

邓 琛 主编

陈益平 滕旭东 副主编



DSP

芯片技术 及工程实例

清华大学出版社

DSP

芯片技术及工程实例

邓 琛 主编

陈益平 滕旭东 副主编

清华大学出版社
北京

内 容 简 介

本书从工程的角度对 DSP 系统所涉及的硬件和软件技术进行了系统的介绍,并从初学者的角度入手,总结了数字信号处理技术与高性能微处理器融合的应用要求。全书共分 8 章,第 1~2 章以 DSP 系统应用开发为核心,从 DSP 入门开始,介绍从初次接触 DSP 到成为 DSP 应用工程师所需要的基本理论知识、硬软件开发平台和工具以及 DSP 系统设计的流程。第 3~4 章以 C54x 系列 DSP 器件为对象,介绍 DSP 芯片的硬软件架构、特性和使用方法。第 5 章重点介绍 DSP 应用程序的设计方法。第 6 章介绍 C54x DSP 片内外设的设计和应用方法。第 7~8 章列举了工程开发实例和应用实验指导,并给出较为详细的设计过程、参考电路和参考程序,便于读者边学习边实践,加快掌握 DSP 应用的进程。

DSP 技术是一种实用性很强的技术。作者在编写本书时力求理论联系实际,在介绍 DSP 芯片基本结构的基础上,着重介绍它的应用技术。在选材上力求循序渐进、由浅入深、通俗易懂,使读者能够尽快地理解和掌握 DSP 芯片的基本概念和应用,并引导读者作进一步思考。

本书是作者在多年从事 DSP 本科教学和 DSP 系统的研究开发的基础上,结合当前流行的 DSP 系统设计方法及其应用发展趋势编写而成的。本书适合作为应用型高等院校的教材,也适合从事 DSP 技术开发的工程技术人员作为参考。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

DSP 芯片技术及工程实例/邓琛主编. --北京:清华大学出版社, 2010.5
ISBN 978-7-302-22428-0

I. ①D… II. ①邓… III. ①数字信号—信号处理 IV. ①TN911.72

中国版本图书馆 CIP 数据核字(2010)第 066092 号

责任编辑:张占奎

责任校对:刘玉霞

责任印制:何 芊

出版发行:清华大学出版社

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

社 总 机:010-62770175

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮 购:010-62786544

印 刷 者:北京市清华园胶印厂

装 订 者:三河市李旗庄少明装订厂

经 销:全国新华书店

开 本:185×260

印 张:20.5

字 数:517 千字

版 次:2010 年 5 月第 1 版

印 次:2010 年 5 月第 1 次印刷

印 数:1~4000

定 价:36.00 元

产品编号:032425-01

进入 21 世纪后,数字化浪潮正在席卷全球,数字信号处理器(digital signal processor, DSP)正是这场数字化革命的核心,DSP 技术已成为人们日益关注并得到迅速发展的前沿技术,其应用领域也已经向各个工程领域拓展。社会迫切需要掌握 DSP 应用技术的人才,高校虽已普遍开设 DSP 基础理论课程,但由于 DSP 架构的复杂性,使学习和掌握 DSP 应用具有一定的难度,上手较困难。我校开设“DSP 技术及应用”课程已经多年,在教学过程中感到现有的大部分 DSP 参考书籍往往针对的是一些技术开发人员,要使 DSP 初学者能够尽快入门,采用现有 DSP 的教材或技术手册显得不够。因此编写一本既适用于 DSP 教学又适用于初次涉足 DSP 领域人员的书籍显得十分重要。为此,我们联合业内工程技术人员、具有丰富 DSP 教学经验和 DSP 应用系统开发能力的教师编写了这本书,书中除了系统介绍 DSP 技术及其系统设计方法外,还对一些实际的工程实例设计及实现方法进行指导,便于读者边学边实践,加快掌握 DSP 应用的进程。

本书从工程的角度对 DSP 系统所涉及的硬件和软件技术进行了系统的介绍,并从初学者的角度入手,以清晰的知识结构为读者提供了大量的工程应用实例,使读者快速上手,具有较高的参考价值。全书共分 8 章,第 1 章主要介绍 DSP 芯片在进行数字信号处理时所做的性能优化以及在特殊应用中使用 DSP 技术所得到的好处,为 DSP 初学者介绍学习 DSP 所需要的入门知识。第 2 章从 DSP 系统设计和实现的最基本要求开始,以 TI 公司 DSP 主流产品为例,介绍 DSP 系统的基本组成、DSP 开发硬、软件平台以及开发工具。第 3 章以 C54x 系列 DSP 器件为对象,介绍 DSP 芯片的硬件结构,包括具有快速处理性能的 CPU、具有哈佛结构的存储器系统和多总线结构以及片内外设和专用电路等。第 4 章介绍 C54x DSP 灵活的寻址方式以及与硬件相配套的指令系统。第 5 章介绍 DSP 程序设计方法,包括汇编、C 语言以及混合编程的编写方法。第 6 章介绍 C54x DSP 的片内外设以及一些典型的应用。第 7、8 章从设计最基本的 DSP 应用系统着手,介绍常用的 DSP 应用系统硬、软件设计思想以及实现方案。同时也为初学者提供了一些 DSP 应用实验指导,为 DSP 系统设计师介绍了在设计 DSP 系统时应该考虑的问题。

