



国外经典教材·电子信息

PEARSON
Prentice
Hall

Digital Signal Processors: Architectures, Implementations, and Applications

数字信号处理器 ——体系结构、实现与应用

郭森楸 颜允圣 著
贾洪峰 译



清华大学出版社

国外经典教材·电子信息

数字信号处理器

——体系结构、实现与应用

郭森琳 颜允圣 著

贾洪峰 译

清华大学出版社
北京

内 容 简 介

本书分为3个主要部分:体系结构、实现和应用,旨在帮助读者理解DSP处理器的体系结构和编程、实时DSP系统与实际应用程序的设计,如何利用定点和浮点处理器实现DSP算法等。所介绍的处理器包括:TMS320C2000, TMS320C54x, TMS320C55x, TMS320C62x, TMS320C64x, TMS320C3x和TMS320C67x。

本书可作为高年级相关专业本科生的教材。由于本书强调DSP实现、实验和应用,所以也可以作为参考书,供那些希望学习DSP概念以及在工作中开发实时DSP应用的工程人员使用。

在开始学习本书之前,读者应基本掌握信号与系统、C语言和汇编语言等知识。

Simplified Chinese edition copyright © 2005 by PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.

Original English language title from Proprietor's edition of the Work.

Original English language title: Digital Signal Processors: Architectures, Implementations, and Applications, 1st Edition by Sen M. Kuo, Woon-Seng Gan, Copyright © 2005

EISBN: 0-13-035214-4

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Pearson Education, Inc.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macao).

本书中文简体翻译版由Pearson Education授权给清华大学出版社在中国境内(不包括中国香港、澳门特别行政区)出版发行。

北京市版权局著作权合同登记号 图字:01-2004-3854

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有Pearson Education(培生教育出版集团)激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

数字信号处理器——体系结构、实现与应用/郭森琳,颜允圣著;贾洪峰译. —北京:清华大学出版社,2005.6
(国外经典教材·电子信息)

书名原文: Digital Signal Processors: Architectures, Implementations, and Applications

ISBN 7-302-10985-0

I. 数… II. ①郭… ②颜… ③贾… III. 数字信号—信号处理—教材 IV. TN911.72

中国版本图书馆CIP数据核字(2005)第048939号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

文稿编辑: 汤涌涛

封面设计: 久久度文化

印 刷 者: 北京牛山世兴印刷厂

装 订 者: 三河市新茂装订有限公司

发 行 者: 新华书店总店北京发行所

开 本: 185 × 260 印 张: 33.5 字 数: 812 千字

版 次: 2005 年 6 月第 1 版 2005 年 6 月第 1 次印刷

书 号: ISBN 7-302-10985-0/TP · 7284

印 数: 1 ~ 4000

定 价: 59.00 元

出版说明

近年来,我国的电子信息产业发展迅速,力争要成为“21 世纪的世界制造中心”。这就迫切需要我国的高校能够培养出大量符合企业需求的高素质电子信息类人才。但是,目前传统的电子信息类教材已经落后于时代发展对电子信息教育的要求,急需一批门类齐全、具有国际水平的经典教材。引进国外的优秀电子信息教材,一方面可以了解和吸收国际现今的教学思想和教学方法,推动国内高校的课程改革和教学国际化进程,乃至对国内重点大学建设国际一流大学都能够提供宝贵的借鉴作用;另一方面能够培育出更多具有国际水准的电子信息类人才,提高我国电子信息产业的核心竞争力。为此,清华大学出版社从全球最大的高等教育出版机构——培生教育出版集团(Pearson Education Group)引进了这套“国外经典教材·电子信息”教材。

在该套教材的引进和出版过程中,我们邀请了一大批国内高校电子信息类专业的知名教授和相应领域的专家,与我们共同成立了专门的教材编审委员会,根据国内高校电子信息各专业的课程体系和培养方向,分期分批地从 Pearson 出版的电子信息系列教材中挑选出质量高、针对性强、适合国内教学的优秀教材。确定选题后,编委会委员直接参与或组织教材的翻译和审校,以确保教材内容的高质量和适应性。另外,为了更好地服务于教学和增强立体化教学的力度,我们在引进教材的同时也引进了与教材相配套的教学资料。

该套教材的范围是电子信息 and 电子工程学科所属各专业的基础专业课和专业课,读者对象是对应专业的本科生,同时兼顾相关工程学科各专业的本科生或研究生。该套教材既可作为相应课程的教材或教学参考书,也适于相应技术领域的工程师和技术人员参考或自学。

由于我们自身能力有限,该套教材在使用过程中很可能还会发现一些缺憾或问题,欢迎使用过的广大师生向我们提出意见或建议。同时,也非常期望广大读者向我们推荐优秀的国外电子信息类教材,同我们一起建设适合我国高等院校电子信息教学的优秀教材。

清华大学出版社

前 言

在当今的工程领域中,采用通用数字信号处理器进行实时数字信号处理(Digital Signal Processing, DSP)是一个非常热门的主题,并且是非常富有挑战性的工作。实时DSP为设计和实现现实应用程序中的各种DSP算法提供了一种有效方式。事实上,许多大学和工业公司目前正在从事实时DSP的研究、教育和开发。近年来,随着DSP在各种应用中的深入,对高性能数字信号处理器的需求已经迅速扩大。对于今天的学生和实践工程人员来说,不仅要掌握DSP理论,还要掌握实时DSP系统设计与实现所涉及的技术,这一点已经变得越来越重要。

本书为读者提供了一种实用方法,用于理解DSP处理器的体系结构和编程、实时DSP系统与现实应用程序的设计,以及利用定点和浮点处理器实现DSP算法,这些处理器包括:TMS320C2000、TMS320C54x、TMS320C55x、TMS320C62x、TMS320C64x、TMS320C3x和TMS320C67x。本书可作为高年级大学生的教材。由于本书强调DSP实现、实验和应用,所以它也可以作为桌面参考书,供那些希望学习DSP概念和在工作中开发实时DSP应用的工程人员使用。应具备的最低预备知识为:基本理解信号与系统的概念、C语言和汇编语言。

