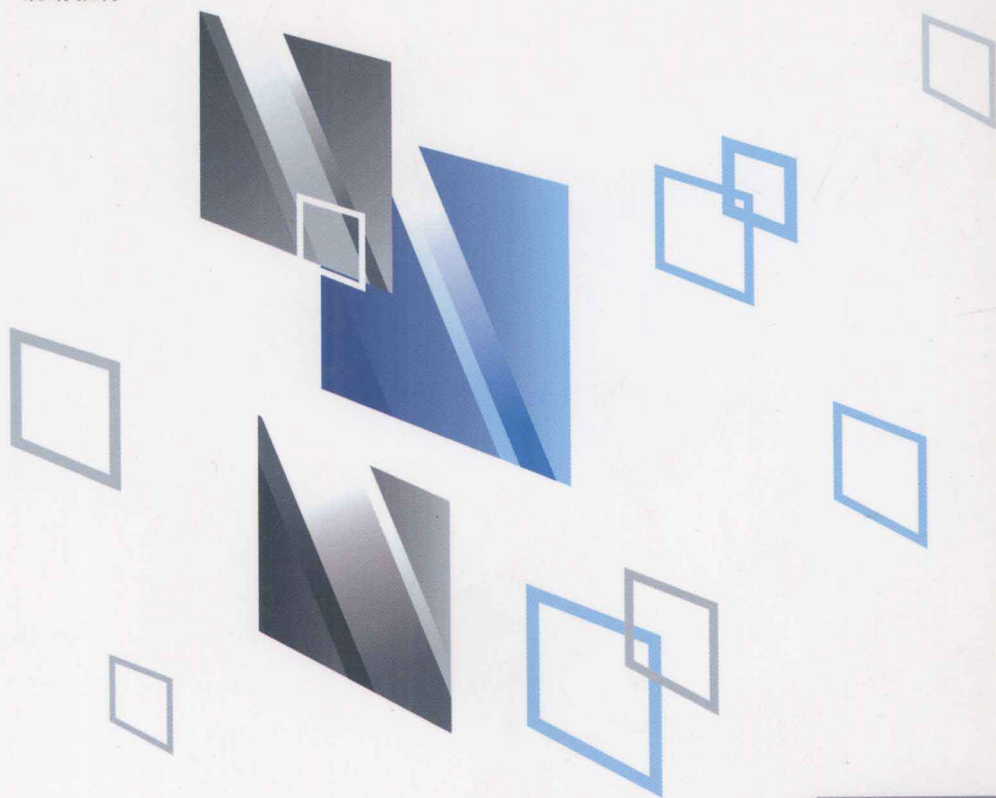




普通高校“十二五”规划教材



苏奎峰 吕强 常天庆 邓志东 编著

TMS320X281x DSP

原理及C程序开发

(第2版)



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS





普通高校“十二五”规划教材

TMS320X281x DSP 原理及 C 程序开发

(第 2 版)

苏奎峰 吕 强 常天庆 邓志东 编著

北京航空航天大学出版社

内 容 简 介

本书是在《TMS320X281x DSP 原理及 C 程序开发》的基础上,结合作者近年来在本领域的教学科研经验以及读者真诚的反馈意见修订而成。本书仍然以 TMS320F2812 数字信号处理器为主线,从 DSP 的基本开发方法入手,介绍基于 DSP 的系统软硬件开发方法。详细介绍处理器外设资源的使用、C 语言编程开发、浮点算法开发、程序固化等内容。此外还根据 DSP 的特点介绍基于定点处理器实现浮点算法的方法。在介绍功能的同时,列举了相应的应用实例,给出了硬件原理和 C 语言程序清单,并标有详细的程序说明,为用户快速掌握处理器各功能单元的使用提供了方便。附光盘 1 张,内含 C 语言程序代码。

本书可以作为大学本科和研究生的“数字信号处理器原理与应用”相关课程的教材,也可以作为数字信号处理器应用开发人员的参考书。

图书在版编目(CIP)数据

TMS320X281x DSP 原理及其 C 程序开发 / 苏奎峰等编
著. —2 版. —北京:北京航空航天大学出版社,
2011.9

ISBN 978-7-5124-0586-8

I. ①T… II. ①苏… III. ①数字信号处理②数字信
号—微处理器③C 语言—程序设计 IV. ①
TN911.72②TP332③TP312

中国版本图书馆 CIP 数据核字(2011)第 177569 号

版权所有,侵权必究。

TMS320X281x DSP 原理及 C 程序开发(第 2 版)

