



普通高等教育“十一五”国家级规划教材

DSP 控制器原理及应用

(第二版)

宁改娣 曾翔君 骆一萍 编著



科学出版社

www.sciencep.com

DSP 控制器原理及应用

(第二版)

宁改娣 曾翔君 骆一萍 编著

科学出版社

北京

内 容 简 介

DSP 是一种具有数字信号处理特长的高性能微处理器。本书总结了微处理器的结构框架、学习目的和方法。在此基础上,以 TMS320C28×为模型介绍其硬件结构、指令系统、存储器组织、中断系统及片内外设的基本原理,强化微处理器学习方法,训练使用微处理器的基本功。本书还加强了 DSP 最小系统介绍,使读者很容易建立自己的硬件平台;增加了 C 语言编程工程模板介绍,使读者可以很方便地进行软件开发;并且介绍了 DSP 的开发环境、C 语言编程和开发流程。在最后一章提供了不同层次的实验内容。书中所提供实例的软硬件都经过了试验验证。本书实践性较强,部分内容以案例编写,可更好地激发学生的学习兴趣。

本书可作为电子、电气、信息类、机电、探测以及生物医学工程等学科的研究生和高年级本科生 DSP 课程教材,包括实验在内约有 40~54 学时;也可供从事 DSP 技术研究和开发的科技人员参考。

图书在版编目 (CIP) 数据

DSP 控制器原理及应用/宁改娣,曾翔君,骆一萍编著. —2 版. —北京:科学出版社,2009

普通高等教育“十一五”国家级规划教材

ISBN 978-7-03-024106-1

I. D… II. ①宁… ②曾… ③骆… III. 数字信号-信号处理-高等学校-教材 IV. TN911.72

中国版本图书馆 CIP 数据核字 (2009) 第 023190 号

责任编辑:贾瑞娜 / 责任校对:桂伟利
责任印制:张克忠 / 封面设计:耕者设计工作室

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

源海印刷有限责任公司印刷

科学出版社发行 各地新华书店经销

*

2002 年 11 月第 一 版 开本:787×1092 1/16

2009 年 3 月第 二 版 印张:26 3/4

2009 年 3 月第五次印刷 字数:578 000

印数:1—4 000

定价:44.00 元

(如有印装质量问题,我社负责调换〈环伟〉)

第二版前言

21 世纪是数字化的时代，DSP 已成为这场数字化革命的重要元素，广泛应用于社会的各个领域。

在知识爆炸、信息爆炸、科技发展日新月异的时代，DSP 的更新换代也非常迅速。《DSP 控制器原理及应用》出版后的几年时间，又有许多的 DSP 不断面市，为我们提供了更多的选择。

很多学校使用《DSP 控制器原理及应用》作为教材，本书也是许多工程师的参考书。使用本书的教师、学生和工程师们提出了一些很好的意见和建议，我们在使用中也体会到改版的迫切性。

第二版做了很大篇幅的修改，主要反映在以下几个方面：

- 增加了处理器学习方法介绍。DSP 的迅速更新换代，给 DSP 处理器的教学和教材编写带来不少问题，使我们对微处理器教学和教材编写有了新的思考：微处理器课程必须通过某一处理器芯片的学习，强调和总结微处理器的学习方法。因此，在第二版中，作者根据 20 多年对各种微处理器课程的教学及使用经验，总结了微处理器的结构特点及学习方法，归纳了学习微处理器需要训练的基本功。

- 重点介绍 TMS320C2812 芯片。介绍其结构、指令系统、存储器组织、中断系统、片内外设及应用等。

- 加强了 DSP 最小系统介绍，使得初学者很容易建立起自己的硬件平台。
- 增加了 C 语言编程工程模板介绍，使得初学者很容易建立起自己的软件平台。
- 补充了一些应用实例。
- 对实验内容进行了更新，提供了不同层次的实验内容。
- 鉴于多数学生已掌握了 Matlab 仿真软件，第二版取消了 Matlab 算法仿真部分。

本修订版由宁改娣担任主编，负责制定编写提纲和全书统稿工作，并编写了第 1、4、8~12、14 章和附录；第 2、3、5~7、13 章由曾翔君编写；第 15 章和实验内容由骆一萍编写。本书部分编辑、实际例程的编写与调试由李晓东、许崇福、邬雪琴、王颖博、任伟、张雷、李平等参与完成，在此谨表示我们衷心的感谢！

