



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 (第三版)

——微控制器的软件和硬件

宁改娣 张虹 著

新形态
教材



科学出版社

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

(第三版)

——微控制器的软件和硬件

宁改娣 张虹 著



科学出版社

北京

内 容 简 介

本书首先脱离具体型号介绍微处理器硬件和软件的共性概念,授人以鱼不如授人以渔;然后以 8051、TMS320F28335、MSP430、MSP432 举例展开共性概念,并结合数字化出版技术,设计了大量二维码辅助实验教学的资源。实验室可以配置任何型号的微控制器,控制器开发的软硬件具体内容可以采用翻转课堂教学模式,选拔部分学生在课外根据课堂介绍的方法查找对应器件手册,熟悉所用硬件平台并进行实验,然后由学生在课堂上进行讲解、演示和讨论。

微控制器类课程的学习目的是“用”,希望本书能够给学生一套用好微控制器的通用方法,并通过某一微处理器的使用,训练学生的软硬件设计能力及调试基本功。

本书可以作为“数字电子技术与微处理器基础”“单片机原理”“DSP 技术及其应用”等课程的教材,适合高等学校电类各专业、机械制造及其自动化等专业的学生使用,也可供数字系统设计工程师和技术人员参考。

图书在版编目(CIP)数据

DSP 控制器原理及应用:微控制器的软件和硬件/宁改娣,张虹著.
—3 版. —北京:科学出版社,2018.3
普通高等教育“十一五”国家级规划教材
ISBN 978-7-03-056817-5
I. ①D… II. ①宁… ②张… III. ①数字信号处理-高等学校-教材 IV. ①TN911.72
中国版本图书馆 CIP 数据核字(2018)第 048350 号

责任编辑:余江 张丽花 / 责任校对:郭瑞芝
责任印制:霍兵 / 封面设计:迷底书装

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>

保定市中华美凯印刷有限公司印刷

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:8 3/4

2018 年 3 月第 三 版 字数:213 000

2018 年 3 月第 13 次印刷

定价:36.00 元

(如有印装质量问题,我社负责调换)

前 言

“DSP 技术及其应用”课程是 20 世纪末各个高校相继开设的介绍数字信号处理器件和应用的一门课程。本书第一版以 TMS320C24x 为模型介绍微控制器结构、指令系统及应用等,于 2002 年 11 月出版。第二版总结了微处理器的结构框架并以 TMS320C28x 为模型介绍微控制器及应用,于 2009 年 3 月出版,并列入普通高等教育“十一五”国家级规划教材。近几年微控制器更新换代的速度很快,导致教师无法以传统教学模式的某一型号微控制器为模型进行教学和出版教材。作者结合其他微处理器课程教学以及多年的微控制器课程教学改革试点经验,以“用”为课程教学目的,总结了微控制器软件和硬件的共性概念,出版第三版,期望本书能够成为微控制器类课程的通用教材并具有相对长的生命力。

自 20 世纪 70 年代起,每出现一种新型处理器,各个高校为了缩短学生应用新处理器的时间,一般会很快开设一门新课程。但是开课都喜新并不厌旧,从而使教学计划中存在一系列的处理器课程。作者也是自 20 世纪 80 年代后期,一路追随着摩尔定律的发展开设和更新了几门相关课程,并长期从事这些课程的教学和实验指导工作,深刻体会到从事微控制器课程教学的艰辛。教师需要不断跟进技术发展,更新教学内容、实验内容和教材。图 1 所示是西安交通大学电气工程学院 2010 年培养方案中与微处理器相关 4 门课程的学时数(理论+实验)、开课学期、选修或必修、学分等情况,例如,“单片计算机原理及应用”课程,上课学时 32,实验学时 4,第 6 学期上课,为选修课程,2 学分。课程对应的主要内容见图左边,由图 1 可见,“单片计算机原理及应用”与“DSP 技术及其应用”课程的很多内容是类似的,但在“DSP 技术及其应用”课程的教学,发现学生并没有很好地用到所学的微机原理和

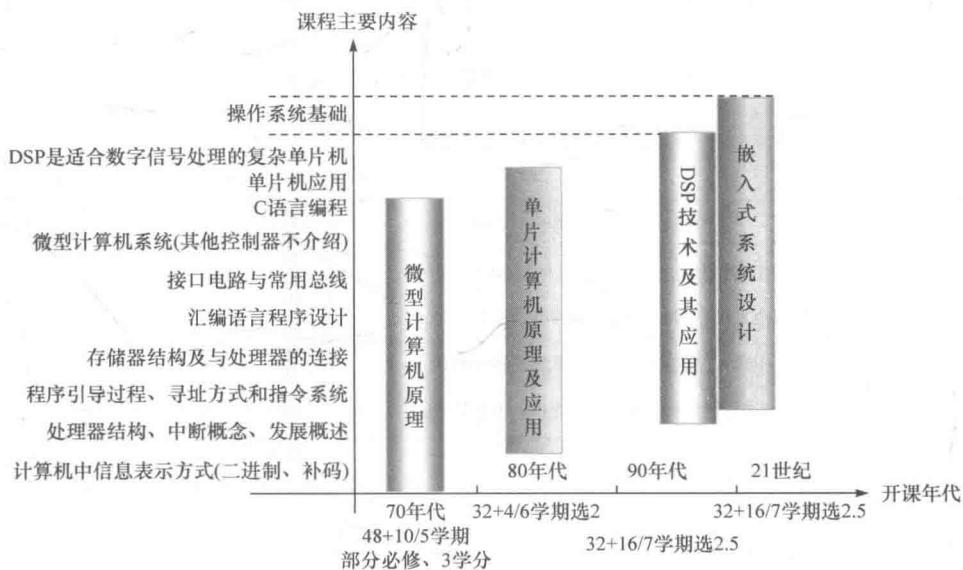


图 1 电气工程学院 2010 年培养方案中与微处理器相关课程(不包括集中实践课程)

