



普通高等教育“十三五”规划教材

单片机与嵌入式系统 原理及应用

王宝珠 冯文果 主编

免费电子课件



机械工业出版社
CHINA MACHINE PRESS

普通高等教育“十三五”规划教材

单片机与嵌入式系统原理及应用

主 编 王宝珠 冯文果
副主编 王 强 湛 丽 何永洪
参 编 黄 沛 喻 婷 费 莉
主 审 黄 俊



机械工业出版社

本书以 8051 单片机和 ARM11 微处理器 S3C6410 为主线,全面、系统地阐述了单片机嵌入式系统的原理和应用,主要内容包括单片机概述、MCS-51 单片机的硬件结构与原理、MCS-51 单片机指令系统、MCS-51 单片机汇编程序设计、MCS-51 单片机的 C 语言程序设计、MCS-51 单片机的内部资源、MCS-51 单片机的常用外设扩展、MCS-51 单片机接口技术、AT89C51 单片机应用设计与开发、嵌入式系统基础知识、ARM 微处理器体系结构、ARM11 微处理器 S3C6410、S3C6410 的 I/O 口及操作、S3C6410 的中断控制、S3C6410 的串口 UART、S3C6410 的 PWM 控制、S3C6410 的实时时钟、S3C6410 看门狗电路、工程项目开发实例。

本书结构严谨、逻辑清晰、叙述详细、通俗易懂,具有较强的编程实例和工程项目开发实例,可作为工科院校电气工程、电子信息工程、通信工程、自动化、机电一体化等专业本科生和专科生教材,也可供嵌入式系统的开发人员以及其他对嵌入式系统有兴趣的技术人员参考。

图书在版编目 (CIP) 数据

单片机与嵌入式系统原理及应用/王宝珠,冯文果主编. —北京:机械工业出版社,2018.2

普通高等教育“十三五”规划教材

ISBN 978-7-111-58893-1

I. ①单… II. ①王… ②冯… III. ①单片微型计算机-系统设计-高等学校-教材 IV. ①TP368.1

中国版本图书馆 CIP 数据核字 (2018) 第 003253 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑:徐凡 责任编辑:徐凡 韩静

责任校对:潘蕊 封面设计:张静

责任印制:孙炜

北京玥实印刷有限公司印刷

2018 年 2 月第 1 版第 1 次印刷

184mm × 260mm · 27 印张 · 661 千字

标准书号:ISBN 978-7-111-58893-1

定价:59.80 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

服务咨询热线:010-88379833

机工官网:www.cmpbook.com

读者购书热线:010-88379649

机工官博:weibo.com/cmp1952

教育服务网:www.cmpedu.com

封面无防伪标均为盗版

金书网:www.golden-book.com

前 言

自 20 世纪 70 年代问世以来,单片机已对人类社会产生了巨大的影响。尤其是美国 Intel 公司生产的 MCS-51 单片机,由于其具有集成度高、体积小、功能强、可靠性高、价格低等优点,已被广泛应用于工业测控、智能仪器仪表、家用电器等领域。此外, MCS-51 单片机也是教学用单片机的最佳选择。嵌入式系统是以应用为中心,以计算机技术为基础,软硬件可裁剪(这是指嵌入式系统的大小和规格会随着具体应用需求而改变),适用于应用系统对功能、可靠性、成本、体积、功耗有严格要求的专用计算机系统。嵌入式系统可以说是当前最热门、最有发展前途的 IT 应用领域之一。嵌入式系统通常会用在一些特定的专用设备上,特别是随着家电的智能化,嵌入式系统更显重要。目前国内嵌入式系统开发人才是很稀缺的,因为这一领域较新,且发展太快,所以熟练掌握这些新技术的人才是相当难找的,人才供需比是 1:8,所以就业前景非常好。

本书以实用为宗旨,采用理论与实际相结合的形式,用众多的实例来讲解理论知识。在内容安排上由浅入深、由易到难、通俗易懂。本书规划授课学时为 64 学时,章节内容安排偏多,但各章节内容之间既有相关继承性,又有一定的独立性,方便了读者学习和参考,各授课教师可根据学习对象的基础及需求不同,对授课学时进行灵活的调整。

本书的主要特点如下:

1. 本书图文并茂、实用性强,既可作为应用型本科自动化、电子信息类专业“单片机与嵌入式系统原理及应用”课程的教材和参考书,同时又可供各类电子工程和自动化技术人员、计算机爱好者以及嵌入式系统自学者参考。

2. 本书从实用角度出发,与传统的单片机与嵌入式基本原理书籍相比较,更注重面向实际应用和实际开发,书中案例大多来源于编者平时的教学、企业工程经验以及科研工作,有利于初学者迅速掌握单片机与嵌入式技术。

