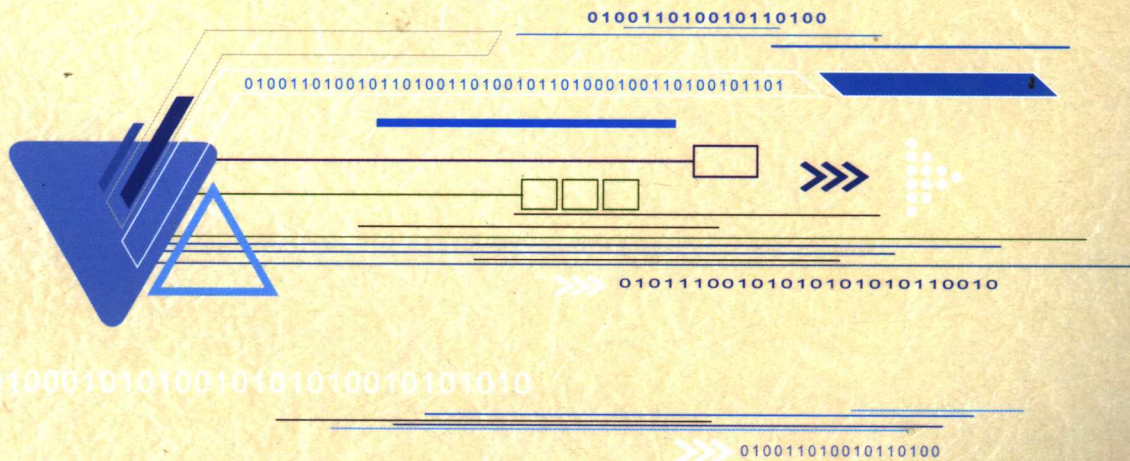


高等院校通用教材

单片机原理及接口技术

(第3版)



李朝青 编著



北京航空航天大学出版社

高等院校通用教材

单片机原理及接口技术

(第3版)

李朝青 编著

北京航空航天大学出版社

内 容 简 介

本书以 89C51 为典型机,深入浅出地讲述单片机原理、接口及应用技术。主要内容包括:微机基础知识、89C51 单片机硬件结构、指令系统、汇编语言程序设计、中断系统、定时器及应用、89C51 串行口及串行通信、89C51 单片机小系统及片外扩展、应用系统配置及接口技术和系统应用程序实例。

该书内容新颖、实用,删去了原版教材中大部分并行扩展的内容,特别增加了较多串行外设芯片扩展的内容,如 SPI、I²C 和单总线串行扩展技术、串行 A/D、D/A、键盘输入和显示器等实例,可用作大中专院校微机原理、单片机及接口技术的教材,也可供从事单片机产品开发的工程技术人员参考。

图书在版编目(CIP)数据

单片机原理及接口技术/李朝青编著. —3 版. —北京:北京航空航天大学出版社,2005.9

ISBN 7-81077-545-6

I. 单… II. 李… III. ①单片微型计算机—理论—高等学校—教材②单片微型计算机—接口—高等学校—教材 IV. TP368.1

中国版本图书馆 CIP 数据核字(2005)第 067897 号

单片机原理及接口技术(第 3 版)

李朝青 编著

责任编辑 王 瑛 崔肖娜

*

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(100083) 发行部电话:(010)82317024 传真:(010)82328026

<http://www.buaapress.com.cn> E-mail: bhp@263.net

北京松源印刷有限公司印装 各地书店经销

*

开本:787×960 1/16 印张:20.75 字数:465 千字

2005 年 10 月第 3 版 2005 年 10 月第 1 次印刷 印数:5 000 册

ISBN 7-81077-545-6 定价:27.00 元

前 言

《单片机原理及接口技术(简明修订版)》自 1999 年出版以来,得到各大中专院校同行的认可,在此我们表示诚挚的谢意。现已第 17 次印刷。鉴于单片机及嵌入式系统技术发展迅速,该教材很多内容已经陈旧,现决定对本教材再次修订。

