

ADI 处理器实用丛书

# Blackfin

## 双核处理器与应用开发

◎ 唐 建 编著

### 本书特色：

- 较系统地介绍了 ADSP-BF561 的体系结构、应用开发模式、uClinux 的移植、VDK 的原理与使用
- 较深入地介绍了视频解码算法 p264 中的一些代码优化实例
- 介绍了基于双核的客户端 / 服务器应用程序、PGO 和 IPA 工具的使用、开发并行程序的方法。



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

ADI 处理器实用丛书

# Blackfin双核处理器与应用开发

唐 建 编著

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

## 内 容 简 介

ADI 公司的 ADSP-BF561 处理器是一款高性能的对称双核嵌入式媒体处理器，可以在一个核上运行嵌入式操作系统，而在另一个核上运行数字信号处理算法，或两个核一起运行一个复杂的算法，且开发方便。本书全面介绍了 ADI 公司的 VisualDSP++ 开发环境的使用要点，系统地说明了 ADSP-BF561 的体系结构，然后讲解了在该处理器上移植视频编解码算法 H.264/AVC 的基本思路并给出了视频解码算法 p264 中的一些代码优化实例，介绍了基于该处理器的几种开发模式和优化功能。最后两章分别介绍了在 ADSP-BF561 上移植嵌入式操作系统 uClinux 的方法、VDK 的原理与使用。

本书适合电子、自动控制、通信、信息处理类专业高年级本科生、研究生，以及相关专业工程技术人员使用。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

### 图书在版编目 (CIP) 数据

Blackfin 双核处理器与应用开发 / 唐建编著. —北京: 电子工业出版社, 2010.8  
(ADI 处理器实用丛书)  
ISBN 978-7-121-11565-3

I. ①B… II. ①唐… III. ①数字信号—信号处理—微处理器 IV. ①TN911.72

中国版本图书馆 CIP 数据核字 (2010) 第 154724 号

策划编辑: 竺南直

责任编辑: 刘 凡

印 刷: 北京市李史山胶印厂

装 订:

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 25.25 字数: 646.4 千字

印 次: 2010 年 8 月第 1 次印刷

印 数: 4000 册 定价: 48.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 [zlts@phei.com.cn](mailto:zlts@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

服务热线: (010) 88258888。

# 序 言

这些年，在与电子技术领域的工程师、学者以及大学师生交流的时候，他们的聪明才智和创新能力给我留下了深刻的印象。而他们所做的设计和项目，无不让我感觉到中国工程师队伍成长之快和中国电子行业巨大的发展潜力。同时，他们的经历和成功，也带给了我很多思考。

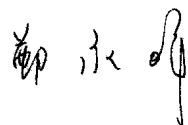
ADI 在模拟和数字信号领域中已经发展了 40 多年。在这几十年间，我们不断推动技术的创新和进步，不断提高相关领域的各类产品性能以满足客户的广泛需求，包括消费类、通信、医疗、运输和工业等方面。令人欣慰的是，截至 2009 年，ADI 已经拥有遍布世界各地的 60000 余家客户。而通过大学计划、培训、研讨会等活动所积累起来的资源更是不计其数。如何让我们的客户，让 ADI 技术产品的使用者和爱好者，真正准确、有效、快捷地掌握相关知识与设计技巧，是我们需要考虑的，也是我们为所有用户提供的非常重要的服务之一。

经过多年的运行和完善，ADI 已经拥有了一整套对中国工程师及在校工科类学生的培养计划，如每年一届的中国大学创新设计竞赛，在高校建立的联合实验室，各类线上线下的研讨会，还有在多个城市开展的高水平培训课程等。这些计划架起了 ADI 与用户之间最直接、最有效的沟通桥梁。同时，为了使更多的电子技术领域从业者和爱好者了解数字信号处理和电子产品设计理念，我们还邀请了业内具有较深影响力的专家、学者、教授共同编写并出版一套基于 ADI 模拟和数字产品的应用技术丛书。

丛书详细介绍了 ADI 产品在医疗电子、通信、工业仪器仪表、汽车电子等行业的应用，以理论与实际案例相结合的方式为读者们讲解了世界先进处理器的设计与使用。

丛书的出版凝聚了来自清华大学、西安电子科技大学、青岛理工大学、解放军理工大学、厦门大学、天津大学、黑龙江大学、中国科学技术大学、辽宁工业大学等多所院校教师丰富的经验和智慧。在此，感谢他们对 ADI 出版计划的大力支持。同时，也感谢电子工业出版社的竺南直博士对本丛书的出版所作出的贡献！