本书对实时DSP技术进行了概述。为了有效说明实时DSP应用的实用方面,在DSP算法的设计、分析与实现中采用了MATLAB和Simulink。此外,TMS320C54x和TMS320C55x的Code Composer Studio (CCS)也被用于实验室实验、项目和应用。这些非常有用的工具有效地说明了实时DSP的概念,并且在信号处理理论与实时实现之间架起了一座桥梁。通过完成实践练习和习题,读者可以获得关于实时实现问题的实践知识。在使用高级DSP体系结构进行快速软件开发和维护时,强调了C程序与汇编程序的混合编程。

本书在实践练习中采用两级步骤。首先,我们介绍软件工具,并一步步地指导读者以浮点和定点格式进行设计、仿真、检验,以及最终的程序开发。接下来,我们提供额外的练习和习题,以强化读者对这些主题的理解。本书列出了实现许多DSP示例和应用程序所用到的MATLAB、Simulink、C和汇编程序,可以在以下站点找到这些程序:www.prenhall.com/kuo和www.ntu.edu.sg/home/ewsgan/book.htm。在此站点还可以找到几个实际数据文件,在本书所介绍的一些实际应用中会用到这些文件。Web应用使作者可以在软件和/或其接口更新时,对本书所引用的程序和数据文件进行连续更新(针对选用本教材授课的教师,作者还提供了相应的教学幻灯片。教师可通过ewsgan@ntu.edu.sg与作者取得联系。——编者注)。

本书分为三个主要部分:应用、体系结构和实现。(1)应用部分(第1~3章)介绍了DSP算法与应用、处理器体系结构、外围设备组件和实时实现问题。(2)体系结构部分(第4~5章)详细描述了定点处理器与浮点处理器,重点介绍了软件开发。(3)实现部分(第6~9章)在定点和浮点处理器中常用的DSP算法的设计、仿真、检验以及最终实现都采用了自顶向下方法。在附录B中给出了采用MATLAB、C语言和CCS的附加实践实验及实际应用。附录C中介绍了用于解决特定应用的外围设备与DSP处理器之间连接的相关编码问题与设计示例。

第1章概述了实时DSP系统和应用、处理器体系结构、软件与硬件开发问题和系统设计的考虑事项。这一章为后续各章设计了框架。最后一节中的实验逐步介绍了一种集成软件开发工具——CCS。第2章回顾了一些基本的DSP概念,使用它们来解释后续各章中给出的主题。

这一章仅总结要点,因此,已经熟悉DSP理论的读者可以略过第2章。但是,这一章提供了许多有用的示例,用于介绍使用MATLAB和C语言来解决DSP问题,还介绍了CCS的一些高级特色,这些特色对于后续各章给出的实验非常有用。第3章介绍了DSP算法的实时实现。解释了几个实际主题,如定点数和浮点数的表示与运算、满足实时约束条件的编程问题,以及与DSP处理器相连的外围设备。此外,利用一种在CCS中进行定点C编程的实用方法说明了一些重要的定点实现问题。

第4章与第5章介绍了几种定点和浮点DSP处理器的体系结构、指令集、编程和系统设计问题。这两章提供了关于如何对不同DSP处理器进行编程的综合理解,它们强调了各种处理器的优点与弱点。示例和实验说明了定点和浮点处理器之间的重要区别。由于本书所用主要DSP处理器指令集的复杂性与长度原因,所以未在附录中包含该指令集。但是,可以很容易地在CCS的“Help”菜单中以及附录D中所列的Web站点中找到这些指令集。

第6~9章介绍了一些常用DSP算法的设计、分析和软件实现,其中包括采用定点和浮点DSP处理器的有限长单位冲激响应(FIR)滤波、无限长单位冲激响应(IIR)滤波、快速傅里叶变换(FFT)和自适应滤波。在不同处理器上实施这些算法强调了对不同处理器进行编程所涉及的许多重要特征和特色。需要着重指出的是,已经开发有一套系统方法,用于指导读者完成DSP软件开发的各个阶段。我们首先在通用计算机上进行定点C编程,然后在C5000处理器上进行浮点C编程、利用C5000内联函数进行定点C编程和汇编编程。最后,我们利用定点C调用C5000 DSP库中的一个汇编优化例程。

附录A中给出了一个快速参考指南,用于说明带有DSP工具箱的MATLAB的应用和支持DSP模块的Simulink的应用。这个附录帮助读者熟悉这些非常有用的MATLAB信号处理工具。我们还将介绍两个功能强大的交互式工具——信号处理工具和滤波器设计与分析工具,它们对于定点和浮点实现中的DSP算法设计与评估都非常有效。附录B提供了附加实践实验、实践应用和深入的设计项目。附录C讨论了涉及带有外围组件的DSP处理器的集成系统设计问题。附录D列出了许多Web站点,它们提供了关于DSP的有用信息。

本书可以在侧重点不同的课程中使用。DSP算法与应用课程可以从附录A开始,然后学习第1章和第3章,最后学习第6~9章中的材料和实验。DSP体系结构课程可以重点学习第1、3、4、5章和附录C。通过第6~9章末尾以及附录B中的实验也可以更深入地理解DSP处理器。最后,侧重于实时DSP系统设计的课程可以从第1、3章开始,然后重点介绍第4、5章的C5000 DSP体系结构,最后学习第6、9章和附录B、附录C中使用C5000处理器设计与实现DSP算法。每章末的习题分为两组:习题A包括使用MATLAB、C和CCS的计算机相关习题,习题B包括巩固基本DSP原理的传统书面习题。