这次修订内容变化较大,从以往的 8051/8031 芯片为主转变为以 89C51 芯片为主的典型机。89C51 具有 80C51 内核,仍属于 MCS-51 系列单片机,简称 51 系列单片机。本书删去 EPROM 扩展及 I/O 口芯片扩展的内容,用户可根据需要选择 89 系列不同容量 Flash ROM 的产品。为了节省 89C51 的 I/O 口线,选择了一些串口(SPI 或 I²C)A/D、D/A、E²PROM、看门狗、键盘和显示器的实例。这样,89C51 在不扩展片外 I/O 口芯片的情况下,即可构成完整的测控系统。

各章习题与原教材区别不大,其习题解答及考题库可在北京航空航天大学出版社 2005 年出版的《单片机学习指导》(与本书配套)一书中找到,敬请关注。

参加本教程编写的还有郝廷柱、刘艳玲、张秋燕、曹文嫣、曹玉珍(天津大学)、王志勇、何宏、沈怡琳、袁其平、李运等。

由于作者水平所限,难免出现错误和不妥之处,敬请同行及读者提出宝贵意见。

本教材还配有教学课件。需要用于教学的教师,请与北京航空航天大学出版社联系。北京航空航天大学出版社联系方式如下:

通信地址:北京海淀区学院路 37 号北京航空航天大学出版社教材推广部

邮 编:100083

电 话:010-82317027

传 真:010-82327026

E-mail: zhujunwei_218@163.com

李朝青
天津理工大学电子系
2005 年 5 月

目 录

第 1 章 微机基础知识

1.1 微处理器、微机和单片机的概念	1
1.1.1 微处理器(机)的组成	1
1.1.2 存储器和输入/输出接口	4
1.2 常用数制和编码	5
1.2.1 数制及数制间转换	5
1.2.2 计算机中常用编码	11
1.3 数据在计算机中的表示	13
1.3.1 有符号数	13
1.3.2 无符号数	14
1.4 89C51 单片机	15
1.5 思考题与习题	15

第 2 章 89C51 单片机硬件结构和原理

2.1 89C51 单片机芯片内部结构及特点	17
2.1.1 89C51 单片机的基本组成	17
2.1.2 89C51 单片机芯片内部结构	18
2.2 89C51 单片机引脚及其功能	21
2.3 89C51 单片机存储器配置	24
2.3.1 程序存储器地址空间	25
2.3.2 数据存储器地址空间	27
2.4 时钟电路及 89C51 CPU 时序	33
2.4.1 片内时钟信号的产生	34
2.4.2 CPU 取指、执指时序	36
2.5 复位操作	37
2.5.1 复位操作的主要功能	37
2.5.2 复位信号及其产生	38
2.5.3 复位电路	39

2.6	89C51 单片机的低功耗工作方式	40
2.6.1	方式的设定	40
2.6.2	空闲(等待、待机)工作方式	41
2.6.3	掉电(停机)工作方式	41
2.7	输入/输出端口	42
2.7.1	P1 口	42
2.7.2	P2 口	43
2.7.3	P0 口	43
2.7.4	P3 口	46
2.7.5	端口的负载能力和接口要求	47
2.8	思考题与习题	47

第3章 指令系统

3.1	汇编语言	49
3.1.1	指令和程序设计语言	49
3.1.2	指令格式	50
3.2	寻址方式	51
3.2.1	7种寻址方式	52
3.2.2	寻址空间及符号注释	56
3.3	89C51 单片机的指令系统	57
3.3.1	数据传送指令	58
3.3.2	算术运算指令	64
3.3.3	逻辑操作指令	69
3.3.4	控制程序转移类指令	72
3.3.5	位操作(布尔处理)类指令	80
3.4	思考题与习题	84