衷心希望能得到读者朋友的意见反馈，在你们提出的问题和建议下，我们将不断完善 ADI 丛书，不断完善 ADI 的产品和技术，与客户们一起共同开拓中国市场。



ADI 公司亚太区副总裁

# 前 言

信息处理与控制是当前应用非常广泛的领域。随着应用的复杂化和实时性等要求，系统设计中迫切需要高性能的数字信号处理器（DSP）功能模块和微控制器（MCU）功能模块。例如在手机系统中，基带通信、语音编码处理等复杂算法是由 DSP 模块实现的，用户界面、通信协议栈等管理功能是由 MCU 模块实现的。更重要的是，在将来更多更复杂的应用中，DSP 功能和 MCU 功能的结合将更加紧密，甚至更加难以明确划分。

DSP 是与数字信号处理算法紧密相关的一种处理器。Analog Devices Incorporation (ADI) 是主要的 DSP 芯片供应商之一。近年来 ADI 推出了 Blackfin 系列多款不同配置的嵌入式处理器产品，采用了结合 DSP/MCU 双功能的微信号架构，设计了增强多媒体算法效能的技术，实现了信号处理、媒体处理与控制功能的汇聚，并提供了丰富的应用接口，能够胜任众多的应用要求。Blackfin 处理器具有很好的性价比，目前已经在多媒体处理、通信系统、控制系统、生物医疗、汽车、航天等众多领域中得到了广泛的应用。

针对高端应用，ADI 推出了对称双核嵌入式媒体处理器 ADSP-BF561。该处理器有两个相同的内核，可以运行相同的代码，共同参与密集计算；具有更多的程序设计模式选择，以适应不同性能、成本和功耗的要求。BF561 也支持 RISC/DSP 编程模型，可以满足对非对称架构的应用需求。充分理解 BF561 双核处理器的特点，并掌握其软件设计与编程是非常重要的。

本书主要介绍 BF561 处理器的体系结构和编程方法，共分为 11 章。

绪论介绍了数字信号处理的发展趋势与需求，数字信号处理器的发展与应用，非对称处理器（AMP）的特点，以及 Blackfin 对称双核处理器（SMP）架构的优势。

第 1 章概述了 BF561 处理器的设计理念和架构特色，包括内核与存储器体系结构、时钟、电源、引导模式、事件处理、外设，然后比较详细地介绍了 VisualDSP++ 5.0 开发环境，包括工程开发、代码开发工具、处理器工程所涉及的诸多概念，为程序开发提供了比较全面的基本知识点。

第 2 章讲解了计算单元，包括寄存器文件、数据类型、算术逻辑单元、乘累加器、桶式移位器、数据地址产生器，通过这些内容便于使读者深入了解处理器计算功能的基本特点，并认识一些基本的编程方法。

第 3 章讲解了 BF561 处理器的运行模式和程序控制，不同的处理器工作模式有不同的使用处理器资源的能力，从而可以提供资源保护；程序控制器是处理器的核心之一，高性能的中断机制及其管理模块控制着用户希望的程序流程，这部分内容对程序设计起着关键作用；处理器运行模式与电源管理关系密切，BF561 的动态电源管理功能模块可以改变处理器的时钟频率，也可以控制工作电压，从而满足用户希望的功耗开销。

第 4 章讲解了 BF561 的存储器机制和使用特点。BF561 支持分级的存储器：快速的小容量的片上核内 L1 存储器、稍慢的容量稍大的片上核外 L2 存储器、慢速的但容量较大的片外 L3 存储器；每个核都有自己的 L1 存储器，两个核共享 L2 和 L3 存储器；部分 L1 存

存储器可配置为 Cache, L2 和 L3 存储器可配置成被 Cache 的页面; 基于页面的存储器管理单元 (MMU) 提供了任务间存储器和 I/O 资源分配的灵活管理; 存储器具有流水线互锁、原子操作指令等多种功能。掌握这些内容对于有效地使用存储器非常关键, 对于复杂的程序设计极为重要。

第 5 章讲解了 BF561 的片上总线和 DMA。片上总线决定了数据在系统中如何有效地传输, 对高效率的程序设计影响很大, 譬如要合理地设计才能提高总线的利用率。DMA 是关键模块之一, 它与总线密切相关。BF561 的高性能 DMA 功能模块为复杂应用环境中的音视频采集、数据传输和其他通信所涉及的数据传输提供了强有力的支持。多种 DMA 传输模式提供了灵活性。DMA 的性能优化与软件管理是使用好 DMA 的关键问题, 影响到整个系统的全局性能。

