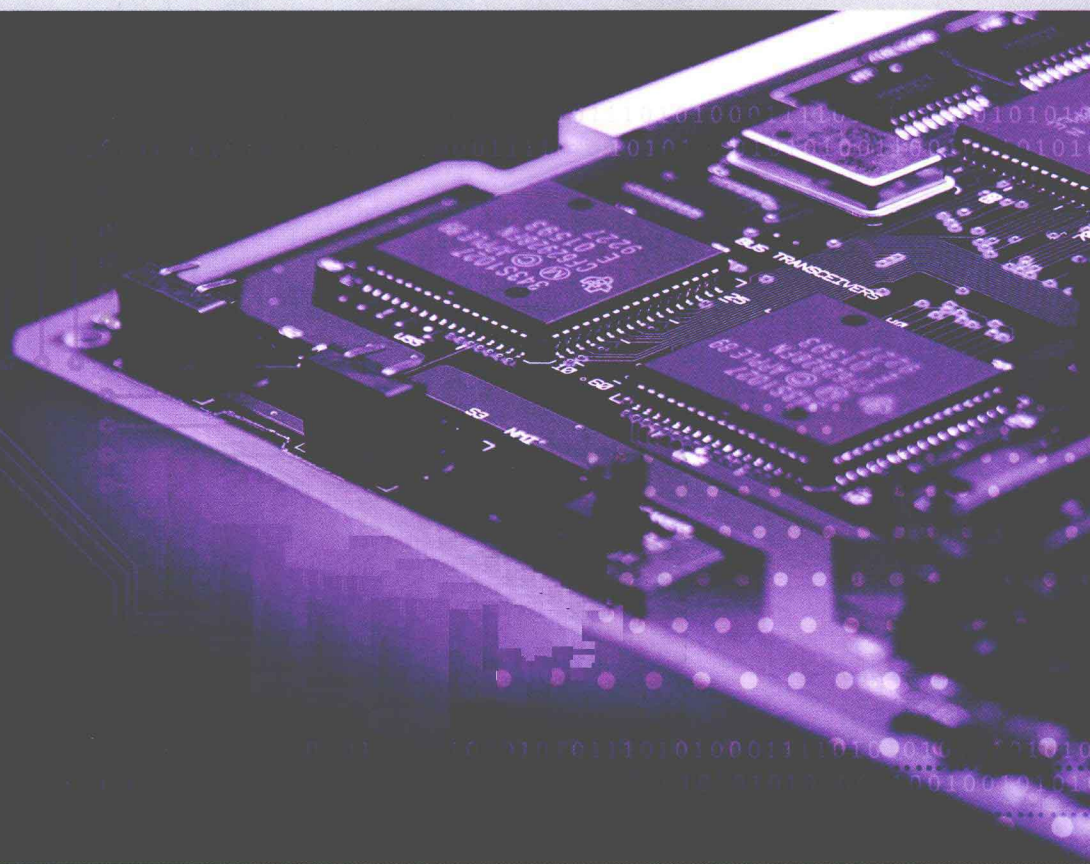


单片机测量与控制基础 实例教程

DANPIANJI CELIANG YU KONGZHIJICHU
SHILI JIAOCHENG



李忠国 蔡海云 / 主编 冯 骥 李 勃 / 副主编

世纪英才高等职业教育课改系列规划教材(机电类)

单片机测量与控制基础 实例教程

李忠国 蔡海云 主 编
冯 骥 李 勃 副主编

人民邮电出版社
北 京

图书在版编目(CIP)数据

单片机测量与控制基础实例教程 / 李忠国, 蔡海云
主编. — 北京: 人民邮电出版社, 2011.12
世纪英才高等职业教育课改系列规划教材. 机电类
ISBN 978-7-115-26062-8

I. ①单… II. ①李… ②蔡… III. ①单片微型计算
机—高等职业教育—教材 IV. ①TP368.1

中国版本图书馆CIP数据核字(2011)第156293号

内 容 提 要

本书借鉴“基于工作过程”的课程改革思想, 实现以学生为教学中心, 以实际工作任务为教学载体, 通过十几个实际单片机测量与控制系统中的分解任务学习单片机测控的基本知识和基本技能。从简单、直观的任务出发, 通过计算机仿真、实验板制作使学生在一个个小项目中边做边学。

将必须掌握的理论知识分解到各个小项目中, 摒弃了部分不易理解、不太常用的理论知识。配套的实验板可以使学生的学习环境接近于工作环境, 为学生从事单片机测控工作打下一个良好的基础。

本书可作为高职高专机电类各专业的教材, 还可供从事电子行业的工程技术人员阅读参考。

世纪英才高等职业教育课改系列规划教材(机电类)

单片机测量与控制基础实例教程

-
- ◆ 主 编 李忠国 蔡海云
副 主 编 冯 骥 李 勍
责任编辑 丁金炎
执行编辑 王小娟
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
大厂聚鑫印刷有限责任公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 18.25
字数: 458千字 2011年12月第1版
印数: 1—3000册 2011年12月河北第1次印刷