本书的目的是以实用方式涵盖实时DSP的全部内容,而不是过多地依赖于数学的应用。本书的独特之处在于它提供了使用C程序和汇编程序进行的浮点仿真和定点实时实现之间的循环链接。读者可以选择不同的主题、处理器和实验,并可以深入理解关于实时DSP的体系结构、实现和应用的重要概念。

在本书中,我们在示例、实验和习题中使用了几种软件工具。大学和工业公司经常用到这些软件工具。下面列出了这些工具及其各自的版本:

- Texas Instruments公司的Code Composer Studio™, 2.10版
- The MathWorks 公司的DSPBlockset, 5.0 版

- The MathWorks公司的Embedded Target for Texas Instruments
- MS320C6000™ DSPPlatform, 1.0 版
- The MathWorks公司的Filter Design Toolbox, 2.2 版
- The MathWorks公司的Fixed-Point Blockset, 4.0 版
- The MathWorks公司的MATLAB®, 6.5 版
- The MathWorks公司的MATLAB Link for Code Composer Studio, 1.0 版
- Microsoft公司的Microsoft® Visual C++®, 6.0 版
- The MathWorks公司的Real-Time Workshop®, 5.0 版
- The MathWorks公司的Signal Processing Toolbox, 6.0 版
- The MathWorks公司的Simulink®, 5.0 版
- The MathWorks公司的Wavelet Toolbox, 2.2 版

也可以使用上述软件的不同版本, 当然, 在结构、选项和设置上面可能会存在一些不同。

有一句中国古语说: “不闻不若闻之, 闻之不若见之, 见之不若知之, 知之不若行之。”

本书强调“见”与“行”。事实上, 我们愿意将本书叫做实时DSP工作手册, 因为它提供了各种机会, 以支持读者使用最有效的工具来研究DSP。我们希望读者乐意完成这些实验和练习, 并希望他们能够体验数字信号处理器对现实应用的强大功能。在每章末的参考文献中可以找到关于DSP的更为严密的数学方法, 以及一些问题的有关附加细节。

本书包含许多示例、实验和练习题, 其中涉及MATLAB、Simulink、C语言和TMS320Cx汇编程序。我们已经尽最大努力来保证书中所有程序的准确性。我们欢迎读者与我们联系, 告知书中的任何错误或分享任何意见, 我们的电子信箱为: kuo@ceet.niu.edu 和 ewsgan@ntu.edu.sg。

郭森琳
颜允圣

目 录

第 1 章 数字信号处理系统导论	1	2.3 数字系统	35
1.1 数字信号处理简介	1	2.3.1 线性时不变系统	35
1.2 数字信号处理系统与应用	3	2.3.2 有限长单位冲激 响应滤波器	37
1.2.1 数字信号处理系统	3	2.3.3 无限长单位冲激 响应滤波器	39
1.2.2 TMS320 系列	4	2.4 频率分析	42
1.2.3 数字信号处理应用	5	2.4.1 离散时间傅里叶变换	42
1.3 数字信号处理器体系结构	6	2.4.2 离散傅里叶变换	45
1.3.1 引言	6	2.4.3 离散傅里叶变换的性质	45
1.3.2 中央处理单元运算	7	2.4.4 快速傅里叶变换	46
1.3.3 存储器配置	9	2.5 随机信号处理	48
1.3.4 外围设备与输入/输出	11	2.5.1 数字随机信号	48
1.4 软件开发	12	2.5.2 时域处理	49
1.4.1 指令集	13	2.6 实验	52
1.4.2 汇编程序	13	2.6.1 简单的无限长单位 冲激响应滤波器	52
1.4.3 C 程序	14	2.6.2 软件开发	54
1.4.4 C 语言与汇编语言 的混合编码	14	2.6.3 信号处理工具	55
1.4.5 软件开发工具	15	2.6.4 使用 Code Composer Studio 测试 C 程序	60
1.5 硬件问题	17	习题	65
1.5.1 硬件选择	17	参考文献	68
1.5.2 硬件配置	17	第 3 章 实现的考虑因素	70
1.5.3 硬件工具	17	3.1 引言	70
1.6 系统考虑事项	18	3.2 数据表示与运算	70
1.6.1 数字信号处理器的选择	18	3.2.1 定点数与运算	70
1.6.2 采样与量化	19	3.2.2 浮点运算	75
1.6.3 数模转换	20	3.2.3 定点格式与浮点格式对比	78
1.6.4 编码译码器	20	3.3 有限字长效应	79
1.6.5 速度优化	21	3.3.1 输入量化	80
1.7 实验	21	3.3.2 系数量化	82
1.7.1 使用汇编器和 链接器-命令文件	22	3.3.3 溢出与解决方法	84
1.7.2 创建一个项目	24	3.3.4 舍入与舍位	86
1.7.3 连编项目	25	3.4 编程问题	88
1.7.4 调试程序	26	3.4.1 寻址模式	88
1.7.5 查看存储器和图形	27	3.4.2 流水线概念	91
1.7.6 使用断点和配置器	28	3.4.3 指令高速缓存	92
习题	29	3.4.4 硬件与软件中断	92
参考文献	30	3.5 实时实现的考虑因素	94
第 2 章 数字信号处理基础	31	3.5.1 信号转换器	94
2.1 数字信号与操作	31	3.5.2 流处理	94
2.1.1 基本信号	31	3.5.3 块处理	95
2.1.2 基本运算	32	3.5.4 矢量处理	96
2.2 变换	34	3.5.5 基准测试	96
2.2.1 定义	34	3.6 硬件接口	98
2.2.2 z 变换的性质	35		