本书内容编排新颖,既强调基本概念,又突出应用性。无论是对于 DSP 初学者,还是从事 DSP 应用开发的科研人员和工程技术人员,都是一本通俗易懂的教科书和工具书,具有较好的参考价值。

本书由邓琛任主编,陈益平、滕旭东任副主编。邓琛编写第 1、2 章,并参与编写第 3、4、6、7、8 章;刘海珊编写第 3、4 章;陈益平编写第 5 章;滕旭东编写第 6 章;李心阳编写第 7、8 章。全书由邓琛统稿。在编写本书的过程中参阅了大量的国内外参考文献以及 TI 公司原版资料,并得到了北京瑞泰创新科技有限责任公司上海办事处技术人员的大力支持,在此表示衷心感谢。

限于作者的水平,书中不足之处在所难免,恳请读者批评指正。

编者

2010 年 1 月

第 1 章 DSP 概述	1
1.1 引言	1
1.2 初步认识 DSP	2
1.2.1 DSP 的定义	2
1.2.2 数字信号处理运算概述	2
1.3 DSP 的优势	4
1.4 DSP 芯片的应用	8
第 2 章 DSP 系统开发入门	10
2.1 DSP 系统的基本组成	10
2.2 DSP 系统设计开发流程	11
2.2.1 如何着手 DSP 系统开发	11
2.2.2 算法可视化仿真开发工具	12
2.2.3 DSP 芯片的选择	12
2.2.4 DSP 系列产品	16
2.2.5 DSP 开发工具	24
第 3 章 C54x DSP 系统硬件结构	30
3.1 C54x DSP 的基本架构	30
3.1.1 C54x DSP 的基本结构图	30
3.1.2 C54x DSP 的主要特征	30
3.2 总线结构	34
3.3 存储器结构	35
3.3.1 DSP 存储器空间的划分	35
3.3.2 程序存储器	38
3.3.3 数据存储器	41
3.3.4 I/O 空间	45
3.4 中央处理器	46
3.4.1 状态和控制寄存器	46
3.4.2 算术逻辑单元	48
3.4.3 累加器	49

3.4.4	桶式移位器	51
3.4.5	乘加器	52
3.4.6	比较、选择和存储单元	54
3.4.7	指数编码器	55
3.5	地址产生单元	55
3.5.1	程序地址生成器	55
3.5.2	数据地址发生器	57
3.6	C54x 的流水线	57
3.6.1	C54x 中的流水线结构	58
3.6.2	指令执行时的流水线图	59
3.6.3	存储器的流水线操作	61
3.6.4	流水线冲突及解决方法	63
3.7	DSP 的中断系统	68
3.7.1	C54x DSP 中断系统结构	68
3.7.2	中断流程	73
3.7.3	中断向量表的编程	74
3.8	C54x DSP 外部总线	76
3.8.1	外部总线的组成	76
3.8.2	外部总线控制性能	78
3.8.3	外部总线接口时序图	81

第 4 章 DSP 指令系统 85

4.1	C54x DSP 的指令表示方法	85
4.2	DSP 数据寻址方式	86
4.2.1	立即寻址	86
4.2.2	绝对寻址	87
4.2.3	直接寻址	88
4.2.4	间接寻址	92
4.2.5	累加器寻址	98
4.2.6	存储器映射寄存器寻址	98
4.2.7	堆栈寻址	99
4.3	DSP 的程序控制	100
4.3.1	分支转移	100
4.3.2	调用	101
4.3.3	返回	103
4.3.4	重复操作指令	104
4.3.5	条件操作	107
4.4	DSP 指令系统概述	109
4.4.1	指令系统分类	113
4.4.2	算术运算类指令	115

