



申请同济大学工学硕士学位论文

基于嵌入式操作系统的 数模一体机开发的研究

(国家 863 “软件重大专项” 项目 编号: 2001AA113400)

培养单位: 电子与信息工程学院

一级学科: 计算机科学与技术

二级学科: 计算机应用

研 究 生: 沈小飞

指导教师: 顾伟楠 教授

二〇〇五年二月



A dissertation submitted to
Tongji University in conformity with the requirements for
the degree of Master of Science

The Study of Combination of Digital and Analog TV Based on Embedded Operating System

(Supported by National 863 High Tech Program, Grant No. 2001AA113400)

School/Department: School of Electronics and
Information Engineering

Discipline: Computer Science and Technology

Major: Computer Application

Candidate: Shen Xiao Fei

Supervisor: Prof. Wei-Nan Gu

February, 2005

基于嵌入式操作系统的数模一体机开发的研究

沈小飞

同济大学

学位论文版权使用授权书

本人完全了解同济大学关于收集、保存、使用学位论文的规定，同意如下各项内容：按照学校要求提交学位论文的印刷本和电子版本；学校有权保存学位论文的印刷本和电子版，并采用影印、缩印、扫描、数字化或其它手段保存论文；学校有权提供目录检索以及提供本学位论文全文或者部分的阅览服务；学校有权按有关规定向国家有关部门或者机构送交论文的复印件和电子版；在不以赢利为目的的前提下，学校可以适当复制论文的部分或全部内容用于学术活动。

学位论文作者签名：

年 月 日

经指导教师同意，本学位论文属于保密，在 年解密后适用本授权书。

指导教师签名：

学位论文作者签名：

年 月 日

年 月 日

同济大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

签名：

年 月 日

摘要

数字电视(DTV)是继黑白电视和彩色电视之后的第三代电视,由于采用了数字信号,因此使得计算机技术的介入和发展成为了可能,在这样的背景下,嵌入式操作系统将逐渐成为数字电视软件开发的标准平台。

国内数字电视产业化的进程将会明显加快,而传统的模拟电视仍然在一定时间内存在,所以数模一体机的开发正适应了当前的市场的需求。本文结合国家 863 软件重大专项课题“基于中间件技术的因特网嵌入式操作系统及跨操作系统的中间件运行平台”,提出了一种基于“和欣”嵌入式操作系统来提供 DTV 软件开发平台中间件的解决方案。“和欣”嵌入式操作系统是完全面向构件的操作系统,操作系统提供的功能模块全部基于构件技术,都是可拆卸的构件。该数模一体机系统分为模拟模块和数字模块两部分。数字模块正是基于“和欣”嵌入式操作系统,而模拟模块的实现则采用了传统的数字电视开发方法,相比之下有一定的局限性。两者之间使用串口进行通信。上层的可视化开发工具——DTV Visual Developer 则为开发者提供了一个数模一体机软件开发的可视化集成开发环境,它将一些原本分散的功能要素封装成为具有一定属性并可以响应相应事件的控件,这样就可以借助可视化开发方法来提高数模一体机软件开发的效率。

关键词: 数字电视, 构件技术, 中间件

ABSTRACT

The digital TV (DTV) is the third generation of TV products following black-and-white TV and color TV. Because of the introduction of the digital signal it is feasible to apply computer technology to DTV products. Under the background, embedded operating system will be the standard platform of DTV software development.

In the DTV system, it is a key problem to allow applications run in a transparent environment without caring about the detail of the infrastructure, such as different operating systems and hardware. With the background of the work in our project "Component oriented Embedded Networking Technology" which belongs to National 863 High Tech Program (863 Program), The solution of DTV software development platform based on "Elastos" is presented. "Elastos" is a fully component based operating system. All the functional module of the operating system are based on ezCOM technology, so it's easy to develop system level components to expand the functions of operating system and applications. The architecture of DTV software development platform has three layers. It runs in component run-time platform as system level component. "DTV foundational components library" and "DTV explanation layer" compose the DTV middleware. Both DTV middleware and visual development tool-"DTV Visual Developer" constitute DTV software development platform. "DTV foundational components library" runs in ezCOM component run-time platform. It provides the encapsulation of DTV chipset functions and serves the "DTV explanation layer" above. "DTV explanation layer" is a component collection, which encapsulates the DTV basic functions. "DTV Visual Developer" acts as an IDE (Integrated Development Environment) that combines the separate elements to the "controls" which have some special attributes and can respond the corresponding events.

Key Words: DTV, component technology, middleware

目录

目录	IV
第 1 章 引言	1
1.1 数字电视的技术现状	1
1.1.1 数字电视和高清晰度电视	2
1.1.2 数字电视的特点	3
1.1.3 数字电视的技术标准	3
1.2 数字电视的市场发展	4
1.3 课题来源及研究意义	6
1.4 论文结构	6
第 2 章 “和欣” 嵌入式操作系统及其构件技术	6
2.1 “和欣” 操作系统概述	6
2.1.1 “和欣” 操作系统简介	6
2.1.2 和欣灵活内核简介	7
2.1.3 和欣操作系统提供的功能	7
2.1.4 和欣操作系统的应用软件开发	8
2.1.5 和欣操作系统的优势	9
2.2 和欣构件运行平台	11
2.2.1 和欣构件运行平台简介	11
2.2.2 和欣构件运行平台的功能	11
2.2.3 和欣构件运行平台的技术优势	13
2.2.4 利用和欣构件运行平台编程	13
2.3 ezCOM 构件技术	14
2.3.1 ezCOM 技术的由来	14
2.3.2 ezCOM 构件技术概要	16
2.3.3 ezCOM 技术的意义	16
2.3.4 ezCOM 技术对软件工程的作用	17
2.3.5 ezCOM 技术在 “和欣” 技术体系中的作用	19
2.3.6 如何用 ezCOM 技术编程	19

第 3 章 模拟模块的系统结构和实现	26
3.1 ATV硬件平台结构	26
3.2 ATV软件系统设计	27
3.2.1 软件的总体架构设计	27
3.2.2 事件的处理	29
3.2.3 模式的处理	34
3.2.4 用户接口的实现	36
3.3 ATV软件开发的缺陷	44
第 4 章 数字模块的系统结构和实现	32
4.1 DTV硬件平台结构	32
4.1.1 硬件组成	32
4.1.2 关键技术	33
4.2 ATSC标准	34
4.3 DTV软件架构	34
4.4 DTV中间件的设计与实现	35
4.4.1 DB模块的设计	36
4.4.2 Parser模块的设计	36
4.4.3 Player模块的设计	37
4.4.4 EPG模块的设计	37
4.5 DTV中间件平台中的消息机制	38
4.5.1 DTV中间件平台中的消息定义	38
4.5.2 DTV中间件平台中的消息流分析	39
第 5 章 数模一体机通讯的设计和实现	51
5.1 数模通讯的整体设计方案	51
5.1.1 串口通讯机制	51
5.1.2 发送机制	51
5.2 数模通讯的具体实现	51
5.2.1 数据结构的定义	52
5.2.2 发送命令的定义	52
第 6 章 可视化开发工具——DTV Visual Developer的设计	54

6.1 可视化开发方法简介	54
6.2 可视化开发方法在DTV软件开发中的应用	54
6.3 DTV Visual Developer的系统设计	55
6.3.1 DTV Visual Developer的系统构成	56
6.3.2 DTV Visual Developer的设计要点	57
6.4 基于DTV Visual Developer的数字电视软件开发方法	58
第7章 结论与展望	58
致谢	59
参考文献	60
个人简历 在读期间发表的学术论文与研究成果	62

第1章 引言

1.1 数字电视的技术现状

自上一世纪20年代电视问世以来,这一伟大发明给人们的生活、娱乐、工作和学习带来了无可估量的作用。从早期的黑白电视到上世纪六十年代彩电问世,到数字电视和高清晰度电视的诞生,无一不引起人们的极大关注和带来播放、家电产品的革命。电视对于当今世界的任何国家来说,都是最重要的消费电子产品,它已经不仅仅是一种电子产品,而是成为了改变人们生活方式的工具。

21世纪是数字化的信息时代,数字电视和高清晰度电视(HDTV)成为人们关注的焦点,国内市场上的电视产品也纷纷打数字电视和高清电视牌来吸引消费者,尤其是最近闹得沸沸扬扬的高清电视概念,风头一时无两。

数字电视是继黑白模拟电视,彩色模拟电视之后的第三代电视。区别于传统电视,它是在拍摄、编辑、制作、传输、播出、接收电视信号的全过程都使用数字技术的电视系统。采用数字技术不仅可以使电视设备获得比模拟电视更高的显示效果,而且还具有模拟技术不能提供的诸如视频点播、数字广播、个性化交互电视、远程教育、Internet、三网合一、电视电子商务和日常信息综合服务增值服务项目。影视数字化从根本上改变了影视的命运。影视节目的制作和播放,由于数字化方式的加入,而变得更加多元化,随机化,全球化和可追求化。

数字电视的诞生是为了满足人们的视觉和控制的需要。数字电视的信号不再是模拟信号,而是采用以数字形式进行传输、处理、存储的数字信号。由于数字电视信号的存储和处理电路便于大规模和超大规模的集成,因而其设备比模拟电路设备元件少,便于调整,其重量轻、体积小、功耗少,寿命长且可靠性高,容易与计算机以及其它数字化设备接口,适合于公用数据通讯网,便于实现生产、运行的自动化和视听信息处理的综合化、网络化。计算机技术的发

展和介入，使得正处于方兴未艾的电视工业得到了新的支持，带来了又一次巨大变革的历史机遇。

1.1.1 数字电视和高清晰度电视

现有的彩色电视包括以下几种不同的制式——欧洲和我国采取的是 PAL 制式，美国和日本采取 NTSC 制式，前苏联采取 SECAM 制式，但利用人的视觉暂留原理顺序扫描、同步扫描的原理却是一样的，因而面临的问题也是相同的，即由于扫描行数的限制而造成的清晰度不够理想。为了提高电视图像的分辨率，从 70 年代开始，工业发达国家开始了对高清晰度电视系统的研究工作。这个工作最早是从日本的 NHK 开始的，到了 80 年代初获得成效，制作了 1125 线的数字电视机，60 场/秒，2：1 隔行扫描标准的高清晰度电视，简称 HDTV。

数字电视概念的含义不仅是指我们一般人家中的电视接收机，而是包含了从发送、传输到接收的全过程。由电视台送出的图像及声音信号，经数字压缩和数字调制后，形成数字电视信号，经过空中无线方式或电缆有线方式传送，由数字电视接收机接收后，通过数字解调和数字视音频解码处理还原出原来的图像及伴音。因此，数字电视就是在电视台播出节目和电视机接收节目全过程都采用数字技术进行处理的电视。它与目前的模拟电视系统在传输方式上是完全不同的。电视节目从摄制、编辑、播送、传输、接收到显示的全过程均采用全数字化的技术处理，因此，信号在整个过程中的损失大大减小，接收到的电视节目质量可以达到与演播现场一样的水平。

真正的数字电视并不等于人们常说的 HDTV，它包括普及型数字电视（PDTV）或者低清晰度数字电视（LDTV）、标准清晰度数字电视（SDTV）和高清晰度数字电视（HDTV）。

到了 90 年代，形成了日本、欧洲、美国三大数字电视制式。日本和欧洲的两种制式出现比较早，图像压缩比较小。采用模拟信号传送，卫星播出方式适合较宽的信道传输；美国的全数字方案吸收了日本和欧洲的优点，采用数字压缩编码和数字通信技术，传送效率高，有效地压缩了宽带，适合于窄频道传输的地面广播，并且对使用相同频道的其它节目不产生干扰，实现了与先行模拟信号电视兼容过渡的根本目的。

世界各国发展数字电视的情况：美国国家电视网要在 2006 年普及数字电

视，全面停止模拟信号。英国全国由模拟电视向数字电视过渡的时间从 2006 年开始，预计 2010 年结束；日本 2001 年开播 6 套卫星高清晰度数字电视，但地面高清晰度电视要在 2003 年才会在主要的大城市开播。我国计划在 2005 年将进行数字电视的商业播出，2008 年用数字电视转播奥运会，2015 年停止模拟电视的播放，全面推行数字电视。

1.1.2 数字电视的特点

相对于传统电视，数字电视具有以下技术特点：

(1) 收视效果好，图像清晰度高，音频质量高，可以更好的满足人们感官的需求。

(2) 抗干扰能力强。数字电视不易受处界的干扰，避免了串台、串音、噪声等影响。

(3) 传输效率高。利用有线电视网中的模拟频道可以传送 8—10 套标准清晰度数字电视节目。

(4) 兼容现有模拟电视机。通过在普通电视机前加装数字机顶盒即可收视数字电视节目。

(5) 提供全新的业务。借助双向网络，数字电视不但可以实现用户自点播节目、自由选取网上的各种信息，而且可以提供多种数据增值业务。

(6) 很容易实现加密/解密和加扰/解扰技术，便于专业应用(如军用)以及广播应用。

(7) 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)的网络中传输，也便于与计算机网络联通。

1.1.3 数字电视的技术标准

数字电视的传输途径可分为三种：数字卫星电视、数字有线电视和数字地面开路电视。这三种数字电视的信源编码方式相同，都是 MPEG-2 的复用数据包，但由于它们的传输途径不同，它们的信道编码也采用了不同的调制方式。

现在国际上的数字电视存在三种标准，第一种是美国的 ATSC 标准，第二种是欧洲的 DVB 标准，第三种是日本的 ISDB 标准。加拿大、韩国、台湾、阿根廷、

墨西哥等国家和地区均采用美国的ATSC标准,香港则计划采用欧洲的DVB标准。中国如果简单选用国外标准,那么企业利润大都会用来支付专利费和授权费。另外,欧美标准在制订过程中,没有考虑多媒体兼容的概念,技术上不能支持无线互联网传输和移动便携式接收。这给我国制订自己的标准提供了技术跨越的机会。

1.2 数字电视的市场发展

世界各国发展数字电视的情况:美国国家电视网要在2006年普及数字电视,全面停止模拟信号。英国全国由模拟电视向数字电视过渡的时间从2006年开始,预计2010年结束;日本2001年开播6套卫星高清晰度数字电视,但地面高清晰度电视要在2003年才会在主要的大城市开播。而中国继中央电视台1999年国庆阅兵式在44频道试播高清晰度电视节目之后,在2000年深圳高交会期间,首次采用自主研发的四种DTV制式第天试播12小时,HDTV的产业化已提上日程。

所有这些迹象均表明:HDTV正在代替传统的模拟电视走入家庭。然而,HDTV的开播和产业化,首当其冲就是制定标准。电视标准不仅涉及节目播送的技术问题,还关系到整机的生产和利润分配等市场问题,目前欧、美、日等主要工业国在数字电视传输制上仍存在分歧,我国的高清晰度电视标准也还未最终敲定。尽管我国在高清晰度电视领域已经取得了可喜的成绩,但与先进国家相比仍有一定的差距,还有不少技术和设备需要进一步研究和开发。经过不间断的技术积累和努力,HDTV总体组已经成为国内最具数字电视技术开发实力的队伍之一。目前,全国已基本形成了4个标准方案:广研院的16QAM(6+2)方案、HDTV总体组的CN YSB和CN COFDM方案、清华大学的DMB-T(SOFDM)方案。一旦标准制定完成,我国的HDTV产业化就为时不远了。

目前,国家计委已在北京、上海、深圳三个城市进行数字电视研究开发及产业化试点,目的是对目前已经提出的数字电视和HDTV的各种方案标准进行深入地考察试验,从实践中验证,为早日制订我国的数字电视和HDTV标准作准备。随着我国加入WTO和2008年在京举办奥运会,国内数字电视产业化的进程将会明显加快,一个全新的产业也正在形成。

数字电视的推广应用。促进了电视制作、播出、传输、接收诸环节的技术

开发和设备制造的全面发展，形成一个比模拟电视更为庞大的产业网络。这个网络主要是由技术服务商、电视运营商、内容制作商、电视厂家和广大数字电视观众组成。由于数字电视已不再是传统意义上的电视机，而相当于一台32位CPU的电脑，由于数字电视接受的是二进位的数字代码，因此电视节目内容制作商有了更大的空间可以动态地控制这些代码，从而达到制作动态节目的目标，这将是电视史上的一次里程碑。类似的动态服务将比比皆是，动态服务将为我们带来除基本音视频业务之外的数字电视增值业务。由于电视节目的多样化，为内容制作商提供技术支持的技术服务商就应运而生，技术服务商的作用就相当于应用软件开发商，他们利用数字电视的操作系统所提供或支持的中间件，开发出大量的数字电视节目开发软件，为内容制作商提供强大的支持。而电视运营商将会从内容制作商那里采购所需的电视节目，编排以后提供给电视观众。电视厂家为电视观众提供数字电视，以便观众们能享用丰富的电视节目。

以下是数字电视产业网的示意图：

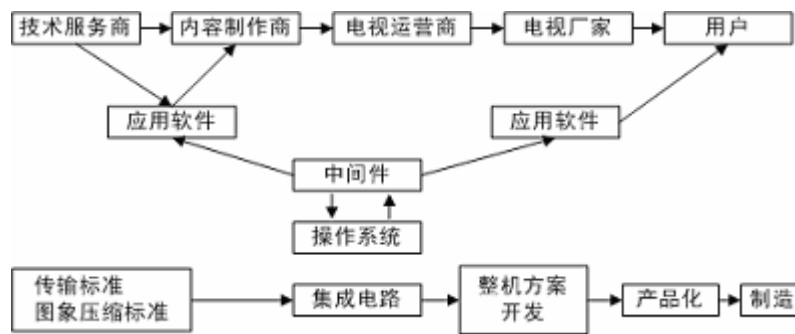


图1.1 数字电视产业网结构图

数字电视服务的最大特点是，它除了支持传统的音频、视频业务以外，还能带来电视增值业务，而这些增值业务大都是建立在基于数字电视的软件产品上的。随着数字电视市场的不断发展，数字电视运营商在寻求一种高效可靠的软件平台，数字电视制造商面对着多种系统实现方案 and 市场需求，其软件开发占技术总投入的比重日益加大，从而产生了对与平台无关而又能够支持业务扩展的开放系统的需求。数字电视中间件技术就是在这种应用需求下产生的。它定义了一组支持数字电视应用功能的标准程序接口，使应用程序与具体的数字电视硬件平台无关。通过中间件的平滑嫁接，同一应用软件可以在不同的操作系统上使用。

1.3 课题来源及研究意义

本课题来源于国家863重大软件专项项目——“基于中间件技术的因特网嵌入式操作系统及跨操作系统的中间件运行平台”。本课题的主要目标是研究和开发基于嵌入式操作系统的数模一体机的开发，本文是在该课题的基础上进行的。首先研究了嵌入式操作系统的关键技术；然后分别分析了数模一体机的模拟模块和数字模块的软件架构和实现，以及数模通讯的设计和实现，最后给出了该系统的前端设计和实现。