单片机知识。作者回顾总结这些课程,发现确实存在一系列的问题,例如,占学时和学分多、低层次重复、课程衔接不当、应用训练少、师资及实验资源浪费、普遍针对某芯片、规律性基本概念几乎没有,导致学习了一系列微处理器课程后,多数学生不清楚哪些内容是微处理器软硬件设计需要掌握的基本功。而且,由于学时有限,所有微处理器课程都改为选修或部分必修课程,使得个别电类专业的学生可能一门处理器课程都没有学习过,后续学习以处理器为核心的集中实践必修课程——“电子系统设计与实践”时,部分学生就没有任何微处理器知识。



针对这种情况,我们在学校研究生公共课——“DSP 技术及其应用”中进行了教学改革试点,提炼介绍微处理器软硬件的共性概念。微处理器厂家和型号繁多,如何提炼微处理器软件和硬件开发过程中不同厂家、不同系列和不同型号共性的基本概念以及开发方法等,需要相当大的工作量。我们结合以往的教学经验进行了多年的教学试点,取得了好的教学效果以及学生的认可。

课程的主要课堂教学内容如图 2 所示,由教师在课堂中介绍微处理器普遍性或者软硬件共性的概念,然后学生根据这些共性概念,自由选择本书第 4~7 章或实验室提供的任一型号的微控制器实验平台,在课外查阅具体使用的微控制器手册、熟悉实验平台、设计实验、软硬件运行调试、课堂讲解和演示甚至参与竞赛活动等,采用这种翻转课堂教学模式,将课堂变成老师与学生之间和学生与学生之间互动的场所,经过多年的教学尝试,并不断听取学生的反馈意见和建议,不断地总结与改进,激发了学生动手实验的兴趣,也取得了好的教学效果。对于不断推出的新型处理器以及处理器最新的开发方法,可以在课程最后以讲座形式介绍给学生,让学生能够及时了解新事物并第一时间应用新技术。这不仅有助于解决高校培养计划学时学分越来越少的情况,还可以有效地克服上述种种问题。更重要的是,可以使学生更好地掌握处理器的最通用的开发方法,面对后续不断涌现的新型处理器就清楚如何开发其软硬件。

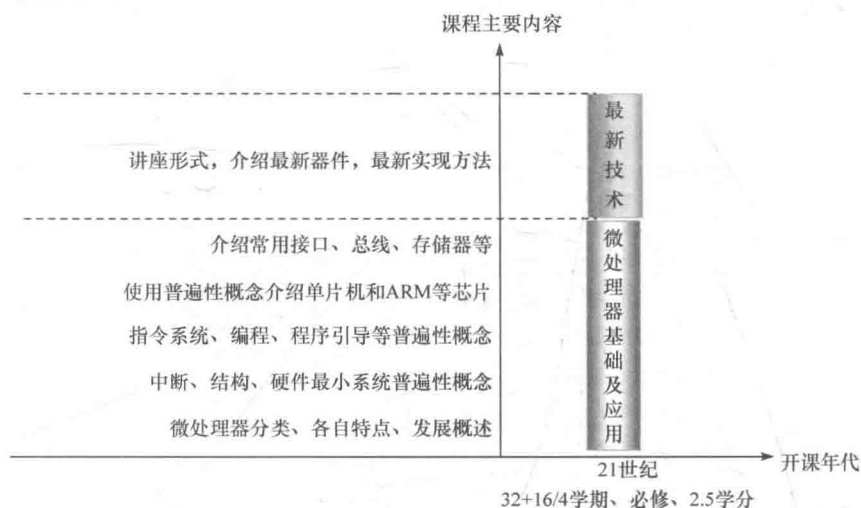


图 2 微处理器类课程教学改革方案

微处理器类课程的基础是“数字电子技术”,两门课程内容有紧密联系。作者分别在 2001

年和 2015 年发表了关于“数字电子技术”和“微处理器类”课程整合的教学改革论文(见本书参考文献),得到同行和出版社的认可,这也是课程发展的趋势。近几年已有几所高校合并了课程,西安交通大学电气工程学院的 2015 年培养方案也推动了课程整合的进程,开设了一门“数字电子技术与微处理器基础”必修新课程。该课程可以从微处理器硬件系统的角度出发介绍数字逻辑的各部分内容,有效地解决之前的“数字电子技术”课程内容零散以及与微处理器类课程内容和课程衔接之间存在的一些问题,减少了总学时数。新课程第 4 学期上课,理论课 72 学时,实验 24 学时,配合后续 80 学时的“电子系统设计与实践”必修课程,可以使学生更好地掌握微控制器的开发应用。

