可以通过修改当前节目播放长度来实现 HOLD 功能。如对于以下节目单:

序号	播放时间	节目名称	入点(h:m:s:ms)	出点(h:m:s:ms)
1	12:00:00	综艺大观	00:00:00:00	00:30:00:00
2	12:30:00	广告一	00:00:00:00	00:05:00:00

要保持(HOLD)播放第一条节目,只需要修改该节目的播放长度。

序号	播放时间	节目名称	入点(h:m:s:ms)	出点(h:m:s:ms)
1	12:00:00	综艺大观	00:00:00:00	03:00:00:00

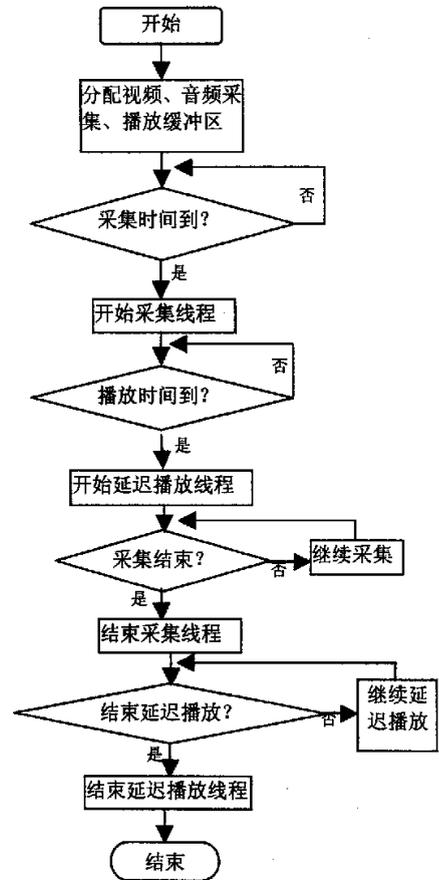


图 4.8 延迟播放程序流程图

2 15:30:00 广告一 00:00:00:00 00:05:00:00

可见，通过修改节目单，节目“综艺大观”的播放时间由原来的 30 分钟变为了 3 个小时，可以预见该节目不会超过 3 个小时，在 TAKE 命令发出前将一直保持播放状态。HOLD 结束后，只需要重新修改节目单就可以了。如果使用这种方式，甚至可以不需专用的 HOLD 和 TAKE 命令，就可以实现 HOLD 和 TAKE 的功能。但是这种方式缺乏灵活性，需要用户临时修改节目单，对用户要求较高。并且如果实际节目素材过短，由于播放出现越界，不加以适当处理，容易造成播出故障，而且 HOLD 和 TAKE 命令一般都是在紧急情况下发出的，要求立即加以处理，通过节目单修改很难做到实时响应，为此我们还是采用了特殊命令、特殊处理的方式，对于 HOLD 命令，我们主要通过共享以下全局变量的方式来实现，即实现方法（2）。

```
extern BOOL HoldCommand; //HOLD 命令有效标志
extern BOOL TakeCommand; //TAKE 命令有效标志
extern LONG NewProgLength;//新的节目播放长度
extern LONG NewProgLoop; //新的节目播放次数
```

通过判断命令有效位和播放长度以及循环次数我们可以在播放程序里做到实时响应各种变化。HOLD 的实现流程如图 4.9 所示。

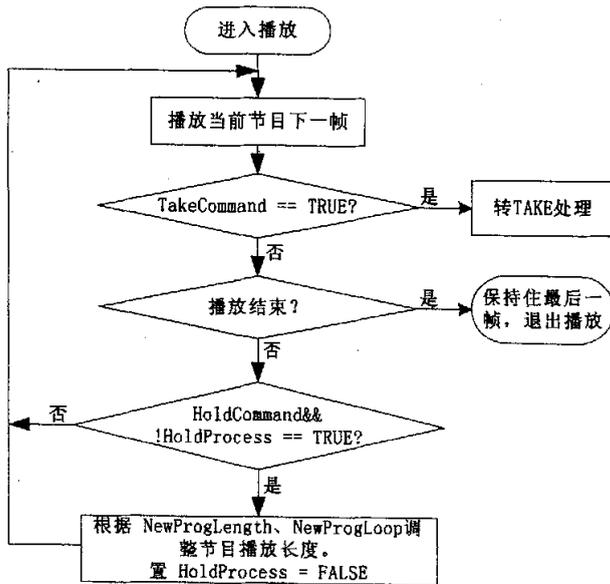


图 4.9 HOLD 功能流程图

(2) 通过保留预取时间实现 TAKE 功能的平滑切换

TAKE 功能要求立即中断当前节目播放，并立即调下一节目播出，该功能必须达

到以下三个指标：一要及时、二要保证节目切换平滑、三要保持节目单切换后的一致性。从命令发出到切换的时间(T_t)取决于下面两个因素：网络延迟(N_t)和节目的预取时间(P_t)，一般有： $T_t = N_t + P_t$ 。为保证切换的及时性，视频服务器在接到 TAKE 命令时只需要考虑预取时间(P_t)就可以了，TAKE 节目的播放时间(S_t)为当前时刻(C_t)加节目预取时间(P_t)： $S_t = C_t + P_t$ ， S_t 时刻之前仍旧保持 HOLD 节目的播放状态，直到下一节目预取了足够的播放数据才转入 TAKE 节目的播放。此时 HOLD 节目的结束时间和播放长度也要进行实时调整，以充分保证节目切换的平滑性。

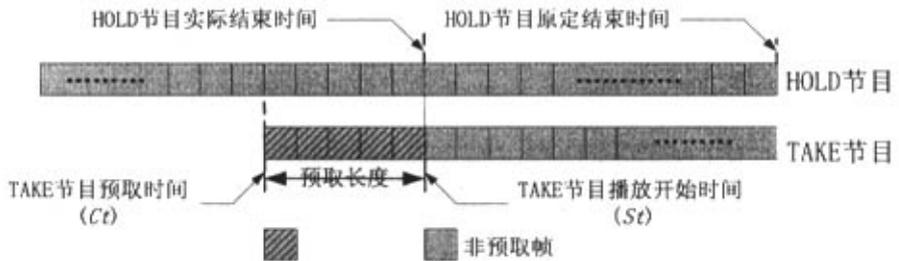


图 4.10 TAKE 节目的平滑切换

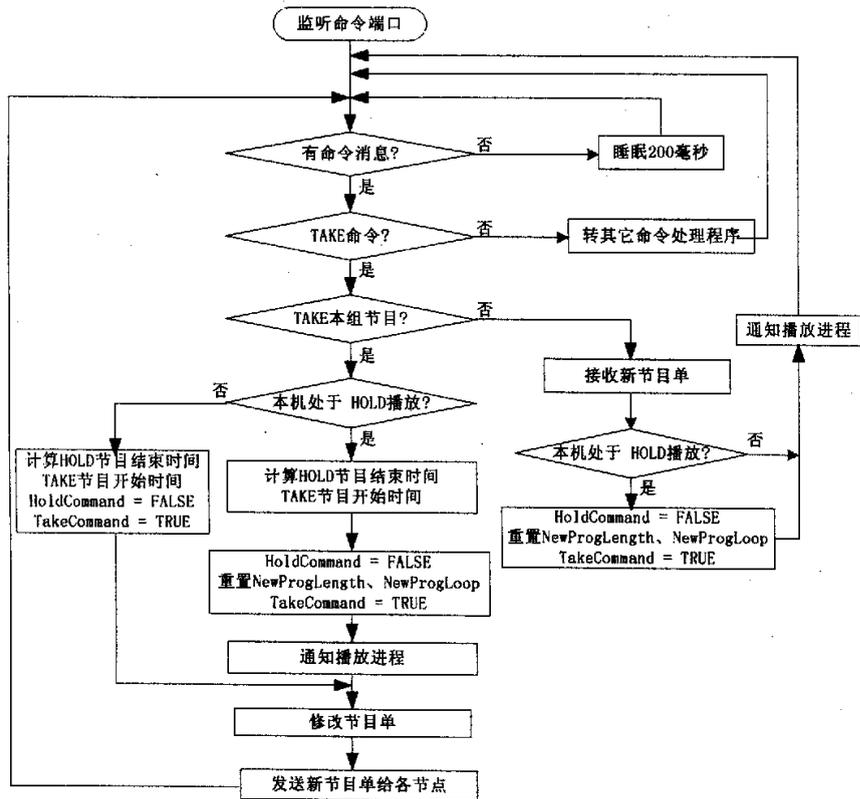


图 4.11 TAKE 功能流程图

所有节目的播放时间将以 TAKE 节目的播放时间(S_i)为基准进行调整。调整后的节目单将重新发给编目工作站作为新节目单使用。可见,这种方式在不考虑网络延迟的基础上,可以充分保证节目切换的实时性。调度的流程图如图 4.11 所示。

4.4 带格式转换的实时播放

对于播放节目文件,要求播放系统能播放的文件格式越多越好,而对于视频卡来说一般只能解码一种格式,如 mpeg2,那么为了实现这些文件格式节目混合时的正常播放(如前一个节目是 DVD 格式即 VOB 文件,后一节目是 VCD 格式即 DAT 文件),有两种办法可实现,一种是在采集过程中将其转换成系统能播放的格式(以 MPEG2 压缩的 AVI 文件),另一种是在播放前一节目的过程中提前将下一待播文件转换成系统能播放的文件格式。这两种方式都有需要的情况,而且它们可共用一个文件格式转换的子过程。

格式转换可以分层次实现,一种是通过硬件电路来底层实现格式转换,如文献^[95-98]介绍的方法,另一种是应用现有工具软件来进行格式转换。若通过底层电路来实现,则付出的代价会很高,若通过工具软件来实现,则很难达到实时性,因此有必要研究一种在底层实现的纯软件的格式转换子过程,本节将讨论文件格式转换子过程的实现要点。

4.4.1 DAT 文件格式到 MPEG 文件格式的转换

VCD 光盘上的音视频文件以 DAT 文件为后缀。DAT 文件用 RIFF(资源交换文件格式)格式将 MPEG-1 系统流数据原样封装在数据段中。VCD 的光盘是 CD-ROM Mode2/XA Form2 格式,允许将计算机数据、压缩声音数据、静态图象或电视图象数据交错放在同一条光道上,其基本单位是 2352 字节大小的扇区^[99],如图 4.12 所示:

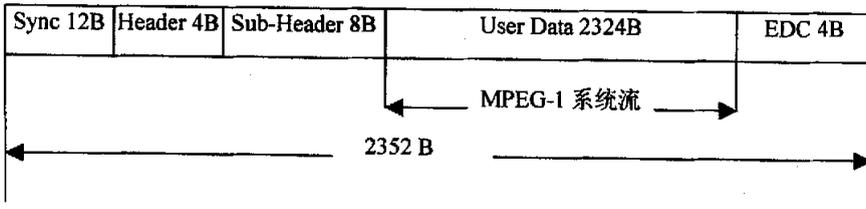


图 4.12 CD-ROM Mode2/XA Form2 扇区图

由图 4.12 可知,除去扇区头的 24 字节和扇区尾的 4 字节 EDC 数据,中间的用户数据区就是 MPEG-1 系统流。当然有些用户数据区还存放一些额外的数据,排除这些用户数据区,将所有标记为音频或视频的用户数据区按序连接,即可得到完整的 MPEG-1 系统流。Sub-Header 第二个和第三个字节指示了用户数据的属性(如视频、

所有节目的播放时间将以 TAKE 节目的播放时间(S_i)为基准进行调整。调整后的节目单将重新发给编目工作站作为新节目单使用。可见,这种方式在不考虑网络延迟的基础上,可以充分保证节目切换的实时性。调度的流程图如图 4.11 所示。

4.4 带格式转换的实时播放

对于播放节目文件,要求播放系统能播放的文件格式越多越好,而对于视频卡来说一般只能解码一种格式,如 mpeg2,那么为了实现这些文件格式节目混合时的正常播放(如前一个节目是 DVD 格式即 VOB 文件,后一节目是 VCD 格式即 DAT 文件),有两种办法可实现,一种是在采集过程中将其转换成系统能播放的格式(以 MPEG2 压缩的 AVI 文件),另一种是在播放前一节目的过程中提前将下一待播文件转换成系统能播放的文件格式。这两种方式都有需要的情况,而且它们可共用一个文件格式转换的子过程。

格式转换可以分层次实现,一种是通过硬件电路来底层实现格式转换,如文献^[95-98]介绍的方法,另一种是应用现有工具软件来进行格式转换。若通过底层电路来实现,则付出的代价会很高,若通过工具软件来实现,则很难达到实时性,因此有必要研究一种在底层实现的纯软件的格式转换子过程,本节将讨论文件格式转换子过程的实现要点。

4.4.1 DAT 文件格式到 MPEG 文件格式的转换

VCD 光盘上的音视频文件以 DAT 文件为后缀。DAT 文件用 RIFF(资源交换文件格式)格式将 MPEG-1 系统流数据原样封装在数据段中。VCD 的光盘是 CD-ROM Mode2/XA Form2 格式,允许将计算机数据、压缩声音数据、静态图象或电视图象数据交错放在同一条光道上,其基本单位是 2352 字节大小的扇区^[99],如图 4.12 所示:

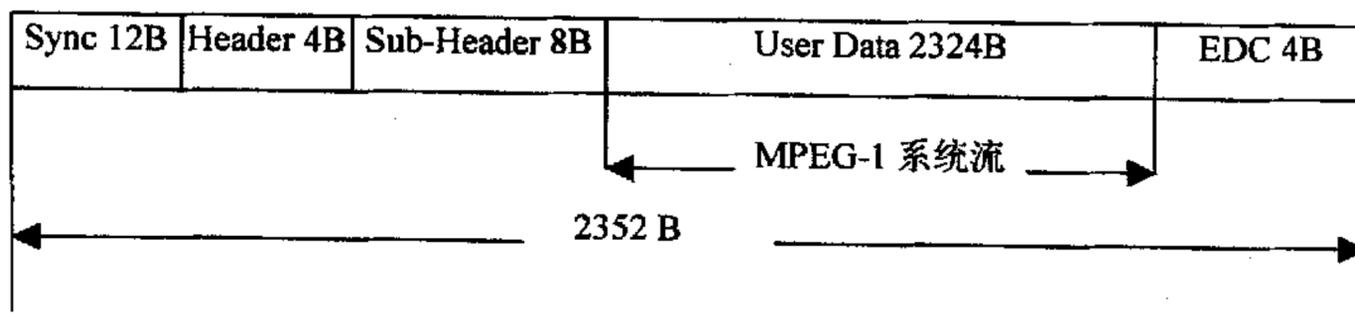


图 4.12 CD-ROM Mode2/XA Form2 扇区图

由图 4.12 可知,除去扇区头的 24 字节和扇区尾的 4 字节 EDC 数据,中间的用户数据区就是 MPEG-1 系统流。当然有些用户数据区还存放一些额外的数据,排除这些用户数据区,将所有标记为音频或视频的用户数据区按序连接,即可得到完整的 MPEG-1 系统流。Sub-Header 第二个和第三个字节指示了用户数据的属性(如视频、

音频、静止图象等), 只需将属性为视频或音频的用户数据提取出来即可。

1) 实现的类结构有

(1) CQTompg

成员变量:

- void (*ShowPercent)(int percent); //函数指针, 转化过程中调用该指针指示的函数在界面上显示完成百分比。

成员函数:

- void SetSowPercentFun(void(*fun)(int)); //设置完成百分比显示的回调函数, 即给 ShowPercent 赋值
- FILETYPE GetFileType(TCHAR *filename); //判断输入的文件类型, 可以判断是不是 DAT 文件
- void DAT2mpg(TCHAR *dat_file, TCHAR *mpg_file); //完成从 DAT 文件到 MPG 文件的转化。
- __int64 SetFilePointerEx (int hf, __int64 distance, DWORD MoveMethod); //设置文件的当前位置

(2) CTompgDlg

成员变量:

- CProgressCtrl m_percent; //转化百分比显示的进度条控制变量。
- CString m_dat_name; //输入的 DAT 文件名。
- CString m_mpg_name; //输出的 MPG 文件名。

成员函数:

- void SetPercent(int percent); //设置显示完成百分比的回调函数。
- virtual void OnOK(); //在该函数中, 启动线程 dat2mpgThread 进行文件转化。
- afx_msg void OnDatBrowse(); //选取输入的 DAT 文件。
- afx_msg void OnMpgBrowse(); //选取输出的 MPG 文件名。

2) 函数 CQTompg::DAT2mpg(TCHAR *dat_file, TCHAR *mpg_file)流程为:

(1) 打开输入的 DAT 文件, 获取 DAT 文件大小, 创建输出的 MPG 文件。

(2) 检查 DAT 文件头, 确认是 RIFF 文件, 类型为 CDXA

(3) 进入循环, 在每一次循环中:

①读取 2352 字节数据;

②忽略 12 个同步字节及 4 字节扇区地址;

③读取 8 字节数据类型 Sub-Header, 判断用户数据属性;

④如果为 MPEG-1 数据, 将 2324 字节 MPEG 数据写到 MPG 文件里, 否则跳过

2324 字节用户数据;

⑤跳过 4 字节校验数据 EDC;

⑥文件未读完则重复①, 否则转换完毕。

4.4.2 VOB 文件到 MPEG 文件的格式转换

若播放系统希望实时播放 DVD 文件, 则需要将 DVD 文件实时转换为 MPEG 文件。DVD 文件以 VOB 为后缀, VOB 文件实际上是 MPEG-2 节目流的一个应用, 它在 MPEG-2 节目流的语法的基础上加一些限制: 考虑到 DVD 扇区大小为 2048 字节, 限定每个组 (Pack) 的大小也为 2048 字节, 使每一扇区恰好容纳一个组。也就是说, 直接 I/O 的基本单位 (扇区) 与 VOB 文件的逻辑单位 (即组 Pack) 一致, 这种组与扇区大小的一致加快了处理速度。

VOB 文件的基本单位是组, 共有 4 种类型: Video Pack、Audio Pack、Sub-picture Pack 和 Navigation Pack。前三种类型的组都只包含一种 PES 分组 (Packet), 分别为 Video Packet、Audio Packet 和 Sub-picture Packet; Navigation Pack 中除了包含一个系统首部外, 还有两种携带控制和搜索信息的 PES 分组。所要提取的视频流在 Video Pack 中, 音频流在 Audio Pack 中, 其语法格式如图 4.13 所示。一个 VOB 文件只能有一个视频流, 故只要将所有的 Video Pack 的 Pack header 和 PES Packet header 去掉, 串连起来即可得到视频基本流。而 DVD 音频格式有多种, 是通过 PES Packet header 中的流 ID 标识的, 其中 AC3 以私用流 1 的形式出现, 流 ID 为 0xBD。DVD 中支持多达 8 个的 AC3 音频子流 (一般一种子流对应一种语言), 由子流 ID (0x80~0x87) 识别。欲提取某个特定的 AC3 音频子流, 首先要跳过 Pack header 和 PES Packet header, 接着还需对 AC3 头部的子流 ID 进行比较, 相等才是所需的音频子流数据。

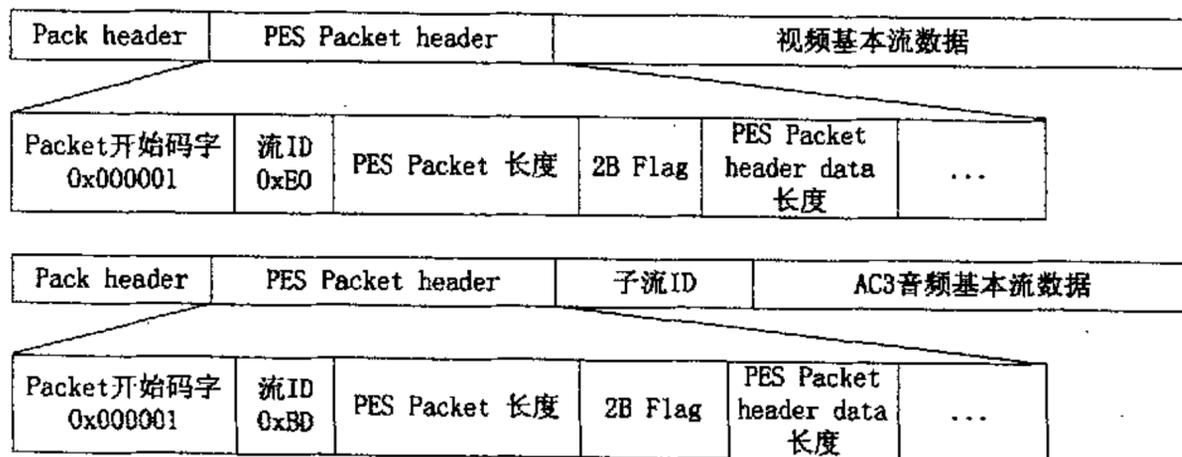


图 4.13 DVD Video Pack 和 Audio Pack 语法

1) 实现 VOB 文件转换成 MPEG 文件的主要数据结构

(1) 全局变量

①share_buf_access *buf_between_demux_mux;//demultiplex和multiplex之间的视频环行缓冲区结构。

②share_buf_access *buf_between_demux_ac3;//demultiplex和ac3 decoder之间的环行缓冲区结构。

③share_buf_access *buf_between_ac3_mp2;//ac3 decoder和mp2 encoder之间的环行缓冲区结构。

④share_buf_access *buf_between_mp2_mux;//mp2 encoder和multiplex之间的环行缓冲区结构。

II) 全局函数

① Demuxthread: 从VOB文件中提取出视频和音频数据分别放到buf_between_demux_mux和buf_between_demux_ac3缓冲区中。

