

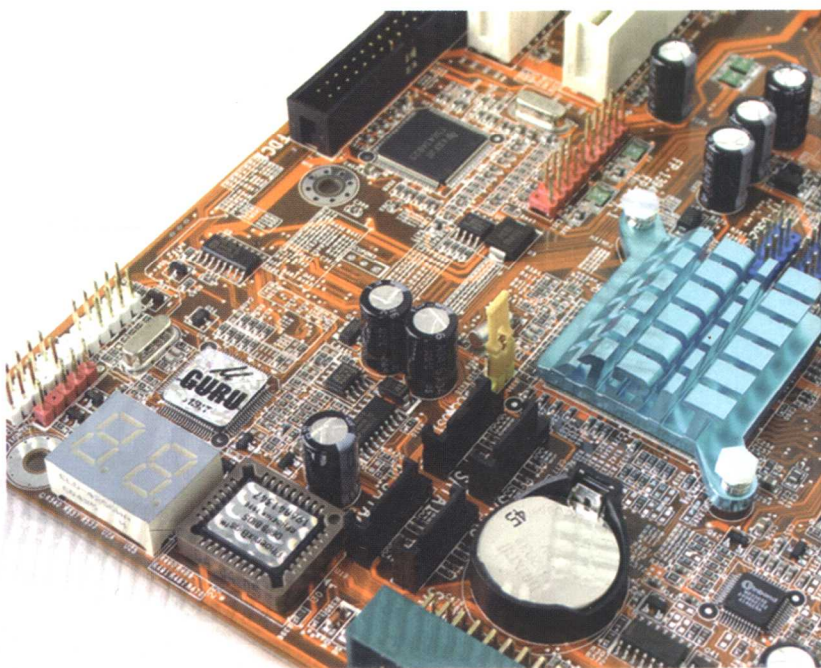
DSP



原理与应用

从入门到提高

◇ 姜沫岐 许涵 俞鹏 段国强 编著



TN911.72

163

2007

DSP 原理与应用从入门到提高

姜沫岐 许 涵 俞 鹏 段国强 编著



机械工业出版社

在科学技术日新月异的 21 世纪, 数字信号处理器 (DSP) 已成为通信、计算机、消费类电子产品等诸多领域的基础器件, 它以其卓越的性能而受到时代的青睐和人们的信赖!

本书按照突出重点、注重实用的思想, 介绍了 TI 公司 DSP 的理论知识 and 开发应用。全书以 TMS320C54x DSP 为重点, 不仅介绍了 DSP 的硬件结构、指令系统、开发工具等知识, 而且从实际应用角度出发, 讲解了 DSP 系统的设计实例, 包括器件选型、最小系统设计和多种常用接口的设计, 给出了丰富、新颖的软件实例和翔实的注解。本书旨在帮助读者尽快掌握 DSP 入门知识及拥有开发实际系统的能力。

本书内容全面、寓学于用、实用性强, 既可作为大中专院校相关专业的教材或参考书, 又可作为广大 DSP 工程技术人员的参考书和培训教材。

图书在版编目 (CIP) 数据

DSP 原理与应用从入门到提高/姜沫岐等编著. —北京: 机械工业出版社, 2006.12

ISBN 978 - 7 - 111 - 20611 - 8

I. D… II. 姜… III. 数字信号 - 信号处理 IV. TN911.72

中国版本图书馆 CIP 数据核字 (2006) 第 156432 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 张俊红 责任编辑: 顾 谦 版式设计: 霍永明

责任校对: 张晓蓉 封面设计: 陈 沛 责任印制: 李 妍

唐山丰电印务有限公司印刷

2007 年 2 月第 1 版第 1 次印刷

184mm × 260mm · 20.25 印张 · 499 千字

0 001 - 4 000 册

标准书号: ISBN 978 - 7 - 111 - 20611 - 8

定价: 33.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

销售服务热线电话: (010) 68326294

购书热线电话: (010) 88379639 88379641 88379643

编辑热线电话: (010) 88379768

封面防伪标均为盗版

前言

最近 40 年中, 数字信号处理技术得到了空前的发展, 它几乎存在于任何电子领域, 大到卫星、宇宙飞船, 小到人们生活中所熟知的各种电子产品 (如手机、数码相机等) 都可以寻到它的踪迹。数字化技术的飞速发展是与数字信号处理器 (DSP) 的强大支持密不可分的, DSP 作为数字信号处理技术的实施者, 是执行数字信号处理任务的尖兵利器。随着 DSP 性价比的不断提高、功能的不断完善以及器件功耗的不断减少, DSP 芯片越来越多地渗透到了各种电子产品当中, 已成为通信、计算机、消费类电子产品等诸多领域的基础器件。进入新世纪以来, 3G 移动通信技术的不断普及, 无线通信新应用的不断涌现, 将进一步推动 DSP 走进千家万户!

随着 DSP 应用的不断扩展, DSP 系统设计和开发技术同样得到了广泛的关注, 相应的对 DSP 开发人员的需求量也与日俱增。目前, 越来越多的电子公司、科研院所和大中专院校把他们的关注焦点都定格在了这一革新技术之上。