3. 本书致力于培养学生对学习方法的掌握,找出并抓住学科知识的内在联系,形成一个完整的体系,有利于学生系统地学习。

全书共 19 章。第 1 章主要介绍了计算机的基本概念、计算机的基本组成及单片微型计算机的结构原理,最后阐述了单片微型计算机的应用与发展。第 2 章主要讲解了 MCS-51 单片机的硬件结构和片内各功能部件的工作原理。第 3 章详细介绍了 MCS-51 单片机的寻址方式及指令系统。第 4 章介绍了汇编程序设计的方法及步骤,并举实例进行了说明。第 5 章介绍了 MCS-51 单片机的 C 语言程序设计,通过本章的学习,读者能够了解单片机 C 语言程序设计的方法。第 6 章对单片机的内部资源 I/O 口、定时/计数器、中断、串口进行了详细介绍。第 7 章主要介绍了程序存储器 (ROM) 扩展、数据存储器 (RAM) 扩展以及并行 I/O 口的扩展。第 8 章介绍了几种单片机常用的外设,包括键盘、显示器、打印机、A-D 和 D-A 转换器等。第 9 章主要介绍单片机产品的设计与开发,具体讨论了有关产品开发设计的问题。第 10 章主要对嵌入式系统的基本知识,包括基本概念、应用领域、特点、组成

及嵌入式处理器分类等进行了详细介绍,使读者对嵌入式系统有一个基本的了解。第11章主要介绍了常用ARM处理器系列,对ARM7内核、存储体系、总线结构、流水线技术、处理器状态与模式、寄存器组织和异常处理等进行了详细介绍,使学生对ARM的体系结构有一个清楚的认识。第12章主要介绍了S3C6410的内部资源,如定义的头文件、常用函数及其使用。第13章主要介绍了GPIO的功能、控制寄存器及其应用。第14章主要介绍了S3C6410中断控制系统的构成及应用实例。第15章主要介绍了S3C6410的串行端口RS232通信及编程方法。第16章主要介绍了PWM的工作原理、输出控制、控制寄存器的功能和编程思路。第17、18章主要介绍了S3C6410的RTC和看门狗的原理,以及利用相关的资源来编写相关的例程。第19章通过实际的工程项目介绍了一般嵌入式系统的开发流程。

本书的第1章由何永洪、王宝珠编写,第2~6章和附录由王宝珠编写,第12~19章由冯文果编写,第7~9章由王强编写,第10、11章由谌丽编写。全书由王宝珠统编定稿,由黄俊教授主审。同时,在本书编写过程中,黄沛、喻婷、费莉帮助收集资料、整理书稿,给予了大力的支持和帮助。

本书在编写过程中参考了大量的相关书籍和资料,在此向这些书籍和资料的编写者表示衷心的感谢。

由于编者水平有限,书中难免有疏漏、错误和不妥之处,敬请读者批评指正。

编 者

目 录

前 言	
第 1 章 概述	1
1.1 计算机基本概念	1
1.1.1 计算机的组成	1
1.1.2 信息在计算机中的表示	2
1.1.3 计算机的软件	5
1.2 微型计算机的基本构成	5
1.2.1 微处理器	5
1.2.2 存储器	6
1.2.3 系统总线	6
1.3 单片机的结构与特点	6
1.3.1 单片机的基本结构	6
1.3.2 单片机的主要特点	7
1.4 单片机的重要指标及类型	8
1.5 单片机的应用与发展	8
1.5.1 单片机的应用	8
1.5.2 单片机的发展	9
习题	12
第 2 章 MCS-51 单片机的硬件 结构与原理	13
2.1 51 系列单片机简介	13
2.2 MCS-51 单片机的硬件结构原理	14
2.2.1 基本结构组成	14
2.2.2 中央处理器	15
2.2.3 存储器	18
2.2.4 外部引脚	23
2.3 MCS-51 单片机的输入/输出端口	25
2.3.1 P0 口	26
2.3.2 P1 口	28
2.3.3 P2 口	28
2.3.4 P3 口	29
2.4 MCS-51 单片机的最小系统	30
2.4.1 电源	30
2.4.2 时钟电路	30
2.4.3 复位电路	33
2.5 MCS-51 单片机的工作方式	34
2.5.1 全速执行方式	34
2.5.2 单步执行方式	34
2.5.3 掉电及节电方式	35
习题	37
第 3 章 MCS-51 单片机指令系统	38
3.1 MCS-51 单片机的指令格式及 描述符号	38
3.1.1 指令格式	38
3.1.2 指令中用到的描述符号	39
3.2 MCS-51 单片机指令的寻址方式	39
3.2.1 立即寻址	40
3.2.2 直接寻址	40
3.2.3 寄存器寻址	40
3.2.4 寄存器间接寻址	41
3.2.5 变址寻址 (基址 + 变址寻址)	41
3.2.6 位寻址	42
3.2.7 绝对寻址	42
3.2.8 相对寻址	42
3.3 MCS-51 的指令系统	43
3.3.1 数据传送类指令	43
3.3.2 算术运算类指令	47
3.3.3 逻辑运算类指令	50
3.3.4 控制转移类指令	53
3.3.5 位操作类指令	57
3.4 MCS-51 单片机汇编程序常用 伪指令	59
习题	62
第 4 章 MCS-51 单片机汇编程序 设计	65
4.1 程序编制的方法和步骤	65
4.2 数据传送程序设计	65
4.3 查表程序设计	66
4.4 运算程序设计	67
4.4.1 算术运算程序设计	67
4.4.2 逻辑运算程序设计	70
4.5 代码转换程序设计	70
4.6 分支程序设计	71
4.6.1 简单分支程序设计	71

