

单片机原理与应用技术

理论课：54学时

实验课：18学时

主要参考书：

1 单片微型机原理与应用技术

魏立峰 王宝兴 北京大学出版社

2 AT89系列单片机原理与接口技术

王幸之 钟爱琴 北京航空航天大学出版社

3 MCS-51单片计算机应用系统设计

何立民 北京航空航天大学出版社

4 MCS-51/96系列单片机原理与应用

孙涵芳 北京航空航天大学出版社

第1章 基础知识

(课时： 6学时)

教学目的

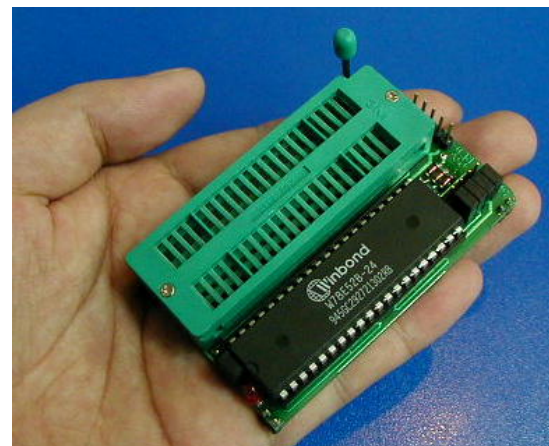
- 了解单片机的分类和发展历史。
- 了解单片机的特点及应用。
- 掌握计算机中常用的数制和码制。
- 掌握微型计算机的基本结构与基本原理。

学习重点和难点

- 不同数制之间的转换和有符号数的加减法运算。
- 微型计算机执行程序的过程。

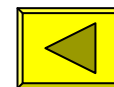
第1章 基础知识

- 1.1 嵌入式系统与单片机
- 1.2 单片机的应用
- 1.3 计算机运算基础
- 1.4 微型计算机的结构与原理
- 本章小结
- 习题



1.1 嵌入式系统与单片机

- 1.1.1 嵌入式系统的概念
- 1.1.2 单片机技术的发展
- 1.1.3 单片机的特点、分类及发展趋势



1.1 嵌入式系统与单片机

计算机发展简介

- 第一台电子计算机于**1946**年问世，发展速度迅猛。
- 计算机的发展日新月异，至今已经经历了由电子管计算机、晶体管计算机、集成电路计算机到大规模集成电路计算机四代。
- 单片机的出现是计算机技术发展史上的一个重要里程碑，它使计算机从海量数值计算进入到智能化控制领域。计算机技术逐步发展形成通用计算机系统和嵌入式计算机系统两大分支。

1.1.1 嵌入式系统的概念

基本概念

- 嵌入式计算机系统是面对测控对象，嵌入到应用系统中的计算机系统的统称，简称嵌入式系统(**Embedded System**)。
- 通用计算机系统主要满足海量、高速数值处理，兼顾控制功能；嵌入式计算机系统主要满足测控对象的控制功能，兼顾数值处理。

嵌入式计算机系统与通用计算机系统的主要区别

- 嵌入式系统最显著的特点是面对工控领域的测控对象。控制对象对嵌入式计算机系统采集、处理、控制的速度要求是有限的，而对控制方式与控制能力的要求是无限的。
- 为了实现海量高速数值计算，通用计算机系统对计算机运行速度的要求是无限的，而对计算机的控制功能的要求是有限的。
- 在数字信号处理(**DSP**)领域的嵌入式系统也要求高速处理能力，在多媒体技术的外设管理领域的通用计算机系统也要求良好的控制能力，但两者存在本质的差别。

嵌入式系统的分类

- 工业控制计算机。
- 通用**CPU**模块。
- 嵌入式微处理器(**Embedded Processor**)。
- 嵌入式微控制器(**Embedded Microcontrollers**)。
- 嵌入式 **DSP** 处理器 (**Embedded Digital Signal Processor, EDSP**)
- 嵌入式片上系统(**System On Chip, SOC**)

什么是嵌入式系统？

嵌入式系统一般指非 PC 系统，有计算机功能但又不称之为计算机的设备或器材。它是以应用为中心，软硬件可裁减的，适应应用系统对功能、可靠性、成本、体积、功耗等综合性严格要求的专用计算机系统。

简单地说，嵌入式系统集系统的应用软件与硬件于一体，类似于 PC 中 BIOS 的工作方式，具有软件代码小、高度自动化、响应速度快等特点，特别适合于要求实时和多任务的体系。嵌入式系统主要由嵌入式处理器、相关支撑硬件、嵌入式操作系统及应用软件系统等组成，它是可独立工作的“器件”。

嵌入式系统几乎包括了生活中的所有电器设备，如掌上 **PDA** 、移动计算设备、电视机顶盒、手机上网、数字电视、多媒体、汽车、微波炉、数字相机、家庭自动化系统、电梯、空调、安全系统、自动售货机、蜂窝式电话、消费电子设备、工业自动化仪表与医疗仪器等。

嵌入式处理器

嵌入式系统的硬件部分，包括处理器 / 微处理器、存储器及外设器件和 I/O 端口、图形控制器等。嵌入式系统有别于一般的计算机处理系统，它不具备像硬盘那样大容量的存储介质，而大多使用 EPROM 、 EEPROM 或闪存 (Flash Memory) 作为存储介质。

软件部分包括操作系统软件（要求实时和多任务操作）和应用程序编程。应用程序控制着系统的运作和行为；而操作系统控制着应用程序编程与硬件的交互作用。

嵌入式系统具体分类

(1) 工业控制计算机

早期嵌入式计算机系统采取的方式，用通用计算机改造。

使得配套软件丰富有周边外设支持，较强数据处理能力，体积大，主要用于较大嵌入空间，如移动运载设备，试验装置。

(2) 通用CPU模块

由通用**CPU**构成的主机板系统，为满足工控要求，常常在通用**CPU**模块上添加接口电路。

数据处理能力较强，借助通用计算机系统可方便地开发应用软件，适合于数据处理量大，较小嵌入空间的环境中，如机顶盒、**POS**收银机。

(3) 嵌入式微处理器 (Embedded Microprocessor Unit, EMPU)

嵌入式微处理器采用“增强型”通用微处理器。由于嵌入式系统通常应用于环境比较恶劣的环境中，因而嵌入式微处理器在工作温度、电磁兼容性以及可靠性方面的要求较通用的标准微处理器高。但是，嵌入式微处理器在功能方面与标准的微处理器基本上是一样的。

根据实际嵌入式应用要求，将嵌入式微处理器装配在专门设计的主板上，只保留和嵌入式应用有关的主板功能，这样可以大幅度减小系统的体积和功耗。和工业控制计算机相比，嵌入式微处理器组成的系统具有体积小、重量轻、成本低、可靠性高的优点，但在其电路板上必须包括 **ROM**、**RAM**、总线接口、各种外设等器件，从而降低了系统的可靠性，技术保密性也较差。

(4) 嵌入式微控制器 (Microcontroller Unit, MCU)

嵌入式微控制器又称单片机，它将整个计算机系统集成到一块芯片中。嵌入式微控制器一般以某种微处理器内核为核心，根据某些典型的应用，在芯片内部集成了ROM/EPROM、RAM、总线、总线逻辑、定时 / 计数器、看门狗、I/O、串行口、脉宽调制输出、A/D、D/A、Flash RAM、EEPROM等各种必要功能部件和外设。

为适应不同的应用需求，对功能的设置和外设的配置进行必要的修改和裁减定制，使得一个系列的单片机具有多种衍生产品，每种衍生产品的处理器内核都相同，不同的是存储器和外设的配置及功能的设置。

这样可以使单片机最大限度地和应用需求相匹配，从而减少整个系统的功耗和成本。和嵌入式微处理器相比，微控制器的单片化使应用系统的体积大大减小，从而使功耗和成本大幅度下降、可靠性提高。

由于嵌入式微控制器目前在产品的品种和数量上是所有种类嵌入式处理器中最多的，而且上述诸多优点决定了微控制器是嵌入式系统应用的主流。

(5) 嵌入式 DSP 处理器 (Embedded Digital Signal Processor)

在数字信号处理应用中，各种数字信号处理算法相当复杂，这些算法的复杂度可能是 $O(n^m)$ 的，一般结构的处理器无法实时的完成这些运算。由于 DSP 处理器对系统结构和指令进行了特殊设计，使其适合于实时地进行数字信号处理。在数字滤波、FFT、频谱分析等方面，DSP 算法正大量进入嵌入式领域，DSP 应用正从在通用单片机中以普通指令实现 DSP 功能，过渡到采用嵌入式 DSP 处理器。

嵌入式 DSP 处理器有两类：(1) DSP 处理器经过单片化 EMC (Electro Magnetic Compatibility) 改造、增加片上外设成为嵌入式 DSP 处理器，TI TMS320C2000/C5000 等属于此范畴；(2) 在通用单片机或 SOC 中增加 DSP 协处理器例如 Intel 的 MCS-296 和 Infineon(Siemens) 的 TriCore。