虽然国内企业对数字电视的研发已经起步，但由于我们普遍采用外国厂家提供的与芯片捆绑销售的操作系统，受国外操作系统技术封闭的影响，国内厂家将很难做到自主开发产品，必须高度依赖国外厂家的技术支持。这样就必然产生开发环境、开发手段落后，手工作坊式生产应用软件甚至是电视节目的弊端。为了更好的保护我国的数字电视产业，一方面要采用具有我国自主知识产权的操作系统，另一方面还必须加快对数字电视软件平台相关技术的研究和开发。因此，本文对数字电视软件开发平台进行的研究，不仅具有重要的理论意义，还具有广阔的应用前景。

1.4 论文结构

本文的内容共分为七章：

第一章简要介绍了数字电视的发展概况、数字电视与软件中间件的关系以及课题来源及研究意义。

第二章重点介绍了“和欣”嵌入式操作系统及其构件技术。

第三章主要介绍了数模一体机的模拟模块的硬件平台和软件系统结构和实现。

第四章主要论述了数模一体机的硬件组成和关键技术，ATSC标准，软件总体架构，并详细介绍了中间件的设计和实现。

第五章具体论述了数模通讯的详细设计和实现。

第六章论述了基于数字电视中间件平台的可视化开发工具DTV Visual Developer的体系结构，并提出了一种数字电视软件开发的新模式。

第七章对目前的研究工作进行了总结，并提出了未来可能的研究方向和关键性问题。

第 2 章 “和欣” 嵌入式操作系统及其构件技术

2.1 “和欣” 操作系统概述

2.1.1 “和欣” 操作系统简介

“和欣” 是 32 位嵌入式操作系统。该操作系统可以从多个侧面进行描述：

32 位嵌入式操作系统。操作系统基于微内核，具有多进程、多线程、抢占式、基于线程的多优先级任务调度等特性。提供 FAT 兼容的文件系统，可以从软盘、硬盘、FLASH ROM 启动，也可以通过网络启动。和欣操作系统体积小，速度快，适合网络时代的绝大部分嵌入式信息设备。

完全面向构件技术的操作系统。操作系统提供的功能模块全部基于 ezCOM 构件技术，因此是可拆卸的构件，应用系统可以按照需要剪裁组装，或在运行时动态加载必要的构件。

从传统的操作系统体系结构的角度来看，和欣操作系统可以看成是由微内核、构件支持模块、系统服务器组成的。

- 微内核：主要可分为 4 大部分：硬件抽象层（对硬件的抽象描述，为该层之上的软件模块提供统一的接口）；内存管理（规范化的内存管理接口，虚拟内存管理）；任务管理（进程管理的基本支持，支持多进程，多线程）；进程间通信（实现进程间通信的机制，是构件技术的基础设施）。
- 构件支持模块：提供了对 ezCOM 构件的支持，实现了构件运行环境。构件支持模块并不是独立于微内核单独存在的，微内核中的进程间通讯部分为其提供了必要的支持功能。
- 系统服务器：在微内核体系结构的操作系统中，文件系统、设备驱动、网络支持等系统服务是由系统服务器提供的。在和欣操作系统中，系统服务器都是以动态链接库的形式存在。

2.1.2 和欣灵活内核简介

和欣操作系统的实现全面贯穿了ezCOM思想，ezCOM构件可以运行于不同地址空间或不同的运行环境。可以把操作系统的内核地址区看成是一段特殊的地址空间，用户可以根据运行时的需求，自主选择将操作系统的某些系统服务构件、文件系统、图形系统、设备驱动构件等运行于内核地址空间或用户地址空间。与传统的操作系统的“大内核”、“微内核”体系结构相比，和欣操作系统内核里提供的系统服务，完全可以由用户依据系统自身的需求动态决定。因此称和欣操作系统内核为“灵活内核”（Agile Kernel）。

和欣灵活内核的体系结构，利用构件和中间件技术解决了长期以来困扰操作系统体系结构设计者的大内核和微内核在性能、效率与稳定性、安全性之间不能两全其美的矛盾。

下图来表示和欣灵活内核及其与系统构件和应用构件的关系：

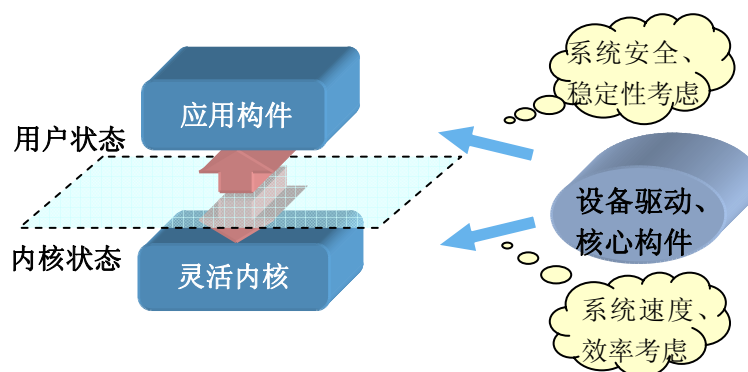


图 2.1 “和欣”灵活内核与系统构件和应用构件的关系

2.1.3 和欣操作系统提供的功能

从应用编程的角度看，和欣操作系统提供了一套完整的、符合ezCOM规范的系统服务构件及系统API，为在各种嵌入式设备的硬件平台上运行ezCOM二进制构件提供了统一的编程环境。

和欣操作系统还提供了一组动态链接构件库，这些构件库通常是开发嵌入式应用系统时不可缺少的：

- 图形系统构件库（方便开发图形用户操作界面）；
- 设备驱动构件库（各种输入输出设备的驱动）；

- 文件系统构件库（FAT 兼容，包括对 FLASH 等的支持）；
- 网络系统构件库（TCP/IP 等网络协议支持）。

系统提供的构件库，以及用户开发的应用程序构件都是通过系统接口与内核交互，从这个意义上说，他们处于同样的地位。用户可以开发性能更好或者更符合需求的文件系统、网络系统等构件库，替换这些构件库，也可以开发并建立自己的应用程序构件库。这就是基于构件技术操作系统的优势之一。

此外，为了方便用户编程，在和欣SDK中还提供了以下函数库：

- 与微软 Win32 API 兼容的应用程序编程接口(zeew32 API)；
- 标准 C 运行库（libc）；
- 和欣提供的工具类函数（zeeutil）。

对程序员来说，和欣操作系统提供的用户编程接口与上一节中介绍的和欣构件运行平台完全一样。所以，在相互兼容的硬件平台上，不管运行的是和欣操作系统还是Windows操作系统，应用程序可以不加区分地在其上运行。

和欣操作系统实现并支持系统构件及用户构件相互调用的机制，为ezCOM构件提供了运行环境。关于ezCOM构件的运行环境，其描述与“和欣构件运行平台”是一样的，在此从简。因此，可以把和欣操作系统看成是直接运行在硬件平台上的“和欣构件运行平台”。

可以用下图来表示和欣操作系统及其主要构成：

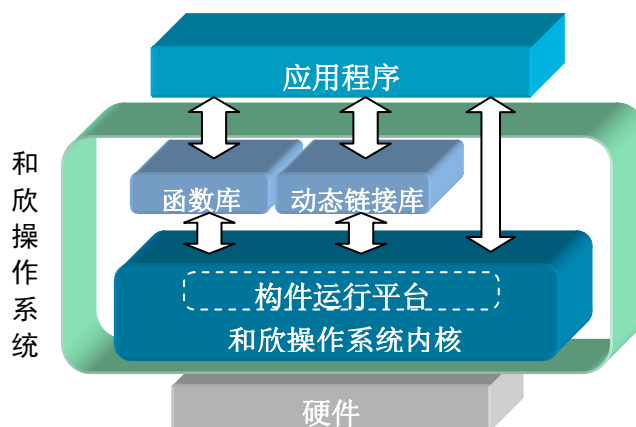


图 2.2 “和欣”操作系统的系统结构图

2.1.4 和欣操作系统的应用软件开发

和欣SDK提供了应用软件的开发环境和工具。开发“和欣”应用软件的开发环境如下图所示：

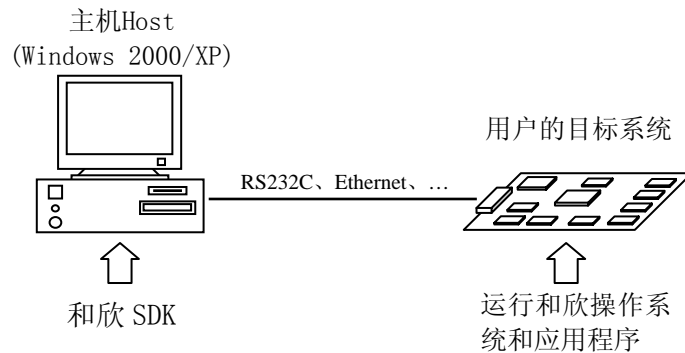


图 2.3 “和欣”应用软件的开发环境

开发“和欣”应用软件的过程，如下图所示：

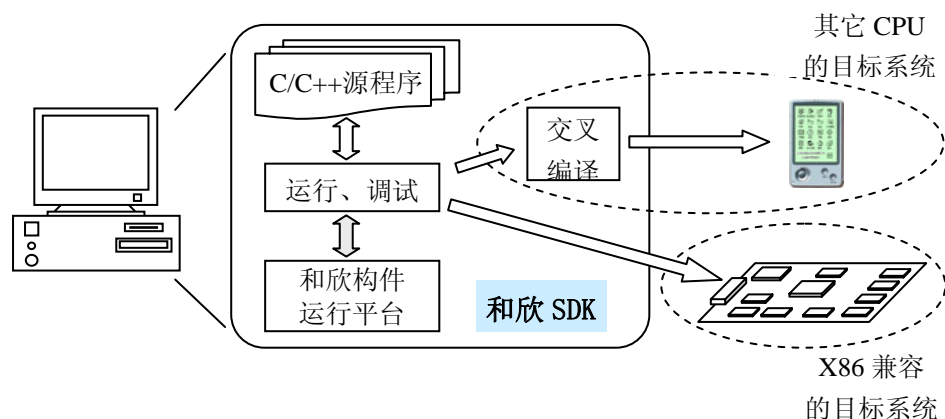


图 2.4 “和欣”应用软件的开发过程

2.1.5 和欣操作系统的优势

和欣操作系统的最大特点就是：

- 全面面向构件技术，在操作系统层提供了对构件运行环境的支持；
- 用构件技术实现了“灵活”的操作系统。

这是和欣操作系统区别于其它商用嵌入式操作系统产品的最大优势。

在新一代因特网应用中，越来越多的嵌入式产品需要支持网络服务，而网络服务的提供一定是基于构件的。在这种应用中，用户通过网络获得服务程序，这个程序一定是带有自描述信息的构件，本地系统能够为这个程序建立运行环境，自动加载运行。这是新一代因特网应用的需要，是必然的发展方向。和欣

操作系统就是应这种需要而开发，率先在面向嵌入式系统应用的操作系统中实现了面向构件的技术。

因此，构件化的和欣操作系统可以为嵌入式系统开发带来以下好处：

- 在嵌入式软件开发领域，导入先进的工程化软件开发技术。嵌入式软件一般用汇编语言、C 语言，在少数系统中已经支持了 C++ 开发，但是由于还没有一个嵌入式操作系统能够提供构件化的运行环境，可以说，嵌入式软件开发还是停留在手工作坊式的开发方式上。和欣操作系统使得嵌入式应用的软件开发能够实现工程化、工厂化生产。
- 可以动态加载构件。动态加载构件是因特网时代嵌入式系统的必要功能。新一代 PDA 和移动电话等移动电子产品，不能再像以前那样由厂家将所有的功能都做好后固定在产品里，而要允许用户从网上获得自己感兴趣的程序。
- 随时和动态地实现软件升级。动态加载构件的功能，同样可以用于产品的软件升级，开发商不必为了添加了部分功能而向用户重新发布整套软件，只需要升级个别构件。
- 灵活的模块化结构，便于移植和剪裁。易于定制成针对不同硬件配置的紧凑高效的嵌入式操作系统。添加或删除某些功能模块也非常简单。
- 嵌入式软件开发商容易建立自己的构件库。在不同开发阶段开发的软件构件，其成果很容易被以后的开发所共享，保护软件开发投资。软件复用使得系列产品的开发更加容易，缩短新产品开发周期。
- 容易共享第三方软件开发商的成果。面向行业的构件库的建设，社会软件的丰富，使得设备厂家不必亲自开发所有的软件，可以充分利用现有的软件资源，充分发挥自己的专长为自己的产品增色。
- 跨操作系统平台兼容，降低软件移植的风险。在和欣开发环境上开发的软件所具有的跨平台特性，使得用户可以将同样的可执行文件不加修改地运行在和欣操作系统（嵌入式设备）与 Windows 2000/XP（PC）上。特别是对于需要将 Windows 上的软件移到嵌入式系统以降低产品成本的用户，这一特点不仅可以大大节约软件移植的费用，还可以避免因移植而带来的其它隐患。
- 功能完备的开发环境和方便的开发工具，帮助嵌入式开发人员学习和掌握先进的构件化软件编程技术，提高软件开发效率。应用软件可以

在开发环境下开发调试，与硬件研制工作同时进行，缩短产品研制周期。

2.2 和欣构件运行平台

2.2.1 和欣构件运行平台简介

和欣构件运行平台提供了一套符合ezCOM（详见2.3节）规范的系统服务构件及支持构件相关编程的API函数，实现并支持系统构件及用户构件相互调用的机制，为ezCOM构件提供了编程运行环境。和欣运行平台有在不同操作系统上的实现，符合ezCOM编程规范的应用程序通过该平台实现二进制级跨操作系统平台兼容。

在和欣操作系统中，和欣构件运行平台与“和欣灵活内核”共同构成了完整的操作系统。

在Windows 2000、WinCE、Linux 等其它操作系统上，和欣构件运行平台屏蔽了底层传统操作系统的具体特征，实现了一个构件化的虚拟操作系统。在和欣构件运行平台上开发的应用程序，可以不经修改、不损失太多效率、以相同的二进制代码形式，运行于传统操作系统之上。

下图显示了和欣构件运行平台在Windows 2000/XP、和欣操作系统中的位置。

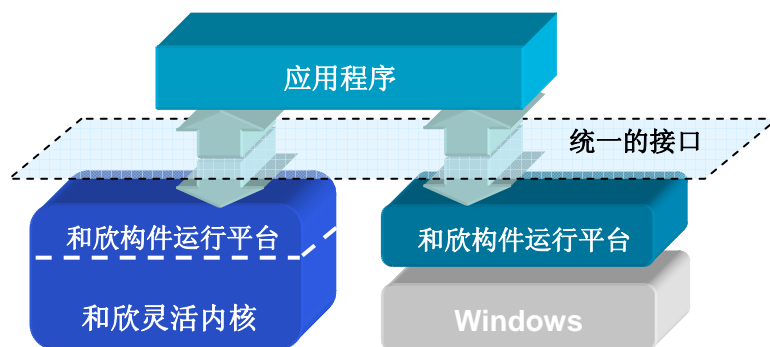


图 2.5 “和欣”构件运行平台与操作系统的关系

2.2.2 和欣构件运行平台的功能

从和欣构件运行平台的定义，知道该平台为ezCOM提供了运行环境。从这

个意义上，这里说的ezCOM技术也可以理解为在运行环境中对ezCOM规范提供支持的程序集合。

从编程的角度看，和欣构件运行平台提供了一套系统服务构件及系统API（应用程序编程接口），这些是在该平台上开发应用程序的基础。

和欣操作系统提供的其它构件库也是通过这些系统服务构件及系统API实现的。系统提供的这些构件库为应用编程开发提供了方便：

- 图形系统构件库；
- 设备驱动构件库；
- 文件系统构件库；
- 网络系统构件库。

从和欣构件运行平台来看，这些构件和应用程序的构件是处于同样的地位。用户可以开发性能更好或者更符合需求的文件系统、网络系统等构件库，替换这些构件库，也可以开发并建立自己的应用程序构件库。

右图显示出和欣构件运行平台的功能及其与构件库、应用程序的关系。

从支持ezCOM构件的运行环境的角度看，和欣构件运行平台提供了以下功能：

- 根据二进制构件的自描述信息自动生成构件的运行环境，动态加载构件；
- 提供构件之间的自动通信机制，构件间通信可以跨进程甚至跨网络；
- 构件的运行状态监控，错误报告等；
- 提供可干预构件运行状态的机制，如负载均衡、线程同步、访问顺序控制、安全（容错）性控制、软件使用权的控制等；
- 构件的生命周期管理，如进程延续（Persistence）控制、事务元（Transaction）控制等；

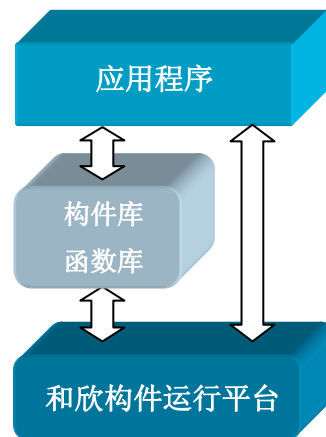


图 2.6 和欣构件运行平台功能图

总之，构件运行平台为ezCOM构件提供了对程序员完全透明的运行环境，构件可以运行在不同地址空间，不同环境，甚至跨网络。构件运行平台自动为构件运行提供支持，配置必要的网络协议、针对不同的输入输出设备的协议。程序员不必过多地去关心诸如网络协议转换及构件运行控制等与其它构件互操

作时的协调问题，只需专注于自己需要解决的程序算法的实现。从而可以从繁杂庞大的应用环境体系中解放出来，大大提高编程的效率。

和欣构件运行平台直接运行二进制构件，而不是像JAVA和.NET那样通过虚拟机在运行程序时解释执行中间代码。因此，与其它面向构件编程的系统相比，具有资源消耗小，运行效率高的优点。

2.2.3 和欣构件运行平台的技术优势

作为总结，和欣构件运行平台的主要技术优势列举如下：

- 开发跨操作系统平台的应用软件；
- 对程序员透明的 ezCOM 构件运行环境，提高编程的效率；
- 直接运行二进制构件代码，实现软件运行的高效率；
- 构件可替换，用户可建立自己的构件库。

需要说明的是，和欣构件运行平台实现的应用软件跨操作系统平台兼容是以具有同样的硬件体系结构为前提的。目前，和欣构件运行平台还不能支持不同指令系统的CPU间的“跨平台”兼容。

2.2.4 利用和欣构件运行平台编程

对程序员来说，编写运行于和欣构件运行平台上的程序，运用ezCOM技术和跨平台技术的具体方法，体现在对构件库的接口方法、通用API函数的调用上。应用程序运行所需要的动态链接库，则是在程序运行时由和欣构件运行平台自动加载的。