微控制器的教学目的是“用”好 MCU。因此,近 10 年来,在“DSP 技术及其应用”课程中,我们对考核方式也进行了大胆改革,逐步取消了笔试,课程成绩主要由实验的各环节确定,课程一开始就告知每个学生课程成绩的评定细则。实验采取口袋实验开放形式,实验内容由学生自己设计。成绩主要由学生设计的实验内容难易程度、实验深度或综合性、实验视频、实验报告的规范性和论述清晰性、实验中遇到问题的多少及解决方法、翻转课堂 PPT 的准备和讲解效果、课堂表现、经验分享等内容确定。发现实验内容或实验报告有照抄行为的,所有违规学生平分取得的百分制成绩,对实验设计和实验报告原创人员也不例外,经过多年试点,同届甚至与往届学生的实验内容基本没有发现复制行为。对于本科生,如果有课内规定实验内容,成绩中还必须考虑实验预习情况,最后一次课内实验也可以是学生课外设计实验的考试验收。这种考核方式虽然会花费老师很大精力和大量时间,也做不到像笔试那样细化分数,但这种考核方式的指挥棒,可以让学生真正做到重视实践环节,提高分析问题和解决问题的能力。

第三版教材就是在上述教学改革思路下,经过多年与课程组老师的不断讨论,与学生共同进行教学和实验试点、不断改进和修订而成的。第 1~3 章总结不同微控制器的 CPU 基本概念、微控制器的硬件框架、软件编程、集成开发环境等共性内容,强化微控制器的中断、堆栈、程序引导、硬件最小系统等普遍性概念。后续几章套用共性的概念介绍几个典型的微控制器。首先介绍最基础、最简单、应用最长久的 8051 单片机,对于学过 8051 的读者,可以总结回顾单片机开发对应的共性概念,提炼推广训练过的软硬件开发经验到其他微控制器的学习当中,对于没有任何一种微控制器基础的读者,选择结构简单的 8051 有利于初学者掌握处理器的开发方法。当然也可以利用共性概念,让学生学习如何使用比较复杂的适合数字信号处理的控制器 TMS320F28335、广泛应用的低功耗 MSP430 和基于 ARM 的微控制器 MSP432 等。教师也可以根据实验室拥有的实验平台、学时数、学生基础等具体情况,选择一种或多种微控制器在课程中进行介绍。

本书的重点在于微控制器共性概念的总结提炼,不求全面。对微控制器内部集成的大量片内外设,只在第 4 章介绍了 MCU 片内 I/O 结构特点、定时器的使用方法、串行通信概念和 UART 等,在第 6 章介绍了串行通信 SPI 和 I²C 的概念以及接口。其他接口特别是 TMS320F28335 内部的大量接口,本书正文中不再详细介绍。使用时也可以查阅器件手册,所有 MCU 片内外设接口的应用基本类似,开发经验可以逐步积累。为了便于使用者了解相关内容,教材以二维码形式给出了大量数据手册、用户手册、实验平台说明等资料(用手机微信“扫一扫”功能,扫描书中相应的二维码,即可在线查看)。

本书的核心思想和主要内容是宁改娣对三十多年微处理器类课程教学和实践的总结,其中张虹编写了1.1节、1.2节、2.3.5节和第3章3.1~3.4节内容的初稿,研究生孙培钦参与了3.5节的编写工作,其余部分由宁改娣编写,教材统稿和校对由宁改娣完成。本书中多数二维码资料是来自TI官方网站和“DSP技术及其应用”课程教学中学生的学习总结以及经验之谈。在此感谢研究生孙培钦、王骏逸、马达、叶严森、王毅(学号:3117058009)、张阳、王毅(学号:3117028010)、梁阳、孙官华、贾培鑫、李雨果、徐龙玉、吴祉谦、王涛、王毅(学号:3117306079)等,总结了程序流程图概念和基于F28335、MSP430、MSP432等不同实验平台的学习经验,以及提供的学习每个模块的讲解资料和工程文件。

本书的出版还要感谢德州仪器公司(TI公司)与教育部2016年产学合作协同育人项目的支持。感谢TI公司网站上提供的各种微控制器详细的器件手册、软硬件工具资料、应用工程文件等,提醒大家在使用这些资源时要关注TI的重要通知、警告和重要声明。另外,要特别感谢TI公司大学计划部经理潘亚涛以及崔萌、谢胜祥等TI大学计划部的工程师对我校相关课程教学工作的长期支持!感谢杭州艾研信息技术有限公司官方网站上提供的大量的硬件和软件资源!

感谢研究生院和电气工程学院领导支持本教材的出版!也要特别感谢课程组的金印彬老师以及其他老师为教学和实验工作所付出的努力!对长期专注于教学且默默地无私奉献的老师致以崇高的敬意!感谢参与第一版和第二版教材编写工作的作者!