另外，在有关智能方面的应用中，也需要嵌入式 **DSP** 处理器，例如各种带有智能逻辑的消费类产品，生物信息识别终端，带有加解密算法的键盘， **ADSL** 接入、实时语音压解系统，虚拟现实显示等。这类智能化算法一般都是运算量较大，特别是向量运算、指针线性寻址等较多，而这些正是 **DSP** 处理器的优势所在。

(6) 嵌入式片上系统 (System On Chip, SOC)

随着 EDI (Electronic Data Interchange) 的推广和 VLSI (Very Large Scale Integration) 设计的普及化, 以及半导体工艺的迅速发展, 可以在一块硅片上实现一个更为复杂的系统, 这就产生了 SOC 技术。各种通用处理器内核将作为 SOC 设计公司的标准库, 和其他许多嵌入式系统外设一样, 成为 VLSI 设计中一种标准的器件, 用标准的 VHDL、Verlog 等硬件语言描述, 存储在器件库中。用户只需定义出其整个应用系统, 仿真通过后就可以将设计图交给半导体工厂制作样品。这样除某些无法集成的器件以外, 整个嵌入式系统大部分均可集成到一块或几块芯片中去, 应用系统电路板将变得很简单, 对于减小整个应用系统体积和功耗、提高可靠性非常有利。

SOC 可分为通用和专用两类，通用 SOC 如 Infineon(Siemens) 的 TriCore 、 Motorola 的 M-Core ，以及某些 ARM 系列器件，如 Echelon 和 Motorola 联合研制的 Neuron 芯片等；专用 SOC 一般专用于某个或某类系统中，如 Philips 的 Smart XA ，它将 XA 单片机内核和支持超过 2048 位复杂 RSA 算法的 CCU 单元制作在一块硅片上，形成一个可加载 Java 或 C 语言的专用 SOC ，可用于互联网安全方面。

嵌入式操作系统的种类

一般情况下，嵌入式操作系统可以分为两类，一类是面向控制、通信等领域的实时操作系统，如 WindRiver 公司的 VxWorks、ISI 的 pSOS、QNX 系统软件公司的 QNX、ATI 的 Nucleus 等；另一类是面向消费电子产品的非实时操作系统，这类产品包括个人数字助理 (PDA)、移动电话、机顶盒、电子书、WebPhone 等。以及国内的Hopen，Delta Os等嵌入式操作系统。

1.1.2 单片机技术的发展

单片机诞生于20世纪70年代末，经历了SCM、MCU、SoC三大阶段。

1.SCM即单片微型计算机（Single Chip Microcomputer）阶段，主要是寻求最佳的单片形态嵌入式系统的最佳体系结构。“创新模式”获得成功，奠定了SCM与通用计算机完全不同的发展道路。在开创嵌入式系统独立发展道路上，Intel公司功不可没。

1.1.2 单片机技术的发展

2.MCU即微控制器（Micro Controller Unit）阶段，主要的技术发展方向是：不断扩展满足嵌入式应用时，对象系统要求的各种外围电路与接口电路，突显其对象的智能化控制能力。它所涉及的领域都与对象系统相关，因此，发展MCU的重任不可避免地落在电气、电子技术厂家。从这一角度来看，Intel逐渐淡出MCU的发展也有其客观因素。在发展MCU方面，最著名的厂家当数Philips公司。

Philips公司以其在嵌入式应用方面的巨大优势，将MCS-51从单片微型计算机迅速发展 to 微控制器。因此，当我们回顾嵌入式系统发展道路时，不要忘记Intel和Philips的历史功绩。

1.1.2 单片机技术的发展

3. SoC单片机(System On Chip), 单片机是嵌入式系统的独立发展之路, 向MCU阶段发展的重要因素, 就是寻求应用系统在芯片上的最大化解决; 因此, 专用单片机的发展自然形成了SoC化趋势。随着电子技术、IC设计、EDA工具的发展, 基于SoC的单片机应用系统设计会有较大的发展。因此, 对单片机的理解可以从单片微型计算机、单片微控制器延伸到单片应用系统。

单片机的发展可分为以下阶段

1970年微型计算机研制成功后，随后就出现了单片机。美国Inter公司在1971年推出了4位单片机4004；1972年推出了雏形8位单片机8008。特别是在1976年推出MCS-48单片机以后的三十年中，单片机的发展和其相关的技术经历了数次的更新换代。其发展速度大约每三四年要更新一代、集成度增加一倍、功能翻一番。尽管单片机出现的历史并不长，但以8位单片机的推出为起点，那么，单片机的发展大致可分为四个阶段。

第一阶段（1976年-1978年）：初级单片机阶段。以Inter公司MCS-48为代表。这个系列的单片机内集成有8位CPU、I/O接口、8位定时器/计数器，寻址范围不大于4K字节，简单的中断功能，无串行接口。

单片机的发展可分为以下阶段

第二阶段（**1978年-1982年**）：单片机完善阶段。在这一阶段推出的单片机其功能有较大的加强，能够应用于更多的场合。这个阶段的单片机普遍带有串行I/O口、有多级中断处理系统、**16位**定时器/计数器，片内集成的**RAM**、**ROM**容量加大，寻址范围可达**64K**字节。一些单片机片内还集成了**A/D**转换接口。这类单片机的典型代表有Inter公司的**MCS-51**、Motorola公司的**6801**和Zilog公司的**Z8**等。

单片机的发展可分为以下阶段

第三阶段（**1982年-1992年**）：**8位单片机巩固发展及16位高级单片机发展阶段**。在此阶段，尽管**8位单片机**的应用已广泛普及，但为了更好地满足测控系统的嵌入式应用的要求，单片机集成的外围接口电路有了更大的扩充。这个阶段单片机的代表为**8051**系列。许多半导体公司和生产厂以**MCS-51**的**8051**为内核，推出了满足各种嵌入式应用的多种类型和型号的单片机。

其主要技术发展有：

（1） 外围功能集成。满足模拟量直接输入的**ADC**接口；满足伺服驱动输出的**PWM**；保证程序可靠运行的程序监控定时器**WDT**（俗称看门狗电路）。

（2） 出现了为满足串行外围扩展要求的串行扩展总线和接口，如**SPI**、**I2C Bus**、单总线（**1-Wire**）等。

（3） 出现了为满足分布式系统，突出控制功能的现场总线接口，如**CAN Bus**等。

（4） 在程序存储器方面广泛使用了片内程序存储器技术，出现了片内集成**EPROM**、**EEPROM**、**FlashROM**以及**MaskRoM**、**OTPROM**等各种类型的单片机，以满足不同产品的开发和生产的需要，也为最终取消外部程序存储器扩展奠定了良好的基础。与此同时，一些公司面向更高层次的应用，发展推出了**16**位的单片机，典型代表有**Inter**公司的**MCS-96**系列的单片机。

单片机的发展可分为以下阶段

第四阶段（**1993年-现在**）：百花齐放阶段。现阶段单片机发展的显著特点是百花齐放、技术创新，以满足日益增长的广泛需求。其主要方面有：

（1）单片嵌入式系统的应用是面对最底层的电子技术应用，从简单的玩具、小家电；到复杂的工业控制系统、智能仪表、电器控制；以及发展到机器人、个人通信信息终端、机顶盒等。因此，面对不同的应用对象，不断推出适合不同领域要求的，从简易性能到多全功能的单片机系列。

(2) 大力发展专用型单片机。早期的单片机是以通用型为主的。由于单片机设计生产技术的提高、周期缩短、成本下降，以及许多特定类型电子产品，如家电类产品的巨大的市场需求能力，推动了专用单片机的发展。在这类产品中采用专用单片机，具有低成本、资源有效利用、系统外围电路少、可靠性高的优点。因此专用单片机也是单片机发展的一个主要方向。