下图简明地表示了编写运行于和欣构件运行平台上的应用程序所需的相关要素之间的关系示意。

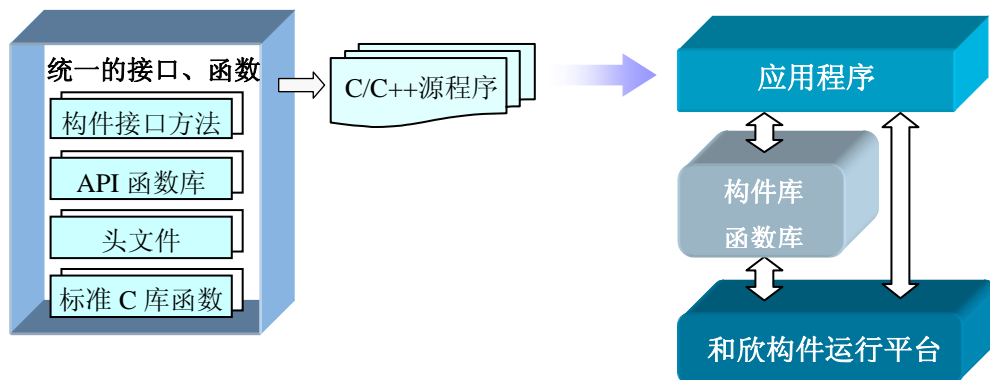


图 2.7 如何编写基于“和欣”构件运行平台的应用程序

2.3 ezCOM 构件技术

2.3.1 ezCOM 技术的由来

80年代以来，目标指向型软件编程技术有了很大的发展，为大规模的软件协同开发以及软件标准化、软件共享、软件运行安全机制等提供了理论基础。其发展可以大致分为以下几个阶段。

(1) 面向对象编程

通过对软件模块的封装，使其相对独立，从而使复杂的问题简单化。面向对象编程强调的是对象的封装，但模块（对象）之间的关系在编译的时候被固定，模块之间的关系是静态的，在程序运行时不可改变模块之间的关系，就是说在运行时不能换用零件。其代表是C++语言所代表的面向对象编程。

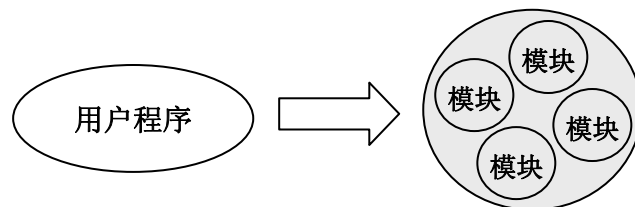


图 2.8 面向对象的编程模型

(2) 面向构件编程

为了解决不同软件开发商提供的构件模块（软件对象）可以相互操作使用，构件之间的连接和调用要通过标准的协议来完成。构件化编程模型强调协议标准，需要提供各厂商都能遵守的协议体系。就像公制螺丝的标准一样，所有符合标准的螺丝和螺母都可以相互装配。构件化编程模型建立在面向对象技术的基础之上，是完全面向对象的，提供了动态构造部件模块（运行中可以构造部件）的机制。构件在运行时动态装入，是可换的。其代表是COM技术。

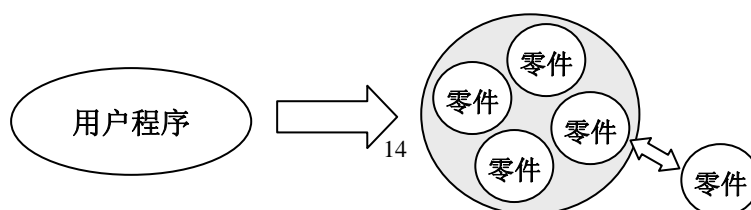


图 2.9 面向构件的编程模型

(3) 面向中间件编程

由于因特网的普及，构件可来自于网络，系统要解决自动下载，安全等问题。因此，系统中需要根据构件的自描述信息自动生成构件的运行环境，生成代理构件即中间件，通过系统自动生成的中间件对构件的运行状态进行干预或控制，或自动提供针对不同网络协议、输入输出设备的服务（即运行环境）。中间件编程更加强调构件的自描述和构件运行环境的透明性，是网络时代编程的重要技术。其代表是ezCOM、JAVA和.NET（C#语言）。

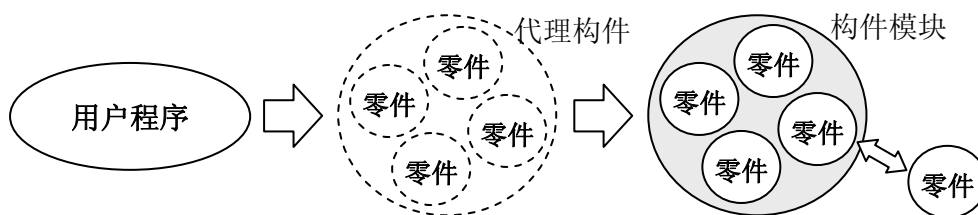


图 2.10 面向中间件的编程模型

在这样的发展过程中，人们逐步深化了对大规模软件开发所需的科学模型、网络环境下软件运行必要机制的理解，使软件技术达到了更高的境界，实现了：

- 构件的相互操作性。不同软件开发商开发的具有独特功能的构件，可以确保与其他人开发的构件实现互操作。
- 软件升级的独立性。实现在对某一个构件进行升级时不会影响到系统中的其它构件。
- 编程语言的独立性。不同的编程语言实现的构件之间可以实现互操作。
- 构件运行环境的透明性。提供一个简单、统一的编程模型，使得构件可以在进程内、跨进程甚至于跨网络运行。同时提供系统运行的安全、保护机制。

ezCOM 技术就是在总结面向对象编程、面向构件编程技术的发展历史和经验，为更好地支持面向以 Web Service（WEB 服务）为代表的下一代网络

应用软件开发而发明的。**ezCOM** 很大程度地借鉴了 **COM** 技术,保持了和 **COM** 的兼容性,同时对 **COM** 进行了重要的扩展。

2.3.2 ezCOM 构件技术概要

ezCOM构件技术是面向构件编程的编程模型,它规定了一组构件间相互调用的标准,使得二进制构件能够自描述,能够在运行时动态链接。

ezCOM兼容微软的**COM**。但是和微软**COM**相比,**ezCOM**删除了**COM**中过时的约定,禁止用户定义**COM**的非自描述接口;完备了构件及其接口的自描述功能,实现了对**COM**的扩展;对**COM**的用户界面进行了简化包装,易学易用。

从上面的定义中,我们可以说**ezCOM**是微软**COM**的一个子集。**ezCOM**很大程度地借鉴了**COM**技术,保持了和**COM**的兼容性,同时对**COM**进行了重要的扩展。在和欣**SDK**工具的支持下,使得高深难懂的构件编程技术很容易被**C/C++**程序员理解并掌握。**ezCOM**中的“ez”源自与英文单词“easy”,恰如其分地反映了这一特点。

ezCOM技术是在总结面向对象编程、面向构件编程技术的发展历史和经验,为更好地支持面向**Web Service** (**WEB**服务)的下一代网络应用软件开发而开发的。

ezCOM的编程思想是“和欣”技术的精髓,它贯穿于整个技术体系的实现中。

为了在资源有限的嵌入式系统中实现面向中间件编程技术,同时又能得到**C/C++**的运行效率,**ezCOM**没有使用**JAVA**和**.NET**的基于中间代码—虚拟机的机制,而是采用了用**C++**编程,用和欣**SDK**提供的工具直接生成运行于和欣构件运行平台的二进制代码的机制。用**C++**编程实现构件技术,使得更多的程序员能够充分运用自己熟悉的编程语言知识和开发经验,很容易掌握面向构件、中间件编程的技术。在不同操作系统上实现的和欣构件运行平台,使得**ezCOM**构件的二进制代码可以实现跨操作系统平台兼容。

2.3.3 ezCOM 技术的意义

对于面向**WEB**服务的应用软件开发,以及开发操作系统这样的大型系统软件而言,采用**ezCOM**构件技术具有以下意义:

- 不同软件开发商开发的具有独特功能的构件，可以确保与其他人开发的构件实现互操作。
- 实现在对某一个构件进行升级时不会影响到系统中的其它构件。
- 不同的编程语言实现的构件之间可以实现互操作。
- 提供一个简单、统一的编程模型，使得构件可以在进程内、跨进程甚至于跨网络运行。同时提供系统运行的安全、保护机制。
- ezCOM 构件与微软的 COM 构件二进制兼容，但是 ezCOM 的开发工具自动实现构件的封装，简化了构件编程的复杂性，有利于构件化编程技术的推广普及；
- ezCOM 构件技术是一个实现软件工厂化生产的先进技术，可以大大提升企业的软件开发技术水平，提高软件生产效率和软件产品质量；
- 软件工厂化生产需要有零件的标准，ezCOM 构件技术为建立软件标准提供了参考，有利于建立企业、行业的软件标准和构件库。

2.3.4 ezCOM 技术对软件工程的作用

大型软件的生产，已经不是传统的手工作坊式的开发所能够胜任的。所谓的软件工厂化生产，是软件工程研究几十年来追求的理想。实现这样的目标，需要有一个科学的编程模型，为所有参与项目开发的软件工程师开发的软件能够正确地对接运行提供技术上的保障。这些软件可以由一个公司的不同部门提供，也可以是不同企业，不同地点、不同时间生产的。COM技术在微软经过十多年的应用与提炼，被证明是行之有效的编程模型，基于COM技术，微软造就了Windows NT以及配套应用软件这样的大型软件产品。汲取了COM技术的精华，并对其进行了扩展和简易包装的ezCOM技术，将成为软件工厂化生产的利器。

ezCOM的重要特点就是：构件的简易化包装；构件的相互操作性；软件升级的独立性；编程语言的独立性；进程运行透明度。在实际的编程应用中，ezCOM技术的优势体现在以下方面：

（1）易学易用

基于COM的构件化编程技术是大型软件工程化开发的重要手段。但是微软COM的繁琐的构件描述体系令人望而生畏。ezCOM的开发环境和欣SDK提供了

结构简洁的构件描述语言和自动生成辅助工具等，使得C++程序员可以很快地掌握ezCOM编程技术。

(2) 可以动态加载构件

在网络时代，软件构件就相当于零件，零件可以随时装配。ezCOM技术实现了构件动态加载，使用户可以随时从网络得到最新功能的构件。

(3) 采用第三方软件丰富系统功能

ezCOM技术的软件互操作性，保证了系统开发人员可以利用第三方开发的，符合ezCOM规范的构件，共享软件资源，缩短产品开发周期。同时用户也可以通过动态加载第三方软件扩展系统的功能。

(4) 软件复用

软件复用是软件工程长期追求的目标，ezCOM技术提供了构件的标准，二进制构件可以被不同的应用程序使用，使软件构件真正能够成为“工业零件”。充分利用“久经考验”的软件零件，避免重复性开发，是提高软件生产效率和软件产品质量的关键。

(5) 系统升级

传统软件的系统升级是一个令软件系统管理员头痛的工程问题，一个大型软件系统常常是“牵一发而动全身”，单个功能的升级可能会导致整个系统需要重新调试。ezCOM技术的软件升级独立性，可以圆满地解决系统升级问题，个别构件的更新不会影响整个系统。

(6) 实现软件工厂化生产

上述几个特点，都是软件零件工厂化生产的必要条件。构件化软件设计思想规范了工程化、工厂化的软件设计方法，提供了明晰可靠的软件接口标准，使软件构件可以像工业零件一样生产制造，零件可用于各种不同的设备上。

(7) 提高系统的可靠性、容错性

由于构件运行环境可控制，可以避免因个别构件的崩溃而波及到整个系统，提高系统的可靠性。同时，系统可以自动重新启动运行中意外停止的构件，实现系统的容错。

(8) 有效地实现系统安全性

系统可根据构件的自描述信息自动生成代理构件，通过代理构件进行安全控制，可以有效地实现对不同来源的构件实行访问权限控制、监听、备份容错、通信加密、自动更换通信协议等等安全保护措施。

2.3.5 ezCOM 技术在“和欣”技术体系中的作用

ezCOM构件技术是“和欣”技术体系的精髓，它贯穿于整个技术体系的实现中。可以说“和欣”操作系统是基于ezCOM实现的，同时它也是全面面向ezCOM构件技术的。

- 构件化的操作系统内核，以及“灵活内核”体系结构；
- 构件化的图形系统、设备驱动、文件系统等操作系统的系统服务；
- 支持应用软件跨平台二进制兼容的“和欣构件运行平台”；
- 构件化的应用软件，支持WEB服务，支持动态加载。

在面向嵌入式系统应用的操作系统中全面导入构件技术，对于开发功能越来越复杂的应用系统有着重要的意义。功能丰富的嵌入式设备将更多地要和网络发生关系，要支持 WEB 服务，前一节中所介绍的软件工程的意义，对于嵌入式系统开发来说同样是重要的。

2.3.6 如何用 ezCOM 技术编程

ezCOM是一个面向构件的编程模型，也可以说是一种编程思想，它表现为一组编程规范，包括构件、类、对象、接口等定义与访问构件对象的规定。

ezCOM的实现，是由一个配套的开发环境和运行环境完成的。

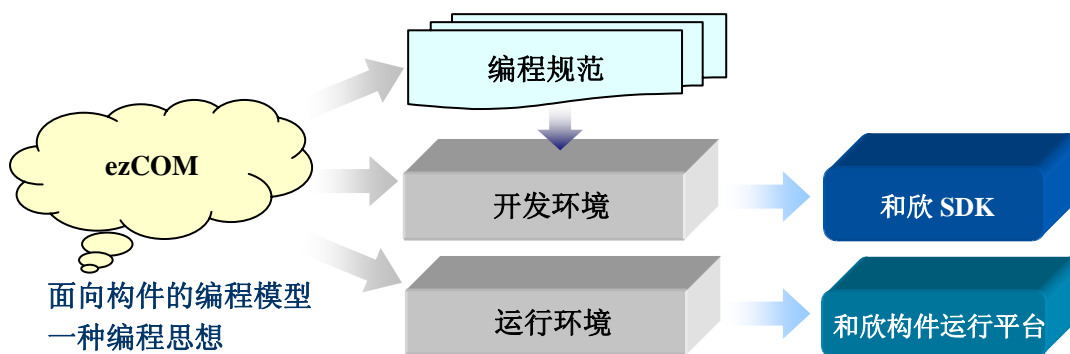


图 2.11 面向中间件的编程模型

第3章 模拟模块的系统结构和实现

3.1 ATV 硬件平台结构

ATV采用了FLI8125+2300的硬件平台。FLI8125是Genesis Microchip公司推出的应用于CRT电视的控制器，其内核采用的是8086，具有如下的特点：

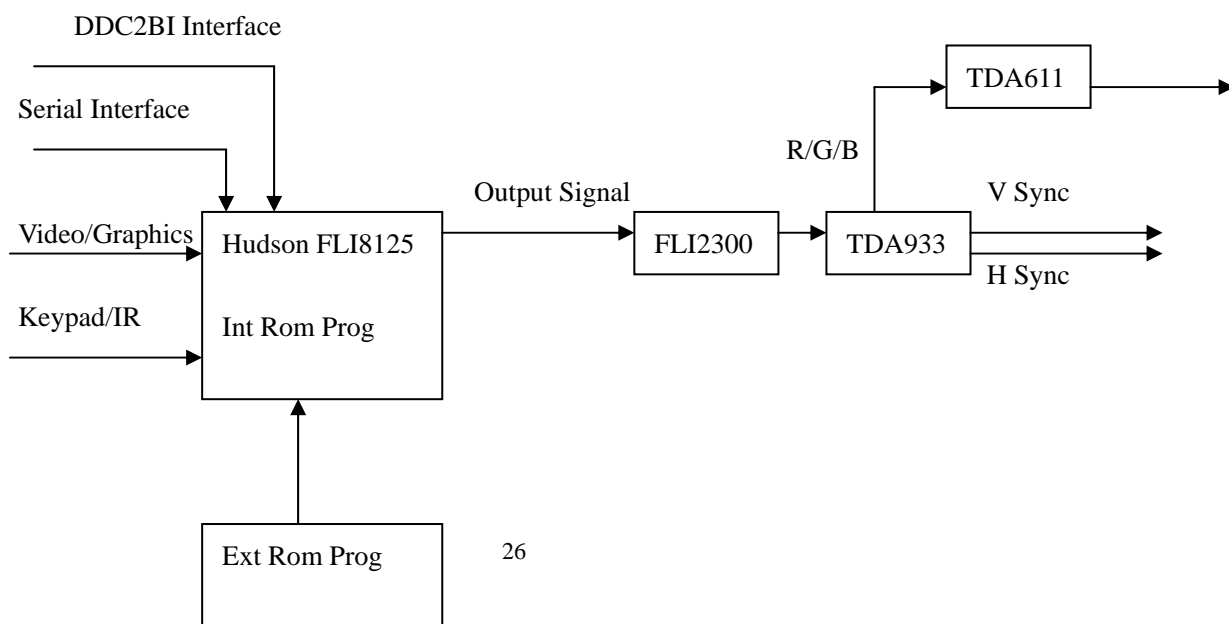
(1) 它配备了一个NTSC/PAL视频输入端口及完整的AD转换器（带有PLL和LVDS传送装置）。

(2) 除了采用了Genesis Microchip公司拥有专利的图象处理先进技术（图象格式转化）外，gm5221系列产品同时具备了色彩及对比度可选择（ACC）和色彩灵活管理（ACM-II）性能，可以改善对比度及增强图象色彩度。