本书正是以占市场份额 40% 的 TI 公司 DSP 为背景, 从应用角度着手, 介绍了 DSP 系统设计和开发的各方面知识。全书共分 6 章。第 1 章首先介绍了数字信号处理技术的概念和特点, 然后讲解了 DSP 的知识及其在多个领域内的广泛应用, 最后展望了 DSP 技术今后的发展趋势, 力图使读者对 DSP 有一个宏观的了解。第 2 章介绍了 TMS320C54x 系列 TI DSP 的体系结构, 包括总线结构、存储器空间、片内外设、特殊寄存器等多方面知识。第 3 章基于 TMS320C54x DSP, 从汇编语言和 C 语言两个方面介绍了 DSP 程序设计的方法, 对 DSP 特有的指令和使用方法进行了重点讲解。第 4 章基于 TMS320C54x DSP 系统, 介绍了 TI 公司 DSP 集成开发工具环境 (Code Composer Studio, CCS) 的用法, 包括 CCS 的基本操作、DSP/BIOS 插件、实时数据交换 (RTDX) 等方面, 并给出调试实例, 以深化读者对 CCS 的认识。第 5 章从应用角度出发, 讲解了 DSP 系统开发的全过程——从器件选型到系统规划、再到软硬件实现, 旨在帮助读者深化理论知识, 缩短从理论学习到实践应用的距离, 以达到所学即所用的目的。第 6 章重在通过讲解 DSP 应用实例, 帮助读者从实践中升华理论, 梳理知识脉络, 真正拥有 DSP 应用开发的能力。

本书贯穿先原理后应用, 先硬件后软件的写作思想, 旨在帮助读者循序渐进地掌握通用 DSP 的基本理论及体系结构, 然后结合一些典型实例和实验程序来深化读者对 DSP 知识的融会贯通, 达到活学活用的目的。在应用部分, 书中引入了一些易于被忽视但是却必不可少的环节, 如 DSP 选型、设计常见问题以及获得帮助的方法等; 注重“渔”与“鱼”的结合, 以使读者尽快、尽好地掌握 DSP 开发的技巧和方法。本书既可作为大中专院校相关专业的教材或参考书, 也可作为广大 DSP 工程技术人员的参考书和培训教

IV

材。

本书由姜沫岐、许涵、俞鹏、段国强共同编写，吕美东、林伟玲、张楠、张杨柏参与了校对和程序调试工作，吝宁、袁晓东、汤劫莹、李英奇、付聪也做了大量的后期工作，在此一并对他们的辛勤劳动表示衷心的感谢！

由于作者水平有限，时间仓促，书中难免存在差错和不足之处，敬请广大读者批评指正！

作者

目 录

前言

第 1 章 DSP 概述	1
1.1 DSP 技术概述	1
1.1.1 数字信号处理发展史	1
1.1.2 DSP 概述	2
1.2 DSP 系统	3
1.2.1 DSP 系统的构成	3
1.2.2 DSP 系统的特点	4
1.3 DSP 应用系统的开发设计流程	5
1.3.1 DSP 系统开发准备	6
1.3.2 DSP 系统整体开发流程	7
1.4 DSP 概述	9
1.4.1 DSP 芯片	9
1.4.2 DSP 芯片的发展	10
1.4.3 DSP 芯片的分类	11
1.4.4 DSP 芯片的选择	12
1.5 DSP 应用	12
1.6 DSP 的发展方向	13
1.6.1 系统级集成 DSP 是潮流	14
1.6.2 可编程 DSP 是主导产品	14
1.6.3 追求更高的运算速度	14
1.6.4 定点 DSP 是主流	15
第 2 章 DSP 体系结构	16
2.1 C54x DSP 体系结构概述	16
2.2 C54x DSP 的总线结构	18
2.2.1 内部总线结构	18
2.2.2 外部总线结构	19
2.3 C54x DSP 存储空间结构	19
2.3.1 存储器空间	22
2.3.2 程序存储器	27
2.3.3 数据存储器	29
2.3.4 I/O 存储空间	31
2.4 片内外设	32
2.4.1 通用 I/O 引脚	32
2.4.2 定时器	32
2.4.3 时钟发生器	35
2.4.4 C54x DSP 的主机接口	38

2.4.5 串行接口	42
2.5 特殊寄存器	47
2.5.1 状态及控制寄存器	47
2.5.2 累加器 A 和 B	51
2.5.3 与中断有关的寄存器	53
第 3 章 DSP 程序开发	55
3.1 汇编语言程序设计概述	55
3.1.1 汇编语言源程序格式	55
3.1.2 符号和缩写	57
3.1.3 记号和运算符	60
3.2 寻址方式	61
3.2.1 立即寻址	61
3.2.2 绝对寻址	61
3.2.3 累加器寻址	62
3.2.4 直接寻址	62
3.2.5 间接寻址	64
3.2.6 存储器映射寄存器寻址	67
3.2.7 堆栈寻址	67
3.3 汇编指令系统	67
3.3.1 TMS320C54x 指令表	68
3.3.2 TMS320C54x 指令集说明	77
3.4 汇编伪指令	117
3.4.1 段定义伪指令	118
3.4.2 常数初始化伪指令	118
3.4.3 段程序计数器定位伪指令	120
3.4.4 输出列表格式伪指令	121
3.4.5 引用其他文件的伪指令	122
3.4.6 条件汇编伪指令	122
3.4.7 汇编时符号定义伪指令	122
3.4.8 其他汇编伪指令	123
3.5 汇编宏指令	123
3.5.1 宏定义	123
3.5.2 宏调用	123
3.5.3 宏扩展	124
3.6 公共目标文件格式	125
3.6.1 块	125
3.6.2 汇编器对块的处理	126