4.4.3	逻辑操作指令·····	121
4.4.4	程序控制指令·····	123
4.4.5	加载和存储指令·····	126
第 5 章	DSP 程序设计 ·····	129
5.1	DSP 系统开发过程 ·····	129
5.1.1	DSP 系统开发过程 ·····	129
5.1.2	软件开发流程·····	130
5.2	汇编语言编写方法 ·····	132
5.2.1	DSP 汇编语言语句 ·····	132
5.2.2	DSP 汇编语言的数据形式、标识符及运算符 ·····	134
5.3	COFF 文件 ·····	136
5.3.1	段·····	136
5.3.2	汇编器对段的处理·····	137
5.3.3	链接器对段的处理·····	137
5.3.4	COFF 文件中的符号 ·····	138
5.4	常用汇编伪指令 ·····	138
5.4.1	段定义伪指令·····	139
5.4.2	常数初始化伪指令·····	139
5.4.3	段程序计数器定位伪指令·····	140
5.4.4	输出列表格式伪指令·····	141
5.4.5	文件引用伪指令·····	142
5.4.6	条件汇编伪指令·····	142
5.4.7	符号定义伪指令·····	143
5.4.8	宏及宏的使用·····	143
5.5	链接器命令文件的编写与使用 ·····	145
5.5.1	链接器命令文件常用指令·····	145
5.5.2	MEMORY 伪指令 ·····	146
5.5.3	SECTIONS 伪指令 ·····	146
5.5.4	汇编源程序的编辑、汇编和链接 ·····	148
5.6	TMS320C54x C 语言编程 ·····	156
5.6.1	C54x C/C++ 编译器支持的数据类型 ·····	157
5.6.2	C 语言的数据访问方法·····	157
5.6.3	存储器模式·····	159
5.6.4	中断函数 ·····	162
5.6.5	其他注意事项·····	163
5.7	用 C 语言和汇编语言混合编程 ·····	163
5.7.1	独立的 C 语言模块和汇编模块接口 ·····	163
5.7.2	直接在 C 程序中嵌入汇编语句 ·····	165
5.8	DSP 程序设计实例 ·····	165

第 6 章 C54x DSP 片内外设及应用	173
6.1 通用 I/O 引脚	173
6.2 C54x DSP 的定时器	174
6.2.1 定时器结构和工作原理	174
6.2.2 定时器编程	176
6.2.3 定时器应用实例	177
6.3 C54x 的主机接口	179
6.3.1 标准 HPI 接口结构框图	179
6.3.2 标准 HPI 的两种工作模式	180
6.3.3 HPI 控制寄存器	181
6.3.4 HPI 与主机的接口设计	182
6.3.5 HPI 应用实例	187
6.4 C54x 的串行口	192
6.4.1 标准同步串行口	193
6.4.2 缓冲同步串行口	197
6.4.3 时分复用串行口 TDM	202
6.4.4 多通道缓冲串行口 (McBSP)	204
第 7 章 DSP 工程开发实例	220
7.1 DSP 最小系统设计	220
7.1.1 复位电路	220
7.1.2 时钟电路	222
7.1.3 电源与省电模式设计	227
7.1.4 仿真接口	230
7.1.5 外扩存储器	231
7.2 DSP 数据采集系统设计	237
7.2.1 前向通道——AD 输入	237
7.2.2 ADS7864 芯片简介	238
7.2.3 ADS7864 与 TMS320VC5416 的接口设计	243
7.2.4 A/D 转换程序设计	246
7.3 D/A 转换器与 DSP 的接口设计	248
7.3.1 DAC7625 芯片简介	249
7.3.2 DAC7625 与 TMS320VC5416 的接口设计	252
7.3.3 D/A 转换程序设计	254
7.4 语音系统的 DSP 实现	255
7.4.1 系统组成	256
7.4.2 硬件接口设计	257
7.4.3 程序流程及部分代码	261
7.5 指纹系统的 DSP 实现	269

7.5.1	引言	269
7.5.2	硬件接口设计	271
7.5.3	程序流程及部分代码	272
第 8 章	DSP 应用实验指导	277
8.1	CCS 入门实验	277
8.1.1	CCS 的安装及设置	277
8.1.2	CCS 程序开发步骤	279
8.1.3	C/ASM 程序举例	285
8.2	I/O 扩展实验	288
8.2.1	TMS320VC5416 与键盘的连接	288
8.2.2	TMS320VC5416 与液晶显示器的连接	289
8.2.3	软件编程	292
8.3	DSP 定时器中断实验	300
8.3.1	硬件设计	300
8.3.2	程序设计	301
8.4	异步串口通信实验	304
8.4.1	硬件设计	304
8.4.2	软件编程	309
	参考文献	314

DSP 概述

1.1 引言

数字信号处理是一门涉及许多学科而又广泛应用于许多领域的新兴学科。20 世纪 60 年代以来,随着计算机和信息技术的飞速发展,数字信号处理技术应运而生,并得到迅速的发展。事实上,它已经改变了人们的生活。数字化技术大大减轻了模拟技术时代科学产品的瓶颈,它将大量信息进行数字化处理,大大提高了现实信息的处理与传送。进入 21 世纪之后,数字化浪潮正在席卷全球,DSP 正是这场数字化革命的核心,DSP 技术已成为人们日益关注并得到迅速发展的前沿技术。