由于作者对处理器共性的总结也是基于几个公司的常用几种处理器,可能存在局限性,书中难免有不足之处,希望使用本书的老师、学生及工程技术人员批评指正。

作者

2017年11月于西安交通大学

目 录

前言

第 1 章 微处理器基本概念	1
1.1 微处理器、微控制器及嵌入式处理器	1
1.2 CPU 结构	4
1.2.1 控制器	4
1.2.2 运算器	6
1.3 CPU 工作流程(程序引导过程)	7
1.4 CPU 对存储器及接口的编址方式	8
1.4.1 程序和数据存储器的地址配置	8
1.4.2 I/O 接口及编址方式	9
1.4.3 硬件接口的三要素	14
思考与习题	15
第 2 章 微控制器硬件框架性概念	16
2.1 MCU 内部结构框架及片内外设简介	16
2.2 MCU 总线概念	17
2.2.1 总线的定义和分类	17
2.2.2 微控制器总线结构	18
2.2.3 总线的基本结构	20
2.3 MCU 硬件最小系统	22
2.3.1 电源和复位电路	22
2.3.2 时钟电路	23
2.3.3 总线扩展接口	25
2.3.4 JTAG 接口	25
2.3.5 MCS-51 系列微控制器的最小系统	26
2.4 CPU 中断概念	27
2.4.1 CPU 与外部的数据通信方式	28
2.4.2 查询和中断	28
2.4.3 中断的响应过程	30
2.5 微控制器建议学习方法	31
思考与习题	33
第 3 章 软件系统和编程语言	34
3.1 软件系统简介	34
3.2 机器语言	34

3.3	汇编语言	35
3.3.1	RISC 与 CISC 指令集	36
3.3.2	汇编指令格式及转换	37
3.3.3	汇编伪指令	38
3.4	高级语言	39
3.5	程序流程图	40
3.5.1	程序流程图的符号和结构	41
3.5.2	画流程图的步骤	42
3.5.3	包含中断的 MCU 流程图画法	44
3.6	微控制器的集成开发环境	45
3.6.1	TI CCS 的特点与安装	45
3.6.2	基于 TI CCS 的软件开发流程	52
3.6.3	基于 Eclipse 的 CCS	58
	思考与习题	59
第 4 章	8051 微控制器及 MCU 常用接口简介	60
4.1	8051 结构框图及总线	60
4.2	8051 的 CPU 结构和寄存器介绍	61
4.3	8051 存储器结构	63
4.3.1	8051 存储空间配置及上电程序引导	63
4.3.2	片内 RAM 和特殊功能寄存器及复位初值	64
4.4	8051 汇编语言指令集	66
4.4.1	寻址方式	66
4.4.2	汇编指令	67
4.5	8051 中断系统及汇编编程举例	71
4.6	MCU 片内 I/O 结构	73
4.6.1	MCU 的 I/O 结构特点	74
4.6.2	8051 的 P0 端口	74
4.6.3	8051 的 P1、P2 和 P3 端口	76
4.7	MCU 片内定时/计数器	77
4.7.1	8051 定时/计数器控制寄存器	77
4.7.2	8051 定时/计数器控制工作方式	78
4.7.3	应用举例	80
4.8	MCU 片内串行通信接口	82
4.8.1	串行通信的基本概念	82
4.8.2	8051 单片机的串行通信接口	85
	思考与习题	90
第 5 章	TMS320F28335 微控制器	93
5.1	TMS320C2000 简介	93

5.2	TMS320F28335 的结构及主要特性	95
5.3	TMS320F28335 的硬件最小系统	99
5.4	TMS320F28335 存储器配置及上电程序引导	101
5.4.1	F28335 的存储器配置	101
5.4.2	F28335 的上电程序引导	101
5.5	F28335 中断系统	104
5.5.1	F28335 中断结构	104
5.5.2	中断响应和向量表	106
5.6	F28335 的片内外设及实验	108
	思考与习题	109
第 6 章	MSP430 微控制器	110
6.1	MSP430 的结构和特点概述	110
6.2	MSP430 的实验平台简介	115
6.2.1	MSP-EXP430G2 LaunchPad	116
6.2.2	LaunchPad G2 口袋实验平台	117
6.3	MSP430 时钟模块结构与实验	117
6.4	MSP430 片内外设模块以及实验	119
6.4.1	基础模块及实验	119
6.4.2	同步 I ² C 模块	120
6.4.3	同步 SPI 模块	121
	思考与习题	123
第 7 章	TI 基于 ARM 核的微控制器 MSP432	125
7.1	MSP432 的特点概述	125
7.2	MSP432 的实验平台简介及实验	126
7.3	MDK-ARM	127
	思考与习题	128
	参考文献	129

第 1 章 微处理器基本概念

微处理器是一种软件程序驱动与外围接口芯片协同工作完成特定任务的数字集成器件。在程序指令的控制下，微处理器能很方便地完成复杂的算术、逻辑运算，更新存储器内容，控制外围设备等工作。当系统性能需要改变时，只要修改相应的程序指令，不用像硬件系统设计那样重新设计和加工整个硬件电路。对于复杂电子系统的设计，基于微处理器的设计方案要明显优于利用中小规模集成电路的设计。随着微处理器性能的提高、功能的日益强大，基于微控制器和可编程逻辑器件的数字系统设计已成为目前电子系统设计的主流。

从广义上讲，微处理器(Micro Processor Unit, MPU 或者 Central Processing Unit, CPU)、微控制器(Micro Controller Unit, MCU)、数字信号处理器(Digital Signal Processor, DSP)、嵌入式处理器等都称为微处理器或嵌入式处理器(嵌入数字系统)，而且各集成器件制造厂家也在不断吸取别家所长，提高自己产品的竞争优势。因此，微处理器并没有一个严格的分类。但狭义地讲，CPU、MCU、DSP 和嵌入式处理器等器件的结构特点与概念有所不同，本书对其不同进行总结。

1.1 微处理器、微控制器及嵌入式处理器

1. 微处理器(CPU)