3.6.3 链接器对块的处理	126	4.5.4 RTDX 工具	191
3.7 C 语言程序设计概述	129	4.5.5 在不含 DSP/BIOS 的目标程序 中使用 RTDX	192
3.8 C 语言开发	130	4.5.6 DSP/BIOS 的 RTDX 模块	197
3.8.1 数据类型	130	4.5.7 RTDX 主机客户程序的开发	205
3.8.2 关键字	131	4.6 软件调试及实例	205
3.8.3 寄存器变量与全局寄存 器变量	132	4.6.1 断点	205
3.8.4 Pragma 伪指令	133	4.6.2 观察窗	205
3.9 C 代码优化	135	4.6.3 探点	206
3.9.1 优化器的使用	135	4.6.4 文件输入/输出能力	206
3.9.2 执行文件级优化	136	4.6.5 图形显示工具	206
3.9.3 执行程序级优化	137	4.6.6 符号浏览器	207
3.9.4 控制程序级优化	137	4.6.7 命令窗口	207
3.10 混合编程	138	4.6.8 调试实例	207
3.10.1 独立的 C 模块和汇编模 块接口	138	第 5 章 TMS320C54x DSP 系统	
3.10.2 在 C 语言程序中使用汇编 程序中的变量和常数	139	硬件设计	217
3.10.3 直接在 C 语言程序的相应位 置嵌入汇编语句	140	5.1 DSP 系统开发流程	217
第 4 章 DSP 集成开发环境	142	5.1.1 DSP 硬件系统开发流程	217
4.1 CCS 概述	142	5.1.2 高频 DSP 系统的 PCB 设计	219
4.2 CCS2 的安装和设置	143	5.1.3 DSP 系统硬件调试技巧	220
4.2.1 CCS2 的安装	143	5.2 DSP 选型	220
4.2.2 CCS2 配置程序的安装	145	5.2.1 TMS320C1x	220
4.3 CCS2 的基本操作	148	5.2.2 TMS320C2x	222
4.3.1 CCS2 集成开发环境的特性	148	5.2.3 TMS320C5x	225
4.3.2 菜单栏	148	5.2.4 TMS320C2xx	227
4.3.3 CCS2 的基本操作	161	5.2.5 TMS320C54x	228
4.4 DSP/BIOS 的原理及应用	168	5.2.6 TMS320C62x	229
4.4.1 概述	168	5.3 DSP 最小系统设计	229
4.4.2 DSP/BIOS 的功能介绍	168	5.3.1 TMS320C54x 芯片简介	230
4.4.3 DSP/BIOS 的组成部分	169	5.3.2 复位、时钟、电源部分 的设计	231
4.4.4 DSP/BIOS 的配置工具	170	5.3.3 总线缓冲部分的设计	236
4.4.5 创建具有 DSP/BIOS 功能 的应用程序	174	5.4 DSP 外围电路的设计	237
4.4.6 DSP/BIOS 内核	177	5.4.1 A/D 转换电路的设计	237
4.4.7 DSP/BIOS 实时分析工具	182	5.4.2 D/A 转换电路的设计	241
4.4.8 DSP/BIOS 的应用实例	184	5.4.3 音频信号编解码模块 的设计	246
4.5 CCS 实时数据交换	190	5.4.4 DSP 存储扩展的设计	255
4.5.1 RTDX 介绍	190	5.4.5 DSP 异步串口的设计	257
4.5.2 RTDX 数据传送机理	190	5.4.6 DSP 以太网接口的设计	258
4.5.3 RTDX 传送模式	191	5.5 DSP 系统中的 CPLD 的设计	259
		5.5.1 CPLD 简介	259
		5.5.2 CPLD 的选择	262

5.5.3 CPLD 在 DSP 中的作用	263	6.3.1 FIR 滤波器基本结构	285
5.5.4 CPLD 在 DSP 系统中的 应用实例	263	6.3.2 Matlab 辅助 FIR 滤波器设计	286
第 6 章 TMS320C54x DSP 软件		6.3.3 FIR 滤波器的 DSP 实现	288
开发实例	266	6.4 软件无线电	292
6.1 基础	266	6.4.1 FSK 调制解调原理	292
6.1.1 加减运算	266	6.4.2 2FSK 程序实例	293
6.1.2 重复操作	266	6.5 UART 模块	300
6.1.3 乘法运算	268	6.5.1 TL16C550	301
6.1.4 长字运算	269	6.5.2 软件编程	302
6.1.5 堆栈使用	270	6.6 CODEC 模块	305
6.1.6 小数运算	270	6.6.1 TLC320AC01	306
6.1.7 除法运算	273	6.6.2 McBSP	306
6.2 FFT	275	6.6.3 CODEC 软件设计举例	306
6.2.1 FFT 算法原理	275	6.7 Flash 存储器模块	311
6.2.2 FFT 算法实现	276	6.7.1 硬件连接	311
6.3 FIR	285	6.7.2 软件编程	311
		参考文献	315

第 1 章 DSP 概述

步入 21 世纪后, 社会进入了数字化的时代, 数字信号处理技术的飞速发展极大地提高了人们对数字世界的把握能力。毫不夸张地说, 数字信号处理技术是这个时代的核心技术之一, 而数字信号处理器 (Digital Signal Processor, DSP) 正是这场数字化革命的核心。

1.1 DSP 技术概述

1.1.1 数字信号处理发展史

数字信号处理 (Digital Signal Processing) 亦即信号的数字处理, 它是一门广泛应用于多领域的新兴学科。在某种意义上, 数字信号处理技术的发源可以追溯到 17 世纪, 当时主要进行信号解析手段的研究, 开发了有限差分方法、数值积分方法和数字内插方法, 用以解决涉及到连续变量和函数的物理问题。20 世纪 50 年代, 随着大型计算机的出现, 数字信号处理开始兴起, 主要是进行各种模拟信号的数字化。到了 60 年代, 科研人员开始将数字信号处理本身考虑为一个独立的领域, 这一阶段主要以 1965 年 Cooley - Tukey 提出的快速傅里叶变换算法为标志, 这时数字信号处理技术主要用于图像处理、快速数据传输、生物医学系统等。从那之后, 数字信号处理理论和应用就有了飞速和长足的发展, 新理论、新算法不断涌现。