世界上第一个单片 DSP 芯片是 1978 年美国 AMI 公司发布的 S2811,而 1979 年美国贝尔研究所发布的商用可编程器件 2920 是 DSP 芯片的一个主要里程碑。这两种芯片内部都没有现代 DSP 芯片所必需的单周期芯片。20 世纪 80 年代前后,陆续有公司设计出适合于 DSP 处理技术的处理器。1980 年,日本 NEC 公司推出的 μ PD7720 是第一个具有乘法器的商用 DSP 芯片。第一个采用 CMOS 工艺生产浮点 DSP 芯片的是日本的 Hitachi 公司,它于 1982 年推出了浮点 DSP 芯片。1983 年,日本的 Fujitsu 公司推出 MB8764,该芯片的指令周期为 120 ns,且具有双内部总线,从而使处理的吞吐量发生了一个大的飞跃。第一个高性能的浮点 DSP 芯片应是 AT&T 公司于 1984 年推出的 DSP32,从此 DSP 开始成为一种高性能处理器的名称。进入 20 世纪 90 年代,有多家公司跻身于 DSP 领域。美国德州仪器公司(Texas Instruments, TI)首家提供可定制 DSP(称做 cDSP)设计,可使 DSP 具有更高的系统集成度,大大缩短了产品的上市时间。同时 TI 瞄准 DSP 电子市场上成长速度最快的领域,适时地提供各种面向未来发展的解决方案。到 20 世纪 90 年代中期,这种可编程的 DSP 器件已广泛应用于数据通信、海量存储、语音处理、汽车电子、消费类音频和视频产品等,其中最为辉煌的成就是在数字蜂窝电话中的成功。

DSP 技术的发展可归结成 3 次浪潮。第一次浪潮由通信技术主导。人们可以将语音信号数字化,大大地提高了传输的效率,让数据传输速率实现了从 kbps 到 Mbps 的转变,实现了语音的数字化,同时数字蜂窝手机开始涉足多媒体应用领域。这个浪潮已经持续了十几年,目前还在进行。第二次浪潮由娱乐主导。家庭中的电视节目,人们的音乐、相片和录像已经是数字化播放与传输。当个人娱乐消费出现在便携式产品上时,时代对 DSP 提出了更高要求,功耗与价格要越低越好。当功耗与成本不断降低时,DSP 就出现在更多的新兴市场领域。如今,人们正经历第三次浪潮,它的主要应用市场将是汽车、监控、医疗与教育领域。

如今,DSP 技术已经被大多数半导体技术公司应用,不仅有“通用”DSP 芯片,也有“专用”DSP 芯片。而当一个 DSP 芯片加上了 I/O、处理器、电源管理等其他功能时,它已经成为了一颗 SoC(System on Chip),或者其芯片的封装采用了 SiP(System in Package),这些 SoC 或是 SiP 也是随着应用不断地成熟才出现的。当然,这些 DSP 针对的是高价值市场的产品。

DSP 产品的应用十分广泛,其传统的应用市场为 3C 领域,即通信、计算机、消费类产品领域。目前,DSP 已经占据嵌入式通信终端和基础设施市场的主导地位,其“触角”也已延伸至更多的嵌入式数字多媒体应用领域,同时以其完全的软件可编程的灵活性,在众多的数字信息产品解决方案中发挥着越来越大的作用。

1.2 初步认识 DSP

1.2.1 DSP 的定义

DSP 有两种解释:数字信号处理(Digital Signal Processing)和数字信号处理器(Digital Signal Processor)。前者代表数字信号处理的理论,它是理论和计算方法上的技术,着重算法的研究。如 20 世纪 60 年代出现的快速傅里叶变换(FFT),使数字信号处理技术发生了革命性的变化。近年来,数字信号处理的理论和方法得到了迅速的发展,诸如:语音与图像的压缩编码、识别与鉴别,信号的调制与解调、加密和解密,信道的辨识与均衡,智能天线,频谱分析等各种快速算法都成为研究的热点并取得了长足的进步,为各种实时处理的应用提供了算法基础。数字信号处理是以众多学科为理论基础的,同时又使自己成为一系列新兴学科的理论基础。

而数字信号处理器是一种适合数字信号处理运算需求的单片可编程微处理器,又称 DSP 芯片。在 20 世纪 80 年代以前,由于受实现方法的限制,数字信号处理的理论还不能得到广泛的应用。直到 20 世纪 80 年代初,世界上第一块单片可编程 DSP 芯片的诞生,才使理论研究成果广泛应用到实际的系统中,并且推动了新的理论和应用领域的发展。可以毫不夸张地说,DSP 芯片的诞生及发展对近 20 年来通信、计算机、控制等领域的技术发展起到十分重要的作用。本书以 DSP 芯片为介绍对象。