第 6 章讲解了 BF561 的片上外设, 串行外设接口 (SPI) 可以连接各种 SPI 兼容设备; 两个并行外设接口 (PPI) 可以连接并行 A/D、D/A、视频编解码器和其他通用外设; 两个同步串行口 (SPORT) 为各种标准的数据转换器和编解码器等串行设备提供了无缝硬件接口, 并支持多处理器系统中处理器间的直接互连; UART 端口支持灵活地异步串行通讯, 并在嵌入式系统设计中通过它建立应用程序和驱动程序的调试界面; 可编程标志引脚 (PF) 可作为通用 I/O 引脚, 或配置为其他外设所需要的引脚; 定时器包括通用定时器、内核定时器和看门狗定时器, 这些定时器可以实现脉宽调制模式、脉宽计数和捕获模式、外部事件计数模式、产生周期性的定时中断等多种功能。

第 7 章介绍了当前比较流行的视频编解码器 H.264/AVC, 其卓越性能使得基于该视频编解码器规范的具体方案在嵌入式视频编解码应用中日益受到重视; 然后讲解了在 BF561 上移植 H.264/AVC 的一些方法和建议, 以及移植视频解码器 p264 的部分代码优化实例。作为视频应用领域中的嵌入式媒体处理器, BF561 有特殊的视频处理单元以及高效的视频处理指令, 为视频编解码算法的移植提供了重要的软硬件支持。本章讲解的这些代码对于移植视频编码器 x264 以及其他编解码器也是可借鉴的。

第 8 章讲解了 BF561 的编程模式和优化功能。BF561 可以实现单核应用、双核单独应用、双核单一应用三类程序设计开发模式。在双核单独应用开发模式下, 资源共享是粗粒度的且由开发者管理; 在双核单一应用开发模式下, 资源共享是细粒度的且受链接器管理, 因此它是最有效的开发方式, 同时也有着更多的细节问题需要考虑, 包括一些限制。接下来讲解了基于该处理器的客户端——服务器程序设计方法、双核开发模式中剖析引导优化 (PGO) 和程序间分析 (IPA) 工具的使用以及文件属性和同步函数的使用。考虑到 BF561 的双核特性使得它具有计算机领域中多机并行计算系统里的共享存储的对称多处理机 (SMP) 的特性, 因此可以借鉴并行算法中的一些理论 (如 PCAM 设计方法学) 来解决应用中的一些问题, 又由于 BF561 有两个对称的内核, 所以在其上开发并行计算程序相对简单, 本章最后以经典的快速傅里叶变换 (FFT) 为例展示了开发并行算法的基本方法。

第 9 章讲解了在 BF561 上开发嵌入式操作系统 uClinux 所涉及的一些基本操作。由于 BF561 的双核特性, 可以让一个核主要运行嵌入式操作系统, 而让另一个核主要运行视频编解码算法等应用, 从而发挥该处理器的潜在能力。uClinux 是一种嵌入式 Linux, Linux 的优良特性使得嵌入式 Linux 得到了广泛的应用, uClinux 在架构上与标准 Linux 完全一致, 具有 Linux 的绝大部分优良特性, 并主要是针对无 MMU 的处理器而设计的, 目前支持多

种处理器，包括 Blackfin 处理器。限于篇幅，本章主要涉及工具链的安装、U-Boot 引导程序的编译与烧写、内核与文件系统的编译与下载、编译与调试应用程序的 NFS 方法，以及串口通信服务、TFTP 服务的安装与使用。

第 10 章讲解了 ADI VDK 的基本原理与使用。VDK 是 ADI 提供的集成在 VisualDSP++ 开发环境中的实时操作系统，是为有效使用 ADI 处理器而特别设计的。通过 VDK，有助于开发结构化的应用软件，便于用户开发更加复杂的程序。限于篇幅，本章重点介绍了线程、线程调度、用于同步和通信的信号（信号量、互斥、消息、事件与事件位、设备标志）机制，最后介绍了 VDK API 函数的使用方法，并给出了几个例子以加深读者对部分 VDK API 函数的理解。

本书的编著过程中涉及较多的翻译工作，最后经过压缩整理而成，若译文有不当之处，敬请读者指出，作者邮箱是 jtang@ustc.edu.cn。在有的地方保留了一些细节，主要是考虑到初学者自学的需要，若有取舍不当，或有疑惑之处，也请告知。由于时间和水平有限，关于代码优化与程序设计，本书未能涉及更多，也没有更加全面地讲解更多的程序优化。

