

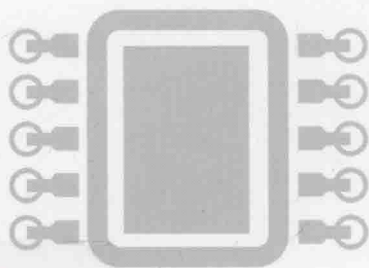
马俊杰 编著

嵌入式

DSP 的

原理与应用

——基于TMS320F28335



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

嵌入式 DSP 的原理与应用 ——基于 TMS320F28335

马俊杰 编著

北京航空航天大学出版社

内 容 简 介

本书以 TMS320F28335(简称 F28335)的工程应用为主线,从电力电子产品的嵌入式应用角度出发,介绍了其软硬件的开发方法;例程内容力求涵盖电力电子技术中所有的控制、算法、逻辑等操作。本书以提高读者的 DSP 应用水平为目的,本着循序渐进的原则,前半部分基础简单,后半部分难度较高,详细介绍了浮点算法开发、程序固化、FLASH 片上升级优化、用户自定义 BootLoader 等内容。本书例程不仅调试通过而且其基本思想均应用于目前主流电力电子产品中,具有很高的参考和实用价值。

本书摒弃以往翻译 TI 数据手册的弊端,结构更加严谨,内容更加注重理论与实际的结合,可作为自动化专业、电气工程专业研究生和工程技术人员的参考用书。

图书在版编目(CIP)数据

嵌入式 DSP 的原理与应用:基于 TMS320F28335 / 马骏杰编著. — 北京:北京航空航天大学出版社,2016.3
ISBN 978-7-5124-2067-0

I. ①嵌… II. ①马… III. ①数字信号处理 IV.
①TN911.72

中国版本图书馆 CIP 数据核字(2016)第 046896 号

版权所有,侵权必究。

嵌入式 DSP 的原理与应用——基于 TMS320F28335

马骏杰 编著

责任编辑 冯 颖

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@buaacm.com.cn 邮购电话:(010)82316936

北京泽宇印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:26.25 字数:559 千字

2016 年 3 月第 1 版 2016 年 3 月第 1 次印刷 印数:3 000 册

ISBN 978-7-5124-2067-0 定价:59.00 元

前言

随着电力电子技术及相关控制理论的快速发展,高性能、高精度、高集成度的电力电子产品目前已经成为市场主流,这也对控制器性能的要求越来越高。TMS320F2833x(简称 F2833x)属于 TI 公司 TMS320C2000(简称 C2000)系列 DSP 产品,具有强大的数字信号处理功能。

2004 年我第一次接触的 DSP 是 2812,遗憾的是只应用了其中的某几个模块。后来加入到全球知名的某电力电子公司从事嵌入式开发工作,系统地在 TI 公司开发的 2812、2803x、28016 等 Piccolo 系列 DSP、F2833x 系列 Delfino 以及 TMS320F2837xD 双核系列 DSP 的基础上进行了产品开发。如果说实验室的工作只是在 DSP 的基础上做一些理论研究,分散、独立地使用 DSP 的各个模块,那么工作中尤其是对于产品的设计,我们会尽可能地去挖掘 DSP 的功能,甚至觉得原本丰富的资源都变得如此紧张了。我在参观 TI 公司总部时与其工程师的交谈更加明确了这一点:C2000 系列并非只是功能强大的“单片机”,而是各个模块内部相互强化,应用在电力电子控制领域的数字信号处理器。

本书是在严格参考 TI 数据手册的基础上结合作者多年的 DSP 开发经验整理、归纳而成的,旨在达到理论与实践的结合、知识与案例的统一,注重培养读者运用知识创新能力和解决实际问题的工程能力。