② ac3thread: 从buf_between_demux_ac3中获取音频数据, 解码为WAVE音频, 放到buf_between_ac3_mp2缓冲区中。

③ mp2thread: 从buf_between_ac3_mp2中获取WAVE数据, 编码为mp2音频, 放到buf_between_mp2_mux缓冲区中。

④ OutPutStreamThread: 把buf_between_demux_mux中的视频数据和buf_between_mp2_mux中的音频数据复合为mpg文件, 存到磁盘中。需要用到VideoScanThread和AudioScanThread输出的信息, 故OutPutStreamThread的速度受到两个扫描线程的限制。

⑤ VideoScanThread: 扫描视频数据, 获取信息存为临时文件, 为OutPutStreamThread所用。VideoScanThread先调用scan_video获取视频数据开始部分的一些参数。输入的数据来源于buf_between_demux_mux, VideoScanThread扫描过后的视频数据才能为OutPutStreamThread所用。

⑥ AudioScanThread: 扫描mp2音频数据, 获取信息存为临时文件, 为OutPutStreamThread所用。输入的数据来源于buf_between_mp2_mux, AudioScanThread扫描过后的音频数据才能为OutPutStreamThread所用。

⑦init_buffer_access (share_buf_access *access, DWORD buf_len): 分配大小为buf_len的环行缓冲区, 并初始化该缓冲区。

⑧size_t write_to_buffer(share_buf_access *access,char *src,size_t count): 将src为首址, 大小为count的数据写入access所指向的环行缓冲区中, 如果空间不够则睡眠, 等待消费者线程读取后数据后唤醒。

⑨size_t read_from_buffer(share_buf_access *access,char *des,size_t count): 从access所指向的环行缓冲区中读取大小为count的数据写入des所指的缓冲区, 如果没有足

够的数据可以读取则睡眠，等待生产者线程写入数据后唤醒。

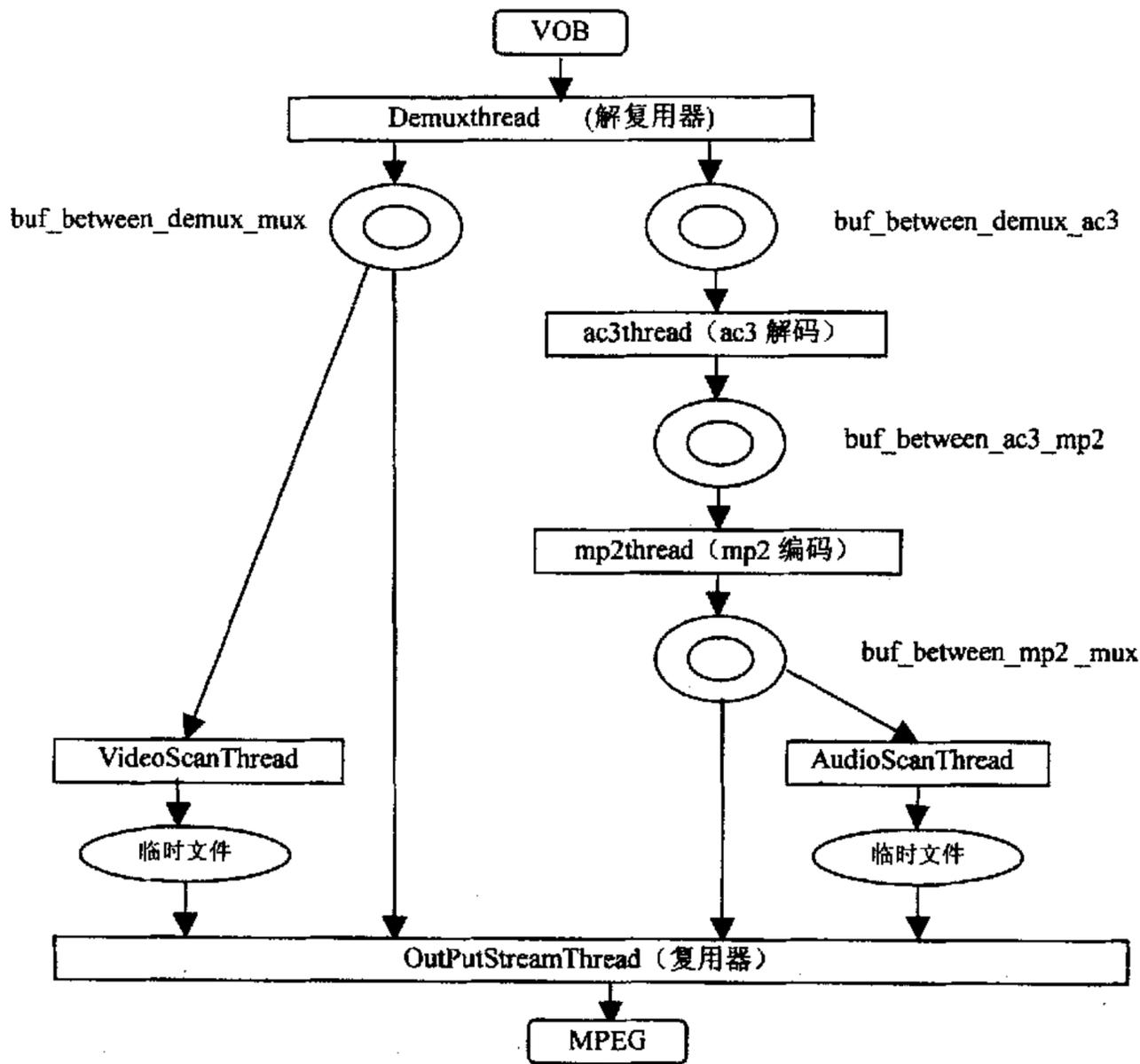


图 4.14 VOB 文件转换成 MPEG 文件的函数调用图

⑩ `size_t scan_read_from_buffer(share_buf_access *access, bool *scan_read_tag, DWORD *scan_read_pos, char *des, size_t count)`: 由于 `OutPutStreamThread` 和 `VideoScanThread` 的视频输入缓冲区都是 `buf_between_demux_mux`，`OutPutStreamThread` 和 `AudioScanThread` 的音频输入缓冲区都是 `buf_between_mp2_mux`，所以 `VideoScanThread` 和 `AudioScanThread` 用该函数进行读环行缓冲区，即两者将各自独立的读指针作为参数 `scan_read_pos` 传入。

如图 4.14 所示，是 VOB 文件转换为 MPEG 文件时（假定只提取第一个 AC3 音频子流），以上这些函数的调用流程。

2) Demuxthread 解复用器

其中 `Demuxthread` (解复用器) 是从 VOB 文件中提取出视频和音频数据分别放到 `buf_between_demux_mux` 和 `buf_between_demux_ac3` 缓冲区中，其流程如下：

- (1) 读取 2048 字节数据;
- (2) 跳过 14 字节组首部 Pack header;
- (3) 读取 Packet 开始码字, 如等于 0x000001E0, 则根据 PES Packet header data 长度跳过 PES Packet header, 将剩下的数据作为视频基本流输出到 buf_between_demux_mux;
- (4) 如果开始码字为 0x000001BD, 则跳过 PES Packet header, 接着读取一个字节长度的子流 ID, 判断子流 ID-0x80 是否等于所需的 AC3 音频子流 ID (此处为 0)。如相等, 则剩下的数据属于第一个 AC3 音频子流, 将其输出到 buf_between_demux_ac3 缓冲区;
- (5) 不符合以上情况且未完成, 则重复 (1)。

如表 4.1 所示是通用的 MPEG-2 解复用器与上述针对 VOB 文件特点的 Demuxthread 解复用器提取音视频流时的性能比较结果。采用的通用解复用器是 Elecard MPEG2 Demultiplexer, 机器配置为 PIII 866、7200RPM Segate IDE 硬盘存储器。

表 4.1 通用解复用器与 VOB Demuxthread 解复用器提取音视频流的时间比较

文件名	文件大小	Elecard MPEG2 Demultiplexer	VOB 解复用器耗费时间
Matrix.vob	115,658,752B	19s	14s
Vts_01_1.vob	1,073,463,296B	108s	79s

由表 4.1 可知, 针对 VOB 特点的提取音视频流方法要比通用解复用器提取音视频流方法的效果好。这是因为通用解复用器需要考虑变长 Pack, 在定位 Pack 边界问题上要耗费更多时间。而 DVD 扇区大小为 2048 字节, 限定每个组 (Pack) 的大小也为 2048 字节, 使每一扇区恰好容纳一个组。我们读写文件的方法都采用直接 I/O 方法, 读写文件的基本单位 (扇区) 与 VOB 文件的逻辑单位 (即组 Pack) 一致, 这样, 因不需考虑组 Pack 变长, 处理速度就加快了, 提取音视频流的时间就缩短了。

3) 编码优化

从实现 VOB 文件转换成 MPEG 文件的主要数据结构中, 和图 4.14 的函数调用流程可看出, 本系统对 MPEG Layer II 进行了软件音频编码, 为满足实时转换必须尽量缩短转化系统中的转化时间, 而编码时间耗用时间比较长, 所以必须对编码进行优化。

MPEG 音频编码使用子带编码方法, 先将每一帧输入的 PCM 音频信号经过 32 个子带滤波器组进行子带滤波, 宽频带的音频信号分成 32 个等带宽的子带, 同时通过心理声学模型来确定每个子带的信掩蔽 (SMR), 然后, 比特分配模块根据 SMR 控制每个子带的量化参数, 使得在满足总比特率条件下, 感知失真最小。最后, 已编码的子带样值和边信息按照一定的格式打包, 形成标准的位流输出^[100-101]。编码的结构

图如图 4.15 所示。

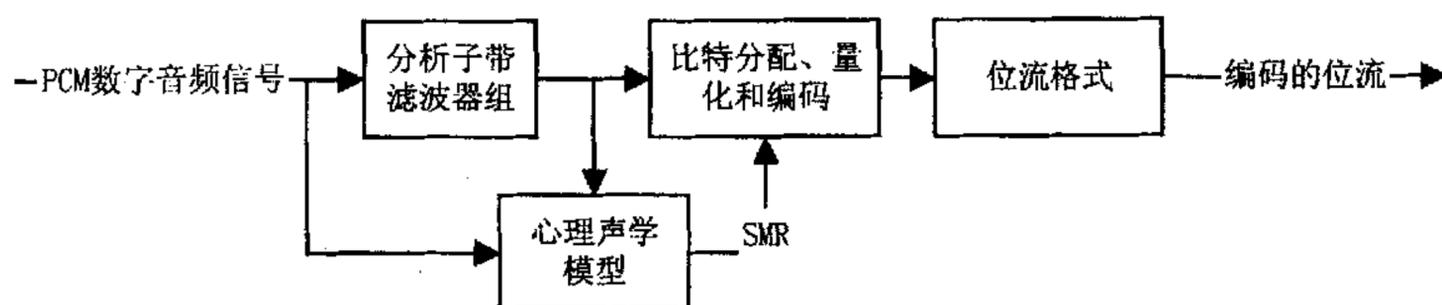


图 4.15 MPEG 音频编码器模型

(1) 子带滤波和心理声学模型的快速算法

MPEG 建议的音频编码方案中，计算量主要集中在分析子带滤波和心理声学模型，分别占总运算量的 21.9% 和 64.9%。要想实现实时运算就必须对标准建议的这两部分算法进行优化，为此，利用余弦函数的对称性和快速余弦变化使子带滤波的计算得以简化；同时，用查表法减少了对数加法的计算复杂度，优化了心理声学模型的计算，下面详细讨论这两个模块的快速算法。

子带滤波器组将时域数字音频信号变换到 32 个等宽的频带中，MPEG ISO/IEC11172-3 标准给出了进行一次子带分析的方案：

- ① $for(i=511; i \leq 32; i--) X[i] = X[i-32]$ 。
- ② $for(i=31; i \geq 0; i--) X[i]$ = 下一组 PCM 输入采样值。
- ③ 用 512 个系数 $C[i]$ 对 $X[i]$ 进行视窗变换，即 $for(i=0; i \leq 511; i++) Z[i] = C[i] * X[i]$ ， $C[i]$ 可从 MPEG 标准中得到。

- ④ 计算中间变量 $Y[i]$ ，即 $for(i=0; i \leq 63; i++) Y[i] = \sum_{j=0}^7 Z[i + 64j]$ 。

- ⑤ 计算 32 个子带样本值：

$$for(i=0; i \leq 31; i++) S[i] = \sum_{k=0}^{63} Y[k] \cos\left[\frac{(2i+1)(k-16)\pi}{64}\right]$$

- ⑥ 输出 32 个子带样值 $S[i]$ ， $i=0 \dots 31$ 。

子带滤波算法的关键是第 5 步，如果未经过算法优化，则在该步需要进行 $64 \times 32 = 2048$ 次带乘法和 $63 \times 32 = 2016$ 次加法。用上述的标准算法进行一次子带分析需要做 2560 次乘法和 2463 次加法运算。每次子带分析输入 32 个 PCM 音频采样，32 个滤波器每个各产生一个子带采样。MPEG Layer II 每帧对应 1152 个采样，需进行 36 次子带分析，故每帧共需要 92160 次乘法和 88668 次加法运算，可见运算量很大。为

减少复杂度，利用余弦函数的周期性和快速傅利叶变化 (FFT)，可极大地降低计算复杂度^[102-105]。

快速算法的关键是对下式进行简化：

$$S[i] = \sum_{k=0}^{63} Y[k] \cos \left[\frac{(2i+1)(k-16)\pi}{64} \right] \quad (i = 0 \dots 31) \quad (4.7)$$

利用余弦函数的周期性及变量代换法有：

$$\cos \left[\frac{(2i+1)(l-64)\pi}{64} \right] = \cos \left[\frac{(2i+1)l\pi}{64} - (2i+1)\pi \right] = -\cos \left[\frac{(2i+1)l\pi}{64} \right],$$

则式 (4.7) 可化简为：

$$S(i) = \sum_{l=0}^{63} \bar{Y}(l) \cos \left[\frac{(2i+1)l\pi}{64} \right] \quad (4.8)$$

因为 $\cos[(2i+1)l\pi/64]$ 是 $e^{-j\pi(2i+1)l/64}$ 的实数部分，对式(4.8)做如下变换：

$$S(i) = \text{real} \left\{ \sum_{l=0}^{63} \bar{Y}(l) e^{-j\pi(2i+1)l/64} \right\} = \text{real} \left\{ \sum_{l=0}^{63} \bar{Y}(l) e^{-j\pi l/64} e^{-j\frac{2\pi}{64}li} \right\} \quad (4.9)$$

其中的 $\bar{Y}(l)e^{-j\pi l/64}$ 与变量 l 有关而与 i 无关，可看作 64 点的复数序列，则式(4.9)是一个标准的 64 点离散傅立叶变换 (DFT)，可通过快速傅里叶变换 (FFT) 计算^[106]，最后取结果的实数部分，就得到 32 个子带样值 $S_i(n)$ 。

用这种方法计算 32 个子带样值 $S_i(n)$ 所需的乘法运算数为 $2*32*\log_2 64=384$ 次，加法运算次数为 $4*32*\log_2 64=768$ 次，对一帧 MPEG Layer II 进行子带分析总共需要乘法次数为 32256 次，加法次数为 43776 次，比未优化算法分别降低了 65% 和 50.6%。

在实现快速傅里叶变换时，可采用 Intel 的 MMX 技术^[107]。它使用了单指令多数据技术 (SIMD)，以并行方式处理多个数据元素，从而提高了运行速度。

心理声学模型采用掩蔽阀估计模型，根据一帧信号的短时功率谱估计出各子带的掩蔽阈值，最后输出用来控制每个子带量化器参数的信号掩蔽比 (SMR)。在心理声学模型中，掩蔽阈值用 dB 来表示，因而需要将功率谱 a 转到对数功率谱 A 域中： $A = 10 \log_{10} a$ ，对数相加占总运算时间的 32.6%，可通过查表来降低运算的复杂度：

$$10 \log_{10}(10^A + 10^B) = \max(A, B) + 10 \log_{10}(1 + 10^{-d/10})$$

其中 $d = |A - B|$ ，以 d 作为索引查表求表达式的第二项值。这样就使对数运算转化为简单的加法和查表运算，运算时间减少为原来的 21.5%。另外，计算量还体现在计算信号的短时功率谱上，可用 1024 点的快速傅里叶变换 (FFT) 来减少计算量。

(2) 实验验证

为验证上述优化算法的有效性，给出三段音频的编码实验结果。实验环境为

Windows2000, 机器型号为 PIII866MHz 及 265M 内存。输入时无压缩音频参数: 采样率均为 44.1kHz, 16bit/采样, 每声道位率为 128kbps。未优化算法指的是 MPEG ISO/IEC11172-3 标准给出的算法, 优化算法为改进后的算法, 剩余是指编码后剩余时间占音频信号播放时间的百分比。

实验结果如表 4.2 所示。从表中的结果可知, 采用优化算法的编码时间远远小于未优化算法的时间, 故对于转换系统的音频实时编码, 用软件来实现是完全可行的。从表 4.2 还可看出, 采用优化算法的编码时间远小于播放时间, 故在实时的格式转换过程中, 不会出现掉帧现象。优化算法只是对计算速度进行了提高, 对音频的编码质量并没有影响。

表 4.2 MP2 编码优化算法与未优化算法实验结果

音频信号	未优化算法	优化算法	
		时间	剩余
277s 立体声	199s	69.1s	75%
161s 立体声	123s	41.7s	74%
67s 单声道	45s	12.1s	82%

4) OutPutStreamThread (复用器)

节目流由一个或多个基本流多路复用而成。音频(视频)基本流就是压缩的音频(视频)数据, 它是由音频(视频)访问单元 AU (Access Unit) 序列构成的, 所谓的访问单元, 对视频来说, 是指一帧图象的编码表示, 对音频是指一帧话音采样数据的编码表示。而音频(视频)访问单元解码后就是音频(视频)显示单元 PU (Present Unit)。

节目流的复用主要是在音频和视频 PES 分组的基础上进行。PES 分组由分组首部及其后的分组数据组成, 基本流就存放在分组数据中。来自同一基本流的数据被分成多段分别存储到多个 PES 分组中, 并保持串行顺序, 通过 PES 分组首部的一个 32 位开始码字来标识该 PES 分组数据所属的基本流。PES 分组从不包含来自 1 个以上基本流的数据, 所携带的数据保持基本流的字节顺序。因此, 在去掉分组首部后, 将带有共同流标识符的所有 PES 分组的分组数据级联, 就可以恢复整个基本流。PES 分组携带的基本流既可以是压缩的音频数据也可以是压缩的视频数据, 还可以是填补流或私有数据。

复用器^[108]由两部分组成: PES 分组器(包括视频 PES 分组器和音频 PES 分组器)和节目流合成器, 分别完成基本流的打包和复用工作。PES 分组器的作用是将基本流数据打包为 PES 分组, 分别输出到 VIDEO_PES_BUF 和 AUDIO_PES_BUF 中。节目流合成器则将 VIDEO_PES_BUF 中的视频 PES 分组和 AUDIO_PES_BUF 中的音频

PES 分组按照一定的复用策略合成，输出节目流。MPEG-2 系统标准只定义了系统层的语法和语义，而对复用器并没有做严格的定义。也就是说，任何能生成合法系统流并且能保证被正确解码的复用器都是正确的。为了实现节目转换的实时性要求，我们对打包和复用采取了等字节打包方法和根据复用比率进行音视频交错的复用策略。