苏奎峰 吕 强 常天庆 邓志东 编著

责任编辑 张 楠 王 松

*

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@gmail.com 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

*

开本:787×1 092 1/16 印张:29 字数:742 千字

2011 年 9 月第 2 版 2011 年 9 月第 1 次印刷 印数:5 000 册

ISBN 978-7-5124-0586-8 定价:59.00 元(含光盘 1 张)

第2版前言

TMS320C2000 系列数字信号处理器是 32 位高性能微控制器,集微控制器和高性能 DSP 的特点于一身,具有强大的控制和信号处理能力,能够实现复杂的控制算法。TMS320C2000 系列 DSP 片上整合了 Flash 存储器、快速的 A/D 转换器、增强的 CAN 模块、事件管理器、正交编码电路接口、多通道缓冲串口等外设。此种整合使用户能够以很便宜的价格开发高性能数字控制系统,为此 TI 公司推出了 Piccolo、Delfino 浮点和 28x 定点三个系列处理器,从而满足不同应用系统的需求。

32 位的 TMS320X28xxx 系列 DSP 整合了 DSP 和微控制器的最佳特性,能够在一个周期内完成 32x32 位的乘法累加运算,或两个 16x16 位乘法累加运算,能够完成 64 位的数据处理,从而使该处理器能够实现更高精度的处理任务。为了提高浮点处理能力,快速的中断响应能够使 28xxx 保护关键的寄存器以及快速(更小的中断延时)的响应外部异步事件。28xxx 有 8 级带有流水线存储器访问流水线保护机制,流水线使得 28xx 高速运行时不需要大容量的快速存储器。专门的分支跳转(branch-look-ahead)硬件减少了条件指令执行的反应时间,条件存储操作更进一步提高了 28xx 的性能。

TMS320X28xxx 信号处理器集成了事件管理器、ePWM、eCAP、eQEP、模数转换模块、SPI 外设接口、SCI 通信接口、I²C 总线、eCAN 总线通信模块、看门狗、通用目的数字量 IO、PLL 时钟模块、多通道缓冲串口、外部中断接口、存储器及其接口、内部集成电路等多种外设单元,为功能复杂的控制系统设计提供了方便。

本书是在《TMS320X281x DSP 原理及 C 程序开发》的基础上,结合作者近年来在本领域的教学科研经验以及读者真诚的反馈意见修订而成。本书仍然以 TMS320F2812 数字信号处理器为主线,介绍 DSP 的基本开发方法、处理器外设资源的使用、C 语言编程开发、浮点算法开发、程序固化等内容。在介绍各功能单元的同时提供了相关的应用实例,给出了硬件原理图和 C 语言程序清单及程序分析。在此基础上根据不同系列处理器的特点和应用,将相关内容推广到 C2000 系列处理器,除了部分内容根据整个系列处理器特点进行扩充外,新增加了在事件管理器一章新增加了与之相对应的 ePWM、eCAP、eQEP 原理和应用内容。并针对部分处理器具有 I²C 总线接口,增加了 I²C 总线接口的原理及其与具有实时时钟功能的铁电存储器的扩展应用。

本书力求为学习 DSP 并希望使用 C 语言或 C 和汇编语言混合编程的人士提供有益的参考,为能够熟练使用 TMS320X28xx 处理器提供帮助。书中提供的所有程序都在 F2812 评估板上经过验证,部分程序在实际项目中也得到了充分的检验。有关评估板或本书内容方面的问题欢迎与作者及时沟通,邮箱地址:sukf@sina.com。

参加本书编写工作的有陈圣俭、张雷、武萌、杨国振、汤霞清、朱斌、王可可、宁固等同志。他

们为本书提供了大量资料并进行了大量实验。本书的成书过程中还得到了装甲兵工程学院张蔚、邱晓波、单东升、王钦钊、陈玉强,清华大学郭晓梅等老师和姚文涛、杨博、黄振、周立谱、陈振华等同学的大力支持,他们参与了本书的编写和录入工作。本书在选题和出版过程中得到了北京航空航天大学出版社的大力支持,在此一并表示感谢。另外,还要感谢我的妻儿和父母,如果没有他们的关爱、鼓励和支持,此书难以完成。

本书先后得到国家自然科学基金“无人驾驶车辆人工认知关键技术与集成验证平台(90820305)”和国家自然科学基金“基于边界扫描的混合电路故障诊断(60871029)”等科研项目的共同资助。