全书共分为 12 章。第 1~8 章为 F2833x 的基本应用,内容涵盖 F28335 几乎所有模块,对比了 2812、2808 等 DSP 与 F2833x 系列的不同点,旨在希望读者快速掌握不同系列的特点。每章的应用实例不仅调试通过且其思想均已应用在市场主流电力电子产品中,具有很高的应用和参考价值。第 9~12 章属于 DSP 的高级应用,难度较高但工程实用性很强。第 9 章系统地分析了 FPU 模块,归纳总结了浮点指令及库函数应用,其中加入了大量的例程也是希望读者能够掌握其指令特点。第 10 章和第 11 章系统、详尽地分析了 API 的使用,BootLoader 和目前常见的片上 FLASH 升级方案。其中如何自行建立 Boot 使其具有远程 FLASH 升级功能有相当高的工程应用价值。第 12 章为电力电子技术的应用,采用电力电子技术基本理论与 DSP 应用相结合,其中所包含的内容均代表了当下电力电子控制领域核心、常用、流行的元素,书中所给出的程序及其思想也为读者扩展思维提供了帮助。

本书由哈尔滨理工大学的马骏杰编写并统稿,张思艳老师参与编写了第 4 章和

前言

第6章。感谢李全利教授、王旭东教授和王哈力教授对本书出版的关心,感谢那些年陪伴我在电力电子科研道路上奋斗过的小伙伴们,感谢TI深圳办事处的工程师对本书编写提出的宝贵意见,还要感谢我的父母、岳父母、妻子给我的关爱和支持,并将此书献给我即将出生的宝贝“子越”。

此外,本书还得到了汽车电子驱动控制与系统集成教育部工程研究中心、国家自然科学基金(51177031)、黑龙江省科技攻关项目(GB08A306)的大力支持,在此对相关的老师与同学表示衷心感谢!

由于作者水平有限,书中不妥之处在所难免,敬请读者提出宝贵意见和建议(作者邮箱:mazhencheng1982@sina.com)。

作者

2015年9月13日

目 录

第1章 绪 论	1
1.1 数字信号处理	1
1.1.1 模拟和数字信号的处理方式	1
1.1.2 数字信号处理的优缺点	2
1.2 DSP 的发展及特点	3
1.3 DSP 产品简介	5
1.3.1 TI 公司的 DSP 产品	5
1.3.2 TMS320C28x 产品简介	6
第2章 F28335 的结构原理	8
2.1 F28335 的内部结构	8
2.2 F28335 的 CPU	10
2.2.1 CPU 运算单元	10
2.2.2 CPU 寄存器组	12
2.2.3 辅助寄存器算术单元 ARAU	14
2.2.4 FPU 浮点运算单元	15
2.3 F28335 的存储器及应用	15
2.3.1 SARAM 存储器	17
2.3.2 FLASH 及使用详解	17
2.3.3 Boot ROM	22
2.3.4 CSM 代码安全模块及使用详解	24
2.3.5 OTP 存储器	31
2.3.6 外设帧	31
2.3.7 外部存储器接口	32
2.4 外设及引脚功能	33

目 录

2.5 F28335 的时钟及控制	36
2.5.1 系统时钟的产生	36
2.5.2 系统时钟的分配	37
2.5.3 F28335 的低功耗模式及相关寄存器	42
2.5.4 F28335 的看门狗模块	44
2.6 F28335 的 CPU 定时器	47
2.6.1 定时器结构	47
2.6.2 定时器中断申请	48
2.6.3 定时器寄存器及位域结构体定义	48
2.6.4 定时器应用例程——如何记录函数的运行时间	51
第3章 集成开发环境及程序开发语言	53
3.1 CCS 集成开发环境	53
3.1.1 CCS 集成的工具软件	53
3.1.2 CCSv5.4 安装及基本配置	55
3.1.3 Control Suite 简介	59
3.1.4 在 CCSv5.4 下运行工程项目	59
3.1.5 CCSv3.3 到 CCSv5.4 的项目迁移	66
3.1.6 在 CCSv5.4 下新建一个工程文件	68
3.2 F28335 汇编语言概述	74
3.2.1 F28335 汇编指令描述	75
3.2.2 寻址方式及常用汇编指令	76
3.2.3 CMD 文件及汇编程序示例	80
3.2.4 汇编语言应用例程	82
3.3 F28335 的 C 语言编程基础	83
3.3.1 F28335 的 C 语言数据类型	84
3.3.2 C 语言的重要关键字	85
3.3.3 C 语言 CMD 文件的编写	86
第4章 F28335 的通用 I/O 口	90
4.1 GPIO 的功能结构	90
4.2 GPIO 寄存器及传统定义方法示例	92
4.2.1 GPIO 控制类寄存器	92
4.2.2 GPIO 数据类寄存器	94
4.2.3 传统寄存器定义方法示例	95
4.3 寄存器的位域结构方法示例	96