ISBN 978-7-115-26062-8

定价: 35.00元

读者服务热线: (010)67132746 印装质量热线: (010)67129223

反盗版热线: (010)67171154

广告经营许可证: 京崇工商广字第0021号

单片机测量与控制是目前电气控制系统中常用的控制方式,越来越多地被应用于各种控制系统之中。《单片机测量与控制基础实例教程》是高职电气信息类专业必不可少的专业课程。目前高职高专学生的基础较薄弱,为了适应目前的学生基础状况和满足教学目标的要求,本教材突出以下几方面的特色。

1. 重点突出:根据单片机测量和控制应用的主要技术设计实例,做到实例有实用。
2. 难点分散:根据学生基础较差的特点,分散指令系统和程序结构内容,在实例中分散讲解。
3. 教法灵活:单片机课程的教学比较困难的问题是实验设备的问题,本书采用书本讲授、计算机仿真和实物实训三种模式同步教学,学生可根据自己的不同情况选择,以满足不同学校和读者的实际情况。
4. 因材施教:本教材选用 C51 作为开发语言,只介绍 C 语言中最基本的内容。由于教材只需要引导学生入门,部分难以理解的内容可以省略。这样提高了应用起点,同时也降低了学生的学习难度。
5. 实用性强:目前职业学校的教学中推行基于工作过程的教学模式,本教材也符合此教学思想。但是基于工作过程的教学模式并不一定需要完全围绕某一个具体产品来开展教学工作,而是教学内容要与生产过程密切结合,能做到学以致用、学有所用。本教材的所有实例均根据单片机测控中常见的单片机应用技术设计,做到教学内容与工作过程密切相关。

本书由武汉铁路技师学院李忠国、蔡海云、冯骥和中国建设银行信息技术管理部武汉开发中心李劼共同编写。其中蔡海云老师负责第一篇基础知识模块的编写;李劼负责第二篇项目一至项目五模块的编写;冯骥老师完成了各模块的程序调试、实验板设计和硬件调试工作。李忠国老师负责全书的组织和统稿。

由于作者水平有限,教材中难免出现错误或不妥之处,敬请各位读者给予批评指正。

编者

第一篇 基础知识

知识模块一 单片机简介	1
知识模块二 51 系列单片机	9
知识模块三 程序的编译和运行	31

第二篇 项目实训

项目一 发光二极管控制	53
项目二 开关及按键控制	75
项目三 数码管控制	96
项目四 LCD 显示	127
项目五 速度测量	154
项目六 A/D、D/A 转换	167
项目七 温度检测	189
项目八 步进电机控制	212
项目九 PWM 控制	226
项目十 时钟控制	244
项目十一 串口通信	269

第一篇 基础知识

知识模块一 单片机简介

单片机是微型计算机发展的一个分支，是一种专门面向控制的微处理器件，故又称之为微控制器（Micro Controller Unit, MCU）。单片机顾名思义就是做在一片集成块内的计算机。尽管只有一片集成块，但是它几乎包含一台计算机的所有部分。

一、计算机的硬件结构

当今计算机飞速发展，各种各样的计算机层出不穷。从大型机、巨型机到微型机、单片机，令人眼花缭乱。但是目前无论什么样的外观、大小及用途的计算机大都属于冯·诺依曼型计算机。也就是说，目前绝大部分电子计算机都采用了冯·诺依曼提出的电子计算机的体系结构。图 1-1-1 就是冯·诺依曼型计算机的硬件结构。

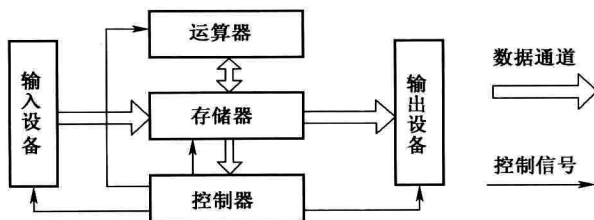


图 1-1-1 计算机基本组成

从图 1-1-1 可以看出一台计算机应当有运算器、控制器、存储器、输入设备和输出设备这 5 个部分。

1. 存储器

存储器主要用来保存程序和数据。无论是程序还是数据，在存储器中均以二进制数表示。这些二进制数如果代表的是程序或者是某些符号则称为二进制代码。例如，存储器中保存的文字、图像、声音信号等均称为二进制代码。保存在存储器中的程序则是由许许多多的二进制指令代码所组成。

8 位单片机中存储器采用 8 位二进制数为一个存取单位，每一位二进制数称为 1bit，而 8 位二进制数则组成一个字节，称为 1Byte。

存储器中能保存的二进制数的数量称为存储器的容量。其容量大小表示为：

$$8 \text{ 位 (bit)} = 1 \text{ 字节 (Byte)}$$

$$2^{10} \text{ 字节} = 1024 \text{ 字节} = 1\text{K 字节}$$

$$2^{10} \text{K 字节} = 1024\text{K 字节} = 1\text{M 字节}$$

例如，十进制数 78 保存在存储器中时若直接转换成二进制数保存为 01001110，若转换成 8421BCD 码保存时为 01111000。字母 A 保存在存储器中为 00100000。而加法指令保存在

存储器中则为 00100100。

单片机中的存储器主要采用半导体存储器，这些存储器分为两类。

(1) 只读存储器 (ROM)