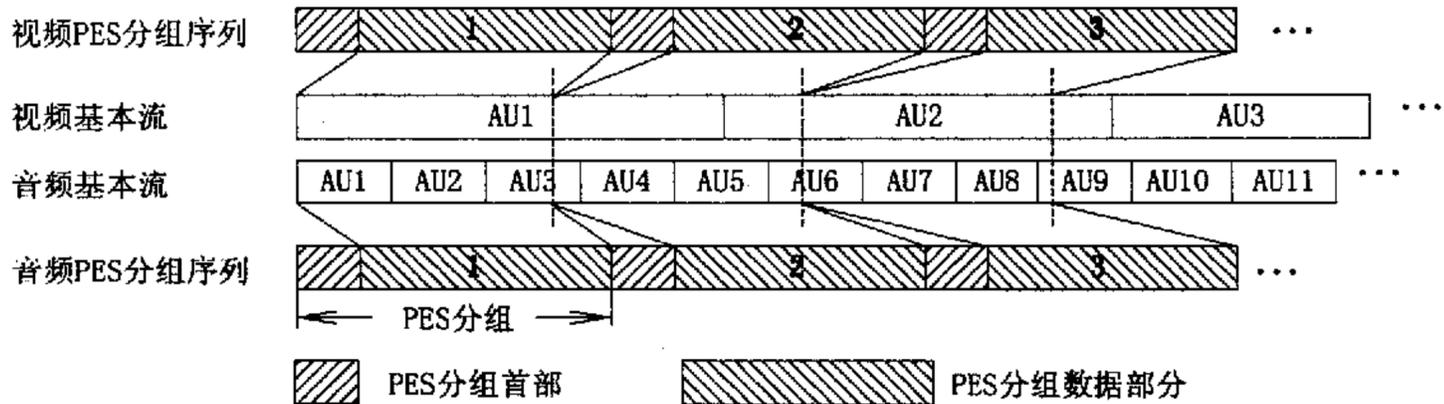


图 4.16 等字节打包示意图

(1) 基本流的等字节打包

对基本流的等字节打包过程描述如下：将基本流分割为很多段，每一段的长度均相等（设长度为 L 字节）。打包的时候，按照顺序取一段作为 PES 分组的数据部分，然后加上 PES 分组首部后形成 PES 分组。此外，我们通过添加填补字节的方法，使得 PES 分组首部长度的相等，这主要是为了后面节目流复用速率的计算方便以及音视频交错的实现方便。经过分组后，基本流就转换成为由长度完全相等的 PES 分组所构成的序列^[109]，其过程如图 4.16 所示，图中的虚线表示对基本流的等字节长度分割。

PES 分组首部包含了用于解码器同步重放的解码时间标签 DTS (Decoding Time Stamp) 和显示时间标签 PTS (Presentation Time Stamp)，分组器的一个重要工作就是计算这些时间标签，并填写到每一个分组首部中去。我们知道，基本流都是由一系列访问单元 AU 组成。访问单元，对视频来说，是指一帧图像的编码表示，对音频，是指一帧话音采样数据的编码表示。由于对基本流进行等长度分段，访问单元的边界和分段边界不一定重合，所以视频帧的开始码字或音频帧的同步字会出现在 PES 分组的数据中。为此，MPEG-2 标准规定，PES 分组首部中的 DTS/PTS 时间标签是依据 PES 分组数据中开始的第一个访问单元的时间标签来确定的，如果分组数据中只包括某个访问单元的中间部分数据（即开始码字或同步字不出现在分组数据中），则分组首部中就不应该包括 DTS/PTS 字段。例如，在图 4.16 中，视频 PES 分组 2 的时间标签指的是视频基本流中访问单元 AU2 的时间标签，音频 PES 分组 2 的时间标签指的是音频基本流中访问单元 AU4 的时间标签，而视频 PES 分组 3 则不出现时间标签字段。

如上所述，PES 分组器为了计算分组首部的时间标签 DTS/PTS，必须明确该时间标签对应的是哪一帧的，为此，分组器还需要知道基本流各帧的大小。在捕获过程中

很容易得知视频帧大小：采集编码卡每次输出一帧数据的同时把该帧的尺寸值加到一个链表中，分组器访问该链表便可确定各帧的大小。同样，MP2 编码器在输出一帧音频帧时，也将该帧大小值加入到一个链表中。在转化系统中，如转化 DVD，视频基本流是原封不动的提取出来，为获取各帧大小，视频基本流必须预先经过一个扫描线程以获取各帧大小值，该扫描线程可以和复用器并行工作，但只有经过扫描线程扫描的视频基本流数据才能为复用器所用。分组器根据各帧大小和基本流的分段长度 L ，即可确定分组首部的时间标签对应的是哪一帧的时间标签。分组器计算该帧的时间标签值并填入 PES 分组首部中。

另外，PES 分组首部信息可预先设置在内存中，打包时根据需要修改必要的值（如 DTS/PTS）。这样可以缓解 CPU 在编码和合成时的数据处理的压力。至于各帧时间标签的计算，下面专门讨论。

(2) 时间标签 DTS/PTS 的计算

在 MPEG-2 标准中定义了一个称为节目流系统目标解码器 (P-STD) 的虚拟解码器来控制复用器生成符合要求的节目流。在 P-STD 中描述了各帧解码显示的时序关系，但是并没有规定如何计算这些时间标签，通常的做法是 DTS/PTS 根据系统时钟的采样值生成。在我们软件实现的复用器中，时间标签是根据 P-STD 的时序要求，由基本流的参数（如 GOP 结构，图象速率，采样率等）计算得到的。这里假设视频基本流和音频基本流都是恒定速率的。

由于 P-STD 忽略了各帧的解码显示延迟，在这种情况下，对音频帧、B 帧图象和低延迟序列的 I/P 帧图象来说，显示时间等于解码时间，即 $DTS=PTS$ ；对于非低延迟序列中的 I/P 帧图象而言，该 I/P 图象的 PTS 等于下一个传送的 I/P 图象的 DTS。I/P 图象的 PTS 与 DTS 之间的差等于 $k+1$ 倍的图象周期（ k 表示在图象帧的编码顺序中，该 I/P 帧与下一个 I/P 帧之间连续的 B 帧数）^[110]。在编码顺序中，每一个访问单元（即编码的音频帧或编码的图象帧）的 DTS 等于前一个访问单元的 DTS 加上一个恒定差值 Δ ， Δ 对于音频和视频而言是不同的，对于音频而言

$$\Delta = \frac{\text{Clock_Frequency}}{\text{采样率}} \times \text{每个音频帧的采样数}, \text{ 对视频而言 } \Delta = \frac{\text{Clock_Frequency}}{\text{图象速率}}.$$

根据上面的介绍，我们总结出基本流各帧的 DTS 和 PTS 的计算公式：

• 对音频基本流：

$$DTS_1 = 0 \quad (\text{当 } n=1)$$

$$DTS_n = DTS_{n-1} + \frac{\text{Clock_Frequency}}{\text{采样率}} \times \text{每个音频帧的采样数} \quad (\text{当 } n > 1)$$

$$PTS_n = DTS_n \quad (\text{当 } n \geq 1)$$

• 对视频基本流:

$$DTS_1 = 0 \quad (\text{当 } n = 1)$$

$$DTS_n = DTS_{n-1} + \frac{\text{Clock_Frequency}}{\text{帧速率}} \quad (\text{当 } n > 1)$$

$$PTS_n = DTS_n \quad (\text{当 } n \geq 1 \text{ 且第 } n \text{ 帧是 B 帧})$$

$$PTS_n = DTS_{n+k+1} = DTS_n + \frac{\text{Clock_Frequency} \times (k+1)}{\text{帧速率}} \quad (\text{当 } n \geq 1 \text{ 且第 } n \text{ 帧是 I 或 P 帧})$$

其中 DTS_n 和 PTS_n 分别表示在基本流中第 n 个访问单元的解码时间标签和显示时间标签, k 表示视频基本流中当前的 I/P 帧与下一个 I/P 帧之间连续的 B 帧数。由于当前帧是 I 帧或 P 帧时, 必须要知道上面所说的 k 值才能计算 PTS, 所以复用器要预先知道 GOP 结构。我们在转换时对视频的编码采用的是规则的 GOP 结构, 或从视频基本流的参数中获取, 所以上述的 k 值是可行的。

例如, 对于特定的 GOP 结构 ($N=12$, 即图象组长度为 12, $M=3$, 即两个 I/P 帧之间连续 B 帧数为 $M-1=2$), 图象周期为 40ms 的视频基本流, 帧的显示顺序为:

1I 2B 3B 4P 5B 6B 7P 8B 9B 10P 11B 12B 13I,

则在编码顺序中根据上述公式计算各帧的 DTS/PTS 得到的结果如表 4.3:

表 4.3 DTS/PTS 计算结果

编码顺序	1I	4P	2B	3B	7P	5B	6B	10P	8B	9B	13I	11B	12B
DTS(ms)	0	40	80	120	160	200	240	280	320	360	400	440	480
PTS(ms)	40	160	80	120	280	200	240	400	320	360	520	440	480

(3) 复用策略

节目流是由一系列有序的组 (Pack) 构成, 而组是由组首部(后跟一个可选的系统首部)和若干 PES 分组构成。因此节目流合成器输出一个组时, 首先生成组首部, 然后按照复合策略交错输出若干个音频或视频 PES 分组。

复用器要保证输出的节目流在解码器中同步重放时缓存既不上溢也不下溢, 因此在复用时就要考虑对视频音频码流进行合理地交错。然而, MPEG-2 标准对此并没有作出具体规定。我们的做法是根据复用比率来进行交错, 复用比率等于视频编码数据率与音频编码数据率的比值。这样做的结果使得时间标签相等或相近的 PES 分组相邻的摆放, 避免了解码器中缓存的溢出^[42,110]。例如, 视频和音频的编码速率分别为 10Mbps 和 192Kbps, 则视频分组与音频分组的复用比率为 $10\text{Mbps}/192\text{Kbps} = 52.08$, 也就是说 1 个音频分组与 52~53 个视频分组间隔放置。为方便计算整个节目流的速率, 我们让每个组内包含常数个 (packet_per_pack) PES 分组。节目流合成器按照复用比率来确定是取视频分组还是取音频分组, 每 packet_per_pack 个 PES 分组加上组首部产生一个组, 输出到缓冲区。至于系统首部, 由于它包含了整个节目流的重要信

息，故按如下的方式出现：在整个节目流的第一个组的组首部之后必须出现，然后按一定的频率（*system_header_ratio*）周期性的出现在一些组的组首部之后（如每 200 个组出现一次）。

为加快速度，我们把节目流的组首部和系统首部事先设置于内存中，需要时经过少量的修改后即可用于输出。此外，我们还做了一些优化，如为了实现位流的长字对齐，所有的 PES 分组都从 32 位边界开始。为达到这个目的，组首部需要填补字节。这样，包含系统首部的组首部长为 32 字节，其中包括 14 字节的组首部和 18 字节的系统首部。不含系统首部的组首部长为 16 字节，其中 14 字节为组首部，2 字节为填补字节。PES 分组长度也以长字为边界，以符合位流长字对齐的要求。

在本算法中，我们假设复用器的输入参数按如下值给定：每个组包含 3 个 PES 分组（*packet_per_pack*=3），PES 分组长度为 2048 字节，系统首部出现的频率为 200，复用比率为 10Mbps/192Kbps=52。

(4) 系统参考时钟 SCR 的计算

节目流合成器输出组首部时，需要填写一个称为系统参考时钟 SCR 的字段，该字段是对节目流进行解码和显示的时间基准，表示该字段最后字节到达系统目标解码器（STD）的时间，系统目标解码器根据 SCR 值调整系统时钟（STC）的瞬时值。在这里，我们通过整个节目流的速率来估算 SCR，下面是系统参考时钟 SCR 的计算公式：

$$SCR_{n+1} = SCR_n + \frac{Clock_Frequency \times Pack_Size}{Byte_Rate}$$

其中 SCR_i 表示节目流中第 i 个组的 SCR；*Clock_Frequency* 为 27MHz；*Pack_Size* 为组的字节大小；*Byte_Rate* 为整个节目流的速率，当音频和视频 PES 分组首部数据大小相等，所携带的数据大小也相等时可以通过下面的公式计算：

$$Byte_Rate = (\text{视频基本流速率} + \text{音频基本流速率}) \times \left(1 + \frac{\text{packet_header_size} + \frac{\text{pack_header_size} + \frac{\text{system_header_size}}{\text{system_header_ratio}}}{\text{packet_per_pack}}}{\text{packet_data_size}} \right)$$

其中 *packet_header_size* 为 PES 分组中去除压缩数据后的长度，*pack_header_size* 为组首部长，*system_header_size* 为系统首部长，*system_header_ratio* 为系统首部出现的频率，*packet_per_pack* 为每个组中包含的分组个数，*packet_data_size* 为 PES

分组中所携带压缩数据的长度。

综上是 DAT 文件及 VOB 文件到 MPEG 文件的格式转换的实现要点, 其中 DAT 文件到 MPEG 文件的格式转换在底层用软件实现较简单, 只需打开文件指针, 丢掉扇区头 24 字节和扇区尾 4 字节, 中间的数据区按序连接即可。而 VOB 文件到 MPEG 文件的格式转换的实现较复杂, 为了实现实时格式转换及播放的实时性, 本节对此子过程的解复用—编码—复用的环节进行了优化, 并通过多线程流水线式调度算法调度格式转换子过程来实现多种不同格式的实时格式转换与播放。DAT 文件及 VOB 文件到 MPEG 文件的格式转换及混合播放, 已经过实验测试达到预期效果。实验结果表明: 针对 VOB 文件特点的解复用器比通用解复用器提取音视频流的耗时要小得多; 采用编码优化算法使得编码时间远小于播放时间及未优化算法的编码时间。

4.5 视频音频同步算法

4.5.1 视频音频同步概述

1) 音视频同步的相关概念^[112]

在多媒体领域, 表现一词指的是向用户提交各种媒体的活动, 对图像和文字来说, 表现意味着数据的显示, 而对于音频和视频分量, 表现指的是数据动态地在听觉和视觉方面的再生。在多媒体应用中, 集成数据的复合表现基于三种对象: 静态、动态、和混合型, 其中混合型更为普遍。静态对象的数据表现主要涉及对象各成分之间的空间组织, 动态对象的数据表现还要考虑时间相关问题, 媒体之间的时间关系是一种同步关系。

媒体对象在时间上的并行、顺序、独立关系, 在对象的时间表现控制上分别对应于并发、串行、异步(任意)执行。对象之间在开始点和结束点指定同步点的同步, 是一种粗略同步, 叫人为同步, 如多媒体幻灯片带口头讲解的复合表示。对象之间的时间关系是连续的同步叫流同步, 如从录制图像和声音得到的视频和音频流之间的同步是流同步。多媒体流同步包括媒体内同步和媒体间同步两个方面。媒体内同步主要是解决连续媒体流的连续性, 是指在单一连接上的单一媒体流内各媒体单元之间存在的时态关系。媒体间同步要维护各种媒体间各个媒体单元之间存在的时态关系, 即在多个连接上的相关媒体流之间存在的时态关系。那么音视频同步指的是动态地再生音视频流数据的时候要保持它们之间存在的时态关系, 它属于媒体间同步, 也包括媒体内同步。

在 MPEG2 中, 解码恢复出来的图像 (Picture) 被称为 PPU (Picture Presentation Unit), 解码恢复出来的语音被称为 APU (Audio Presentation Unit); 它们在码流中的相

应部分叫 PAU(Picture Access Unit)和 AAU(Audio Access Unit)。一般 PPU 和 APU 是不同的、或说不相关的帧周期。例如,一个 Audio 序列,每帧有 1152 个采样,若采样率为 44.1KHz,则帧周期为 26.1ms;而一个 Video 序列,如果帧频为 29.97Hz,则帧周期为 33.76ms。可见 PAU 和 APU 的时间边界并不一样。在编码器中,有一个共同的系统时钟, MPEG2 的系统流(以传输流为例)中,节目参考时钟是这个系统时钟的采样。在系统流中,有 Video 的显示时间标签(Presentation Time Stamp)和 Audio 的显示时间标签,表示 Picture 显示的时间和相应 Audio 回放的时间。显示时间标签也以这个系统时钟为参考,它是工作于 90KHz 的计数器的采样值,用 33bit 表示,可以记录下 24H 内的任何时钟周期。节目参考时钟和显示时间标签都被编码到码流中。相邻的节目参考时钟和显示时间标签间隔一般小于 700ms。解码端可以按照节目参考时钟,通过一个锁相环恢复与编码端一致的本地系统时钟。显示时间标签在 MPEG2 中是以一个理想的解码器(decoder)为基础定出的。这个理想的解码器假设了通道 buffer 永不上溢、下溢(对下溢有些特例),对码流的处理是瞬时的、理想的。因此,如果节目参考时钟和显示时间标签在编码端被正确编码,被无错地存储和传送,在解码端被进行正确的解码,而解码端又以节目参考时钟为基础恢复出与编码端一致的系统时钟,并且在正确的显示时间标签时间显示图像、回放声音,则视频和音频达到同步。

2) 影响音视频同步的因素

一个实际解码器要比理想的解码器复杂得多。在实际的实现中,码流可能会出错;解码处理要花时间。实际上解码器视频和音频同步实现的困难之处在于^[71]:

(1) 被解码的码流不能瞬时地从通道 Buffer 中读出来;解码过程要花时间;显示和回放要花时间;

(2) 基于不同的解码硬件系统,有相应一些同步措施;

(3) 显示时间标签和相应的 P U 在码流中分别处于不同的层中,而系统头信息和基本流的处理在不同的过程中,这在实现中存在着如何找到相配对的显示时间标签和 P U 的问题。

(4) 当解码工作于从模式时,视频同步信号或音频采样频率由外部供给,这时 P U 的输出实际上并不由解码器主动控制;

(5) 码流出错。节目参考时钟和显示时间标签在传送中都可能误码、出错。

以上是解码时影响音视频同步实现的因素,而在播放过程中由于磁盘读取速度、数据总线方式、计算机处理速度、显示缓冲写入速度、磁盘高速缓冲等的不同,可能导致视频音频显示的速度不同,也就是说,某种影响图像显示速度变化的因素就可能导致视频序列的播放速率与采集速率不一致。而模拟音频数字化时,是以固定频率采样形成数字波形音频的,并且音频的数据量较小,因此,固定频率的播放音频无论是

以中断方式或后台 DMA 方式播放，速度都是一致的。所以，当播放环境不同于采集环境时，由于视音频表现的速度不同就可能不同步现象^[113]。

以 PAL 制式的节目为例。音频采用双声道，16 位采样，48000Hz 采样频率，每秒的采样数据为 $16/8 \times 2 \times 48000 = 192000\text{Byte}$ ，播放速度的要求约为 0.19MB/S，有以前的计算结果可知，10:1 的压缩比，视频播放速度大约需要 2.48MB/S 的速度，每播出 187.5KB 的音频数据，就需要同步播出 2539.52KB 的视频数据。当视音频同步播放时，因为音频的数据量较少，播出速度不存在任何问题，但是当读取视频速度变慢(或加快)时就会导致视音频的不同步现象。这种不同步现象在目前流行的 PC 多媒体开发平台上反映较为明显。这些多媒体开发平台开发的多媒体产品，如果使用环境不同于开发环境，动画演播与声音播放就会不同步，从而导致开发的多媒体产品只能是在特定环境下使用，难以推广。我们知道，人类的听觉较视觉敏感，固定频率的声音播放时，暂停及速率或高或低都使人难以接受，因媒体对象的表现时间要以音频的表现时间为基准，视频及其它多媒体对象在表现时要同步于音频对象。

3) 音视频同步的一般方法

对于编码解码中的同步问题可在信源解码器中利用解码芯片提供的时间信息，加上外部控制器来完成同步机制。

而对于播放过程中的音视频同步问题，一般有两种解决方法^[114-119]：一种是参考时间同步，把时间作为一个度量维度，每一媒体对象与时间线联系在一起，时间线表示一个抽象的时间。表现对象以位置安排，以时间偏移和时间标记安排在时间线上，协同表现。另一种是参考点同步，依赖时间的媒体看成以周期性的时间间隔表现的离散子单元序列。每个子单元分隔点称作一个参考点。对象间的同步根据这些等间隔的参考点来实现。