4.6.2 散转程序设计	72	5.9.5 C51 的中断函数	110
4.7 循环程序设计	75	5.9.6 C51 函数的寄存器组选择	111
4.8 子程序设计	76	5.9.7 C51 的重入函数	112
4.8.1 子程序的现场保护	77	习题	113
4.8.2 主程序和子程序间的参数传递	77		
习题	79		
第5章 MCS-51 单片机的 C 语言		第6章 MCS-51 单片机的内部	
程序设计	80	资源	114
5.1 C51 概述	80	6.1 MCS-51 的并行 I/O 口	114
5.1.1 C51 基本知识	80	6.2 MCS-51 单片机的中断系统	114
5.1.2 C51 程序结构	81	6.2.1 中断的概念	115
5.2 C51 的数据类型	81	6.2.2 MCS-51 单片机的中断源	116
5.2.1 C51 的基本数据类型	82	6.2.3 中断的控制	117
5.2.2 C51 特有的数据类型	83	6.2.4 中断响应	119
5.3 C51 的变量与存储类型	83	6.2.5 中断的编程及应用	120
5.3.1 C51 的普通变量及定义	84	6.3 MCS-51 单片机的定时/计数器	121
5.3.2 C51 的特殊功能寄存器变量	86	6.3.1 定时/计数器的主要特性	121
5.3.3 C51 的位变量	86	6.3.2 定时/计数器的结构和	
5.3.4 C51 的指针变量	87	工作原理	122
5.4 C51 的运算符和表达式	88	6.3.3 定时/计数器的控制	123
5.5 绝对地址的访问	90	6.3.4 定时/计数器的工作方式	124
5.5.1 使用 C51 运行库中的预定义宏	90	6.3.5 定时/计数器的编程及应用	127
5.5.2 通过指针访问	91	6.4 MCS-51 单片机的串行接口	132
5.5.3 使用 C51 扩展关键字_at_	92	6.4.1 串行通信的基本概念	132
5.6 C51 的并行接口	92	6.4.2 串行接口结构原理	134
5.7 流程控制语句	93	6.4.3 串行口的工作方式	136
5.7.1 表达式语句	93	6.4.4 串行口的编程及应用	140
5.7.2 复合语句	93	习题	149
5.7.3 条件语句	94		
5.7.4 开关语句	95	第7章 MCS-51 单片机的常用	
5.7.5 循环语句	96	外设扩展	151
5.7.6 跳转语句	98	7.1 存储器扩展设计	151
5.8 构造数据	99	7.1.1 单片机程序存储器概述	151
5.8.1 数组	100	7.1.2 EPROM 扩展	152
5.8.2 指针	101	7.2 数据存储器扩展	154
5.8.3 结构	103	7.2.1 SRAM 扩展实例	154
5.8.4 联合	106	7.2.2 外部 RAM 与 I/O 同时扩展	156
5.8.5 枚举	107	7.3 并行 I/O 口扩展	157
5.9 C51 中的函数	107	7.3.1 简单 I/O 口扩展	158
5.9.1 C51 函数的参数传递	108	7.3.2 基于可编程芯片 8255A 的扩展	159
5.9.2 C51 函数的调用与声明	109	习题	165
5.9.3 C51 函数的返回值	110		
5.9.4 C51 函数的存储模式	110	第8章 MCS-51 单片机接口技术	166
		8.1 MCS-51 单片机与 LED 显示器的	
		接口	166
		8.1.1 LED 显示器的结构与原理	166

8.1.2 LED 数码管的显示方式	168	10.6 嵌入式操作系统的概念与分类	225
8.1.3 LED 显示器与单片机的接口	169	10.6.1 嵌入式操作系统的概念	225
8.2 MCS-51 单片机与键盘的接口	172	10.6.2 嵌入式操作系统的分类	226
8.2.1 键盘的工作原理	172	习题	227
8.2.2 独立式键盘与单片机的接口	173	第 11 章 ARM 微处理器体系结构	228
8.2.3 矩阵式键盘与单片机的接口	174	11.1 ARM 简介	228
8.3 MCS-51 单片机与 A-D 或 D-A		11.1.1 RISC 结构特性	228
转换器的接口	176	11.1.2 常用 ARM 处理器系列	229
8.3.1 MCS-51 单片机与 D-A 转换器的		11.2 ARM7 TDMI 模块、内核和功能	
接口	176	框图	232
8.3.2 MCS-51 单片机与 A-D 转换器的		11.2.1 ARM7 TDMI 模块框图	232
接口	180	11.2.2 ARM7 TDMI 内核框图	232
习题	185	11.2.3 ARM7 TDMI 功能框图	233
第 9 章 AT89C51 单片机应用设计与		11.3 ARM 的存储体系	234
开发	186	11.4 ARM 的总线结构	235
9.1 AT89C51 单片机系统设计步骤	186	11.5 ARM 的流水线技术	237
9.1.1 设计任务	186	11.5.1 流水线的概念与原理	237
9.1.2 应用系统设计	186	11.5.2 流水线的分类	238
9.1.3 硬件设计	187	11.5.3 影响流水线性能的因素	240
9.1.4 软件设计	188	11.6 ARM 的工作状态	241
9.1.5 系统调试	189	11.7 ARM 的工作模式	242
9.2 AT89C51 单片机系统抗干扰技术	189	11.8 ARM 的寄存器组织	243
9.2.1 干扰源及其传播途径	189	11.8.1 ARM 状态下的寄存器组织	243
9.2.2 抗干扰措施的电源设计	190	11.8.2 Thumb 状态下的寄存器组织	245
9.2.3 产品的地线设计	193	11.8.3 程序状态寄存器	246
9.2.4 A-D 和 D-A 转换器的抗干扰		11.9 ARM 的异常处理	248
措施	194	习题	252
9.2.5 传输干扰	195	第 12 章 ARM11 微处理器 S3C6410	253
9.2.6 抗干扰措施的元器件	196	12.1 S3C6410 简介	253
9.3 单片机应用系统设计实例	198	12.2 S3C6410 芯片结构	253
9.3.1 数字时钟设计	198	12.3 S3C6410 封装及引脚定义	257
9.3.2 市电频率测量设计	210	12.4 存储器映射	269
习题	218	12.4.1 高地址区域	270
第 10 章 嵌入式系统基础知识	219	12.4.2 低地址区域	270
10.1 嵌入式系统的概念	219	12.5 S3C6410 处理器时钟和电源管理	271
10.2 嵌入式系统的特点	220	12.5.1 时钟源的选择	271
10.3 嵌入式系统的应用	221	12.5.2 PLL 和总线时钟	271
10.4 嵌入式系统的组成	222	12.5.3 电源管理	273
10.4.1 嵌入式处理器	222	12.5.4 复位方式	274
10.4.2 外围设备	222	12.6 S3C6410 内部资源定义的头文件及	
10.4.3 嵌入式操作系统	223	常用函数	274
10.4.4 应用软件	223	12.6.1 头文件	274
10.5 嵌入式处理器的类型	223	12.6.2 常用函数	283