3.6.1	外部存储器接口	98	4.6.5	舍入	197
3.6.2	计时器和主时钟	99	4.6.6	提取保护位	198
3.6.3	串行端口连接	100	4.6.7	一个乘法特例	198
3.6.4	直接存储器存取控制器	101	习题	199	
3.6.5	并行端口连接	102	参考文献	200	
3.6.6	主机端口连接	102	第 5 章 浮点数字信号处理器	201	
3.6.7	多处理技术	103	5.1	引言	201
3.6.8	电源调节器	104	5.2	TMS320C3x	202
3.6.9	仿真器互连标准	104	5.2.1	体系结构概述	202
3.7	实验	105	5.2.2	中央处理器	203
3.7.1	利用 MATLAB 的实验	105	5.2.3	寄存器组织与寻址模式	205
3.7.2	采用定点 C 的实验	110	5.2.4	指令集	209
3.7.3	采用 C5000 CCS 的实验	111	5.2.5	编程考虑因素	212
习题	116	5.2.6	系统问题	215	
参考文献	118	5.2.7	一个应用: 伪随机数 产生器	217	
第 4 章 定点数字信号处理器	120	5.3	TMS320C67x	218	
4.1	引言	120	5.3.1	体系结构问题	219
4.1.1	源语句格式	120	5.3.2	指令集	221
4.1.2	汇编命令	121	5.3.3	流水线体系结构	222
4.1.3	软件开发过程	122	5.3.4	编程考虑因素	222
4.2	TMS320C2000	122	5.3.5	实时实现	223
4.2.1	体系结构概述	123	5.4	实验	226
4.2.2	中央处理器	124	5.4.1	加法与减法	226
4.2.3	程序控制	131	5.4.2	乘法与除法	228
4.2.4	编程问题	133	5.4.3	有限长单位冲激响应滤波	229
4.2.5	系统问题	135	5.4.4	无限长单位冲激响应滤波	230
4.2.6	一个应用: 锁相环	138	5.4.5	使用 DSPLIB	231
4.3	TMS320C54x	140	习题	233	
4.3.1	体系结构概述	140	参考文献	234	
4.3.2	寻址模式	143	第 6 章 有限长单位冲激响应滤波	236	
4.3.3	指令集	147	6.1	有限长单位冲激响应滤波器	236
4.3.4	编程考虑因素	153	6.1.1	定义	236
4.3.5	系统问题	154	6.1.2	滤波器特性	237
4.4	TMS320C55x	157	6.1.3	滤波器结构	240
4.4.1	体系结构概述	157	6.1.4	滤波器设计	242
4.4.2	中央处理器	158	6.1.5	有限字长效应	247
4.4.3	寻址模式	161	6.1.6	有限长单位冲激 响应滤波器的实现	248
4.4.4	指令集	167	6.2	用 MATLAB 设计有限长 单位冲激响应滤波器	250
4.4.5	编程考虑因素	169	6.2.1	滤波器设计与工作 工具简介	250
4.4.6	C 程序的优化	173	6.2.2	有限长单位冲激响应 滤波器的设计与分析	251
4.4.7	系统问题	175	6.2.3	量化过程与分析	254
4.5	TMS320C62x 与 TMS32064x	177	6.3	定点实现	258
4.5.1	体系结构概述	177	6.3.1	用 TMS320C2000 实现	258
4.5.2	存储器系统	180	6.3.2	用 TMS320C54x 实现	259
4.5.3	外部存储器寻址	181	6.3.3	用 TMS320C55x 实现	262
4.5.4	指令集	182	6.3.4	用 TMS320C62x 实现	266
4.5.5	编程考虑因素	187	6.3.5	用 TMS320C64x 实现	268
4.5.6	系统问题	191			
4.6	实验	192			
4.6.1	表示不同的 Q 格式	192			
4.6.2	系数量化	194			
4.6.3	溢出处理	195			
4.6.4	缩放和饱和模式	196			

6.4 浮点实现	269	7.5 浮点实现	322
6.4.1 用 TMS320C3x 实现	269	7.5.1 用 TMS320C3x 实现	322
6.4.2 用 TMS320C67x 实现	270	7.5.2 用 TMS320C67x 实现	323
6.5 应用: 多速信号处理	270	7.6 应用: 双音多频生成与检测	324
6.5.1 内插	271	7.6.1 双音多频音频信号的生成	324
6.5.2 抽选	272	7.6.2 双音多频信号检测	326
6.5.3 采样速率转换	273	7.7 实验	330
6.5.4 数字信号处理实现	274	7.7.1 用 MATLAB 设计	330
6.6 实验	275	7.7.2 用浮点 C 实现	332
6.6.1 用 MATLAB 设计一个 带通有限长冲激响应滤 波器	276	7.7.3 用定点 C 实现	333
6.6.2 用浮点 C 实现	278	7.7.4 用 C5000 处理器的 定点 C 实现	334
6.6.3 用定点 C 实现	280	7.7.5 用带有内联函数的 定点 C 实现	336
6.6.4 采用 C5000 处理器 的定点实现	281	7.7.6 用 C5000 处理器 汇编语言实现	336
6.6.5 用带有内联函数的 定点 C 实现	283	7.7.7 用 C5000 数字信号 处理库实现	339
6.6.6 用 C5000 处理器的 汇编语言实现	284	习题	341
6.6.7 用 C5000 数字信号 处理库实现	286	参考文献	345
习题	288	第 8 章 快速傅里叶变换	346
参考文献	291	8.1 离散傅里叶变换简介	346
第 7 章 无限长单位冲激响应滤波	292	8.1.1 离散傅里叶变换与离散 傅里叶逆变换的计算量	346
7.1 无限长单位冲激响应滤波器	292	8.1.2 旋转因子的性质	347
7.1.1 定义	292	8.2 快速傅里叶变换算法	348
7.1.2 滤波器特性	293	8.2.1 按时间抽取基 2 快速 傅里叶变换	349
7.1.3 滤波器结构	295	8.2.2 按频率抽取的基 2 快速 傅里叶变换	351
7.1.4 无限长单位冲激响应 滤波器的稳定性	300	8.2.3 其他基与混合基快速 傅里叶变换算法	353
7.1.5 有限长长效应	301	8.2.4 快速傅里叶逆变换	353
7.2 无限长单位冲激响应 滤波器的设计与实现	301	8.3 利用 MATLAB 和 C 语言 进行分析和实现	354
7.2.1 模拟-数字滤波器设计	302	8.3.1 离散傅里叶变换与快速 傅里叶变换的优点与弱点	354
7.2.2 算法滤波器设计	308	8.3.2 离散傅里叶变换的性质	354
7.2.3 用于无限长单位冲激响应 滤波器设计的 MATLAB 函数	308	8.3.3 MATLAB 实现	358
7.2.4 无限长单位冲激响应 滤波器的实现	310	8.3.4 C 实现	360
7.3 使用 MATLAB 来设计无限长 单位冲激响应滤波器	311	8.4 实现考虑因素	361
7.3.1 无限长单位冲激响应 滤波器设计与分析	311	8.4.1 频率分辨率与加窗	361
7.3.2 量化过程和分析	313	8.4.2 计算问题	365
7.4 定点实现	317	8.4.3 有限长效应	367
7.4.1 用 TMS320C2000 实现	317	8.4.4 使用 MATLAB 评估量化 快速傅里叶变换	368
7.4.2 用 TMS320C54x 实现	318	8.5 定点实现	371
7.4.3 用 TMS320C55x 实现	319	8.5.1 用 TMS320C54x 实现	371
7.4.4 用 TMS320C62x 实现	321	8.5.2 用 TMS320C2000 实现	374
7.4.5 用 TMS320C64x 实现	322	8.5.3 用 TMS320C55x 实现	375
		8.5.4 用 TMS320C62x 实现	376
		8.5.5 用 TMS320C64x 实现	377