数字信号处理是围绕着数字信号处理的理论、实现和应用等几个方面发展起来的, 它以众多的学科为理论基础, 所涉及的范围极其广泛。在数字领域中, 微积分、概率统计、随机过程、高等代数、数值分析、近代代数、复变函数、线性代数、泛函分析等都是数字信号处理的基本工具, 网络理论、信号与系统、控制论、通信理论等均是它的理论基础。

在 20 世纪 80 年代以前, 由于方法的限制, 数字信号处理技术还处于理论研究阶段, 在此阶段, 人们利用通用计算机进行数字滤波、频谱分析等算法的研究以及数字信号处理系统的模拟和仿真。80 年代初, 随着大规模集成电路的发展, 世界上第一片可编程 DSP 芯片的诞生为数字信号处理理论的实际应用开辟了一条广阔的通道。这种 DSP 采用微米工艺、NMOS 技术制作, 虽然功耗和尺寸稍大, 但运算速度却比通用计算机快几十倍, 尤其在语音合成和编解码中得到了广泛的应用。DSP 芯片的诞生是个里程碑, 使 DSP 应用系统由大型系统向小型化迈进了一大步。到了 80 年代中期, 随着 CMOS 技术的进一步发展, 第二代基于 CMOS 工艺的 DSP 应运而生, 其存储容量和运算速度都得到了成倍的提高, 成为语音处理及图像处理技术的基础。低成本、高效率的 DSP 不断涌现, 更加促进了数字信号处理技术发展的进程。

20 世纪 90 年代以后, DSP 芯片的发展更加突飞猛进。随着 DSP 芯片功能的日益强大、运算精度和运算速度的不断提高、性价比的不断上升以及开发手段的不断改进, DSP 芯片已经成为集成电路中发展最快的电子产品, 并成为众多电子产品的核心器件。目前几乎所有的

图像编码压缩系统和语音带宽压缩系统都倾向于全数字化。数字信号处理已成为现代化雷达和声纳,甚至医疗诊断仪器中不可缺少的部分。除了专用数字信号处理硬件有所发展外,还出现了可编程的数字信号处理专用计算机,实现了对信号的实时处理和分析。而 DSP 系统也被广泛地应用于当今技术革命的各个领域——通信电子、自动控制、信号处理、雷达、军事、航空航天、医疗、电力电子、家用电器,并且新的应用领域正在不断地被挖掘、拓展。数字信号处理的重要性仍在不断提高,其迅猛发展的势头丝毫没有减弱的迹象,完全可以预料,这门技术将来的发展将比上述介绍的发展成果更令人瞩目。

1.1.2 DSP 概述

DSP 的解释有两种:一种是数字信号处理技术,英文为 Digital Signal Processing;另一种是数字信号处理器,英文为 Digital Signal Processor。它们的英文名均可缩写为 DSP,本书以 DSP 作为数字信号处理器的缩写。

1. 数字信号处理 数字信号处理是一种用离散的数字或符号序列来表示信号,并用数字硬件或计算机对这些离散序列进行有效处理的技术。它利用计算机或专用处理设备,以数字形式对信号进行采集、变换、滤波、估值、增强、压缩和识别等处理,得到符合需要的信号形式。它主要从事研究的内容有两个方面:一方面是滤除混杂在有用信号中的噪声和干扰,削弱采集信号中的多余成分,这就是数字滤波;另一方面是为了分离两个或多个按某种方式合并在一起的信号,也可能是希望增强一个信号中的某一分量,使之便于对它们进行分析和识别,这就是各类变换和算法。

数字信号处理理论经过 30 年的发展,已经形成了比较完善的理论体系,它的主要内容有:信号的采集(A/D 转换技术、采样定理、多采样率、量化噪声分析等),离散信号的分析(时域及频率分析、多种变换技术、信号特征的描述等),离散系统分析(系统的描述、系统的单位采样响应、转移函数及频率特性等),信号处理中的快速算法(快速傅里叶变换、快速卷积与相关运算等),信号的估值(各种估值理论、相关函数与功率谱估计等),滤波技术(各种数字滤波器的设计与实现),信号的建模(最常用的是 AR(自回归模型)、MA(滑动平均模型)、ARMA(自回归滑动平均模型)、PRONY(普罗尼模型)等各种模型),信号处理中的特殊算法(如抽取、插值、奇异值分解、反卷积、信号重建等)。

数字信号处理的实现方法一般有以下五种:

(1) 在通用的计算机(如 PC(个人计算机))上用软件(如 C 语言)实现。

该方法速度太慢,适合于算法仿真。

(2) 在通用计算机系统上加上专用的加速处理机实现。

该方法专用性较强,应用受限制且不利于系统的独立运行。

(3) 用通用的单片机(如 MCS-51、96 系列等)实现。

这种方法多用于一些不太复杂的数字信号处理,如简单的 PID 控制算法。

(4) 用通用的可编程 DSP 芯片实现。

与单片机相比,DSP 芯片具有更加适合于数字信号处理的软件及硬件资源,可用于复杂的数字信号处理算法。

(5) 用专用的 DSP 芯片实现。

在一些要求信号处理速度极高的特殊场合,用通用的 DSP 芯片很难实现极高速运算,

而专用的 DSP 芯片可以将相应的信号处理算法在芯片内部用硬件实现, 无须进行编程。

2. 数字信号处理器 数字信号处理器也称为 DSP 芯片, 是一种专门用于进行数字信号处理运算的、具有特殊结构的微处理器, 主要用于实时快速地实现各种数字信号处理算法。DSP 芯片的内部采用程序存储空间和数据存储空间分离的哈佛结构, 具有专门的硬件乘法器, 广泛采用流水线操作, 提供特殊的数字信号处理指令, 可以用来快速地实现各种数字信号处理算法。