限于编者水平,书中难免存在错误和不当之处,恳请读者批评指正。

苏奎峰

2011 年 6 月于装甲兵工程学院

目 录

第1章 绪 论

1.1 DSP 概述	2
1.1.1 DSP 的发展	2
1.1.2 DSP 结构和特点	3
1.1.3 DSP 的选型	8
1.1.4 TI 公司的 DSP	9
1.1.5 C2000 实时控制器平台	10
1.2 DSP 的典型应用	11
1.3 DSP 的发展	12
1.4 DSP 系统开发	13
1.4.1 系统的需求分析	13
1.4.2 系统的基本结构	14
1.4.3 系统开发	15

第2章 CCS 软件应用基础

2.1 CCS 介绍	19
2.2 Code Composer Studio 3.1 的安装与配置	19
2.2.1 Code Composer Studio 3.1 的安装	19
2.2.2 目标系统配置	20
2.2.3 启动 GEL 文件	22
2.2.4 主机开发环境设置	23
2.3 Step-by-Step 简单应用	24
2.3.1 CCS 常用工具	24
2.3.2 简单程序开发	25
2.4 代码创建	26
2.4.1 新建一个工程	26
2.4.2 工程配置	28
2.5 CCS3.1 基本应用	28
2.5.1 编辑源程序	29
2.5.2 查看和编辑代码	30
2.5.3 查找替换文字	30
2.5.4 书签的使用	31

2.5.5	全速运行(Running)/单步运行(Step Run)	31
2.5.6	断点设置	33
2.5.7	探针的使用	35
2.5.8	观察窗口	37
2.6	分析和调整	38
2.6.1	应用代码分析	39
2.6.2	应用代码优化	40
第 3 章 C/C++ 程序编写基础		
3.1	C/C++ 编辑器概述	41
3.1.1	C/C++ 语言的主要特征	41
3.1.2	输出文件	41
3.1.3	编译器接口	42
3.1.4	编译器操作	42
3.1.5	编译器工具	44
3.2	TMS320X28xx 的 C/C++ 编程	44
3.2.1	概 述	44
3.2.2	传统的宏定义方法	45
3.2.3	位定义和寄存器文件结构方法	46
3.2.4	位区和寄存器文件结构体的优点	53
3.2.5	使用位区的代码大小及运行效率	54
3.3	C/C++ 程序结构及实例	57
3.3.1	Include 文件	57
3.3.2	链接文件	58
3.3.3	程序流程	62
3.4	C/C++ 语言与汇编混合编程	63
3.5	TMS320X28xx 定点处理器算法实现	70
3.5.1	定点与浮点处理器比较	70
3.5.2	采用 Iqmath 库函数实现定点处理器的运算	71
第 4 章 TMS320X28xx 系列 DSP 综述		
4.1	TMS320X28xx 系列 DSP 内核特点	86
4.1.1	C28xx 系列定点处理器特点	86
4.1.2	C28x 浮点处理器	89
4.2	TMS320x28xxx 系列处理器比较	91
4.2.1	工作频率和供电	91
4.2.2	存储器	91
4.2.3	外 设	96
4.3	TMS320X28xx 处理器外设功能介绍	97
4.3.1	事件管理器(281x 处理器)	97
4.3.2	ePWM、eCAP、eQEP(F2808、F2806、F2801 处理器)	97

4.3.3 A/D 转换模块	100
4.3.4 SPI 外设接口	100
4.3.5 SCI 通信接口	100
4.3.6 CAN 总线通信模块	101
4.3.7 看门狗	101
4.3.8 通用目的数字量 I/O	101
4.3.9 PLL 时钟模块	101
4.3.10 多通道缓冲串口	101
4.3.11 外部中断接口	102
4.3.12 存储器及其接口	102
4.3.13 内部集成电路(T ² C)	102
4.4 TMS320x281x 和 TMS320x2833x 的区别	102
4.4.1 概 述	102
4.4.2 中央处理单元(CPU)	103
4.4.3 存储单元	103
4.4.4 时钟和系统控制	106
4.4.5 通用目的 I/O(GPIO)	108
4.4.6 2833x 系列处理器新增外设	108
4.4.7 2833x 系列处理器改进外设	112
4.4.8 2833x 系列处理器未改动外设	114
4.4.9 中 断	114
4.5 TMS320X28xx 的应用领域	115
4.6 TMS320F2812 硬件平台	116
4.6.1 TMS320F2812 硬件描述	121
4.6.2 电源接口	121
4.6.3 复位电路	121
4.6.4 TMS320F2812 存储器接口	123
4.6.5 晶振选择	124
4.6.6 扩展总线	125
4.6.7 JTAG 接口	125
4.6.8 板上串行通信接口	125
4.6.9 CAN 总线接口	126
4.6.10 AD 变换单元	126
4.6.11 DAC 扩展	126
第 5 章 双供电 DSP 电源设计	
5.1 总线冲突	128
5.2 内核和 I/O 供电次序控制策略	129
5.2.1 3.3 V 单电源上电次序控制	129
5.2.2 输入电压大于 3.3 V 的上电次序控制	134