本书的编写得到了西安交通大学张太镒教授的支持，第一版在评选西安交大优秀教材时，也得到张教授的鼎力推荐和好评，在此致以我们深深的谢意！

由于作者的水平和视野所限，书中错误和不当仍然在所难免，欢迎专家、学者、本书的读者提出意见和建议，以便今后不断改进。

作者

2008 年 12 月于西安交通大学

第一版前言

DSP 一方面是指数字信号处理技术, 另一方面是指数字信号处理器。数字信号处理器即 DSP 芯片的诞生, 使得数字信号处理技术的理论研究成果广泛应用到实际系统中, 成为数字信号处理技术和应用之间的桥梁, 并进一步促进数字信号处理技术的深入发展和应用领域的拓宽。本教材主要介绍数字信号处理器。

在微电子技术发展的带动下, DSP 芯片功能日益强大, 性能价格比不断提高, 开发环境日臻完善, 应用领域不断扩大。在步入数字化时代的进程中扮演着举足轻重的角色。

虽然早在 20 世纪 80 年代中期, DSP 芯片在国内已有许多应用, 但直到最近几年, 各大专院校才有相应的课程设置, 而且许多院校开设的课程都是以 TI 的 TMS320C5000 系列芯片为模型机进行介绍, 教材较少。介绍 TMS320C2000 系列及其应用的教材更是凤毛麟角。本教材就是为了方便教学和应用 DSP 控制器的科技人员参考为目的而编写的。

全书共分 8 章, 主要包括 3 个部分。

第一部分为基础部分, 包括第 1~3 章和第 5、6 章。主要介绍 DSP 芯片的发展、特点, TI 公司 TMS320DSP 各系列特点以及数字基础; 以 TMS320C24× DSP 控制器为模型介绍了其指令系统和硬件结构, 在对硬件深入介绍的同时还给出部分模块的扩展方法。本部分还详细介绍了 DSP 指令系统的寻址方式, 对各种指令只给出分类列表, 而对每条指令的理解和使用, 我们给大家提供了一个学习的途径和方法。

第二部分为应用开发部分, 包括第 4 章和第 8 章。主要介绍 DSP 开发流程、开发工具的使用及实验。在第一部分基础上, 介绍 DSP 的开发环境 (软件 CCS 和 PCI 仿真卡), 使得开发者能够很快熟悉开发环境, 缩短开发周期。根据以往的教学经验和学生的反馈意见, 对于学习微处理器芯片及其应用类课程, 实验是非常重要的一个环节。因此, 在第 8 章中提供了大量的实验内容, 部分实验给出了通过调试的源程序。

第三部分 (第 7 章) 为算法仿真, DSP 的主要特点是适用于数字信号处理。在具体实现 DSP 算法以前, 一般首先要对其进行模拟仿真以判断其正确性、复杂性和可靠性。以前 DSP 算法一般用 C 语言或其他高级语言来模拟实现, 但模拟过程繁琐、调试不便。目前这种模拟可以用 Matlab 快速方便地实现, 仿真结果可以转化为 C 语言。另外 Matlab 还可以与 TI 的 DSP 仿真环境通讯, 但这部分有待大家一起探索。由此可见, 借助 Matlab 将会非常有利于数字信号处理的实现。因此, 本书对 Matlab 仿真软件做简单介绍, 并给出通过调试的算法仿真程序。

本书由宁改娣和杨控科编写, 第 1、3 章由杨控科编写; 其余部分由宁改娣编写。在编写过程中, 参阅了 TI DSP 原始资料和国内外相关参考书, 学习和吸取了不少经验。在此向 TI

和作者致以谢意！在书稿的录入过程中，刘涛、杜倩宁、何永威、雍亮、廉海涛等同志参与了少部分工作，在此表示感谢！另外，书中一些实例是由选修 DSP 课程的部分研究生提供，陶坤宇工程师也对本书提出了许多建议，在此一并表示感谢！本教材能得以出版还要特别感谢樊捷和李洁的支持。本书由申忠如教授主审，并提出了许多宝贵意见，在此表示衷心的感谢。