习题	285	16.1.3 S3C6410 的自动重新加载和 双缓冲功能	375
第 13 章 S3C6410 的 I/O 口及操作 ...	286	16.1.4 定时器的基本操作示例	376
13.1 S3C6410 I/O 概述	286	16.2 PWM 输出电平控制	376
13.1.1 GPIO 特性	286	16.2.1 PWM 工作原理	376
13.1.2 GPIO 控制寄存器分类	287	16.2.2 PWM 输出控制	377
13.2 S3C6410 I/O 端口控制寄存器	287	16.3 PWM 定时器控制寄存器	378
13.3 I/O 控制的 C 语言编程实例	317	16.4 定时器控制编程实例	386
13.3.1 硬件电路	318	16.4.1 硬件电路	386
13.3.2 实现功能和编程思路	318	16.4.2 参考程序	386
13.3.3 参考程序	318	习题	388
习题	319	第 17 章 S3C6410 的实时时钟	389
第 14 章 S3C6410 的中断控制	320	17.1 S3C6410 的实时时钟概述	389
14.1 S3C6410 中断控制器概述	320	17.1.1 S3C6410 的 RTC 单元	389
14.2 S3C6410 中断源及中断号	320	17.1.2 RTC 控制寄存器	391
14.3 外部中断与控制寄存器	322	17.2 RTC 应用编程实例	398
14.3.1 外部中断源分组	323	习题	400
14.3.2 外部中断控制寄存器	323	第 18 章 S3C6410 看门狗电路	401
14.3.3 外部中断优先级仲裁及 中断号	339	18.1 S3C6410 看门狗概述	401
14.4 中断处理过程及控制器	341	18.1.1 S3C6410 看门狗的工作原理	401
14.4.1 中断流程	341	18.1.2 S3C6410 看门狗的功能	401
14.4.2 中断控制器	342	18.1.3 S3C6410 看门狗控制寄存器	402
14.5 中断程序编写实例	350	18.2 看门狗控制编程实例	404
14.5.1 编程思路	351	18.2.1 例程思路	404
14.5.2 实例程序	351	18.2.2 参考程序	404
习题	355	习题	406
第 15 章 S3C6410 的串口 UART	356	第 19 章 工程项目开发实例	407
15.1 S3C6410 的串口概述	356	19.1 工程项目任务和软硬件准备	407
15.1.1 S3C6410 串行通信单元	356	19.1.1 项目任务	407
15.1.2 UART 通信操作	358	19.1.2 项目的软硬件准备	407
15.2 UART 的控制寄存器	359	19.2 工程项目建立步骤	407
15.3 UART 通信程序实例	369	19.2.1 项目整体思路	407
15.3.1 RS232 接口电路	369	19.2.2 建立工程项目	409
15.3.2 编程思路	370	19.2.3 编写(参考)程序	410
15.3.3 UART 实例程序	370	19.2.4 工程环境配置	412
习题	372	19.2.5 工程编译方法	417
第 16 章 S3C6410 的 PWM 控制	373	19.2.6 工程文件下载	417
16.1 PWM 定时器概述	373	附录 MCS-51 单片机指令表	420
16.1.1 脉宽调制的概念和原理	373	参考文献	424
16.1.2 S3C6410 的 PWM 定时器	373		

第 1 章 概 述

自从 1946 年世界上第一台电子计算机诞生以来，电子计算机得到了飞速的发展。从当初的运算速度仅每秒 5000 次加法、400 次乘法，到 2013 年我国推出的超级计算机“天河二号”，已达到每秒 3.39 亿亿次的浮点运算速度，其发展历程只有半个多世纪，可谓日新月异。如今，随着计算机的广泛应用，它正在深刻地影响着人们的生活，我们很难设想没有了计算机的生活会怎样。

1.1 计算机基本概念

1946 年，美国为计算弹道轨迹而研制成功了世界第一台现代电子计算机。其基本结构是由冯·诺依曼在 1946 年于 EDVAC 报告中提出的，称为冯·诺依曼体系结构，当代计算机仍在采用这样的体系结构。