5.3 TMS320F28xx 电源设计	136
第 6 章 TMS320F2812 的时钟及看门狗	
6.1 时钟单元	137
6.1.1 时钟单元基本结构	137
6.1.2 锁相环电路	137
6.1.3 时钟单元寄存器	141
6.2 看门狗	144
6.2.1 看门狗基本结构	145
6.2.2 看门狗基本操作	145
6.2.3 看门狗寄存器	147
6.2.4 看门狗应用	149
第 7 章 可编程数字量通用 I/O	
7.1 数字接口的结构和实现方法	150
7.1.1 基本结构	150
7.1.2 实现方法	151
7.2 DSP 数字量 I/O 功能概述	152
7.3 端口配置	152
7.4 数字量 I/O 寄存器及其应用	155
7.4.1 I/O 复用寄存器及其应用	155
7.4.2 I/O 数据寄存器及其应用	161
7.5 数字量 I/O 应用举例	163
第 8 章 中断系统及其应用	
8.1 C28x 处理器中断概述	164
8.2 PIE 中断扩展	165
8.2.1 外设级中断	166
8.2.2 PIE 级中断	166
8.2.3 CPU 级中断	167
8.3 中断向量	167
8.3.1 中断向量的分配	167
8.3.2 中断向量的映射方式	168
8.4 中断源	176
8.4.1 复用中断处理过程	176
8.4.2 使能和禁止外设复用中断过程	177
8.4.3 从外设到 CPU 的复用中断请求流程	178
8.5 可屏蔽中断处理	179
8.5.1 中断标志设置(产生中断)	179
8.5.2 中断使能(单独使能中断)	179
8.5.3 全局使能(全局使能中断)	180
8.6 定时器中断应用举例	180

8.6.1 定时器基本操作	180
8.6.2 定时器寄存器	181
8.7 定时器中断应用举例	184
第9章 事件管理器及其应用	
9.1 事件管理器概述	185
9.2 通用定时器	186
9.2.1 通用定时器计数模式	187
9.2.2 定时器的比较操作	189
9.2.3 通用定时器寄存器	196
9.3 比较单元及 PWM 输出	201
9.3.1 比较单元功能介绍	201
9.3.2 PWM 信号	202
9.3.3 与比较器相关的 PWM 电路	203
9.3.4 PWM 输出逻辑及死区控制	204
9.3.5 PWM 信号的产生	206
9.3.6 比较单元寄存器	211
9.4 捕获单元	217
9.4.1 捕获单元的应用	217
9.4.2 捕获单元的结构	218
9.4.3 捕获单元的操作	218
9.4.4 捕获单元相关寄存器	220
9.5 正交编码脉冲单元	222
9.5.1 光电编码器原理	222
9.5.2 正交编码脉冲单元结构及其接口	223
9.5.3 QEP 电路时钟	224
9.5.4 QEP 的解码	224
9.5.5 QEP 电路的寄存器设置	225
9.5.6 QEP 电路应用	225
9.6 事件管理器中断	227
9.6.1 中断产生及中断矢量	229
9.6.2 定时器的中断	229
9.6.3 捕获中断	230
9.6.4 中断寄存器	230
9.7 事件管理器应用举例	234
9.8 增强型外设	236
9.8.1 ePWM 功能	236
9.8.2 增强捕捉单元	240
9.8.2 增强正交编码脉冲模块(eQEP)	248

第 10 章 SPI 接口及其应用

10.1 SPI 模块功能概述	256
10.2 SPI 的数据传输	258
10.2.1 主控制器模式	259
10.2.2 从设备模式	260
10.2.3 FIFO 操作	260
10.3 SPI 寄存器	261
10.3.1 SPI 配置控制寄存器(SPICCR)	261
10.3.2 SPI 操作控制寄存器(SPICTL)	263
10.3.3 SPI 状态寄存器(SPISTS)	265
10.3.4 SPI 波特率设置寄存器(SPIBRR)	266
10.3.5 SPI 仿真缓冲寄存器(SPIRXEMU)	267
10.3.6 SPI 串行接收缓冲寄存器(SPIRXBUF)	267
10.3.7 SPI 串行发送缓冲寄存器(SPITXBUF)	268
10.3.8 SPI 串行数据寄存器(SPIDAT)	268
10.3.9 SPIFFTX 寄存器	269
10.3.10 SPIFFRX 寄存器	270
10.3.11 SPIFFCT 寄存器	271
10.3.12 SPI 优先级控制寄存器(SPIPRI)	272
10.4 应用实例	273