我校 DSP 课程的开设以及本书的出版也得到电气工程学院各位领导及电工电子教学中心领导和同事们的大力支持和关心，在此表示诚挚的谢意。

由于时间仓促，加之作者的水平和掌握的资料有限，书中错误和不当在所难免，恳请读者批评指正。

作者

2002 年 6 月于西安交通大学

目 录

第二版前言

第一版前言

第 1 章 绪论	1
1.1 DSP 的概念与主要特点	1
1.1.1 DSP 的概念	1
1.1.2 DSP 的主要特点	1
1.2 DSP 芯片的发展及应用	3
1.2.1 DSP 芯片的发展	3
1.2.2 DSP 的分类	5
1.2.3 DSP 的应用	5
1.3 TMS320 系列 DSP	6
1.3.1 TMS320 系列 DSP 命名	6
1.3.2 TMS320 系列概况	6
1.3.3 TMS320C2000 系列	7
1.4 TMS320C28×系列 DSP 性能简介	8
1.4.1 TMS320C28×系列 DSP 的结构及性能	8
1.4.2 TMS320F281×的引脚分布及引脚功能	12
1.5 微处理器结构特点及学习方法	21
1.5.1 微处理器结构特点及学习目的	22
1.5.2 微处理器学习方法	23
1.5.3 TMS320C28×的片内外设学习指导	24
参考文献	25
第 2 章 F2812 的总体结构、最小系统及程序开发	26
2.1 F2812 的总体结构	26
2.2 F2812 的最小系统	29
2.3 F2812 的软件开发工具	31
2.4 F2812 的汇编语言编程基础	36
2.4.1 COFF 文件与汇编器指令	37
2.4.2 链接器与链接命令文件	39
2.5 F2812 的 C 语言编程及运行时环境	41
2.5.1 C 语言数据类型	41
2.5.2 C 语言关键词	42
2.5.3 C 语言编程头文件	46
2.5.4 预编译器指令	46

2.5.5 C/C++ 编译器默认的段和段的分配	49
2.6 F2812 的 C 语言编程工程模板	50
2.6.1 工程模板的头文件 (*.h)	52
2.6.2 工程模板的源文件 (*.c, *.asm) 和库文件 (*.lib)	57
2.6.3 工程模板的链接器命令文件 (*.cmd)	59
2.6.4 使用静态库缩短工程编译时间	61
2.6.5 编程举例	61
参考文献	62
第 3 章 C28× 的 CPU、存储器配置以及上电引导	63
3.1 TMS320C28× 的 CPU	63
3.1.1 CPU 的结构	63
3.1.2 CPU 的寄存器	67
3.2 F2812 的存储器配置	74
3.2.1 F2812 的片内存储器配置	74
3.2.2 外部存储器扩展和中断向量表	77
3.3 F2812 的上电引导	78
3.3.1 Boot ROM 简介	78
3.3.2 引导加载程序的工作流程	80
3.3.3 退出 ExitBoot 函数至执行 main 函数之间系统的动作	85
参考文献	87
第 4 章 C28× CPU 寻址方式	88
4.1 寻址方式	88
4.2 寻址方式选择位 (AMODE)	88
4.2.1 AMODE 对指令操作码的影响	88
4.2.2 汇编器/编译器对 AMODE 位的跟踪	89
4.3 直接寻址方式	89
4.4 堆栈寻址方式	90
4.5 间接寻址方式	91
4.5.1 C28× 间接寻址方式	91
4.5.2 C2xLP 间接寻址方式	93
4.5.3 循环间接寻址方式	95
4.6 寄存器寻址方式	96
4.6.1 32 位寄存器寻址方式	96
4.6.2 16 位寄存器寻址方式	96
4.7 数据/程序/IO 空间立即寻址方式	96
4.8 程序空间间接寻址方式	97
4.9 字节寻址方式与 32 位操作数的定位	98
4.9.1 字节寻址方式	98
4.9.2 32 位操作数的定位	99

4.10 C240×与 C28×指令系统的区别	99
参考文献	100
第 5 章 F2812 的系统控制单元及中断机制	101
5.1 F2812 的时钟和系统控制单元概述	101
5.2 F2812 的振荡器 OSC 和锁相环 PLL 时钟模块	103
5.3 F2812 的低功耗方式模块	106
5.4 F2812 的看门狗模块	110
5.5 F2812 的中断机制	112
5.5.1 TMS320F2812 中断功能的硬件结构	113
5.5.2 PIE 模块工作原理及 CPU 对可屏蔽中断的响应流程	114
5.5.3 PIE 中断优先级	116
5.5.4 不可屏蔽中断	118
5.5.5 外部中断 XINT1、XINT2 和 XNMI_XINT13	119
5.5.6 中断向量表	120
5.5.7 PIE 模块的配置和控制寄存器	122
5.5.8 在工程模板中对 PIE 向量表的定义和使用	124
5.5.9 中断的应用举例	128
参考文献	129
第 6 章 C28×的 CPU 定时器、GPIO 及外部存储器扩展	130
6.1 C28×的 CPU 定时器结构和工作原理	130
6.2 F2812 的 GPIO	135
6.2.1 GPIO 的结构和多路选择器寄存器	136
6.2.2 GPIO 的数据寄存器	140
6.3 F2812 的外部存储器扩展接口 (XINTF)	142
参考文献	150
第 7 章 模数转换器 (ADC)	151
7.1 ADC 模块的特点	151
7.2 ADC 模块排序器工作原理	153
7.2.1 级联模式下排序器的工作原理	153
7.2.2 双排序器模式的工作原理	155
7.2.3 排序器的输入触发和中断操作	158
7.3 ADC 模块的时钟	160
7.4 ADC 模块的低功耗模式和加电顺序	160
7.5 ADC 模块的寄存器	161
参考文献	167
第 8 章 事件管理模块 (EV)	168
8.1 EV 功能概述	168
8.2 通用定时器	169
8.2.1 通用定时器功能	170