8.6 浮点实现	378	9.8.2 用浮点 C 实现	434
8.6.1 用 TMS320C3x 实现	378	9.8.3 用定点 C 实现	435
8.6.2 用 TMS320C67x 实现	379	9.8.4 用 C5000 处理器的 定点 C 实现	435
8.7 应用	379	9.8.5 用带有内联函数 的定点 C 实现	437
8.7.1 快速卷积	379	9.8.6 用 C5000 处理器 的汇编语言实现	438
8.7.2 功率谱估计	382	9.8.7 用 C5000 数字信号 处理库实现	441
8.7.3 短时傅里叶变换	385	习题	442
8.7.4 小波变换	385	参考文献	444
8.8 实验	387	部分习题答案	445
8.8.1 用 MATLAB 进行谱估计	387	附录 A MATLAB 与 Simulink 简介	448
8.8.2 用浮点 C 实现	388	A.1 使用 MATLAB	448
8.8.3 用定点 C 实现	389	A.1.1 启动	448
8.8.4 用 C5000 处理器 的定点 C 实现	390	A.1.2 有用的语法	449
8.8.5 用带有内联函数 的定点 C 实现	391	A.1.3 绘图	450
8.8.6 用 C5000 数字信号 处理库实现	392	A.1.4 编程	451
习题	394	A.1.5 数据类型	452
参考文献	398	A.1.6 有用的命令	453
第 9 章 自适应滤波	399	A.2 使用数字信号处理工具箱 与交互工具	453
9.1 自适应滤波器简介	399	A.2.1 信号处理工具箱	453
9.2 自适应滤波器结构和算法	400	A.2.2 信号处理工具	457
9.2.1 滤波器结构	400	A.2.3 滤波器设计工具箱	460
9.2.2 自适应算法	402	A.2.4 滤波器设计与分析工具	460
9.3 自适应滤波器的性质	406	A.3 使用 Simulink	463
9.3.1 稳定条件	406	A.4 使用模块组	468
9.3.2 收敛速度	406	A.4.1 DSP 模块组	468
9.3.3 稳态性能	407	A.4.2 定点模块组	470
9.3.4 有限精度效应	407	A.5 MATLAB Link for Code Composer Studio	473
9.4 应用	409	参考文献	475
9.4.1 自适应系统识别	409	附录 B 附加实验与应用	476
9.4.2 自适应逆建模	411	B.1 应用有限长单位冲激响应 滤波器的 3 维声效定位器	476
9.4.3 自适应噪声消除	413	B.1.1 简介	476
9.4.4 自适应预测	415	B.1.2 使用 MATLAB/Simulink 的实验室实验	476
9.5 用 MATLAB 和 C 设计与实现	417	B.1.3 使用 C5000 Code Composer Studio 的实验室实验	476
9.5.1 设计举例	418	B.1.4 附加练习	477
9.5.2 自适应滤波器的量化	421	B.2 使用有限长单位冲激响应 滤波器的图形均衡器	477
9.5.3 用 C 语言实现的 自适应滤波	423	B.2.1 简介	477
9.6 定点实现	425	B.2.2 使用 MATLAB/Simulink 的实验室实验	478
9.6.1 用 TMS320C2000 实现	425	B.2.3 使用 C5000 Code Composer Studio 的实验室实验	478
9.6.2 用 TMS320C54x 实现	426		
9.6.3 用 TMS320C55x 实现	428		
9.6.4 用 TMS320C62x 实现	429		
9.6.5 用 TMS320C64x 实现	429		
9.7 浮点实现	430		
9.7.1 用 TMS320C3x 实现	430		
9.7.2 用 TMS320C67x 实现	431		
9.8 实验	432		
9.8.1 用 MATLAB 设计 自适应滤波器	432		