参加本书编著的有实验室学生郑建峰、唐润鸿、张传金、金发逊、陶为、周芳燕、丁聪敏等人，对他们表示衷心感谢！感谢电子工业出版社的竺南直先生，ADI 公司的景妮女士、高威先生、孔庆峰先生、刘利先生，以及《ADI 处理器使用丛书》编委会的同事们，大家在一起认真讨论了书稿内容，并提出了宝贵意见。

借此机会，向北京亿旗创新科技发展有限公司的吴勇先生表示衷心感谢，他对我们实验室提供了很大的帮助和支持。

唐 建

2010 年 4 月于中国科学技术大学

# 目 录

绪论	1
第 1 章 概述	15
1.1 内核与存储器体系结构	15
1.1.1 内核体系结构	15
1.1.2 存储器体系结构	16
1.2 时钟、电源、引导模式、事件处理	18
1.2.1 时钟信号	18
1.2.2 动态电源管理	18
1.2.3 引导模式	19
1.2.4 事件处理	20
1.3 ADSP-BF561 外设	20
1.4 VisualDSP++ 5.0	24
1.5 工程开发	28
1.6 代码开发工具	33
1.7 处理器工程	38
第 2 章 计算单元	46
2.1 寄存器文件	46
2.1.1 寄存器文件	46
2.1.2 使用计算状态	49
2.2 数据类型	49
2.2.1 使用数据格式	49
2.2.2 使用乘法器整数和小数格式	52
2.3 算术逻辑单元	54
2.3.1 ALU 操作	55
2.3.2 ALU 数据流详细情况	58
2.3.3 ALU 除法支持与视频操作	60
2.4 乘累加器	60
2.4.1 乘法器操作	60
2.4.2 乘法器数据流描述	62
2.5 桶式移位器	64
2.6 数据地址产生器	67
2.6.1 数据地址产生器功能与寄存器	67



2.6.2	DAG 的寻址	69
<b>第 3 章</b>	<b>运行模式与程序控制</b>	<b>75</b>
3.1	运行模式	75
3.1.1	用户模式	76
3.1.2	监控模式	77
3.1.3	仿真模式	78
3.1.4	复位与空闲状态	78
3.1.5	系统复位和上电配置	79
3.1.6	引导方式	82
3.2	程序控制	83
3.2.1	程序控制器相关寄存器	83
3.2.2	指令流水线	85
3.2.3	分支和程序设计	86
3.2.4	循环和程序设计	88
3.2.5	事件和控制	89
3.2.6	内核事件控制寄存器	96
3.2.7	事件向量表	97
3.2.8	中断服务	101
3.2.9	中断的嵌套	101
3.2.10	异常处理	103
3.2.11	使用性问题	105
3.3	动态电源管理	107
3.3.1	时钟	107
3.3.2	动态电源管理控制器	110
<b>第 4 章</b>	<b>存储器</b>	<b>117</b>
4.1	存储器结构概述	117
4.2	L1 指令存储器	118
4.2.1	指令存储器	118
4.2.2	L1 指令 Cache	120
4.2.3	指令 Cache 管理	122
4.2.4	指令测试寄存器	123
4.3	L1 数据存储器	124
4.3.1	L1 数据存储器	124
4.3.2	L1 数据 Cache	126
4.3.3	数据测试寄存器	129
4.4	片上 L2 存储器	129
4.5	存储器保护与属性	130
4.5.1	存储器管理单元	130

4.5.2	存储器页面	131
4.5.3	CPLB 管理	131
4.5.4	MMU 应用	132
4.5.5	有关寄存器	133
4.6	存储器有关操作	134
4.6.1	加载/存储操作	134
4.6.2	其他存储器有关操作	136
4.7	描述存储器的一些术语	137
<b>第 5 章</b>	<b>片上总线与 DMA</b>	<b>139</b>
5.1	片上总线	139
5.1.1	内核接口	139
5.1.2	系统接口	140
5.1.3	外设与外部访问总线 (PAB、EAB)	144
5.1.4	DMA 相关的总线	145
5.2	直接存储器访问	147
5.2.1	DMA 概述	147
5.2.2	DMA 和存储器 DMA MMR	148
5.2.3	DMA 操作流	156
5.2.4	2D DMA	159
5.2.5	存储器 DMA	160
5.2.6	DMA 性能优化	161
5.2.7	DMA 的软件管理	165
5.2.8	DMA 错误	169
<b>第 6 章</b>	<b>片上外设</b>	<b>171</b>
6.1	SPI 兼容端口控制	171
6.1.1	SPI 功能	171
6.1.2	接口信号	171
6.1.3	SPI 寄存器	173
6.1.4	SPI 通用操作	177
6.1.5	DMA	180
6.2	并行外设接口	182
6.2.1	并行外设接口功能	182
6.2.2	PPI 存储器映射寄存器	183
6.2.3	ITU-R 656 模式	187
6.2.4	通用 PPI 模式	190
6.2.5	DMA 操作	195
6.2.6	数据传送方案	195
6.3	串口控制器	196