8.2.2	通用定时器计数模式	173
8.2.3	通用定时器比较操作	175
8.3	全比较单元	177
8.3.1	与比较单元相关的 PWM 电路	177
8.3.2	可编程死区单元与输出逻辑	178
8.3.3	全比较单元的 PWM 输出	179
8.3.4	空间矢量 PWM (SVPWM) 的原理与实现	180
8.3.5	全比较单元寄存器	189
8.4	捕获单元和正交编码	191
8.4.1	捕获单元概述	191
8.4.2	正交编码脉冲电路	194
8.5	EV 中断逻辑	195
8.5.1	EV 中断概述	195
8.5.2	EV 中断寄存器	197
	参考文献	198
第 9 章	串行通信接口	199
9.1	F281x SCI 模块介绍	199
9.1.1	SCI 与 CPU 的通信及功能	199
9.1.2	波特率计算	201
9.1.3	数据格式 (帧格式)	201
9.1.4	SCI 的数据接收与同步	202
9.1.5	SCI 接收和发送时序及中断	202
9.2	SCI 多处理器通信	204
9.2.1	识别地址字节及接收过程	205
9.2.2	空闲线多处理器模式	205
9.2.3	地址位多处理器模式	206
9.3	C28x 系列 SCI 增强的功能	207
9.3.1	接收/发送 FIFO 的特点	207
9.3.2	SCI 自动波特率检测	208
9.4	SCI 的寄存器	208
9.5	SCI 应用实例——PC 机与 DSP 串行通信	215
9.5.1	硬件设计	215
9.5.2	软件设计	218
	参考文献	221
第 10 章	串行外围接口	222
10.1	F281x SPI 模块介绍	222
10.1.1	SPI 与 CPU 的通信及功能	222
10.1.2	SPI 的主从模式	224
10.1.3	SPI 数据传送及数据格式	225

10.1.4 波特率和时钟	226
10.2 SPI 的 FIFO 和中断	230
10.3 SPI 的寄存器	231
10.4 SPI 应用实例	234
10.4.1 硬件设计	235
10.4.2 软件设计	235
参考文献	237
第 11 章 多通道缓冲串行口 (McBSP)	239
11.1 McBSP 模块介绍	239
11.1.1 McBSP 的主要特点	239
11.1.2 McBSP 功能框图	240
11.1.3 McBSP 的数据传输	241
11.1.4 数据压缩与解压	242
11.1.5 时序和数据帧	243
11.1.6 帧分段	245
11.1.7 McBSP 的接收和发送	246
11.2 McBSP 内部时钟发生器	248
11.2.1 时钟源与极性配置	248
11.2.2 输出时钟 CLKG 的频率及同步	249
11.2.3 帧同步信号产生	249
11.2.4 时钟发生器输出和外部时钟的同步	250
11.2.5 时钟发生器的复位和初始化	251
11.2.6 时钟发生器产生时钟举例	252
11.3 McBSP 的意外/出错情况	253
11.4 多通道选择模式	255
11.4.1 通道、模块和分区	255
11.4.2 A-bis 模式	259
11.4.3 SPI 协议	260
11.5 配置接收器和发送器	264
11.5.1 接收器配置	264
11.5.2 发送器配置	265
11.6 McBSP 的仿真和复位事项	266
11.6.1 McBSP 的仿真模式	266
11.6.2 McBSP 复位	267
11.6.3 初始化 McBSP 串口	268
11.6.4 数据打包实例	268
11.7 McBSP 的 FIFO 和中断	270
11.7.1 McBSP 的 FIFO 概述	270
11.7.2 McBSP 的 FIFO 操作	271