所谓哈佛结构是指将程序的存储空间和数据的存储空间分开, 各有各的地址总线 and 数据总线, 它允许同时取指令 (来自程序存储器) 和取操作数 (来自数据存储器), 而且还允许在程序存储空间和数据存储空间之间相互传送数据。这样, 同一条指令就可以同时对不同的数据空间进行读操作或写操作, 从而提高处理速度。和哈佛结构相配合使用的就是流水线操作。所谓流水线操作, 就是将各条指令的执行时间重叠, 执行完第一条指令第一步后, 紧接着执行该指令的第二步, 同时执行下一条指令的第一步, 使得指令执行加快, 使大多数指令都可以在单个指令周期内完成, DSP 执行一条指令需要经过取指令、译码、访问、执行等几个步骤。

DSP 是一种专用且特殊的微处理器。它的可编程性、特殊的内部结构、强大的信息处理能力以及远远超过通用微处理器的较高的运行速度等特点, 使它成为高性能系统的核心。它接收模拟信号, 如光和声音, 将它们转化为数字信号, 并实时地对大量数据进行数字技术处理, 这种实时处理的能力使 DSP 在图像处理、声音处理等不允许延时的领域内的应用相当理想和广泛。

根据数字信号处理的技术要求, DSP 芯片一般具有如下一些主要特征:

- (1) 在一个指令周期内可完成一次乘法和一次加法。
- (2) 程序空间和数据空间分开, 可以同时访问指令和数据。
- (3) 片内的快速 RAM, 通常可以通过独立的数据总线在两块芯片中同时访问。
- (4) 具有低开销或零开销的循环和跳转的硬件支持。
- (5) 具有在单周期内操作的多个硬件地址产生器。
- (6) 可以并行执行多个操作。
- (7) 支持流水线操作, 使取指令、译码、访问和执行等操作可以流水重叠执行。
- (8) 快速的中断处理和硬件 I/O 支持。

1.2 DSP 系统

1.2.1 DSP 系统的构成

图 1-1 所示为一个典型的 DSP 系统框图。它的基本部件包括抗混叠滤波器 (Anti-aliasing Filter)、模/数 (A/D) 转换器 (Analog-to-Digital Converter, ADC)、DSP、数/模 (D/A) 转换器 (Digital-to-Analog Converter, DAC)。图中的输入信号可以有各种各样的形式, 它可以是电信号、声音信号、物理信号、化学信号、连续信号、数字信号、强信号、弱信号。

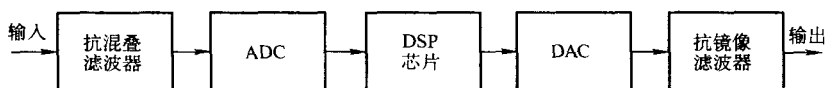


图 1-1 典型的数字信号处理系统框图

一般地说，首先通过传感器将各种信号转化为一定幅值的电信号，再通过抗混叠滤波器将输入的模拟信号进行带限滤波和采样，将信号中高于 Nyquist 频率的频率成分滤掉，然后通过 A/D 转换将模拟信号转换成 DSP 可以处理的并行或串行的数字比特流。根据 Nyquist 采样定理，对低通模拟信号，为保持信息的不丢失，采样频率至少必须是输入带限信号最高频率的两倍。DSP 芯片的输入是 A/D 转换后得到的以采样形式表示的数字信号，DSP 芯片对输入的数字信号进行某种形式的处理，如进行一系列的乘法累加操作（MAC）。数字处理是 DSP 的关键，这与其他系统（如电话交换系统）有很大的不同，在交换系统中，处理器的作用是进行路由选择，它并不对输入数据进行修改。因此，虽然两者都是实时系统，但两者的实时约束条件却有很大的不同。最后，经过处理的数字信号再经数/模（D/A）转换为模拟信号之后，由抗镜像滤波器完成模拟波形的重现，得到连续的模拟波形。

需要指出的是，上面给出的 DSP 系统是一个典型的 DSP 系统模型，所以，并不是所有的 DSP 系统都要包括模型中的所有部分。如频谱分析中输出的不是连续波形而是离散的频谱；语音、图像识别系统中，输出端并不是连续的波形而是识别结果，如数字、文字等；在很多场合、输入信号本身可能就已经是数字信号了，因此根本不需要进行 A/D 转换了。

1.2.2 DSP 系统的特点

DSP 系统是以数字信号处理为基础的，因此具有数字信号处理的全部优点，数字信号处理有别于普通的科学计算和分析，它强调运算处理的实时性，因此 DSP 除了具备普通微处理器所强调的高速运算、控制功能外，还强调实时数字信号处理，它在处理器结构、指令流程、指令系统上做了很大的改动，因而有一些特殊的结构特点。

1. 存储器结构 一般微处理器的存储器结构可分为两大类：冯·诺依曼结构和哈佛结构。典型的冯·诺依曼结构的特点是数据和程序共用一套总线，即只有一个存储器空间、一套地址总线和一套数据总线，指令、数据都存放在这个存储空间中，统一分配地址，所以处理器必须分时访问程序和数据空间，执行速度较慢，通常作一次乘法会发生 4 次存储器访问，用掉至少 4 个指令周期。DSP 普遍采用了数据总线和程序总线分离的哈佛结构及改进的哈佛结构，即数据存储空间和程序存储空间分离，并采用多套地址、数据总线，程序和数据存于不同的存储空间中，每个存储空间独立编址，独立访问。因此处理器可以同时取指令（取自程序存储空间）和取操作数（取自数据存储空间），并且允许在程序存储空间和数据存储空间之间相互传送数据。哈佛结构使 DSP 拥有了比传统处理器更高的指令执行速度。