1.1.1 计算机的组成

计算机系统由硬件与软件两个部分构成。硬件由相关设备与硬件电路构成，是计算机的运行平台。软件由程序与数据构成，程序在硬件平台上运行，实现了计算机的逻辑功能。

计算机硬件由算术逻辑单元、控制器、存储器、输入设备和输出设备五大部件构成，如图 1-1 所示。其体系结构沿用至今。

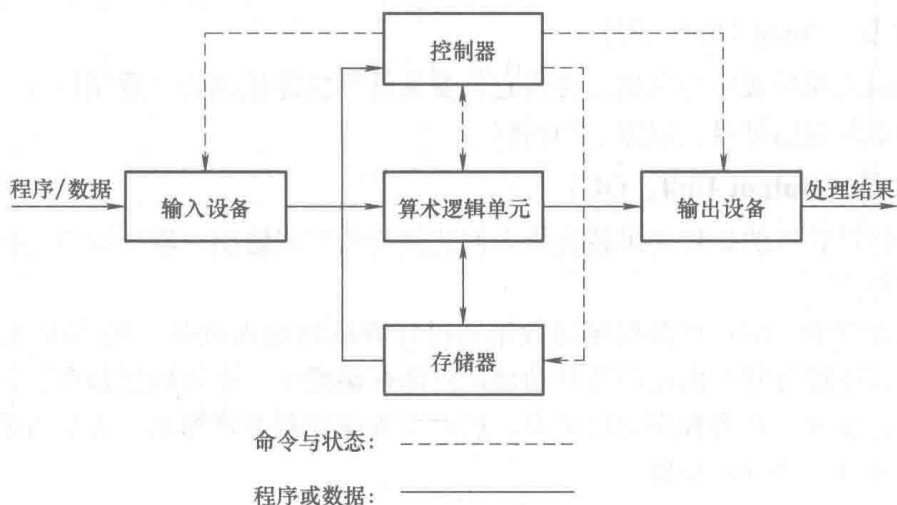


图 1-1 计算机结构原理

1. 算术逻辑单元 (Arithmetic Logic Unit, ALU)

算术逻辑单元包括运算器与通用寄存器。运算器用于算术与逻辑运算；算术运算实现各种数值运算，比如加、减、乘、除等；逻辑运算进行逻辑处理与判断，比如与、或、非、比

较、移位等。通用寄存器用来暂时存放参加运算的原始数据（操作数）与中间结果。操作数来自内存或输入设备。

2. 控制器 (Control Unit, CU)

控制器是执行程序部件。程序是完成某一特定功能的指令序列。指令在控制逻辑电路控制下，从内部存储器取指令、分析指令、执行指令。控制器由程序计数器、指令寄存器、指令译码器及控制逻辑等电路构成。其功能作用如下：

1) 程序计数器 (Program Counter, PC)：用于存放下一条将要执行的指令地址。计算机将根据其地址取出指令。PC 在取出一个指令字节后会自动加 1，以指向下一指令字节。因此，计算机通常是自动地顺序执行程序，只有在执行转移类指令的时候才会改变顺序。PC 的初值是程序在存储器中的起始位置，是存程序的第一条指令的地方。

2) 指令寄存器 (Instruction Register, IR)：用于存放当前正在执行的指令，并且保存至执行完毕。指令由内部存储器输入到该寄存器中。

3) 指令译码器 (Instruction Decoder, ID)：分析解释指令寄存器中的指令功能。电路将当前指令的操作码分解成若干不能再分解的微操作，并将这些微操作信号序列输出。

4) 控制逻辑 (Sequential Control Logic, SCL)：由时钟电路、分频器、节拍发生器等电路构成。系统由它来安排执行上述微操作信号序列的工作时序，包括取指令与执行指令。

3. 存储器 (Memory, MEM)

计算机系统通常采用二级存储。内存储器采用半导体存储器，用来存放计算机正在运行的程序与数据，具有易失性、速度快的特点，但由于价格昂贵所以容量有限。外存储器采用硬盘、光盘、U 盘等，被用来存放暂时没有运行的程序与数据，具有非易失性、速度慢的特点，但由于价格低廉所以容量大，被称为海量存储器。外存储器属于计算机的 I/O 设备。

4. 输入设备 (Input Unit, IU)

输入设备输入原始程序与数据，并将它们变为计算机能识别的二进制码存入内部存储器中。输入设备通常包括键盘、鼠标、扫描仪等。

5. 输出设备 (Output Unit, OU)

输出设备将计算机的处理结果转换成人们能接受的形式输出。输出设备包括显示器、打印机、绘图仪等。

在计算机运行中，用户将源程序与数据通过计算机的输入设备，经 ALU 或者直接存入内部存储器。程序运行时，由控制器自动地从内部存储器中一条条取出指令、分析指令并执行，对数据进行处理，直到程序运行结束。然后在程序控制下将结果以人们需要的方式经输出设备以打印或显示等方式输出。

1.1.2 信息在计算机中的表示

在计算机中，无论是程序还是数据都是放在存储器内，且都是以二进制方式表示的。人们将信息以不同的时间加以区分。在取指周期中，计算机从存储器中取出的是指令，这些“0”“1”代码称为指令码；在指令的执行周期中，从存储器取出的是数据，是指令的操作数。