微处理器是一个功能强大的中央处理单元(CPU)，其内部主要包括算术运算单元和控制单元(ALU 和 CU)。这种芯片往往是个人计算机和高端工作站的核心 CPU。最常见的微处理器是 Motorola 的 68K 系列和 Intel 的 x86 系列。“微型计算机原理”课程中就是以 Intel 的 80x86CPU 为例介绍微处理器的结构、指令系统、接口及应用的。

微处理器片内普遍没有用户使用的存储器、定时器和常用接口。因此，微处理器在电路板上必须外扩存储器、总线接口及常用外设接口与器件，从而降低了系统的可靠性。例如，“微型计算机原理”学习的 Intel 8088/8086 CPU，使用时需要 244/245/373 构成总线，外加 Intel 8087 浮点运算协处理器、并行可编程接口芯片 8255A、计数/定时器 8253/8254、DMA 控制器 8237、中断控制器 8259A、串行通信接口 8250/8251 等芯片，构成早期的个人计算机 IBM XT/AT 的主板系统。微处理器的功耗普遍较大，如 Intel 的 CPU 多在 20~100W。

微处理器，自 20 世纪 70 年代问世以来，得到了迅速的发展。以字长和典型芯片作为标志，CPU 的发展主要经历了以下几个阶段。

1971~1972 年为第一阶段，主要是 4 位和 8 位 CPU。典型产品为 Intel 4004 和 Intel 8008。采用机器或简单的汇编语言，指令数目少，主要用于家电和简单的控制场合。

1973~1977年为第二阶段,主要是8位中高档微处理器时代。典型产品为Intel 8080和Intel 8085、Motorola的MC6800和Zilog的Z80。特点是指令系统较完善,具有典型的CPU体系结构。

1978~1984年为第三阶段,主要是16位微处理器时代。典型产品为Intel的8086/8088、80286, Motorola的M6800和Zilog的Z8000。特点是指令系统更加丰富,采用多种寻址方式、硬件乘法部件等。

1985~1992年为第四阶段,主要是32位微处理器时代。典型产品为Intel公司的80386/80486、Motorola的M68030/68040等。同期,其他一些微处理器生产厂商,如AMD等也推出了80386/80486系列的芯片。特点是具有32位数据线和32位地址线,每秒可完成600万条指令,为多任务的处理提供了可能。

1993~1994年为第五阶段,主要是Pentium系列微处理器时代。典型产品为Intel公司的Pentium系列芯片和与之兼容的AMD的K6系列微处理器芯片。特点是内部采用了流水线结构,并具有处理高速数据缓存的能力。

1995~2006年为第六阶段,主要是P6和NetBurst架构的微处理器时代。P6架构的典型产品为Intel公司的Pentium Pro、Pentium II、Pentium III等,内部采用3条超标量流水线结构,工作频率和总线频率显著提高。NetBurst架构的典型产品为Pentium 4。

2007~至今为第七阶段,采用Core架构的酷睿系列微处理器时代。产品代表有Core 2 Duo、Core 2 Quad、Core 2 Extreme等。Core 2系列为双核结构的微处理器。另外,2008年Intel公司还推出了64位四内核的微处理器Core i7。此外,Intel公司基于14nm制作工艺的全新Cherry Trail芯片也已经出货,该芯片主要应用于平板电脑产品。

2. 微控制器(MCU和DSP)

微控制器诞生于20世纪70年代后期,这类处理器片内除具有通用CPU所具有的ALU和CU,还集成有存储器(RAM/ROM)、计数器、定时器、各种通信接口、中断控制、总线、A/D和D/A转换器等适合实时控制的功能模块。因此,这类处理器称为单片机(Single Chip Computer)或微控制器。经过30多年的发展,其成本越来越低,而性能越来越强大,这使其应用已经无处不在,遍及各个领域。例如,电机控制、条码阅读器/扫描器、消费类电子、游戏设备、电话、HVAC、楼宇安全与门禁控制、工业控制与自动化、家电等领域。

学过“数字信号处理技术”的读者应该还记得,像FIR和IIR数字滤波、卷积、快速傅里叶变换等数字信号处理算法中,乘积和(Sum of Product, SOP)是最基本的单元。在处理器实时信号处理中,就要求完成SOP的速度尽可能地快。20世纪80年代中后期,出现了一种结构更加复杂的高性能的微控制器或单片机,其内部采用多总线结构(数据和程序有各自的总线)、指令执行使用多级流水线结构(多条指令同时运行在不同阶段)、片内集成有硬件乘法器、具有进行数字信号处理的特殊指令等。这种处理器可以更加快捷地完成SOP算法,因此得名数字信号处理器(DSP),DSP可以在一个指令周期完成乘法与加法。最常见的有TI的TMS320系列, Motorola的MC56和MC96系列, AD公司的ADSP21系列等。

某些专用微控制器设计用于实现特定功能，从而在各种电路中进行模块化应用，而不要求使用人员了解其内部结构。如音乐集成微控制器，它将音乐信号以数字的形式存于存储器中，由微控制器读出，转化为模拟音乐电信号。这种模块化应用极大地缩小了体积，简化了电路，降低了损坏和错误率，也便于更换。