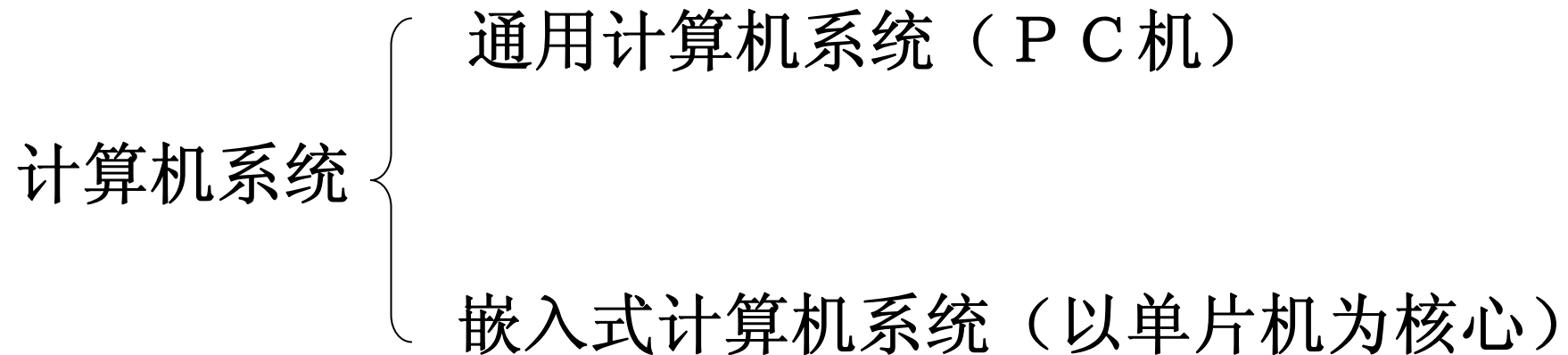
(3) 致力于提高单片机的综合品质。采用更先进的技术来提高单片机的综合品质，如提高I/O口的驱动能力；增加抗静电和抗干扰措施；宽（低）电压低功耗等。

1.1.3 单片机的特点、分类及发展趋势

单片机的主要特点

- 集成度高
- 控制功能强
- 可靠性高
- 低功耗、低电压
- 外部总线丰富
- 功能扩展性强
- 体积小、成本低
- 性价比高

◆在一片集成电路芯片上集成微处理器、存储器、I/O接口电路等元器件，从而构成了“单芯片微型计算机”，简称单片机。



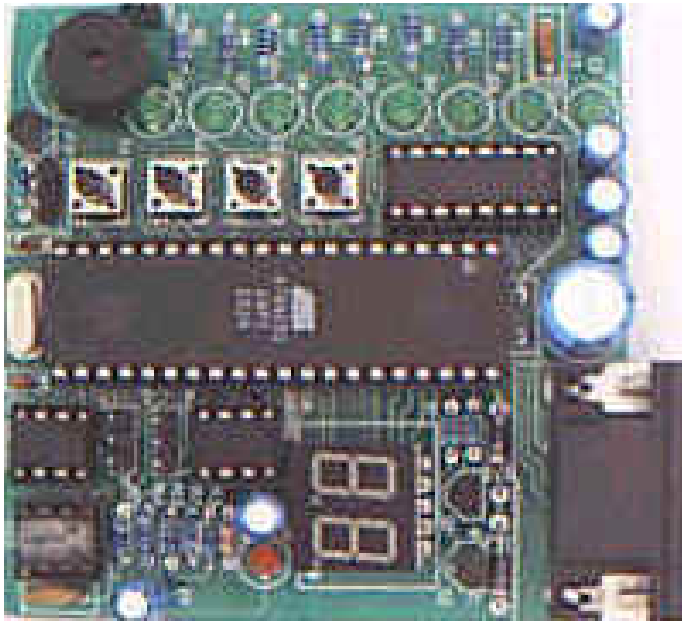


图1.1 单片机系统与PC系统实例图



图1.2 PC的鼠标

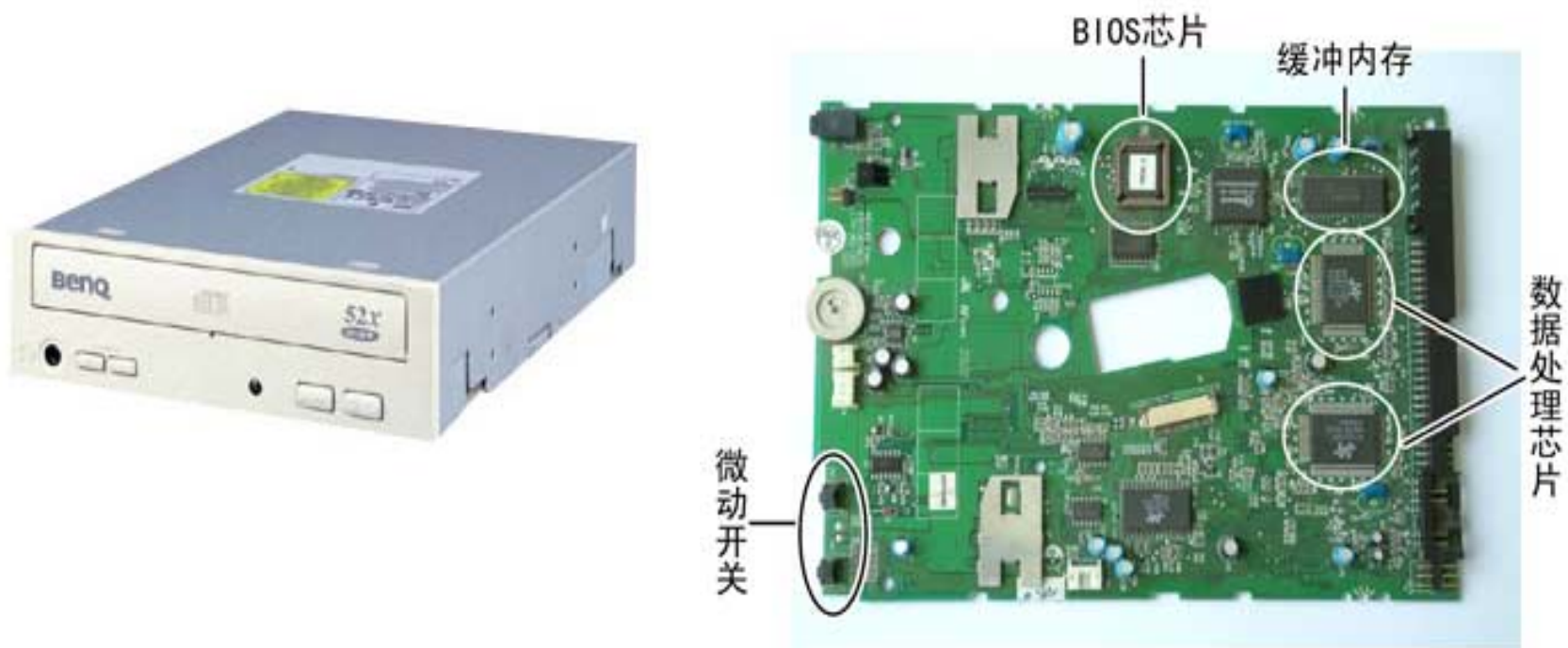


图1.3 PC的光驱



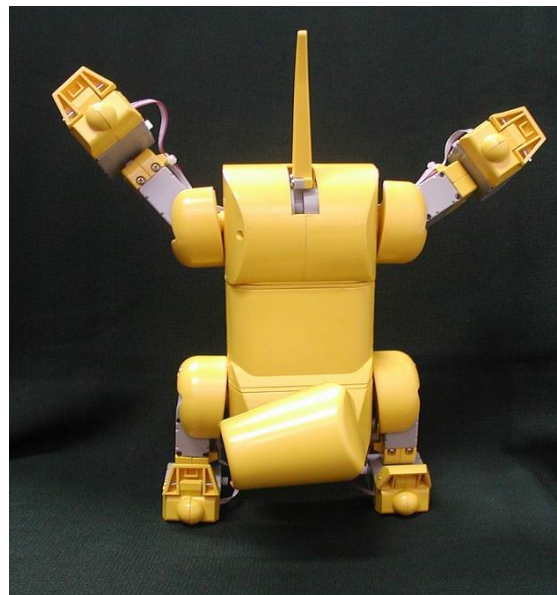
图1.4 优盘

表1-1 单片机和PC的差别

项目	PC	单片机
概念	形态标准，外部设备齐全，应用多个单片机和一个微处理器、通过装配不同的应用软件，多个部件组成的适应社会各个方面的计算机应用系统	芯片级产品。它以某一种微处理器为核心，将 RAM 、总线、 ROM/EPROM 、总线逻辑、定时/计数器、并行 I/O 口、串行 I/O 口、看门狗、脉宽调制输出、 A/D 、 D/A 等集成到一块芯片内
主机板	复杂	简单
CPU	Intel 、 AMD 等	片内集成
存储器	硬盘、内存条	片内集成或外扩展芯片
操作系统	Windows 或 Linux 等	自己编制、自行发展
输出	CRT 或 LCD 屏幕等	端口输出电信号驱动 LED 数码管或 LCD 、发光管指示
输入	标准键盘、鼠标等	端口输入非标准键盘及电信号
编程语言	VC 、 VB 等	汇编语言或 C 语言
应用	常在办公室、家庭见到	已经嵌入到产品中，几乎见不到

单片机的分类

- 按单片机数据处理位数来划分
 - 4位单片机**
 - 8位单片机**
 - 16位单片机**
 - 32位单片机**
- 按单片机适用范围来划分
 - 通用型单片机
 - 专用型单片机
- 按单片机并行总线来划分
 - 总线型单片机（如**AT89C51**）
 - 非总线型单片机（如**AT89C2051**）