(3) gm5221系列支持PC机频率输入（VGA、UXGA）和标准画质及高画质图象输入格式。图象最大显示分辨率可达1280x1024。

2300也是Genesis Microchip公司推出的一款芯片，用于进行电视信号隔行的处理。

硬件系统设计图：



3.2 ATV 软件系统设计

3.2.1 软件的总体架构设计

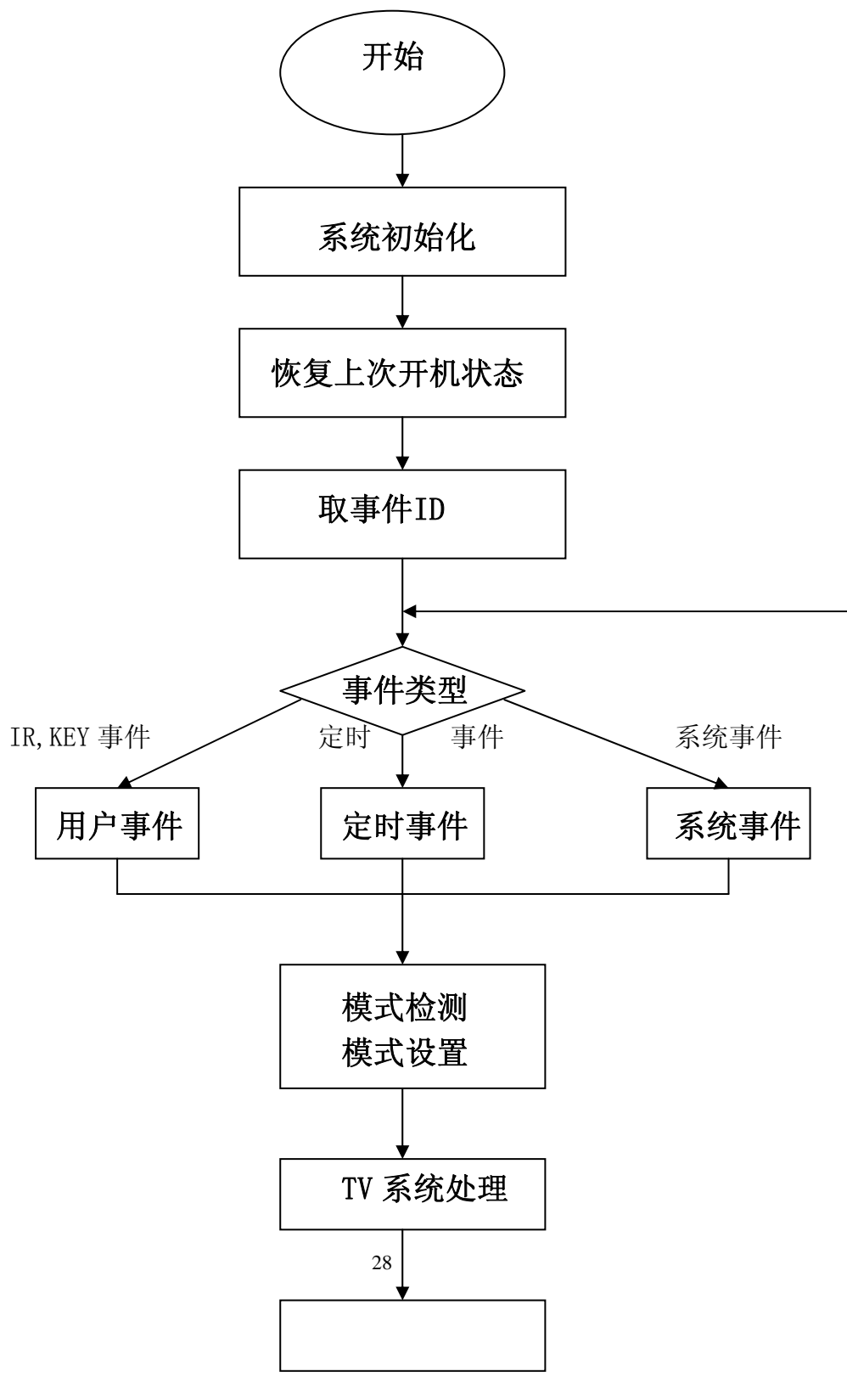
为了软件模块之间的松散耦合，该系统从功能上分为五层：

- (1) Application Layer
- (2) API Interface Layer
- (3) API Driver Layer
- (4) Device Driver Layer
- (5) Hardware Device Layer

每一层所实现的功能都提供给上层调用，将下面各层提供的功能给屏蔽掉。这样的分层结构设计有利于系统的更改和维护，可读性也更高了。

这里的Application Layer处于最上面一层，它也是由几个模块组成：System Initial, Event Handler, Mode Handler, OSD Handler等，这一层视具体的应用而改变，可以增加任意功能模块。

Application Layer中主循环流程图



OSD 更新

|

3.2.2 事件的处理

在ATV系统中事件被划分为三类：用户事件，定时事件，系统事件。

用户事件包括了用户所有的按键所触发的事件，如上下左右键，数字键，快捷键等等；定时事件指得是系统中与时间有关的事件，如睡眠时间，预约时间，屏显时间等等；系统事件指系统提示的一些事件，如自动搜台完毕，切换频道完毕等。

事件的定义如下：

```
typedef enum rsEventsEnum
{
    /**PRESS KEY EVENT**/
    OE_NONE,
    OE_KEY_START,
    OE_NUM_0 = OE_KEY_START,
    OE_NUM_1,
    OE_NUM_2,
    OE_NUM_3,
    ...
    ...
    OE_NUM_SET,
    OE_DIAPLAY,
    OE_RECALL,
    OE_SWAP,
    OE_POWER,
    OE_HOLD,
    OE_SOURCE,
```

```

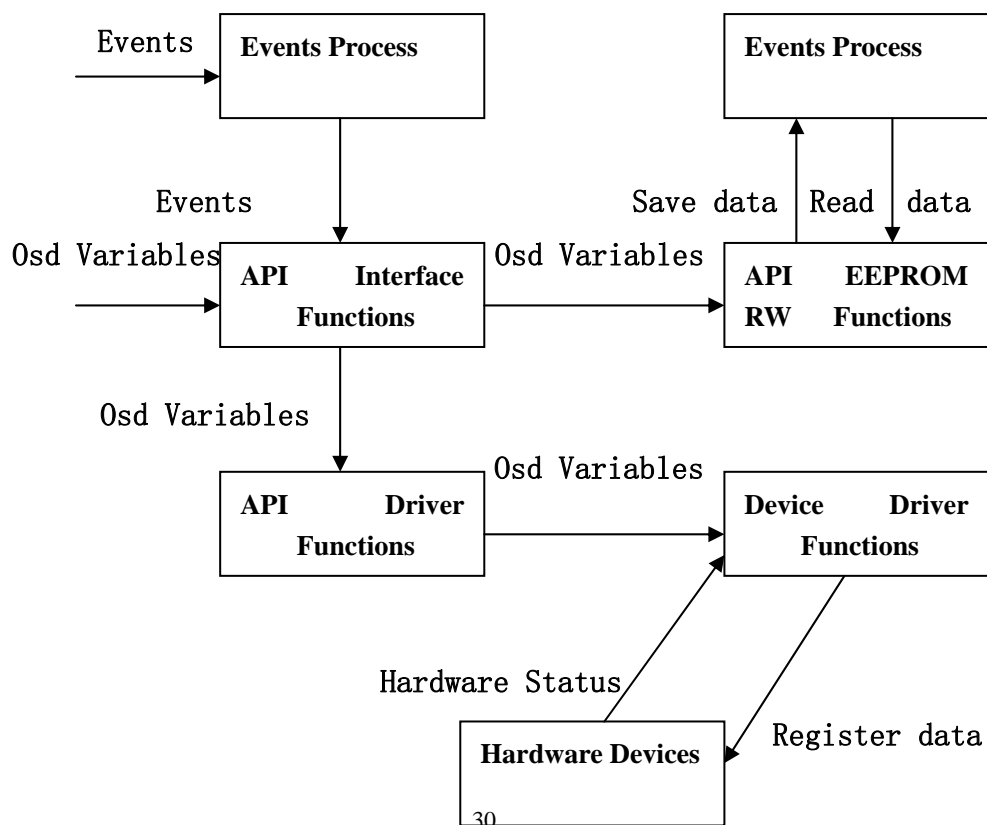
OE_UP,
OE_DOWN,
...
...

/*****Timer Events*****/
OE_RESERVE_TIMER,
OE_SLEEP_TIMER,
...
...

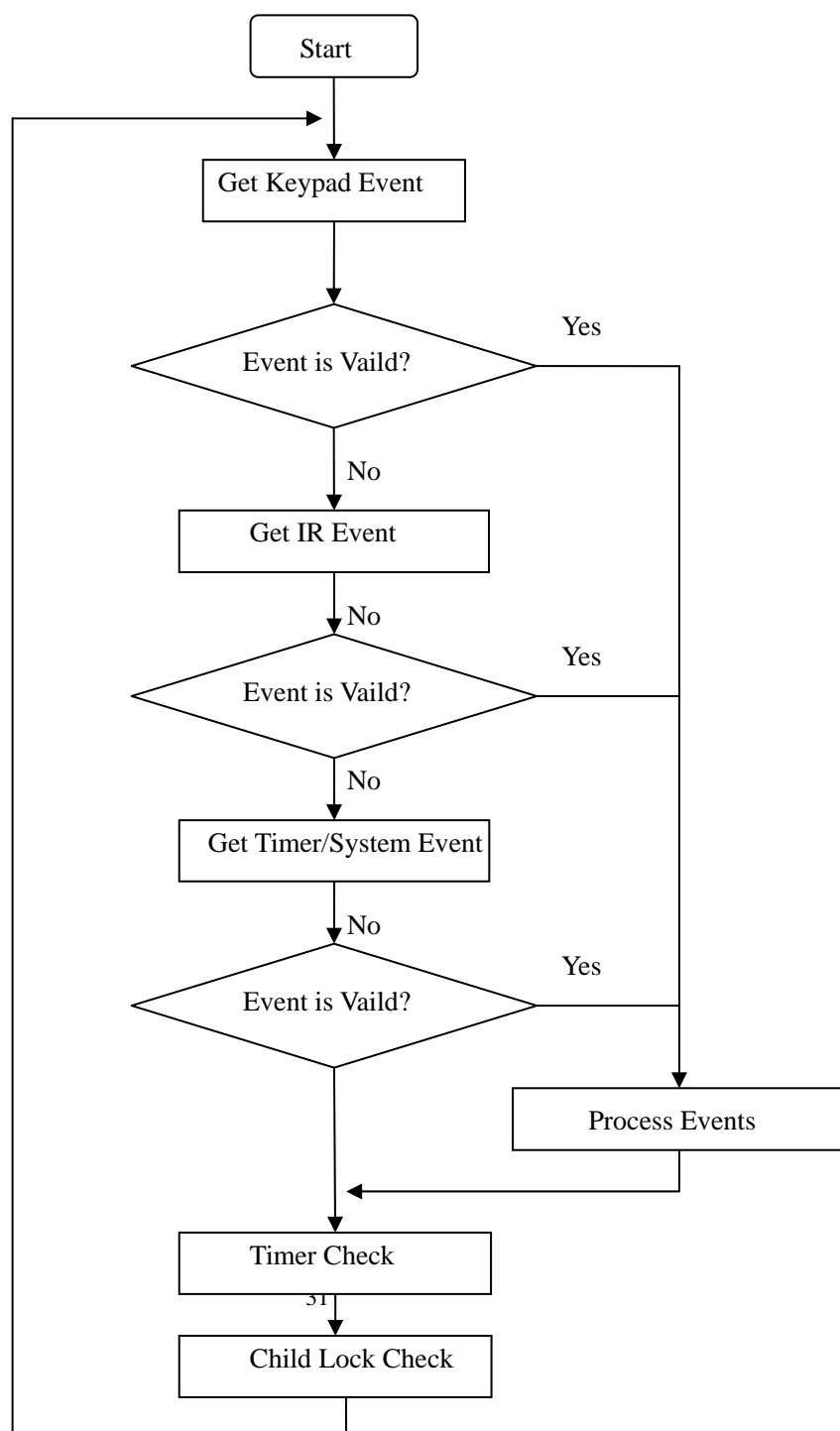
/*****System Events*****/
OE_AUTO_SEARCH_FINISHED,
OE_CHANGE_CHANNEL_FINISHED,
...
...
} rsTvProjEvents;

```

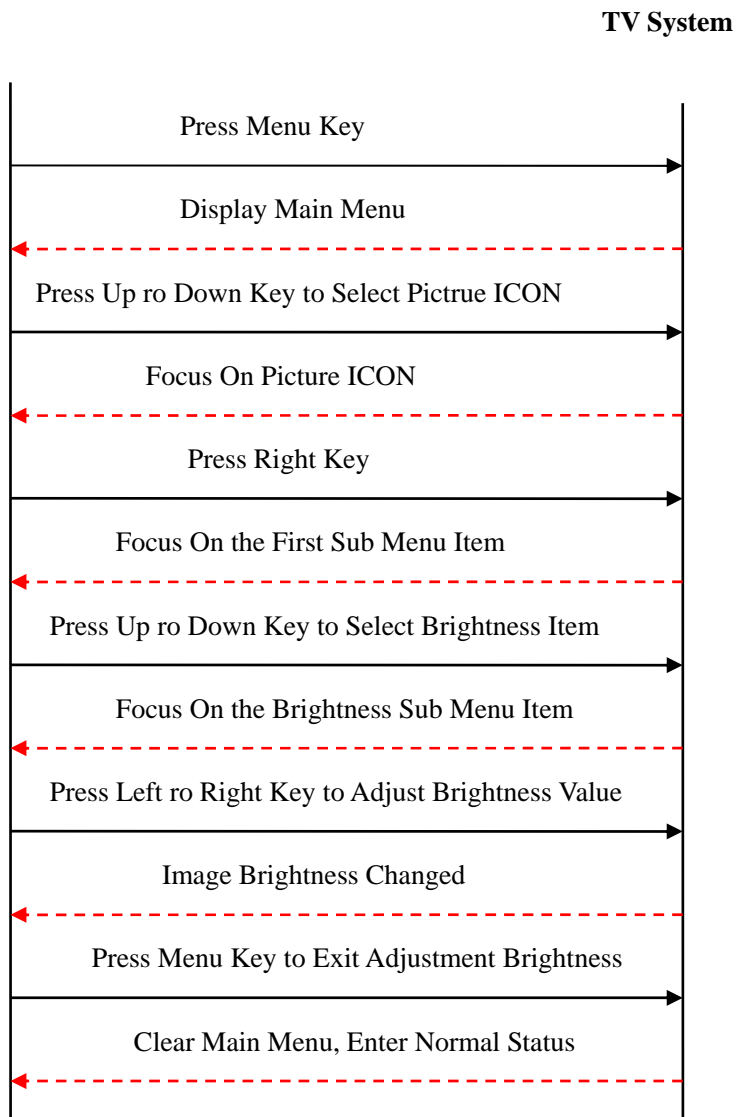
事件处理的数据流:

