

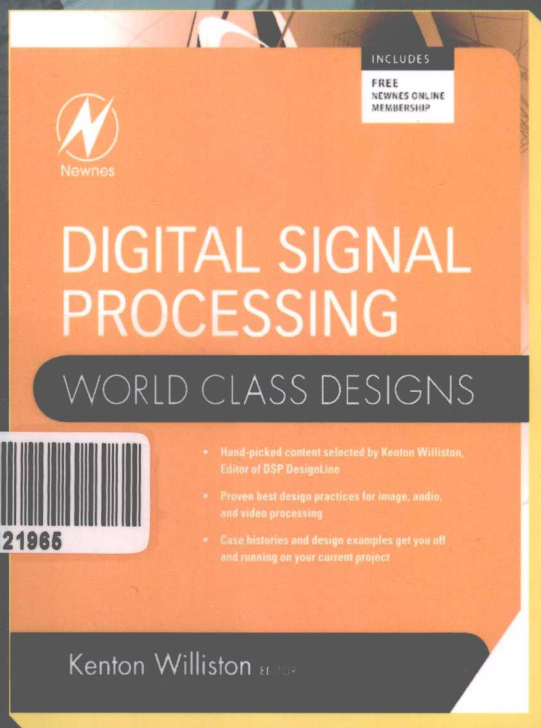
数字信号处理 权威指南

Digital Signal Processing World Class Designs

[美] Kenton Williston 编著
杨鸿武 裴东 译



YZLI0890121966



- Hand-picked content selected by Kenton Williston, Editor of DSP DesignLine
- Proven best design practices for image, audio, and video processing
- Case histories and design examples get you off and running on your current project

Kenton Williston EDITOR



人民邮电出版社
POSTS & TELECOM PRESS

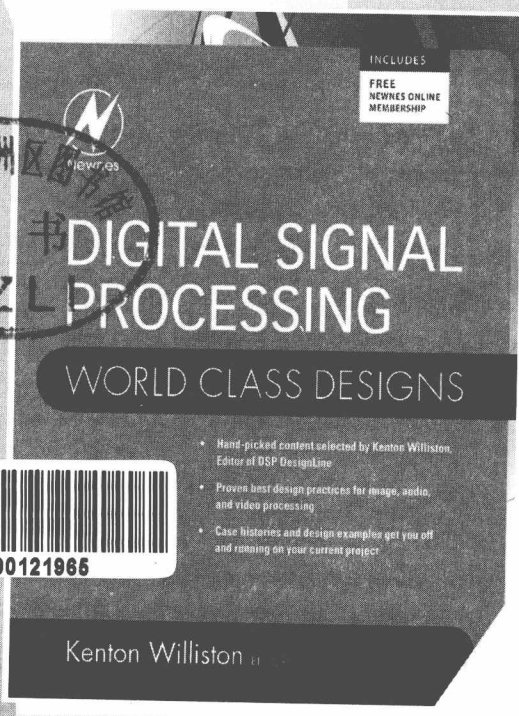
TURING

图灵电子与电气工程丛书

数字信号处理 权威指南

Digital Signal Processing
World Class Designs

[美] Kenton Williston 编著
杨鸿武 裴东 译



人民邮电出版社
北京

图书在版编目(CIP)数据

数字信号处理权威指南 / (美) 威利斯顿
(Williston, K.) 编著; 杨鸿武, 裴东译. -- 北京: 人
民邮电出版社, 2012. 1

(图灵电子与电气工程丛书)

书名原文: Digital Signal Processing: World
Class Designs

ISBN 978-7-115-26971-3

I. ①数… II. ①威… ②杨… ③裴… III. ①数字信
号处理—指南 IV. ①TN911.72-62

中国版本图书馆 CIP 数据核字(2011)第 243793 号

内 容 提 要

本书是目前销量最好的数字信号处理类图书的精华章节集粹, 汇聚了数字信号处理领域中读者最为关心的话题, 包括模数转换和数模转换、数字滤波、音频视频处理、解调、数字信号处理以及数字信号处理系统的测试和调试等。本书化繁为简, 直指问题本质, 处处闪现编著者的真知灼见。

本书适合软硬件研发工程师和研究人员、电子电气类本科生和研究生阅读参考。

图灵电子与电气工程丛书

数字信号处理权威指南

◆ 编著 [美] Kenton Williston

译 杨鸿武 裴 东

责任编辑 朱 巍

执行编辑 罗词亮

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京铭成印刷有限公司印刷

◆ 开本: 787×1092 1/16

印张: 21.5

字数: 508 千字

2012 年 1 月第 1 版

印数: 1-3 000 册

2012 年 1 月北京第 1 次印刷

著作权合同登记号 图字: 01-2010-3991 号

ISBN 978-7-115-26971-3

定价: 69.00 元

读者服务热线: (010)51095186 转 604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Digital Signal Processing: World Class Designs by Kenton Williston.