这种存储器的内容由生产厂家存入，用户使用过程中只能读取其中内容而不能修改内容。它们主要用来存储程序和某些固定不变的数据，因此也称为程序存储器。断电后存储器中的内容保持不变，这种存储器又称为非易失性存储器。

(2) 随机存储器 (RAM)

这种存储器的内容由用户自己写入和读出，主要用来保存工作过程中的各种数据，因此它也称为数据存储器。但是存储器的内容会因为断电而丢失。这种存储器又称为易失性存储器。

计算机工作过程中经常会有一些在断电后也需要保存的用户数据，如用户的某些参数和设定值等。早期采用的方法是使用 RAM 加后备电池的方法，用户存入数据后如果断电则由后备电池向存储器供电，使存储器中的内容保持不变，以便下次工作时继续使用。近年来随着闪存的出现，单片机中也采用了闪存，这种存储器可以由用户写入数据，断电后其内容保持不变，方便用户保存此类数据。

2. 运算器

运算器主要用来完成算术运算和逻辑运算。计算机中的运算器只能完成非常简单的基本运算，而对于那些复杂的运算则需要通过程序将其变换为许多简单的运算来完成。例如，某计算机的运算器只能完成加、减法运算，在执行 $23 + 5$ 时可以直接使用加法指令实现，但是需要执行 23×5 时运算器就无能为力了，此时人们就将 23×5 转变为 $23 + 23 + 23 + 23 + 23$ 来实现。这是计算机编程实现复杂运算的一种基本思路。

3. 控制器

控制器是控制计算机中各个部件协同工作的部件。计算机工作过程就是执行程序中的指令的过程，这些指令首先送给控制器，由控制器根据指令的功能控制其他部件执行指令。控制器所能识别的指令就是计算机的指令集。51 系列单片机的指令集一共有 111 种指令。

在一般的微型计算机中，控制器和运算器一起集中做在一个集成块中，这个集成块就是人们常说的 CPU——中央处理器。它是一台计算机的核心部件，CPU 的性能直接决定了这台计算机的性能。

4. 输入设备

计算机在工作过程中经常需要从外部输入各种数据信息，输入设备可以将外界的信息转换为二进制代码并送到存储器中，普通计算机的输入设备有键盘、鼠标等设备。工业控制计算机的输入信息种类很多，如用户工作时输入的数据、系统设备的检测数据等。外界数据的种类很多，范围也很广，对应的输入设备也比较多。

例如，计算机系统的键盘、开关等，这种信息为二值信息。可以直接转换为二进制数据送到计算机中。设定某一位信号对应一个开关，当开关断开时该位为“0”，而开关接通时该位为“1”。计算机根据数据的值就可以知道开关的状态。

计算机的输入信息中许多信息并不是二值信息甚至是模拟量，这些信息经常需要经过 A/D 转换器转换为二进制数据再送到计算机中。有关 A/D 转换器的内容将在本书后续内容中介绍。

有不少外界信息甚至都不是电信号，或者是范围太大或太小的电信号，这样就需要传感器将这些信号转换为范围合适的电信号。例如，某温度控制系统需要检测温度，而温度并不

是一个电信号，这样就需要使用温度传感器将温度转换为电信号。又如某电力系统设备中需要检测高压电网的电压，尽管高压电网的电压是一种电信号，但是过高的电压是无法直接送到计算机中的，这样就需要使用电压传感器，将这种高电压转换为适合计算机处理的低电压，再经过 A/D 转换器转换为二进制数据就可以传送到计算机中了。

5. 输出设备

任何一台计算机都需要输出信息，普通计算机的输出设备有显示器、打印机等。工业控制用的计算机除了显示器和打印机等通用设备以外还需要输出对设备进行控制的信息。这些信息可以分为两大类：一类是对开关量设备的控制信息，如指示灯，继电器、半导体开关器件等设备；另一类是模拟量控制设备，计算机输出的二进制数据经过 D/A 变换器输出模拟信号来控制此类设备，如流量控制设备、角度控制设备等。

当计算机用于工业控制时，一般系统结构的框图如图 1-1-2 所示，用户键盘和生产设备的各种信息通过输入设备送到计算机的输入接口中，再经输入接口送到计算机的存储器中。这些信息经过计算机程序的处理后得到了所需要的控制信息，再经输出接口送到输出设备，由输出设备对生产设备进行控制。同时运行状态和参数也可通过显示器显示出来。当然这一切工作过程是在计算机内部的程序控制之下进行的。

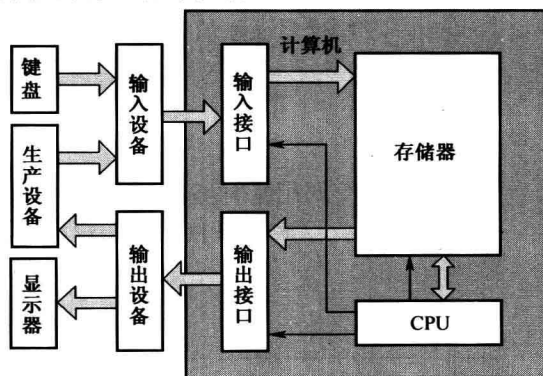


图 1-1-2 一般工业控制计算机系统的典型结构框图

二、计算机的软件结构

计算机的工作就像是一场文艺演出，计算机的硬件只是为演出提供了演出环境，如舞台、音响、灯光等。但是光有这个演出环境并不能完成文艺演出，文艺演出还需要各种节目，这些节目就是计算机软件。如果一台计算机只有硬件，那么这台计算机性能再好也只是一堆废铁。而计算机软件也需要相应的硬件支持，没有相应的硬件，再好的软件也无法工作。