B.2.4 附加练习.....	478	B.8.2 使用 MATLAB/Simulink 的实验室实验.....	488
B.3 完美重建滤波器组.....	479	B.8.3 使用 C5000 CCS 的实验室实验.....	488
B.3.1 简介.....	479	B.8.4 附加练习.....	489
B.3.2 使用 MATLAB/Simulink 的实验室实验.....	480	B.9 缩放快速傅里叶变换的实现.....	489
B.3.3 使用 C5000 Code Composer Studio 的实验室实验.....	481	B.9.1 简介.....	489
B.3.4 附加练习.....	481	B.9.2 使用 MATLAB/Simulink 的实验室实验.....	489
B.4 应用无限长单位冲激响应 滤波器的数字信号生成器.....	481	B.9.3 使用 C5000 Code Composer Studio 的实验室实验.....	489
B.4.1 简介.....	481	B.9.4 附加练习.....	490
B.4.2 使用 MATLAB/Simulink 的实验室实验.....	482	B.10 声音回响对消.....	490
B.4.3 使用 C5000 Code Composer Studio 的实验室实验.....	482	B.10.1 简介.....	490
B.4.4 附加练习.....	482	B.10.2 使用 C 程序的 实验室实验.....	491
B.5 使用无限长单位冲激响应 滤波器的数字回响.....	482	B.10.3 使用 C5000 Code Composer Studio 的实验室实验.....	491
B.5.1 简介.....	483	B.10.4 附加练习.....	492
B.5.2 使用 MATLAB/Simulink 的实验室实验.....	483	B.11 有源噪声控制.....	492
B.5.3 使用 C5000 Code Composer Studio 的实验室实验.....	484	B.11.1 简介.....	492
B.5.4 附加练习.....	484	B.11.2 使用 MATLAB 的实验室实验.....	493
B.6 使用无限长单位冲激响应 滤波器的参数均衡器.....	484	B.11.3 使用 C5000 Code Composer Studio 的实验室实验.....	493
B.6.1 简介.....	484	B.11.4 附加练习.....	494
B.6.2 使用 MATLAB/Simulink 的实验室实验.....	485	参考文献.....	494
B.6.3 使用 C5000 Code Composer Studio 的实验室实验.....	485	附录 C 数字信号处理器的外围设备编程.....	495
B.6.4 附加练习.....	486	C.1 通过直接存储器访问与模拟接口 电路或 CODER-DECODER 连接.....	495
B.7 快速卷积方法.....	486	C.1.1 初始化多通道缓冲 串行端口.....	495
B.7.1 使用 MATLAB/Simulink 的实验室实验.....	487	C.1.2 初始化直接存储器读取.....	505
B.7.2 使用 C5000 Code Composer Studio 的实验室实验.....	487	C.1.3 举例.....	511
B.7.3 附加练习.....	487	C.2 通过主端口接口同主 处理器进行连接.....	514
B.8 滑动快速傅里叶变换实现.....	487	C.3 未来研究.....	515
B.8.1 简介.....	487	C.4 外围设备编程的趋势.....	516
		参考文献.....	516
		附录 D 有用的站点.....	517

第1章 数字信号处理系统导论

随着数字技术的引入,数字信号处理(DSP)在20世纪60年代获得普及。随着数字硬件速度更快、使用更简单、价格更便宜、可用性更强,数字信号处理已成为信号处理中的一种可选方法。在1979年,英特尔公司提出了第一种可用于DSP的处理器(Intel 2920),它具有专为DSP应用设计的体系结构和指令集。今天有许多由德州仪器公司、摩托罗拉、Analog Devices、Agere、DSP Groups和许多其他公司生产的通用DSP处理器可供商业应用。DSP处理器的价格越来越便宜、功能越来越强大,市场上现实DSP应用获得迅速发展,如高速调制解调器、互联网访问、无线和蜂窝电话、音频和视频播放器和数码相机等。在本书中,我们将重点介绍德州仪器公司出品的TMS320系列DSP处理器。

1.1 数字信号处理简介

信号是一个物理量,它通常是时间、位置、压力等的函数。例如,一个麦克风输出的电压将声压表示为时间的一个函数。在日常生活中经常遇到的信号包括语音、音乐、数据、图像和视频信号。信号处理的目的是传输或存储信号,以增强有用信号分量,并提取信号所携带的有用信息。

在实际应用中,人们所遇到的信号中有很多是模拟信号,这种信号在任何时间均存在,并且可以是任何幅度。例如,记录在盒式磁带上的音乐信号就是模拟信号。为了使用DSP处理器处理这些连续时间信号,必须对其进行采样和量化,以生成一个数值序列。这一离散时间序列的幅度为离散数值,由个数有限的二进制位表示,这种序列称作数字信号。例如,光盘(CD)中所存储的音乐被记录为数字信号。可以在不损失任何信息的情况下,几乎无限制地将数字信号存储在各种存储介质上,如磁盘和光盘。相反,所存储的模拟信号会快速恶化,不能将其恢复到原始形式。

DSP理论建立在离散时间信号和系统的基础上,这是因为它们易于进行数学处理。可以通过对模拟信号进行采样(不进行量化)来获得离散时间信号。离散时间信号在任意离散时刻均存在,并且可以是任意幅度。它们需要用无限个比特来表示每个采样,这不可能用物理数字硬件实现。此外,多数DSP算法最初都是利用通用计算机开发的,对于通用计算机,修改和测试算法的速度比执行算法的速度更重要。为了将这些算法应用于实际应用,应当考虑使用有限字长的影响(见第3章)、硬件、执行速度和算法复杂性。对于一个给定应用,必须使用所获得的实际(采样)数据对算法进行改进。

如图1.1所示,DSP系统用DSP处理器或其他数字硬件、模数转换器(ADC)和数模转换器(DAC)代替放大器、调制器和滤波器等模拟装置。利用ADC采样待处理模拟信号,并以二进制形式将其编码为数字信号(见第3章)。DSP处理器基于以软件实现的一种特定信号处理算法(或计算说明)执行数字运算,以处理数字信号。可以利用DAC将输出数字信号转换回模拟形式。DSP处理器可以处理数字输入信号,然后将输出信号传递给其他数字系统,或者其他数字系统进行处理,而不需要使用转换器。在一些应用中,DSP系统在其内部生成数字信号。例如,可以使用一个DSP处理器来合成一对正弦波(拨号音),使电话振铃。