微控制器与微处理器相比，最大的优点是将适合实时控制的一些接口和微处理器一起单片化，体积大大减小，从而使功耗和成本下降，可靠性提高。微控制器可单独完成现代工业控制所要求的智能化控制功能，这是微控制器最大的特征。微控制器由于其体积小、灵活性大、价格便宜、使用方便等优点，自问世以来，在工商、金融、科研、教育、国防、航空航天等领域都有着十分广泛的用途。可以这样说，微控制器现已渗透到人们日常生活的方方面面。

Intel 公司作为最早推出 CPU 的公司，同样也是最早推出微控制器的公司。继 1976 年推出 MCS-48 后，又于 1980 年推出了 MCS-51 系列微控制器，产品包括 8031、8051、8751、89C51 等。MCS-51 系列微控制器设置了经典的 8 位微控制器总线结构，具有 8 位数据总线、16 位地址总线、控制总线及具有多机通信功能的串行通信接口；体现了工控特性的位地址空间及位操作方式。另外，指令系统趋于丰富和完善，并且增加了许多突出控制功能的指令。MCS-51 系列微控制器为发展具有良好兼容性的新一代微控制器奠定了基础。

在 8051 技术实现开放后，Philips、Atmel、Dallas 和 Siemens 等公司纷纷推出了基于 8051 内核的微控制器，将许多测控系统中使用的电路技术、接口技术、多通道 A/D 转换部件、可靠性技术等应用到微控制器中，增强了其外围电路的功能，强化了智能控制的特征。另外，随着技术的不断进步，这些微控制器自身性能已得到大幅提升，例如，现在 Maxim/Dallas 公司提供的 DS89C430 系列微控制器，其单周期指令速度已经提高到了 8051 的 12 倍。

8051 微控制器基于复杂指令集(Complex Instruction Set Computer, CISC)架构，即采用一整套指令来完成各种操作。基于这种架构的微控制器除了 8051 外，还有 Motorola 提供的 68HC 系列微控制器等。

除此之外，还有一些基于精简指令集(Reduced Instruction Set Computer, RISC)架构的微控制器，包括 Microchip 的 PIC 系列 8 位微控制器、Maxim 公司推出的 MAXQ 系列 16 位微控制器、TI 公司的 MSP430 系列 16 位微控制器等。

20 世纪 90 年代后随着消费电子产品大发展，微控制器技术得到了巨大提高。出现了以 ARM 系列、Atmel 的 AVR32 为代表的 32 位微控制器，尤其是 ARM 系列微控制器在高端市场的使用，使其迅速进入了 32 位微控制器的主流。

微控制器可从不同方面进行分类：根据数据总线宽度可分为 8 位、16 位、32 位机等；根据存储器结构可分为哈佛(Harvard)结构和冯·诺依曼(Von Neumann)结构；根据内嵌程序存储器的类别可分为 OTP、掩模、EPROM/EEPROM 和闪存 Flash；根据指令结构又可分为 CISC 和 RISC 微控制器等。

3. 嵌入式处理器

从狭义上讲，嵌入式处理器是一种处理器的 IP 核(Intellectual Property Core)。开发公司

开发出处理器结构后向其他芯片厂商授权制造，芯片厂商可以根据自己的需要进行结构与功能的调整。嵌入式处理器的主要产品有 ARM(Advanced RISC Machines)公司的 ARM、Silicon Graphics 公司的 MIPS、IBM 和 Motorola 联合开发的 PowerPC、Intel 的 x86 和 i960 芯片、AMD 的 Am386EM、Hitachi 的 SH RISC 芯片等。

嵌入式处理器的主要设计者是 ARM 公司，靠转让设计许可，由合作伙伴公司来生产各具特色的芯片，是一个不生产芯片的芯片商。在全世界范围的合作伙伴超过 100 个，其中包括 TI、Xilinx、Samsung、Philips、Atmel、Motorola、Intel(典型芯片有 StrongARM 和 XScale)等许多著名的半导体公司。ARM 公司专注于设计，设计的处理器内核耗电少、成本低、功能强。采用 ARM 技术的微处理器遍及各类电子产品，在汽车电子、消费娱乐、成像、工业控制、网络、移动通信、手持计算、多媒体数字消费、存储安保和无线等领域无处不在。自成立至 2017 年初芯片出货量累计超过一千亿片。

狭义上的嵌入式系统是指，使用嵌入式微处理器构成独立系统，具有自己的操作系统并且具有某些特定功能的系统。

1.2 CPU 结构

本章以下内容主要介绍狭义的微处理器(CPU)的硬件结构和工作流程。

CPU 芯片中制作了执行各种功能的硬件逻辑电路，它可以读懂程序指令代码，并按照一定的顺序执行，完成人们给它的任务。CPU 是一切基于微处理器的电子设备的核心部件或“大脑”，负责对整个系统的各个部分进行统一的协调和控制。

CPU 主要包括运算器和控制器两大部分，控制器的主要作用是自动完成取指令和执行指令等任务。每个微处理器都有自己的指令系统，每条指令都有其确定的二进制微代码，控制器读取指令微代码通过指令译码就可以知道指令的作用，CPU 执行指令来协调并控制微处理器的各个部件。运算器的主要作用是算术运算和逻辑运算。CPU 的性能大致反映出微处理器的性能。