1. 计算机的软件

计算机软件就是使计算机完成某项工作的程序及其与之相关的各种文档。例如，人们上街买一套计算机游戏软件，它包含有游戏程序、使用说明、版权证书、硬件需求说明等资料，其中最重要的就是程序，因此人们往往简单地将软件理解为程序。

2. 计算机的程序

计算机的程序就是使计算机完成某一特定工作的指令集合。也就是说计算机程序是由许许多多的指令组成的，每一条指令完成一个简单的操作，按顺序执行这些指令就能完成我们所需要执行的工作。例如以下为单片机中的一个简单程序，程序中每一行就是一条语句，计

计算机工作时按顺序执行这些语句，完成程序编制者安排的工作。

下面为单片机教学中的一个简单流水灯控制的 C 语言程序清单。计算机工作时从第一行语句开始执行，计算机的工作就是按顺序执行这个程序中的一条条语句。

```

/* -----
简单流水灯控制程序
----- */
#include <REG52. H>           //指定头文件 REG52. H
#include <INTRINS. H>         //指定头文件 INTRINS. H
#define uint unsigned int
void delay( uint ) ;          // 函数说明
/* -----
主函数
----- */
void main (void) {
uint s ;
    s = 10 ;                  //给变量 s 赋值 10
    P0 = 7FH ;
while (1)
{
    delay(s) ;                //调用延时函数,延时 1s
    P0 = _crol_(P0,1) ;       //寄存器 P0 中内容左移一位并送回 P0 中
}
}
/* -----
延时函数,调用时提供参数 x,延时时间为 x × 100ms
----- */
void delay( uint x)
{
uint n,m;
    for (n=0; n< x;n++ )
        for(m=0;m<20000;m++ ) ;
}

```

3. 计算机的指令系统

一般计算机的程序都有数千条以上的指令，大型的程序甚至有几万甚至几十万条以上的指令。一个程序的指令虽然很多，但是指令的种类并不多。就像写文章，一篇小说几十万字很常见，但是所用的汉字也只需要两三千个一样。学习单片机编程就是要学会单片机有哪些指令，它们分别具备哪些功能，然后学习用这些指令来实现某一个控制功能。51 机汇编语言程序的指令一共只有 111 条，学习汇编语言首先就要学习这些指令的格式和它们的功能，就像学文化要先学认字一样。

计算机中的指令都是存放在存储器中的二进制代码，这种用二进制代码表示的程序称为机器

语言程序。这种二进制代码既难看也难懂，特别容易出错。例如，某程序的机器语言代码如下。

```
01110100
00000001
11110101
10100000
01111111
11111111
01111110
11111111
11011110
11111110
11011111
11111010
00100011
10000000
11110011
```

这里我们无法直接看懂上述程序的作用和功能。

为了方便使用，人们将每一种机器语言指令使用为一个英语缩写单词来代替它，这个缩写单词称为助记符，利用助记符人们就比较方便地看懂和记忆指令。用这种指令编写的程序称为汇编语言程序，也称为汇编语言源程序。下面一段程序就是上述程序的汇编语言源程序，一般稍加学习就能看懂这段程序的功能。

```
ORG      0000H      ;程序从 0 地址开始
START: MOV      A,#0FEH      ;让 A 的内容为 11111110
LOOP:  MOV      P2,A      ;让 P2 口输出 A 的内容
      RL        A      ;让 A 的内容左移
      CALL     DELAY      ;调用延时子程序
      LJMP     LOOP      ;跳到 LOOP 处执行
;0.1 秒延时子程序(12MHz 晶振) = = = = =
DELAY: MOV      R7,#200      ;R7 寄存器加载 200
D1:    MOV      R6,#250      ;R6 寄存器加载 250
      DJNZ     R6,$      ;本行执行 R6 次
      DJNZ     R7,D1      ;D1 循环执行 R7 次
      RET        ;返回主程序
```

由于计算机中只能执行机器语言程序，这种汇编语言程序需要将助记符还原为机器指令才能保存到单片机的程序存储器中，这一过程称为编译。如果人工将汇编语言程序翻译成机器语言程序，这一工作称为手工汇编。手工汇编既复杂又容易出错，现在人们都采用计算机程序来实现这一翻译，称为机器汇编，实现这种编译的程序称为汇编程序。51 系列单片机常用的汇编程序为 ASM51.exe 或 A51.exe。

注意汇编语言源程序与汇编程序的区别：汇编语言源程序是用户使用汇编语言编写的程

序，而汇编程序是将汇编语言源程序翻译为机器语言程序的工具软件。

为了编制较大型的程序，人们往往将程序分解为若干个模块，由多人或多个小组分别完成单独编译，然后再合成到一起生成最终的机器语言程序。这样汇编程序不是直接将源程序编译成机器语言程序，而是先产生一个中间文件，这个中间文件称为目标文件，一般为 *.obj。每一个模块分别编译为一个目标文件，然后再使用一个链接程序将所有的目标文件链接成最终的机器语言程序。这个链接程序一般为 RL51.exe 或 L51.exe。