第 11 章 I²C 总线接口及其应用

11.1 TMS320c28xxx 处理器 I ² C 总线	276
11.1.1 I ² C 主要特点	276
11.1.2 功能概述	276
11.1.3 时钟产生	277
11.2 I ² C 总线操作	278
11.2.1 输入和输出电平	278
11.2.2 数据状态要求	278
11.2.3 操作模式	278
11.2.4 I ² C 模块 START 和停止条件	279
11.2.5 串行数据格式	280
11.2.6 不响应信号(NACK)产生	282
11.3 I ² C 总线应用举例	282

第 12 章 eCAN 总线及其应用

12.1 CAN 总线概述	295
12.1.1 CAN 总线特点	295
12.1.2 CAN 总线数据格式	295
12.1.3 CAN 总线的协议	297
12.2 C28x 的 eCAN 模块介绍	301

12.2.1 eCAN 总线模块概述	301
12.2.2 eCAN 总线模块特点	302
12.3 eCAN 总线模块的使用	305
12.3.1 eCAN 模块初始化	305
12.3.2 消息发送	323
12.3.3 消息接收	327
12.3.4 过载情况的处理	333
12.3.5 远程帧邮箱的处理	334
12.3.6 CAN 模块中断及其应用	336
12.3.7 eCAN 模块定时器管理	345
12.3.8 CAN 模块的掉电模式	348
12.4 CAN 总线应用举例	349
第 13 章 SCI 接口应用	
13.1 SCI 接口特点	351
13.2 SCI 数据格式	353
13.3 SCI 增强功能	356
13.3.1 SCI 的 16 级 FIFO 缓冲	356
13.3.2 SCI 自动波特率检测	357
13.3.3 多处理器通信	358
13.4 SCI 接口应用	361
13.4.1 硬件设计	361
13.4.2 SCI 寄存器	362
13.4.3 SCI 初始化	363
13.4.4 SCI 发送数据	369
13.5 接收发送数据	382
第 14 章 A/D 转换单元	
14.1 A/D 转换单元概述	390
14.2 排序器操作	392
14.2.1 排序器操作方式	393
14.2.2 排序器的启动/停止模式	406
14.2.3 输入触发源	407
14.2.4 排序转换的中断操作	407
14.3 ADC 的时钟控制	409
14.4 ADC 参考电压	410
14.5 ADC 单元寄存器	411
14.5.1 ADC 模块控制寄存器 1	411
14.5.2 ADC 模块控制寄存器 2	413
14.5.3 ADC 模块控制寄存器 3	416
14.5.4 最大转换通道寄存器(MAXCONV)	418

14.5.5	自动排序状态寄存器 (AUTO_SEQ_SR)	419
14.5.6	ADC 状态和标志寄存器 (ADC_ST_FLG)	420
14.5.7	ADC 输入通道选择排序控制寄存器	422
14.5.8	ADC 转换结果缓冲寄存器 (RESULTn)	422
14.6	ADC 应用举例	423
第 15 章 存储器应用及 Boot 引导模式		
15.1	F28xx 映射空间概述	424
15.2	XINTF 接口扩展	426
15.2.1	XINTF 接口概述	426
15.2.2	XINTF 接口操作	427
15.2.3	XINTF 接口应用举例	428
15.3	Flash 及其应用	429
15.3.1	Flash 存储器特点	429
15.3.2	Flash 存储器寻址空间分配	429
15.3.3	C28x 启动顺序	430
15.3.4	Flash 初始化	431
15.3.5	Flash 编程	432
15.4	其他引导方式	438
15.4.1	处理器引导配置	438
15.4.2	C28x 中断向量表	440
15.4.3	BOOTROM 基本情况介绍	440
15.4.4	BootLoader 数据流	441
15.4.5	BootLoader 传输流程	443
15.4.6	初始引导汇编函数	443
15.4.7	SCI 引导装载	444
15.4.8	并行 GPIO 装载	444
15.4.9	SPI 引导模式	446