6.3.1	串口控制器功能	196
6.3.2	SPORT 寄存器	199
6.3.3	立体声串行操作	207
6.3.4	多通道操作	208
6.3.5	支持 H.100 标准协议	212
6.4	UART 端口控制器	213
6.4.1	串行通信	213
6.4.2	UART 控制和状态寄存器	213
6.4.3	Non-DMA 与 DMA 模式	217
6.4.4	IrDA 支持	218
6.5	可编程标志	220
6.5.1	可编程标志功能	220
6.5.2	可编程标志存储器映射寄存器	222
6.6	定时器	226
6.6.1	通用定时器寄存器	226
6.6.2	通用定时器的使用	230
6.6.3	内核定时器	242
6.6.4	看门狗定时器	243
<b>第 7 章</b>	<b>H.264/AVC 标准简介及部分程序分析</b>	<b>245</b>
7.1	H.264/AVC 标准简介	245
7.1.1	H.264/AVC 的基本框架	245
7.1.2	H.264/AVC 的新特性	247
7.2	H.264/AVC 在 ADSP-BF561 上的移植	248
7.2.1	源码简介	248
7.2.2	在 ADSP-BF561 上移植 p264 概述	249
7.3	变换和量化过程的优化	252
7.3.1	H.264/AVC 的整数变换及量化原理 <sup>[5]</sup>	252
7.3.2	反量化过程的优化	254
7.3.3	DCT 反变换的优化	258
7.3.4	Hadamard 变换及其优化	264
7.4	非整数样点插值过程的优化	267
7.4.1	非整数样点的插值过程	267
7.4.2	插值过程的优化	270
<b>第 8 章</b>	<b>ADSP-BF561 编程模式与优化功能</b>	<b>275</b>
8.1	编程模式	275
8.1.1	单核应用	275
8.1.2	双核单独应用	276
8.1.3	双核单一应用	279

8.1.4	双核应用中的一些考虑	282
8.2	双核程序示例分析	283
8.2.1	客户端-服务器应用示例	283
8.2.2	双核系统中的 PGO	285
8.2.3	程序间分析和文件属性	287
8.2.4	同步函数	290
8.3	双核 FFT 编程设计	290
8.3.1	基 2 DIT FFT 基本原理与程序设计	290
8.3.2	并行计算设计方法学	292
8.3.3	基于多 DSP 系统的 FFT 并行计算	293
8.3.4	基 2 DIT FFT 双核单一应用并行程序设计	295
<b>第 9 章</b>	<b>uClinux 系统开发</b>	<b>298</b>
9.1	uClinux 概述	298
9.1.1	历史	298
9.1.2	uClinux 的内存管理	300
9.1.3	uClinux 进程管理特点	301
9.1.4	uClinux 的开发	302
9.2	ADSP-BF561 uClinux 快速指南	303
9.2.1	基本环境	304
9.2.2	编译、烧写、下载	304
9.3	Ubuntu 基本服务配置	308
9.3.1	串口通信服务配置	308
9.3.2	TFTP 服务配置	309
9.3.3	NFS 服务配置	312
9.4	在 Ubuntu 中安装 Blackfin 工具链	314
9.4.1	工具链概述	314
9.4.2	可执行文件格式	315
9.4.3	工具链安装	317
9.4.4	LDR 与 ldr-utils	320
9.4.5	在 Windows 中安装 Blackfin 工具链	323
9.5	U-Boot Bootloader	323
9.5.1	U-Boot 基本知识	323
9.5.2	U-Boot 移植与编译	329
9.5.3	U-Boot 下载与更新	334
9.6	uClinux-dist	336
9.6.1	文件系统	336
9.6.2	uClinux-dist 镜像文件	338
9.6.3	内核更新管理	340
9.6.4	移植 Linux 到用户目标板	343

9.6.5	内核编译	346
9.7	利用 NFS 调试程序	351
<b>第 10 章</b>	<b>VDK 基本原理与使用</b>	<b>354</b>
10.1	VDK 简介	354
10.2	线程	356
10.3	线程调度	360
10.4	信号	363
10.4.1	信号量	363
10.4.2	互斥	364
10.4.3	消息	365
10.4.4	多处理器消息传递	366
10.4.5	事件和事件位	370
10.4.6	设备标志	372
10.5	中断服务程序	372
10.6	VDK API 参考	374
10.7	典型 API 简介与应用	377
10.7.1	信号量 API 简介与应用	377
10.7.2	事件和事件位 API 简介与应用	381
10.7.3	消息的 API 简介与应用	384