1.2.2 数字信号处理运算概述

1. 为什么选择数字信号处理

数字信号处理就是在数字域中对真实世界的信号进行处理,如语音、音乐、视频、压力、速度、振动等信号,这些信号大都是模拟信号,与模拟信号处理相比,数字信号处理的优势是显而易见的。例如在语音处理系统里使用模拟滤波技术,由于模拟技术存在温度漂移、元件误差和老化等问题,滤波器的质量得不到保证,并且参数调整不灵活。而数字滤波器只是一些数字处理运算,它对二进制数字进行乘、加、减等运算,得到的结果完全可预测和可复制。又如高质量数字音频和视频已经替代了模拟音频和视频。

正因为如此,数字信号处理已成为数字革命的基础。在手机的核心部分、音频和视频播放

器、数码摄像机、电话基础设施、电机控制系统甚至生物辨识安全设备中,都能找到数字信号处理器。

2. 数字信号处理运算的特点

大多数常见的数字信号处理算法可归结为以下几类:

- (1) 卷积(信号混合);
- (2) 相关(信号比较);
- (3) 信号滤波;
- (4) FFT(信号变换)。

这些运算都有如下一些特点:

- (1) 基本类型是大量的乘法运算或乘累加(MAC)运算: $Y(n) = \sum X(i) \times A(i)$;
- (2) 寻址操作很有规律而且可以预测。

一个常见的例子是数字滤波器运算,数学表达式 $y(n)=h(0) \cdot x(n)+h(1) \cdot x(n-1)+\cdots+h(N-1) \cdot x(n-N)$ 描述了数字滤波的过程,它需要大量的乘累加运算。将滤波器系数 $h(r)$ 存放在存储器中,将输入样值 $x(n-r)$ 也存放在存储器中, $h(r)$ 和 $x(n-r)$ 相乘。每当一个新的样值到来便将其放入输入样值表的上端,并删掉表中最老的样值,这样对任意滤波器的设计,样值表总能够保持固定的长度。要使该算法能够快速实现,对处理器有以下要求:

- (1) 快速的乘累加运算;
- (2) 扩展精度;
- (3) 双操作数存取;
- (4) 循环缓冲器;
- (5) 零开销循环。

另一个常见的例子是 FFT 运算,在进行蝶形运算时,相关节点从相邻两点直至跨越 $N/2$ 间隔的地址范围,级间按一定规律排列,虽然要运算 $\log_2 N$ 次,但每级的地址都可以预测,也就是寻址操作很有规律而且可以预测。

这样的运算可以用通用微处理器来完成,但受到其成本和结构的限制,不可能有很高的实时处理能力。而无论是专用 DSP 芯片或通用 DSP 芯片,在结构考虑上都能适应 DSP 运算的这些特点。DSP 提供了许多架构特性,有效减少了进行高效信号处理所需的指令数,因而有更高的运算速度。

3. 数字信号处理的实现方法

数字信号处理器用硬件、软件或软硬结合的方法来实现各种算法,数字信号处理的实现一般有以下几种方法。

- (1) 为不同算法而专门设计的专用芯片

用于做卷积/相关并具有横向滤波器结构: INMOS 公司的 A100、A110, Harris 公司的 HPS43168, PLESSY GEC 公司的 PDSP16256 等;用于做 FFT: Austek 公司的 A41102, PLESSY GEC 公司的 PDSP16150 等。这些都是为做 FIR、IIR、FFT 运算而设计的,因而运算速度高,但是具有有限的可编程能力,灵活性差。

- (2) 为某种应用而设计的系统,即 ASIC 系统

因为 ASIC 只为特定应用提供最佳性能,所以它们在灵活性上非常有限。如果更改了任

意参数,整个 ASIC 必须重新制造,这个流程需要花费数月以及可能数十万美元的 NRE(非重发性工程)成本。使用 ASIC 增加了设计的复杂性,缺乏对市场变更的适应能力,推向市场时间长,因此对设计人员来说,通常 ASIC 并不是一个理想的选择。

(3) 用 FPGA 实现 DSP 的各种功能

现场可编程门阵列(FPGA)器件具有现场可编程能力,允许根据需要迅速重新组合基础逻辑来满足使用要求,它提供了与 ASIC 相近的性能,却没有与重新制造 ASIC 相关的延迟和成本花销。但是,使用 FPGA 进行设计与使用 ASIC 一样复杂,而且只要小小的更改就可能导致需要完全重新布局设计。FPGA 器件的价格也很高,导致它们无法用于高产量应用。在特定应用中,FPGA 可用作 DSP 的补充。