参考文献

第 1 章

绪 论

自上世纪 70 年代以来,可编程数字信号处理器(programmable digital signal processors, PDSP)在多媒体信号处理、通信、工业控制、雷达、天气预报等领域得到了广泛的应用,也正是数字信号处理技术的应用使得诸多领域取得了革命性的变化,数字信号处理技术本身也逐渐成为应用最广和最有潜力的技术之一。通常情况下可编程数字信号处理器在整个系统中扮演着双重角色:一方面作为可编程微处理器使用,完成通用微处理器实现的功能,另一方面,提供专门针对数字信号处理算法实现的特殊指令和结构,能够高效地完成数字信号处理算法。数字信号处理将信号以数字形式描述,然后针对数字化后的数据进行信号和信息处理。因此,在进行数字信号处理之前需要将信号从模拟域转换到数字域,这通常通过模数转换器实现。而数字信号处理的输出经常也要变换到模拟域,这是通过数模转换器实现的。数字信号处理的算法需要利用计算机或专用处理设备如数字信号处理器(DSP)和专用集成电路(ASIC)等。如图 1.1 所示为典型的数字信号处理系统结构框图。

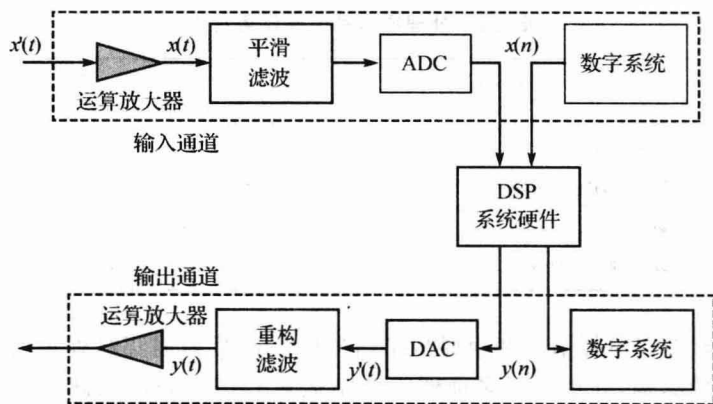


图 1.1 数字信号处理系统的结构框图

数字信号处理主要关注信号的数字描述,采用数字系统进行信号分析、调整、存储或信息特征提取等。绝大部分研究关注数字信号处理算法及其实时应用,而数字信号处理器则是根据通用数字信号处理算法的结构特点而设计的可编程处理器,为提高算法的执行效率、满足实时性要求,相对于通用处理器结构上做了适当的调整。近半世纪以来,由于集成电路技术的高速发展,使得用硬件来实现各种数字滤波和快速傅立叶变换成为可能,从而使 DSP 得到了极其迅速的发展和广泛的应用。在数字化的世界和互联网的时代,DSP 变得越来越重要,可以说是无处不在。

军事方面是 DSP 最早应用的领域。例如,侦察卫星收集到了由照相机或摄像机拍摄到的模拟图像资料后,必须对它们进行处理以便去除背景噪声,获得有用的信息,同时还要发回地面接收站。在整个过程中,以数字化形式处理信号具有显而易见的优势。DSP 可以使这些信号以加密的方式,高速传回地面。DSP 用于 GPS 制导系统中,可以高速分析定位卫星信号并将指令传给飞行器,大大提高了制导效率和精度。另外,军事通信、数据处理和传输都是 DSP 的应用范围。

在民用方面,数字移动蜂窝电话是 DSP 最重要的应用领域。DSP 强大的计算能力以及低廉的价格使得数字移动通信系统迅速普及,原来笨重、昂贵、功能单一的模拟机被小巧、廉价、功能丰富的手机取代。另外,由于采用 DSP 技术,蜂窝电话的升级换代更加方便,在统一的硬件平台上就可以通过软件设计进行,这也是新款手机不断推出的主要原因。

1.1 DSP 概述

在上世纪 60 年代,科学家曾经预测人工智能方法将会使人与计算机或其他设备之间的交互产生革命性的变化,他们相信到上世纪末可以实现机器人提供家庭服务,车辆自动驾驶、人机之间的语音交互等。然而这些诱人的愿景并没有完全实现,任务本身要远比想象的复杂得多,并不能单纯依赖计算机实现。但是在过去五十多年中,计算机本身在数据操作(如文字处理、数据库管理等)和数学运算(科学、工程计算和数字信号处理等)两个方面得到了广泛的应用。几乎所有微处理器都能完成上述两方面的工作,但很难做到两个方面工作都能够完成得很好。由于主要处理方法上的差异,如表 1.1 所列,处理器在设计上必须在指令集、中断处理、系统结构上进行折衷,而且受诸多市场因素的影响,如开发生产成本、竞争对象和市场定位、产品生存期等。综合上述多种因素,传统的微处理器,如 Pentium 系列主要专注于数据操作,而 DSP 则更擅长数学运算。