# 绪 论

数字信号处理是一门发展迅速的重要学科，涉及面越来越广泛，理论越来越深入，算法越来越复杂。数字信号处理器是实现数字信号处理的首选器件，其体系结构在设计上考虑了数字信号处理一般算法的特点。近年来音频、视频媒体处理方面应用普及程度非常高，同时也对操作控制方面提出了更高的要求，产生了将计算和控制有机融合在一个芯片上的趋势。随着嵌入式芯片技术和数字信号处理器技术的发展，新型数字信号处理器在实现复杂音视频媒体处理算法基础上，提供了完成事务管理的控制功能，于是出现了一类嵌入式媒体处理芯片。为了实现更加复杂的计算和灵活的控制，以及更加有效的应用模式，又推出了多核嵌入式媒体处理器，其中对称多核媒体处理器具有重要特色，本书介绍的ADSP-BF561就是一款对称双核嵌入式媒体处理器。

## 0.1 数字信号处理

### 0.1.1 数字信号处理基本概念

世界上有各种各样需要研究的信号。信号主要是以物理量形式存在的，在数学上用函数来表示。信号可分为声、光、电等各种类型，常见的物理信号有语音信号、音乐、各种视频、生理信号、医学成像、遥感遥测信号、GPS信号、各种数据等，为了便于获取和传输，通常将其他形式的信号转换为电信号。

人们用感知认识世界、获取信息。信号是信息的载体，信息是从信号中获取的新知识或对某个方面发布的新要求。人们常说信息时代就体现了信息的重要性。信息对政治、军事、经济等举足轻重，也影响到个人具体的工作、生活、医疗、娱乐等各个方面，人们对信息的依赖越来越大。

人们会通过各种手段不断改变世界。信息处理是对信号进行处理的简称，结合各种各样的信息处理技术，产生了积极有效的作用。通过从信号中获取信息，并进行加工处理，可以得到想要的结果（如通过滤波得到更适合的信号），或最终做出某种判决（如说话人确认中的“是”与“不是”）。俗话说，聪明聪明，耳聪目明，可见听觉和视觉的重要性。从听觉和视觉获取的信息是人们感知、认识世界的主要渠道，更重要的是人们的主要交流方式。关于这方面的研究也最普遍、最丰富，其应用也更多。就语音来说，相关产业有电话通信，包括后来发展起来的移动电话，还有语音识别、说话人辨认、说话人确认、文语转换等应用。对于图像和视频，早期有拍照、电视等重要应用，目前指纹识别已在很多地方用上，未来人脸识别、虹膜识别、步态识别、人体跟踪、视频理解、手语自动识别等，都是可以应用的方面。

数字信号处理（Digital Signal Processing, DSP）是20世纪60年代前后发展起来的一门新兴学科，利用计算机或专用数字处理设备对信号进行采集、滤波、变换、估值（特征提取）、增强、识别、压缩等处理，以得到符合人们需要的结果，它涉及模拟信号、采样、

数字信号、信号特征量、系统分析与设计、系统实现、测试等多个方面。它涉及众多学科而又广泛应用于众多领域，把许多经典理论体系作为自己的理论基础，同时又成为一系列新兴学科的理论基础，因此它与其他交叉学科相结合并渗透到了各个应用领域中<sup>[1]</sup>。

现代信号处理理论，包括模糊逻辑、神经网络、小波变换、自适应计算、智能计算等，是非常重要的研究方向，一些应用中可能涉及非常复杂的算法和大量的计算（如同时涉及语音信号、图像信号、视频信号的处理），增加了实时处理难度。实时是指为完成某项任务而对算法的实现时间有一定要求，如果算法的处理时间可以满足这个时限要求，就认为该算法是可以实时实现的。现代信息处理的一个重要发展方向是分布式、并行计算，并借助于大型并行计算机系统推广了其应用，在一些情况下，基于数字信号处理器（Digital Signal Processor, DSP）芯片的阵列处理系统也得到了应用。随着大规模集成电路（VLSI）和处理器技术的不断发展，通用计算机和 DSP 芯片都朝着多核发展，产生了多种新型处理器，也必将进一步推动现代信息处理的巨大发展和广泛应用。