基于参考时间线的同步方法是把多个具有时间关系的媒体对象依附于一个时间线上进行描述。移去或更换一个对象不影响其它对象的同步。同步技术的关键是维持一个公共的时间线，每个对象将时间线映射到对象的局部时间，并沿此局部时间线表现。当公用

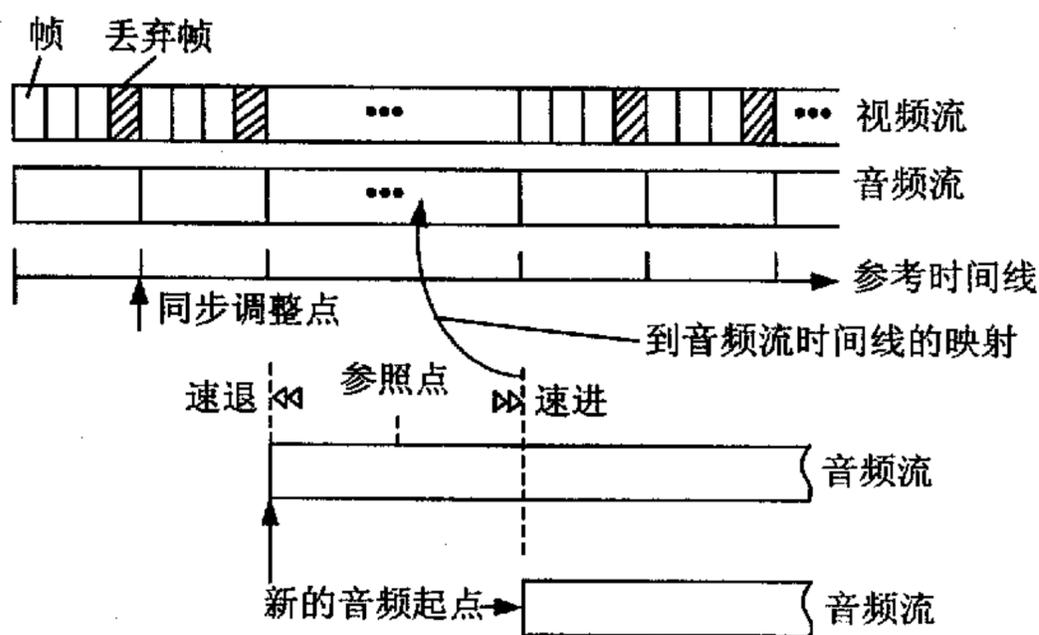


图 4.17 参考时间线的同步方法

时间线与局部时间的误差超出一个指定范围时，则要求重新同步。

利用参考时间线进行同步时，音频和视频数据是分开存放的。视频数据由帧图像序列组成，它构成一个视频图像库。音频数据是一个数字化的声音文件，该文件包括文件头和含有控制播放信息的和数字化采样数值的音频控制块。音频对象随时间的延伸映射到时间线形成轨迹，叫音轨。同样，视频对象随时间的延伸，映射到时间线上为视频轨迹。音频轨迹在视频图库的存储形式是一个带有控制信息的数据结构，这个数据结构中包含音频媒体的定位指针、音频段的开始时间（相对于视频段的位置）等参数。

如图 4.17 所示是采用参考时间线实现影视节目播放的同步控制方法。除了采用丢帧和帧延时方法控制不同环境下媒体的同步外，一项重要的技术是采用声音数据的重定位播放，重定位以参考时间线为标准，以实现视频流的暂停、速进、速退控制。同步控制算法如下：①测试计算机速度和磁盘 IO 速度以确定丢帧数目和帧延迟时间；②非正常顺序播放，以参考时间线的映射为基础，对视频流和音频流重新定位，定位新的视频数据起点，并从时间线的映射关系中找到音频流的帧位置。③视频数据前台播放，音频数据后台播放，两种媒体并行表现；④在同步点进行帧调整，同步调整点由下面计算式得到：当 $\Delta > 0$ 时音频速度快于视频速度，需要视频每隔 $VideoSampleRate / (VideoRate - VideoSampleRate)$ 帧跳过一帧，以达到同步。当 $\Delta < 0$ 时，视频速度快于音频速度，视频每帧需要延迟 $1 / (VideoRate - VideoSampleRate)$ 秒，以达到同步。

参考时间线同步的突出优点是媒体以非切割方式分开存储，不同段可重复用于不同的应用程序，而不必重复地切割成多个拷贝，这在多文种多语种对同一视频图像进行解说的应用中是非常合适的，在这种情况下，语种文种可以改变而不必重复构造切割的媒体。虽然基于时间线的同步模型能够较好地表达媒体对象内部结构的抽象，例如在一个视频流中定义一个与场景有关的主体的表现起始位置，同步就不再要求有关视频帧的参数。但是，这种模型对表现各种数据流之间的同步关系还不充分，例如当用户交互作用于表现对象时，媒体间的同步保持较为困难。

参考时间线不同，基于参考点的同步没有明确的时间参考，描述对象之间的时间关系。视频和音频这样的动态对象被看作是由等间隔的子单元序列（视频帧、音频采样流）组成的，同步的关系以同时出现的不同对象的子单元来定义。另外，附加媒体表现的延时来控制媒体的表现速度。在参考点同步控制方法中，一个对象中子单元的位置称作参考点。一个参考点可以是一个绝对自然数，例如，视频帧的帧号，或是相对参考点，例如 0 到 1 之间的小数，表现之前，相对参考点需要映射到最相近的绝对

参考点。

基于参考点的同步把多媒体对象分为静态和动态的基本对象，基本对象由表现属性和信息对象（如图像、文本、声音、视频段等）的索引构成。基本对象的同步由参考点定义，一个参考点与相应基本对象的共同描述称为同步元素，二个或多个元素可以结合成一个同步点，完整的多媒体对象之间的同步由这些同步点定义。

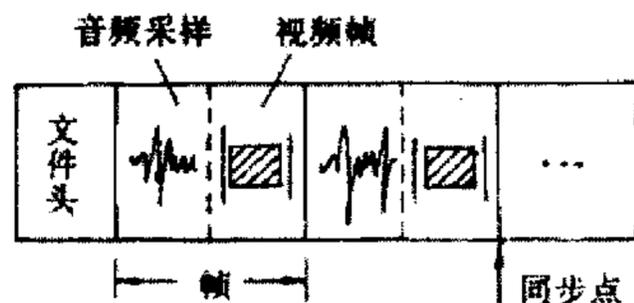


图 4.18 音频视频交错存储的文件格式

如图 4.18 所示，视频和音频数据交错存放在一个影像文件中，一个影像帧由音频段和视频图像组成，声音和图像保持严格的同步关系。在这种结构中，参考点与音频或视频构成同步元素，一帧构成同步点。帧与帧之间的分隔点是音频和视频媒体的参考点。

基于参考点的同步具有一些明显的优点：（1）对内存要求降低，不需要一次性将所有音频数据调入内存，而是利用双缓冲技术实现音频数据的连续播放；（2）播放过程中非常方便地进行速进、速退、暂停和慢放（表现对象延时）等控制操作。

新起点的定位变得十分简单，同步自然保持。也就是说，对无确定表现时间长度的对象，容易进行同步管理，对动态对象的松紧关联同步灵活进行控制选择。但是，这种同步结构不能方便地替换同步对象。要替换同步对象，必须重构复合的音频视频序列。

尽管如此，这种基于参考点同步的影像结构是一种较好的用于计算机表现音频视频对象的文件存储结构。

4.5.2 HQDB 的视频音频同步算法

判断播放效果好坏的一个重要指标是视频与音频的同步。当动态地再现图像与声音序列时，必须保持同步，主要表现为节目的配音必须与讲话者的口型符合。由于本系统是电视台的数字化播放控制系统，所播放的视频文件已是成品不需编辑，而且音视频分开存放不利于节目素材的管理和节目单编辑，因此系统采用基于参考点同步的方式进行音视频的同步。

目前，许多多媒体的数据文件格式(如 MOV 与 AVI 格式)采取了复合视频与音频数据的方式。AVI 是一种视频音频交错存放的数字视频文件格式。这种按交错方式组织视频和音频数据的方式，可以在读取视频音频数据流时，更有效地从存储媒介得到连续的信息。AVI 文件音频部分是采用 WAV 音频格式。在 AVI 文件的视频部分可以采用多种数据压缩方式，这使得 AVI 文件在存放不同格式压缩的视频数据时也比较

方便。传统的 AVI 文件对文件大小有限制，这在存放无压缩或较小压缩比的视频文件时存在问题。针对广播级的视频，对 AVI 文件提出了一种改进格式：ODML AVI 格式。ODML AVI 格式基本上兼容 AVI 格式，它通过在文件后部增加文件块的方式扩展了文件大小，突破了老的 AVI 文件 4GB 大小的限制。另外，ODML AVI 除了包含老的 AVI 文件中的视频、音频流，还增加了一个时间流，用来支持广播级视频中的时间戳。正是由于 AVI 文件的普遍性与适用性，我们采用 AVI (Audio Video Interleave) 文件格式作为视频文件存放格式，如图 4.19 所示。在这种视频音频复合序列文件中，视频音频数据分别被分割成固定时间间隔播放的离散序列，而同时播放的音频和视频帧复合在一起作为子单元。这种情况下，每个子单元分隔点可作为一个参考点，也就是用于控制同步的同步点。

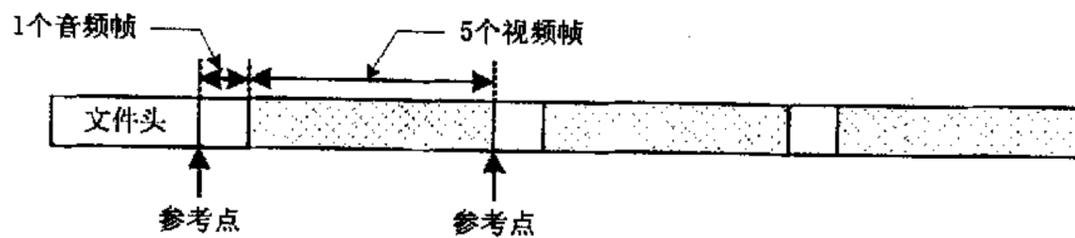


图 4.19 视频音频复合序列文件结构图

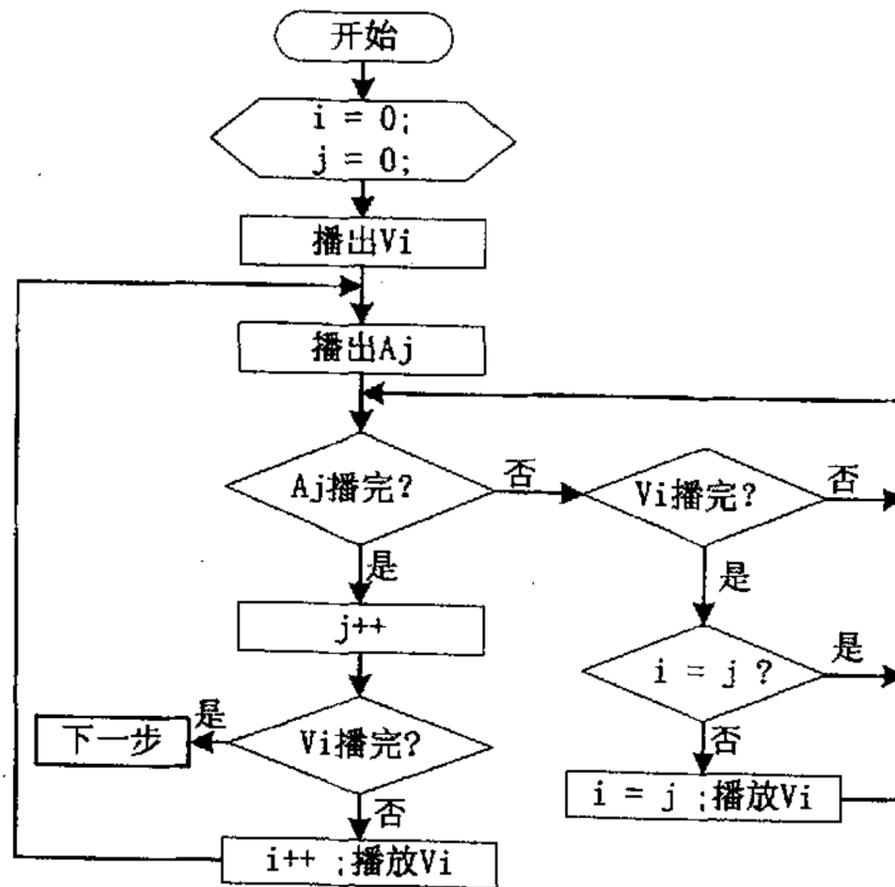


图 4.20 视频音频复合序列同步流程图

同步控制的关键是：音频、视频播放同步于参考点。同步控制算法流程如图 4.20 所示，同步算法的步骤如下：

- (1) 进行初始化，也就是打开相应文件、根据文件头及相关内容设置环境并把

读写指针指向首帧。

(2) 依次播出各帧：在播放每帧时，如果音频帧播放完成，就播出下一音频帧，否则，判断当前的视频帧是否播放完成。

(3) 如果没完成，则继续视频帧与音频帧的播放。

(4) 如果当前视频帧播放完成，判断该视频帧是否当前音频帧的同步帧。

(5) 如果不是，继续播出当前音频帧及与当前音频帧同步的视频(视频显示速度较慢时，该同步视频帧就可能不是刚播放完视频帧的相邻帧，也就是说出现丢帧)；否则继续播放当前音频帧(视频帧播出快于音频帧播出，直到当前音频帧播放完成下一音频帧开始播出，下一视频帧才得以表现，也就是说视频帧播放出现延迟)。

从上述流程不难看出，视频数据播放快于音频数据时，视频播放就要等待，而视频数据播放慢于音频数据时，视频播放就要采取丢帧的方法以便能够同步于音频数据的播放。

这种媒体信息存储法对非正常播放(暂停、快进、快退等)较为方便，各媒体播放帧较易于定位，例如快进到第 m 帧，只需把读取指针简单地定位到第 m 帧信息所在位置即可。

小 结

电视直播由于其能及时、准确、真实、迅速地将现场传播给观众，极大地提高了电视的现实性及观众的收视率而被越来越多的电视台所要求。目前通用的电视直播系统都是由大量的专用设备实现的。这样的直播系统，虽然功能较全，具有热线电话等，但硬件繁多而且操作复杂，存在人员协调同步问题、硬件故障问题、价格昂贵的问题及音视频同步的问题。本章从提高性价比出发提出了一种以视频卡为基础的主要由软件实现的直播技术，与通用的直播系统相比，它的实现更简单且更可靠。由于它的音视频同时实现，因此不存在音视频同步的问题，它的实时性、同步性更好。只需一人操作，不存在人员协调同步问题，在直播的同时也将现场的实况存到本地磁盘存储器上，因此性价比要高。此直播技术已经过一年的实际运行效果良好，对同类系统具有很好的借鉴作用。

由于电视台需要对某些直播节目进行实时监控，在转播直播节目时不是立即播出，而是在对节目进行审阅后，延迟一段时间再播出，以满足电视台的特殊要求，因此有必要研究延迟播放技术。目前的延迟播放系统大多由大规模集成电路实现，价格昂贵，国内也较罕见，针对这种情况，本章在分析比较了四种不同处理方案的性能特点的基础上，创造性地提出了利用磁盘缓存共享压缩视频文件来实现延迟播放的最优算法。

此算法可以有效地实现延迟播放功能，不存在延迟时间限制的瓶颈，用软件实现，所需代价最小。此算法也已经过一年的实际运行，达到预期效果，为延迟播放技术提供了新方法。

由于播放系统在播放节目文件时要求能播放的文件格式越多越好，因此播放节目时存在实时格式转换的需求。以往的格式转换要么通过底层硬件电路，要么通过工具软件来实现，不能满足系统的需求。因此本章提出了一种多线程流水调度算法，根据 DVD 的 VOB 文件特点，对文件格式转换的各环节---解复用、编码、复用，进行优化，较好地实现了 DVD 文件与 AVI 文件以及 VCD 文件与 AVI 文件混合的无掉帧实时播放。实验结果表明：针对 VOB 文件特点的解复用器比通用解复用器提取音视频流的耗时要小得多；采用编码优化算法使得编码时间远小于播放时间及未优化算法的编码时间；且 DAT 文件及 VOB 文件到 MPEG 文件的格式转换及混合播放，可达到帧精确。

5 数字视频播放系统的响应时间分析

5.1 分析数字视频播放系统响应时间的必要性

响应时间是指从提出任务请求服务开始到服务结束为止的时间。任务服务请求的响应时间(或平均响应时间)是用户评价系统的重要指标,其值越小,系统实时性越好。

在节目的播放过程中存在临时更改后续节目的情况,例如:由于上级部门发布了紧急命令,要临时改变节目单;或直播过程中因现场的变化(如球赛的加时赛)要改变节目的长度和下一节目的内容等。各种原因,甚至是错误的原因,节目单可能会随时改变,播放系统需要及时响应节目单的各种变化情况,并以尽量合理有效的方式予以处理。使系统在可容忍的时间内响应用户的要求。因此对于数字视频播放系统,我们要分析的响应时间是指用户提交当前时间即将开始的节目单到节目开始播出的时间。

修改节目单有以下几种情况:(1)即时插入一个节目,要求马上响应新插入的节目,原来的节目在新节目播完后继续播出。(2)加长或截短在播节目的播放时间,要求后续节目相应变化。(3)插入、删除、修改在播节目的下一节目,等。(4)插入、删除、修改在播节目的不相邻的下一节目。除第(4)外,其余三种对时间响应的要求都很严峻,用户希望播放系统的响应时间越小越好。但更新节目单后到底允许等待的最大响应时间是多少,应该给出一个数值,以便用户在此时间之前修改好节目单,使系统能正确响应。此外,有一个确定的响应时间,便于用户检查系统故障是否因节目处理不当引起。同时,对系统响应时间的分析有利于设计人员对系统性能进行优化设计,找出相应策略真正做到节目单的实时修改,使系统能实时响应。

5.2 播放请求服务的 IO 路径分析

播放系统的网络结构如图 5.1 所示,从图 5.1 可看出,播放系统的 IO 路径如图 5.2 所示:用户在编目工作站上修改节目单,并通过光纤网发送修改后的节目单到网管工作站,网管工作站再通过光纤网将修改的节目单发送到存储播放服务器,存储播放服务器收到此命令后,以多线程完成此任务。其存储播放服务器多线程中,线程一(预读线程),负责监视当前节目的播放进度,并在当前节目播放结束前预打开下一个节目文件,提取下一节目文件的信息,向线程二(读磁盘线程),传送下一个节目文件的文件句柄、播放起始位置和结束位置。

5 数字视频播放系统的响应时间分析

5.1 分析数字视频播放系统响应时间的必要性

响应时间是指从提出任务请求服务开始到服务结束为止的时间。任务服务请求的响应时间(或平均响应时间)是用户评价系统的重要指标,其值越小,系统实时性越好。

在节目的播放过程中存在临时更改后续节目的情况,例如:由于上级部门发布了紧急命令,要临时改变节目单;或直播过程中因现场的变化(如球赛的加时赛)要改变节目的长度和下一节目的内容等。各种原因,甚至是错误的原因,节目单可能会随时改变,播放系统需要及时响应节目单的各种变化情况,并以尽量合理有效的方式予以处理。使系统在可容忍的时间内响应用户的要求。因此对于数字视频播放系统,我们要分析的响应时间是指用户提交当前时间即将开始的节目单到节目开始播出的时间。

修改节目单有以下几种情况:(1)即时插入一个节目,要求马上响应新插入的节目,原来的节目在新节目播完后继续播出。(2)加长或截短在播节目的播放时间,要求后续节目相应变化。(3)插入、删除、修改在播节目的下一节目,等。(4)插入、删除、修改在播节目的不相邻的下一节目。除第(4)外,其余三种对时间响应的要求都很严峻,用户希望播放系统的响应时间越小越好。但更新节目单后到底允许等待的最大响应时间是多少,应该给出一个数值,以便用户在此时间之前修改好节目单,使系统能正确响应。此外,有一个确定的响应时间,便于用户检查系统故障是否因节目处理不当引起。同时,对系统响应时间的分析有利于设计人员对系统性能进行优化设计,找出相应策略真正做到节目单的实时修改,使系统能实时响应。