ISBN: 978-1-85617-623-1.

Copyright © 2009 by Elsevier. All rights reserved.

Authorized Simplified Chinese translation edition published by the Proprietor.

ISBN: 978-981-272-451-9.

Copyright © 2011 by Elsevier (Singapore) Pte Ltd. All rights reserved.

Elsevier (Singapore) Pte Ltd.

3 Killiney Road

#08-01 Winsland House I

Singapore 239519

Tel: (65) 6349-0200

Fax: (65) 6733-1817

First Published 2012

2012 年初版

Printed in China by POSTS & TELECOM PRESS under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in China only, excluding Hong Kong SAR and Taiwan. Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书简体中文版由 Elsevier (Singapore) Pte Ltd. 授权人民邮电出版社在中华人民共和国境内（不包括香港特别行政区和台湾地区）出版与销售。未经许可之出口，视为违反著作权法，将受法律之制裁。

撰稿人简介

James Bryant (第1章) 是 *Mixed Signal and DSP Design Techniques* 一书的联合撰稿人。

Rick Gentile (第9章) 于2000年加入 ADI (Analog Devices, Inc., 亚德诺) 公司, 成为一名 DSP 高级应用工程师。目前, 他是该公司应用工程组的主管, 负责 Blackfin、SHARC 和 TigerSHARC 处理器相关的应用工程工作。在加入 ADI 公司之前, Rick 是麻省理工学院林肯实验室的一名技术人员, 在那里他设计了多个广泛应用于雷达传感器的信号处理器。他于1987年获得马萨诸塞大学阿默斯特分校电气与计算机工程学士学位, 并于1994年获得(美国)东北大学电气与计算机工程硕士学位。

Keith Jack (第5章) 是 Sigma Designs 公司产品营销总监。Sigma Designs 公司是提供高性能片上系统 (SoC) 解决方案的核心供应商, 这些芯片主要用于网络电视 (IPTV)、蓝光光碟 (Blu-ray) 和高清晰度电视 (HDTV) 上。此前, 他是 Innovision 公司产品营销总监, 专注于数字电视解决方案。Jack 还曾担任过哈里斯半导体公司和 Brooktree 公司战略市场经理。他为消费市场构建并推出了超过35个多媒体 SoC, 还是 *Video Demystified* 一书的作者。

David Katz (第9章) 有超过15年的电路与系统设计经验, 目前是 ADI 公司 Blackfin 处理器应用经理, 主要负责详细说明新的处理器。他在国内和国际期刊上已经发表过百余篇关于嵌入式处理的文章, 还发表过许多电子领域的会议论文。此外, 他还是 *Embedded Media Processing* (Newnes, 2005) 一书的合著者。在此之前, 他是摩托罗拉公司电缆调制解调器和自动化系统方面的高级设计工程师。他拥有康奈尔大学电子工程学士学位和硕士学位。

Nasser Kehtarnavaz (第3章) 是 *Digital Signal Processing System Design* 和 *Real-time Digital Signal Processing* 的作者。

Walt Kester (第1章) 是 ADI 公司的一名应用工程师。在 ADI 公司的35年多里, 他在高速数模转换、模数转换、采样保持、运算放大器和模拟多路复用器方面做过很多设计、开发和应用支持工作。除了发表论文和写文章外, 他还起草和编辑了11种重要的应用方面的书籍, 这些书是 ADI 公司全球技术系列研讨会的基础, 研讨会的主题包括运算放大器、数据转换、电源管理、传感器信号调理、混合信号以及模拟设计实用技术。他还编有厚达900多页的数据转换大全 *Data Conversion Handbook* (Elsevier, 2005)。Walt 拥有北卡罗来纳州立大学电子工程学士学位和杜克大学电子工程硕士学位。

Dake Liu (第7章) 是瑞典林雪平大学的教授, 也是该大学电气工程学院计算机工程系主任。他专注于 ASIP (应用相关指令集处理器) 和基于 VLSI (超大规模集成电路) 的片上多核集成方面的研究。他的研究目标是探索不同处理器体系结构与处理器间体系结构和 DSP 固件的并行编程。在处理器水平上, 他的研究重点是基于异构指令集和异构架构的集成 DSP-MCU 解决方案。在 ILP (指令级并行) 水平的研究上, 他目前的主要研究工作是架构和基于模板的编

程工具，特别是冲突的空闲并行内存访问。在 CMP（芯上多处理器）水平的研究上，他目前的研究重点是实时计算中灵活的低延迟和低硅芯片成本系统。他的研究应用于嵌入式 DSP 计算、通信和多媒体信号处理。他在国际期刊和会议上发表了约 100 篇论文。

Robert Meddins（第 2 章）是 *Introduction to Digital Signal Processing* 一书的作者。

Robert Oshana（第 10 章）有 25 年多的嵌入式实时系统和 DSP 系统开发经验。他在相关领域写过很多文章和专著。他还是嵌入式系统咨询委员会的会员，并在全球性的嵌入式系统会议上作过多次演讲。Robert 是 IEEE 的高级会员，有 PE（高级工程师）执照，还是南方卫理公会大学的兼职教授。