4. 低级语言与高级语言

使用汇编语言编写的程序尽管比使用机器语言要好读好懂一些，但是汇编语言指令与机器语言指令是一一对应的关系，也就是说有多少种机器语言指令就有多少条汇编语言指令。这种指令系统属于面向机器的指令系统，这种语言称为低级语言。这种低级语言与人们习惯的自然语言有很大的差距，例如，需要做一个 $23 + 5$ 的加法，并将结果保存在变量 JG 中，使用汇编语言需要如下指令完成操作：

```
MOV A, #23
```

```
ADD A, #5
```

```
MOV JG, A
```

初学者是看不懂以上指令的，这就是汇编语言使用比较困难的原因之一。为了克服这一困难人们研究出了一些比较符合自然语言的计算机语言，常用的有 C 语言等。使用 C 语言完成上述操作时的指令为：

```
JG = 23 + 5;
```

显然这一指令我们都能看懂，因为它比较符合人们的自然习惯。这种接近人们自然语言的计算机语言称为高级语言，使用高级语言编写的源程序需要专门的编译软件将高级语言源程序翻译成目标文件，如 51 系列单片机用的 C 语言编译软件一般为 C51.exe。

C 语言中的一行称为一个语句，经过编译软件翻译后可能会生成多条指令，这些指令一起来完成这个语句的功能。

使用汇编语言或 C 语言编写单片机程序时，工作过程如图 1-1-3 所示。首先使用普通的编辑软件编写源程序，使用汇编语言时源程序文件扩展名为 .ASM 或 .A51，而使用 C 语言时扩展名为 .C 或 .C51。源程序编写完成后使用编译程序将源程序编译为目标文件 .obj，再使用连接程序将目标文件连接成机器语言程序文件。这样就完成了源程序到机器语言程序的转换，然后使用专用工具将机器语言程序传送到单片机的存储器中，交由单片机执行。在这个过程中会产生一些辅助文件用于调试和检查程序。

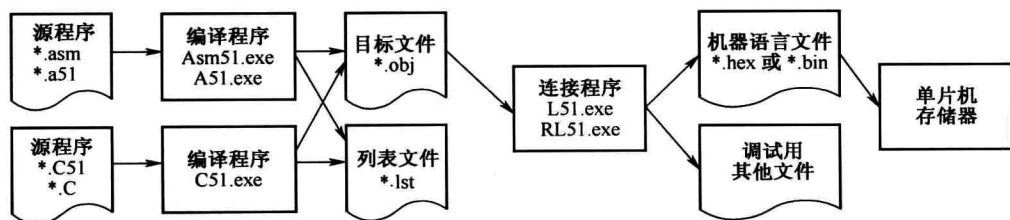


图 1-1-3 单片机软件编译过程

目前有许多单片机的开发环境将编译、连接过程安排到一起，使用时只需要下达一个命令就能直接将源程序编译为机器语言程序，方便了开发者的操作。对单片机的源程序编辑、编

译、连接以及单片机程序测试和程序调试、运行控制等功能集中在一个软件中，这个软件称为集成开发环境（IDE）。目前有多个公司的 IDE 软件可供用户使用，大家可根据自己的条件选用。

三、计算机的工作过程

计算机工作前需要将机器语言程序存放到存储器中，存储器中存放指令的区域称为程序存储器，每一条指令存放的位置都对应有一个地址。表 1-1-1 表示了一个简单程序存放在存储器中的情况，从表中可以看到以下两个规律。

① 此程序一共有 8 条指令，其中有些指令占用了两个字节，有些指令只占了一个字节。在单片机中一条指令需要占用 1~3 个字节。这些指令可分别称为单字节指令、双字节指令和三字节指令。

② 第一条指令存放在存储器地址为 0 的位置，后续指令顺序存放。

表 1-1-1

程序存储器示意

存储器地址	机器指令		对应的汇编指令
	十六进制	二进制	
0000H 0001H	74H 01H	01110100 00000001	MOV A, #01
0002H 0003H	F5H A0H	11110101 10100000	L1: MOV P2, A
0004H 0005H	7FH FFH	01111111 11111111	MOV R7, #0FFH ^①
0006H 0007H	7EH FFH	01111110 11111111	L3: MOV R6, #0FFH
0008H 0009H	DEH FEH	11011110 11111110	L2: DJNZ R6, L2
000AH 000BH	DFH FAH	11011111 11111010	DJNZ R7, L3
000CH	23H	00100011	RL
000DH 000EH	80H F3H	10000000 11110011	SJMP L1

注：①汇编语言中十六进制数的表示方法为在十六进制数后加 H，如十六进制数 3A 表示为 3AH。若十六进制数第一位为字母时需在前面加一个“0”，如十六进制 A3 表示为 0A3H。