单片机的发展趋势

- **CMOS化**——单片机将具有更低的功耗、更低的电压。
- **高性能化**——精简指令集(RISC)结构和流水线技术将得到广泛应用。
- **高可靠性**——提高单片机的抗电磁干扰能力。
- **大容量化**——扩大片内存储器容量。
- **多功能化**——把众多的各种外围功能器件集成在片内，如模/数转换器、数/模转换器、液晶显示驱动器 等。
- **串行扩展技术**—— SPI、I²C、Microwire、1-Wire等串行总线的引入，可以使单片机的引脚设计得更少，单片机系统结构更加简化。

●单片机的发展趋势

低功耗CMOS化: 80C51采用CHMOS

互补高密度
金属氧化物
半导体工艺

特点: 高速度
低功耗

微型单片化: 集成

1. 基本功能:

CPU

RAM

ROM

并行通信接口

串行通信接口

中断系统

定时电路

时钟电路

2. 增强功能:

A/D转换器

D/A转换器

SPI总线

SMBus

PCA

PGA

电压比较器

温度传感器

脉宽调制电路(PMW)

看门狗(WDT)

LCD(液晶)驱动电路等

主流与多品种共存:

1. 主流: 以80C51为核心的系列兼容单片机(兼容结构及指令)

如: PHILIPS、ATMEL(89)、Winbond及C8051F0XX系列单片机

2. 支流: Motorola、Zilog、Microchip、ATMEL(AVR)

MSP430系列16位超低功耗单片机

1.2 单片机的应用

●单片机的应用

- 在智能仪器仪表上的应用：采用单片机控制使得仪器仪表数字化、智能化、微型化，且功能比起采用模拟电路或数字电路更加强大。
- 在工业控制中的应用：构成形式多样的控制系统和数据采集系统。
- 在家用电器中的应用：电饭煲、洗衣机、电冰箱、空调机、彩电、其他音响视频器材。
- 在计算机网络和通信领域中的应用：手机、电话机、小型程控交换机、楼宇自动通信呼叫系统、列车无线通信。
- 单片机在医用设备领域中的应用：医用呼吸机、各种分析仪、监护仪、超声诊断设备及病床呼叫系统等。

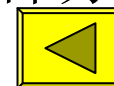


表1-2 目前世界上著名的8位单片机的生产厂家和主要机型

公司	产品型号	兼容性
Intel公司	MCS-51及其增强系列单片机	与MCS-51兼容
Atmel公司	AT89X51系列Flash单片机	
Philips公司	8XC552及89C66X系列高性能单片机	
Winbond公司	W78C51及W77C51系列高速低价单片机	
LG公司	GMS90/97系列高速低压单片机	
Cygnal公司	C8051F系列高速SOC单片机	与MCS-51不兼容
Motorola公司	6801和6805系列高性能单片机	
Zilog公司	Z8系列特殊应用设计单片机	
Microchip公司	PIC系列RISC结构单片机	
Atmel公司	AVR系列RISC结构单片机	

89系列单片机的型号说明

- 89系列单片机的型号由3部分组成
- 分别是前缀、型号、后缀
- 格式如下：

后缀:第1个参数: 速度 12MHz、24MHz
第2个参数: 封装 J 塑料 P 双列直插式
第3个参数: 温度 C:0~70度
第4个参数: 工艺

AT89CXXXX—XXXX

前缀:
Atmel公司

89CXXXX C:CHMOS
型号: 89LVXXXX LV: 低电压
89SXXXX S:可下载Flash

- (1) 后缀中的第1个参数X用于表示速度。它的含义如下：
- X=12 表示速度为12MHz
- X=16 表示速度为16MHz
- X=20 表示速度为20MHz
- X=24 表示速度为24MHz
- (2) 后缀中的第2个参数X用于表示封装。它的含义如下：
- X=J 表示塑料J引线芯片载体
- X=L 表示无引线芯片载体
- X=P 表示塑料双列直插DIP封装
- X=S 表示SQIC封装
- X=Q 表示PQFP封装
- X=A 表示TQFP封装
- X=W 表示裸芯片
- (3) 后缀中第3个参数X用于表示温度范围。它的含义如下：
- X=C 表示商业产品 温度范围为0~+70℃
- X=I 表示工业产品 温度范围为-40~+85℃
- X=A 表示汽车用产品 温度范围为-40~+125℃
- X=M 表示军用产品 温度范围为-55~+150℃
- (4) 后缀中的第4个参数X用于说明产品的处理情况。它的含义如下：
- X=Null 表示处理工艺是标准工艺
- X=1883 表示处理工艺采用MIL—STD 883标准

1.2.1 智能仪器的应用

- 单片机用于各种仪器仪表，使仪器仪表智能化，可以提高测量的自动化程度和精度；简化仪器仪表的硬件结构，减小体积，提高其性价比。例如：温度智能控制仪表、医用仪表、汽车电子设备、数字示波器等。
- 例如，在普通模拟示波器的基础上用单片机进行改造而成的数字存储示波器，克服了普通模拟示波器的缺点，并增加了许多功能，如可以显示大量的预触发信息，可以长期贮存波形，可以在打印机或绘图仪上制作硬拷贝以供编制文件使用，可以将采集的波形和操作人员手工或示波器全自动采集的参考波形进行比较，波形信息可用数学方法进行处理。

1.2.2 通信设备的应用

- 单片机与通信技术相结合促使通信设备的智能控制水平大大提高，广泛应用于通信的各个领域。例如：调制解调器、传真机、复印机、打印机、移动电话机、固定电话机等。
- 例如，传统的电话机只能实现简单的拨号、响铃、通话等功能，使用单片机后，可以开发出来电显示、存储电话号码、时钟显示、免提、重拨、声控等功能。功能更多的无绳电话机、录音电话机、可视电话机等多功能电话机也已走进人们的生活。

1.2.3 家用电器的应用

- 传统的家电配上单片机以后，提高了智能化程度，增加了功能，倍受人们喜爱；单片机使人类生活更加方便、舒适、丰富多彩。例如：洗衣机、电冰箱、电子玩具、收录机、微波炉、电视机、录像机、音响设备、程控玩具、游戏机等。
- 例如，单片机控制的全自动洗衣机集洗涤、脱水于一体，能自动完成洗衣全过程，并有多种洗涤程序供用户自由选择，能任意调节工作时间，显示工作状态、洗涤时间和脱水时间，能自动处理脱水不平衡，具有各种故障和高低电压自动保护功能，工作结束或电源故障会自动断电以确保安全。目前，有的全自动洗衣机还采用了模糊技术，即洗衣机能对传感器提供的信息进行逻辑推理，自动判断衣服质地、重量、脏污程度，从而自动选择最佳的洗涤时间、进水量、漂洗次数、脱水时间，并显示洗涤剂的用量，达到整个洗涤过程自动化，使用方便，节能节水。

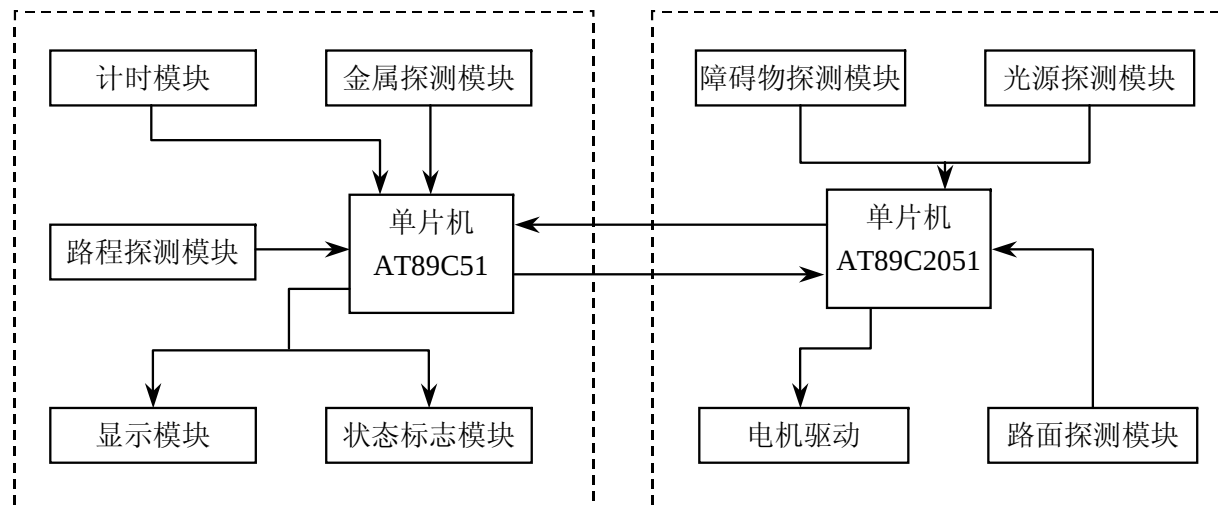
1.2.4 工业控制的应用

- 机电一体化是机械工业发展的方向。机电一体化产品是指集机械技术、微电子技术、计算机技术于一体，具有智能化特征的机电产品，例如微机控制的车床、钻床等。单片机作为产品中的控制器，能充分发挥它体积小、可靠性高、功能强等优点，可大大提高机器的自动化、智能化程度。
- 单片机广泛用于导弹的导航装置、飞机上各种仪表的控制、计算机的网络通信与数据传输、机器人、工业自动化过程的实时控制和数据处理。例如，在这些实时控制系统中，都可以用单片机作为控制器，单片机的实时数据处理能力和控制功能，可使系统保持在最佳工作状态，提高系统的工作效率和产品质量。
- 在比较复杂的系统中，常采用分布式多机系统。多机系统一般由若干台功能各异的单片机组成，各自完成特定的任务，它们通过串行通信相互联系、协调工作。单片机在这种系统中往往作为一个终端机，安装在系统的某些节点上，对现场信息进行实时测量和控制。单片机的高可靠性和强抗干扰能力，使它可以置于恶劣环境的前端工作。