Ian Poole（第 6 章）是一位著名的电子工程顾问，在通信和手机市场有着丰富的工作经验。他写过很多无线电和电子产品方面的书，并在英国和世界各地的许多杂志上发表过文章。他也是首届 ARRL 技术写作奖——Bill Orr 奖得主。

Khalid Sayood（第 4 章）分别于 1977 年和 1979 年获得罗切斯特大学电子工程学学士学位和硕士学位，并于 1982 年获得得克萨斯 A&M 大学电子工程学博士学位。1982 年他进入内布拉斯加州大学任教，并成为电子工程专业的汉森教授。他的研究领域主要包括数据压缩、信源信道联合编码和生物信息学。

Dan Sheingold（第 1 章）是 *Mixed Signal and DSP Design Techniques* 一书的联合撰稿人。

Li Tan（第 8 章）是德锐大学资深教授及电子工程技术专业课程协调员。

前言

人们常说的 DSP 是什么？它有哪些不同寻常之处？DSP 工程师是做什么的？要成为 DSP 工程师，该如何入手呢？问得好！如果你想知道答案，就和我一起开始探索之旅，揭示这所有的一切吧。

DSP 的定义

首先，我们来弄清楚什么是 DSP。缩写 DSP 有两种含义：数字信号处理（Digital Signal Processing）和数字信号处理器（Digital Signal Processor）。先来看看数字信号处理。“数字”的含义很明显，它的意思是我们工作在“1”和“0”的世界中，而不是在模拟世界中。“信号”的含义有点麻烦。维基百科将其定义为“任何时间或空间变量”。语音是时间变量，声音的音高和音量随着时间的变化而变化。照片是空间变量，图像的颜色和亮度在照片不同的区域是不同的。现在就剩下定义“处理”了。这是一个很广泛的概念，通常涉及利用数学算法进行分析和处理。例如，可以分析一段录音来确定它的音高，也可以通过调整颜色来处理照片。

DSP 的应用主要分为 4 个方面：

- ☐ 通信；
- ☐ 音频、视频和成像（有时称为媒体处理）；
- ☐ 运动控制；
- ☐ 军事和航天。

在这些领域中，通信和视频受到了极大关注。这两个领域的发展非常迅速，两者都施加了很高的计算负荷。此外，这两个领域都包括严格限制功耗的系统。

了解了以上知识，定义“数字信号处理器”就很容易了。数字信号处理器是一种专用于数字信号处理的微处理器。例如，许多信号处理算法涉及先乘后加，这个运算通常称为乘法累加或 MAC。因为 MAC 运算在信号处理中很常见，所以所有数字信号处理器都有专门的 MAC 硬件。此外，许多 DSP 应用限制功耗。因此，很多 DSP 具有先进的省电功能，比如在工作状态下改变速度和电压（即频率和电压动态缩放功能）。

值得注意的是，数字信号处理不需要数字信号处理器。很多信号处理系统使用通用处理器（例如利用 ARM 和 MIPS）或者定制硬件（利用 ASIC 和 FPGA）。许多系统使用混合硬件。例如，许多系统包含 DSP 和通用处理器（GPP）。

DSP 工程师的角色

基础知识到此为止。下面要考虑的是，DSP 工程师到底是做什么的呢？我的朋友 Shiv Balakrishnan 最近就这一主题写了一篇文章，我同意他的结论，DSP 工程师通常分为 3 类。

(1) 系统设计师：建立算法和（在某些情况下）整体系统。

(2) 硬件设计师：用硬件实现 (1)。

(3) 程序员：用软件实现 (1)，要么利用 (2) 建立的硬件，要么利用现成的硬件。

本书主要面向程序员。这是 DSP 工程师最常见的角色，而且这个角色最容易胜任，不需要了解研究生层次的概念。尽管如此，深入研究每一种角色并探索其技能组合还是很有意义的。

系统设计师

系统设计师着眼于大局。这类工程师可以设计系统的整体功能，包括所有随之而来的算法，也可以侧重于具体的子系统和算法。后一种情况对于手机等产品来说是很常见的，因为这些产品系统对于一个工程师来说过于复杂。

系统设计师被看作是领域专家，因为人们需要一位了解系统需求并知道如何满足这些需求的专家。这包括在模拟世界方面的专业知识，因为大多数 DSP 系统具有模拟输入与输出。例如，无线通信系统设计师必须知道信号在空气中传播时是如何衰减的。系统设计师也需要最优算法知识。为了满足这些要求，系统设计师往往需要具有硕士学位甚至是博士学位。