第4章 汇编语言程序设计知识

4.1	编程的步骤、方法和技巧	89
4.1.1	编程步骤	89
4.1.2	编程的方法和技巧	91
4.1.3	汇编语言程序的基本结构	92
4.2	汇编语言源程序的编辑与汇编	106
4.2.1	源程序编辑	106

4.2.2 源程序的汇编	107
4.2.3 伪指令	108

第5章 中断系统

5.1 微机的输入/输出方式	112
5.1.1 无条件传送方式	112
5.1.2 查询传送方式	112
5.1.3 直接存储器存取(DMA)方式	113
5.2 中断的概念	113
5.3 89C51 中断系统结构及中断控制	114
5.3.1 89C51 中断源	115
5.3.2 中断控制	116
5.4 中断响应及中断处理过程	121
5.4.1 中断响应	121
5.4.2 中断处理	123
5.4.3 中断返回	124
5.4.4 关于具体的中断服务程序	124
5.5 中断程序举例	125
5.5.1 主程序	125
5.5.2 中断服务程序	126
5.6 思考题与习题	129

第6章 定时器及应用

6.1 定时器概述	131
6.2 定时器的控制	132
6.2.1 工作模式寄存器 TMOD	132
6.2.2 控制寄存器 TCON	133
6.3 定时器的4种模式及应用	134
6.3.1 模式1及应用	135
6.3.2 模式2及应用	135
6.3.3 模式3及应用	138
6.3.4 综合应用举例	140
6.4 思考题与习题	145

第7章 89C51 串行口及串行通信技术

7.1 串行通信基本知识	147
7.1.1 数据通信	147
7.1.2 串行通信的传输方式	148
7.1.3 异步通信和同步通信	149
7.1.4 串行通信的过程及通信协议	151
7.2 串行口及应用	152
7.2.1 89C51 串行口	153
7.2.2 89C51 串行口的应用	164
7.3 89C51 与 89C51 点对点异步通信	174
7.3.1 通信协议	174
7.3.2 波特率设置	174
7.3.3 通信程序举例	176
7.4 89C51 与 PC 机间通信	180
7.4.1 单片机与 PC 机通信的接口电路	180
7.4.2 PC 机通信软件设计	182
7.4.3 89C51 通信软件设计	187
7.5 思考题与习题	192

第8章 单片机小系统及外扩展

8.1 串行扩展总线接口技术	194
8.1.1 SPI 串行外设接口总线	194
8.1.2 I ² C 总线	199
8.1.3 单总线	214
8.2 并行扩展三总线的产生	220
8.2.1 片外三总线结构	221
8.2.2 系统扩展的实现	221
8.3 扩展数据存储器	222
8.3.1 常用的数据存储器芯片	222
8.3.2 访问片外 RAM 的操作时序	224
8.3.3 89C51 扩展 2 KB RAM	226
8.4 思考题与习题	226

第 9 章 应用系统配置及接口技术

9.1 人-机通道配置与接口技术	229
9.1.1 键盘接口及处理程序	229
9.1.2 LED 显示器接口及显示程序	240
9.1.3 串行口控制的键盘/LED 显示器接口电路	244
9.2 前向通道中的 A/D 转换器及接口技术	247
9.2.1 单通道串行输出 A/D 芯片 TLC1549 及接口	248
9.2.2 多通道串行输出 A/D 芯片 TLC2543 及接口	250
9.2.3 逐次逼近型并行输出 A/D 转换器及接口	259
9.2.4 $3\frac{1}{2}$ 位双积分 A/D 转换器及接口技术	263
9.3 系统后向通道配置及接口技术	266
9.3.1 后向通道中的功率开关器件及接口	266
9.3.2 串行输入 D/A 芯片 TLC5615 接口技术	269
9.3.3 并行输入 D/A 芯片及接口技术	273
9.4 思考题与习题	277