(4) 通用处理器(GPP)

GPP 提供可编程性,因此具有灵活性。但是,许多非信号处理应用(如电子邮件、数据库管理和文字处理)则不要求乘法的扩展使用。为了不断降低这些应用的芯片成本,GPP 通常适度地提供需要几个周期来完成的高效乘法指令。另外,添加每个乘法的结果需要其他指令,这使它们对信号处理应用并不那么理想。

(5) 通用可编程 DSP 芯片

这是目前用得最多的数字信号处理应用器件,它具有可编程性和强大的处理能力,从而可以重新构造以满足很多方面的不同应用。生产这些器件的制造商通常会提供软件开发工具和硬件开发平台,帮助工程师来设计和开发新的应用,在实时 DSP 领域中处于主导地位。

(6) 片上系统 SoC

这是数字化应用及微电子技术迅速发展的产物,是新一代基于 DSP 产品的主要发展方向之一。它把一种应用系统集成在一个芯片上。通常,为满足系统的性能要求和提高功率效率,会把 DSP 和 MCU 的多处理器处理平台集成在一起。它支持语音、音频、图像和视频信号处理应用的各种性能。其中关键器件有:低功耗的 DSP 芯片,用来做媒体处理;MCU,用来支持应用操作系统及以控制为核心的应用处理;MTC 是内存和流量控制器,确保处理器能高效访问外部存储区,避免产生瓶颈现象,提高整个平台的处理速度。

开发人员在开发数字信号处理应用时,应根据应用系统所具有的特色,选择以上合适的数字信号处理架构和技术。

1.3 DSP 的优势

在讨论 DSP 基本架构前,比较一下微控制器、微处理器和数字信号处理器的区别是很有必要的。

(1) 微控制器

- ① 一块芯片上集成 CPU、RAM、ROM、串/并接口、定时器/计数器、中断电路等。
- ② 适合应用于 I/O 和控制能力比速度更重要的地方。
- ③ 紧凑指令集。
- ④ 8051、Atmel、PIC 等单片机。

(2) 微处理器

- ① 单片 CPU,附加外围电路。

- ② RISC 计算机。
- ③ CISC 计算机。
- ④ Pentium 系列、PowerPC、MIPS 等通用计算机。

(3) 数字信号处理器 (DSPs)

- ① 一块芯片上集成 CPU、RAM、ROM、串/并接口、定时器/计数器、中断电路等。
- ② 优化 CPU 架构,使之适合实时信号处理中的快速、重复运算。
- ③ TMS320 系列 DSP。

与单片机、通用处理器相比,DSP 处理器在信号处理应用中在速度、成本以及功率效率方面更具优势。因为 DSP 具有先进而性能优化的 CPU 内核,通过一些诸如单周期乘累加(MAC)、并行处理、流水线、专用内部计算引擎和集成外设的技术获得高性能。这些处理器的性能通常以每秒百万次或十亿次操作来计算。具体来说,在 DSP 架构后面提供了以下一些技术,大大减少了进行高效信号处理所需的时间。

1. 单周期乘累加单元

快速运算是 DSP 最基本的要求。如前所述,数字信号处理的基本类型是大量的乘法运算或乘累加运算。任何一个 DSP 内部都有一个或多个硬件 MAC 单元,能够在一个指令周期内完成一个或多个乘累加运算。而 GPP 用指令实现乘累加操作。因此,DSP 中的 MAC 单元很容易加速实现这种乘累加运算。表 1-1 列出了 GPP 与 DSP 在进行乘累加运算时所花费的时间,显然 GPP 需要 6 个时钟周期执行一次乘和一次加,而 DSP 只需要一个时钟周期就能完成一次乘累加。

表 1-1 MAC 处理效率

	乘	加	乘累加
GPP 所需时钟周期(Cycles)	5	1	6
DSP 所需时钟周期(Cycles)	—	—	1

2. 多级流水线结构

为什么 DSP 执行一次乘累加操作只需要一个时钟周期,而 GPP 执行一次乘和一次加需要 6 个时钟周期? 这是因为 DSP 具有流水线结构。流水线结构是大多数 DSP 实现高速运算的一个非常重要的技术手段,它使实际需要几个时钟周期才能完成的工作以“单周期”的方式呈现出来。DSP 具有多级流水线。它将每条指令分解为多步,并让各步操作在时间上重叠,从而实现几条指令并行处理。程序中的指令仍是一条条顺序执行,但可以预先取若干条指令,并在当前指令尚未执行完时提前启动后续指令的另一些操作步骤。这样显然可加速一段程序的运行过程。在理想情况下,每步需要一个时钟周期。当流水线完全装满时,每个时钟周期平均有一条指令从流水线上执行完毕,输出结果。