计算机的 CPU 中有一个程序计数器（PC），它里面保存着 CPU 下一步需要执行的指令的地址，CPU 每执行一条指令后 PC 自动加 1（执行双字节指令时加 2，执行三字节指令时加 3），使它指向下一条指令地址。这里“指向”的概念就是 PC 中保存着一个数据，这个数据是下一条指令的地址，也就说它指向下一条指令，这一过程可用图 1-1-4 表示。

计算机工作时首先给 CPU 一个启动信号，这个信号称为复位信号，复位信号使 PC 的内容清 0，然后 CPU 从 PC 指向的程序存储器中读出指令，交给控制器分析，根据指令内容控制存储器、运算器等其他设备执行这一条指令。由于复位后 PC 的内容为 0，故计算机启动

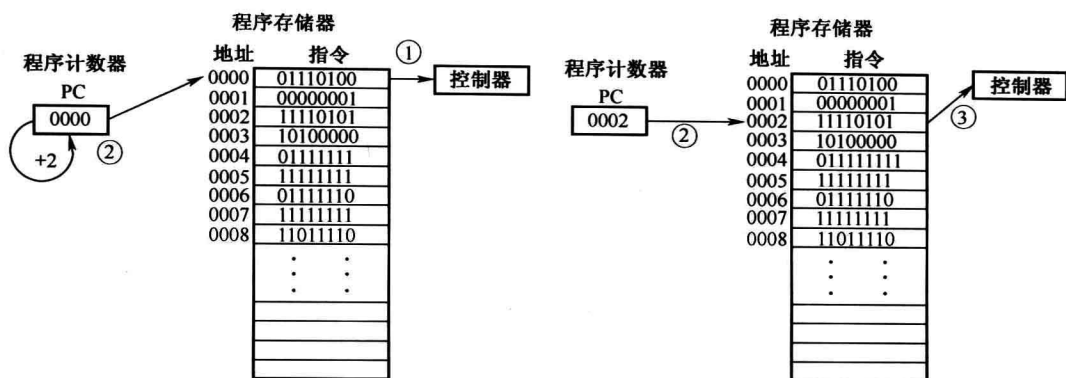


图 1-1-4 程序计数器的作用

后执行的第一条指令应当放在地址为 0 的程序存储器中。本条指令执行完后再从 PC 所指出的程序存储器中读下一条指令，如此不断地重复。这就是计算机的工作过程，如图 1-1-5 所示。



图 1-1-5 计算机工作过程

练习

1. 名词解释

ROM RAM 程序存储器 数据存储器 机器语言 汇编语言 低级语言
高级语言 源程序 目标文件 机器语言文件 编译 连接 IDE 环境
软件 硬件 程序 指令

2. 下表为存储器容量与地址范围的对应关系，填写表中括号内的数据。

容量大小	地址范围	容量大小	地址范围
4KB	0000H ~ 0FFFH	128B	000 ~ 07FH
2KB	0000H ~ ()	256B	000 ~ ()
()	0000H ~ 03FFH	()	000 ~ 1FFH

3. 将汇编语言源程序转换成机器语言程序时需要经过哪几个步骤，需要使用哪几个工具程序？

4. 将 C 语言源程序转换成机器语言程序时需要经过哪几个步骤，需要使用哪几个工具程序？

5. 为什么 MCS51 机中程序的第一条指令必须放在地址为 0000H 的程序存储器中？

知识模块二 51 系列单片机

常见的 8 位单片机主要包含有 51 系列、AVR 系列和 PIC 系列三大系列。其中 51 系列是一种比较典型的单片机，特别适合于初学者学习单片机。51 系列单片机是 Intel 公司于 1980 年开始推出的单片机系列，30 多年来 51 系列单片机的功能也不断地发展，市场上有许多 51 系列的简化或扩充版本的器件，是单片机中的主流机型。

51 系列单片机的基本品种见表 1-2-1。它们之间的主要区别在于制造工艺和片内存储器容量。除此之外，许多公司还有与 51 系列兼容的单片机系列，它们在功能上也有许多扩充，但基本结构都相同。

表 1-2-1

51 系列单片机主要型号及其区别

型号	工艺类型	内部程序存储器		片内数据存储器	在线编程 ISP
		类型	容量		
8031	HMOS		0	128B	不支持
8051	HMOS	PROM	4KB	128B	
8751	HMOS	EPROM	4KB	256B	
80C31	CHMOS		0	128B	
80C51	CHMOS	EEPROM	4KB	128B	
80C52	CHMOS	EEPROM	8KB	256B	
89C51	CHMOS	Flash	4KB	128B	
89C52	CHMOS	Flash	8KB	256B	
89S51	CHMOS	Flash	4KB	128B	支持
89S52	CHMOS	Flash	8KB	256B	
89S53	CHMOS	Flash	12KB	256B	

表中几个术语的说明如下。

① PROM：一种只读存储器的类型，它的内容只能由工厂写入，用户只能读出。

② EPROM：紫外线可擦除只读存储器，用户可以使用紫外线擦除其中的内容，使用专用的编程器写入内容。

③ EEPROM：电可擦除只读存储器，用户可以写入内容，并可以使用电信号擦除内容。

④ Flash：闪速存储器，用户可写也可擦除的只读存储器，目前的 U 盘均使用此种存储器。

⑤ ISP：在线可编程。将程序写入单片机的一种方法，在 ISP 技术出现以前必须将单片机放在专用的编程器上才能将程序写入单片机。ISP 技术出现后可以在单片机安装到用户板上后使用专用的电缆将程序写入单片机中。