11.7.3	McBSP 中断的产生	272
11.7.4	McBSP FIFO 数据寄存器访问约束和错误标志	273
11.8	McBSP 寄存器	275
11.9	McBSP 应用实例	279
	参考文献	282
第 12 章	增强型控制器区域网 (eCAN)	283
12.1	CAN 总线简介	283
12.1.1	CAN 总线特点	283
12.1.2	CAN 总线帧格式	284
12.2	C28x 系列的 eCAN 介绍	285
12.2.1	eCAN 的特点	285
12.2.2	eCAN 的结构与接口	285
12.2.3	eCAN 寄存器及存储器映射	287
12.2.4	消息邮箱	287
12.3	eCAN 配置	290
12.3.1	eCAN 初始化	290
12.3.2	eCAN 的配置步骤	293
12.3.3	远程帧邮箱的处理	295
12.3.4	eCAN 中断	296
12.3.5	eCAN 掉电模式	297
12.4	eCAN 应用实例	298
	参考文献	304
第 13 章	F2812 的片内 Flash 与 IQmath 库的应用	306
13.1	F2812 的片内 Flash	306
13.1.1	F2812 片内 Flash 的结构特点	306
13.1.2	F2812 片内 Flash 的配置寄存器	308
13.1.3	F2812 片内 Flash 配置寄存器的访问方法和限制	310
13.2	F2812 的代码安全模块 (CSM)	311
13.2.1	CSM 的密码保护机制	311
13.2.2	CSM 的加密和解密流程	313
13.3	F2812 片内 Flash 的配置和应用实例	316
13.4	F2812 程序开发中 IQmath 库的应用	321
13.4.1	Q 表示法与 IQmath 库的组成	321
13.4.2	IQmath 库的使用	322
13.4.3	IQmath 库的函数列表	324
13.4.4	使用 IQmath 库实现 SVPWM	326
	参考文献	328
第 14 章	现代 DSP 软件设计方法和简单应用	329
14.1	Matlab 与 DSP 工具简介	329

14.2 基于 Matlab 的 DSP 设计步骤	330
14.3 现代 DSP 软件设计生成的 Project 工程简介	333
14.3.1 Project 工程文件和运行	333
14.3.2 现代 DSP 设计与传统编程方法所形成的工程比较	334
参考文献	334
第 15 章 TMS320F2812 代码调试工具 (CCS) 教程	335
15.1 CCS 的特点	335
15.2 CCS v3.3 的安装和设置	336
15.2.1 CCS 的 Simulator 模式的设置	336
15.2.2 CCS 的 Emulator 模式的设置	336
15.3 实验系统的硬件连接和 CCS 的启动	338
15.4 CCS 应用程序功能描述	338
15.4.1 CCS 的工作界面介绍	338
15.4.2 CCS 的常用调试工具	340
15.4.3 CCS 的主菜单	340
基础实验	353
实验 1 简单输出设备 LED 实验	353
实验 2 基于 F2812 的数据采集和波形发生实验	356
实验 3 基于 F2812 的人机交互接口设计	360
参考文献	364
综合实验	365
实验 4 基于 F2812 的无刷直流电机控制	365
实验 5 基于 F2812 的 USB2.0 总线设备开发	373
实验 6 基于 F2812 和嵌入式操作系统的以太网接口开发	391
参考文献	402
附录	404
附录 1 eZdsp™ F2812 实验系统介绍	404
附录 2 eZdsp™ F2808 实验系统介绍	408

第 1 章 绪 论

本章将对 DSP 的概念、DSP 的主要特点、发展及应用、TI 的 TMS320 DSP、C28×系列 DSP 的主要性能以及引脚给予介绍。并列表对比 TMS320C2000 系列典型 DSP 芯片的资源配置情况。

1.1 DSP 的概念与主要特点

1.1.1 DSP 的概念

DSP 是一种适合数字信号处理的高性能微处理器，其结构与微型计算机原理或单片机原理课程中学习的处理器相比更为复杂。但这些课程学习方法类似，核心都是在介绍一个超大规模集成微处理器内部结构和指令系统，以及如何使用处理器实现数字控制，训练软硬件设计和调试技能。因此，学习 DSP 可以借鉴以前积累的微处理器软硬件开发经验。