4.3.1	GPIO 寄存器组类型定义	96
4.3.2	定义存放寄存器组的存储器段	101
4.3.3	寄存器组的存储器段地址定位	102
4.3.4	寄存器位结构定义的使用	102
4.4	GPIO 应用例程	103
第 5 章	F28335 的中断系统	104
5.1	中断系统的结构	104
5.1.1	中断管理机制	104
5.1.2	中断处理及响应过程	106
5.1.3	CPU 中断向量	106
5.1.4	CPU 级中断相关寄存器	108
5.2	PIE 外设中断扩展模块	109
5.2.1	PIE 模块的结构	109
5.2.2	PIE 中断向量表映射	110
5.2.3	PIE 模块相关寄存器	117
5.2.4	PIE 模块寄存器的程序操作	119
5.3	非屏蔽中断	123
5.3.1	软件中断	123
5.3.2	非法指令中断	124
5.3.3	硬件 NMI 中断	124
5.3.4	硬件复位中断 XRS	125
5.4	中断应用实例——如何创建中断嵌套服务程序	125
第 6 章	模/数转换单元 ADC	127
6.1	ADC 模块概述	127
6.1.1	ADC 模块构成及原理	127
6.1.2	时钟及采样频率	130
6.1.3	ADC 采样工作方式	131
6.2	ADC 模块校准功能及使用详解	134
6.3	ADC 模块寄存器	136
6.4	ADC 性能实验分析	143
6.5	ADC 模块的应用——如何进行数据处理及程序校准	146
第 7 章	F28335 片上控制外设	148
7.1	增强型脉宽调制模块 ePWM	148

7.1.1	时间基准子模块 TB 原理及应用	150
7.1.2	计数器比较子模块 CC 原理及应用	155
7.1.3	动作限定子模块 AQ 原理及应用	157
7.1.4	死区控制子模块 DB 原理及应用	163
7.1.5	错误控制子模块 TZ 原理及应用	165
7.1.6	事件触发子模块 ET 原理及应用	169
7.2	增强型捕获模块 eCAP	174
7.2.1	eCAP 工作模式	174
7.2.2	捕获模块寄存器	177
7.2.3	eCAP 程序例程——如何捕获外部脉冲信号	183
第 8 章	F28335 的片上串行通信单元	185
8.1	串行通信的基本概念	185
8.1.1	异步通信和同步通信	185
8.1.2	串行通信的传输方向	186
8.1.3	串行通信的错误校验	186
8.2	SCI 通信模块及应用	187
8.2.1	SCI 通信模块简介	187
8.2.2	SCI 工作原理	188
8.2.3	SCI 基本数据格式	188
8.2.4	多处理器通信方式	190
8.2.5	SCI 相关寄存器	191
8.2.6	SCI 应用实例——如何实现异步通信数据的收发	194
8.3	SPI 通信模块及应用	195
8.3.1	SPI 模块简介	195
8.3.2	SPI 工作原理	196
8.3.3	SPI 的时钟与波特率	198
8.3.4	SPI 相关寄存器	198
8.3.5	SPI 模块应用实例——如何建立有效的全双工数据通信	202
8.4	McBSP 模块及应用	203
8.4.1	McBSP 数据收发原理	204
8.4.2	数据的压缩和扩展	205
8.4.3	McBSP 数据帧	205
8.4.4	时钟及采样率发生器	207
8.4.5	McBSP 工作模式简介	210
8.4.6	McBSP 相关寄存器	210