本教材以市面上最普及的 AT89S52 为例，学习单片机的基础知识。

一、MCS51 机的硬件组成

MCS51 机的硬件结构如图 1-2-1 所示，除了普通计算机的基本部件外，单片机中还包

含有定时/计数器 CTC、串行接口、中断控制电路、振荡电路等部件，它们的功能及用法将在后续课程中陆续介绍。

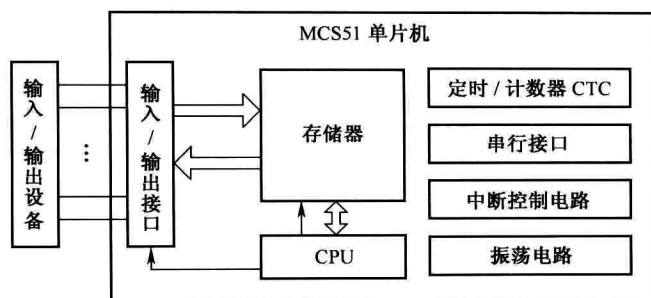


图 1-2-1 MCS51 机的硬件结构

1. MCS51 机的外部端口

(1) 89S51

① 89S51 外观，如图 1-2-2 所示。

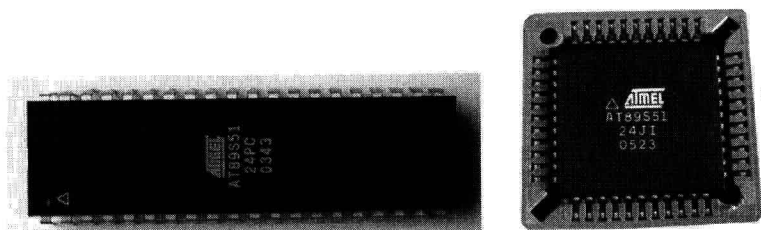


图 1-2-2 89S51

② 89S51 型号的含义，如图 1-2-3 所示。

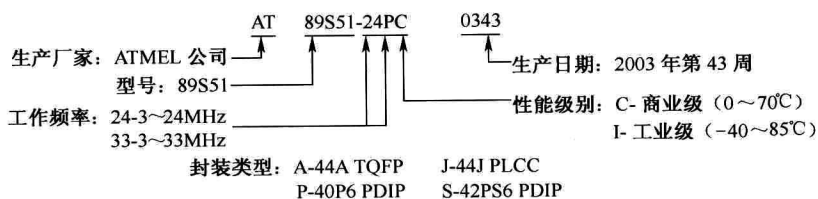


图 1-2-3 89S51 的型号

③ 89S51 的封装，如图 1-2-4 所示。

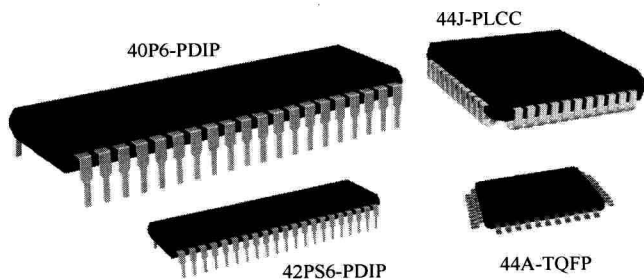


图 1-2-4 89S51 的封装

89S51 的外部引脚随封装不同而有所不同。从图 1-2-4 可以看到, 外部引脚有 40、42、44 3 种。实际使用的只有 40 个引脚。本教材中均以 40 引脚的 40P6 - PDIP 封装为例介绍 89S51 的使用方法。