图1.1 一个典型DSP系统

DSP算法可以执行于很多种数字硬件上，可以用许多计算机语言来执行，如C语言和C++语言。DSP硬件包括可编程和不可编程逻辑、通用微处理器与微控制器，以及通用数字信号处理器。例如，有可能在一块超大规模集成(VLSI)电路片内集成非常先进的DSP系统，如视频编码和视频解码算法。但是，硬件解决方案集中于特定问题。这种解决方案比可编程解决方案更快速、更小巧，但其设计时间较长。此外，其开发成本更高，且难于修改或升级。

可以为各种任务对可编程处理器进行编程。它被用于因为过分复杂不能用非可编程电路实现的系统、需要较短开发时间和较低开发成本的产品，或者需要经常使用新算法和标准进行升级的系统。对于一个特定任务，处理器的速度要慢于专为该任务定制的超大规模集成电路芯片。

通用微处理器或微控制器设计用于处理最一般形式的处理，其采用冯·诺伊曼体系结构——指令和数据采用统一的存储器。与此相反，哈佛体系结构将存储器空间分为程序存储器和数据存储器，所以可以同时访问两者。多数DSP微处理器采用经过修改的哈佛体系结构，它给出了程序存储器与数据存储器之间的交叉路径。还对多数处理器进行了优化，以处理重复乘加运算，这些运算顺序访问存储于连续存储器位置的数据。在本书中，主要介绍采用DSP处理器实现DSP系统。

DSP系统正在逐渐代替现有模拟系统，并产生新的应用。DSP的主要优点在于其可编程能力。DSP系统使用户可以通过在系统存储器中编写新的代码来改变任务。它还允许使用算法来升级系统的参数，以适应动态变化的环境。因为在产品寿命内、不同环境中，存储器与处理器基本上与温度和老化无关，所以DSP系统的性能能够保持一致。此外，可以用不同语言对相同的DSP算法进行编码，并在不同的数字硬件上运行，而得到相同的结果。DSP系统的成本一直在持续降低，而性能一直在升高。随着每一代新处理器的出现，DSP的功耗都在降低，非常适于便携式装置。此外，一些信号处理运算是不能用模拟技术来实现的。

随着数字硬件技术的发展，对DSP算法和应用的新研究也在齐头并进。对DSP的此类研究主要集中于提高算法性能并降低系统复杂性，由此发展了一些新的DSP技术，其中包括用于未知或时变环境的自适应滤波(见第9章)和快速算法，如用于计算离散傅里叶变换的快速傅里叶变换(FFT)(见第8章)。

随着DSP硬件与算法功能的发展，处理需求也在发展，这样就导致了更高性能系统的发展，其具有用于新一代应用程序的更先进算法。在当今发展的DSP应用中，设计的灵活性和可升级性是更长生产周期的关键因素。许多行业标准处于早期开发阶段，这些标准中的一部分必须保持与其他标准的兼容性。一个非常好的例子是数字蜂窝电话，它已经从2G

升级到了2.5G、3G和4G标准。可编程DSP处理器特别适用于需要多操作模式和未来升级能力的设计。

1.2 数字信号处理系统与应用

DSP系统经常被内嵌于一个更大的系统中执行特殊的DSP操作,从而使整个系统可以处理通用任务。例如,用于数据传输的调制解调器中的DSP处理器是一个计算机的内嵌DSP系统。通常,这类DSP系统只运行一个应用程序,不能由终端用户进行编程。事实上,用户可能根本不知道有一个处理器(例如在硬盘驱动器中用于马达控制和磁盘位置控制的DSP处理器)被内嵌于计算机中。在本节,我们将简要介绍DSP系统设计的概念、TMS320系列和一些典型的实际应用。

1.2.1 数字信号处理系统

DSP的研究结果正在被越来越多地应用于制定完整的解决方案,这些解决方案将算法、软件和硬件集成于一个系统中。软件是控制一组计算机硬件资源实现一个算法的程序,而硬件由特定的计算组件组成。软件开发可能与处理器无关,也可能是设备专用的。因为在多数DSP系统中,软件开发的成本已经大于硬件开发的成本,所以与处理器无关的设计具有将软件用于不同处理器的优点,而且能够在将来移植到更为先进的处理器中。

在与处理器无关的设计中,诸如C或C++之类的高级语言为优选语言,它们可用于大多数DSP处理器。C程序的编写更为容易、更为快速,而且只需要利用新处理器的C编译器将源代码进行重新编译,就可以将C程序由一个处理器移植到另一个处理器。如果算法需要实现于几种不同处理器中,这一便携式特性将非常重要。处理器相关软件需要编写汇编程序,这些汇编程序反映了某个特定处理器的体系结构。手工编写的汇编代码运行速度快、所需存储器少,但其编写时间更长,而且对于每种不同的处理器,必须重新进行编写。因此,如果C语言在给定应用中所用处理器的运行速度足够快,那么它将是优选实施方法。

对于处理和存储器资源非常重要的应用中,可以采用折衷方式的解决方案——以汇编语言实现关键部分,用C语言进行其余部分的编码。这种C语言与汇编语言的混合式编程在简化编码与有效实现之间提供了一种良好的平衡。目前,C编译器的效率已经获得了显著提高,许多优化的汇编代码DSP库使用户可以简单而有效地开发混合式代码。在本书中,我们使用调用汇编优化子程序的C程序来进行实验。

大多数DSP系统必须实时产生其结果。在实时DSP应用中,系统的输入端接收到一个采样(或者是称作“帧”或“块”的一组采样)。系统必须在一段时间内完成与一输入信号的采样(或帧)相关的输出计算,这段时间不能超出采样(或帧)的持续时间。例如,在采样处理中,应在下一采样到来之前处理完每个采样。在诸如FFT等块处理中, N 个采样被分为一块,只有在获取这 N 个采样后才能对其进行处理。因此,与采样处理相比,实时块处理的延迟时间更长一些。