第 10 章 系统实用程序

10.1 主程序和子程序的概念	279
10.1.1 主程序	279
10.1.2 子程序及参数传递	279
10.1.3 中断服务子程序	282
10.2 数据采集及简单控制程序	282
10.2.1 $3\frac{1}{2}$ 位数据采集程序	282
10.2.2 航标灯控制程序	284
10.2.3 水位控制程序	286
10.2.4 蜂鸣音报警子程序	289
10.3 数据处理程序	290
10.3.1 排序程序	290
10.3.2 数字滤波程序	291
10.3.3 标度变换(工程量变换)	294
10.4 代码转换程序	296

10.5	软件抗干扰·····	298
10.5.1	软件陷阱技术·····	299
10.5.2	软件看门狗·····	301
10.5.3	硬件(专用芯片)看门狗·····	303
10.6	最短程序·····	306
附录 A 89C51 指令表		
附录 B 89C51 指令矩阵(汇编/反汇编表)		
附录 C 8255A 可编程外围并行接口芯片及接口		
参考文献		

第 1 章 微机基础知识

1.1 微处理器、微机和单片机的概念

首先介绍一下微处理器(Microprocessor, 简称 μP)、微型计算机(Microcomputer, 简称微机, μC)和单片机(Single-Chip Microcomputer)的概念。

微处理器(芯片)本身不是计算机,但它是小型计算机或微型计算机的控制和处理部分。

微机则是具有完整运算及控制功能的计算机,除了包括微处理器(作为它的中央处理单元 CPU——Control Processing Unit)外,还包括存储器、接口适配器(即输入/输出接口电路)以及输入/输出(I/O)设备等。图 1-1 所示为微机的各组成部分。其中,微处理器由控制器、运算器和若干个寄存器组成;I/O 设备与微处理器的连接需要通过接口适配器(即 I/O 接口);存储器是指微机内部的存储器(RAM、ROM 和 EPROM 等芯片)。

将微处理器、一定容量的 RAM 和 ROM 以及 I/O 口、定时器等电路集成在一块芯片上,构成单片微型计算机,简称单片机。

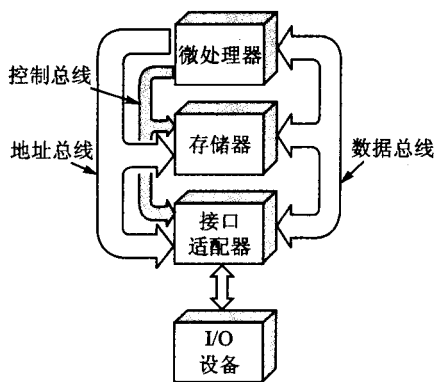


图 1-1 微机的组成

1.1.1 微处理器(机)的组成

微处理器包括两个主要部分: 运算器和控制器。

图 1-2 所示是一个较详细的由微处理器、存储器和 I/O 接口组成的计算机模型。为了简化问题,在 CPU 中只画出了主要的寄存器和控制电路,并且假设所有的计数器、寄存器和总线都是 8 位宽度,即要求多数主要寄存器和存储器能保存 8 位(bit)数据,传送数据的总线由 8 根并行导线组成。