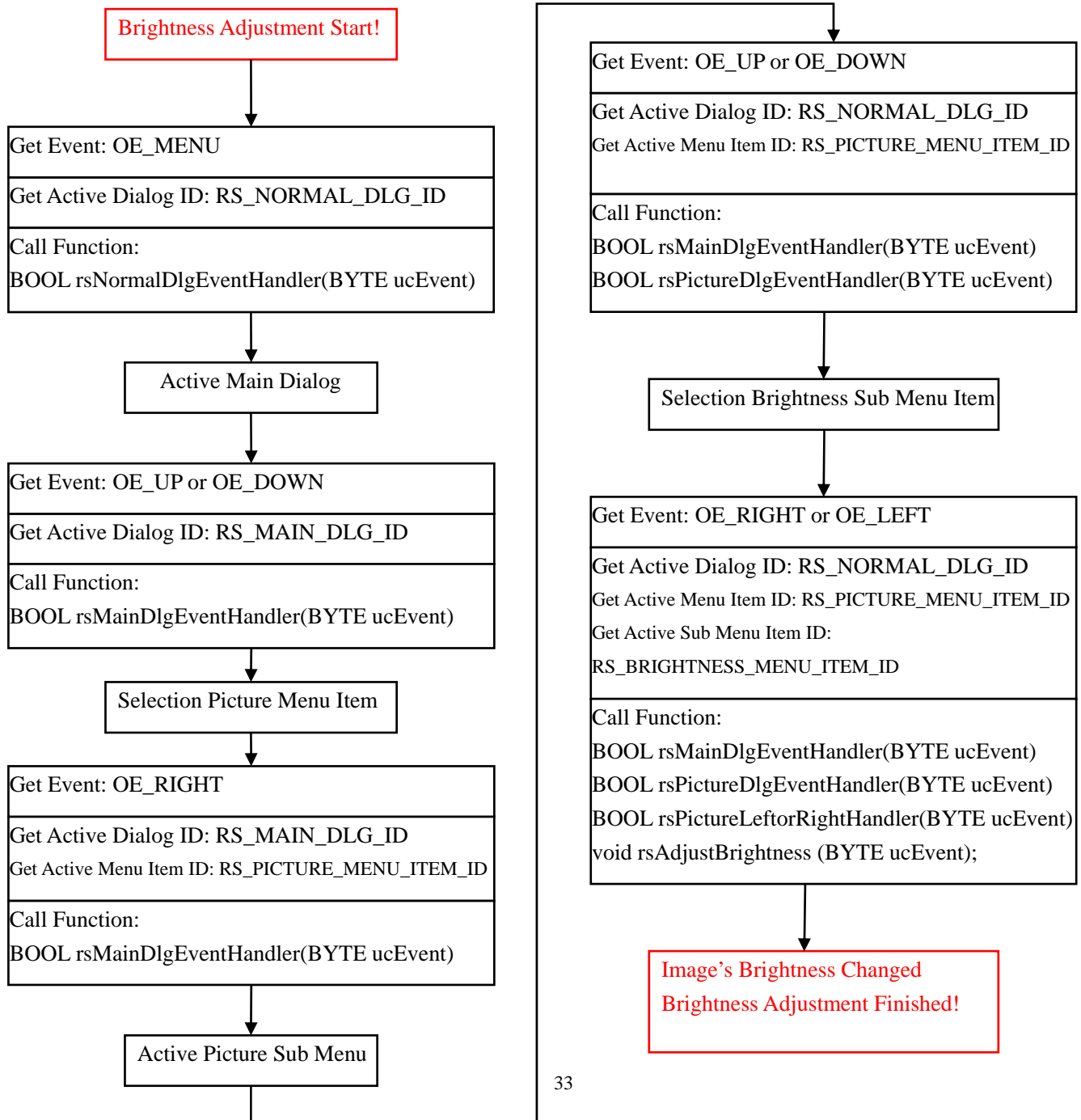


事件处理的逻辑流程图：



下面以亮度调节为例看看系统如何响应和处理事件的。

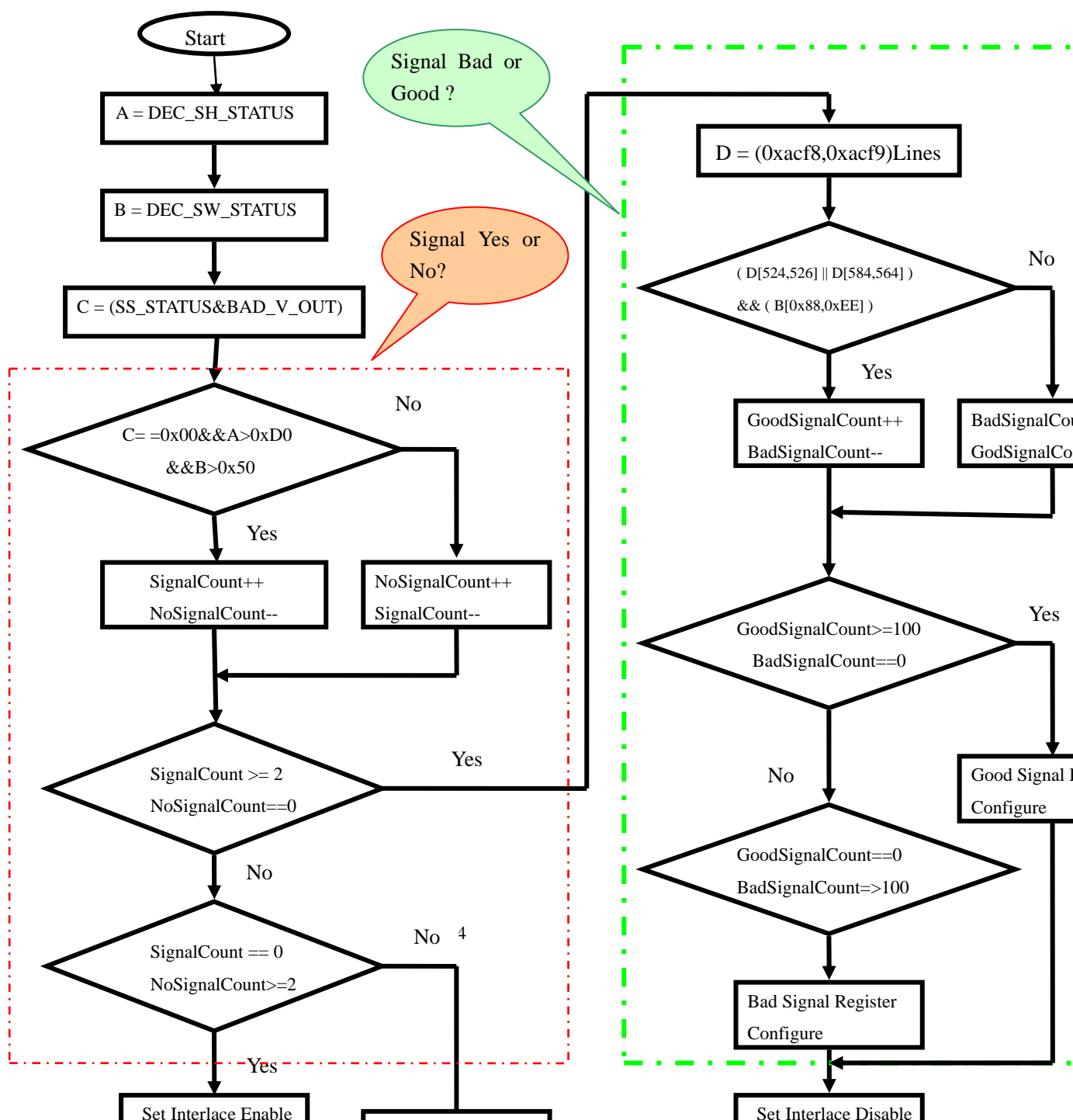




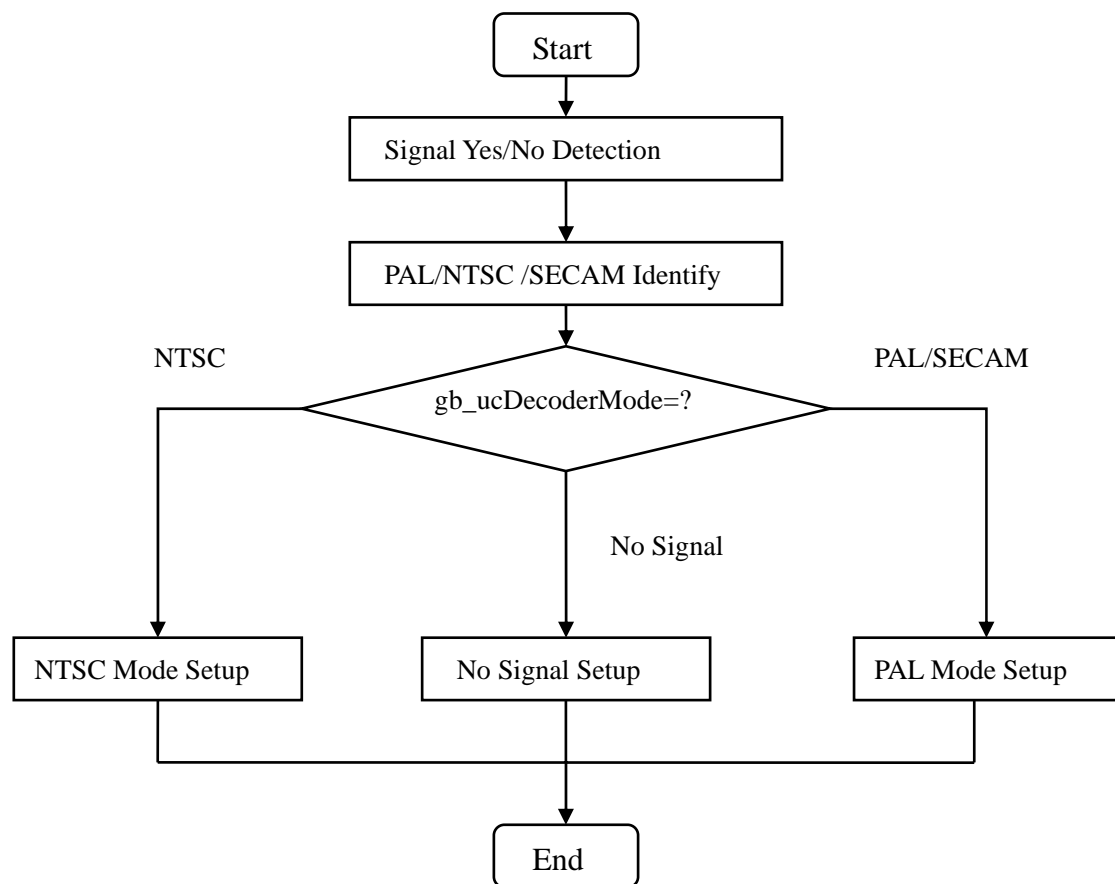
3.2.3 模式的处理

在FLI8125系统中模式的处理有两部分：1. TV、AV信号的处理；2. PC/HDTV信号的处理。

Decoder Signal:

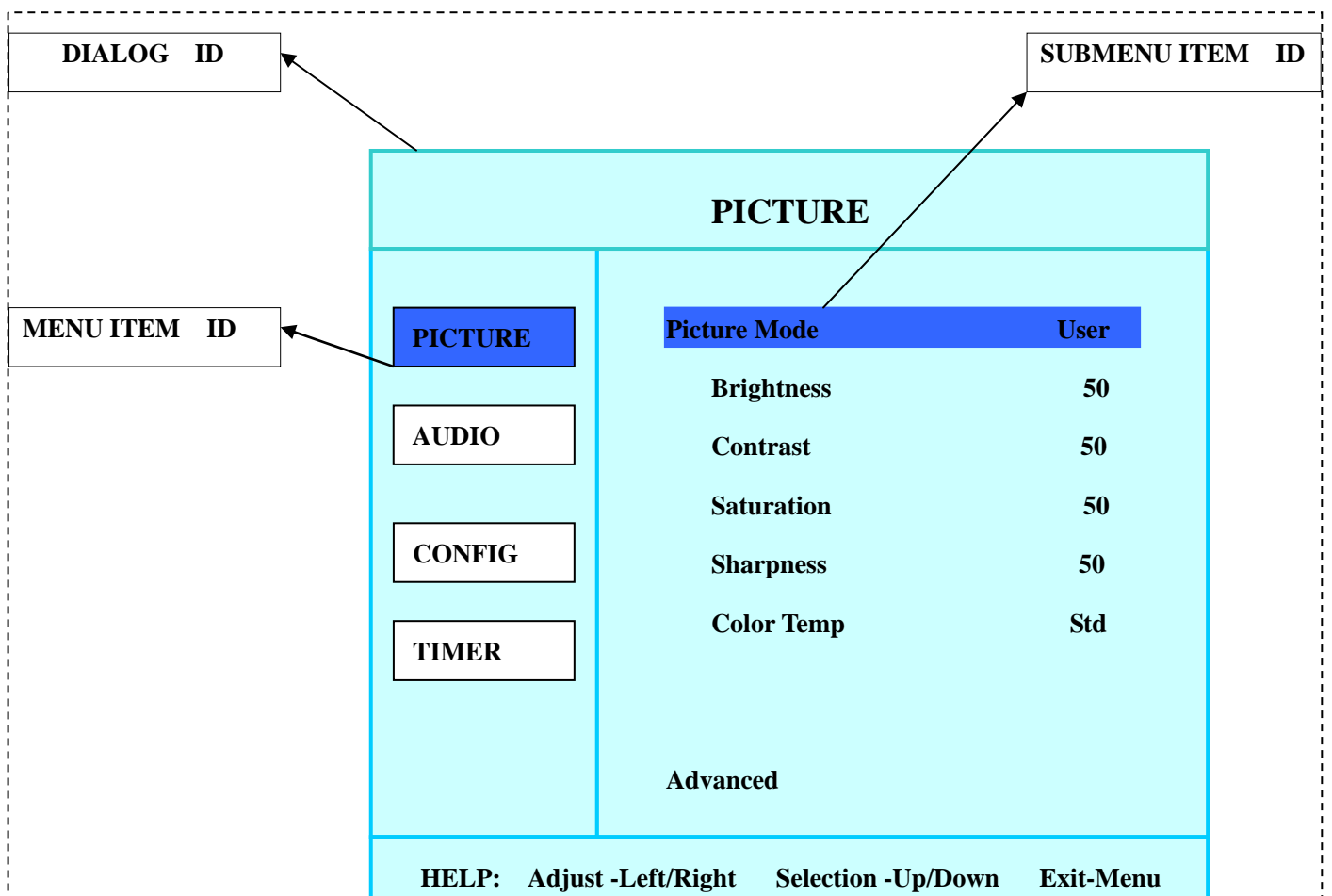


Decoder Mode Set Up

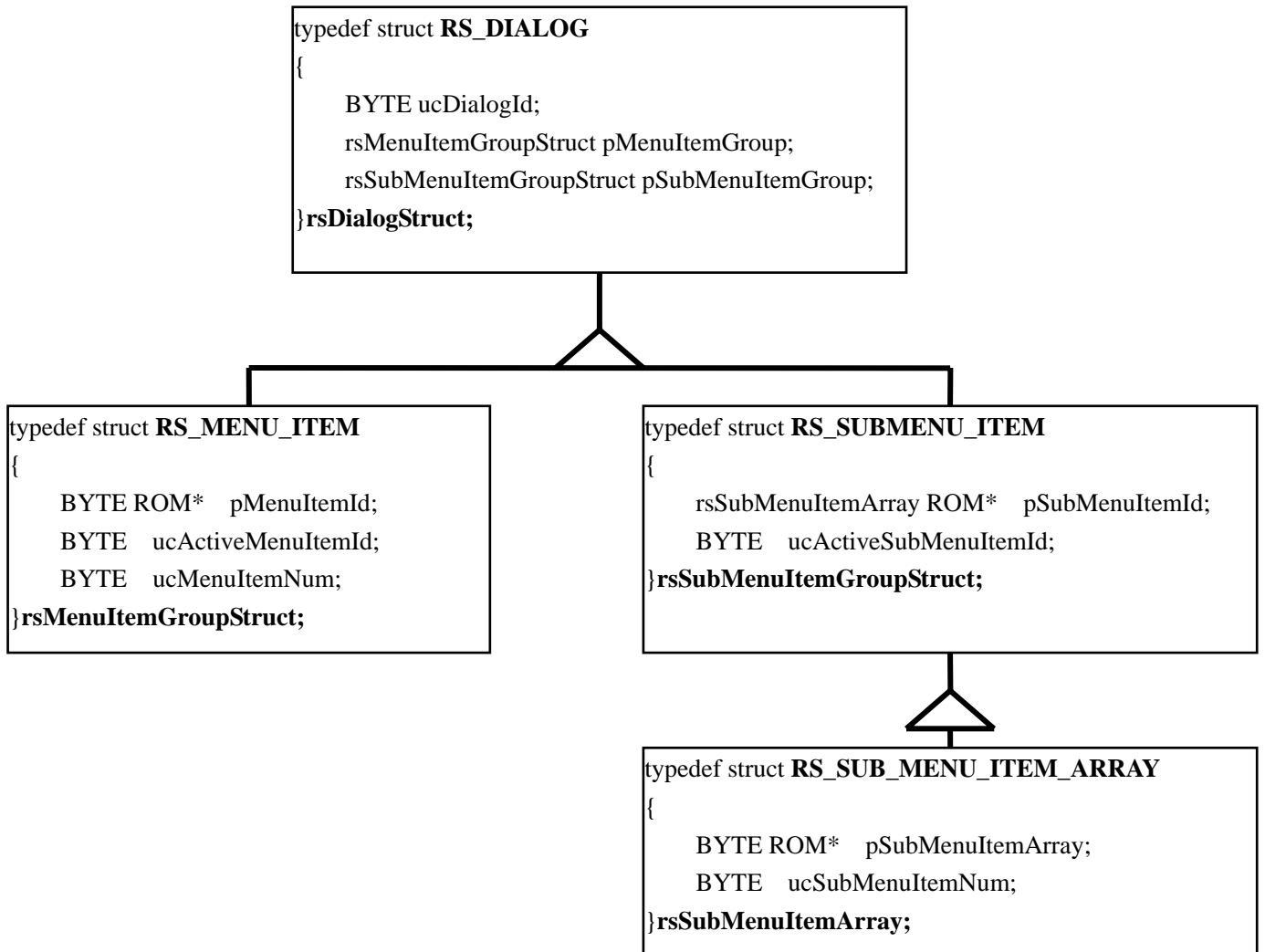


3.2.4 用户接口的实现

用户接口的实现指的是菜单的实现，因为一台电视最终要通过菜单由用户操作。在软件实现中，系统将每副菜单抽象为三层：第一层是Dialog, 任何菜单都属于一个Dialog, 菜单的设计者会将不同功能的菜单组织在各个不同的Dialog中，每一个Dialog都有一个ID来标记。第二层是Menu Items，即在Dialog下将菜单按功能细分为若干组，每组都用ID来标记。第三层就是具体的菜单项，成为Sub Menu Items, 不同的菜单项属于不同的Menu Item。



Dialog Struct (结构体)定义



如何构造一个Dialog

①首先在status.h文件中找到已定义好的DIALOG_ID, MenuItem_ID and SubMenuItem_ID, 该文件有专门的前端OSD工具生成。

rsStatusDefine.h

```
#ifndef __RSSTATUSDEFINE_H__
#define __RSSTATUSDEFINE_H__
// DIALOG_ID enum
enum
{
    DLG_ID_NORMAL_DIALOG,
    .....
    DIALOG_ID,
    .....
    DLG_ID_END
};

// MENU_ITEM enum
enum
{
    MENU_ITEM_ID_BEGIN = DLG_ID_END,
    MENU_ITEM_PICTURE_ID = MENU_ITEM_ID_BEGIN ,
    MENU_ITEM_AUDIO_ID,
    MENU_ITEM_CONFIG_ID,
    MENU_ITEM_TIMER_ID,
    MENU_ITEM_ID_END,
};
#define MENU_ITEM_ID_COUNT (MENU_ITEM_ID_END -
MENU_ITEM_ID_BEGIN)
```

```
// SUB_MENU_ITEM enum
Enum
{
SUB_MENU_ITEM_ID_BEGIN = MENU_ITEM_END,
SUB_MENU_ITEM_PICTUREMODE_ID = SUB_MENU_ITEM_BEGIN,
SUB_MENU_ITEM_BRIGHTNESS_ID,
SUB_MENU_ITEM_CONTRAST_ID,
SUB_MENU_ITEM_SATURATION_ID,
SUB_MENU_ITEM_SHARPNESS_ID,
SUB_MENU_ITEM_ADVANCED_ID,
SUB_MENU_ITEM_ID_END
};

#define SUB_MENU_ITEM_ID_COUNT (SUB_MENU_ITEM_ID_END -
SUB_MENU_ITEM_ID_BEGIN)

.....

.....

.....

#endif /*__RS_STATUSDFINE_H__*/
```

②构造MenuItemGroup

```
BYTE ROM rsMenuItemArray[]=
{
    MENU_ITEM_PICTURE_ID,
    MENU_ITEM_AUDIO_ID,
    MENU_ITEM_CONFIG_ID,
    MENU_ITEM_TIMER_ID
};

rsMenuItemGroupStruct rsMenuItem =
{
    rsMenuItemArray,           // Menu item array
```

```

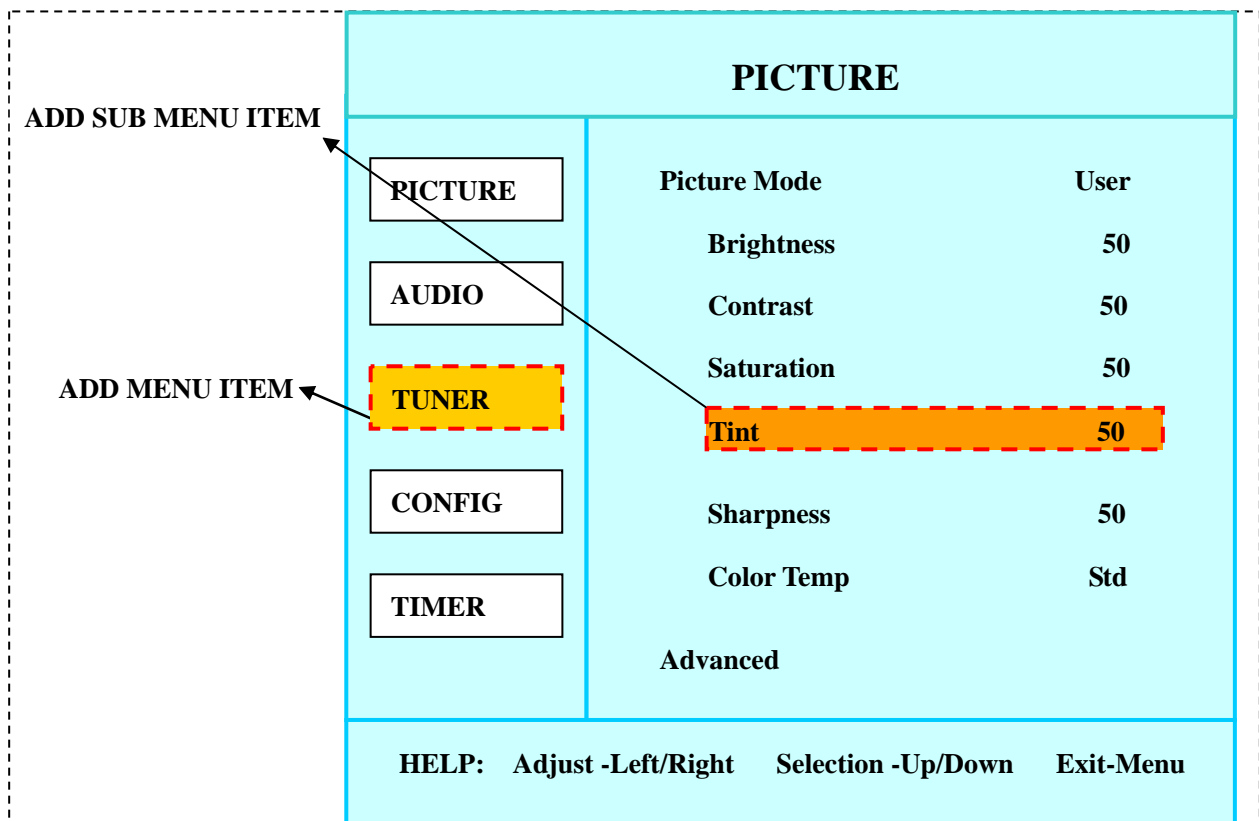
        MENU_ITEM_PICTURE_ID,                // Defaut Active Menu Item Id
        MENU_ITEM_ID_COUNT                    // Menu items Num
    };
3. Contruct SubMenuItemGroup
BYTE ROM rsPictureSubMenuItemArray[] =
{
    SUB_MENU_ITEM_PICTUREMODE_ID
    SUB_MENU_ITEM_BRIGHTNESS_ID,
    SUB_MENU_ITEM_CONTRAST_ID,
    SUB_MENU_ITEM_SATURATION_ID,
    SUB_MENU_ITEM_SHARPNESS_ID,
    SUB_MENU_ITEM_AD VANCED_ID
};
rsSubMenuItemArray ROM rsPictureSubMenuItemArray=
{
    rsPictureSubMenuItemArray,
        SUB_MENU_ITEM_ID_COUNT,
};
rsSubMenuItemArray ROM rsAudioSubMenuItemArray=
{
    ... ...
};
rsSubMenuItemArray ROM rsTunerSubMenuItemArray=
{
    ....
};
rsSubMenuItemArray ROM rsConfigSubMenuItemArray=
{
    .....
};
rsSubMenuItemArray ROM rsTimerSubMenuItemArray=