最初的计算机主要用于科学计算，如今的计算机还要用于字符、语音、图像等信息的处理，这些信息通常采用特定的编码形式表示。

1. 数在计算机中的表示

(1) 无符号数

无符号数二进制的的所有位都是数值的有效位。所以一个 8 位的无符号数的表示范围是 0 ~ 255，对应的二进制为 00000000B ~ 11111111B。

(2) 符号数

符号数最高位为符号位，其余二进制位是数值位。计算机的符号数通常采用补码方式表示。一个 n 位二进制整数（不包括符号位）的补码定义如下：

$$[X]_{\text{补}} = \begin{cases} X & 0 \leq X < 2^n \\ 2^{n+1} + X & -2^n \leq X < 0 \end{cases}$$

所以，对于一个 7 位的二进制符号数，加上它的一个符号位即 8 位二进制数，其补码表示范围由上式可算得。

例如：一个 8 位二进制最大的正数的补码是原码本身，即 01111111B，对应十进制数为 127；一个 8 位二进制最小的负数的补码为：10000000B + (-10000000B) = 10000000B，对应十进制数为 -128。所以对应 8 位二进制补码表示范围是：-128 ~ +127。

补码转换也可以采取在原码基础上除符号位取反加 1 的方法求得。

求补运算，就是将一个包括符号位的二进制数全部取反加 1 的运算。这是补码运算中将正数变负数，负数变正数的补码运算。

(3) BCD 码

BCD 码又称二-十进制码，即用二进制方式来表示的十进制码，见表 1-1。一位 BCD 码由 4 位二进制数表示，按其位权计值，也称 8-4-2-1 码。

表 1-1 BCD 码

十进制数	BCD 码	说 明
0	0000	有效编码
1	0001	有效编码
2	0010	有效编码
3	0011	有效编码
4	0100	有效编码
5	0101	有效编码
6	0110	有效编码
7	0111	有效编码
8	1000	有效编码
9	1001	有效编码
10 ~ 15	1010 ~ 1111	无效编码

由于BCD码编码效率低(只有0.625%的效率),因此很少直接用于数字运算,而通常用于数据的输入与输出,以符合人们熟悉的数的表示习惯。所以,在计算机中数值运算与处理通常采用二进制数进行。BCD码分为非压缩BCD码与压缩BCD码两种形式。

1) 非压缩BCD码:一个字节一位BCD码。

	BCD
--	-----

2) 压缩BCD码:一个字节两位BCD码。

BCD	BCD
-----	-----

2. 字符在计算机中的表示

计算机中的文本文件,比如用户设计的C语言源程序、文字信息等,采用的是二进制的编码形式。其中,英文采用的是ASCII码,汉字采用的是汉字编码,如国标码。

ASCII——American Standard Code for Information Interchange,美国信息交换标准代码,由文本字符与控制字符组成。

文本字符包括大小写英文字母、数字、标点符号、运算符、制表符等,编码为20H~7EH,共96个。其中十六进制20H代表空格,如表1-2给出的ASCII码文本字符对照表所示。

表 1-2 ASCII 码文本字符对照表

十六进制	字符	十六进制	字符	十六进制	字符	十六进制	字符	十六进制	字符	十六进制	字符
20H		30H	0	40H	@	50H	P	60H	,	70H	p
21H	!	31H	1	41H	A	51H	Q	61H	a	71H	q
22H	"	32H	2	42H	B	52H	R	62H	b	72H	r
23H	#	33H	3	43H	C	53H	S	63H	c	73H	s
24H	\$	34H	4	44H	D	54H	T	64H	d	74H	t
25H	%	35H	5	45H	E	55H	U	65H	e	75H	u
26H	&	36H	6	46H	F	56H	V	66H	f	76H	v
27H	'	37H	7	47H	G	57H	W	67H	g	77H	w
28H	(38H	8	48H	H	58H	X	68H	h	78H	x
29H)	39H	9	49H	I	59H	Y	69H	i	79H	y
2AH	*	3AH	:	4AH	J	5AH	Z	6AH	j	7AH	z
2BH	+	3BH	;	4BH	K	5BH	[6BH	k	7BH	{
2CH	,	3CH	<	4CH	L	5CH	\	6CH	l	7CH	
2DH	-	3DH	=	4DH	M	5DH]	6DH	m	7DH	}
2EH	.	3EH	>	4EH	N	5EH	^	6EH	n	7EH	~
2FH	/	3FH	?	4FH	O	5FH	_	6FH	o	7FH	DEL