8.4.7 McBSP 的应用——如何实现在 SPI 模式下的数据收发	218
8.5 I ² C 通信模块及应用	220
8.5.1 I ² C 总线基础	220
8.5.2 I ² C 相关寄存器	223
8.5.3 I ² C 应用实例——EEPROM 数据的读/写	227
8.5.4 I ² C 真实波形数据格式分析	229
8.6 CAN 通信模块	232
8.6.1 CAN 模块简介	232
8.6.2 CAN 相应寄存器	233
8.6.3 CAN 应用注意事项及数据收发程序详解	239
第 9 章 浮点运算单元	245
9.1 浮点单元简介	245
9.1.1 C28x+FPU 的特点	245
9.1.2 浮点指令流水线结构	246
9.1.3 IEEE754 单精度浮点格式	247
9.2 FPU 的寄存器	248
9.3 浮点汇编指令详解	250
9.3.1 移动指令	251
9.3.2 浮点算术运算指令	259
9.3.3 寄存器数据传递指令	272
9.3.4 特殊运算指令	275
9.3.5 寄存器清 0 指令	276
9.4 F28335 库函数使用详解	277
9.4.1 FPU Fast RTS 库简介	278
9.4.2 FPU Fast RTS 库使用方法	279
9.4.3 FPU Fast RTS 库软件优化	280
9.4.4 FPU 软件应用实例	281
第 10 章 BootLoader 原理及应用	283
10.1 BootLoader 基本工作流程	283
10.1.1 F2833x 片内引导过程	284
10.1.2 基本工作流程代码解析	285
10.2 BootLoader 基本数据传输协议	289
10.2.1 16 位数据流结构	289
10.2.2 8 位数据流结构	290

10.2.3	数据引导装载过程	291
10.2.4	数据格式转换	292
10.3	引导模式之跳转模式	294
10.3.1	FLASH 上电复位跳转模式及代码解析	294
10.3.2	片上其他跳转模式	296
10.4	引导模式之加载模式	297
第 11 章	打开 FLASH 升级的“潘多拉”盒子	299
11.1	F2833x FLASH 烧录基础	299
11.1.1	FLASH 烧录的一般要求	299
11.1.2	FLASH 烧录步骤	300
11.2	CCS 插件升级方式	300
11.2.1	CCSv3.3 版本下的 FLASH 升级	300
11.2.2	CCSv5.4 版本下的 FLASH 升级	301
11.3	SDFLASH 插件操作方式	305
11.3.1	SDFLASH 的串行升级基本操作	305
11.3.2	如何更改串行升级文件	310
11.4	用户自定义升级方式	311
11.4.1	FLASH API 的应用解析	311
11.4.2	基于 SCI 总线的远程 FLASH 加载方案	315
第 12 章	基于 F28335 的电力电子应用案例分析	323
12.1	数据定标	323
12.2	电路基本变量数学建模及实现	324
12.2.1	数学模型的搭建	324
12.2.2	数学模型的软件实现	327
12.3	电力电子常见拓扑及发波算法分析	328
12.3.1	单相半桥电路及 SPWM 的 DSP 应用	328
12.3.2	单相全桥电路及单极倍频 SPWM	334
12.3.3	三相桥式电路及 SVPWM 相关算法应用	342
12.3.4	三相四桥臂电路及 3D-SVPWM 算法应用	356
12.3.5	三电平电路及 DSP 应用	366
12.4	三相 PWM 整流器设计	371
12.4.1	三相坐标变换基础	371
12.4.2	三相 PWM 整流器的数学模型	372
12.4.3	控制器的数学模型及系统设计	376

12.5 数字锁相环设计.....	379
12.5.1 锁相环的工作原理.....	379
12.5.2 锁相环的数学建模.....	380
12.5.3 算法分析.....	381
12.5.4 软件代码详解.....	383
12.6 数字滤波器的设计.....	385
12.6.1 FIR 滤波器的数学模型及算法设计.....	385
12.6.2 IIR 滤波器的数学模型及算法设计	387
12.7 基于 F28335 有源滤波器设计	389
12.7.1 谐波的基本概念.....	390
12.7.2 并联 APF 工作原理	391
12.7.3 数学模型及算法分析.....	393
12.7.4 DFT 变换法控制器系统设计	395
12.7.5 软件算法关键代码解析.....	396
附录 A CRC 数据表	402
附录 B SCI Boot 参考代码	404
参考文献.....	407