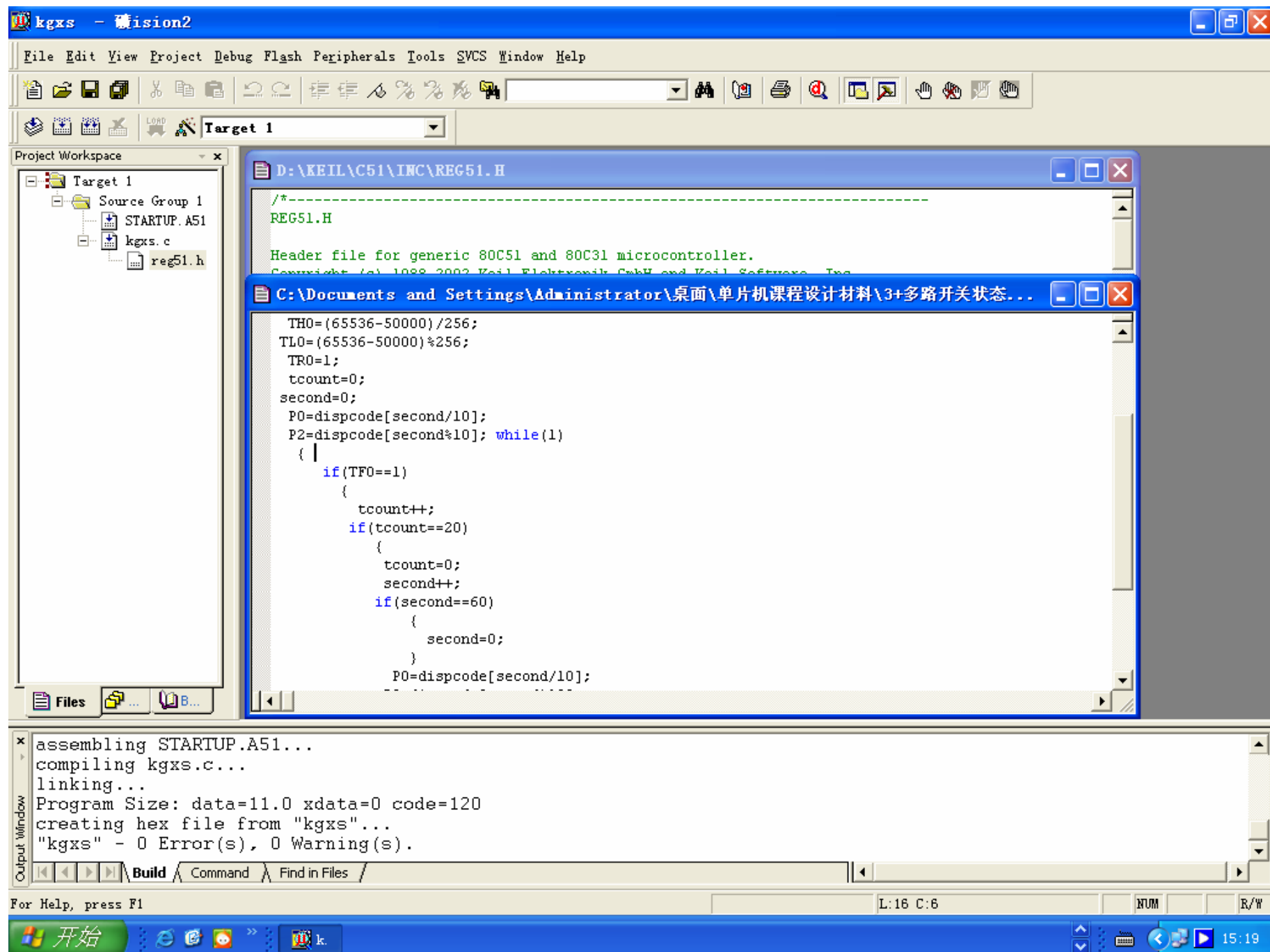
1.2.4 工业控制的应用

- 在教育部倡导的全国大学生电子设计大赛中，有一道设计简易智能电动车的题目，参赛者用51系列单片机完成了设计任务并取得了较好的成绩。该设计采用单片机AT89C51和AT89C2051芯片为核心部件，利用光电检测技术、超声波测距、电涡流检测技术并配合一套独特的软件实现了电动车识别并选择正确的行进路线、电机的方向和速度控制、自动寻找光源、判断并自动躲避障碍物、时间显示、蜂鸣器报警等功能。