大多数DSP算法和应用都是在通用计算机上开发和检验的,对执行时间、存储器大小和系统成本没有限制。为实际应用准备算法需要权衡和综合关于算法、软件与硬件的知识。例如,人们可以选择采用硬件还是采用软件来执行算法,是选择速度还是灵活性,是改变特定软件的算法结构,还是改变实现该算法的硬件,是为该算法设计定制硬件,还是在

开发和维护上保持成本最低。

DSP系统设计涉及对应用问题的理解和对满足所有需要的系统设计的理解。要实现所设计的DSP系统,需要一个系统、硬件、软件和测试工程人员的团队。因为DSP系统是基于计算机的系统,所以可以首先用软件对设计进行仿真,然后进行修改,以大幅缩短产品开发时间。要分析和设计一个满足应用要求的有效DSP算法是一个复杂的任务。利用计算机辅助设计(CAD)工具进行DSP系统设计的结构化方法是必需的。在本书中,我们广泛应用The MathWorks公司[6]的带有Simulink、工具箱和模块的MATLAB 6.5进行设计和分析。

1.2.2 TMS320 系列

在1982年,德州仪器公司[7]在TMS320系列中引入了第一个DSP处理器——TMS32010。今天,它由定点和浮点处理器组成。16位定点处理器包括TMS320C2000(C24x和C28x)、C5000(C54x和C55x)和C6000(C62x和C64x)代。定点处理器将在第4章进行讨论。32位浮点处理器包括C3x、C4x和C67x代。浮点处理器将在第5章进行讨论。TMS320系列同一代中的处理器采用相同的核心体系结构,但所用的芯片存储器与外围设备配置不同。

在1982年,TMS32010体系结构与指令集的设计使其由时钟速率为5MHz的限定集成电路技术中获得了最大的计算能力和灵活性。今天,由于体系结构的改进与更快速的时钟速率,最快速的C64x已经将其执行速率(吞吐率)提高到1GHz。与时钟速率无关的体系结构改进包括:使用多个乘加计算(MAC)单元、执行密集流水线和并行运算、以多个片内部总线提供更多、更快的存储器、引入功能更强大的算法专用指令和硬件引擎,用于实现一些广泛使用的DSP算法。

在图1.1中已经给出了一个DSP系统,其需要ADC、DAC和DSP处理器。图1.2说明如何围绕TMS320处理器配置一个DSP系统。所需要的主要外部模块为存储器(程序和/或数据)与外围设备。DSP处理器通常提供一些片内缓存、程序只读存储器(ROM)、数据随机存取存储器(RAM)和外围设备。ADC、DAC等外围设备既可以利用一个专用地址连接至数据总线,也可以在片内的串行端口可用时连接至串行接口。

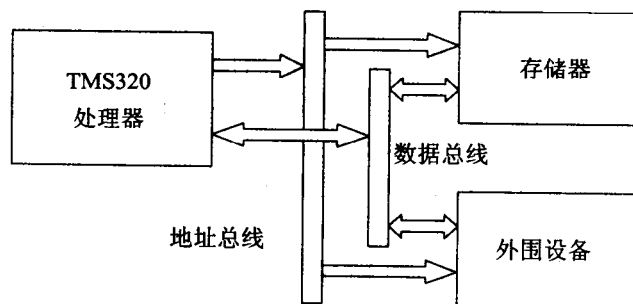


图1.2 TMS320处理器的外部接口

在DSP应用中,处理器与外部世界相互联系,以进行数据传输,如图1.1所示。有大量的商业输入/输出(I/O)设备可供使用,人们必须从硬件和软件的角度选择出能够满足所有应用要求的产品。在大多数DSP处理器中,利用从地址总线(较低位)和专用控制线解码得到的控制信号完成I/O设备的选择与启用。需要一个外部控制信号来区分存储器寻址和数字I/O寻址。软件的任务就是设置正确的外部设备地址并激活I/O控制线。

1.2.3 数字信号处理应用

DSP技术正在替代许多领域内的模拟信号处理方法。DSP现在已经成为商业现实，并且正在变为许多领域的标准技术，这些领域包括：通信、生物医学的信号分析与处理、数字控制、语音、音频和视频处理、数字仪表、雷达与声纳，等等。对于许多应用的理解都需要对其所应用领域的知识。在本书中，将在每一章介绍一些简单的应用。DSP应用的示例总结如下(改编自[8])：

音频

- 音频水印
- 编码与解码
- 音响效果发生器
- 环绕音效处理
- 三维音频合成器

汽车

- 有源噪声与振动控制
- 自适应行车控制
- 蜂窝电话
- 数字无线电
- 娱乐系统
- 导航与全球定位
- 语音命令

控制与工业

- 磁盘驱动控制
- 激光打印控制
- 电动机控制
- 数字控制
- 机器人控制
- 安全访问

图形/图像

- 动画/数字地图
- 计算机与机器人视觉
- 数码相机
- 图像压缩/传输
- 图像增强
- 模式识别
- 三维旋转

医学

- 诊断设备
- 数字助听器
- 胎儿监护
- 病人监护
- 超声波设备

通信

- 通信安全(加密编码)
- 检测
- 编码与解码

软件无线电

- 波形合成器

消费者

- 数字应答机
- 数字音频/视频播放器与记录器
- 教育工具与玩具
- 音乐合成器与调音器
- 寻呼机
- 个人数字助理(PDA)的插件
- 雷达探测器

通用

- 自适应滤波
- 卷积与相关
- 数字滤波
- FFT与快速余弦变换
- 希尔伯特变换
- 波形产生

仪器

- 数字滤波
- 函数产生
- 测量设备
- 模式匹配
- 锁相环
- 频谱分析
- 瞬态分析

军用

- 图像处理
- 导航制导
- 导航
- 雷达与声纳处理
- 安全通信

电信与声音/语音

- 自适应均衡器
- 扬声器电话
- 蜂窝电话
- 说话者验证
- 通道复用
- 语音增强
- 数据加密