第 1 章

绪 论

随着计算机技术和集成电路技术的飞速发展,数字信号处理 DSP 技术在国防科技、工业控制、消费类产品等诸多领域得到了广泛的应用。

为了使读者对数字信号处理技术在控制领域及数字信号处理领域的应用有一个整体的认识,本章将对数字信号处理的概念、数字信号处理器的特点、数字信号处理系统的开发过程等进行介绍。

1.1 数字信号处理

数字信号处理的英文简称为 DSP,对它可以有两种理解:

① Digital Signal Processing,即数字信号处理技术,讨论数字信号的理论方法,是一个重要学科。

② Digital Signal Processor,即数字信号处理器,它是一个微处理器(Micro Processor),是专门用于数字信号处理的处理器。本书讨论的 DSP 技术即是通过数字信号处理器来实现数字信号的理论方法。

1.1.1 模拟和数字信号的处理方式

1. 模拟信号

模拟信号(Analog Signal)是与离散的数字信号相对的,在幅值和时间上都是连续变化的信号,例如现实生活中的电压信号、电流信号、温度信号等。

2. 数字信号

数字信号(Digital Signal)是离散时间信号(Discrete-Time Signal)的数字化表示,通常可由模拟信号(Analog Signal)经过模/数(Analog to Digital)采样获得,在时间上是离散的。

3. 信号处理的方式

信号是信息的载体,信息反映系统的特征。一方面,可用频谱仪进行信号分析,了解其频谱分布、噪声、杂散等,从而加深对信号特性的理解;另一方面,可对信号进行处理,即改变信号的某些特性来满足要求。

信号处理分为模拟和数字两种方式:前者可使用集成运算放大器及 R、L、C 电路元件来构成 PID 调节器,从而实现信号的提取和处理;后者采用模拟信号的数字化方式来进行数字的 PID 调节,如图 1.1 所示。

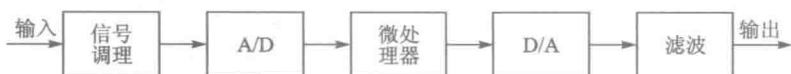


图 1.1 数字信号处理方式

首先模拟信号输入,经过信号调理(Signal Conditioning)去噪、整形处理后,达到理想的模拟信号;然后进行模/数转换,将模拟信号通过 A/D 模块变为数字量输入 DSP 进行相关的数字算法处理,一方面可用于数据传输(Data Transmission),另一方面可通过 D/A 输出,在经过 Filter(信号滤波)和功率放大后作为系统输出。

1.1.2 数字信号处理的优缺点

1. 数字信号处理的优点

数字信号处理伴随数字系统的发展从 20 世纪 40 年代开始,发展历史并不长,它具有数字系统的一些共同优点,例如抗干扰、可靠性强、便于大规模集成等。与传统的模拟信号处理方法相比较,它具有以下优点。

(1) 灵活性高

模拟处理系统:需要修改硬件设计或调整硬件参数。

数字处理系统:只需要改变软件参数设置。

以滤波器的设计为例:对于模拟滤波器,若要改变滤波器的参数,则需要改变元器件的值,或是改变运放的工作状态;对于数字滤波器,只需要改变程序的参数就可以了。由此可明显看出,数字处理系统的灵活性更高。

(2) 精度高

模拟处理系统的精度严重依赖于元器件精度,但元器件都有很大的离散性,因此构成的系统误差较大。

数字处理系统的精度依赖于 A/D 转换的位数、计算机字长及相关的算法。这样不仅可以估算出所设计系统的精度,还可以采用适当的处理方式来进行进一步提高其处理精度。

(3) 可靠性高,可重复性好

模拟系统受环境温度、噪声、电磁场等影响较大,即使两个设计参数完全相同的运算放大器,在输入同源的情况下其输出也会不同;但数字系统不同,在任何时间、任何地点、任何环境下,只要系统工作正常,那么计算结果肯定相同。

2. 数字信号处理的缺点

尽管数字信号处理有其明显的优势,但不能说明模拟信号处理能够被完全替代。

例如在乘法器的设计中,如果原始数据和所需要的输出均为模拟形式,那么完成从模拟信号向数字信号转换所需要的模/数转换器和数/模转换器的成本及其复杂程度通常会超过模拟乘法器设计本身。数字信号处理的劣势体现在如下几个方面。