### 0.1.2 数字信号处理算法基本特点

数字信号处理中一个最常见也是最重要的操作就是乘累加（MAC），它在求系统响应、计算信号相关函数等方面广泛应用。根据线性时不变离散时间系统的单位脉冲响应，系统响应可采用卷积和来计算：

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \quad (0-1)$$

式中， $h(n)$ 、 $x(n)$ 、 $y(n)$  分别是系统的单位脉冲响应、输入序列、输出序列。如果系统是用差分方程描述的，其响应计算为

$$y(n) = -\sum_{i=1}^N a_i y(n-i) + \sum_{i=0}^M b_i x(n-i) \quad (0-2)$$

式中， $a_i$  和  $b_i$  是系统结构的参数。

对于信号  $x(n)$  和  $y(n)$ ，它们的互相关函数  $r_{xy}(m)$  表示了二者在时刻  $m$  上的相似性， $r_{xy}(m)$  计算为

$$r_{xy}(m) = \sum_{n=-N}^N x^*(n)y(n+m) \quad (0-3)$$

在很多场合需要计算信号的频谱，就要计算时间序列的离散傅里叶变换（DFT）：

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn} = \sum_{n=0}^{N-1} x(n)\exp(-j(2\pi kn/N)) \quad (k=0,1,\Lambda,N-1) \quad (0-4)$$

在 H.264 等视频压缩算法中，离散余弦变换（DCT）是一个基本操作：

$$X(k, l) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x(n, m) \cos \frac{2n+1}{2N} k\pi \cos \frac{2m+1}{2M} l\pi \quad (0-5)$$

$(k=0,1,\Lambda,N-1 \quad l=0,1,\Lambda,M-1)$

数字信号处理中还有大量的矩阵乘法等运算，这些操作中有大量的 MAC 操作。同时

算法中普遍存在循环操作，它是一种非连续流执行方式，对系统性能影响较大。

### 0.1.3 数字信号处理实现方法

数字信号处理是围绕理论、实现、应用发展起来的，理论发展推动了应用发展，反过来，应用又促进了理论的提高，而实现则是理论和应用之间的桥梁。信号处理的基础理论和应用的深入研究开始于 1822 年傅里叶级数理论提出之后，到 20 世纪 60 年代初期基本理论已经比较成熟，重点是研究各种应用算法和快速算法，涉及到离散时间信号与系统分析、各种变换理论、DFT 快速算法、数字滤波器分析与设计等。1965 年 Cooler 和 Tukey 提出的快速傅里叶变换（FFT）使傅里叶分析速度提高了数百倍，从而为数字信号处理应用奠定了基础，标志着数字信号处理的开端。

数字信号处理的实现方法经历了一个较长的发展过程。20 世纪 60 年代数字信号处理实现技术处于实验室阶段，计算速度不高，这期间主要是采用计算机模拟方法研究数字信号处理、变换和数字滤波等问题，一般要花费很多时间才能处理少量信号，无法像模拟信号系统那样做到实时处理。70 年代后研制出了 FFT 和数字滤波器硬件，最初是用多个集成电路（IC）芯片实现数字信号处理中的一个基本运算部件（如加法器、乘法器、延迟器等），后来发展到一个芯片内集成多个运算部件，同时采用了微处理器技术，实现了可编程和通用性，但内部结构主要是为通用计算和控制而设计的，在许多场合都难以实现实时处理。进入 80 年代，电子计算机、VLSI 技术有了较大发展，1982 年美国得州仪器（Texas Instruments, TI）公司研制出了第一片 DSP 芯片 TMS320C10，使得在音频范围内实现了实时处理，之后各种 DSP 芯片不断问世并取得了广泛应用，以后 DSP 更多的就是指数字信号处理器。

数字信号处理的实现方法一般有以下几种：① 通用计算机上软件实现；② 通用计算机系统中专用加速处理机实现；③ 专用 DSP 芯片实现；④ 通用单片机实现；⑤ FPGA 实现；⑥ 通用可编程 DSP 实现。在早期，第 1 种方法速度较慢但使用方便，一般用于算法模拟，现在通用处理器速度可以很快，但还是不利于小型化；第 2 种方法专用性强，不便于系统独立运行；第 3 种方法也有专用性强的特点，比较适合某些特殊场合，如专用于 FFT、数字滤波、卷积、相关、数字图像信号处理等算法的 DSP 芯片，这种芯片将相应算法在芯片内部用硬件实现，无须进行编程；第 4 种方法适用于简单的数字信号处理算法，如简单的数字控制系统；第 5 种方法成本较高，编译时间较长；第 6 种方法最适合实现数字信号处理应用，具有丰富的数字信号处理的软件和硬件资源，适用于复杂的数字信号处理算法，可满足大多数应用中的实时约束，也便于小型化。因此 DSP 芯片为数字信号处理提供了很好的实现平台，现在的高性能 DSP 芯片可以实现更加复杂的现代数字信号处理（自适应滤波、卡尔曼滤波、同态滤波等），从而极大地拓展了数字信号处理和数字信号处理器的应用。