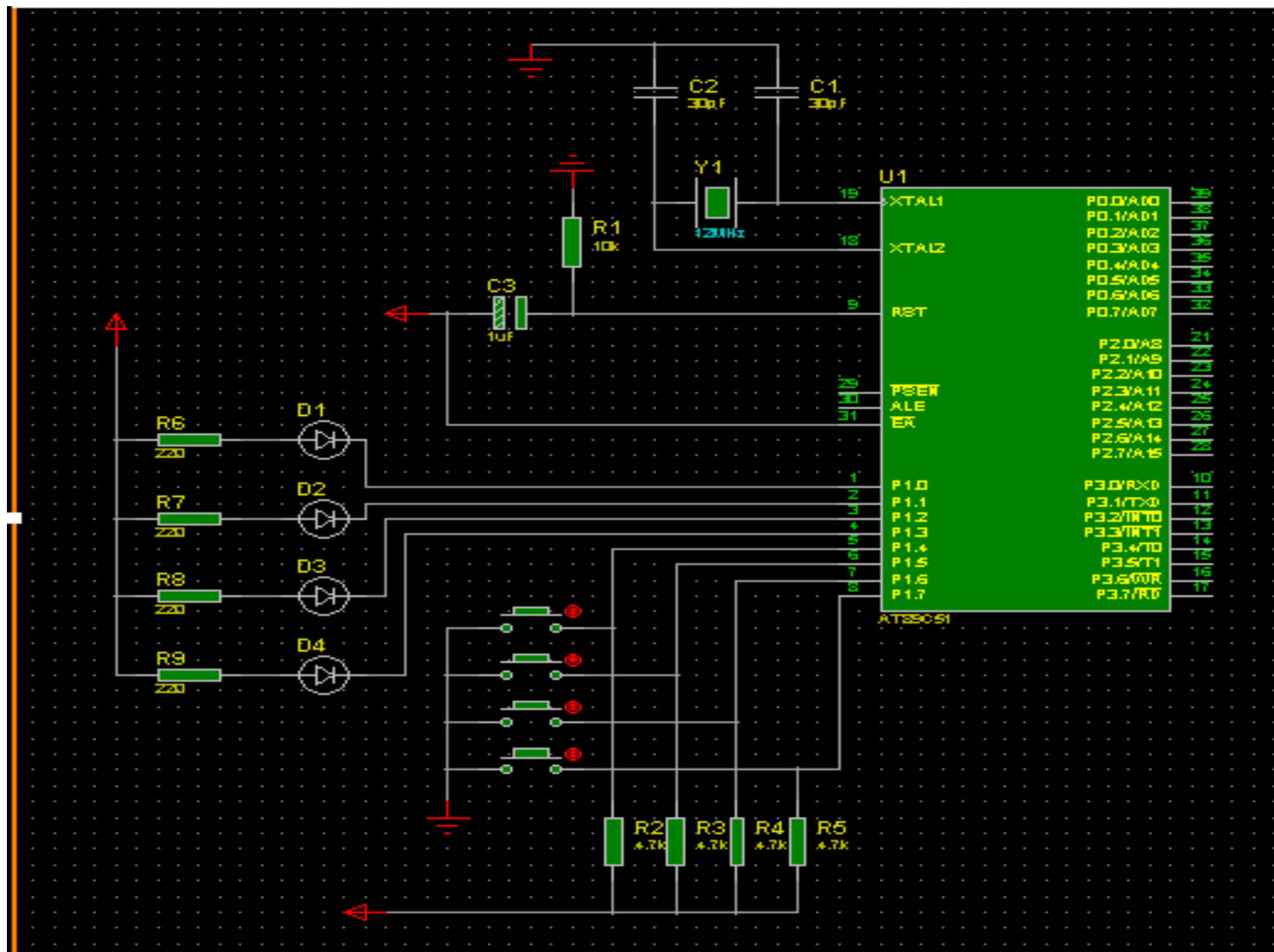
电动车控制方案如下图所示。

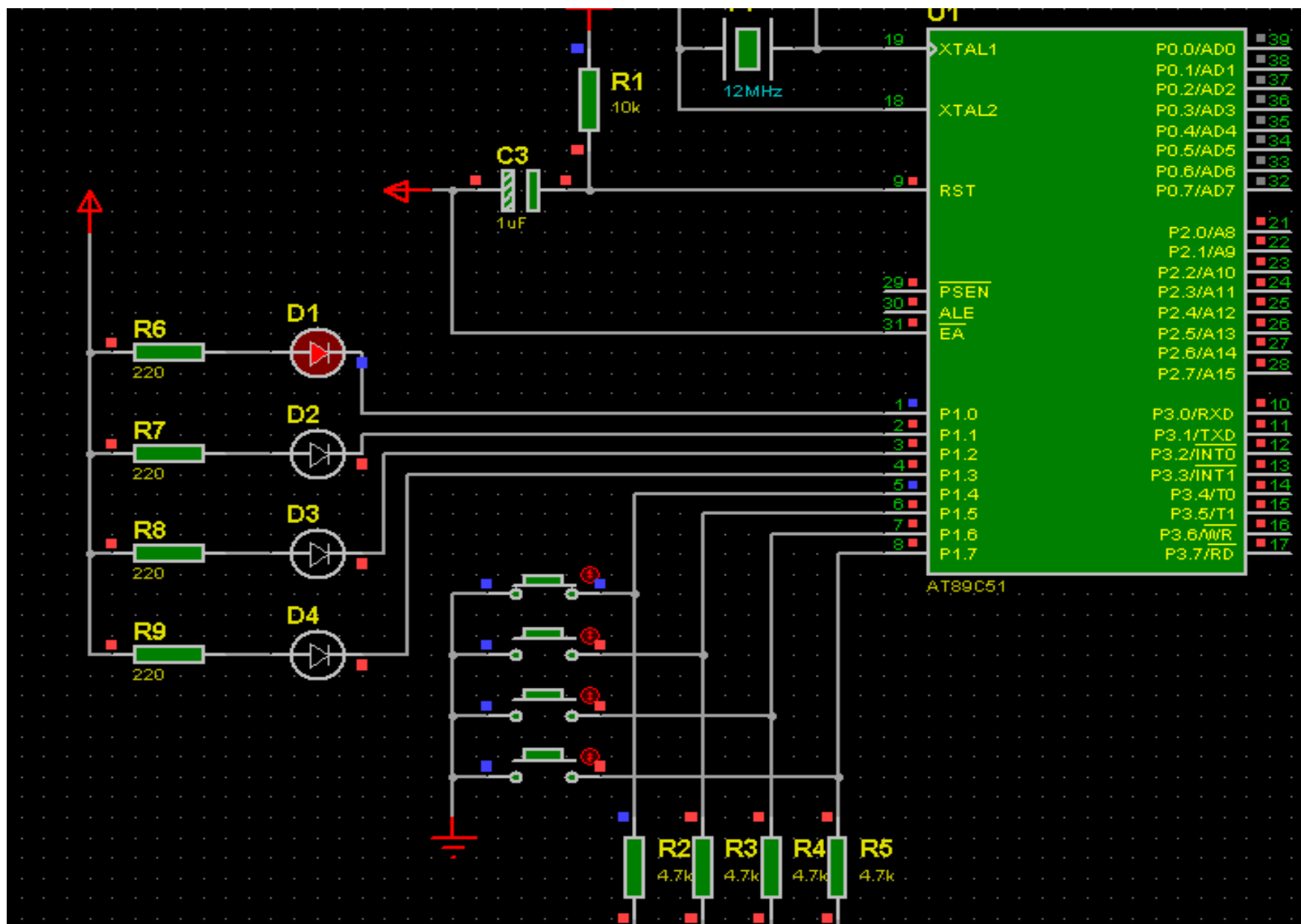


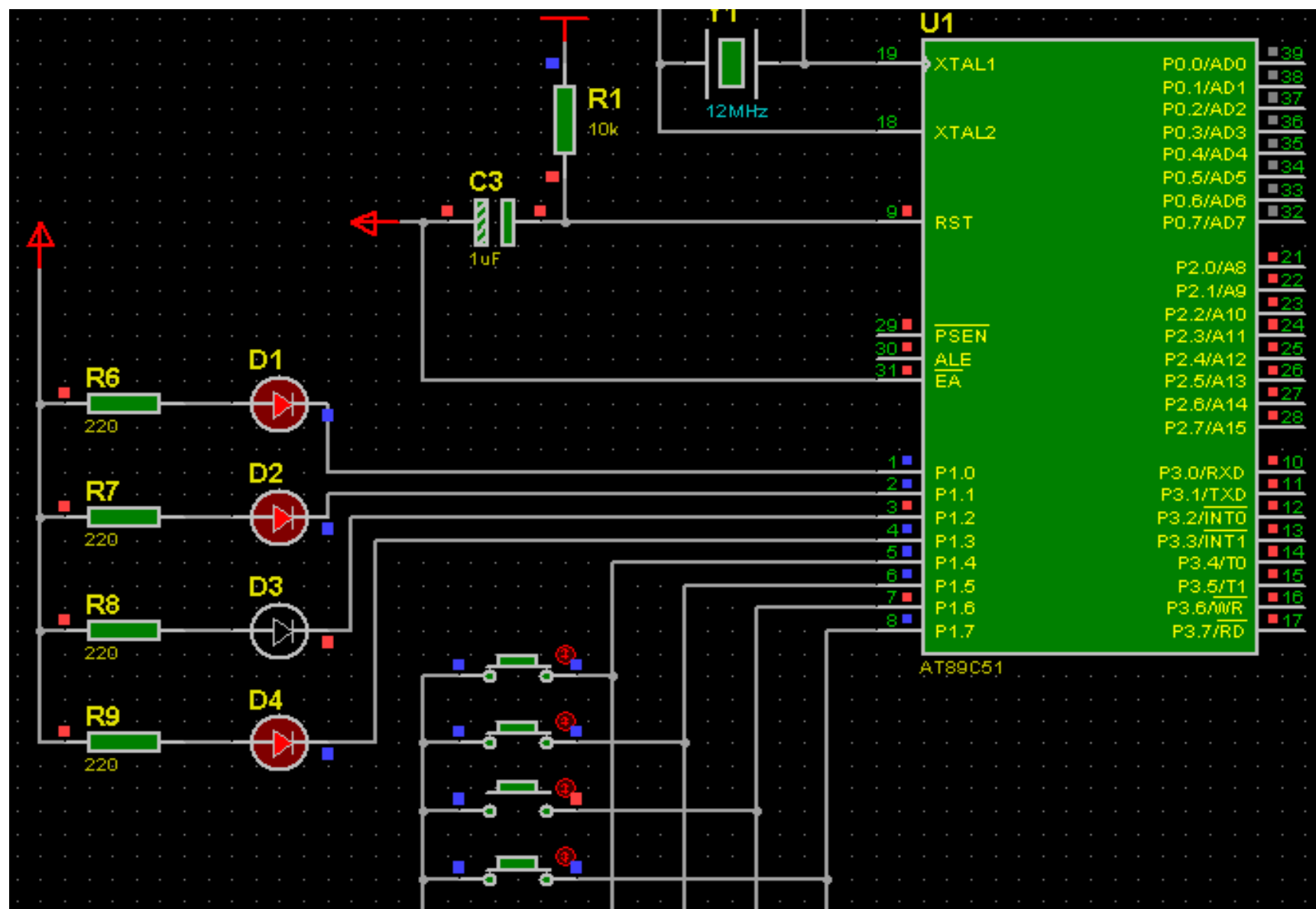
智能电动车控制方案

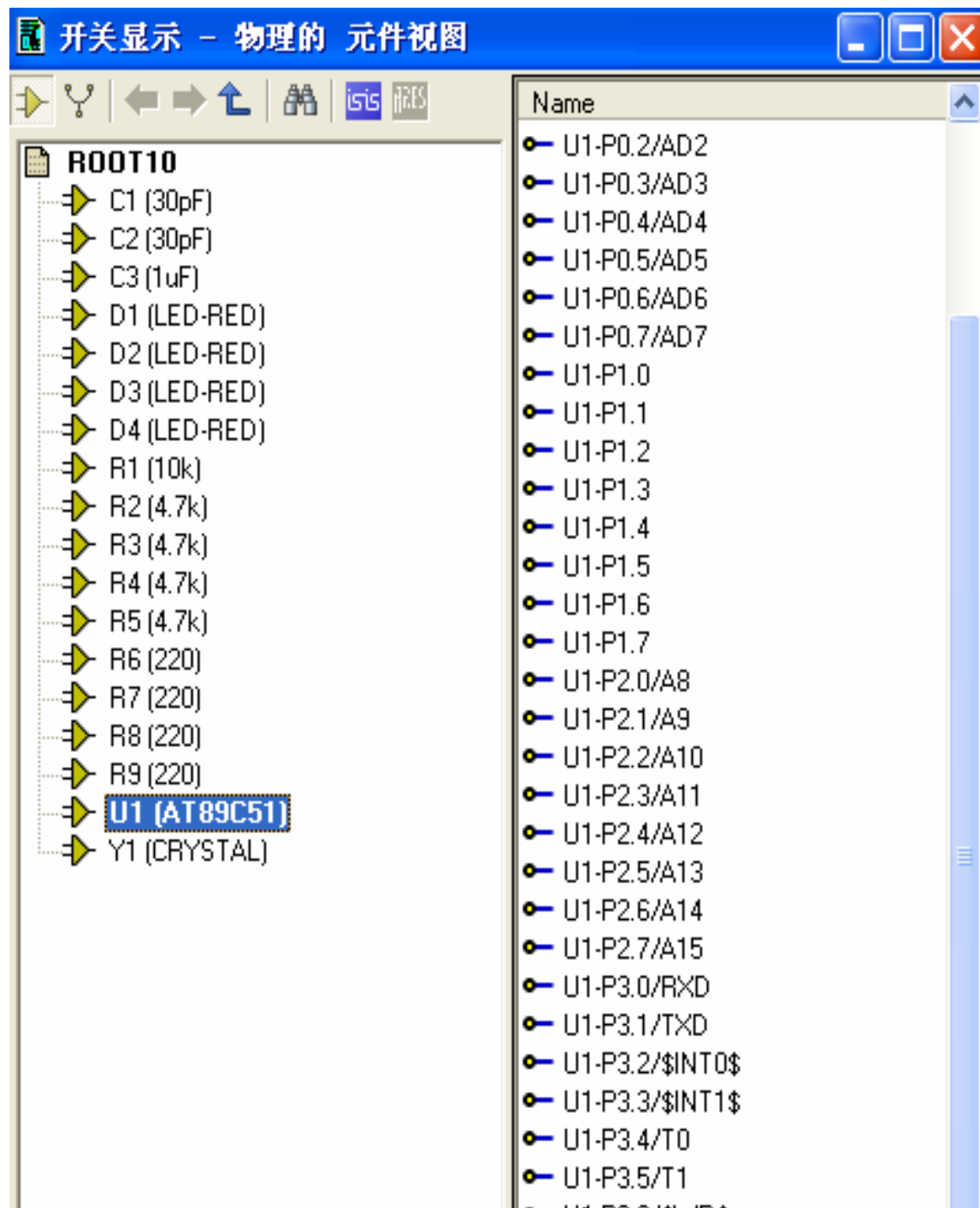


Keil+Proteus软件仿真方法



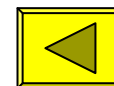






1.3 计算机运算基础

- 1.3.1 数制
- 1.3.2 有符号二进制数的编码
- 1.3.3 二进制编码



1.3.1 数制

- 数制是指数的制式，是人们利用符号进行计数的科学方法。
- 数制有很多种，在计算机中常用的数制有十进制、二进制和十六进制。

1. 十进制

- 十进制中共使用0~9十个数字符号(称为数码)，数码的个数称为基数。十进制的基数为10，任何一个十进制数都可用这十个数码的组合来表示。
- 十进制数一般有以下特点：
 - 有0~9十个数字符号，基数为10，各位的权为 10^n (n 为整数)。
 - 运算规则是加法逢十进一、减法借一为十。
 - 用后缀D(Decimal)表示十进制数，通常对十进制数可不加后缀。

- 任一个数中每一位的值可用该位的数字乘以基数的幂次来表示，基数的幂次称为权。
- 十进制数各位的权是以10为底的幂，任何一个十进制数都可用权展开式来表示。

例如：十进制数1986.129可如下式展开：

$$1986.129 = 1 \times 10^3 + 9 \times 10^2 + 8 \times 10^1 + 6 \times 10^0 + 1 \times 10^{-1} \\ + 2 \times 10^{-2} + 9 \times 10^{-3}$$

上式中 10^0 、 10^1 、 10^2 、 10^3 分别称为0权位、1权位、2权位、3权位。

2. 二进制

- 二进制中共使用0、1两个数字符号。
- 二进制数一般有以下特点：
 - 有0、1两个数字符号，基数为2，各位的权为 2^n (n 为整数)。
 - 运算规则是加法逢二进一、减法借一为二。
 - 用后缀B(Binary)表示二进制数。

- 二进制数各位的权是以2为底的幂，任何一个二进制数都可用权展开式来表示。

例如：二进制数1011.11可如下式展开：

$$\begin{aligned} 1011.11\text{B} &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} \\ &= 11.75 \end{aligned}$$

由上式可知，二进制数1011.11B相当于十进制数11.75。

3. 十六进制

- 十六进制中共使用0~9及A~F十六个数字符号，其中A~F相当于十进制数的10~15。
- 十六进制数一般有以下特点：
 - 有0~9及A~F十六个数字符号，基数为16，各位的权为 16^n (n 为整数)。
 - 运算规则是加法逢十六进一、减法借一为十六。
 - 用后缀H (Hexadecimal) 表示十六进制数。