系统设计师利用诸如 Simulink 和 LabVIEW 的图形工具以及 MATLAB 和 C 语言这样基于文本的工具建立系统函数模型。系统设计师通常不深入硬件及软件设计的细节，但他们必须了解这些学科的基础知识。如果没有人可以实现，即使是最精心的设计也是毫无价值的！系统设计师还必须确保硬件设计师和程序员能够充分理解功能模型。这意味着系统设计师必须提供针对系统模型来测试硬件和软件的方法，而且还有其他要求。为了达到这些目标，系统设计人员越来越多地致力于电子系统级 (ESL) 工具。ESL 工具执行许多函数，包括自动生成软硬件参数以及测试向量。

硬件设计师

系统设计好后，就轮到硬件设计师上场了。硬件设计师的作用很广泛。与系统设计师一样，硬件设计师可以针对整个系统，也可以侧重于具体的子系统。硬件设计师可以创建定制硬件，或者利用现成硬件构建系统。对于定制硬件，硬件设计师曾求助于 ASIC，但是除非针对最高产量的产品，ASIC 的设计成本还是过高，因此，硬件设计师通常采用 FPGA。结构化 ASIC 也是一种选择，尤其是在中等产量的应用中。

在这些情况下，硬件设计师意识到硬件是一组块 (block)。传统上，硬件设计师用手工编写的 RTL (VHDL 或 Verilog) 实现每个块，验证它，优化它。虽然用手工编写的 (hand-coded) RTL 仍在使用，但硬件工程师越来越依赖于 ESL 工具来生成硬件。FFT 和 FIR 滤波器等关键 DSP 算法的实现更是如此。对于这些算法的硬件生成，ESL 工具已经非常精通。除了定制块，大多数应用程序还包括一个或多个可编程处理器。该处理器可以利用 ASIC 或 FPGA 实现，或者，硬件设计师也可以使用现成的处理器。

对于利用现成硬件的系统，硬件设计师的工作就简单多了。然而，硬件设计师仍然要精心

选择来满足系统的需求。处理器必须有足够高的性能来完成工作量，总线要有足够的带宽来处理数据，等等。在很多情况下，不大可能找到现成的硬件来完全满足系统的需求，所以硬件设计师必须创建少量的定制硬件，例如，许多系统利用 FPGA 来实现用现成的处理器无法实现的 I/O。

这一切的结果是硬件设计师需要掌握多种技能。虽然 ESL 工具减轻了工作负担，但是硬件设计师必须具备良好的 RTL 编码技巧。显然，硬件设计师必须了解算法实现。他们还必须了解软件的需求，以作出明智的设计决策。尽管学士水平的教育可能就可以满足这些需求，但硕士学位会更有帮助。

程序员

下一个：程序员。程序员写代码来实现系统处理器的其余功能。像任何其他领域的程序员一样，DSP 程序员通常也用 C/C++ 写代码。但是，DSP 代码有两点不同。首先，DSP 代码通常始于 MATLAB 或 Simulink 模型。因此，MATLAB 到 C 语言的转换工具引起了很多人的兴趣。其次，DSP 代码需要对性能、尺寸和功率进行大量优化。在许多情况下，程序员需要用汇编语言来优化代码的关键部分。为了实现最大优化，DSP 工程师必须非常熟悉硬件细节。

如前所述，许多 DSP 系统包含多个处理器。例如，许多系统包括 GPP 和 DSP。近年来，多核处理器也已普遍。（多核处理器在单芯片上结合两个或更多的处理器核。）因此，多处理器编程成为很多 DSP 程序员要掌握的关键技能。

就在不久前，DSP 程序员还在自己动手写大部分软件。而今天，对于系统的大部分，DSP 程序员经常使用现成的通用软件。对于语音和媒体处理编解码器尤其如此。这些编解码器已变得高度标准化和商品化。

显然，程序员必须了解他们所实现的算法。使用现成 DSP 软件的程序员只需要知道基本算法的基础知识。无论在哪种情况下，学士的文化程度通常是足够的，而硕士教育会更有帮助。

DSP 项目流程

到目前为止，我们相对独立地描述了 DSP 的 3 个角色，但在实践中这 3 个角色往往是重叠的。例如，一个工程师可以在一个较小的项目中充当所有 3 个角色。而且要特别注意的是，我们还暗示了从一个角色切换到另一角色的线性项目。工程实践中很少采用这种方式。相反，大多数项目都遵循一个在不同角色间进行大量反馈的重复过程。此外，系统设计、硬件设计和程序设计往往会同步进行。

DSP 展望

过去 DSP 只应用专业领域，应用范围很窄。5 年前，DSP 几乎就是电信的代名词。如果有人告诉你，他们是 DSP 工程师，你就可以清楚地了解他们是做什么的。你可以肯定，他们具有良好的数学能力，可以准确地解释 FIR 滤波器是如何工作的。今天，DSP 无处不在，往往被