5.2 播放请求服务的 IO 路径分析

播放系统的网络结构如图 5.1 所示,从图 5.1 可看出,播放系统的 IO 路径如图 5.2 所示:用户在编目工作站上修改节目单,并通过光纤网发送修改后的节目单到网管工作站,网管工作站再通过光纤网将修改的节目单发送到存储播放服务器,存储播放服务器收到此命令后,以多线程完成此任务。其存储播放服务器多线程中,线程一(预读线程),负责监视当前节目的播放进度,并在当前节目播放结束前预打开下一个节目文件,提取下一节目文件的信息,向线程二(读磁盘线程),传送下一个节目文件的文件句柄、播放起始位置和结束位置。

线程二则负责根据当前节目文件句柄和节目的播放位置，在节目播放之前，从磁盘预读视频数据到压缩数据缓冲区中的空缓冲块中。

线程三（解码线程），则负责控制 CODEC 预解码压缩数据到播放数据缓冲区的空缓冲块中。

线程四（播放线程），负责根据节目单将播放缓冲区中的视频数据发送到视频服务器的播出通道。

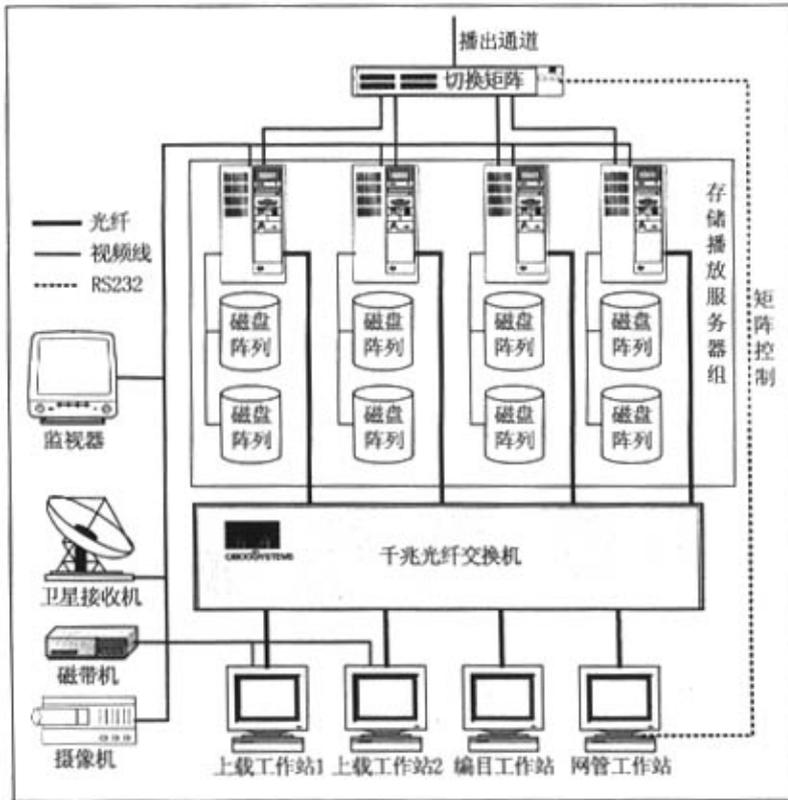


图 5.1 播放系统的网络结构

线程三与具体的播放节目无关，只负责将压缩数据缓冲区中的压缩数据解码到播放数据缓冲区中的空缓冲块中。这样，当前一个节目还没有完全播完时，后一个播放节目的视频数据就已经解码到播放数据缓冲区等待播出，从而做到前一个播放节目的视频数据与后一个播放节目的视频数据紧密衔接，线程四只需要根据节目的开始位置、结束位置、播出时刻以及已播出的帧数决定是否将当前播放数据缓冲区的视频数据送到播放输出通道。这样，四个线程协调工作就可以真正实现不同节目之间播放的

连续性,从而达到播放连续无黑场的要求。只有系统每天第一次开始播放节目和节目更改时,才必须经过这四个线程的响应时间,其余情况下,这四个线程都是并行运行,线程一、线程二、线程三的响应时间均隐藏在线程四的响应时间即播放时间的间隙里。

5.3 影响播放服务系统响应时间的因素

从播放服务请求的 IO 路径分析可知,播放系统中修改节目单的播放服务响应时间分为:阶段一,节目单从编目工作站到网管工作站千兆网卡的传输时间,设此时间为 T_{B-W} ;阶段二,网管工作站将节目单通过光纤网传给存储播放服务器千兆网卡,设此时间为 T_{W-F} ;阶段三,存储播放服务器接收到修改节目,并做出相应响应的的时间,设为 T_{FR} 。

阶段一 T_{B-W} ,阶段二 T_{W-F} ,这两阶段都与千兆网的传输速度及节目单的大小有关。阶段三 T_{FR} ,与千兆网的传输速度、修改与创建链表时间、磁盘阵列响应时间、节目数据解码时间及音视频数据播放时间有关。

1) 千兆网的传输速度,千兆以太网是对 IEEE 802.3 以太网标准的扩展,在基于以太网协议的基础之上,将快速以太网的传输速率(100Mbps)提高了 10 倍,达到了 1 Gbps。千兆以太网网卡分两类,一种是铜线千兆网卡,一种是光纤千兆网卡。在实际运行环境中,使用这两种千兆网卡的网络传输速度都尚未达到 1000Mbps 的线速,这是因为:网络速度与所设置的进程多少和 TCP 窗口大小有关,除此之外,被测 PC 服务器的 I/O 总线、设备如硬盘等对网络速度影响也非常大^[120-125]。

对 UGS-1000T 铜线千兆网卡使用端对端测试,由于该网卡(传输数据无需占用 CPU 时间,可以不通过 CPU 直接与内存进行数据交换,采用 RJ45 双绞线接入实现方便、快捷、简单、可靠的连接)支持 32 位和 64 位两种 PCI 槽,所以我们将该网卡分别安装在这两种 PCI 槽上进行了测试,操作系统使用 Windows 2000,因为 Windows 2000 增强了对千兆以太网性能的改进,交换机是 UCOM 8 端口 1000Base-T UGS-1008T。

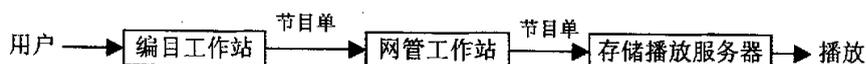


图 5.2 播放系统的 IO 路径

在测试中我们选用了两种工具,以便从不同角度反映网卡速度。第一种工具^[126]是 Chariot 4.0,该软件通过模拟应用层数据包测试网络设备、应用程序等的性能,我们利用 Chariot 的 Throughput.scr 脚本,将客户端 Chariot Endpoint 安装在两台被测服务器之上进行数据传输,选择另一台机器用作控制台。第二种工具是微软 Windows

连续性,从而达到播放连续无黑场的要求。只有系统每天第一次开始播放节目和节目更改时,才必须经过这四个线程的响应时间,其余情况下,这四个线程都是并行运行,线程一、线程二、线程三的响应时间均隐藏在线程四的响应时间即播放时间的间隙里。

5.3 影响播放服务系统响应时间的因素

从播放服务请求的 IO 路径分析可知,播放系统中修改节目单的播放服务响应时间分为:阶段一,节目单从编目工作站到网管工作站千兆网卡的传输时间,设此时间为 T_{B-W} ;阶段二,网管工作站将节目单通过光纤网传给存储播放服务器千兆网卡,设此时间为 T_{W-F} ;阶段三,存储播放服务器接收到修改节目,并做出相应响应的的时间,设为 T_{FR} 。

阶段一 T_{B-W} ,阶段二 T_{W-F} ,这两阶段都与千兆网的传输速度及节目单的大小有关。阶段三 T_{FR} ,与千兆网的传输速度、修改与创建链表时间、磁盘阵列响应时间、节目数据解码时间及音视频数据播放时间有关。

1) 千兆网的传输速度,千兆以太网是对 IEEE 802.3 以太网标准的扩展,在基于以太网协议的基础之上,将快速以太网的传输速率(100Mbps)提高了 10 倍,达到了 1 Gbps。千兆以太网网卡分两类,一种是铜线千兆网卡,一种是光纤千兆网卡。在实际运行环境中,使用这两种千兆网卡的网络传输速度都尚未达到 1000Mbps 的线速,这是因为:网络速度与所设置的进程多少和 TCP 窗口大小有关,除此之外,被测 PC 服务器的 I/O 总线、设备如硬盘等对网络速度影响也非常大^[120-125]。

对 UGC-1000T 铜线千兆网卡使用端对端测试,由于该网卡(传输数据无需占用 CPU 时间,可以不通过 CPU 直接与内存进行数据交换,采用 RJ45 双绞线接入实现方便、快捷、简单、可靠的连接)支持 32 位和 64 位两种 PCI 槽,所以我们将该网卡分别安装在这两种 PCI 槽上进行了测试,操作系统使用 Windows 2000,因为 Windows 2000 增强了对千兆以太网性能的改进,交换机是 UCOM 8 端口 1000Base-T UGS-1008T。

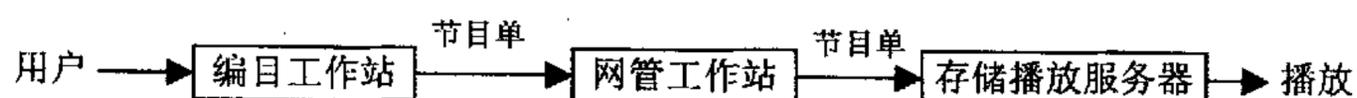


图 5.2 播放系统的 IO 路径

在测试中我们选用了两种工具,以便从不同角度反映网卡速度。第一种工具^[126]是 Chariot 4.0,该软件通过模拟应用层数据包测试网络设备、应用程序等的性能,我们利用 Chariot 的 Throughput.scr 脚本,将客户端 Chariot Endpoint 安装在两台被测服务器之上进行数据传输,选择另一台机器用作控制台。第二种工具是微软 Windows

2000DDK 的 NTTCP 工具, 该工具采用内存到内存测试方法, 能够测试不同网络拓扑结构和硬件上的网络吞吐量, 对网络上数据传输提供多线程、异步性能测试基准, 在发送端和接收端分别运行相应的软件进行测试。

在使用 32 位/33MHz PCI 槽时, 我们选择了配置 PIII866MHz/256M RAM/18G HDD 的 HP 服务器作为被测端。使用 Chariot 进行测试时, 速度可达到 521Mbps, 使用 NTTCP 进行测试时, 发送端速度是 588Mbps, 接收端速度是 600Mbps。

而国外测试机构对千兆网卡的测试中, 往往是用 Jumbo 帧(9018 字节, 目前是千兆以太网领域非常受关注的帧长度, 由 Alteon WebSystems 公司提出的对 802.3 以太网的扩展帧长度)测试时, 网络速度才可能接近线速, 并可能获得比 1518 字节帧环境下高 50% 的速度, 主要是因为 Jumbo 帧比较大, 而千兆以太网比快速以太网拥有大得多的带宽, 帧越大就越有效, 因为这样将减少包处理次数和中断次数, 从而加快了传输速度。如果使用的是 1518 字节帧的话, 速度基本也在 600Mbps 左右。

在测试中我们还发现网络速度与所设置的进程多少和 TCP 窗口大小有关, 除此之外, 被测 PC 服务器的 I/O 总线、设备如硬盘等对网络速度的影响也非常大。我们发现在 32 位/33MHz 环境下测试速度最高可达到 600Mbps。

由于硬盘读写非常花费时间, 曾有报告显示通过硬盘测试千兆网卡的速度只有 130 兆左右, 所以我们在测试中使用的是内存的数据传输方式, 避开了硬盘读写所造成的网络性能下降。可是实际应用中通过网络进行的数据传输要通过硬盘读写, 所以将来要提高网络性能, 还需要加快硬盘速度, 并且提供更大的用于优化 I/O 的磁盘缓存。

我们还发现, 将 TCP 校验和计算从计算机卸载至网卡也使得测试速度有所提高, 以往的 Windows 机器将每个网络上要传输的数据包交给操作系统处理, 也就意味着机器的 CPU、总线和内存都要进行包处理, 这阻碍了网络性能的提高。微软的 Windows 2000 将数据包的处理重新交给网卡, 而且 Windows 2000 改善了处理 IP 堆栈的方法, 大分 TCP/IP 实现可自我调节, 允许网卡执行 TCP 校验和计算, 使服务器有更多的 CPU 资源可用于执行其他操作, 从而增强了网络性能^[127-128]。

总之, 虽然由于各种外在因素, 我们在测试中使用铜线千兆网卡的网络速度还远未达线速, 但最高测试速度达到了 600Mbps。

光纤千兆以太网网卡可以保证更高的数据安全性, 用作骨干连接, 并可以支持更长的连接距离和更好的噪音过滤。相较而言, 铜线千兆以太网网卡用作从快速以太网向千兆以太网升级更简便, 成本更低, 并用作工作组级连接。光缆具有高度的抗电磁干扰、数据包侦测/嗅探及信号衰减能力, 这使之非常适合较长的缆线距离、高安全性局域网及存在电子“噪音”的环境。而且光缆支持无限的吞吐率(铜缆的

限制为 1000 Mbps)。为获得扩展业务的空间,并可在采用 10 GB 以太网技术时获得足够的带宽,本播放系统中采用了 10 块 3COM 3C985B-SX 千兆网卡,此款服务器网卡是 3COM 公司专为千兆以太网联网环境而设计的基于标准的高性能服务器网卡。具有 DynamicAccess 服务技术,在性能上胜过了其他竞争性产品平均 30%,同时较其前一代产品性能提升了 44%。此款服务器网卡的性能较其前一代产品性能的提升包括:

- 支持带有 64 位/66MHz PCI 插槽的新型服务器,能够利用 PCI 接口实现更高的性能并更为有效地进行数据传输。
- 具备多种故障恢复速度的弹性服务器链路,可以通过更低价的解决方案增强服务器的可用性。
- 可以与任何类型的快速以太网或千兆位以太网交换机相配合的双向负载均衡技术,使用户可以在多厂商产品的交换环境中更加方便地拓展服务器连接的带宽并杜绝服务器拥塞现象的发生。
- 支持 Windows NT、Netware 和 Unix 操作系统下的多种驱动程序,增强了与大部分常用的服务器环境之间的兼容性。备有 SCO Unixware 7 驱动。

基于 LANQuest 实验室独立的评测基准程序,通过在典型的网络环境中进行的测试,3Com 公司的千兆位 EtherLink 服务器网卡无论是在双向流量传递还是下行数据流量的传输方面都比较好。如 3C985B-SX 的性能测试。测试环境为:

- (1) 服务器: Compaq Proliant7000、Quad PII Xeon 450, 1GB RAM, WIN 2000 Server
- (2) 服务器端 NIC: 3C985B-SX
- (3) 客户机: Dell Precision 410, Pentium II 450×2, 256MB, WIN 2000 Workstation
- (4) 客户端 NIC: 3C985B-SX
- (5) 测试拓扑结构: 1 段, 6 客户
- (6) 基准: Ganymede's Chariot
- (7) Chario 配置: 6 客户机 (3 发送、3 接收)
- (8) 协议: TCP/IP
- (9) 交换机: 3Com SuperStack™ II Switch 9000

测试结果: 3C985B-SX 的客户机到服务器的传输测试的吞吐量达到了 942.204Mbps, 同时有最低的 CPU 占用率 52.917%。而在服务器到客户机的传输中吞吐量也达 940.655Mbps, CPU 占用率为 46.333%。

由以上论述可知铜线千兆网卡的传输速度的最大线速为 600 Mbps, 最小值 V_{q-min} 为 133 Mbps 左右, 3C985B-SX 光纤千兆网卡的传输速度 V_{q-max} 为 942 Mbps 左右。在

本系统中 HP 服务器、HP 工作站与测试环境中的服务器、客户机的性能配置相当，因此 3C985B-SX 光纤千兆网卡的传输速度可以拟用此数据。

2) 节目单大小，节目单是传输在编目工作站、网管工作站、存储播放服务器之间的数据，其大小也是影响响应时间的因素之一，节目单越大，网络传输时间越长，响应时间也越长。本系统的节目单数据结构定义如下：

```
typedef struct _CRecordset
{
    long    m_ID;           //节目单条号
    CString m_ProgName;     //节目名称(含路径)
    long    m_Millisecond; //毫秒
    long    m_start;       //开始点
    long    m_end;         //结束点
    int     m_loop;        //循环次数
    int     m_pbRaidNum;   //节目来源阵列组数
    CTime   m_time;       //播放时间
    BOOL    m_tag;        //节目是否传送标志
} Crecordset
```

使用 `sizeof(Crecordset)` 函数可知：一条节目所有字段共占 36 字节，那么 n 条节目需要光纤网传输的字节大小为 $B=36*n$ 字节。

3) 磁盘写速度，节目单从编目工作站传到网管工作站网卡内存后，网管工作站要从千兆网卡内存接收节目单到磁盘，假设其接收时间为 $T_{waccept}$ ，同理节目单从网管工作站传到服务器后，服务器要从其自身的千兆网卡内存接收节目单到磁盘，假设其接收时间为 T_{facept} ， $T_{waccept}$ 与 T_{facept} 都与磁盘写速度有关，而磁盘写速度又与磁盘写的方式有关^[103-106]。假如采用写回方式，本系统磁盘参数是 160MB/s，若采用写穿方式，磁盘写速度为 215-315Mb/s；设磁盘写速度为 V_{disk-w} ，则 $T_{waccept} = T_{seek} + T_{rotal} + B/V_{disk-w}$ ， $T_{facept} = T_{seek} + T_{rotal} + B/V_{disk-w}$ 相应地因磁盘写方式的不同而不同。

4) 修改与创建链表，存储播放服务器接收到节目单后，要修改并创建链表，设节目单条数为 n ，则其所耗时间为 $O(n)$ 。

5) 磁盘 IO 响应时间，服务器读取节目单并创建链表后，根据节目单命令，向磁盘阵列发出读请求。从服务器发出读数据命令到获得视频数据的时间叫磁盘 IO 响应时间。它与磁盘数传率 v_{d-r} 、读数据块大小 B_d ，磁头所在位置有关；包含寻道时间 T_{seek} ，旋转等待时间 T_{rotal} ，数据传送时间 T_{tran} ，其中磁盘数传率 v_{d-r} 可由磁盘性能参数查知。读数据块大小 B_d ，是以一帧音视频数据为单位，为 0.92442MB 字节，设磁盘阵列 RAID0

磁盘数为 N ，数据分块大小为 $B_{\text{分块}}$ ，而 T_{seek} ， T_{rotal} 是随机的，与磁头所在位置有关，则读一帧音视频数据，磁盘阵列 RAID0 的 IO 响应时间为

$$T_{c-db} = \left[\frac{B_d}{B_{\text{分块}} * N} \right] (T_{\text{seek}} + T_{\text{rotal}} + B_{\text{分块}}/v_{d-r}) \quad (5.1)$$

6) 解码帧大小、解码速度，节目数据从磁盘阵列读出到主机内存后要送到解码缓冲区解码，这里节目数据从内存到内存的传送时间可忽略。只考虑解码时间，而节目数据解码时间 T_{dec} 与解码帧大小和解码速度有关。解码帧大小为一帧音视频数据大小，解码速度由解码卡性能参数可查出。本系统所用解码卡为 Targa3000，其解码速度为 10-50Mb/s。

7) 播放速度，播放时间与播放速度有关，而播放速度又与节目的制式有关，本系统节目是 PAL 制式，则应满足每秒 25 帧的播放速率，则系统播放一帧的时间为 40ms。

5.4 播放服务系统响应时间的数学模型

从播放系统 IO 路径分析与影响播放系统响应时间的因素可知，播放系统响应节目单修改的总的响应时间为：

$$T_{\text{response}} = T_{B-w} + T_{W-F} + T_{FR} \quad (5.2)$$

T_{W-F} 又由网管工作站从千兆网卡接收节目单接收时间 $T_{waccept}$ 、过滤处理时间 T_{filt} 、传送节目单数据到存储播放服务器千兆网卡的时间 T_{wtrans} 组成。 T_{FR} 包括存储播放服务器从千兆网卡接受节目单的时间 T_{facept} 、修改与创建链表时间 T_{link} 、从磁盘阵列预取节目数据到解码缓冲区的时间 T_{c-db} 、节目数据解码时间 T_{dec} 、节目数据从播放缓冲区到播放卡的时间 T_{b-p} 。那么 (5.2) 式可演变成：

$$T_{\text{response}} = T_{B-w} + T_{waccept} + T_{\text{filt}} + T_{wtrans} + T_{\text{facept}} + T_{\text{link}} + T_{c-db} + T_{\text{dec}} + T_{b-p} \quad (5.3)$$