DSP 有两个含义：其一是 Digital Signal Processing 的缩写，是指数字信号处理技术，它不仅涉及许多学科还广泛应用于多个领域。特别是在 20 世纪 60 年代，随着计算机和信息技术的迅猛发展，进一步推动了数字信号处理技术的理论和应用领域的发展；其二是 Digital Signal Processor 的缩写，即数字信号处理器（也称为 DSP 芯片），它不仅具有可编程性，而且其实时运行速度远远超过通用微处理器，是一种适合数字信号处理的高性能微处理器。多数情况下 DSP 是指数字信号处理器，DSP 已成为数字信号处理技术和实际应用之间的桥梁，并进一步促进了数字信号处理技术的发展，也极大地拓展了数字信号处理技术的应用领域。

在微电子技术发展的带动下，DSP 芯片的功能日益强大，性能价格比不断提高，开发环境日臻完善，应用领域也在不断扩大。在步入数字化时代的进程中，数字信号处理器扮演着举足轻重的角色。

1.1.2 DSP 的主要特点

与通用的微处理器（ μP ）或微控制器（MCU）相比，DSP 在硬件结构和指令体系上都采用了许多措施来提高数据处理速度。尽管各 DSP 生产厂商所采用的技术不尽相同，但一般 DSP 都具有以下一些共同的特点。

1) 改进的哈佛结构

通用的微处理器都采用冯·诺依曼结构，程序和数据存储空间合二为一，存储空间统一编址，且共享一套地址和数据总线，如图 1-1 (a) 所示。

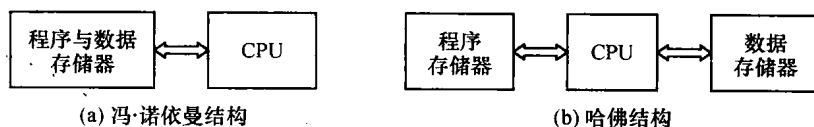


图 1-1 微处理器的冯·诺依曼结构与哈佛结构

哈佛结构是一种并行体系结构，主要特点是程序和数据存储空间独立。多数微控制器也采用的是哈佛结构，但与微控制器不同的是 DSP 内部采用了多总线技术，程序和数据有各自的总线，如图 1-1 (b) 所示，分离了数据流和指令流，使得读取指令和存取操作数可以完全重叠运行，提高了指令的运行速度。为了进一步提高速度和灵活性，TI 公司对 TMS320 系列 DSP 产品做了进一步的改进，一是允许程序存储在高速缓存 (Cache) 中，提高指令读取速度；二是允许数据存放在程序存储器中，并被算术运算指令直接使用，增强芯片的灵活性。另外，DSP 中的双口 RAM (DARAM) 及读写有各自的总线使数据存取速度更高。

2) 流水线技术

如果完成一条指令需要取指令、指令译码和执行指令三个步骤，在冯·诺依曼结构的微处理器中，由于取指令和存取数据要从同一个存储空间存取，经由同一总线传输。因此，每条指令按顺序取指令、指令译码和执行指令，然后再进行下一条指令的处理，即一条条指令都是顺序执行的，时间上没有重叠的步骤。

DSP 芯片广泛采用流水线技术，增强了处理器的处理能力。TMS320 系列流水线深度为 2~8 级不等，也就是说，处理器在一个时钟周期可并行处理 2~8 条指令，每条指令处于流水线的不同阶段。图 1-2 为 3 级流水线操作的例子。在 3 级流水线操作中，每条指令经过取指令、指令译码和执行指令 3 个独立的执行过程，在每个时钟周期有 3 条指令被激活。例如，在取第 N 条指令时，前一条指令 (即第 $N-1$ 条) 处于指令译码阶段，而第 $N-2$ 条指令则在执行阶段。从而提高了处理器的指令处理速度。

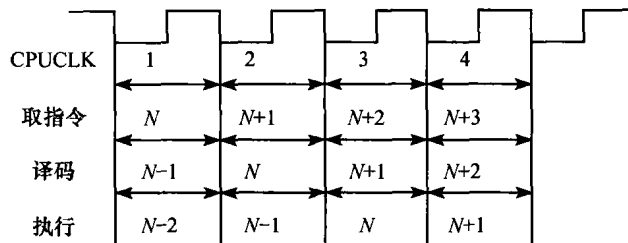


图 1-2 3 级流水线操作的例子

3) 硬件乘法器

在数字信号处理的许多算法中，需要做大量的乘法和加法运算。比如，数字滤波、FFT、卷积和矩阵运算等。通用处理器的乘法运算都是用软件来实现的，一次乘法往往需要多个机器周期才能完成。为了提高数字信号处理速度，DSP 的 CPU 中都无一例外地集成了硬件乘法器，而且一般都有乘累加指令，可以在一个机器周期完成两个操作数的相乘，并将乘积加到累加器中。