(1) 实时性差

数字系统由计算机的处理速度决定。尽管计算机技术不断发展,处理速度也得到了极大的提高,但 DSP 都是由一个系统时钟来决定它的一个机器周期。机器周期是 DSP 处理速度的最小单位。因此在要求实时性能极高的情况下,模拟系统不能够被完全替代。

(2) 高频信号的处理

模拟系统可处理很高的频率,包括微波信号、光信号;而数字系统必须按照奈奎斯特准则的要求,受模/数转换和处理速度的限制。对于高频信号的处理,目前使用数字系统是很难实现的。

(3) 模拟与数字信号的转换

现实世界的信号绝大部分是模拟的,例如温度、湿度、压力等通过相应的传感器转换成的电信号也是模拟的。要实现数字处理就必须要进行模/数转换,因此这部分处理电路也必须是模拟的,所以模拟处理的环节肯定不能被替代。

1.2 DSP 的发展及特点

20 世纪 70 年代大规模集成电路的发展导致微处理器的产生,之后微处理器一直沿着以下几个方向发展:

① 通用处理器 General Purpose Processor:简称 GPP,以大家熟知的 286、386、486、奔腾系列 CPU 为代表。

② 单片机 Single Chip Computer:更多的时候被称之为 MCU(Micro Controller Unit,微控制器),它是为中、低成本控制领域而设计开发的。

③ 数字信号处理器 DSP:DSP 的产生发展同前面讲到的 GPP、MCU 几乎是同时的,随着应用市场的变化,20 世纪 90 年代后期得到了非常广泛的应用。尤其在 2000 年之后,DSP 在硬件结构上更适合数字信号处理的要求,能进行硬件乘法和单指令滤波处理。

DSP、单片机以及 GPP 三者各有所长,随着技术的发展使得三者之间相互借鉴对方的优点,取长补短。

DSP 可以看作是单片机的一个分支,也可看作是一个微型 CPU。DSP 具有较高的集成度,具有速度更快的 CPU 和容量更大的存储器。它采用改进的哈佛结构,具有独立的程序和数据空间,并内置高速的硬件乘法器以及增强的多级流水线,使 DSP 器件具有高速的数据运算能力。DSP 器件还提供了高度专业化的指令集,提高了 DFT 快速傅里叶变换和滤波器的运算速度。此外,DSP 器件提供外部 JTAG 接

第 1 章 绪 论

口,具有更先进的开发手段,不占用用户任何资源。软件配有汇编/链接 C 编译器、C 源码调试器等。

DSP 作为一种微处理器,其结构总是与其他处理器有类似的地方,如包含 CPU、存储器、总线、外设、接口等,但又有自己独特的特点。

1. 哈佛结构

计算机和微处理器有两大结构:冯·诺依曼(von Neuman)结构(如图 1.2 所示)和哈佛(Harvard)结构(如图 1.3 所示)。从 20 世纪 40 年代开始,这两种结构共存。



图 1.2 冯·诺依曼结构

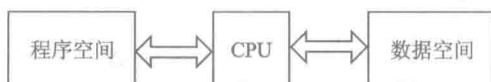


图 1.3 哈佛结构

以奔腾为代表的通用 CPU 无一例外地都采用了冯·诺依曼结构,它将程序和数据都存储在一起,然后通过总线与 CPU 相连。哈佛结构则把程序和数据空间分开了,分别使用程序和数据总线与 CPU 相连。几乎所有厂家的 DSP 都采用了哈佛结构,该结构使得取指和读数可以同时进行,提高了运算的速度。目前已到了每秒进行 90 亿次浮点运算(9 000 MFLOPS)的水平。

2. 流水线操作

DSP 的另一个重要特点是流水线操作(Pipeline),也有不少文献将其翻译成“管道”操作。TMS320F28x 系列 DSP 将指令执行分成 8 个阶段,如图 1.4 所示。