T_{B-w} 与节目单的字节大小和千兆网的传输速度有关，设节目单大小为 B 字节，千兆网传输速度为 v_q ，则

$$T_{B-w} = B/v_q \quad (5.4)$$

$T_{waccept}$ 是网管工作站从千兆网卡内存接受节目单到网管工作站的时间， $V_{\text{disk-w}}$ 是网管工作站磁盘写速度，则

$$T_{waccept} = T_{\text{seek}} + T_{\text{rotal}} + B/V_{\text{disk-w}} \quad (5.5)$$

T_{filt} 是网管工作站根据节目库过滤掉不合法的节目，因为是由 CPU 处理，时间为微秒级，所以可忽略不计。

T_{wtrans} 是网管工作站从内存传送节目单数据到存储播放服务器网卡内存的时间，

与 T_{B-w} 相同, 则

$$T_{wtrans} = B/v_q \quad (5.6)$$

$T_{faccept}$, 是存储播放服务器从千兆网卡内存接收节目单到本地磁盘的接收时间, 与 $T_{waccept}$ 类似。假设存储播放服务器磁盘写速度与网管工作站磁盘写速度相等, 则

$$T_{faccept} = T_{seek} + T_{rotal} + B/V_{disk-w} \quad (5.7)$$

设 T_{link} 为修改与创建链表时间, 与创建节目单规模大小有关, 假设节目单条数为 n , T_{link} 的时间数量级为 $O(n)$, 设此系数为 a 则

$$T_{link} = a * n \quad (5.8)$$

T_{c-db} 是从磁盘阵列预取节目数据到解码缓冲区的时间, 它与磁盘数传率 v_{d-r} 、读数据块大小 B_d , 数据分块大小 $B_{分块}$, 磁盘数 N , 磁头所在位置有关, 包含寻道时间 T_{seek} , 旋转等待时间 T_{rotal} , 数据传送时间 T_{tran} , 则

$$T_{c-db} = \left[\frac{B_d}{B_{分块} * N} \right] (T_{seek} + T_{rotal} + B_{分块} / v_{d-r}) \quad (5.9)$$

T_{dec} 是节目数据解码时间, 与解码卡解码速度 v_{dec} 、解码数据块大小 B_{dec} 有关,

$$T_{dec} = B_{dec} / v_{dec} \quad (5.10)$$

T_{b-p} 是节目数据从播放缓冲区到播放卡的时间, 为播放一帧的时间 40ms。

那么由(5.3)--(5.9)式, 则(5.2)式变为

$$T_{response} = 2B/v_q + 2B/v_{disk-w} + 2(T_{seek} + T_{rotal}) + a * n + \left[\frac{B_d}{B_{分块} * N} \right] (T_{seek} + T_{rotal} + B_{分块} / v_{d-r}) + B_{dec} / v_{dec} + T_{b-p} \quad (5.11)$$

从(5.11)式可看出, 系统响应时间随着千兆网传输速度、节目单大小、磁盘写速度、磁盘阵列 IO 响应时间、数据分块大小、解码帧大小、解码速度、播放速度的变化而变化。因此有

$$T_{response} = \begin{cases} 2B/v_{q-min} + 2B/V_{disk-w-min} + 2(T_{seek-max} + T_{rotal-max}) + a * n + \left[\frac{B_d}{B_{分块} * N} \right] (T_{seek-max} + T_{rotal-max} + B_{分块} / v_{d-r-min}) + B_{dec} / v_{deomin} + T_{b-p} & \text{最坏情况下} \\ 2B/v_{q-max} + 2B/V_{disk-w-max} + 2(T_{seek-min} + T_{rotal-min}) + a * n + \left[\frac{B_d}{B_{分块} * N} \right] (T_{seek-min} + T_{rotal-min} + B_{分块} / v_{d-r-max}) + B_{dec} / v_{deomax} + T_{b-p} & \text{最好情况下} \\ 2B/v_{q-均} + 2B/V_{disk-w-均} + 2(T_{seek-均} + T_{rotal-均}) + a * n + \left[\frac{B_d}{B_{分块} * N} \right] (T_{seek-均} + T_{rotal-均} + B_{分块} / v_{d-r-均}) + B_{dec} / v_{deo-均} + T_{b-p} & \text{平均情况下} \end{cases} \quad (5.12)$$

式(5.11)中 $B=36*n(\text{Byte})$, 因为创建链表所耗时间是微秒级, 所以可设 $a=2*10^{-6}\text{s}$, 且系统中 $B_d=B_{\text{dec}}=0.92442\text{MB}$, 由磁盘性能参数可知, $T_{\text{seek-max}}=9.65\text{ms}$, $T_{\text{rotal-max}}=5.98\text{ms}$, $T_{\text{seek-min}}=0.75\text{ms}$, $T_{\text{rotal-min}}=0\text{ms}$ 。

将 $v_{q-\text{min}}=282\text{Mbps}$ $V_{\text{disk-w-min}}=215\text{Mb/S}$ $v_{d-r-\text{min}}=19\text{MB/S}$ $v_{\text{dec-min}}=5\text{Mbps}$ 代入式(5.12), 则有:

在最坏情况下:

$$T_{\text{response}} = (6.72 * 10^{-3} * n + \left[\frac{924.42}{(B_{\text{分块}} * N)} \right] (B_{\text{分块}} / 19 + 15.63) + 1550.33)(\text{ms}) \quad (5.13)$$

在最好情况下时, $v_{q-\text{max}}=942\text{Mbps}$ $V_{\text{disk-w-max}}=315\text{Mb/S}$ $T_{\text{seek-min}}=0.75\text{ms}$
 $T_{\text{rotal-min}}=0$ $v_{d-r-\text{max}}=23\text{MB/S}$ $v_{\text{dec-max}}=50\text{Mbps}$, 将上述值代入式(5.12), 则有,

在最好情况下:

$$T_{\text{response}} = (4.44 * 10^{-3} * n + \left[\frac{924.42}{(B_{\text{分块}} * N)} \right] (B_{\text{分块}} / 23 + 0.75) + 189.4)(\text{ms}) \quad (5.14)$$

在一般情况下时, $v_{q-\text{均}}=612\text{Mbps}$ $V_{\text{disk-w-均}}=265\text{Mb/S}$ $T_{\text{seek-均}}=5.2\text{ms}$ $T_{\text{rotal-均}}=2.99\text{ms}$
 $v_{d-r-\text{均}}=21\text{MB/S}$ $v_{\text{dec-均}}=27.5\text{Mbps}$, 将上述值代入式(5.12), 则有,

在一般情况下:

$$T_{\text{response}} = (5.11 * 10^{-3} * n + \left[\frac{924.42}{(B_{\text{分块}} * N)} \right] (B_{\text{分块}} / 21 + 8.19) + 325.3)(\text{ms}) \quad (5.15)$$

从式(5.13)、(5.14)、(5.15)可看出: 当节目单条数 $n < 1000$ 时, 节目单的

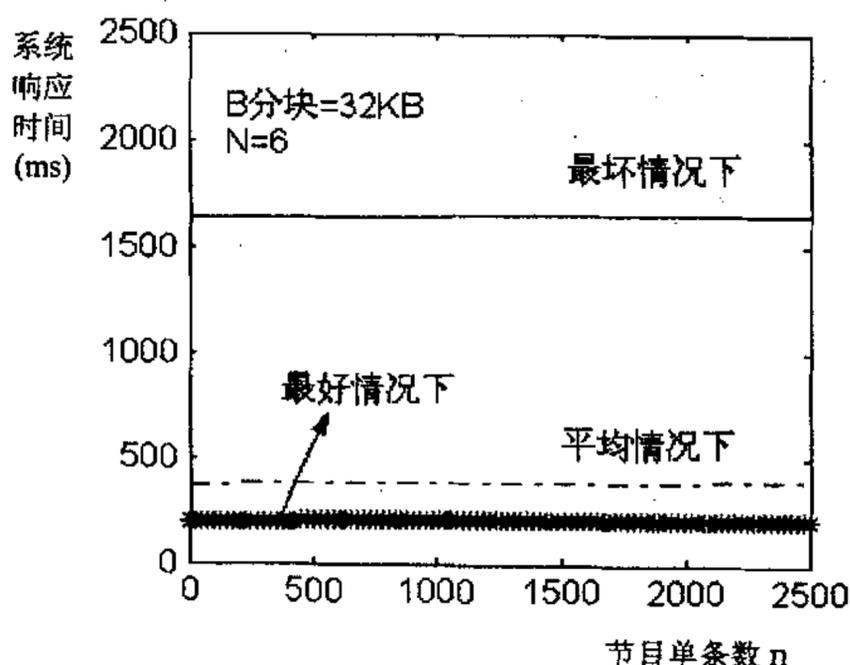


图 5.3 当磁盘阵列的分块大小为 32KB, 磁盘数为 6 时, 节目单长度 n 对系统响应时间的影响

长度对系统响应时间的影响不大，影响程度不到 10 毫秒；当节目单条数 $n > 1000$ 时，对系统响应时间随着节目单的长度 n 的增大而缓慢增大。如图 5.3 至图 5.5 所示，在不同的分块大小和磁盘数时，在最好与平均情况下节目单长度变化对系统响应时间的影响都不大，在最坏情况下，当 $n > 1000$ 时，系统响应时间随着 n 的增大而缓慢增加。

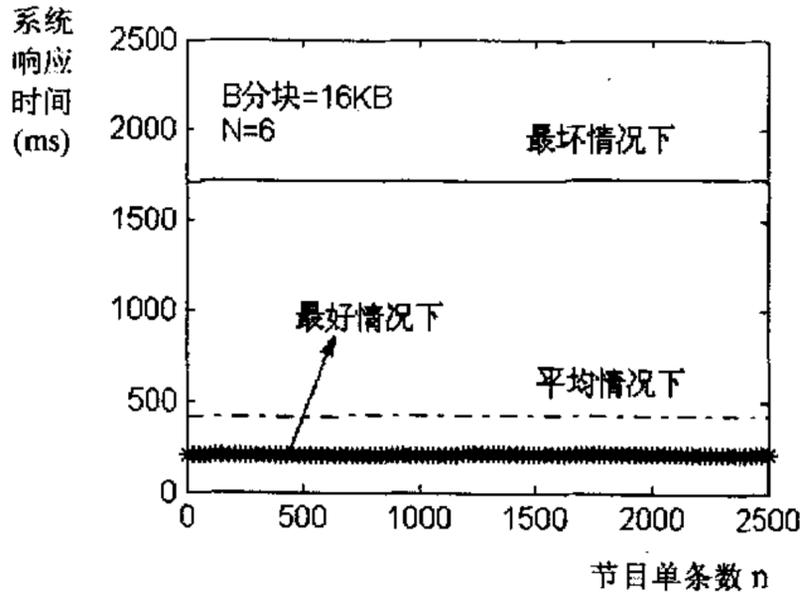


图 5.4 当磁盘阵列的分块大小为 16KB，磁盘数为 6 时，节目单长度 n 对系统响应时间的影响

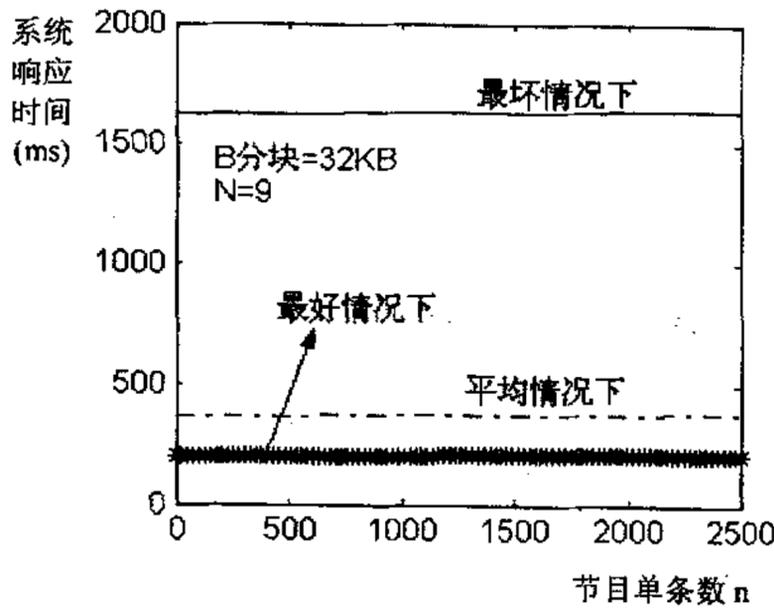


图 5.5 当磁盘阵列的分块大小为 32KB，磁盘数为 9 时，节目单长度 n 对系统响应时间的影响

因 n 的取值对系统响应时间的影响不大，因此取 $n=1000$ （方便计算），考察在不同的磁盘数下，不同的分块大小对系统响应时间的影响。分别取 $N=6, N=9, N=16$ ，如图 5.6 至图 5.8 所示，在这三种取值的最坏与一般情况下，分块大小对系统响应时间的影响基本一致，即当分块大小从小增大到 32KB 时，系统响应时间急剧减少，当分块大小再增加时，系统响应时间趋于平稳。而在最好情况下，分块大小的变化对系统响应时间的影响不大。

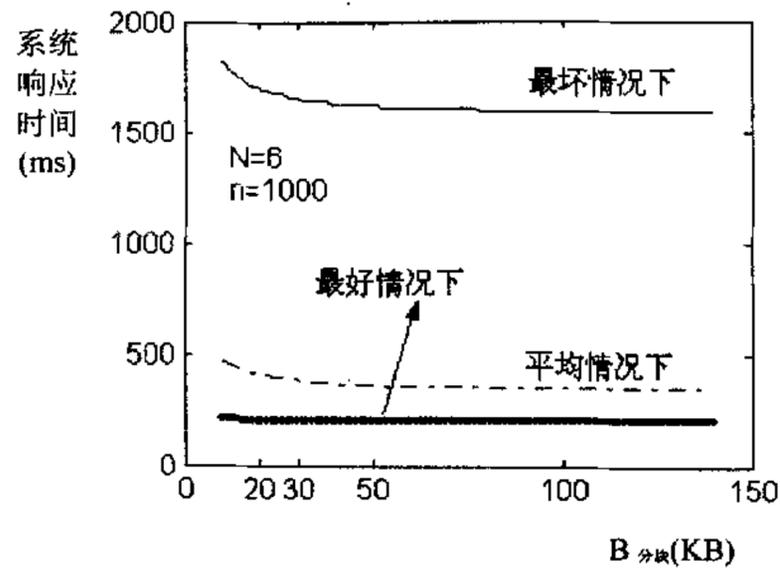


图 5.6 当节目单长度为 1000, 磁盘阵列的磁盘数为 6 时, 分块大小对系统响应时间的影响

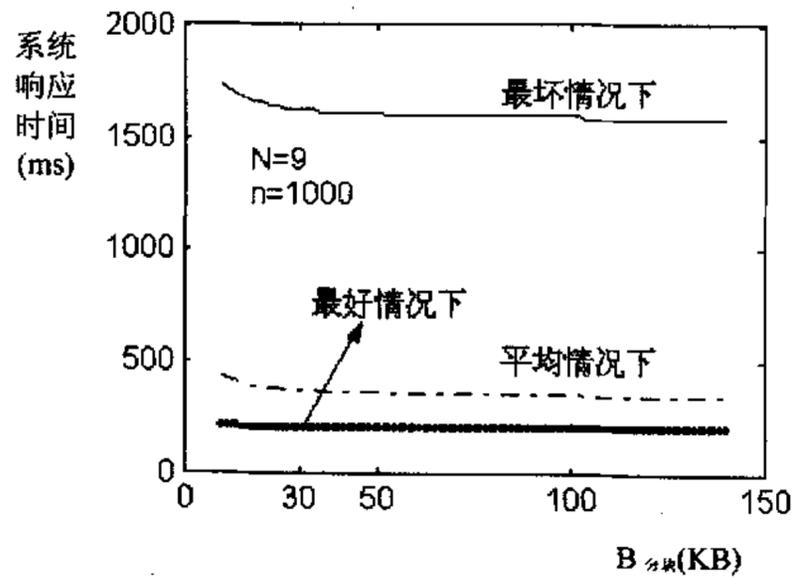


图 5.7 当节目单长度为 1000, 磁盘阵列的磁盘数为 9 时, 分块大小对系统响应时间的影响

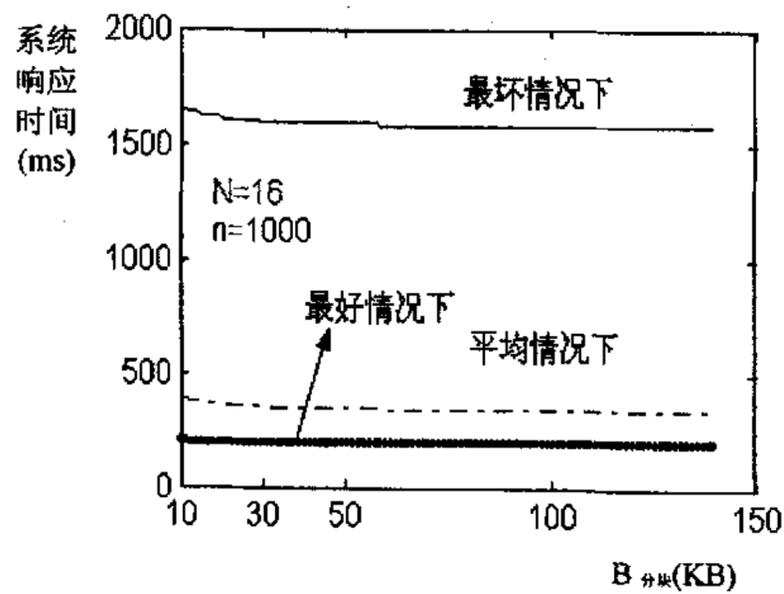


图 5.8 当节目单长度为 1000, 磁盘阵列的磁盘数为 16 时, 分块大小对系统响应时间的影响

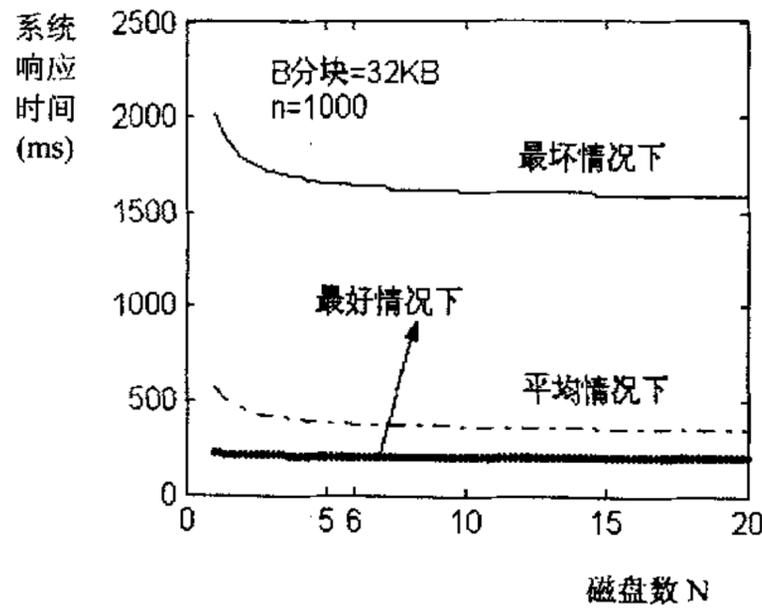


图 5.9 当节目单长度为 1000，磁盘阵列的分块大小为 32KB 时，磁盘阵列的磁盘数对系统响应时间的影响

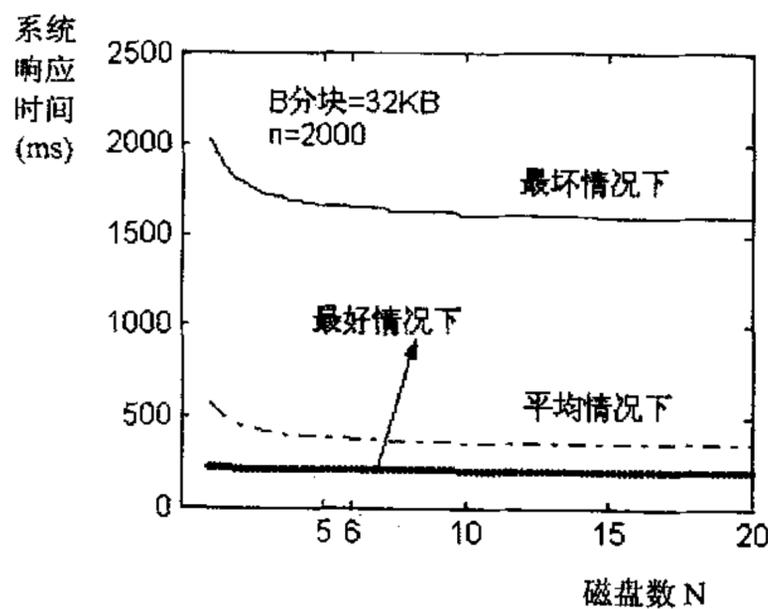


图 5.10 当节目单长度为 2000，磁盘阵列的分块大小为 32KB 时，磁盘阵列的磁盘数对系统响应时间的影响

因此取 $B_{\text{分块}}=32\text{KB}$, $n=1000$ 和 $B_{\text{分块}}=32\text{KB}$, $n=2000$ 来考察磁盘阵列的磁盘数对系统响应时间的影响。从图 5.9、图 5.10 可看出，磁盘阵列的磁盘数 $N=6$ 是系统响应时间的饱和点，当 N 从 2 增大到 6 时，系统响应时间急剧减少， N 从 6 再增加时，对系统响应时间影响不大。综上所述，系统中 $B_{\text{分块}}$ 大小取 32KB，磁盘阵列的磁盘数 N 取 6，可使系统响应时间取得最小值。如图 5.11 所示，当 $B_{\text{分块}}=32\text{KB}$, $N=6$ 时，节目单长度 n 从 1000 到 10000 所对应的系统响应时间。