### 0.1.4 数字信号处理系统及特点

对信号进行处理从而获取信息的过程是基于系统来实现的，系统是一个比较大、比较泛的概念，与本书相关的有控制系统、滤波器、通信系统等。对系统的研究有两个主要方面：系统分析和系统设计，系统分析是系统设计的基础，系统设计是为了更好地实现任务、解决问题。系统的实现方式与系统的研究紧密相关，实现方式上先后经历了分别以模拟和



数字为主要的时代，后来发展起来的以数字为主的系统是为了避免出现模拟系统的缺陷。

模拟系统主要是用电阻、电容、电感、运放等组成的系统，如模拟电话、模拟电视机、模拟滤波器、模拟控制系统等。数字系统现在一般指用单片机、计算机、数字信号处理器等数字处理器件为主组成的系统，如数字电视机、数字机顶盒、数字手机、数字滤波器、数字控制系统等。

数字信号处理系统既可以处理数字信号，也可以处理模拟信号，一般的系统框图如图 0-1 所示。模拟输入信号可以是来自扬声器的语音信号、来自摄像机的图像信号或模拟视频信号等。输入信号在带限滤波和采样后进行 A/D 变换得到数字信号。数字处理器件对输入的数字信号进行处理，如进行一系列的 MAC 操作。处理后的数字信号再经过 D/A 变换转换为模拟样值，最后进行内插和平滑滤波以得到连续的模拟波形。如图 0-1 所示的数字系统框图是一般性的，实际系统中可以省去一些子系统。例如对于说话人确认系统，其输出为“是”或“不是”，该系统就不需要后续的数模转换和平滑滤波子系统。

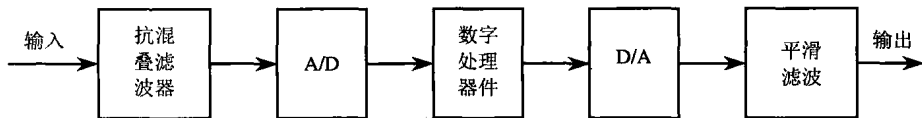


图 0-1 数字信号处理系统框图

相对于模拟系统，数字系统有如下优点：① 接口方便，符合工业标准的数字系统或设备在设计上都考虑了兼容性，因此其连接要比模拟系统接口容易得多。② 编程方便，数字器件一般具有可编程性，便于开发过程中灵活方便地对软件进行修改和升级。③ 稳定性好，数字系统受环境温度以及噪声的影响较小，因此可靠性高。④ 精度高，数字器件一般可实现 16-bit、24-bit、32-bit 数据，甚至更高，因此数字系统的精度可以达到很高。⑤ 可重复性好，数字系统基本上不受元器件参数变化的影响，便于调试和大规模生产。⑥ 集成方便，数字系统中的数字部件一般都有高度规范性，因此便于大规模集成。数字系统的突出优点使其得到了广泛应用。基于 DSP 芯片的系统设计的一般过程描述如下：

(1) 事前准备阶段。在设计之前首先必须根据具体应用确定系统的性能指标、信号处理的各种具体要求。

(2) 算法模拟阶段。根据系统的应用要求，可能需要对输入信号进行适当处理、抽取有效的特征、进行优化计算才能得到理想结果。一个关键是要研究有效算法，并用高级语言快速验证。例如，视频压缩算法要考虑到编码速率和编码质量，不同的应用有不同的要求，因为二者具有矛盾性。

(3) 系统设计阶段。包括硬件设计和软件设计两个方面。硬件设计上要根据算法运算量、运算精度、接口、系统成本、功耗等要求选择器件并设计电路。软件设计主要是根据算法和所选择的器件编写程序，为缩短软件开发周期和提高程序可读性、移植性、执行效率，一般采用高级语言和汇编语言混合编程。

(4) 系统调试阶段。硬件调试一般采用硬件仿真器、示波器、逻辑分析仪等进行，简单情况下可借助于普通工具。软件调试一般要借助于开发工具，如软件模拟器、开发系统等。算法调试一般采用比较法，在输入相同情况下，系统调试结果应与高级语言算法模拟结果的输出相一致。

(5) 运行调试阶段。系统开发一般是一个反复调试的过程。实际运行状况要比实验室