2. 流水线技术 DSP 大多采用流水线操作技术，如图 1-2 所示将指令的执行分解为取指令、译码、访问和执行等几个阶段，几个阶段由片内多个功能单元分别完成。在程序运行过程中，不同指令的不同阶段在时间上是可以重叠的，这种技术在不提高时钟频率的条件下减少了每条指令的执行时间，有助于保证数字信号处理的实时性。图 1-2 所示为三级流水线操作的例子。

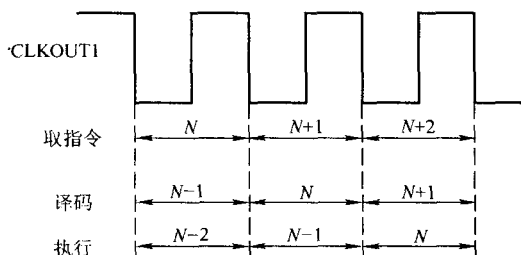


图 1-2 三级流水线操作

3. 乘法累加器 由于 DSP 的任务，如滤波、相关、矩阵运算等需要大量乘法累加运算，所以 DSP 大都有独立的乘法器和加法器，用这些专门的硬件来实现单周期乘法，并使用累加器、寄存器来处理多个乘积的累加，使得 DSP 可以在同一时钟周期内完成相乘、累加两个运算。有的 DSP 可以同时进行乘、加、减运算，大大加快了像傅里叶快速变换（FFT）蝶形运算等的运算速度。

4. 特殊的寻址方式 片内有多条总线可以同时进行取指令和多个数据存取的操作，并且有辅助寄存器用于寻址，它们可以在当前访问前/后自动修改内容，以指向下一个要访问的地址（自动变址）。除了立即数寻址、直接寻址、间接寻址等常见寻址方式之外，DSP 支持一些特殊的寻址方式。例如，为了降低卷积、自相关算法和 FFT 算法的地址计算开销，多数 DSP 支持循环寻址和位反序寻址。

5. 特殊高效的 DSP 指令 DSP 指令集设计了一些特殊的 DSP 指令用于专门的数字信号处理操作。这些指令充分利用了 DSP 的结构特点，提高了指令执行的并行度，从而大大提高了完成操作的效率。例如，TMS32010 中的 DMOV 就是一个特殊的 DSP 指令，它主要用来完成数据移位功能，在数字信号处理中，延迟操作非常重要，这个延迟操作就是由 DMOV 来实现的。TMS32010 中的另一个特殊指令是 LTD，它在一个指令周期内完成 LT、DMOV 和 APAC 这三条指令。LTD 和 MPY 指令可以将有限脉冲响应（FIR）滤波器抽头计算从 4 条指令降为 2 条指令。

6. 零开销循环 为适应 DSP 算法的自身特点，多数 DSP 均有专门的支持循环结构的硬件。所谓“零开销（Zero Overhead）”，是指循环技术、条件转移等循环机制由专门硬件控制，而处理器本身不用花费任何时间去处理循环，从而有效地提高了处理器的有效利用率和完成操作的速度。

7. 接口方便 DSP 具有多种接口，可与各种外设方便地连接。数字化是 21 世纪最重要的技术特征之一，由于 DSP 系统与其他数字化技术为基础的系统或设备都是相互兼容的，这不仅能丰富和完善 DSP 自身的功能、使得使用方便，而且与这种高效的数字系统结合使用会大大缩短高科技产品的开发周期，延长高科技产品的使用寿命，使产品在竞争中处于优势地位，取得良好的经济效益。

1.3 DSP 应用系统的开发设计流程

DSP 系统具有精度高、可靠性强、集成度高等特点，这就给 DSP 系统的开发提出了更高的要求。各个 DSP 系统的设计流程和设计内容根据具体的要求会有很多不同的地方，处理

数据量较大的 DSP 系统可能需要在设计前进行算法的仿真和分析,复杂的系统设计还会包括模拟信号的发生器或者相应的接口,而简单的系统设计只包括数字电路的设计,有的 DSP 系统借助 DSP 厂商或者第三方提供的通用硬件平台,只需要编写软件应用程序即可。

但就 DSP 系统的一般典型设计而言,DSP 系统的开发从总体上分为硬件开发和软件开发两个方面,具体的硬件开发和软件开发的详细过程,我们将分别在本书的第 5 章和第 6 章中具体介绍,本小节主要是从总体上介绍 DSP 系统开发的一个整体设计流程,使读者能对 DSP 系统的开发设计有一个整体的初步概念。

1.3.1 DSP 系统开发准备

DSP 系统的开发需要一定的软、硬件环境,对开发人员的软、硬件知识也有一定的要求。在开始阶段,可能由于设计者对 DSP 不熟悉,会遇到不少问题,但只要经过不断的学习和实践,就能够设计出高性能的 DSP 系统。

1. 软、硬件环境 DSP 系统开发的软件环境应该包括针对特定 DSP 芯片的汇编器 (Assembler)、链接器 (Linker)、编译器 (Compiler)、模拟器 (Simulator)、仿真软件 (Emulator)、固化代码生成程序、库管理程序等可执行文件,还应包括基本的算法或函数库、C 语言库、C 头文件等。大部分 DSP 芯片设计厂商在推出芯片的同时,都会提供集成化的开发环境,将上述程序集成在一个窗口环境下,如 TI 公司的 CCS,AD 公司的 Visual DSP++ 等。

硬件环境包括 DSP 仿真器、示波器和逻辑分析仪。其中,最主要的就是 DSP 仿真器,各 DSP 厂家为不同的 DSP 准备了不同的仿真器。通过仿真器,设计者可以对电路板上的 DSP 进行程序加载、单步/全速调试、查看等进行操作。示波器主要用于观察 DSP 电路板的各路引脚状态,在软、硬件调试中用于确定电路板的运行状况。逻辑分析仪主要用于测试 DSP 和外围电路的时序逻辑。