(2) 89S51 的 I/O 口

PDIP 和 PLCC 封装的 89S51 的外部引脚及其名称如图 1-2-5 所示。

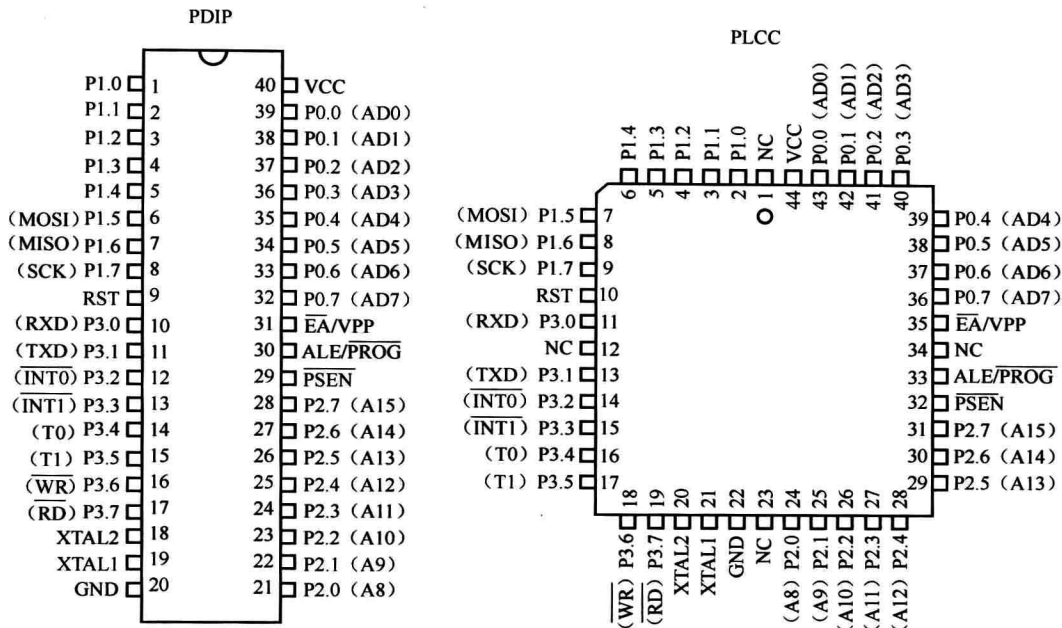


图 1-2-5 89S51 的引脚

① 89S51 的 I/O 口的功能。

89S51 共有 40 个外部引脚, 提供给用户使用的有 32 个, 分为 P0、P1、P2、P3 四组, 每组 8 位。其中 P0、P2、P3 都有两个功能, 一个是用于输入/输出端, 称为 I/O 口; 另一个为特殊功能, 称为第二功能。其第二功能的作用将在后续内容中介绍, 如表 1-2-2 所示。

② 外部端口的驱动能力。

a. 89S51 直接驱动负载时每个端口可驱动的最大灌电流负载 (I_{OL}) 为 10mA; 每组端口 8 个引脚的总灌电流负载驱动能力 P0 口为 26mA, P1 ~ P3 为 15mA; 4 组 (P0、P1、P2、P3) 端口 32 个引脚的总灌电流负载驱动能力为 71mA。

b. 89S51 驱动其他器件时, P0 口可驱动 8 个 LS TTL 负载, 其他端口可驱动 4 个 LS TTL 负载。

③ 外部端口使用注意事项。

外部端口用做通用 I/O 口时, 应当注意以下问题。

表 1-2-2

89S51 的端口

符号	PDIP 引脚位置	第一功能		第二功能	
		符号	功能	符号	功能
P0	39 ~ 32	P0.0 ~ P0.7	通用 I/O 口	AD0 ~ AD7	地址/数据总线 (低位)
P1	1 ~ 8	P1.0 ~ P1.7	通用 I/O 口		

续表

符号	PDIP 引脚位置	第一功能		第二功能	
		符号	功能	符号	功能
P2	21 ~ 28	P2.0 ~ P2.7	通用 I/O 口	A8 ~ A15	地址总线 (高位)
P3	10	P3.0	通用 I/O 口	RXD	串行通信接收口
	11	P3.1		TXD	串行通信发送口
	12	P3.2		INT0	外部中断 0
	13	P3.3		INT1	外部中断 1
	14	P3.4		T0	计数器 0 输入端口
	15	P3.5		T1	计数器 1 输入端口
	16	P3.6		WR	外部存储器写使能
	17	P3.7		RD	外部存储器读使能

a. 由于 P0 口用做 I/O 口时为 OC 输出, 输出 “1” 时实际为开路状态, 如果需要驱动 CMOS 或 TTL 器件时必须接上拉电阻, 如图 1-2-6 所示。

b. 端口用做输入时, 必须先向端口写 “1”。

c. 系统复位后所有外部端口状态为 “1”。

2. MCS51 单片机的存储器

存储器是计算机中十分重要的部件, 计算机中的程序和运行过程中的各种数据都存放在存储器中。计算机存储器只能存储二进制数据, 存储在计算机中的任何信息都必须转换为二进制数据。

(1) 存储器的单位长度

MCS51 机存储器的单位长度为 8 位, 即每一单位存储器可以储存 8 位二进制数, 称为一个字节。每次从存储器中读出数据, 或将数据保存到存储器中都是以 8 位二进制数进行的。

例如, 指令 MOV 20H, #7AH 就是将数据 7AH 保存到数据存储器中地址为 20H 的字节单元中。

为了方便使用, MCS51 机中还有部分可以每次读、写 1 位二进制数的位寻址单元和每次读、写 16 位二进制数的双字节存储器。

“寻址”的概念: “寻址”从字面上看就是寻找地址, 在将数据保存到存储器前或从存储器中读出数据前都必须先找到该存储单元的位置, 找到存储器单元的位置的过程就称为“寻址”, 找到存储器单元的方法也就称为“寻址方式”。

例如, 存储器 P0 为一个 8 位存储单元, 其中每一位又可进行位寻址, 如图 1-2-7 所示。

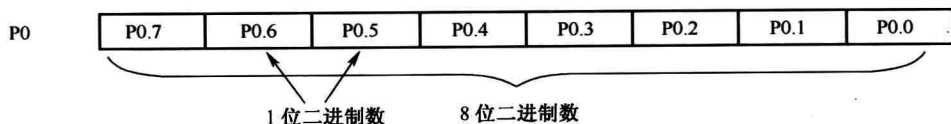


图 1-2-6 P0 口接上拉电阻

图 1-2-7 字节地址与位地址