00H~1FH及7FH,共33个控制字符,主要用于文本的编辑与控制。如表1-3列出的ASCII码控制字符对照表所示。

表 1-3 ASCII 码控制字符对照表

十六进制	缩写	名称及意义	十六进制	缩写	名称及意义
00H	NUL	Null (空)	11H	DC1	Device Control 1 (设备控制 1)
01H	SOH	Start of Heading (报头开始)	12H	DC2	Device Control 2 (设备控制 2)
02H	STX	Start of Text (正文开始)	13H	DC3	Device Control 3 (设备控制 3)
03H	ETX	End of Text (正文结束)	14H	DC4	Device Control 4 (设备控制 4)
04H	EOT	End of Transmission (传输结束)	15H	NAK	Negative Acknowledge (否认)
05H	ENQ	Enquiry (查询)	16H	SYN	Synchronous Idle (同步空闲)
06H	ACK	Acknowledge (确认)	17H	ETB	End of Transmission Block (传输块结束)
07H	BEL	Bell (振铃)	18H	CAN	Cancel (取消)
08H	BS	Backspace (退格)	19H	EM	End of Medium (介质结束)
09H	HT	Horizontal Tab (水平制表)	1AH	SUB	Substitute (替换)
0AH	LF	Line Feed (换行)	1BH	ESC	Escape (转义)
0BH	VT	Vertical Tab (垂直制表)	1CH	FS	File Separator (文件分隔符)
0CH	FF	Form Feed (换页)	1DH	GS	Group Separator (分组符)
0DH	CR	Carriage Return (回车)	1EH	RS	Record Separator (记录分隔符)
0EH	SO	Shift Out (移出)	1FH	US	Unit Separator (单元分隔符)
0FH	SI	Shift In (移入)	7FH	DEL	Delete (删除)
10H	DLE	Data Link Escape (数据链路转义)			

1.1.3 计算机的软件

计算机软件由系统软件与应用软件构成。

系统软件 (System Software) 包括操作系统、计算机语言处理系统、数据库管理系统、系统实用程序等。其中操作系统是计算机与程序员的用户界面, 其他软件都是程序员的工具软件。

应用软件是用户自行设计的软件, 如自动化控制系统、电信计费系统等。

1.2 微型计算机的基本构成

1.2.1 微处理器

微型计算机由中央处理器、存储器、输入/输出设备构成。中央处理器是指计算机内部对数据进行处理并对处理过程进行控制的部件。伴随着大规模集成电路技术的迅速发展, 芯片集成密度越来越高, CPU 可以集成在一个半导体芯片上, 这种具有中央处理器功能的大规模集成电路器件, 被统称为微处理器。系统各功能部件采用单总线方式连接, 这种系统被称为微型计算机, 如图 1-2 所示, 这样的系统结构简单, 体积小。由于它的小型化、低成本, 过去也称其为台式机 (Desktop) 或桌面机, 又叫个人计算机。

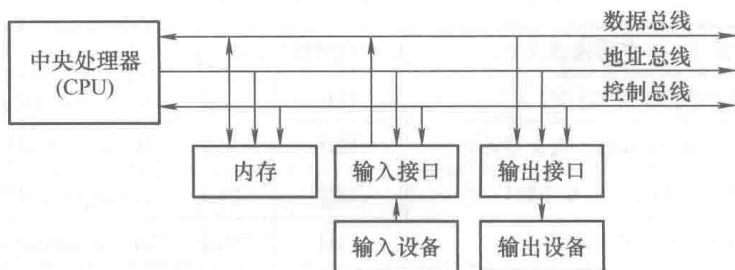


图 1-2 微型计算机硬件结构

1.2.2 存储器

为了追求最快的存取速度与最好的性价比，微型计算机的存储器往往采用多级存储结构，按所在物理位置分为内部存储器与外部存储器。

内部存储器用来存放当前正在运行的程序（指令）与数据。目前内部存储器主要由半导体器件构成，相对外部存储器来说，其速度快、价格贵，所以容量小。

外部存储器也称辅助存储器，用来存放备用数据或程序，如软盘、硬盘、CD-ROM 等，都属于计算机的外部存储器。其价格相对低，容量大，所以又称为海量存储器。

在微型计算机系统中，各种用户程序存放在外部存储器中。当系统运行程序时，计算机首先将用户程序从外部存储器读入内存，再由计算机启动 CPU 运行程序。

1.2.3 系统总线

系统总线是 CPU 与计算机其他各功能部件信息传送的公共通道，采用总线方式连接可以大大减少各功能部件之间的连接复杂度，使系统连接简单灵活。按照信息的不同类型，系统总线通常由三组信号线组成，即数据总线、地址总线和控制总线。

数据总线是 CPU 与存储器之间及 CPU 与 I/O 之间传送数据或指令信息的公共通道，其传送是双向的，数据总线的宽度与 CPU 字长相同。

地址总线用来传送 CPU 发出的所要访问的单向的设备地址信息。计算机系统的每一个存储单元、每一个 I/O 端口都分配有唯一地址，用于 CPU 对设备的访问。

控制总线用来传送 CPU 发出到其他设备的或者由其他各设备发出到 CPU 的操作控制信号。对于每一个控制信号其方向是固定的，其控制总线长度与系统的基本操作类型有关。

1.3 单片机的结构与特点

随着微电子技术与计算机技术的不断发展，计算机的应用越来越广泛，它不仅向大型化发展，也在不断地向小型化方面发展。单片机就是由微型计算机发展而来的。

1.3.1 单片机的基本结构

单片机就是在一片半导体硅片上集成了中央处理单元（CPU）、存储器（RAM、ROM）、并行 I/O、串行 I/O、定时/计数器、中断系统、系统时钟电路及系统总线的微型计算机。它具有微型计算机的属性，因而被称为单片微型计算机，简称单片机。图 1-3 所示为单片机结构图。