2. 设计者知识储备 DSP 芯片厂商多,每个厂商的 DSP 都有不同特点,而且每个厂商又拥有不同系列的 DSP 芯片,每年都会推出新的产品。一方面,设计者要从面上了解各个厂商各个系列的 DSP 芯片的特点;另一方面,设计者还要及时跟踪和掌握 DSP 发展的新技术,并且从点上对主流 DSP 芯片的应用有比较深入的了解。这样,在具体开发一个系统时,设计者才能根据系统的特点,选择最合适的芯片及其外围电路。另外,在各 DSP 厂商的网站上,可以浏览其产品介绍,还能下载到系统级别的解决方案,以备设计者参考。

DSP 芯片主要应用于数字信号的实时处理,这就要求设计者应当具有扎实的数字信号处理的理论知识,了解各种常用的信号处理算法,并且能够对数字信号处理算法结果进行评估和比较。

DSP 系统不仅仅由一片 DSP 芯片构成,还包括周围的 A/D 转换器、D/A 转换器、CPLD (复杂可编程逻辑器件)、FPGA (现场可编程门阵列)、Flash 存储器和 RAM 等芯片。设计者对于这些外围电路芯片的特点也应深入了解,因为如果 DSP 芯片和这些芯片在读写速度或者电气特性上出现不匹配的话,会影响到整个系统的性能。

对于高速的数字、模拟电路设计,设计者应当具备相当的知识和经验。在设计中,应当能采取恰当的抗干扰措施。

最后,一个实际的 DSP 系统的调试总要经过多次反复,需要设计者和调试者有足够的耐心、坚强的意志,设计者必须不断地纠正开发、测试中出现的错误,这样才能取得真正的

进步。

1.3.2 DSP 系统整体开发流程

DSP 系统整体设计框图如图 1-3 所示。

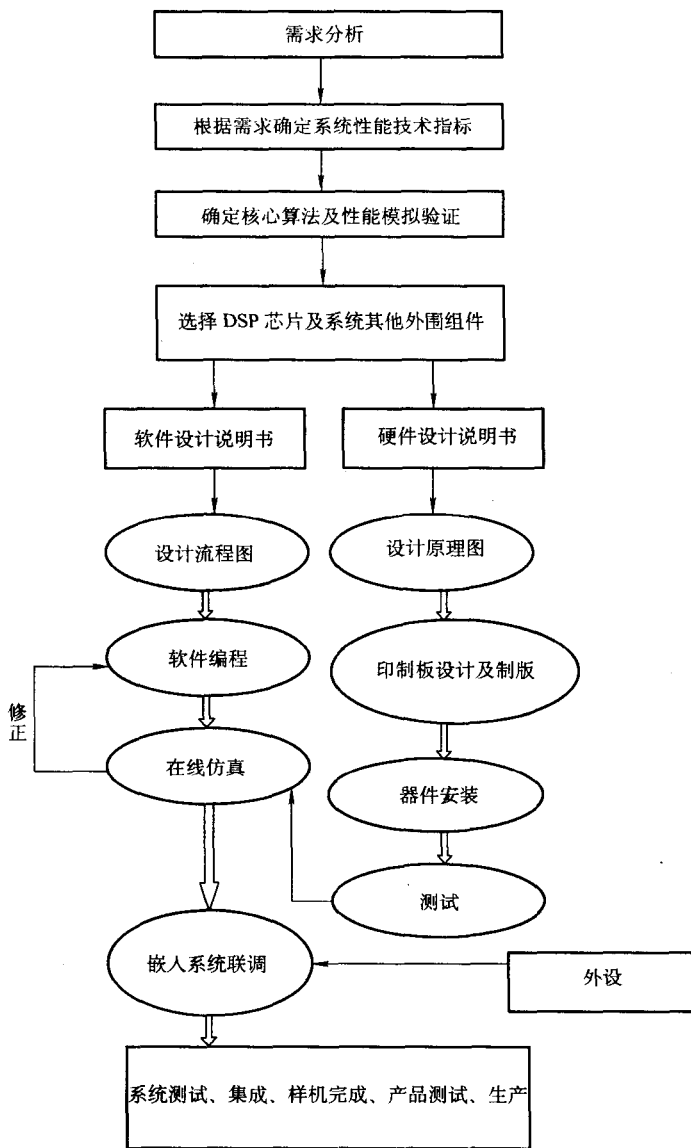


图 1-3 DSP 系统整体设计流程

1. DSP 系统的需求分析 在进行 DSP 系统实质设计之前, 首先要分析 DSP 系统的需求, 明确设计任务。在需求分析文档中, 应当将系统要求的功能准确、清楚地描述出来。描述的方式可以是人工语言, 也可以是系统框图或者算法描述。DSP 系统的需求分析中要考虑以下

几个重要指标:

(1) 实时性。系统是用于实时信号的处理(如语音通信),还是用于非实时信号的处理(如某些静态图形的加工处理),对系统的要求是完全不同的。

(2) 稳定性。和其他硬件系统一样,DSP系统是应用于特定的环境中的,在选择DSP芯片以及外围电路芯片的时候要考虑到系统的应用场合,原则上,星载系统采用宇航级,军事应用采用军品级,工业场合选用工业级器件,民用选用商品级。

(3) 算法的复杂度。算法越复杂,系统的处理性能往往越好,但却对系统处理数据的速度提出了更高的要求。

(4) 数据量的大小。由数据量及程序的长短决定是否需要扩展片外RAM以及扩展容量。

(5) 成本要求。在军事和航天用途中,为了提高性能、增加可靠性并留有一定的发展余地,往往尽量采用高性能DSP,甚至不计成本。而在工业和消费领域中,为了保持最终产品在市场上的竞争力,往往要寻找性价比最好的产品。