隐藏在“媒体处理”这样的名词之下。许多从事 DSP 系统开发的工程师都只对基本算法有个大概了解。由于这些变化，很难明确界定究竟什么是 DSP 工程师，或者他们是做什么的——随着 DSP “扩散”到越来越多的应用中，这种混乱可能会不断加剧。不过，有一点是肯定的：DSP 技术在多年内仍将是一个受欢迎的重要技能。

目 录

第 1 章 模数转换器、数模转换器和

采样定理 1

1.1 编码和量化 1

1.1.1 单极性码 3

1.1.2 格雷码 5

1.1.3 双极性码 8

1.1.4 补码 11

1.1.5 DAC 和 ADC 静态传递函数和 直流误差 11

1.2 采样定理 19

1.2.1 取样-保持放大器 (SHA) 功能 19

1.2.2 奈奎斯特准则 22

1.2.3 基带抗混叠滤波器 24

1.2.4 欠采样 (谐波采样、带通采样、 中频 (IF) 采样、IF 信号直接 转数字信号) 26

1.2.5 用于降采样的抗混叠滤波器 27

1.3 数据转换器的交流误差 29

1.3.1 N 位理想转换器的理论量化 噪声 29

1.3.2 实用 ADC 的噪声 34

1.3.3 数据转换器的动态性能 36

1.3.4 DAC 的动态性能 65

1.4 通用数据转换器指标 73

1.4.1 数据转换器的整体分析 73

1.4.2 逻辑接口 74

1.4.3 数据转换器逻辑电路: 时序 等问题 75

1.5 定义指标 75

参考文献 89

第 2 章 数字滤波器 92

2.1 本章预览 92

2.2 模拟信号处理 92

2.3 替代方案 93

2.4 完整的 DSP 系统 94

2.4.1 抗混叠滤波器 95

2.4.2 采样-保持电路 95

2.4.3 ADC 96

2.4.4 处理器 96

2.4.5 DAC 97

2.4.6 重构滤波器 97

2.5 回顾 97

2.6 数字数据处理 97

2.7 滑动平均滤波器 98

2.8 处理系统表示法 99

2.9 反馈 (递归) 滤波器 100

2.10 本章总结 102

参考文献 102

第 3 章 信号的频域处理 103

3.1 离散傅里叶变换 (DFT) 和快速 傅里叶变换 (FFT) 103

3.2 短时傅里叶变换 (STFT) 104

3.3 离散小波变换 (DWT) 106

3.4 信号处理工具箱 107

Lab 3: FFT、STFT 和 DWT 107

L3.1 FFT 与 STFT 107

L3.2 DWT 112

参考文献 116

第 4 章 音频编码 117

4.1 概述 117

4.2 简介 117

4.2.1 频谱掩蔽现象 119

4.2.2 时域掩蔽 119

4.2.3 心理声学模型 120

4.3 MPEG 音频编码 120

4.3.1 layer I 编码	121	5.6.3 超黑色和亮度键控	161
4.3.2 layer II 编码	122	5.7 视频比例缩放	161
4.3.3 layer III 编码——MP3	123	5.7.1 像素增减	162
4.4 MPEG AAC	127	5.7.2 线性插值	162
4.4.1 MPEG-2AAC	128	5.7.3 抗混叠重采样	162
4.4.2 MPEG-4 AAC	130	5.7.4 比例缩放显示实例	164
4.5 Dolby AC-3 (Dolby Digital)	131	5.8 扫描速率转换	168
4.6 其他标准	133	5.8.1 帧增减或段增减	168
4.7 总结	133	5.8.2 瞬时插值	168
参考文献	133	5.8.3 2:2 下拉处理	172
第5章 视频处理	135	5.8.4 3:2 下拉处理	173
5.1 近似法	136	5.8.5 3:3 下拉处理	173
5.1.1 截断	136	5.8.6 24:1 下拉处理	174
5.1.2 四舍五入法	136	5.9 非交错式到交错式视频的转换	174
5.1.3 误差反馈近似	136	5.9.1 扫描线抽取	174
5.1.4 动态近似	136	5.9.2 垂直滤波	175
5.2 SDTV-HDTV YCbCr 转换	137	5.10 交错式视频到非交错式视频的 转换	175
5.2.1 SDTV 数据在 HDTV 上显示 ...	137	5.10.1 视频模式: 场内处理	176
5.2.2 HDTV 数据在 SDTV 上显示 ...	137	5.10.2 视频模式: 场间处理 方法	177
5.3 4:4:4 到 4:2:2 的 YCbCr 转换 ...	138	5.10.3 运动补偿去交错法	179
5.3.1 Y 滤波	138	5.10.4 电影模式 (反电视电影 法)	179
5.3.2 CbCr 滤波	139	5.10.5 频响	179
5.4 显示增强	140	5.11 基于 DCT 的压缩方法	180
5.4.1 亮度、对比度、饱和度和 色度	140	5.11.1 DCT	181
5.4.2 彩色瞬态改善	142	5.11.2 量化	182
5.4.3 亮度瞬态改善	142	5.11.3 zig-zag 扫描	182
5.4.4 清晰度	142	5.11.4 行程编码	183
5.4.5 蓝电平延伸电路	143	5.11.5 可变长度编码	183
5.4.6 绿色增强	144	5.12 固定像素点显示	183
5.4.7 动态对比度	144	5.12.1 扩展色彩重现	183
5.4.8 色彩校正	144	5.12.2 细节校正	183
5.4.9 色温校正	145	5.12.3 非均匀量化	184
5.5 视频混合和图形叠加	145	5.12.4 缩放与去交错	184
5.6 亮度和色度键控	150	5.13 应用实例	184
5.6.1 亮度键控	150		
5.6.2 色度键控	151		