在计算机术语中,数据单元是一组二进制数,是计算机中使用的基本信息单元。它可以作为数据,也可以是计算机完成某操作的一条指令码,还可以是 ASCII 码字符等。

在 8 位微处理器中,数据单元由 1 字节(Byte)组成;在 16 位机中,数据单元由 2 字节组

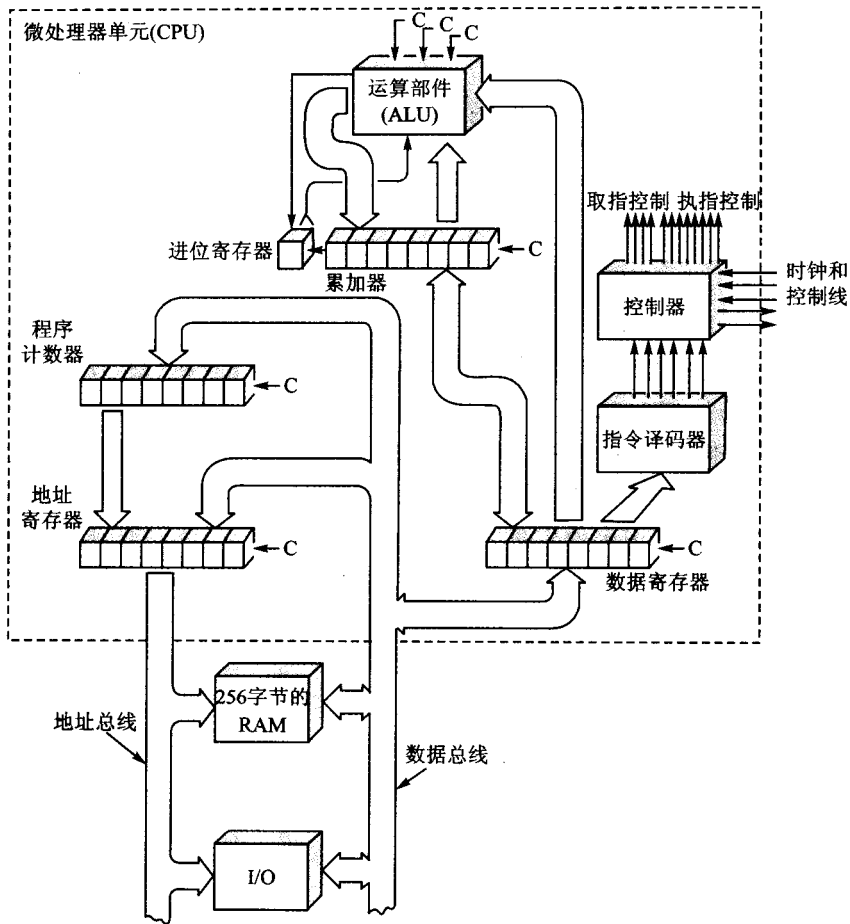


图 1-2 一个计算机模型

成。图 1-3 表示了组成计算机数据单元的位数。

1. 运算器

运算器由运算部件——算术逻辑单元 (Arithmetic & Logical Unit, 简称 ALU)、累加器和寄存器等几部分组成。ALU 的作用是把传送到微处理器的数据进行算术或逻辑运算。ALU 具有两个主要的输入来源：一个来自累加器，另一个来自数据寄存器。ALU 能够完成这两个输入数据的相加或相减运算，也能够完成某些逻辑运算。ALU 执行不同的运算操作是由不同控制线上的信号 (在图 1-2 方框图上的标志为 C) 所确定的。

通常，ALU 接收来自累加器和数据寄存器的两个 8 位二进制数。因为要对这些数据进行某些操作，所以将这两个输入的数据均称为操作数。

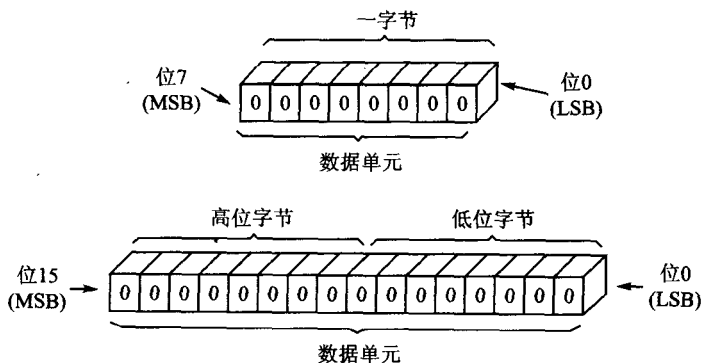


图 1-3 计算机中的数据单元

ALU 可对两个操作数进行加、减、与、或和比较大小等操作,最后将结果存入累加器。例如,两个数 7 和 9 相加,在相加之前,操作数 9 放在累加器中,7 放在数据寄存器中,执行两数相加运算的控制线发出“加”操作信号,ALU 即把两个数相加,并把所得结果 16 存入累加器,取代累加器原来存放的数 9。总之,运算器有两个主要功能:

- 执行各种算术运算;
- 执行各种逻辑运算,并进行逻辑测试,如零值测试或两个值的比较。

通常,一个算术操作产生一个运算结果,而一个逻辑操作产生一个判决。

2. 控制器

控制器由程序计数器、指令寄存器、指令译码器、时序发生器和操作控制器等组成,是发布命令的“决策机构”,即协调和指挥整个计算机系统的操作。控制器的主要功能有:

- 从内存中取出一条指令,并指出下一条指令在内存中的位置;
- 对指令进行译码或测试,并产生相应的操作控制信号,以便执行规定的动作,比如一次内存读/写操作、一个算术/逻辑运算操作或一个输入/输出操作等;
- 指挥并控制 CPU、内存和输入/输出设备之间数据流动的方向。

相对控制器而言,运算器接收控制器的命令而进行动作,即运算器所执行的全部操作都是由控制器发出的控制信号来指挥的。

ALU、计数器、寄存器和控制器除在微处理器内通过内部总线相互联系外,还通过外部总线与外部的存储器和输入/输出接口电路联系。外部总线一般分为数据总线 DB、地址总线 AB 和控制总线 CB,统称为系统总线。存储器包括 RAM 和 ROM。微型计算机通过输入/输出接口电路可与各种外围设备连接。

3. CPU 中的主要寄存器

1) 累加器(A)

累加器是微处理器中最忙碌的寄存器。在算术和逻辑运算时,它具有双重功能:运算前,

用于保存一个操作数;运算后,用于保存所得的和、差或逻辑运算结果。

2) 数据寄存器(DR)

数据(缓冲)寄存器是通过数据总线向存储器和输入/输出设备送(写)或取(读)数据的暂存单元。它可以保存一条正在译码的指令,也可以保存正在送往存储器中存储的一个数据字节等等。

3) 指令寄存器(IR)及指令译码器(ID)

指令寄存器用来保存当前正在执行的一条指令。当执行一条指令时,先把它从内存取到数据寄存器中,然后再传送到指令寄存器(图中未画出)。指令分为操作码和地址码字段,由二进制数字组成。为执行给定的指令,必须对操作码进行译码,以便确定所要求的操作。指令译码器就是负责这项工作的。指令寄存器中操作码字段的输出就是指令译码器的输入。操作码一经译码后,即可向操作控制器发出具体操作的特定信号。

4) 程序计数器(PC)

为了保证程序能够连续地执行下去,CPU必须采取某些手段来确定下一条指令的地址。程序计数器正是起到了这种作用,所以通常又称其为指令地址计数器。在程序开始执行前,必须将其起始地址,即程序第一条指令所在的内存单元地址送入PC;当执行指令时,CPU将自动修改PC的内容,使之总是指示出将要执行的下一条指令的地址。由于大多数指令都是按顺序执行的,所以修改的过程通常只是简单的加1操作。

5) 地址寄存器(AR)

地址寄存器用于保存当前CPU所要访问的内存单元或I/O设备的地址。由于内存和CPU之间存在着速度上的差别,所以必须使用地址寄存器来保持地址信息,直到内存读/写操作完成为止。

显而易见,当CPU和内存进行信息交换(即CPU向/从存储器存/取数据或者CPU从内存读出指令)时,都要使用地址寄存器和数据寄存器。同样,如果把外围设备的地址作为内存地址单元来看待的话,那么,当CPU和外围设备交换信息时,也需要使用地址寄存器和数据寄存器。