表 1.1 处理器两种主要任务对比

	数据操作	数学运算
典型应用	文字处理 数据库管理 操作系统任务调度等	数字信号处理 运动控制 科学、工程计算和仿真等
主要操作	数据移动($A \rightarrow B$) 数据判定($\text{if}(A=B) \text{ then} \dots$)	加法($A+B=C$) 乘法($A \times B=D$)

数字信号处理器是在模拟信号转换成数字信号以后进行高速实时处理的专用处理器,由于 DSP 采用改进的哈佛结构,并集成了多种便于数字运算和信号处理的硬件,其数字信号处理速度比普通的 CPU 快得多。在当今的数字化时代背景下,DSP 已成为通信、计算机、消费类电子产品以及控制等领域的基础器件。

1.1.1 DSP 的发展

DSP 的发展历程大致分为 3 个阶段:20 世纪 70 年代理论先行,80 年代产品普及,90 年代突飞猛进。在 DSP 出现之前,数字信号处理只能依靠 MPU(微处理器)来完成,但 MPU 较低的处理速度无法满足高速实时的要求。直到 20 世纪 70 年代,才有人提出了 DSP 的理论和算法基础。那时的 DSP 仅仅停留在教科书上,即便是研制出来的 DSP 系统也是由分立元件组成的,其

应用领域仅局限于军事及航空航天部门。

随着大规模集成电路技术的发展,1982年世界上诞生了首枚DSP芯片。这种DSP器件采用微米工艺NMOS技术制作,虽功耗和尺寸稍大,但运算速度却比MPU快了几十倍,在语音合成和编码解码器中得到了广泛应用。DSP芯片的问世标志着DSP应用系统由大型系统向小型化迈进了一大步。随着CMOS技术的进步与发展,第2代基于CMOS工艺的DSP芯片应运而生,其存储容量和运算速度成倍提高,成为语音处理、图像硬件处理技术的基础。20世纪80年代后期,第3代DSP芯片问世,运算速度进一步提高,其应用范围逐步扩大到通信和计算机领域。

20世纪90年代DSP发展最快,相继出现了第4代和第5代DSP器件。现在的DSP属于第5代产品,与第4代相比,系统集成度更高,将DSP内核及外围元件综合集成在单一芯片上。这种集成度极高的DSP芯片不仅在通信、计算机领域大显身手,而且逐渐渗透到人们的日常消费领域。经过20多年的发展,DSP产品的应用已扩大到人们的学习、工作和生活的各个方面,并逐渐成为电子产品更新换代的决定因素。目前,对DSP爆炸性需求的时代已经来临,前景十分可观。

1.1.2 DSP 结构和特点

1. 系统结构

图1.2给出了典型的基于微处理器的运动控制系统基本结构,该结构适用于PC等类型的计算机系统和绝大多数嵌入式系统,其中CPU是从程序存储器读取指令并逐条执行的时序逻辑控制机,顺序时钟控制CPU的执行时序。程序存储器主要用于程序代码固化或缓冲,数据存储器主要完成数据缓冲或变量固化,而外设则是嵌入式系统和外部对象实现信息交换的接口,通过总线将上述模块连接起来,实现彼此之间的互联和信息交换。