- 十六进制数各位的权是以16为底的幂，任何一个十六进制数都可用权展开式来表示。

例如：十六进制数5D6E. 4可如下式展开：

$$\begin{aligned} 5D6E. 4H &= 5 \times 16^3 + 13 \times 16^2 + 6 \times 16^1 + 14 \times 16^0 + 4 \times 16^{-1} \\ &= 23918. 25 \end{aligned}$$

由上式可知，十六进制数5D6E. 4相当于十进制数23918. 25。

十进制数、二进制数和十六进制数对照表

十进制数	二进制数	十六进制数	十进制数	二进制数	十六进制数
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

4. 不同数制之间的转换

- 二进制数转换成十六进制数——采用“四位合一位”法
- 十六进制数转换成二进制数——采用“一位分四位”法
- 二进制数转换成十进制数——按权展开后相加
- 十六进制数转换成十进制数——按权展开后相加
- 十进制整数转换成二进制整数——采用“除以2取余”法
- 十进制整数转换成十六进制整数——采用“除以16取余”法
- 十进制小数转换成二进制小数——采用“乘2取整”法
- 十进制小数转换成十六进制小数——采用“乘16取整”法

我们用二进制和十进制间的相互转换来说明十进制和其他进制间的转换问题

假设把一个二进制数表示为：

$$(S)_2 = x_n 2^n + x_{n-1} 2^{n-1} + \dots x_1 2^1 + x_0 2^0 + x_{-1} 2^{-1} + x_{-2} 2^{-2} + \dots x_{-m} 2^{-m}$$

$$\text{如}(101.1)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1}$$

1 十进制整数转换为二进制整数

(1) 减权定位

如果以代码序列表示转换后得到的二进制结果，则代码序列为 $x_n x_{n-1} \dots x_1 x_0$ 。如果以多项式表示转换结果，则有 $(S)_2 = x_n 2^n + x_{n-1} 2^{n-1} + \dots x_1 2^1 + x_0 2^0$ ，多项式中每一项都有各对应的权值，且互不相同，若该项系数 $x_i = 0$ ，则该项的数值为0。反过来看，若该项值不为0，则相应的 x_i 必为1。根据这个对应关系可以导出由十进制整数转换为二进制整数的规律：自高位起依次与二进制各位的权值进行比较，若够减则对应位 $x_i = 1$ ，减去该位权值后再往下比较；若不够减则对应位 $x_i = 0$ ，越过该位后与下一位的权值相比较；如此进行，直到所有二进制位的权值都比较完毕。

求 $(326)_{10} = (?)_2$

因为 $512 > 326 > 256$ ，所以从权值256对应位开始比较

减权比较	x_i	位权
$326 - 256 = 70$	1	256
$70 < 128$	0	128
$70 - 64 = 6$	1	64
$6 < 32$	0	32
$6 < 16$	0	16
$6 < 8$	0	8
$6 - 4 = 2$	1	4
$2 - 2 = 0$	1	2
$0 < 1$	0	1