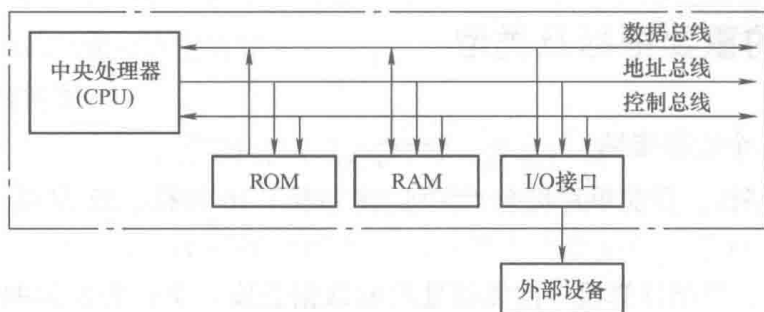


图 1-3 单片机结构图

单片机的存储器由 ROM 与 RAM 构成。ROM 用来固化应用程序，使程序上电就能运行，这点与微型计算机不一样。RAM 用来存放程序相关数据。对于单片机的 I/O 接口，生产商会根据不同应用植入对应的 I/O 来满足需求，在实际应用中，用户只需要考虑如何将应用设备与单片机的 I/O 相连就可以了，这就大大简化了单片机应用系统的硬件设计。如果片内没有所需 I/O 接口电路，用户可以使用芯片的可扩展端口扩展 I/O 接口电路。

单片机体积小、功耗低，作为系统功能部件很容易被嵌入到系统中，最初的单片机是应用于测控领域中。国际上也称单片机为嵌入式控制器 (Embedded MicroController Unit, EM-CU)，或称微控制器 (MicroController Unit, MCU)。

1.3.2 单片机的主要特点

单片机本身就是一个微型计算机，因此只要在单片机的外部适当增加一些必要的外围扩展电路，就可以灵活地构成各种应用系统。单片机具有如下特点。

1. 体积小

在应用系统中，单片机作为系统中的一个部件嵌入，因此要求单片机应尽可能小。目前号称全球最小封装的单片机是 PIC10F20X，它采用了 SOT-23-6 封装，其尺寸仅为 $3 \times 3 \text{mm}^2$ 。如此小的封装，几乎可以在任何设计中使用，不必考虑 PCB 空间。

2. 通常采用哈佛结构

单片机的存储器采用哈佛 (Harvard) 结构，程序存储器与数据存储器分开。程序存储器采用只读存储器 (ROM)，用来存放程序与常数。数据存储器采用随机存储器 (RAM)，用来存放与程序相关的数据。

3. 引脚复用

为了降低单片机的体积与成本，又能提供更多的功能，单片机的引脚多采取复用方式，包括功能复用与分时复用。

1) 功能复用。一个引脚可以有多种功能，但是同一个应用只能选用其中的一种功能而不能同时占用多种。例如，MCS-51 单片机的 P3 口的 P3.0 与 P3.1 引脚，既可以作为普通 I/O 端口用，又可以作为串行接口用，但具体应用时只能使用其中一种。

2) 分时复用。一个引脚可以有多种功能，应用可以分时复用这些功能。例如，MCS-51 单片机的 P0 口在总线扩展时，可以是地址总线的 A0 ~ A7，也可以是数据总线的 D0 ~ D7，但这两种不能同时出现在该端口上。

1.4 单片机的重要指标及类型

1. 单片机的几个重要指标

1) 单片机的字长。目前单片机有 4 位机、8 位机、16 位机、32 位机，可以根据需求来选择。

计算机的字长，是指计算机一次能够处理的数据长度。字长为 8 位的计算机处理 16 位数据需要 2 次，字长为 16 位的计算机处理 16 位数据只需要 1 次。显然，后者的处理能力比前者强。

2) 单片机的时钟。单片机的速度与时钟成正比，可以根据应用需求选择。

3) 低电压。2.7V 即可工作。

4) 低功耗。可以低到 nA 级，更适用于电池供电。

2. 单片机的两种类型

按照用途不同，单片机可分为通用型单片机和专用型单片机。

(1) 通用型单片机

单片机内部资源（如存储器空间与 I/O 资源）是有限的，通用型单片机是将这些资源通过端口全部开放给用户。用户可以此为核，再配以外围接口电路针对应用需求进行单片机扩展。例如，89C51 内部有 128B 数据存储器、4KB 程序存储器，I/O 口包括 1 个串行口、2 个定时/计数器、4 个 8 位并行口，显然内部资源是匮乏的。但是它的 P0、P2 口可以扩充 16 位地址总线与 8 位数据总线，这使得 89C51 的程序存储器与数据存储器可以扩展到 64KB；片内没有的 I/O 口也可以在这些总线上增加新的 I/O 接口。

(2) 专用型单片机

专用型单片机是专门针对特定用途而设计的单片机产品，这种单片机只用于特定产品开发。较之通用型单片机，专用型单片机的系统结构简单、成本低，所以专用型单片机具有十分明显的综合优势与较高的性价比。例如，nRF24E1 片内集成了 8051 内核，I/O 接口集成了 nRF2401 射频收发器、10 位/100ks/s A-D 转换器、PWM、SPI，它是专为射频产品开发的单片机。

在单片机应用开发中，可先使用通用型单片机设计成开发平台，针对不同的产品进行开发，再与 IC 厂商合作定制成专用型单片机，这样的应用系统结构简单，可靠性高且成本低。

1.5 单片机的应用与发展

1.5.1 单片机的应用

单片机的应用就是以单片机为硬件平台，用计算机语言（C 语言、汇编语言）来设计系统功能。即便是产品的功能升级也仅需修改软件而已，而不必修改硬件电路，这是用纯数字电路设计的系统不可比拟的。

随着人们对单片机应用的广泛需求，IC 生产商针对不同市场，开发出了各种不同类型