参考文献	186	7.4.2 在 GP DSP 处理器上实现 DSP	224
第 6 章 调制	187	7.4.3 在 ASIP 上实现 DSP	224
6.1 无线电载波	187	7.4.4 用 ASIC 实现 DSP	225
6.2 调幅	188	7.4.5 实现的权衡和决策	226
6.3 调制指数	191	7.5 回顾处理器和系统	227
6.4 频率调制	191	7.5.1 DSP 处理器体系结构	227
6.5 调制指数和偏移系数	192	7.5.2 DSP 固件	228
6.6 边带	193	7.5.3 嵌入式系统综述	230
6.7 带宽	194	7.5.4 嵌入式系统中的 DSP	231
6.8 提高信噪比	195	7.5.5 嵌入式计算的基本原理	231
6.9 频移键控	195	7.6 设计流程	232
6.10 相位调制	196	7.6.1 通常的硬件设计流程	232
6.11 相移键控	196	7.6.2 ASIP 硬件设计流程	233
6.12 最小相移键控	198	7.6.3 ASIP 设计自动化	235
6.13 正交振幅调制	200	7.7 总结	238
6.14 展频技术	201	参考文献	238
6.15 跳频技术	201	第 8 章 DSP 处理器和定点运算	240
6.16 直接序列展频	201	8.1 DSP 处理器体系结构	240
6.17 正交频分多路复用	203	8.2 DSP 处理器硬件单元	243
6.18 带宽和数据容量	206	8.2.1 乘加器	243
6.19 总结	207	8.2.2 移位器	243
第 7 章 DSP 硬件选项	208	8.2.3 地址生成器	243
7.1 适用于硬件设计师的 DSP 理论	208	8.3 DSP 处理器及制造商	244
7.1.1 回顾 DSP 理论和基本原理	208	8.4 定点格式和浮点格式	245
7.1.2 ADC 和有限长度建模	209	8.4.1 定点格式	246
7.1.3 数字滤波器	211	8.4.2 浮点格式	252
7.1.4 变换	213	8.4.3 IEEE 浮点格式	255
7.1.5 自适应滤波器和信号增强	215	8.4.4 定点 DSP 处理器	258
7.1.6 随机过程和自相关	215	8.4.5 浮点处理器	259
7.2 理论、应用和实现	216	8.5 在定点系统中实现有限脉冲响应和 无限脉冲响应滤波器	261
7.3 DSP 应用	218	8.6 数字信号处理程序设计实例	265
7.3.1 实时的概念	218	8.6.1 TMS320C67xDSK 综述	265
7.3.2 通信系统	218	8.6.2 实时处理的概念	268
7.3.3 多媒体信号处理系统	219	8.6.3 线性缓冲	269
7.3.4 应用综述	223	8.6.4 样本 C 程序	272
7.4 DSP 实现	223		
7.4.1 用 GPP 实现 DSP	223		

8.6.5 定点实现实例	273	9.6.1 同步载入/存储	292
8.7 总结	275	9.6.2 排序	293
参考文献	275	9.6.3 原子操作	294
第9章 代码优化与资源分配	277	9.7 存储器体系结构——对管理的 需要	295
9.1 引言	277	9.7.1 存储访问的取舍	295
9.2 事件的产生和处理	278	9.7.2 指令存储器管理——高速缓存 还是 DMA	295
9.3 程序设计方法学	280	9.7.3 数据存储器管理	297
9.4 高效编程的体系结构特点	281	9.7.4 系统向导——如何在 DMA 和 高速缓存之间选择	299
9.4.1 每个周期的多重操作	281	9.7.5 内存管理单元 (MMU)	300
9.4.2 硬件循环结构	281	9.8 数据移动的物理学	303
9.4.3 专门的寻址模式	282	参考文献	308
9.4.4 互锁指令流水线	285	第10章 DSP 系统的测试与调试	309
9.5 出于高效编程的编译器考虑	287	10.1 多核片上系统	309
9.5.1 选择数据类型	288	10.2 SoC 的软件架构	315
9.5.2 数组与指针	288	10.3 SoC 系统引导顺序	318
9.5.3 除法	289	10.4 SoC 的支持工具	318
9.5.4 循环	289	10.5 SoC 的视频处理范例	318
9.5.5 数据缓冲器	290	参考文献	330
9.5.6 内联函数和内嵌	291		
9.5.7 易失性数据	292		
9.6 系统与核同步	292		