3. 并行处理

由于 DSP 处理器的更高性能不能从传统结构中得到解决,因此提出了各种提高性能的策略。其中提高时钟频率似乎是有限的,最好的方法是提高并行性。提高操作并行性可以由两条途径实现:提高每条指令执行的操作数量,或者是提高每个指令周期中执行的指令数量。

这两种并行要求产生了现代 DSPs 新结构。

(1) 增强型 DSP

在保持 DSP 原有结构和指令集不变的情况下增加执行单元,例如,一些高端的 DSP 有多个乘法器,而不是一个,把使用这种方法的 DSP 叫做增强型 DSP。因为它们的结构与前一代的 DSP 相似,但性能在增加执行单元后大大增强了。当然,指令集必须也同时增强,这样编程者才能在一条指令中指定更多的并行操作,以利用额外的硬件。例如,TI 的 C64x 系列 DSP 有 4 个 MAC 单元,因此处理 4 次乘累加运算也只需一个时钟周期,速度提高 4 倍。表 1-2 列出了 GPP 与 DSP 进行多次乘累加运算的效率比较。

表 1-2 GPP 与 DSP 进行 4 次乘累加运算的效率比较

	1 次乘累加	4 次乘累加
GPP 所需时钟周期(Cycles)	6	24
DSP 所需时钟周期(Cycles)	1	1

(2) 超长指令字(VLIW)结构

超长指令字是高端 DSP 普遍采用的体系结构,VLIW 的特点是它能从应用程序中提取高度并行的指令数据,并把这些机器指令均匀地分配给芯片中的众多执行单元,同时还需要智能化的编译软件配合以安排这些指令的执行。例如,Siemems 的 Carmel、Philips 的 TriMedia 和 TI 的 TMS320C62xx 处理器族都使用了超长指令字结构。C62xx 处理器每次取一个 256 位的指令包,把包解析为 8 个 32 位的指令,然后把它们引到其 8 个独立的执行单元。在最好的情况下,C62xx 同时执行 8 个指令,这种情况下达到了极高速度(如 1600MIPS)。VLIW 结构的优点是性能高、结构规整(潜在的易编程和好的目标编译系统)。

(3) 超标量体系结构

同 VLIW 结构一样,超标量体系结构并行地流出和执行多个指令,不同在于超标量体系结构依靠硬件来实现并行处理的调度,VLIW 则充分发挥软件的作用,从而使硬件简化、性能提高。超标量运算在大多数 DSP 中得到重用,它使得 DSP 器件使用多个执行单元,同时执行两个或更多独立指令。超标量结构的优点是性能有大的跨越、结构规整、代码宽度没有明显增长;缺点是功耗非常高、指令的动态安排使代码优化困难。

4. 哈佛结构

传统的 GPPs 使用冯·诺依曼存储结构,在这种结构中,有一个存储空间通过两条总线(一条地址总线和一条数据总线)连接到处理器内核,如图 1-1(a)所示,这种结构不能满足 MAC 必须在一个指令周期中对存储器进行 4 次访问的要求。DSP 芯片普遍采用数据总线和程序总线分离的哈佛结构或改进的哈佛结构,如图 1-1(b)所示,其特点是把程序代码和数据的存储空间分开,并有各自的地址和数据总线,每个存储器独立编址,用独立的一组程序总线和数据总线进行访问。图 1-1(c)所示为改进的哈佛结构示意图,除了哈佛结构的特征外,程序存储空间与数据存储空间之间还可以进行数据交换,这种结构可以并行进行数据操作。例如在做数字滤波时把系数放在程序存储空间,把待处理的样本数据放在数据存储空间,处理时可以同时提取滤波器系数和样本进行乘累加操作,从而大大提高运算速度。改进哈佛结构还可以从程序存储空间来初始化数据存储空间,或把数据存储空间的内容转移到程序存储空间,这

样可以复用存储器、降低成本、提高存储器使用效率。

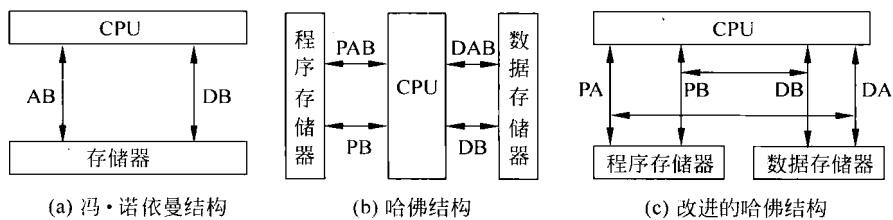


图 1-1 存储器结构

5. 零开销循环控制

大多数 DSP 算法有一些共同特征：高速重复执行一系列的 MAC 指令，如滤波器和 FFT 算法等；大部分处理时间花在执行包含几行代码的循环体上。因此，大部分 DSP 处理器具有零消耗循环控制的专门硬件。零消耗循环是指处理器不用花时间测试循环计数器的值就能执行一组指令的循环，硬件完成循环跳转和循环计数器的递减。有些 DSPs 还通过一条指令的超高速缓存实现高速的单指令循环。以一个语音编解码器为例，将一段语音信号转换成低比特率的数据流需要重复执行 MAC 指令上千次或更多，若没有硬件循环，而是用软件来管理循环计数器变量，该过程将很低效。但对于 DSP 芯片来说，循环计数器由硬件而非软件维持，从而减少了处理器内核的负担。