1.2.1 控制器

控制器(Control Unit, CU)为 CPU 的指挥和控制中心，结构如图 1.2.1 所示，控制器主要由指令寄存器(Instruction Register, IR)、指令译码器(Instruction Decoder, ID)、程序计数器(Program Counter, PC)、堆栈指针(Stack Pointer, SP)、控制和时序发生电路等组成。CPU 根据 PC 的内容，通过总线读取外部存储器中的指令，并暂存在 IR 中，ID 进行指令译码，并根据译码后产生的控制信号，指挥各部件间协同工作。

IR 主要用于存放即将执行的指令。CPU 在执行程序时，首先从存储器中取出要执行的一条指令代码存放在 IR 中。

ID 的作用是分析 IR 中的二进制程序代码，让 CPU 知道本条指令要执行的是什么操作。IR 和 ID 是 CPU 内部的专用寄存器，用户一般无法访问它，也无须关心它。

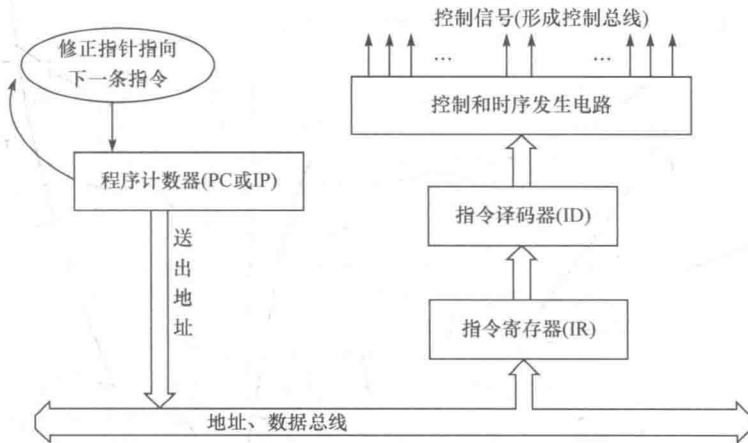


图 1.2.1 CPU 控制器结构框图

控制和时序发生电路根据指令译码分析的结果发出相应的控制命令，按照时间顺序发出各种控制信号，以保证指令能按照一定节拍顺序执行。控制和时序发生电路负责对整个系统进行控制，它还向 CPU 之外的其他各部件发出相应的控制信号，使 CPU 内、外各部件间协调工作。

CPU 内部的地址和数据总线用来连接 CPU 内部各功能部件，并在功能部件之间传送数据和控制信号。

软件设计者需要熟悉 CPU 控制器中的两个重要寄存器，一个是程序计数器(PC)，另一个是堆栈指针(SP)。有些 CPU 中将 PC 用指令指针(Instruction Pointer, IP)表示。

1. 程序计数器(PC)

程序计数器是专门用于保存程序指令存储地址的一个寄存器。CPU 每执行一条指令 PC 自动地增加一个量，这个量等于本条指令代码所占用的存储器单元数，以便使 PC 保持的总是将要执行的下一条指令的地址。由于程序在执行前已按照先后顺序存放在存储器中，因此，只要把程序第一条指令的地址，即程序入口地址装入 PC，程序就可以在 PC 的引导下逐条执行了。由于大多数指令都是按顺序来执行的，所以修改的过程通常只是简单地对 PC 加 1 或加 2(多数指令一般占用 1 个或 2 个存储单元)。如果 CPU 要执行子程序、跳转指令和后续要介绍的中断服务程序等，也都是通过修改 PC 的值，使程序转向对应的分支去执行程序。不同厂家甚至同一厂家不同系列的 CPU，其 PC 位数和初值可能不同，CPU 复位后，PC 值被设置为处理器器件手册中介绍的值。一般情况下，PC 复位后的值是指向存储器地址低端(例如，8051 复位后 16 位的 PC 初值为 0000H)或高端地址(例如，F2812 复位后 22 位的 PC 寄存器值为 3FFFC0H)。多数情况下，PC 是专为处理器提供的，用户一般无法通过指令访问它，但 MSP430 系列 MCU 可以通过指令寻址 PC，即用户可编程 PC。但学习任何一款微处理器时，必须清楚指令指针 PC 在 CPU 复位后的初值是什么。如果处理器芯片内部出厂时没有固化引导程序，用户的第一条指令代码就必须从 PC 初值所指的存储单元开始存放。

2. 堆栈指针(SP)

堆栈是一个特定的数据存储区域，它按“后进先出或先进后出”方式存储信息(类似生活中的存储桶)。堆栈指针 SP 是用来存放堆栈栈顶地址的一个寄存器。主要用于保存程序断点地址、主程序现场、重要数据等。当通过指令或者 CPU 的某些特定操作(执行中断服务或转向子程序)要将数据压入堆栈时，SP 会自动加 1 或减 1，不同 CPU 对 SP 的调整方式不同，SP 加 1 入栈为地址向上增长型，SP 减 1 入栈为地址向下增长型，栈中原存信息不变，只改变栈顶位置保存数据。当数据从堆栈弹出时，弹出的是栈顶位置的数据，弹出后 SP 自动减 1 或加 1。也就是说，数据在进行入栈和出栈操作时，SP 总是指向栈顶。堆栈指针 SP 和 PC 一样，CPU 上电或复位时被初始化，不同处理器 SP 的复位初值不同。多数 CPU 的 SP 值用户可以通过指令修改，这种 CPU 的堆栈一般设定在数据存储器的空间的某一区域，作为堆栈使用的数据存储空间，用户不能用于存储数据。也有个别处理器的堆栈是硬件堆栈，且堆栈存储容量和位置固定，不允许用户对堆栈指针进行操作，当然也没有入栈和出栈的指令，仅在 CPU 执行中断服务或转向子程序等操作时完成与上述堆栈操作类似的方式入栈或出栈，保存程序断点处的 PC 值。例如，TI 的 C24×系列 MCU 只具有 8 级深度的硬件堆栈，即只可以存储 8 个字。堆栈溢出一般不会有错误提示，使用时要谨慎。