1.1.2 存储器和输入/输出接口

1. 存储器

如图1-4所示,假设某台微型计算机使用256字节的8位随机存储器(RAM)与CPU交换数据,经常把这种规格的存储器称作 256×8 位读/写存储器。

两根8位总线和若干控制线把存储器和微处理器(机)连接起来。地址总线将一组8位二进制数(能表示256个单元)从CPU送到存储器的地址译码器。每个存储单元被赋予一个唯一的地址,规定第一单元地址为0,最后一单元的地址为255(用二进制表示为11111111B,用十六进制表示为FFH)。在地址总线上,通过8位地址线选择指定的单元。地址译码器的输

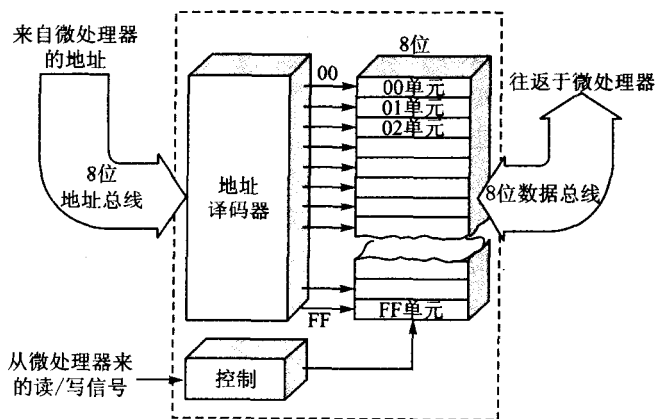


图 1-4 随机存取存储器

出可以惟一确定被选择的存储单元。

存储器还从 CPU 接收控制信号,从而确定存储器执行何种操作。“读”信号表明要读出被选单元的内容,并将数据放到数据总线上,由总线送到 CPU。“写”信号表明要把数据总线上的数据写入指定的存储单元中。

2. I/O 接口及外设

从图 1-2 可以看到,I/O 接口与地址总线、数据总线的连接同存储器一样,而每个外部设备与微处理器的连接必须经过接口适配器(I/O 接口)。每个 I/O 接口及其对应的外部设备都有一个固定的地址,在 CPU 的控制下实现对外部设备的输入(读)和输出(写)操作。

1.2 常用数制和编码

计算机是用于处理数字信息的,单片机也是如此。各种数据及非数据信息在进入计算机前必须转换成二进制数或二进制编码。下面介绍计算机中常用数制和编码以及数据在计算机中的表示方法。

1.2.1 数制及数制间转换

1. 数制——计数的进位制

单片机中常用的有 3 种数制:二进制、十进制和十六进制。其中只有二进制数是计算机能直接处理的;但是二进制数表达过于繁杂,所以引入十六进制数;十进制是人们最熟悉的数制。这 3 种数制在单片机中都是经常使用的。

1) 十进制 (Decimal, 用 D 表示)

大约在公元 400 年左右,印度数学家首先发明了用十进制计数,这可能是由于人有十个手指和十个脚指的缘故吧。约在公元 800 年,阿拉伯人开始使用它,所以又称它为阿拉伯数制,以后传到了欧洲,才被命名为“十进制数制”。

十进制用 0,1,2,3,4,5,6,7,8,9 十个数字来表示数。十进制数的基数是 10,当计数时,每一位计到十就往上进一位,也就是逢十进一;或者说,上一位的数是下一位的十倍。

如果用 α 表示任何一个十进制数字,那么一个含有 n 位整数、 m 位小数的十进制数的通用表示式是:

$$N = \alpha_{n-1} \times 10^{n-1} + \alpha_{n-2} \times 10^{n-2} + \cdots + \alpha_0 \times 10^0 + \alpha_{-1} \times 10^{-1} + \cdots + \alpha_{-m} \times 10^{-m}$$

或写成:

$$N = \sum_{i=-m}^{n-1} \alpha_i \times 10^i$$

十进制是人们习惯的数制,但不是惟一的数制,比如,还有二进制、八进制、十二进制、十六进制和六十进制等。

2) 二进制数 (Binary, 用 B 表示)