指令地址生成	取指	指令译码	操作数地址生成	操作数寻址	取操作数	执行	存储	
	指令地址生成	取指	指令译码	操作数地址生成	操作数寻址	取操作数	执行	存储
		指令地址生成	取指	指令译码	操作数地址生成	操作数寻址	取操作数	执行
			指令地址生成	取指	指令译码	操作数地址生成	操作数寻址	取操作数
				指令地址生成	取指	指令译码	操作数地址生成	操作数寻址
					指令地址生成	取指	指令译码	操作数地址生成
						指令地址生成	取指	指令译码
							指令地址生成	取指

图 1.4 TMS320F28x 流水线保护机制

DSP 采用了哈佛结构,这为实施流水线设计提供了条件。其操作步骤如图 1.4 所示:每一个操作都可以认为是一个节拍,假设一个节拍需要一个机器周期,那么完成这个操作需要 8 个机器周期。

通用的 CPU 采用串行操作,完成前一条之后才能操作后面一条。DSP 不是这样,在第一条指令执行译码的时候,取址功能已经释放,可供第二条使用。以此类推,后一条指令比上一条推迟一拍,每一条指令都会完成这 8 个操作。当流水线填满时,它可同时执行 8 条指令,平均每一条指令只需要一个机器周期,这也是很多 DSP 文档中所说的“单周期指令”。近年来,像奔腾之类的 CPU 也大量应用这种流水线操作方式。

3. 独立的乘加器

在数字信号的理论学习中可以知道,在进行卷积、数字滤波、FFT、矩阵等一类运算时,需大量重复的乘法和累加。为提高其运算速度,DSP 中设立了独立的硬件乘法器,即用 MAC 单元。

该硬件乘法器可以实现 32 位×32 位或 2 个 16 位×16 位的乘法。将乘法中的一个乘数通过数据总线传给 T 寄存器,被乘数通过数据总线或程序总线直接送到 MAC 单元,经硬件乘法器相乘后将结果存在 P 寄存器。P 寄存器中的结果经过加速器(Accelerator,在此为 Shift 移位器)后送入算术逻辑单元(ALU),实现与其他数据的算术逻辑运算。计算的结果一方面通过数据总线存入数据空间,另一方面作为 ALU 的输入,以此实现如公式 $ACC = \sum X_i \times Y_i$ 所示的累加运算。

4. 独立的 DMA 总线和控制器

前面看到的 MAC 硬件乘加器能够使数据运算速度大大提高,但取数的过程需要通过总线从存储器存取。DMA 指的是存储器的直接存取(Direct Memory Access),包含读、写两个过程。所谓的“直接”是指不经过 CPU 干预,通过独立的 DMA 控制器和 DMA 总线完成上述操作。

DMA 并不是 DSP 独有的,通用 CPU 中也存在这种概念。尽管通用 CPU 存在独立的 DMA 控制器,但存取过程依旧占用 CPU 总线,所以 CPU 在这个 DMA 过程只能空闲,算是间接占用 CPU 资源。因此,通常所说的 DSP 运算速度快,不仅需要 MAC 单元参与工作,还与运算的吞吐率有关。随着 DSP 技术的发展,DMA 的吞吐率还再提高,目前已达到 900 MB/s。

1.3 DSP 产品简介

目前,DSP 芯片产品市场主要由以下公司把控:TI(Texas Instrumental)、ADI(Analog Devices)、Freescale(Motorola)和 Agere(Lucent)。由于 TI 公司优秀的大学计划、灵活广泛的产品类别,其产品占有一半以上的市场份额。本书主要介绍 TI 公司 2000 系列的产品。

1.3.1 TI 公司的 DSP 产品

TI 公司自 20 世纪 90 年代推出第二代 DSP 芯片 TMS32010,2000 年之后推出 TMS320C1x 系列后,又陆续推出了上百种 DSP 芯片。这些芯片尽管品种繁多,但依面向领域可以分成三大类:TMS320C2000 系列(实时控制)、TMS320C5000 系列(低功耗)和 TMS320C6000 系列(高性能)。

① TMS320C2000 系列,称为 DSP 控制器(DSC)。集成了 FLASH(或 ROM)存储器、高速 A/D 转换器、完备的通信接口及数字电动机控制的外围电路,适用于三相

试读结束,需要全本PDF请购买 www.ertongbook.com