第 1 章 模数转换器、数模转换器 和采样定理

Walt Kester Dan Sheingold James Bryant

专门拿出一章来讲模拟知识？这不是一本关于 DSP 的书吗？还要作为本书第 1 章？放松！是的，进行 DSP 并不需要你对模拟知识有很深的了解，然而，了解一些模拟信号的知识还是大有裨益的。几乎所有的 DSP 系统都支持模拟信号的输入，处理这些输入的模拟数据前，我们需要对它们进行模数转换，这种处理过程往往是有缺陷的，转换成的数字信号不能完全精确地重构出原来的模拟信号。模数转换器（ADC）在转换的时候会带来噪声和失真，而更严重的是模数转换可能带来混叠现象，这种混叠现象使得在模拟域完全不同的两个信号在数字域无法区分。

在输出端也存在同样问题。大多数 DSP 系统支持模拟信号的输出，因此我们需要把处理完的数据再进行数模转换。数模转换器（DAC）也不能完全重现出原来的模拟信号，往往会产生噪声和失真。

以上问题听起来很麻烦，其实不用担心，只要你了解了 ADC 和 DAC 的缺陷，那么这些缺陷就很好解决。这一章将会教你怎样处理这些问题。在本章中，作者讲解了模数转换和数模转换的原理，其中包括最重要的奈奎斯特采样定理。本章还讲解了模数转换和数模转换的基本操作，并且解释了理想转换器产生误差的过程，然后进一步阐述了在实际中转换器的缺陷。

本章的知识对于大多数 DSP 工程师已经足够了，如果想了解更多，可以参考下面的网站。

ADC: <http://www.dspdesignline.com/howto/202200877>

DAC: <http://www.dspdesignline.com/howto/205601725>

——Kenton Williston

1.1 编码和量化

ADC 是将现实生活中普遍存在的模拟量转换成数字量，以便用于信息处理、计算、数据转换和控制系统。DAC 则用来将转换后或存储的数据，或者是经过处理的数字量转换成模拟量，以便进行控制、信息显示，或者进一步在模拟域进行处理。图 1-1 给出了转换器输出和输入之间的关系。

不管源自什么，模拟输入变量最常被传感器转换为电压或电流。这些电量可能以 3 种形式出现：（1）对时域中的现象或快或慢的直流连续直接测量；（2）解调的交流波形（使用多种解调技术）；（3）在某些组合中，相关变量的空间配制用来表示轴角。第一种形式的例子有热

电偶的输出、直流参考上的电位器，以及模拟运算电路；第二种形式的例子有斩波光学测量、直流应变计或电桥输出，以及噪声中的数字信号；第三种形式的例子有同步器和分解器。

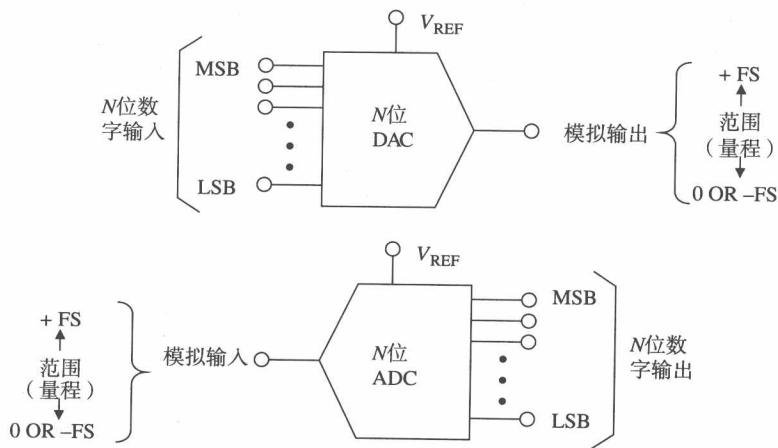


图 1-1 DAC 和 ADC 的输入输出

本章要处理的模拟变量都是电压或电流量，这些模拟量有宽带和窄带之分。它们可能是直接进行度量得到的结果，或者是经过预处理的模拟量，例如经过线性化、组合、解调、滤波、采样保持等处理。

作为处理的一部分，我们对电压和电流都进行了归一化，以满足规定的 ADC 输入范围。经过数模转换的模拟域的电压和电流是直流的且进行了归一化，模数转换后的这些模拟量还要进行后处理，例如缩放、滤波和放大等。

数字形式的信息通常用以地为参考电势的固定电压值表示，这些信息或者是逻辑门的输出或者是它们的输入。数字化后的信息一般是二进制，也就是说每一“位”或者每个信息单元都是两个状态中的一种，这些状态用 0 表示“关”、“假”，用 1 表示“开”、“真”，当然也可以用两种不同的电流值来表示这两种逻辑状态，然而，我们更多的是用电压值来表示。我们也不必以大地为参考电势，射极耦合逻辑电路、正射极耦合逻辑电路、低电平差分信号逻辑电路的参考电势就不是大地。