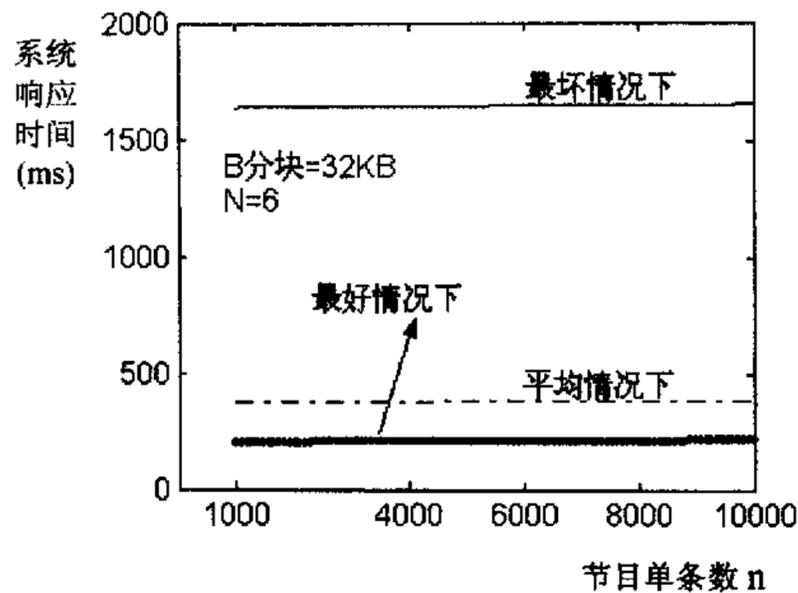


图 5.11 当磁盘阵列的分块大小为 32KB，磁盘数为 6 时，节目单长度 n 对系统响应时间的影响

假如电视台一天播放节目 24 小时， $24 \times 60 \times 60 = 86400s$ ，除广告节目与天气预报节目外，大部分节目时间都会超过 5 分钟，以每节目 5 分钟长度计算，则一天的节目单有 288 条，假设再增加 2 倍数量的广告节目条——这相当于每两分钟节目插一条广告，那么一天的节目单长度有 864 条，没超过 1000 条，我们以 1000 条节目为极限计算。如果用户一次编目 3 天的节目，则不超过 3000 条。若用户编目一星期的节目，则节目单的长度不超过 7000 条。对应图 5.11，我们可看出，当节目单条数为 1000 条、3000 条、7000 条时，在最坏情况下的系统响应时间均不超过 2 秒；在一般情况下，系统响应时间均不超过 500 毫秒；在最好情况下，系统响应时间均不超过 250 毫秒。

5.5 减少系统响应时间的策略

从图 5.3 至图 5.11 及式(5.11)可知，要减少系统响应时间，可采取以下几个方面的措施：

- 1) 将解码器的解码速度设定成较大值。解码器的速度一般是处于一定范围的值，可在系统初始化时，在程序中将解码速度设定成较大值。
- 2) 选取最佳的分块大小值及磁盘阵列磁盘数。由于磁盘阵列的底层分块与磁盘数可由我们控制，因此可根据理论分析的值，选取最佳的分块大小值及磁盘阵列磁盘数。
- 3) 存储视频数据时采取顺序存放，同一节目尽量放在同一磁道，同一扇区或相邻磁道，相邻扇区，以减少磁盘阵列的响应时间，这可在底层存储中实现。

4) 采用磁盘写回方式提高磁盘写速度, 采用光纤千兆网卡提高网络传输速度。使系统响应时间达到最小, 即图 5.11 的最好情况下的系统响应时间值。这样系统每天开始运行的初始等待时间或修改节目单的提前时间可达最小。

小 结

本章通过对数字视频播放系统的 I/O 路径上影响播放系统服务响应时间的因素, 如千兆网传输速度、节目单大小、磁盘写速度、磁盘阵列 I/O 响应时间、数据分块大小、解码帧大小、解码速度、播放速度等的分析, 建立了数字视频播放系统服务响应时间的数学模型。指出当节目单条数 $n < 1000$ 时, 节目单的长度对系统服务响应时间的影响不大, 影响程度不到 10 毫秒; 当节目单条数 $n > 1000$ 时, 系统服务响应时间随着节目单的长度 n 的增大而缓慢增大。在一定节目长度下, 当分块大小从小增大到 32KB 时, 系统响应时间急剧减少, 当分块大小再增加时, 系统响应时间趋于平稳。在 $B_{\text{分块}}=32\text{KB}$, $n=1000$ 和 $B_{\text{分块}}=32\text{KB}$, $n=2000$ 的情况下, 磁盘阵列的磁盘数 $N=6$ 是系统响应时间的饱和点, 当 N 从 2 增大到 6 时, 系统响应时间急剧减少, N 从 6 再增加时, 对系统响应时间影响不大。最后针对影响系统服务响应时间的因素, 提出了四条相应的改进措施, 以便使系统服务响应时间达到理想中的最小。

全文总结

数字电视技术具有强大的优越性。在网络技术、计算机技术、存储技术、影视高新技术的推动下,近年来电视台数字化的进程明显加快,数字化正向电视系统的多个环节,包括制作、传输、发射和接收,全方位展开。

播出系统作为电视台的最后一道关口,直接关系到节目播出的质量。它是数字电视台建设的核心,对电视台整个数字化体系的建立产生长期决定性的影响。长期以来,播出系统一直是电视系统数字化改造的瓶颈,因而成为当前广播电视领域研究的热点之一。为了开发有自主知识产权和核心技术的数字播放系统,我们与企业合作,从事电视台数字播放系统的研究和开发。

针对已有硬盘播放系统的价格昂贵、编辑网与播控网分离、帧精确与延迟播放等功能依赖硬件实现等缺点,我们提出了一种中大型电视台的全数字视频播放系统的技术方案,该方案可使无压缩播放和现行标准的压缩播放运行于同一硬件平台,可使编辑网与播放网相融合,同时亦具有从标准清晰度电视到高清晰度电视播出的扩展能力,其中无压缩播放属国内首创,该方案已被实施并取得了很好的效果。本文就是在对视频卡底层驱动程序进行二次开发的实践基础上,围绕着音视频播放这个核心,对自己所从事的研发从播放系统的体系结构、无压缩播放与压缩播放等方面进行的理论概括。所做的主要工作和取得的主要结论如下。

(1) 体系结构是电视台数字视频播放系统最重要的内容之一,而存储播放子系统是整个系统的核心,它的体系结构设计至关重要。为满足高质量存储播放子系统的高可靠性、高数传率、海量存储容量和可扩展性的要求,本文在理论指导下,对存储播放子系统进行了性能平衡设计,并建立了它的可靠性模型,计算出它的可用度,其性能完全满足电视台停播率的要求。该设计与可靠性模型对设计存储播放系统和分析系统性能均具有指导意义。

(2) 播出质量是数字视频播放系统一项最重要的技术要求。无压缩数字播放系统必须实现节目播出时的不掉帧、无黑场、且能按帧切换。传统的 PRE-CACHE 顺序存取策略不能满足此性能要求,本文提出了一种 MPPDIO 算法,并建立了其共享缓冲区的大小配置的计算模型。

实验结果表明: MPPDIO 存取算法完全能实现无压缩媒体流的不掉帧、无黑场的关键性能,既能保证播放质量,又提高了磁盘 I/O 通道的利用率,降低了 CPU 负载。传统的 PRE_CACHE 顺序存取算法则不能实现系统的这些关键性能。MPPDIO 存取算法已成功应用于电视台无压缩数字播放系统中,经过了用户两年的实际运行,效果良

好。MPPDIO 存取算法也可以在未来用于高清晰度电视的数字播放系统中。

(3)为了解决基于 MPEG2 的压缩数字视频播放系统的帧精确及两个节目之间平滑转换的问题,本文对缓冲区的设计进行了研究,提出并实现了冗余双播放缓冲区多线程调度算法,很好地解决了这一问题。指出:该算法既不影响播放帧的及时处理,又不使预解码数据量受播放缓冲区限制。与现有的压缩播放系统靠双解码卡实现帧精确与平滑转换相比,具有巧妙性,成本低和可推广性。

(4)电视台需要对某些直播节目进行实时监控,要在对节目进行审阅后,延迟一段时间再播出,因此有必要研究延迟播放技术。目前的延迟播放系统大多由大规模集成电路实现,价格昂贵,国内也较罕见,针对这种情况,本文在分析比较了四种不同处理方案的性能特点的基础上,另辟蹊径地提出了利用磁盘缓存共享压缩视频文件来实现延迟播放的新方案,并创建了一种最优算法。使用此算法可以有效地实现延迟播放功能,且不存在延迟时间限制的瓶颈,用软件实现,所需代价最小。此算法已经过一年的实际运行,达到预期效果,为延迟播放技术提供了新的思路。

(5)由于播放节目时存在实时格式转换的需求。以往的格式转换要么通过底层硬件电路,要么通过工具软件来实现,不能满足系统的需求。本文提出了一种多线程流水调度算法,根据 VOB 文件特点,对文件格式转换的各环节—解复用、编码、复用,进行优化,较好地实现了 DVD 文件、VCD 文件及 MPEG2 文件混合的无掉帧实时播放。实验结果表明:针对 VOB 文件特点的解复用器比通用解复用器提取音视频流的耗时要小得多;采用编码优化算法使得编码时间远小于播放时间及未优化算法的编码时间;且 DAT 文件及 VOB 文件到 MPEG 文件的格式转换及混合播放,可达到帧精确。

(6)为实现播出时能实时修改节目单,有必要分析播放系统服务响应时间。本文建立了数字视频播放系统服务响应时间的数学模型。指出了系统中磁盘阵列的最佳磁盘数与最佳数据分块大小。为数字播放系统的设计提供了理论依据。最后针对影响数字视频播放系统服务响应时间的因素,提出了四条改进措施。

数字播放系统是一个复杂庞大的多媒体网络系统,由于作者的理论水平、研究能力、开发时间等因素的限定,本文只是研究了其中几个关键技术问题,还有许多其它问题有待于进一步研究,如数据的自动上下载和高速传输、系统的故障监测等。同时还有如下问题需要作更深入的研究。

(1)进一步研究实时格式转换与播放的算法,使之支持更多文件格式间的格式转换和同时进行多路文件格式转换。

(2)优化存储播放服务系统的多任务动态调度算法,使之初始启动延迟更小和实时响应性更好。

致 谢

论文完成之际，我首先要感谢我的导师张江陵教授六年来对我科研上的悉心指导和生活中的热情关怀。“谁言寸草心，报得三春晖”。我自 98 年读硕士以来，张老师就一直深深教诲着我。张老师知无不言，言无不尽，从工作到科研到生活等方面至情至理的教导和无微不至的关怀，使我永远铭刻在心，可以说没有张老师的谆谆教诲，就没有我今天的博士论文，我论文上的每一丝进步都凝聚着张老师的心血。在此我谨向张老师致以最衷心的感谢和最诚挚的敬意。

在本文课题的研究过程中，我得到了许多老师和同学的大力帮助和支持，在此表示深深的谢意。

感谢冯丹教授对我无私的指导和帮助。冯老师具体指导了我的研究工作，且在论文的撰写过程中，冯老师又提出了许多宝贵的建设性意见。冯老师严谨的治学态度和敏睿的思维给我的影响很深。在此我对她表示衷心的感谢。也感谢金海教授曾在百忙之中对我的论文提出宝贵意见。

能在一个团结、民主、充满浓厚学术气氛的课题组从事研究工作，我觉得非常幸运。特别感谢同组的博士生覃灵军、宫曙光和硕士生吴戈平为本课题付出的心血和劳动。本课题的顺利完成得益于他们齐心协力的共同努力。在本课题的研究过程中，我还得到了陈进才副教授，王芳副教授，吴涛、狄晓涛、童薇、田磊、陈俭喜、肖红等人的热情帮助，在此表示感谢。

特别感谢湖北汽车工业学院的院长兼党委书记王超教授，电气工程系系主任钱新恩教授，是他们给了我这次学习的机会，满足了一个学子进一步深造的愿望。

另外，我还要感谢实验室的刘景宁副教授，周可博士，施展讲师，李洁琼老师，博士生邓玉辉、刘劲松、刘群，曾令仿，和硕士生鲁毅、黄浩丹、熊建刚以及实验室全体师弟师妹们的支持和帮助。

最后，我要感谢我的父母、我的公公、婆婆给我的鼓励、支持和无私的爱。同时我要深深地感谢我的丈夫肖忠先生，六年来他作为我坚强的后盾，对我充分理解并全力支持，没有他的理解和支持，也没有我的今天。

参考文献

- [1] 赵春涛. 数字网络时代的电视台. In: 徐威. 北京国际电视技术研讨会文集. 北京, 2000. 北京: 中国广播电视出版社, 2000: 154-158
- [2] 喻春轩. 新世纪的数字网络电视. In: 徐威. 北京国际电视技术研讨会文集. 北京, 2000. 北京: 中国广播电视出版社, 2000: 134-142
- [3] Peter Marshall. Impact of Computer Technology and Multimedia on Traditional Television. In: 徐威. 北京国际电视技术研讨会文集. 北京, 2000. 北京: 中国广播电视出版社, 2000: 103-113
- [4] Kon, F., Campbell, R. H., Nahrstedt, K. Using dynamic configuration to manage a scalable multimedia distribution system, *Computer Communications*, January 1, 2001, 24(1): 105-123
- [5] Francisco Javier Ruiz; ángel Fernández; Carlos Miguel; et. al., Multimedia systems based on satellite technology, *Computer Networks and ISDN Systems*, September 30, 1998, 30(16-18): 1543-1549
- [6] Chen, B., Sundberg, C.-E. W. An integrated error correction and detection system for digital audio broadcasting, *Broadcasting. IEEE Transactions on*, March 2000, 46(1): 68-78
- [7] Jian-Guo Zhang. Design of integrated services digital broadcasting systems using multirate optical fiber code-division multiplexing, *Broadcasting. IEEE Transactions on*, Sept. 1999, 45(3): 283-293
- [8] Moriyama, S., Tsuchida, K., Sasaki, M., et. al. Digital outside-broadcasting-link using OFDM modulation scheme, *Broadcasting. IEEE Transactions on*, Sept. 1996, 42(3): 266-271
- [9] Isobe, T., Ohya, A., Kai, K., et. al. Interactivity in broadcasting and its application to ISDB services, *Broadcasting. IEEE Transactions on*, Sept. 1996, 42(3): 179-186
- [10] Sostawa, B., Speidel, J. Investigations on bit error performance for video over DAB, *Broadcasting. IEEE Transactions on*, Dec. 1998, 44(4): 445-448
- [11] Kai, K., Seno, H., Ueno, M., et. al. TV newspapers in ISDB-multimedia information broadcasting services, *Broadcasting. IEEE Transactions on*, Sept. 1996, 42(3): 187-193
- [12] 数字电视的技术和发展, 华为技术报, 2003. 03. 15(171)
- [13] Digital HDTV Broadcasting for the 21st Century Experimented during the Kyushu-Okinawa Summit, NHK Broadcasting Services Roundup, Summer,

- 2000(13)
- [14] Andrea Donaldson. MCMA prepares for the age of digitalization. Daily Egyptian, Tuesday, February 1, 2000
 - [15] Television Management Conference. Available on line as <http://www.nab.org/conventions/nab2002/televisionconference.asp>
 - [16] Margarita Ledo Andión. Towards an Integrated European Minority Language Television Service, Available on line as http://www.usc.es/oda/pdfs/bangor/ingles_bangor.pdf
 - [17] Fully digital, fully operational, <http://www.eurovision.net/about/nslt/nslt3.pdf>
 - [18] AIB NEWS for OCTOBER 2002, Available on line as <http://www.aibcast.demon.co.uk/newsoctober.html>
 - [19] 武卫平. 福建电视台数字化实施方案及其相关思路. 广播与电视技术, 2001(10)
 - [20] 陈丰. 中小电视台数字化进程探讨. 电视工程, 2001(2): 2-4
 - [21] 汪来玖. 非线性制播网在电视台数字化、网络化中所起的作用. 广播电视信息, 2001(8): 41-44
 - [22] 饶刚, 陈明梁. 关于电视台数字化问题的探讨. 广播与电视技术, 1996(2): 10-15
 - [23] 林美爱. 中小型电视台数字化过渡的构想. 广播与电视技术, 2001(5): 92-94
 - [24] 黄伟. 采编播一体做节目: 视频服务器网络在湖南经济电视台的应用. 中国计算机报, 2001, 6, 14, 总期号: 1028 本年期号: 43
 - [25] 韩伟. 珠海台在播出网络化安全方面的考虑. 广播电视与制作, 2002
 - [26] Usui, K., Kai, K., Ohya, A., et. al. Multimedia services in the HDTV MUSE system, Broadcasting. IEEE Transactions on, Sept. 1996, 42(3): 194-199
 - [27] Kawai, N., Namba, S., Yamazaki, S. Performance of multimedia broadcasting through ISDB transmission system, Broadcasting. IEEE Transactions on, Sept. 1996, 42(3): 151-158
 - [28] Kurioka, T., Minami, H., Okuda, H. Ohshima, E., A new video hard disk system for multimedia program production, Broadcasting, IEEE Transactions on, Sept. 1996, 42(3): 222-228
 - [29] Atzori, L., De Natale, F. G. B., Gregorio, M. D., et. al., Multimedia information broadcasting using digital TV channels, Broadcasting. IEEE Transactions on, Dec. 1997, 43(4): 383-392
 - [30] Kon, F., Campbell, R. H., Nahrstedt, K. Using dynamic configuration to manage a scalable multimedia distribution system. Computer Communications, January 1, 2001, 24(1): 105-123
 - [31] Saito, T., Murakami, K., Nakagawa, H., Nishida, H., 2. 6 GHz band satellite ISDB system for mobile reception, Broadcasting. IEEE Transactions on, Sept. 1996, 42(3):
-