6. 特殊寻址模式

为了加速 DSP 复杂运算的执行效率，DSP 芯片提供了许多不同的取数据和确定数据地址的机制。DSP 经常包含有专门的地址产生器，它能产生信号处理算法需要的特殊寻址，如循环寻址和位翻转寻址。循环寻址对应于 FIR 滤波算法，位翻转寻址对应于 FFT 算法。

以上是 DSP 芯片为获得高性能在体系结构上所做的优化。表 1-3 列出了 GPP 与 DSP 在上述性能方面的比较。显然 DSP 在架构上所做的这些改进能较好地适应 DSP 运算的特点，因此 DSP 是当前应用较为广泛的数字信号处理应用器件。

表 1-3 GPP 和 DSP 在架构上性能的比较

	GPP	DSP
硬件乘累加单元	Some	Yes
单周期 MAC 单元	Some	Yes
每周期执行多个 MAC 操作	No	Yes, architecture dependent
优化指令集	Some	Yes
每周多存储器存取	No	Yes
硬件零开销循环	No	Yes
硬件循环寻址	No	Yes

为了解释 DSP 较 GPP 的优越性，举一个的例子来说明。表 1-4 列出了 Metalink 公司

的 MIPS 4KE 处理器与 TI 公司的 C55x DSP 在 VoIP 相关信号处理算法上运算时间的比较。

表 1-4 应用于 VoIP 上的 MIPS 处理器与 DSP 的性能比较

	MIPS 4KE	C55x	比例(MIPS/C55x)
16 ms 回波抵消	20	5	4
G. 723	45	15	3
G. 729A	26	10	2.6
G. 728	77	25	3.08
平均			3.17

从表 1-4 中看到在 MIPS 处理器上运行 VoIP 相关软件的时间平均是 C55x DSP 的 3.17 倍,这是因为 C55x DSP 的架构具有更好的性能,如高级流水线、优化的指令集、双 MAC 单元、每周期多存储器访问的能力以及硬件循环等,综合比较后得出以下结论。

① 具有相同功能的处理程序在 MIPS 处理器中运行需要 300 MHz,但在 DSP 中运行仅需要 100 MHz。

② 由于 DSP 以较低的频率运行,其功耗比完成相同处理程序的 GPP 处理器功耗小得多。因此,选择 DSP 进行高级数字信号处理的好处在这里显现无疑。

1.4 DSP 芯片的应用

实时系统对数据处理的要求促进了 DSP 的出现和发展,因此 DSP 芯片常用于一些实际信号(如语音、视频、音乐等)的实时检测、处理和产生。理解 DSP 技术对设计的影响的最佳方法是仔细观察 DSP 已取得对替代技术的明显优势的一些应用领域。数字化让开发人员能够大大超出他们自己的期望,提供模拟处理远不能达到的功能。就手机来讲,目前它的功能越来越强大,它能够处理实时的音视频信号完全依赖于高性能的 DSP 芯片。单片 DSP 器件可以执行实时语音压缩、回波抵消、语音识别、调制/解调、视频压缩、波形编码、隔行扫描、多径均衡、软判决解码、卷积、自动频率、自动功率和自动增益控制等功能。

以下列出了 DSP 的一些典型应用。

- (1) 信号处理: 数字滤波、自适应滤波、快速傅里叶变换、相关运算、频谱分析、卷积等。
- (2) 通信: 调制解调器、自适应均衡、数据加密、数据压缩、回波抵消、多路复用、传真、扩频通信、纠错编码、波形产生等。
- (3) 语音: 语音编码、语音合成、语音识别、语音增强、说话人辨认、说话人确认、语音邮件、语音存储等。
- (4) 图形/图像: 二维和三维图形处理、图像压缩与传输、图像增强、动画、机器人视觉等。
- (5) 军事: 保密通信、雷达处理、声呐处理、导航等。
- (6) 仪器仪表: 频谱分析、函数发生、锁相环、地震处理等。
- (7) 自动控制: 引擎控制、声控、自动驾驶、机器人控制、磁盘控制。
- (8) 医疗: 助听、超声设备、诊断工具、病人监护等。