4) 多处理单元

DSP 与一般处理器类似，有中央算术逻辑单元 (CALU)，但是一般 DSP 都具有多处理单元，除了上述的硬件乘法器外，DSP 还设置了移位器和辅助寄存器算术单元 (ARAU)，ARAU 单元用来产生操作数的地址，操作数地址的产生与 CPU 的工作是并行的。这些都为高速度数字信号处理提供了完备的硬件基础。

5) 特殊 DSP 指令及零开销循环

DSP 芯片的另外一个特点就是采用了特殊的寻址方式和指令。比如，TMS320 系列的位

反转寻址方式, LTD、MPY、RPTK 等特殊指令。采用这些适合数字信号处理的寻址方式和指令, 进一步缩短了数字信号处理的时间。

数字信号处理 DSP 算法的一个共同的特点就是, 重复进行乘和累加。因此, 大多数的 DSP 内部都有硬件循环计数器进行自动循环计数处理, DSP 不用软件花时间去检查循环计数器的值, 硬件计数器使 DSP 具有零开销循环功能。

6) 片内存储器和强大的片内硬件配置

DSP 片内设置了多种类型的程序存储器、数据存储器等, 且容量不断增大, 缓解外部存储器访问的瓶颈现象。同时, 随着 DSP 的发展, DSP 逐步集成了串行口、主机接口 (HPI)、事件管理模块、DMA 控制器、CAN 总线控制模块、软件等待状态发生器、锁相环、A/D 模块等片内外设。

7) JTAG 标准测试接口

通过 JTAG 标准测试接口 (IEEE 1149 标准接口) 和专用的仿真器, 支持 DSP 的在线仿真和多 DSP 条件下的调试。与单片机替代式的仿真方式相比, DSP 仿真时序与目标系统一致, 仿真不占用系统资源。DSP 应用系统的开发更为方便。

上述的 DSP 特征, 以及 DSP 的时钟频率提高, 执行周期的缩短, 丰富的片内外设和种类多样的片内存储器, 使得 DSP 成为迅速发展的数字信号处理技术与实现之间的一个桥梁。

1.2 DSP 芯片的发展及应用

1.2.1 DSP 芯片的发展

在数字信号处理技术发展的初始阶段, 人们只能在通用的计算机上进行算法的研究和系统的模拟与仿真。随着数字信号处理技术和集成电路技术的发展, 以及数字系统的显著优越性, 导致了 DSP 芯片的产生和迅速发展。

第一片 DSP 器件是 1978 年 AMI 公司宣布的 S2811。

1979 年 Intel 公司推出的 Intel 2920 是第一块脱离了通用型微处理器结构的 DSP 芯片, 成为 DSP 芯片的一个主要里程碑。

1980 年前后, 日本 NEC 公司推出的 μ PD7720 是第一个具有硬件乘法器的商用 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 公司推出的 MB8764, 其指令周期为 120ns, 且具有双内部总线, 从而使处理器的数据吞吐量发生了大的飞跃。第一片高性能的浮点 DSP 芯片应是 AT&T 公司于 1984 年推出的 DSP32。

1982 年前后, 美国德州仪器公司 (Texas Instrument, TI) 成功推出其第一代 DSP 芯片 TMS32010 及其系列产品 TMS32011、TMS32C10/C14/C15/C16/C17 等, 之后相继推出了第二代 DSP 芯片 TMS32020、TMS320C25/C26/C28, 第三代 DSP 芯片 TMS32C30/C31/C32, 第四代 DSP 芯片 TMS32C40/C44, 第五代 DSP 芯片 TMS32C50/C51/C52/C53 以及集多个 DSP 于一体的高性能 DSP 芯片 TMS32C80/C82, 第六代为更高性能的 TMS320C64 \times /C67 \times 和高性能的 DSP 控制器 C28 \times 等。