```

```
{
.....
};
rsSubMenuItemArray ROM rsSubMenuItemArray[]=
{
rsPictureSubMenuItemArray,      // Picture SubMenu
rsAudioSubMenuItemArray,        // Audio SubMenu
rsTunerSubMenuItemArray,        // Tuner SubMenu
rsConfigSubMenuItemArray,       // Config SubMenu
rsConfigSubMenuItemArray        // Timer SubMenu
};
RsSubMenuItemGroupStruct ROM rsSubMenuItem=
{
    rsSubMenuItemArray,          // Sub Menu Item Array
    MENU_ITEM_PICTURE_ID,       // Active Sub menu ID
};
4. Construct a Dialog
rsDialogStruct ROM rsTVMainDlg =
{
    DIALOG_ID,                  // Dialog Id
    rsMenuItem,                 // Menu Item
    rsSubMenuItem               // Sub Menu Item
};
```

增加一个菜单项（MenuItem）或者子菜单项（SubMenuItem）的方法如下：

- ① 在OSD工具中添加相应的菜单，生成资源文件



② 在新生成的资源文件status.h文件中会有如下的变化：

```
// MENU_ITEM enum
enum
{
MENU_ITEM_ID_BEGIN = DLG_ID_END,
```



```

    MENU_ITEM_PICTURE_ID = MENU_ITEM_ID_BEGIN ,
    MENU_ITEM_AUDIO_ID,
    MENU_ITEM_TUNER_ID,          // This Item is new added
    MENU_ITEM_CONFIG_ID,
    MENU_ITEM_TIMER_ID,
    MENU_ITEM_ID_END,
};
// SUB_MENU_ITEM enum
Enum
{
    SUB_MENU_ITEM_ID_BEGIN = MENU_ITEM_END,
    SUB_MENU_ITEM_PICTUREMODE_ID = SUB_MENU_ITEM_BEGIN,
    SUB_MENU_ITEM_BRIGHTNESS_ID,
    SUB_MENU_ITEM_CONTRAST_ID,
    SUB_MENU_ITEM_SATURATION_ID,
    SUB_MENU_ITEM_TINT_ID,          // This Sub Menu Item is new
added
    SUB_MENU_ITEM_SHARPNESS_ID,
    SUB_MENU_ITEM_ADVANCED_ID,
    SUB_MENU_ITEM_ID_END
};

```

③ 更改相应的结构体

rsPictureSubMenuItemArray

```

BYTE ROM rsMenuItemArray[]=
{
    MENU_ITEM_PICTURE_ID,
    MENU_ITEM_AUDIO_ID,
    MENU_ITEM_TUNER_ID,          // added new menu item
    MENU_ITEM_CONFIG_ID,

```

```
MENU_ITEM_TIMER_ID
};
BYTE ROM rsPictureSubMenuItemArray[] =
{
SUB_MENU_ITEM_PICTUREMODE_ID
SUB_MENU_ITEM_BRIGHTNESS_ID,
SUB_MENU_ITEM_CONTRAST_ID,
SUB_MENU_ITEM_TINT_ID,          // added new sub menu item
SUB_MENU_ITEM_SATURATION_ID,
SUB_MENU_ITEM_SHARPNESS_ID,
SUB_MENU_ITEM_AD VANCED_ID
};
```

3.3 ATV 软件开发的缺陷

目前，绝大多数的DTV软件开发都是在芯片厂商提供的API函数库的基础上进行功能扩展或开发的。用户实现了自己的应用系统后通过编译、链接把应用系统和底层函数库连接起来，使得它们协同工作。这同时带来一些缺点：

(1) 当更换芯片时尽管有可能不用修改源程序，但由于API函数库发生了改变，那么应用系统仍然需要重新进行编译、链接，否则就无法正常运行。

(2) 应用系统可移植性差。由于上层应用高度依赖基于特定厂商芯片的API函数库，使得应用系统的可移植性很差，即便是同一应用要想移植到不同厂商提供的芯片上也需要进行重复开发。

(3) 当API非常多时，使用会非常不方便，需要对函数进行组织。往往很多厂商提供的API函数库都附带了大量的函数使用手册，开发人员首先要花费很长的时间来熟悉这些函数的功能和用法，而这些函数由于数量庞大而又缺乏组织，所以难于被开发人员尽快掌握。

DTV软件开发平台是针对当前数字电视软件开发的具体特点进行设计的，它

具有很多传统开发方法所无法比拟的技术优势:

(1) 是面对数字电视领域的中间件平台。其架构满足数字广播领域中间件的需求, 它提供了跨网络和平台的透明性的应用或服务的交互, 比如通过广播网络传输的各种多媒体应用。

(2) 通过该平台可以使应用软件能够的动态加载到电视节目里, 做到即插即用。

(3) 支持数字电视的硬件设备的即插即用, 即实现了软件的跨平台, 可以做到“一次编写, 多处运行”, 降低软件移植的风险。

(4) 平台遵守esCOM标准, 这样一来技术服务商只要遵循这标准, 就可以自由开发应用软件, 技术服务商的工作效率将大为提高。

(5) 具有可拆卸的可视化开发环境。基于DTV中间件平台的可视化开发环境——DTV Visual Developer为应用软件开发提供了统一的开发方法, 它基于“控件”模型, 具有可视化开发方法的诸多优点, 同时它是可拆卸并支持动态配置的, 用户可以根据自己的需要来对其进行自定义。基于该工具可以极大的提高数字电视软件开发的效率。

(6) 在数字电视软件开发领域, 导入先进的工程化软件开发技术。数字电视软件一般用汇编语言、C语言, 在少数系统中已经支持了C++开发, 但是由于还没有一个开发环境能够提供构件化的开发运行环境, 可以说, 数字电视软件开发还是停留在手工作坊式的开发方式上。DTV软件开发平台使得数字电视的应用软件开发能够实现工程化、工厂化生产。

第 4 章 数字模块的系统结构和实现

4.1 DTV 硬件平台结构

4.1.1 硬件组成

硬件部分由高频调谐器，QAM信道解调器，信源数据传输流解复用器，MPEG-2 解码器，视频编码器，音频PCM解码器，嵌入式CPU与存储器模块和接口电路，电缆调制解调（CM）模块，条件接受模块等组成。高频调谐器棘手来自有线电视网络的射频信号，通过QAM信道解调器完成信道解码，从载波中解调出包含视音频和其他数据信息的MPEG-2传输流。传输流中一般包含多个视音频流及一些数据信息，传输流解复用器来区分不同的节目，提取相应的视音频流和数据流，其输出为MPEG-2视音频基本流（ES）以及数据净荷，然后由MPEG-2解码器和相应的解析软件完成数字信息的还原。对于付费电视，条件接受模块还对视音频流实施解密。MPEG-2解码器完成视音频信号的解压缩，经视频编码器和音频PCM解码器变换，还原出模拟视音频信号，在模拟电视机上显示高质量图像。并提供多声道立体声音频节目。电缆调制解调模块 由一个双向调谐器和媒体访问控制器（MAC）组成，该部分完

调谐 且含有 A/D 转换 并行记 QAM 解调 能卡 解复用 去用户正

常收看。MPEG-2解码器完成视音频信号的解压缩，经视频编码器和音频PCM解码器变换，还原出模拟视音频信号，在模拟电视机上显示高质量图像。并提供多声道立体声音频节目。电缆调制解调模块 由一个双向调谐器和媒体访问控制器（MAC）组成，该部分完

视频解码 解调 PAL/NTSC 编码 嵌入 视频输出 模块

用来存 PES 分析 件系统，并对各个硬件模块进行控制。接口电路提供了丰富的外部： 串行接口USB 高速串行接口1394 以太网接口 RS232， IDE硬盘接口，视音频接

音频解码 PCM 解码 音频输出

硬件结构图：

4.1.2 关键技术

信道解调技术

再有线电视网络中传输数字电视及给类增值业务采用的是QAM信道调制技术，如64QAM。相应地，数字电视接收机顶盒对信道解调采用的是QAM信道解调技术。

信源解码技术

数字电视广播采用MPEG-2视频压缩标准，信源解码器必须适应不同编码策略，正确地还原原始视音频数据。

大规模集成芯片技术

为实现实时地解复用和数据信息处理，目前数字电视接收机顶盒大多采用专用芯片，将嵌入式CPU与MPEG-2传输流解复用器，DVB-C通用解扰器，MPEG-2视音频解码器，NTSC/PAL编码器等硬件进行集成，形成STB核心芯片，甚至还可以集成电缆调制解调模块，以完成电缆调制解调器的所有功能。

条件接收（CA）技术

CA负责完成数字电视广播系统的用户认证，授权和节目加扰工作，它解决如何从用户处收取费用和阻止用户收看那些未经授权地付费频道这两个问题。CA的基本原理是对数字电视节目进行加解扰和加解密处理，采用加扰控制字加密传输的办法，用户端利用IC卡解密，具体地讲，就是将复用后的传输流与一个加扰伪随机序列进行模2相加，这个伪随机加扰序列由控制字发生器提供的控制字确定，CA的核心实际上是CW传输的控制。在MPEG-2传输流中，与CW传输相关的数据流为授权控制信息（ECM）和授权管理信息（EMM），授权管理信息中包括地址，用户授权信息等，由业务密钥加密处理后CW在授权控制信息数据流中传送，对CW加密的SK在授权管理信息数据流中传送，并且SK在传送前要经过用户个人分配密钥（PDK）的加密处理，PDK存放在用户的智能卡中。用户端数字电视接收机顶盒为了再生出解扰伪随机序列，必须获取相关的条件接收信息，STB从接收到的传输流中过滤出ECM和EMM数据流后，通过SMART CARD接口送给SMART CARD，SMART CARD首先读取PDK，利用PDK对EMM解密，取出SK，再利用SK对ECM解密，

取出CW，并讲CW通过SMART CARD接口送给解扰引擎，解扰引擎利用CW即可将已加扰的TS进行解扰。

4.2 ATSC 标准

美国高清晰度数字电视联盟（ATSC）是由国内外一百多个电视技术公司组成的，其主要业务之一是制订包括数字式高清晰度电视（HDTV）在内的先进电视系统的技术标准。1996年12月24日，美国联邦通信委员会（FCC）正式采纳了ATSC数字电视标准（A/53）的主要要素作为美国数字电视的标准。

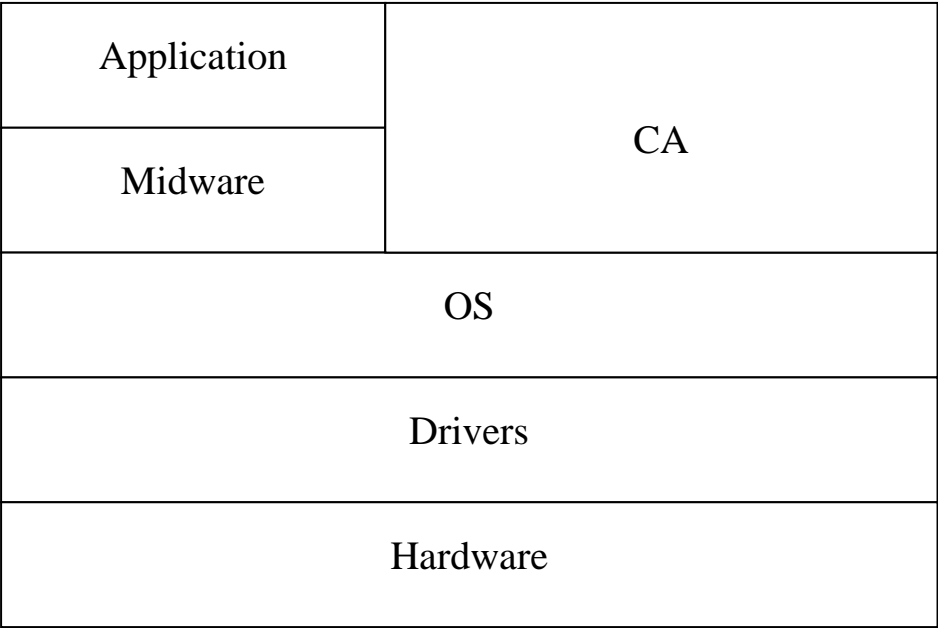
ATSC数字电视标准由四个分离的层级组成，层级之间有清晰的界面。最高为图像层，确定图像的形式，包括像素阵列、幅型比和帧频。接着是图像压缩层，采用MPEG-2压缩标准，再下来是系统复用层，特定的数据被纳入不同的压缩包中，采用MPEG-2压缩标准。最后是传输层，确定数据传输的调制和信道编码方案。对于地面广播系统，采用Zenith公司开发的8-VSB传输模式，在6MHz地面广播频道上可实现19.3Mb/Sr 传输速率。该标准也包含适合有线电视系统高数据率的16-VSB传输模式，可在6MHz有线电视信道中实现38.6Mb/s的传输速率。

下面两层共同承担普通数据的传输。上面两层确定在普通数据传输基础上运行的特定配置，如HDTV或SDTV；还确定ATSC标准支持的具体图像格式，共有18种（HDTV 6种、SDTV 12种），其中14种采用逐行扫描方式。

4.3 DTV 软件架构

DTV端包括硬件和软件两部分，可以分为四层，从底向上分别为：硬件，底层软件，中间件，应用软件。硬件提供数字电视接收的硬件平台。底层软件提供操作系统以及各种硬件驱动程序。中间件指居于操作系统与应用软件中间的软件部分，它将应用软件与倚赖于硬件的底层软件分隔开来，使应用软件不倚赖于具体的硬件平台。从而将各种应用的开发变得更加简捷，使产品的开放性和可移植性更强。应用软件独立于硬件平台，负责执行服务商提供的各种服务功能，如电子节目指南（EPG），视频点播（VOD），Internet接入，条件接受等。

软件架构图：

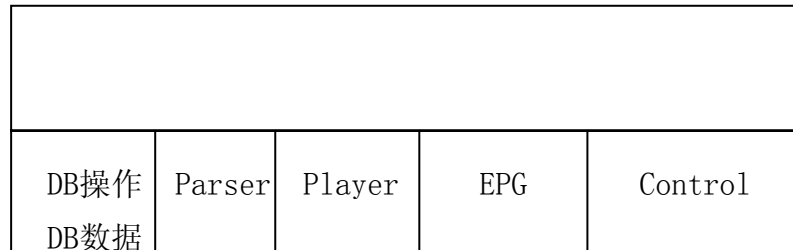


DTV基础构件库是整个DTV中间件平台的基础，它封装了针对数字电视软件开发所必需的函数接口，并以构件化的形式提供给上层用户。这些构件都是可拆卸的，系统可以按照需要进行剪裁组装，或在运行时动态加载必要的构件，可以很容易定制成为针对不同硬件配置的紧凑高效的DTV中间件平台。

下面将详细的介绍OSD(On Screen Display)构件、IO（Input/Output）构件、视频解码构件和音频解码构件的设计。

4.4 DTV 中间件的设计与实现

系统纵向分为App layer,MW layer,Porting layer,OS四个层次，上层只能调用相邻的下层所提供的借口，不可以越层调用。其中MW layer分为5个模块，为App layer提供所有的功能借口。



系统层次结构图

4.4.1 DB 模块的设计

DB模块分为分成两个部分：数据和操作接口，数据在本模块外是不可见的。

数据部分设计：

- (1). MPEG II 协议对应的表：PAT, PMT, CAT。
- (2). ATSC标准中对应的表：VCT。

这里的每个表对应内存中的一个全局数组和flash上的一个文件。对应于每个表的数据结构只包含相应协议中包含的字段，不添加额外的字段。

数据操作部分设计：

对应于每个表只提供六个操作：

- 表名_Add(record) 增加一条记录
- 表名_search(key) 定位一条记录
- 表名_Query_field(fieldname) 查询一个字段
- 表名_Load() 加载数据到内存
- 表名_Reset() 重置数据
- 表名_Save() 保存至文件

4.4.2 Parser 模块的设计

这个模块用于解析DB中涉及到的每个表。对于每个表提供了如下的借口：

Parse_表名 (timeout)