- 167-172
- [32] 陶黎生, 胡强. 基于 PDR300 的多通道硬盘播出系统. 电视中心技术, 1999(1): 66-72
- [33] 苏琦. 浅谈硬盘播出系统. 电视工程, 1999(4): 21-24
- [34] 王常生, 邝榆宁. 硬盘播出系统的改造. 世界广播电视, 2002(4): 20-21
- [35] 王伟, 陶黎生. 网络化自动播出系统的设计与实现. 广播与电视技术, 2001(8): 87-91
- [36] Tsao Shiao-Li, Huang Yueh-Min, Making a cost-effective storage server for broadcasting digital video services, IEEE Transactions on Broadcasting Sep, 1998, 44(3): 300-308
- [37] Okuda H., Kurioka T., Storage technology for television home server, Journal of Magnetism and Magnetic Materials, October 2001, 235(1-3): 20-24
- [38] 卞德森, 李韶红. 电视播出系统的现状和发展趋势探讨. 广播与电视技术, 2001(12): 61-68
- [39] 宋磊, 霍德存, 刘建宁. 数字播控技术探讨. 电视技术, 2001(1): 80-82
- [40] 李晓华. HDTV/DTV 的迅速发展途径. 电子器件, 2000.9, 23 (3) : 227-232
- [41] 巨改生. 电视发射台系统可靠性分析[J]. 西部广播电视, 1999(12): 9-12
- [42] 张江陵, 冯丹. 海量信息存储[M]. 北京: 科学出版社, 2003
- [43] 疏松桂. 控制系统可靠性分析与综合[M]. 北京: 科学出版社, 1992
- [44] Wyss, Gregory D., Schriener, Heather K., Gaylor, Timothy R. Probabilistic logic modeling of network reliability for hybrid network architectures, Proceedings of the 1996 21st Conference on Local Computer Networks, Oct 13-16 1996, Minneapolis, MN, USA, IEEE 1996:404-413
- [45] Schulze M., Gibson G., Katz R., Patterson D.A., How reliable is a RAID, COMPCON Spring '89, Thirty-Fourth IEEE Computer Society International Conference: Intellectual Leverage, Digest of Papers. , 27 Feb.-3 March 1989:118 - 123
- [46] Yu, David C., Bayesian Nguyen, Thanh C. Haddawy, Peter network model for reliability assessment of power systems. IEEE Transactions on Power Systems, May, 1999, 14(2): 426-432
- [47] Fitzgerald, Kent (TRW); Latifi, Shahram; Srimani, Pradip K. Reliability modeling and assessment of the star-graph networks. IEEE Transactions on Reliability, March, 2002, 51(1): 49-57
- [48] Welke, Stephen R., Johnson, Barry W., Aylor, James H. Reliability modeling of hardware/software systems. IEEE Transactions on Reliability, Sept, 1995, 44(3): 413-418
-

- [49] 周兴铭, 徐明, 肖刚. 多线程技术的现状与前景展望. 计算机工程与科学, 1999, 4(21): 1-6
- [50] Tingting Zhang, Youshi Xu. Unequal packet loss protection for layered video transmission, Broadcasting. IEEE Transactions on, June 1999, 45(2): 243-252
- [51] Ahmed Karmouch, James Emery. A playback schedule model for multimedia documents. IEEE Multimedia, Spring 1996: 50-61
- [52] Hui, J. Y., Karasan, E., Li, J., Zhang, J. Client-server synchronization and buffering for variable rate multimedia retrievals, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 226-237
- [53] Heng-Yow Chen, Ja-Ling Wu, MultiSync: a synchronization model for multimedia systems, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 238-248
- [54] Radhika Thekkath and Susan J. Eggers, The Effectiveness of Multiple Hardware Contexts, ACM 1 994
- [55] Pinnacle System Nexus CODI SDK, Programming Reference Guide Version 2. 0 Beta 4, 2001
- [56] Li-Shen Juhn; Li-Ming Tseng, Adaptive fast data broadcasting scheme for video-on-demand service, Broadcasting. IEEE Transactions on, June 1998, 44(2): 182-185
- [57] Hui, J. Y., Karasan, E., Li, J., Zhang, J. Client-server synchronization and buffering for variable rate multimedia retrievals, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 226-237
- [58] Su, Z., Washizawa, T., Katto, J., et. al. Performance improvement of graceful image caching by using request frequency based prefetching algorithms, Electrical and Electronic Technology, 2001. TENCON. Proceedings of IEEE Region 10 International Conference on, 2001, 1: 370-376
- [59] Mumolo, E., Bernardis, G. A novel demand prefetching algorithm based on Volterra adaptive prediction for virtual memory management systems, System Sciences, 1997, Proceedings of the Thirtieth Hawaii International Conference on, 1997, 5: 160-167
- [60] Fitzek, Reisslein, prefetching protocol for continuous media streaming in wireless environments, Selected Areas in Communications, IEEE Journal on, Oct. 2001 19(10): 2015- 2028
- [61] Min-You Wu, Wei Shu. Efficient support for interactive browsing operations in clustered CBR video servers, Multimedia. IEEE Transactions March 2002, 4(1): 48-58
- [62] Harizopoulos, S., Harizakis, C., Triantafillou, P. Hierarchical caching and

- prefetching for continuous media servers with smart disks. *IEEE Concurrency*, July-Sept. 2000, 8(3): 16-22
- [63] 黄永峰. 因特网媒体网关的关键技术研究. 华中理工大学博士学位论文, 2000, 7: 28-31
- [64] Chang-Wook Kim., Jung-Ahn Han, Hwan-Gyu Lee, Woo-Zoo Kim, Byung-Gi Kim, Moon-Ho Lee. An efficient buffer control and packet scheduling for the access point of IP networks and ATM network. *Parallel and Distributed Systems, ICPADS 2001 Proceedings*, 2001: 675-678
- [65] Tsang-Ling Sheu, Jiin-Tarnng Fang. A buffer allocation mechanism for VBR video playback. *Communication Technology Proceedings, WCC- ICCT 2000*, 2: 1641-1644
- [66] La Corte, A., Lombardo, A., Palazzo, S. Buffer compensation analysis for synchronization of multimedia services in wireless networks. *Global Telecommunications Conference, GLOBECOM 1998*, 1: 101-106
- [67] Yuen J., Kam-Yiu Lam, Chan E. Adaptive buffer sensitive scheduling for transmitting video streams in a mobile multimedia system. *Real-Time Technology and Applications Symposium, 2001. Proceedings, Seventh IEEE*, 2001: 125-126
- [68] Orallo E. H., Vila-Carbo J. An optimal method for allocating buffer and bandwidth for MPEG traffic in a network node. *Euromicro Conference, Proceedings. 27th*, 2001: 354-361
- [69] Hac A., Xue C. X. Buffer control scheme in multimedia synchronization. *Performance, Computing, and Communications Conference, 1997, IPCCC 1997, IEEE International*, 1997: 516-522
- [70] Blakowski G., Steinmetz R. A media synchronization survey: reference model, specification, and case studies. *Selected Areas in Communications. IEEE Journal on*, Jan. 1996, 14(1): 5-35
- [71] 宋军, 顾冠群. 多媒体通信媒体间同步技术综述. *电信科学*, 1996, 12(9): 14-19
- [72] 张海亮. 浅议电视信号的数字化与码率压缩. *广播电视网络技术*, 1999(2): 10-12
- [73] 赵健, 俞卞章. MPEG 的发展与前瞻. *电视技术*, 2002(3): 17-21
- [74] 何锦池. MPEG-2 存储格式选择的探讨. *广播与电视技术*, 2002(2): 47-49
- [75] 邱林海, 周权利, 余胜生. 几种典型图像编码标准及其 QOS 评估. *计算机工程与应用*, 1999, 8: 23
- [76] 高文, 吴枫. MPEG—4 编码的现状和研究. *计算机研究与发展*, 1999, 36 (6) : 642
- [77] overview of MPEG—4 Standard. ISO / IEC JTC1 / SC29 / WG11, N2323. Dublin,

- Ireland, 1998
- [78] 章毓晋. 基于内容的多媒体信息检索 CMIR 与国际标准 MPEG—7. 第九届全国图像图形学学术会议论文集, 13
 - [79] 章毓晋. 国际标准 MPEG—7 问答. 中国图像图形学报, 2000, 5 (11—12) : 7
 - [80] 章毓晋. MPEG—21 刚开始制订的国际标准. 中国图像图形学报, 2000, 9~10(B): 12—13
 - [81] Sun, Huifang, Zdepski, Joel W., Kwok, Wilson, Raychaudhuri, D. Error concealment algorithms for robust decoding of MPEG compressed video, *Signal Processing: Image Communication*, September, 1997, 10(4): 249-268
 - [82] Wang Shuo, Lin-xin Ding, Zhang Jiang-Ling. THE ACCESSING AND PLAYING OF MULTIMEDIA STREAMS OF UNCOMPRESSED DIGITAL PLAYING SYSTEM. *Proceedings of the Fourth IASTED International Conference SIGNAL AND IMAGE PROCESSING August 12-14, 2002, Kaua'i, Hawaii, USA, P654-659*
 - [83] 张江陵, 王烁, 冯丹. 无压缩数字播放系统中媒体流的多线程存取控制策略. *小型微型计算机系统*, 2003, 11: 1996-1999
 - [84] 王烁, 张江陵. 电视台无压缩数字播放系统中的缓冲管理模型. *计算机工程与科学*, 2004, 3: 49-52
 - [85] Shuo Wang, Jiangling Zhang, Dan Feng. Multi-thread IO in High Quality Digital Broadcasting System. *The 2003 International Conference on Computer Networks and Mobile Computing (ICCNMC-03), IEEE Computer Society, October 20~23, 2003, Shanghai China, 442-445*
 - [86] Marc Farley, *Building Storage Network*, McGraw-Hill Companies, 2000
 - [87] 宫曙光, 冯丹. 小型电视台视频播放系统的实现技术及其播放策略分析. *计算机工程*, 2003.5, 29(7): 44-46
 - [88] S. Chen and D. Towsley. The Design and Evaluation of RAID 5 and Parity Striping Disk Array Architectures. *Journal of Parallel and Distributed Computing*, January 1993, 17
 - [89] M. Y. Kim and A. N. Tantawi. Asynchronous Disk Interleaving: Approximating Access Delays. *IEEE Trans. on Computer*, 1991
 - [90] Cooper, R B. *Introduction to Queuing Theory*. New York: the Macmillan Company, 1981: 235-240
 - [91] Chen J S C, Stern T E. Performance of a generic non-blocking packet switch. In *proc. Of 4th international conference on data communication System and their performance*, Barcelona, 1990, Elsever Science publishers B. V. 1990: 315-323
 - [92] 贺飞云. ATM 交换的最佳缓存器分配. *通信技术*, 1995, 89 (2) : 1-6

- [93] 孙绵嵩. 中小电视台直播系统的构成与应用. 广播与电视工程, 1999(8): 132-136
- [94] 郭延芬. 可编程数字式语音信号延迟系统. 电子技术, 1995(4): 16-17
- [95] 吴勇, 张光烈, 郑南宁. 扫描格式转换的自适应帧频提升算法及其 FPGA 实现. 中国工程科学, 2001, 3 (4) : 56-62
- [96] 朱秀昌. 会议电视图像编解码器中的视频格式转换. 南京邮电学院学报, 1996, 16 (4) : 26-28
- [97] 张光烈, 郑南宁, 吴勇. 面向格式转换的数字视频处理方法及其硬件实现. 中国工程科学, 2001, 3 (6) : 41-47
- [98] 王栋, 许晓鸣, 王玮. 基于运动补偿的视频格式转换技术综述. 计算机工程与应用, 2001, 15: 49-51
- [99] 计新, 金虎范. VCD 信息结构与音视频信息处理. 延边大学学报(自然科学版), 1998 (6) : 70-72
- [100] Stoll, G. MPEG audio layer II: A generic coding standard for two and multichannel sound for DVB, DAB and computer multimedia. Broadcasting Convention. 1995, IBC 95., International, 14-18 Sep 1995: 136-144
- [101] Pan, D. A tutorial on MPEG/audio compression. Multimedia, IEEE, Summer 1995, 2(2): 60-74
- [102] Kumar, M., Zubair, M. A high performance software implementation of MPEG audio encoder. Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on, 7-10 May 1996, 2: 1049-1052
- [103] Konstantinides, K. Fast subband filtering in MPEG audio coding. IEEE Signal Processing Letters, Feb 1994, 1(2): 26-28
- [104] 何东梅, 高文. 高质量、低复杂度的纯软件实时 MPEG 音频编、解码器. 计算机工程与应用, 1999 (12) : 7-10
- [105] Chen-Wei Shih, Nam Ling. Polyphase filter architectures for MPEG audio using fast IDCT. Signals, Systems & Computers, 1997. Conference Record of the Thirty-First Asilomar Conference on, 2-5 Nov 1997, 1: 416-420
- [106] W Saidi, A.. Generalized FFT algorithm. Communications, 1993. ICC 93. Geneva. Technical Program, Conference Record. IEEE International Conference on, 23-26 May 1993, 1: 227-231
- [107] Yi-Shin Tun, Chia-Chiang Ho, Ja-Ling Wu. MMX-based DCT and MC algorithms for real-time pure software MPEG decoding. Multimedia Computing and Systems. 1999. IEEE International Conference on Jul 1999, 1: 357-362

- [108] 覃灵军, 冯丹. MPEG-2 程序流实时复用器的软件实现. 计算机工程, 2003.8, 29 (13) : 54-56
- [109] 陈维强, 高文. MPEG-2 多路编码流系统复用器. 哈尔滨工业大学学报, 2000, 4(2): 12-15
- [110] 钟玉琢, 乔秉新, 祁卫. 运动图象及其伴音通用编码国际标准--MPEG-2. 北京: 清华大学出版社, 1997: 108-111
- [111] Pham, H., Upadhyaya, S. J. Reliability analysis of a class of fault-tolerant systems[J]. IEEE Transactions on Reliability. Aug. 1989, 38(3): 333-337
- [112] Thomas D. C. Little. Synchronization and Storage Model for Multimedia Objects. IEEE Journal on Selected Areas in Comm., April 1990, 3(8): 413-427
- [113] 贾建平. 多媒体视频与音频同步的探讨. 现代计算机, 1995, 9 (44) : 16-20
- [114] Gerold Blakowski. Tool Support for the Synchronization and Presentation of Distributed Multimedia. Computer Communication, Dec. 1992, 10(15): 611-618
- [115] Blakowski, G., Steinmetz, R. A media synchronization survey: reference model, specification, and case studies, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 5-35
- [116] Perez-Luque, M. J., Little, T. D. C. A temporal reference framework for multimedia synchronization, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 36-51
- [117] Rangan, P. V., Kumar, S. S., Rajan, S. Continuity and synchronization in MPEG, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 52-60
- [118] Haindl, M. A new multimedia synchronization model, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 73-83
- [119] Chun-Chuan Yang, Jau-Hsiung Huang. A multimedia synchronization model and its implementation in transport protocols, Selected Areas in Communications. IEEE Journal on, Jan. 1996, 14(1): 212-225
- [120] Cruz, R. L. A calculus for network delay. II. Network analysis, Information Theory. IEEE Transactions on, Jan. 1991, 37(1): 132-141
- [121] Cruz, R. L. A calculus for network delay. I. Network elements in isolation, Information Theory. IEEE Transactions on, Jan. 1991, 37(1): 114-131
- [122] Cidon, I., Gopal, I., Grover, G., Sidi, M. Real-time packet switching: a performance analysis, Selected Areas in Communications. IEEE Journal on, Dec. 1988, 6(9): 1576-1586
- [123] Philp, I. R., Liu, J. W. S. End-to-end scheduling in real-time packet-switched networks, Network Protocols, 1996. Proceedings. 1996 International Conference on, 29 Oct.-1 Nov. 1996: 23-30

- [124] Minseok Song, Naehyuck Chang, Heonshik Shin. A new queue discipline for various delay and jitter requirements in real-time packet-switched networks. *Real-Time Computing Systems and Applications*, 2000. Proceedings. Seventh International Conference on, 12-14 Dec. 2000: 191-198
- [125] Fong-Hao Liu, Yang, C. C., Wei-Tsong Lee. The building, analyzing and simulating of a multimedia network based on CATV system, Broadcasting. *IEEE Transactions on*, March 2000, 46(1): 79-87
- [126] 千兆以太网卡测试方法, <http://online-ddd.com>
- [127] 施驰, 王宏远, 黄载禄. MPEG-2 编解码系统的恒定延迟技术. *电子学报*, 2000, 28 (6) : 131-134
- [128] 卢炎声, 谢晓东, 朱英武. 分布式实时数据库的通讯延迟模型与协议. *软件学报*, 1998, 9 (3) : 237-230

附录 1 攻读学位期间发表的学术论文目录

- [1] Shuo Wang, Jiangling Zhang, Dan Feng. Multi-thread IO in High Quality Digital Broadcasting System, The 2003 International Conference on Computer Networks and Mobile Computing (ICCNMC-03). IEEE Computer Society, October 20~23, 2003, Shanghai China, 442-445
- [2] Wang Shuo, Lin-xin Ding, Zhang Jiang-Ling. The Accessing and Playing of Multimedia Streams of Uncompressed Digital Playing System, Proceedings of the Fourth IASTED International Conference SIGNAL AND IMAGE PROCESSING, August 12-14, 2002, Kaua'I, Hawaii, USA, P654-659
- [3] 张江陵, 王烁, 冯丹. 无压缩数字播放系统中媒体流的多线程存取控制策略. 小型微型计算机系统, 2003(11): 1996-1999
- [4] 王烁, 张江陵. 电视台无压缩数字播放系统中的缓冲管理模型. 计算机工程与科学, 2004, 3: 49-52
- [5] 王烁, 张江陵, 冯丹. 电视播放系统中数据存储子系统的研究与实现. 计算机工程与科学, 2004(8)(已录用)
- [6] Shuo Wang, Jiangling Zhang, Dan Feng. The Response Time Analysis in High Quality Digital Broadcasting System(待发表)
- [7] 王烁, 张江陵, 冯丹. 高质量数字播放系统中延迟播放的研究与实现(待发表)
- [8] 王烁, 张江陵, 覃灵军. 数字播放系统中实时格式转换的研究与实现(待发表)

电视台数字化全自动视频服务系统 验收报告

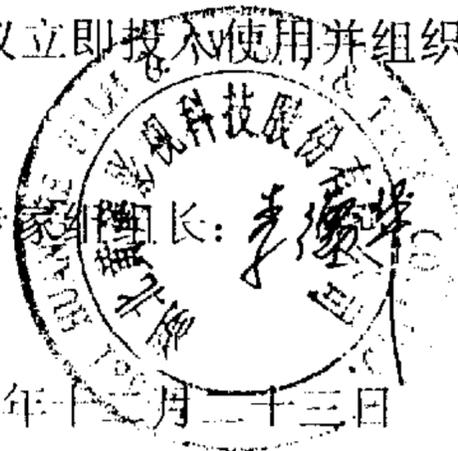
2002年12月23日，由湖北黄鹤影视科技股份有限公司（以下简称公司）主持、由公司资深专家组成验收专家组对华中科技大学外存储国家专业实验室研制的“电视台数字化全自动视频服务系统”进行验收。验收组审查了系统研制技术报告、用户手册、系统软件说明、系统硬件配置说明、测试报告等技术文件，考察了全套设备和观看了现场演示，经认真讨论和评审后认为：

- 1、该课题组已高质量完成了公司委托的研究开发任务，除完成合同上原规定的内容之外，应公司要求增加了按MPEG2规范的播放、直播和延迟播放等内容，超额完成了研发任务。
- 2、系统实现了电视台数字化全自动视频服务系统的无压缩和MPEG2压缩两种软件版本，并可运行于同一系统硬件环境。
- 3、软件系统采用分布式组件化设计与实现方法，易于移植到不同的操作系统平台；硬件系统构架兼顾了模拟信号播出到数字化播出的过渡需求，同时亦具有从标清播出到高清播出的扩展能力。
- 4、本系统使用方便，人机界面友善，运行安全可靠，系统设计合理，实现技术先进。
- 5、技术文档符合规范，内容齐备。

验收组一致认为，该系统的无压缩播放属国内首创，压缩播放具有自主知识产权，处于国内领先水平，建议立即投入使用并组织批量生产。

验收专家组组长：李德华

二零零二年十二月二十三日



主要研制人员名单

序号	姓名	性别	出生年月	技术职称	文化程度	工作单位	具体工作内容
1	冯丹	女	1970.5	教授	博士	华中科技大学	项目负责人
2	陈进才	男	1960.12	副教授	博士后	华中科技大学	总体设计与总控实现
3	周可	男	1974.6	讲师	博士生	华中科技大学	磁盘阵列控制器
4	施展	男	1976.8	助教	硕士	华中科技大学	设备购置与调试
5	王烁	女	1971.9	讲师	博士生	华中科技大学	非压缩播放与压缩延迟
6	吴涛	男	1975.9		博士生	华中科技大学	上下载客户、时钟同步与监控
7	覃灵军	男	1975.4		硕士生	华中科技大学	压缩采集与上下载调度
8	宫曙光	男	1973.12		硕士生	华中科技大学	压缩播放及服务器控制
9	吴戈平	女	1977.4		硕士生	华中科技大学	士下载服务
10	狄晓涛	男	1977.1		硕士	华中科技大学	总控界面与系统集成
11	肖红	女	1972.5		硕士	华中科技大学	无压缩采集
12	陈俭喜	男	1976.2		博士生	华中科技大学	磁盘阵列控制器
13	童薇	女	1977.3		博士生	华中科技大学	磁盘阵列控制器
14	田磊	男	1978.6		硕士生	华中科技大学	系统状态组织与声音报警