Motorola 公司, 1986 年推出 MC56001 定点 DSP 芯片, 1990 年推出了与 IEEE 浮点格式兼容的 MC96002 浮点 DSP 芯片。Motorola 的 DSP 芯片上设置了一个 OnCE (On-Chip Emu-

lation) 片上仿真单元, 其功能包括检查片内的寄存器、存储器及外设, 用单步、断点和跟踪等方式控制和调试程序。目前在 DSP 市场仍有一定影响。

美国模拟器件公司 (Analog Devices, AD) 也相继推出了一系列具有自己特点的 DSP 芯片, 在 DSP 市场上也占有一定份额。ADSP21×× 为定点处理器, 定点 DSP 有 ADSP2101/2103/2105、ADSP2111/2115、ADSP2161/2162/2164 以及 ADSP2171/2181。ADSP21××× 为浮点处理器, 浮点 DSP 有 ADSP21000/21020 以及 ADSP21060/21062 等。

另外还有许多其他厂家生产 DSP, 市场占有率排名前 4 位的公司有 TI、Agere (原 Lucent, 中文名为朗讯)、Motorola 和 ADI。它们的市场份额分别是: TI 占 43.5%, Agere 占 16.1%, Motorola 占 12.0%, ADI 占 8.2%。2001 年 DSP 市场总额为 42.6 亿美元^[1]。

自 20 世纪 80 年代, DSP 芯片得到突飞猛进的发展, DSP 芯片的应用越来越广泛。从运算速度来看, MAC (一次乘法和一次加法) 时间已经从 80 年代初的 400ns (如 TMS32010) 减小到 10ns 以下 (如 TMS32C54×), 处理能力提高了几十倍。DSP 芯片内部关键的硬件乘法器占模片区 (Die Area) 从 1980 年的 40% 左右下降到 5% 以下, 片内 RAM 增加一个数量级以上。从制造工艺来看, 1980 年采用 NMOS 工艺, 而现在则普遍采用亚微米 CMOS 工艺。DSP 芯片的引脚数量从 1980 年的最多 64 个增加到现在的 200 个以上, 引脚数量的增加, 意味着结构灵活性的增加。

随着 DSP 的时钟频率和处理速度越来越高, 功能越来越强, 芯片的功耗也急速加大。尽管各生产厂家几乎无一例外地采用 CMOS 工艺等技术来降低功耗, 但对于用于电池供电的便携式设备 (如笔记本电脑、移动通信设备和 PDA 等) 中的 DSP, 迫切要求在提高性能的同时, 进一步降低工作电压, 减小功耗。为此, 各 DSP 生产厂家陆续推出低电压 (3.3V/2.7V/1.8V 等) DSP 芯片, 多数的 DSP 芯片还设置了多种低功耗工作方式。

未来 10 年, 全球 DSP 产品将向高性能、低功耗、加强融合和拓展多种应用的趋势发展, DSP 芯片将越来越多地渗透到各种电子产品中, 成为各种电子产品尤其是通信类电子产品的技术核心, 将会越来越受到业界的青睐。据 TI 预测, 到 2010 年, DSP 芯片的集成度将会增加 11 倍, 在单个芯片内将能集成 5 亿只晶体管。

随着 DSP 应用的日益广泛, DSP 已成为许多应用系统设计中不可缺少的组成部分, 其结果使 DSP 厂商的投资集中于 DSP 体系结构的完善和支持软件的升级。比如, TI 为 TMS320 系列提供了 eXpressDSP 实时软件技术的支持, 它包括 Code Composer Studio 集成开发环境、DSP/BIOS 实时软件内核、TMS320 算法标准以及第三方提供的可重用的软件模块。

由于汇编语言是面向机器的, 具体地说是面向芯片的, 即不同厂商的 DSP 有不同的汇编语言指令系统, 使用汇编语言编写 DSP 应用软件是一件烦琐与困难的事。随着 DSP 寻址空间越来越大, 减小了对程序目标代码容量的限制。因此, 各公司陆续推出了高级语言编译器, 主要是 C 语言编译器, 它可以将 C 语言程序编译并优化处理成相应的 DSP 汇编程序或目标程序。对于 TI 的高性能 TMS320C6000 系列及 TMS320C2000 的新成员 C28×, 用 C 语言编程效率非常高, 代码优化可达 85% 左右, 缩短了软件开发周期, 程序可移植性好。

我国 DSP 技术起步早, 基本上与国外同步发展。我国已有上百所大学从事 DSP 的教学和科研, 在信号处理理论和算法上与国外处于同一水平。但 DSP 芯片几乎完全依赖进口, 其中 TI 公司产品占 80% 以上^[2]。

在 VLSI 技术高速发展的基础上, DSP 芯片的性能不断提高、价格连续下降、单位运算量的功耗显著降低。DSP 自身设计也在不断采用新的技术, 并与其他处理器技术不断融合, 朝