(6) 系统类型。系统可以分为数据处理型和控制型,它们对于DSP芯片以及输入输出的要求是不同的。

(7) DSP系统处理的模拟带宽。根据这个带宽,选择合适的A/D采样率,并由该采样率完成该系统所需最复杂算法所需的时间及实时程度来判断系统是否合格。

(8) 精度要求。由系统所要求的精度决定是16位还是32位,是定点运算还是浮点运算。

(9) 开发便利性。为了方便开发仿真,最好选用带联合测试工作组(Joint Test Action Group, JTAG)硬件仿真接口的DSP芯片,既方便开发,又便于以后生产中的测试。

2. 确定系统技术性能指标 在设计DSP系统之前,首先根据系统的目标要求确定系统的技术性能指标、信号处理的要求,这些技术性能指标通常可用数据流程图、正式的符号、数学运算序列或自然语言来描述。结合DSP系统设计,这些技术指标主要包括:

(1) 由信号的频率决定的系统采样频率。

(2) 由采样频率完成最复杂的算法所需最大时间及系统对实时程度的要求判断能否完成工作。

(3) 由系统要求的精度决定是16位还是32位,是定点运算还是浮点运算。

(4) 由数据量及程序的长短决定片内RAM的容量,以及是否需要扩展片外RAM及需扩展的片外RAM的容量。

(5) 根据系统计算对输入输出的要求。

3. 确定核心算法及性能模拟验证 一般而言,为了实现系统的最终目标,需要对输入的信号进行适当的处理,而不同的处理方法将会导致不同的系统性能,因此这一步必须确定最佳的处理方法,以便得到最佳的系统性能。实现过程是,首先应对一个实时数字信号处理的任務选择一种方案和多种算法,用计算机高级语言(如C、Matlab等工具)验证算法能否满足系统的性能指标,然后从多种信号处理算法中找出最佳或准最佳算法。DSP系统所选用的算法无非是各种通用算法的组合和改进。改进算法不是DSP设计师的任务,DSP系统设计师应尽量选用成熟可靠、经过时间考验的算法。支持各种通用算法的DSP模拟软件在市场上已有不少。下面介绍的是其中一些优秀的产品:

(1) SPW 工作站: Cadence 公司产品, 含有大量的算法库, 对通信系统设计尤其适用。

(2) Matlab 工具包: 该软件在国内高校中已经开始流行, 该软件的矢量矩阵处理功能很强, 最新版本中信号处理功能也已大大加强。

(3) Dalisp 软件: 优秀的信号处理软件包, 网上有免费的高校版, 读者可自行下载使用。

用如上的工具模拟挑选出合适的算法组合后, 设计师就可应用高级语言在 PC 上进行实际编程验证, 设计出 DSP 的软件处理流程, 并给出最终可实现的软件需求分析。

4. 选择 DSP 芯片及其他外围组件 DSP 是处理系统的核心, 设计 DSP 应用系统, 选择 DSP 芯片是其中非常重要的一个环节。只有选定了 DSP 芯片才能进一步设计其外围电路及系统的其他电路。总之, DSP 芯片的选择要根据设计应用系统的需要进行确定。一般来说, 应该从应用的具体要求出发, 来选择合适的 DSP 型号。

5. 软、硬件设计 当选定 DSP 芯片及其外围组件以后, 就可以对 DSP 系统进行设计了。一般而言, DSP 系统的设计可以分为软件设计和硬件设计两部分。软件是指用 DSP 的汇编语言或者通用的高级语言 (如 C 语言) 编写出来的, 用于实现信号处理功能的程序。这些程序可以放在 DSP 片内或片外存储器中, 在程序工作时, DSP 会执行与 DSP 外围设备传递数据或互相控制的指令, 因此 DSP 的软件、硬件设计和调试是不可分的。

6. 嵌入式系统联调 当系统进入联调阶段, 在软件平台上, DSP 系统通过仿真器和计算机通信, 从而实现软件调试的各种方法, 如断点设置、观察寄存器、单步执行等; 在硬件平台上, 可以借助示波器或者逻辑分析仪等工具对系统的信号进行测量和分析。

当软、硬件联调满足要求以后, 还需要将程序固化到系统中, 即利用 DSP 厂家提供的软件将程序生成固定的格式, 写入到 EPROM 或者 Flash 存储器中, 这时 DSP 系统就可以脱离仿真器独立运行了。

7. 系统集成测试和生产 DSP 系统在可以独立运行以后, 还应该继续进行一系列的测试, 以使之满足工业化生产的需求。在确定 DSP 系统可以稳定地运行以后, 才可以投入生产。

1.4 DSP 概述

1.4.1 DSP 芯片

数字信号处理是一种将现实世界中的真实信号 (专业术语称之为连续信号) 转换为计算机能够处理的离散信号的过程。比如人们说话的声音, 这就是一个连续信号, 此外, 现实生活中还有很多这样的信号, 比如光、压力、温度等信号。这些信号通过一个模拟向数字的转换过程 (称之为 A/D 转换), 变成数字信号送给处理器, 进行数字计算, 处理结束后, 再把结果通过数字向模拟的转换过程 (称之为 D/A 转换) 重新变成连续信号。其实, 用一般的通用微处理器就可以完成这些工作, 但是面临的问题是满足如此高的计算速度的同时就很难保证耗电量很低, 更难保证价格足够便宜。因此, 另一种微处理器应运而生: 数字信号处理器 (DSP)。

DSP 芯片是一种具有特殊结构的微处理器。DSP 芯片的内部采用程序存储空间和数据存