这个函数要求是异步函数，同时需要设置超时。成功后，通过消息机制通知调用线程。函数内部将调用DB的借口操作取得的数据。

4.4.3 Player 模块的设计

这个模块是对播放操作的封装，需要实现一个包含以下4个状态的状态机，idle, opening, opened, starting。

该层对上层提供两个函数：

play() 启动播放

stop() 停止播放

这两个函数都要求是异步函数。

4.4.4 EPG 模块的设计

EPG信息包括两部分：当前节目和下一个节目，一段时间内的所有节目。

根据标准，每个频点的EPG信息应该都是完整的，即每个频点都包含所有频点的EPG的信息。

EPG模块初始化：开机时，在程序初始化阶段，会初始化EPG模块的一个线程，该线程根据其它模块发过来的消息，执行相应的解析工作。

流程说明:EPG TASK在开机后建立。建立后通过发送消息开始解析当前频道的EIT信息。EPG TASK的运行状态由内部的状态机控制。

两种情况下可以调整其运行状态（调整通过向EPG TASK发送消息实现）。

- (1) 搜台。搜台前消息EPG TASK暂停当前解析（state=stop）。等搜台结束后，消息EPG TASK继续开始解析。EPG TASK重新开始解析，会清空结构rsEventData。
- (2) 换台。换台前EPG TASK暂停当前解析（state=stop）。等换台结束后，消息EPG TASK继续开始解析。EPG TASK重新开始解析，同样清空结构rsEventData。

4.5 DTV 中间件平台中的消息机制

4.5.1 DTV 中间件平台中的消息定义

在 DTV 中间件平台中主要定义了四大类的消息，它们基本涵盖了数字电视软件开发过程中所能遇到的消息集合，下面将分别介绍这四类消息。

(1) 系统消息:这一类消息来源于DTV解释层中的“模式处理构件”，当前模式类型的状态（包括模式的有效性、模式的幅度等）、模式间的切换、屏幕的刷新等都会产生这一类消息，并由“模式处理构件”将它们发送给允许接收这些消息的其它构件或上层的应用程序。系统消息的定义及其对应的事件如表 5.1。

表5.1 系统消息及对应事件的定义

消息定义	事件定义
	TVE_NONE
TVM_NO_SYNC	TVE_NoSync
TVM_NO_CABLE	TVE_NoCable
TVM_VALID_MODE	TVE_ValidMode
TVM_MODE_CHANGE	TVE_ModeChange
TVM_OUT_OF_RANGE	TVE_OutOfRange
TVM_SPLASH_SCREEN	TVE_SplashScreen
TVM_AUTO_ADJ	TVE_AutoAdjSuccess
TVM_SYNC_STANDBY	TVE_NoSync
TVM_SYNC_SUSPEND	TVE_NoSync

(2) 键盘消息: 键盘中某一按键或组合键的按下就会触发键盘事件，对于每个按键的具体解释由用户自定义的按键翻译表来决定。

(3) 电源管理消息: 电源管理消息主要来源于对电源状态的监测，主要包括电源的打开、关闭和待机等。电源管理消息的定义及其对应的事件如表5.2。

表5.2 电源管理消息及对应事件的定义

消息定义	事件定义
TVM_POWER_ON	TVE_PowerOn
TVM_POWER_STANDBY	TVE_NoSync
TVM_POWER_SUSPEND	TVE_NoSync
TVM_POWER_DOWN_NOTICE	TVE_RemoveOSD
TVM_POWER_DOWN	TVE_Sleep

（4）时钟消息：系统时钟在每个时钟周期都会发出一个时钟消息，这主要用于满足系统同步的需要或用在某些需要计时的场合。

4.5.2 DTV 中间件平台中的消息流分析

从硬件加电到关机，系统中的各种消息按照某种特定的轨迹流动于中间件平台的各部分构件之间，消息流是系统构件之间沟通的最主要方式，图 5.2 表示了 DTV 中间件平台中的消息流动状况。

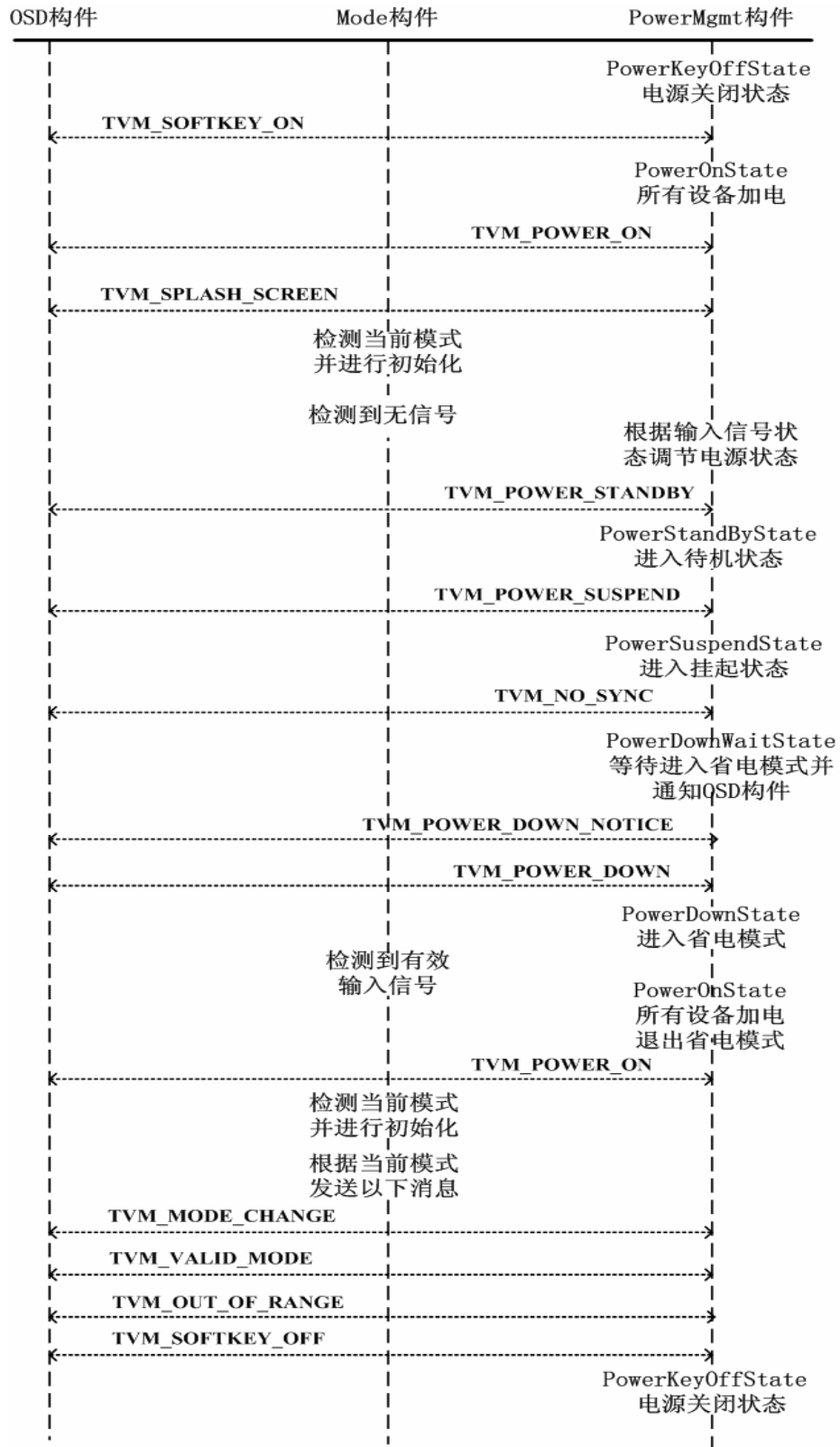


图 5.2 DTV 中间件平台的消息流分析

第5章 数模一体机通讯的设计和实现

5.1 数模通讯的整体设计方案

5.1.1 串口通讯机制

ATV 和 DTV 通讯全部通过串口进行,当 ATV 要调用 DTV 功能,发送串口命令到 DTV,DTV 串口程序接收命令,发送用户事件到 DUMMYGUI,DUMMYGUI 负责转发和执行,最后由 DUMMYGUI 得到要发还的 ACK,转发到串口通讯程序,发还 ATV

所有通讯都由 ATV 主控,DTV 响应,ATV 只发送 COMMAND 给 DTV,DTV 只发送 ACK 到 ATV,不会发送 COMMAND,

ATV 发送 COMMAND 到 DTV,等待 ACK,如果在 100MS 内未等到,则重发,如果连续三次重发都未等到,结束本次命令,返回错误。

5.1.2 发送机制

数据拆分由 DTV 端完成。串口通讯部分只接收数据小于等于 25byte 的数据,如果数据内容超过 25byte,将要对数据进行分块,使每次发送到串口的数据长度不超过 25bytes.

由于 ATV 端直接使用一次收到的数据,而不等待后续包拼装,,DTV 端拆分数据需要加以控制,不能把同一类型的数据拆分在两个包中传输(如一个 WORD 类型的数据有两个字节,不能把两个字节拆分在两个包中)。由于节目介绍字符串有可能大于 25bytes,所以在分块时可以将字符串分在不同的包中,但是对于其中某个字符占用大于一个 BYTE 的情况(如一个中文字符),必须放在同一个包中传输。

5.2 数模通讯的具体实现

5.2.1 数据结构的定义

根据 ATV 和 DTV 传送的数据量的不同而定义了两种类型的数据结构：

1. 传送的数据少于 6 个字节，采用如下的结构

CmdType	<u>CmdType</u>	Length	CmdCode	Data1	Data2	Data3	Data4	Data5	Data6
---------	----------------	--------	---------	-------	-------	-------	-------	-------	-------

“Length”代表了整个结构的长度，该字段的大小与不同的命令有关；

2. 传送的数据多于 6 个字节，采用如下的结构

CmdType	<u>CmdType</u>	Length	CmdCode	Data 1	Data 2	Data 3	Data4	Data 5	Checksum
.....									
.....									
CmdType	<u>CmdType</u>	Length	CmdCode	Data 1	Data 2	0x00	0x00	0x00	Checksum

此时会将数据划分到不同的包中，每个包的长度固定，为 10 个字节，字段“Length”表示剩下的未发送的字节数（CmdType, length, CmdCode, CheckSum 不包括在其中）。

5.2.2 发送命令的定义

Command Code & Command Data

CmdCode	Value	Data bytes	Description
ITC_TEST_DATA_GET	0x30	0	Test communication between DTV and ATV. ATV will get test data 0x01, 0x02, 0x03, 0x04 from DTV.
ITC_ACK	0xD0	5	DTV reply with test data 0x01, 0x02, 0x03, 0x04 Data 1: 0x30 (ITC_TEST_DATA_GET CmdCode) Data 2: 0x01 Data 3: 0x02 Data 4: 0x03 Data 5: 0x04

ITC_TEST_DATA_SET	0x31		Test communication between DTV and ATV. ATV (or DTV) will send test data 0x05, 0x06, 0x07, 0x08 to DTV (or ATV). Data 1: 0x05 Data 2: 0x06 Data 3: 0x07 Data 4: 0x08
ITC_ACK	0xD0	1	Acknowledge for data received. Data 1: 0x31 (ITC_TEST_DATA_SET CmdCode)
ITC_PWD_GET	0x32	0	ATV requests password setting in DTV
ITC_ACK	0xD0	5	DTV reply password to ATV (The input character is between 0-9) Data 1: 0x32 (ITC_PWD_GET CmdCode) Data 2: 1 st password digit. Data 3: 2 nd password digit. Data 4: 3 rd password digit. Data 5: 4 th password digit.

ATV 发送数据给 DTV 后必须要得到 DTV 的响应命令 (ITC_ACK)。

当 ATV 需要得到 DTV 的密码，相应的数据结构如下：

ATV -> DTV: 0xD8 0xD8 0x04 0x32

DTV -> ATV: 0xD8 0xD8 0x09 0xD0 0x32 0x01, 0x02, 0x03, 0x04.

第6章 可视化开发工具——DTV Visual Developer 的设计

6.1 可视化开发方法简介

所谓可视化开发,就是指:在软件开发过程中,用直观的具有一定含义的图标按钮、图形化的对象取代原来手工的抽象的编辑、运行、浏览操作,软件开发过程表现为鼠标点击按钮和拖放图形化的对象以及指定对象的属性、行为的过程。这种可视化的编程方法易学易用,而且可以大大提高工作效率。它的特点是把原来抽象的数字、表格、功能逻辑等用直观的图形、图象的形式表现出来。可视化编程技术已经成为当今软件开发最重要的工具和手段,尤其是 Power Builder、Visual C++等开发工具的出现,大大推动了可视化编程技术的发展。

传统的编程方法使用的是面向过程、按顺序进行的机制,其缺点是程序员始终要关心什么时候发生什么事情,应用程序的界面都需要程序员编写语句来实现,对于图形界面的应用程序,只有在程序运行时才能看到效果,一旦不符合需求,还需要修改程序,因而使得开发工作非常烦琐。用可视化开发工具进行应用程序开发主要有两部分工作:即设计界面和编写代码。在开发过程中所看到的界面,与程序运行时的界面基本相同,同时可视化开发工具一般还向程序员提供了若干界面设计所需要的对象(称为控件),在设计界面时,只需将所需要的控件放到窗口的指定位置即可,整个界面设计过程不需要编写代码。

可视化编程的另一个重要特点是它为用户提供了所谓的 IDE (Integrated Development Environment),即集成开发环境。较早期程序设计的各个阶段都要用不同的软件来进行处理,如先用字处理软件编辑源程序,然后用链接程序进行函数、模块连接,再用编译程序进行编译,开发者必须在几种软件间来回切换,不仅操作烦琐,而且极易出错。而集成开发环境则将编辑、编译、调试等功能集成在一个桌面环境中,开发者所有的工作都可以在这个环境下一次完成,这样就大大方便了用户。

6.2 可视化开发方法在 DTV 软件开发中的应用

数字电视软件开发与传统的 PC 机上的软件开发既有相似之处也有其自身的特点。它们的共同点主要有：

（1）与用户交互的方式都是基于图形界面，用户通过对图形界面中的相关控件的操作对应用程序发出指令。

（2）图形界面中的控件都应该能够响应事件，事件可以由用户操作触发，也可以由来自系统的消息触发。

（3）应用程序没有预定的执行路径，而是在响应不同的事件时执行不同的代码片段。

数字电视软件开发由于受到硬件资源的限制，相对于基于PC机的软件开发又有自身的特点：

（1）用户界面相对简单，主要的表现形式是菜单，图形界面中的控件也只包括按键和滚动条等最基本的元素。

（2）用户主要通过遥控器来发出指令并由此触发系统中相应的事件，这一过程与PC机上通过鼠标和键盘来进行输入操作具有很大的不同，输入手段的单一性和简单性决定了用户界面必须具有易于操作和导航功能强大的特点。

（3）考虑到系统资源对软件运行的限制，用户发出事件后，若处理器负载较重，可以允许响应有相应的延迟。

综上所述，在数字电视软件开发中引入可视化开发方法是可行的，尤其是随着数字电视芯片性能的不不断提升以及对数字电视应用软件需求的不断攀升，可视化编程将是大势所趋，这也是提高当前数字电视软件开发效率的重要手段之一。

6.3 DTV Visual Developer 的系统设计

DTV Visual Developer 是建立在 DTV 中间件平台上的可视化开发工具，它根据数字电视软件开发的特点，把界面设计中使用频繁的对象封装为控件，以供用户方便的调用。在开发过程中所看到的用户界面框架，与程序运行时的界面基本相同，这就使得软件的设计阶段可以在脱离真机的情况下完成，而不必把大量的时间花在重复下载程序到芯片以及频繁开机调试的过程中，从而大大的提高软件开发的效率。下图是 DTV Visual Developer 的主应用程序界面：

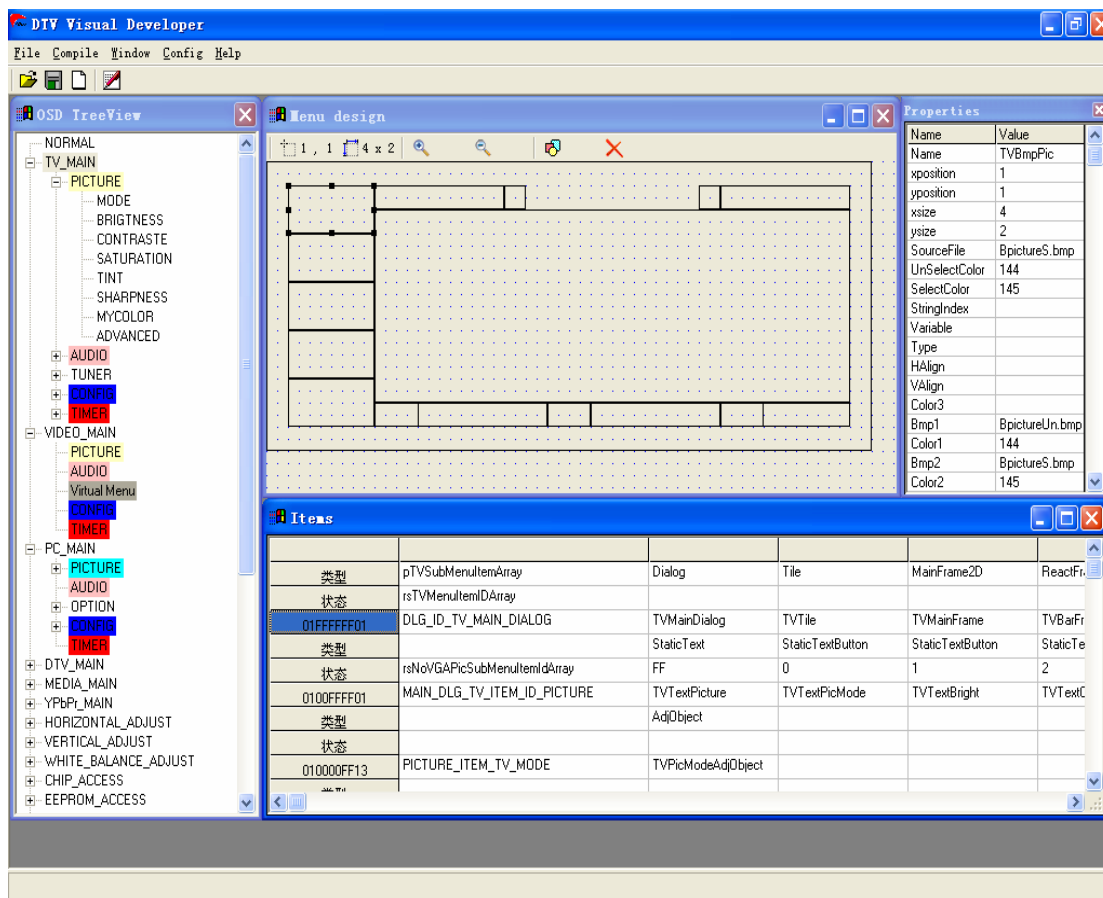


图 6.1 DTV Visual Developer 的主应用程序界面

6.3.1 DTV Visual Developer 的系统构成

DTV Visual Developer 主要由四个工作区组成：

(1) OSD TreeView: 是编辑菜单状态的主要场所，菜单状态采用树状结构表示，父菜单与子菜单的关系通过父节点与子节点的关系来模拟。

(2) Menu Designer: 是用户界面设计的工作区，在这里设计的用户界面框架与程序运行时的图形界面基本相同，它提供了添加、删除、以及调整控件形状和功能的功能。