1.2.2 运算器

运算器是执行算术运算和逻辑运算的部件，它的任务是对信息进行加工处理。运算器主要由算术逻辑单元(Arithmetic Logic Unit, ALU)和各种寄存器组成。主要的寄存器包括累加寄存器、程序状态字(Program Status Word, PSW)寄存器、暂存器(暂时存放数据)等。运算器主要负责加工和处理各种数据。相对控制器而言，运算器接收控制器(CU)的命令而进行动作，即运算器所进行的全部操作都是由 CU 发出的控制信号来指挥的，所以它是执行部件。图 1.2.2 为运算器的结构图，不同型号的 CPU 稍有不同。

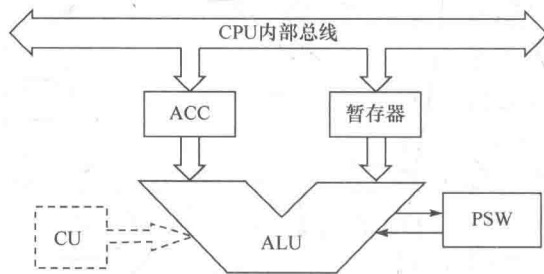


图 1.2.2 CPU 运算器结构框图

累加器是 ALU 使用最为频繁的一个寄存器，记为 A 或 ACC。两数相加，可以将加数或被加数之一存放在累加器中，完成求和运算后，结果仍送回 ACC，可以继续与暂存器中的数据相加，可见，ACC 具有累计的含义，故常称为累加器。

程序状态字 PSW 是反映程序运行状态的一个寄存器，主要用于存放一些操作结果的特征或标志，如用于寄存加、减运算结果是否溢出的标志，运算结果是否为零的标志，以及运算结果奇偶性的标志等。通过读取这些标志，可以了解运算结果的一些特征。

随着数字设计及制造技术的发展,微处理器功能的增强,CPU的内部除了上述基本部分,还会增加更多的寄存器、存储器管理部件、高速缓存部件等。

任何微处理器出厂后,PC、SP、ACC、PSW等寄存器的值确定,上电复位正常,寄存器的值则会被复位为器件手册中给定的值,搞清楚这些寄存器的初始值对于软件设计者非常重要。

1.3 CPU 工作流程(程序引导过程)

MCU的核心是CPU,无论CPU还是MCU,上电程序的引导都是由CPU处理的,CPU的工作过程就是执行程序的过程。CPU上电后,在程序指针PC的引导下,去程序存储器中读取用户程序。随着MCU处理器功能越来越强大,程序引导方式也越来越多,这就意味着程序存放的地方越来越多,不仅可以存储在系统本身的存储器中,还可以存储在便携移动存储器中。程序引导方式比较复杂的处理器,如TI DSP,出厂时片内都固化了引导加载程序(常称为Boot Loader),但无论多复杂,程序的引导都是在PC指引下,然后配合处理器规定的确定程序引导的I/O引脚(这种情况就作为输入引脚)的电平高低,则可以将程序PC指向期望的用户程序区域。换句话说,用户程序可以存储在多个地方,由引导的I/O引脚的不同取值,可以让PC转移到用户程序并开始读取第一条程序,提高程序存放的灵活性。

不同CPU执行程序指令的方式各不相同,执行指令的过程一般都包含取指令、指令译码、取操作数、执行指令等过程,各过程的意义如下。

- (1) 取指令: 控制器发出信息从存储器取一条指令。
- (2) 指令译码: 指令译码器将取得的指令翻译成对应的控制信号。
- (3) 取操作数: 如果需要操作数, 则从存储器取得该指令的操作数。
- (4) 执行指令: CPU按照指令操作码的要求, 在控制信号作用下完成规定的处理。

早期多数的CPU是按照上述顺序,一条一条串行地执行程序的,一条指令执行完才进行下一条指令的取指令环节。假设某CPU在第100个机器周期开始执行一条包含了取指令(简记为F)、指令译码(简记为D)、取操作数(简记为R)、执行指令(简记为E)等4个部分的指令,顺序执行程序的CPU工作流程如图1.3.1上面部分所示。这种方式的程序执行速度显然很慢。

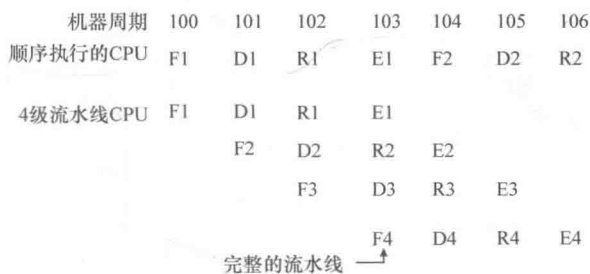


图 1.3.1 顺序执行和流水线执行程序对比