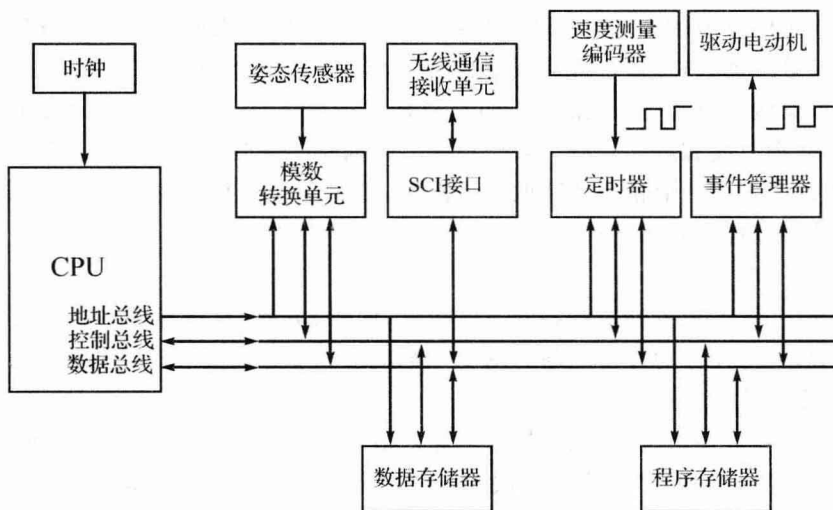


图 1.2 基于数字信号处理器的运动控制系统结构

系统结构确定了系统各单元之间的互联和数据交换方式,决定了CPU访问外设或存储器的方法。目前微处理器主要包括冯·诺依曼(Von Neumann)和哈佛(Harvard)两种结构,冯·诺依曼结构是目前微处理器市场应用最广泛的结构,其采用单一总线系统,在一条总线上实现CPU与外设之间的指令和数据交换,如图1.2所示。哈佛结构和冯·诺依曼结构相比,采用两条独立的总线系统分别传输指令和数据,如图1.3所示。基于哈佛结构的微处理器能够同时访

问指令和数据,从而提高了处理器的速度,并且独立的总线结构可以避免将数据写到程序存储空间,防止在系统运行过程中毁坏程序。然而由于哈佛结构必须采用独立的程序和数据存储空间,而且有些处理器两种存储空间不能互换,因此缺少灵活性。对于嵌入式系统而言,由于功能单一,且通常仅运行有限的应用程序,哈佛结构的程序和数据独立的存储空间影响不大。而对于计算机系统,由于需要运行多种应用程序,而且一般需要的程序和数据空间都比较大,冯·诺依曼结构可以实现程序和数据空间之间的灵活转换,因此更适合此类应用。近年来随着技术的发展,部分处理器采用程序、数据和输入/输出外设独立的增强哈佛结构,从而极大地提高了 CPU 的数据的吞吐能力。

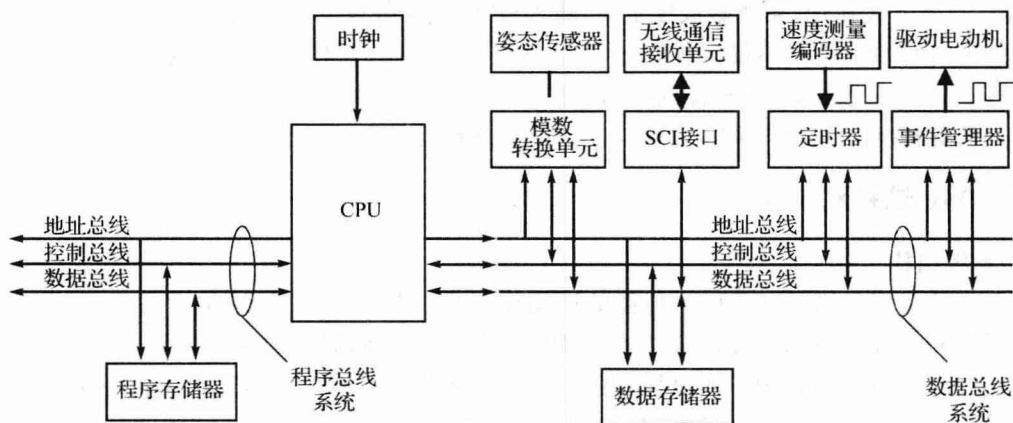


图 1.3 基于哈佛结构的系统结构

虽然应用于不同领域的 DSP 有不同的型号,但其内部结构大同小异,一般都具有哈佛结构的特征。它包括处理器内核、指令缓冲器、数据存储器、程序存储器、I/O 接口控制器、程序地址总线和程序数据总线等单元,其中最核心的是处理器内核。DSP 采用改进的哈佛总线结构,内部有两条总线:数据总线和程序总线。采用程序与数据空间分开结构,分别有各自的地址总线和数据总线,可以同时完成获取指令和读取数据操作,目前运行速度已经达到 1G 次定点运算/秒;概括起来数字信号处理器有如下特点:

2. 流水线

采用流水操作,每条指令的执行划分为取指令、译码、取数、执行等若干步骤,由片内多个功能单元分别完成,支持任务的并行处理。图 1.4 给出了典型 8 级流水线处理器的指令执行操作。

3. 单周期乘法累加运算

在通用数字信号处理算法中,乘法累加运算是最常见的操作,如表 1.2 给出的典型信号处理算法。在通用处理器,如 PC 机上实现一次乘法累加运算通常需要多个周期完成。而在 DSP 处理器上则可以在一个指令周期内实现一次或多次乘法累加(MAC)运算。

表 1.2 典型数字信号处理算法

算法名称	算法方程
FIR 滤波器	$y(n) = \sum_{k=0}^M a_k x(n-k)$
IIR 滤波器	$y(n) = \sum_{k=0}^M a_k x(n-k) + \sum_{k=1}^N b_k y(n-k)$