(3) Items Window: 其中的每一项都对应于一个菜单的状态，它包含了这个菜单状态下的所有控件，无论该控件是可见的或是不可见的。

(4) Properties Window: 在这里可以编辑选中控件的各种属性和行为。

此外，DTV Visual Developer 还提供了诸如变量表、字符串表和颜色表等

方便用户使用的功能。同时，这个集成开发环境也是可配置的，例如控件的种类，每种控件的属性以及程序中使用的字库等都是可以根据用户的需要来定制的，这就保证了对开发工具的灵活性和适应性的要求。

6.3.2 DTV Visual Developer 的设计要点

在 DTV Visual Developer 中，所有的菜单状态均被抽象成为 OSD TreeView 中的一个树节点，每个状态都对应一个唯一的编号，编号中除最后两位外每两位代表一个菜单的层次，如“01FFFFFF01”，就表示这个程序中的菜单深度最大为四层，而最后两位则表示了这一个菜单状态所对应的通道，如 01 代表 TV，02 代表 AV 等。子菜单状态的标号的前几位应与父菜单状态的标号的相应位一致，如菜单状态“01FFFFFF01”的第一个子菜单的编号就应该是“0100FFFF01”，这样设计的好处就是当程序在各菜单状态中跳转时可以方便的通过它们的状态编号来进行调度。

每个菜单状态都被封装成为一个枚举结构，如 TV 通道下的主菜单状态就由以下的结构表示：

```
enum
{
    MAIN_DLG_TV_ITEM_ID_BEGIN = DLG_ID_END,
    MAIN_DLG_TV_ITEM_ID_PICTURE = MAIN_DLG_TV_ITEM_ID_BEGIN,
    MAIN_DLG_TV_ITEM_ID_AUDIO,
    MAIN_DLG_TV_ITEM_ID_TUNER,
    MAIN_DLG_TV_ITEM_ID_CONFIG,
    MAIN_DLG_TV_ITEM_ID_TIMER,
    MAIN_DLG_TV_ITEM_ID_END
};

#define MAIN_DLG_TV_ITEM_ID_COUNT (MAIN_DLG_TV_ITEM_ID_END -
MAIN_DLG_TV_ITEM_ID_BEGIN)
```

当主程序需要查找一个菜单状态时可以通过状态编号定位到一个数组结构，通过这个结构就可以找到对应子状态的内存地址和子状态的数目，如 TV 通道下的主菜单数组结构如下所示：

```
rsSubMenuItemArray ROM pTVSubMenuItemArray[]=
{
    rsNoVGAPicSubMenuItemIdArray, PICTURE_ITEM_TV_COUNT,
    rsAudioSubMenuItemIdArray, AUDIO_ITEM_COUNT,
    rsTunerSubMenuItemIdArray, TUNER_ITEM_COUNT,
    rsConfigSubMenuItemIdArray, CONFIG_ITEM_COUNT,
    rsTimerSubMenuItemIdArray, TIMER_ITEM_COUNT
};
```

6.4 基于 DTV Visual Developer 的数字电视软件开发方法

在使用可视化开发工具 DTV Visual Developer 的数字电视软件开发过程中，所有的 DTV 应用程序都将抽象成为“控件集合”+“资源集合”的模式，图 6.2 解释了这种开发方法。

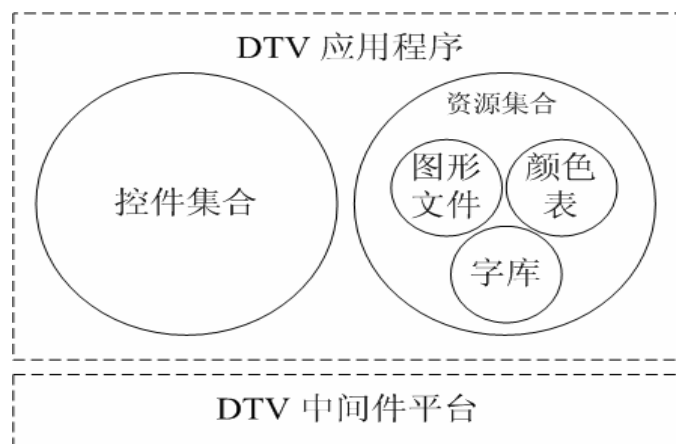


图 6.2 使用 DTV Visual Developer 开发的 DTV 应用程序结构图

这种开发模式的应用使得软件功能的实现将不再以各个功能模块的实现为标志，而是按照“创建用户界面——设置控件对象的属性——编写事件响应代码”这样的流程来完成，这样做的结果不仅极大的提高了软件开发的效率，同时也使得所开发的应用程序逻辑性更强也更易于维护。

在开发数字电视应用程序界面时，一旦遇到对显示效果不满意的情况，例

如希望把图 6.3 中左边的工具栏与右边的详细内容区对调，此时如果采用传统的开发方法，那么整个屏幕的界面几乎要完全重新安排，对于在大量的源程序中完成这项任务的工作量是可想而知的。现在利用 DTV Visual Developer 提供的可视化设计功能，只需在菜单设计窗口中通过鼠标拖动操作来完成对程序界面的修改然后重新编译即可，如图 6.4 所示。

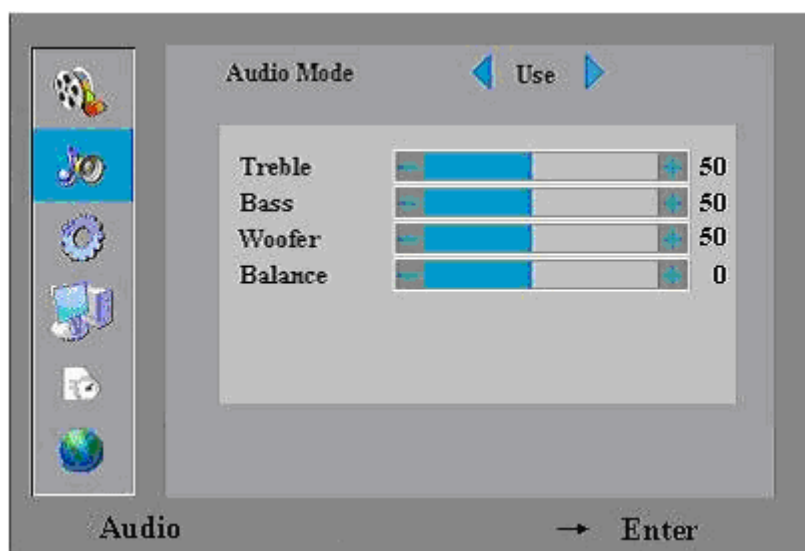
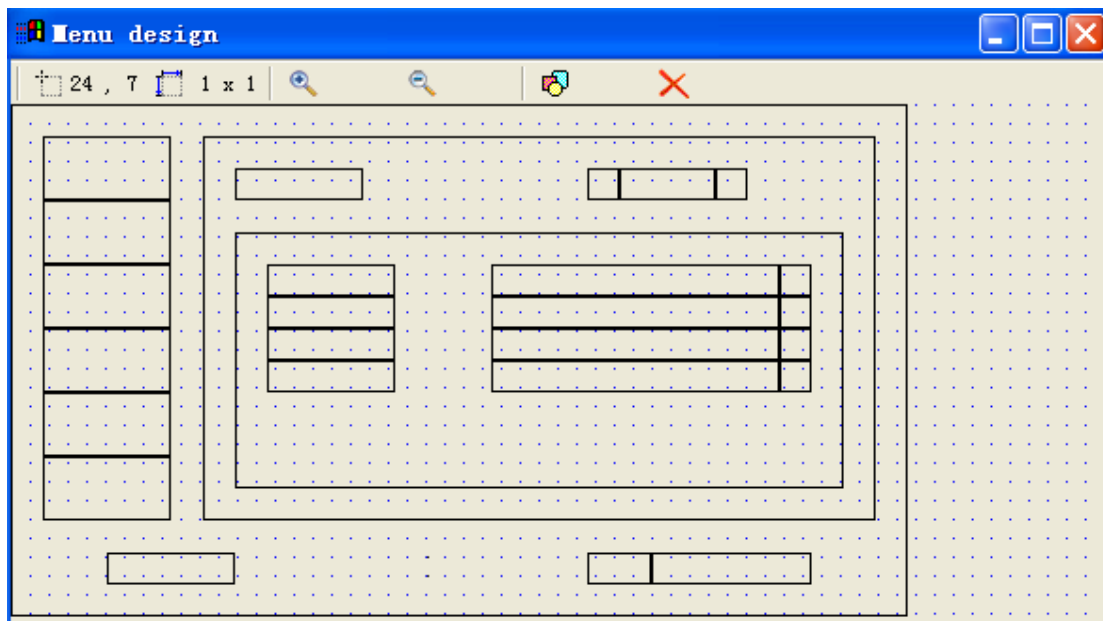


图 6.3 修改前的应用程序界面

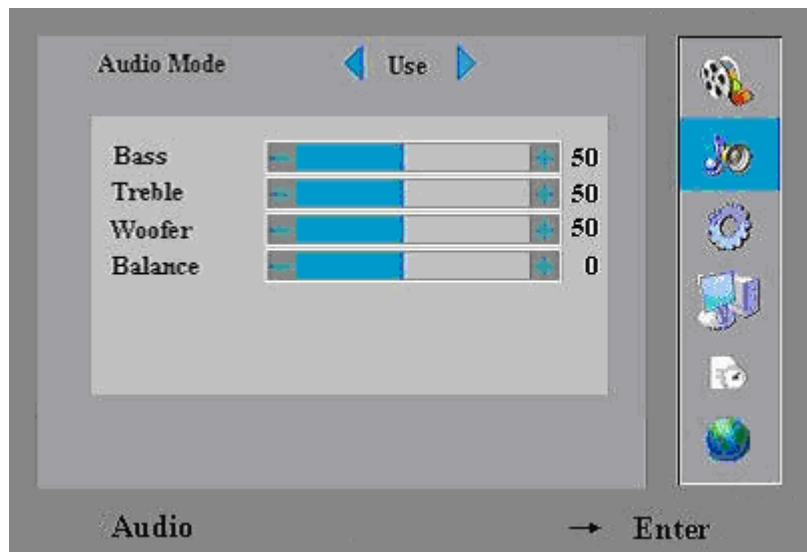
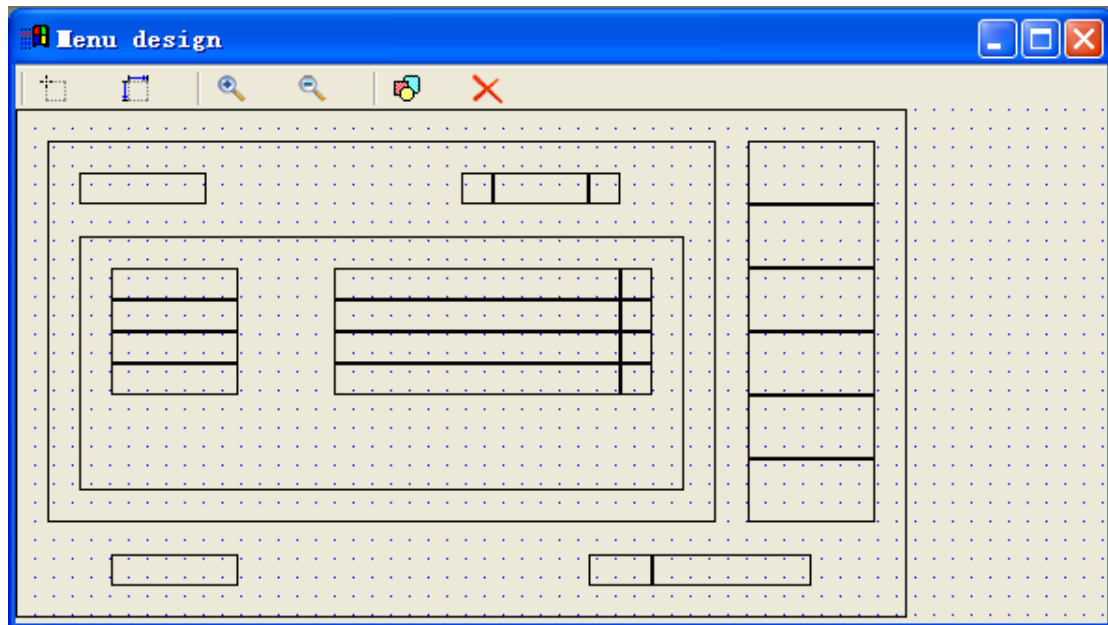


图 7.4 修改后的应用程序界面

第7章 结论与展望

本文结合国家863课题“基于构件、中间件技术的因特网操作系统及跨操作系统的构件、中间件运行平台”，首先对当前数字电视软件开发的现状进行了分析，然后提出了一种基于“和欣”嵌入式操作系统的数字电视解决方案，即基于数字电视软件开发平台的全新开发模式。文中具体阐述了数字电视软件开发平台的系统结构和构成特点，并给出了DTV中间件层的详细设计方案，最后简要分析了基于中间件平台的可视化开发工具——DTV Visual Developer。

本文提出的数字电视软件开发平台可以很好的解决多媒体应用的跨平台问题，应用程序能够不经任何修改，甚至不必重新编译、链接就可以运行在不同硬件厂商提供的数字电视平台上，从而可以做到“一次编写，多处运行”，这对提高数字电视软件开发的效率，增强软件产品的开放性以及提高软件产品的可维护性都具有一定的理论和实践意义。

数字电视技术已被国家列入“十五计划”中的十大优先发展的信息技术产业，2005年我国将进行数字电视的商业播出，2008年用数字电视转播奥运会，2015年停止模拟电视的播放，全面推行数字电视，这一时间表，为像我国这样拥有3亿多电视用户的国家来说，是巨大的发展机会。同时，随着中国数字电视产业标准的即将推出，使得数字电视真正成为不同厂家都可参与竞争的一个产业，国外公司的垄断会被彻底打破，这无疑是运营商所要求的，同时也是数字电视生产商所期待的。因此，我国有关部门应该努力加大对数字电视软件开发技术的研究力度，使得我们的数字电视产业能够后来居上，成为信息产业中的新亮点。

致谢

在论文完成之际，我要衷心感谢所有关心和帮助过我的老师和同学们。

首先，我要感谢我的导师顾伟楠教授。本文是在顾老师的悉心指导和无微不至的关怀下顺利完成的。从论文的选题、资料搜集、理论分析到论文的撰写的全部过程中，都得到了顾老师的精心指导。两年半的研究生学习过程中，我得到了顾老师的极大帮助。顾老师严谨求实，一丝不苟的工作作风以及对学生的严格要求都使我受益匪浅。

其次，我要感谢“上海晖悦数字视频科技有限公司”的各位工程师和领导们，是他们为我提供了良好的试验环境，并且针对我研究过程中所遇到的困难提出了很多重要的建议，他们的工作热情和敬业精神都深深的感动了我。

最后我还要感谢同济大学MIS研究所的全体老师和同学们，是你们伴随着我度过了一段十分有意义的研究生生活。

在此，我向他们表示最衷心的感谢。

参考文献

- [1] D. Batory and S. O'Malley, "The Design and Implementation of Hierarchical Software Systems with Reusable Components," ACM Trans. Software Eng. And Methodology, Oct. 1992
- [2] Fox B., Digital television comes down to earth. IEEE Spectrum, Volume:35, Issue: 10, Oct. 1998
- [3] C. Szyperski, Component Software Beyond Object-Oriented Programming . Addison-Wesley, ADM Press, New York, 1998
- [4] The Koala Component Model for Consumer Electronics Software IEEE 2000
- [5] Steve Maillet, Using COM for Embedded Systems (Part II), Embedded System Conference Session #425, 2000
- [6] D. Stewart, "Software components for real time," Embedded Systems Programming, Dec 2000
- [7] Evain J. P., The multimedia home platform—an overview. E13U Technical Review, Spring 1998
- [8] Michi Henning, Steve Vinoski, 《Advanced CORBA® Programming with C++》Addison Wesley 1999
- [9] Overview of the CORBA Component Model, <http://www.omg.org>, 1999
- [10] M. Fayad and D. Schmidt, "Object-Oriented Application Frameworks", Comm. ACM, Oct. 1997, pp. 32-38
- [11] Evain J. P., The multimedia home platform—an overview. E13U Technical Review, Spring 1998, pp. 4-10
- [12] 科泰世纪有限公司, 《和欣1.1》资料大全。2003
- [13] DON BOX. 《COM本质论》。中国电力出版社, 2001
- [14] 潘爱民. 《COM原理与应用》。清华大学出版社, 1999
- [15] 张俊. 数字电视软件平台——中间件及第三代机顶盒。中国广电技术论文, 2004
- [16] Chengyuan Peng, Pablo Cesar, and Petri Vuorimaa, "Intergration Of Applications Into Digital Television Environment" .the 7th International Conference on Distributed Multimedia systems, September 26-28, 2001
- [17] G. Sivaraman, P. Cesar, and P. Vuorimaa, "System software for digital television applications". the IEEE International Conference on Multimedia and Expo, Tokyo, Japan, August 22-25, 2001
- [18] P. Vuorimaa, "Digital television service architecture". Proceedings of IEEE International Conference on Multimedia and Expo, ICME2000, New York City, NY, USA, July 30—Aug. 2, 2000
- [19] Jose C. Lopez-Ardao, Candido Lopez, Alberto Gil, Rebeca Diaz, Ana Fernandez, Manuel Fernandez, Andres Suarez, and Javier Munoz, "A MHP Receiver over RT-Linux for Digital

TV”. IEEE Region 8 International Symposium on Video/Image Processing and Multimedia Communications, 2002

- [20] 张俊。中间件——数字电视软件平台。中国广电技术论文，2004
- [21] 赵季伟。采用中间件平台开展数字电视交互业务的实践分析。中国广电技术论文，2004
- [22] 王斌。浅谈互动电视中间件技术及其选择。中国广电技术论文，2004
- [23] 陈焕经，徐朝晖。数字电视中间件技术发展浅析。中国广电技术论文，2004

个人简历 在读期间发表的学术论文与研究成果

个人简历:

沈小飞, 男, 1982 年 11 月生。

1999 年 9 月~2003 年 7 月, 同济大学计算机专业, 获学士学位。

2003 年 9 月~2006 年 3 月, 同济大学计算机科学与技术系, 攻读硕士学位。

已发表论文:

[1] 沈小飞。基于嵌入式操作系统的面向方面编程的研究。微计算机应用, 2006 年 1 月