字 (word) 是数字域值的表示形式，这些值可能在一条总线上或者作为一组门电路输入输出值以并行的形式出现，也可能串行出现，也可能作为并行字节序列 (即字节串行) 或半字节出现。例如，一个 16 位的字可能占据一条 16 位总线的 16 位，或者它被分成一条 8 位总线的两个顺序字节形式，或者分成占据一条 4 位总线的 4 个 4 位半字节形式。

虽然逻辑电路有几种形式，但是应用最广泛的还是 TTL (晶体管-晶体管逻辑) 逻辑电路，也就是正值 (即 1) 对应最小输出电平 2.4 V (输入端 1 对应大于 2.0 V 的电平)，假值 (即 0) 对应最大输出电平 0.4 V (输入端 0 对应小于 0.8 V 的电平)。虽然现在 CMOS 电路比 TTL 电路应用更广泛，但是需要强调的一点是，CMOS 逻辑电平值一般与更老版本的 TTL 逻辑值兼容。

与每一部分模拟水平值相对应的并行或串行形式的数字水平值（即数字或编码）都是独一无二的，也就是说每个代码与模拟域的一个既定范围相对应。

一个常见的数字代码具有下列形式：

$$a_7 a_6 a_5 a_4 a_3 a_2 a_1 = 10111001$$

上面的代码由 8 位组成，最左边的“1”称为最高有效位（MSB）或称为第 1 位，最右边的 1 称为最低有效位（LSB）或称为第 N 位（这里为第 8 位）。这种码可以代表一个数字、一个字符，或者一个模拟变量，在代码和转换关系确定之前其具体意义是不可知的。注意不要混淆了与 $a_7, a_6, a_5, a_4, a_3, a_2, a_1$ 相对应的位（即第 1 位，第 2 位，等等）。每个脚标作为 2 的幂，得到的值作为序列中每一位的权。

除了十进制编码外，最常见的编码形式就是自然二进制编码（基-2 编码）了。二进制编码主要用于表示整数，也就是说一个自然二进制整数代码包含 N 位，它的 LSB 的权为 2^0 （即 1），下一位的权为 2^1 （即 2），以此类推，MSB 权为 2^{N-1} （即 $2^N/2$ ）。二进制数值通过所有非零位的加权和得到，加权后的各位相加得到 $0 \sim 2^N - 1$ 的任意值，如果有后补零位的话，每个后补零位起到加倍数值的作用。

在转换器技术中，满量程（FS）与分辨位数 N 无关。一种更有用的编码形式是分数二进制编码，这种编码通常被归一化到 FS。如果将代码中的每个整数都除以 2^N ，那么这样的整数二进制码就可认为是分数二进制码。例如，MSB 具有权 $\frac{1}{2}$ （ $2^{(N-1)}/2^N = 2^{-1}$ ），下一位的权为 $\frac{1}{4}$ （即 2^{-2} ），以此类推，LSB 的权为 $1/2^N$ （即 2^{-N} ），将加权后的各位相加，得到包含任意 2^N 值的数，在整个数字范围内从 0 到 $(1 - 2^{-N})$ 倍 FS，后补零位不影响 FS，它们只起到细化结构的作用。图 1-2 给出了十进制数和二进制数之间的关系，并分别举了例子。

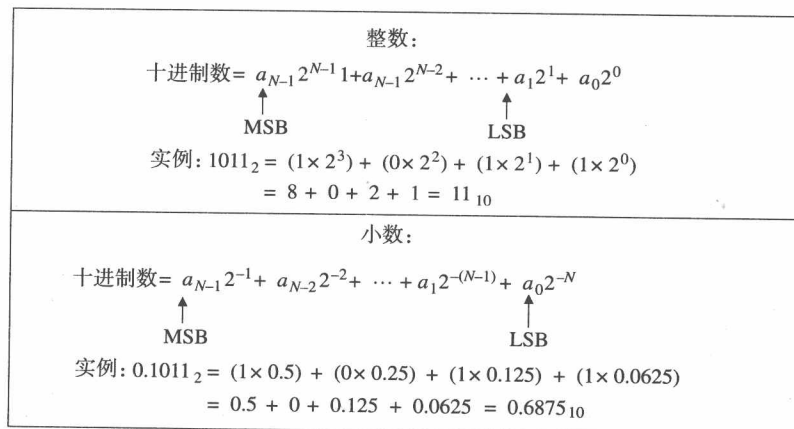


图 1-2 二进制数和十进制数的转换关系

1.1.1 单极性码

在数据转换系统中，编码方法必须与 ADC 的模拟数据输入范围或 DAC 的模拟数据输出范