以 2 为基数的数制叫二进位计数制,计算机中采用的是二进制数。它只包括两个符号,即 0 和 1。在一个二进制数中,前一位的权是后一位的两倍,即逢二进一。对于整数,从右往左各位的权是 1,2,4,8,16,32,……;对于小数,从左往右各位的权是 $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \dots$ 。把十进制表示式中的 10 都换为 2 就得到二进制的表示式:

$$N = \alpha_{n-1} \times 2^{n-1} + \alpha_{n-2} \times 2^{n-2} + \cdots + \alpha_0 \times 2^0 + \alpha_{-1} \times 2^{-1} + \cdots + \alpha_{-(m-1)} \times 2^{-(m-1)} + \alpha_{-m} \times 2^{-m}$$

或简写成为:

$$N = \sum_{i=-m}^{n-1} \alpha_i \times 2^i$$

式中: α_i 是 0 或 1,具体取值由 N 决定。例如,

二进制数 10101101.1011 B(Binary)表示的十进制数值是:

$$1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} = 128 + 32 + 8 + 4 + 1 + 0.5 + 0.125 + 0.0625 = 173.6875$$

日常生活中人们习惯于十进位计数制,所以感到二进位计数制很不方便,那么电子计算机

为什么还采用二进制计数制呢？因为目前研究与应用最成熟的是具有两个稳定状态，且具有记忆功能的电子电路，使用二进制能够很方便又直观地表示出机器中双稳态电路的两个稳定，并可相互变换的物理状态；反过来，一个双稳态电路的0或1两个状态可以用来表示一位二进制数，几个电子器件就可以代表一组多位二进制数。

3) 十六进制数 (Hexadecimal, 用 H 表示)

尽管用二进制数表示计算机中的信息很方便，但为了便于书写和阅读，经常采用十六进制，即在计数时，逢十六进一。这样，书写的长度非常短，且可很方便地将十六进制数转换为二进制或将二进制转换为十六进制。大部分计算机所处理的数据位长都是4的整数倍(如4位、8位、16位、32位等)，所以计算机经常采用十六进制。它有以下3个基本特征：

- 具有十六个数字符号：0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F；
- 逢十六进一；
- 一位十六进制数可用4位二进制数表示，它们之间存在直接而又惟一对应的关系。

例如：0011B=3H, 1010B=AH。这样，一个位数较多的二进制数可以用位数较少的十六进制数来书写，既简单，又易于转换。再如：

$(0101101010110111)_2 = (5AB7)_{16} = 5AB7H$ (H表示十六进制数)。

若十六进制数最高位是A~F中的符号之一，则应在前边加0，说明是数字而不是文字，例如十六进制数A7CEH应写成0A7CEH。

其按权展开式为：

$$N = \sum_{-m}^{n-1} \alpha_i \times 16^i = \alpha_{n-1} \times 16^{n-1} + \alpha_{n-2} \times 16^{n-2} + \cdots + \alpha_0 \times 16^0 + \alpha_{-1} \times 16^{-1} + \alpha_{-2} \times 16^{-2} + \cdots + \alpha_{-m} \times 16^{-m}$$

式中， α_i 表示十六进制数的第*i*位，权为 16^i ， α_i 从0~9、A、B、C、D、E、F十六个数码中选用；*m*、*n*为正整数，*n*为小数点左边的位数，*m*为小数点右边的位数。例如，

$$A4B.CH = 10 \times 16^2 + 4 \times 16^1 + 11 \times 16^0 + 12 \times 16^{-1}$$

部分数的3种数制对照见表1-1。

2. 不同数制之间的转换

在使用计算机的过程中，经常需要在二进制、十进制和十六进制之间进行相互转换。

1) 二进制↔十进制

(1) 二进制→十进制

把二进制数转换为相应的十进制数，只要将二进制中出现1的数位权相加即可，整数和小数的位权如图1-5所示。