注意，第一步求得的是最高位。这种方法比较直观，易于验算，常用于手工转换场合。但这种方法需要查找各位的权值，判断每次该减还是不该减，各步的减数也不同，因此用程序实现较繁。

(2) 除基取余法

将 $(S)_2 = x_n 2^n + x_{n-1} 2^{n-1} + \dots + x_1 2^1 + x_0 2^0$ 两边同除以基数2
得 $S/2 = (x_n 2^{n-1} + x_{n-1} 2^{n-2} + \dots + x_1 2^0) + x_0/2$

显然，括号内为商， x_0 是余数。若余数为0，则表明 $x_0=0$ ；若余数为1，表明 $x_0=1$ 。继续对商除以基数2，可依次判定 $x_1 \sim x_n$ 。

由此得到除基取余法的转换规律如下：将十进制整数除以2，所得余数即为对应的二进制数低位的值；继续对商除以2，所得的各次余数就是所求二进制数的各位值；如此进行，直到商等于0为止，最后一项余数为所求二进制数最高位的值。

求 $(326)_{10} = (?)_2$

	余数	
2/326		
2/163...	0	x0
2/81...	1	x1
2/40...	1	x2
2/20...	0	x3
2/10...	0	x4
2/5...	0	x5
2/2...	1	x6
2/1...	0	x7
0...	1	x8

这种方法每步都重复执行除2运算，操作统一，所以便于编程实现。

2 十进制小数转换为二进制小数

(1) 减权定位法

与整数转换所用的减权定位法相同，但转换后的二进制小数有可能位数较多，需根据规定的字长或所要求的精度来决定所取位数。

求 $(0.635)_{10} = (?)_2$

$0.635 - 0.5 = 0.135 \dots$	1	0.5
$0.135 < 0.25 \dots$	0	0.25
$0.135 - 0.125 \dots$	1	0.125
$0.010 < 0.0625 \dots$	0	0.0625
.....		

(2) 乘基取整法

与整数转换相似，在小数中也可采用类似的思路，从高位开始，乘以基数，分析其产生的整数部分，以决定相应的二进制高位值。

用多项式表示二进制小数如下

$$(S)_2 = x_{-1}2^{-1} + x_{-2}2^{-2} + \dots + x_{-m}2^{-m}$$

两边同乘以2

$2S = x_{-1} + (x_{-2}2^{-1} + \dots + x_{-m}2^{-m+1})$ 显然，括号内仍为小数，括号外是整数部分。若 $x_{-1} = 0$ ，则乘以基数2之后的结构 $2S$ 仍为小数；若 $x_{-1} = 1$ ，则 $2S$ 中出现整数为1的部分。重复上述将小数部分乘以2的操作，将能根据各次是否出现整数部分，依次判定 $x_{-2} \sim x_{-m}$

由此得到乘基数取整法的转换规律:将待转换的十进制小数乘以基数2, 所得到的整数就是二进制小数的高位值; 继续对所余小数部分乘以2, 所得到的整数就是次高位; 如此继续, 直到乘积已全部为整数, 或已满足所需精度为止。

求 $(0.8125)_{10} = (?)_2$

0.8125	$\times 2 = 1.625$		1
0.625	$\times 2 = 1.25$		1
0.25	$\times 2 = 0.5$		0
0.5	$\times 2 = 1$		1

这种算法简单而统一, 易于编程实现

3 二进制整数转换为十进制整数

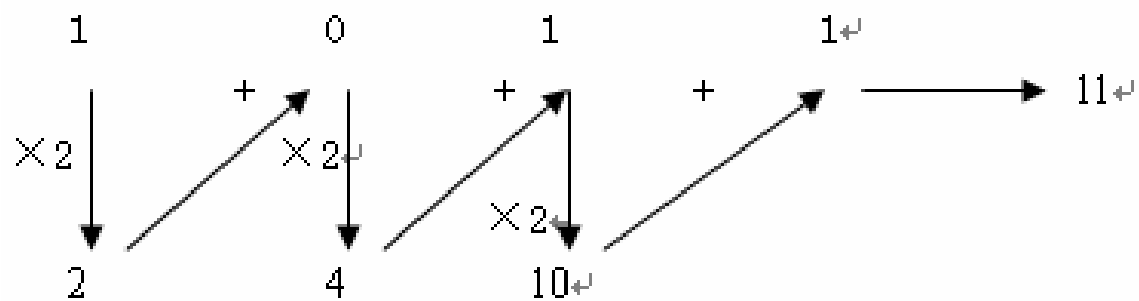
- (1) 按权相加法 易于手工实现
- (2) 逐次乘基相加法 易于编程实现

由于相邻两位的位权之比就是基数2，因此转换算法可改进为逐次乘以2相加，使操作统一便于编程实现。将：

$$S = x_n 2^n + x_{n-1} 2^{n-1} + \dots x_1 2^1 + x_0 2^0 \text{ 改写成} \\ = \{ \dots [(x_n \cdot 2 + x_{n-1}) \cdot 2 + x_{n-2}] \cdot 2 + \dots \} \cdot 2 + x_0$$

从上式可得出转换规律：从最高位开始，乘以基数2，再与次高位相加；所得结果再乘以2，并与相邻位相加；如此继续，加到最低位为止。

求 $(1011)_2 = (?)_{10}$



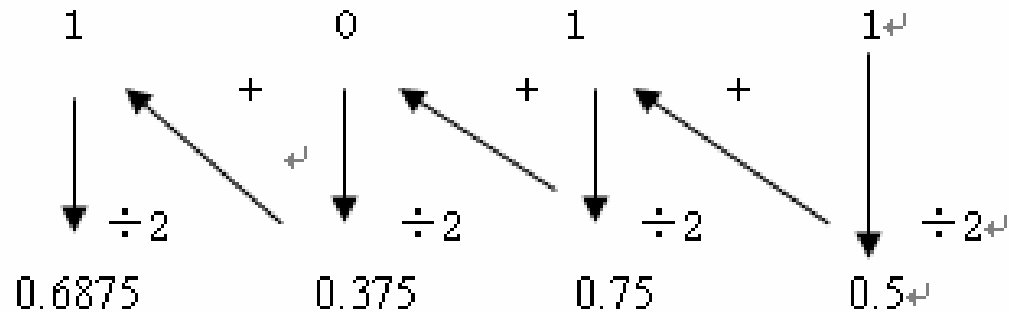
4 二进制小数转换为十进制小数

- (1) 按权相加法 与整数转换方法相同
- (2) 逐次除基相加法 根据小数序列各位权从高位到地位是逐次除以基数这一性质，可将式

$$S = x_{-1}2^{-1} + x_{-2}2^{-2} + \dots + x_{-m}2^{-m} \text{改写成}$$
$$= 1/2 \{x_{-1} + 2^{-1}[x_{-2} + \dots + 2^{-1}(x_{-m+1} + 2^{-1}x_{-m})]\}$$

从上式可导出转换方法如下：从最低位开始除以基数2，再与次低位相加；所得结果再除以2，并与相邻高位相加；如此继续，直到与小数点后第一位相加并除以2为止。

求 $(0.1011)_2 = (?)_{10}$



至此，我们对每种转换都提供了两种方法。一种比较直观，适于手算验证；另一种更适合编制转换程序，希望同学们熟练掌握。

【例1】 将十进制整数100转换成二进制整数。

解：用2连续除100，直至商小于2，算式如下：

2		100		余数 0	最低位 ↑ 最高位
2		50		余数 0	
2		25		余数 1	
2		12		余数 0	
2		6		余数 0	
2		3		余数 1	
		1		余数 1	

运算结果：100=1100100B。

【例2】 将十进制小数0.618转换成二进制小数。

解：用2连续乘以0.618，小数部分为0或保留4位小数，算式如下：

$\begin{array}{r} \times 0.618 \\ 2 \\ \hline 1.236 \end{array}$	 整数 1	最高位 ↓ 最低位
$\begin{array}{r} \times 0.236 \\ 2 \\ \hline 0.472 \end{array}$	 整数 0	
$\begin{array}{r} \times 0.472 \\ 2 \\ \hline 0.944 \end{array}$	 整数 0	
$\begin{array}{r} \times 0.944 \\ 2 \\ \hline 1.888 \end{array}$	 整数 1	
$\begin{array}{r} \times 1.888 \\ 2 \\ \hline 3.776 \end{array}$	 整数 1	

运算结果：0.618 $\approx$ 0.1001B。

1.3.2 有符号二进制数的编码

基本概念

- 在计算机中把有符号数、字母、数字和字符用二进制代码按一定规律编排，使每组代码具有特定的含义，称为计算机编码。
- 有符号二进制数的正负号分别用“+”和“-”来表示。一般规定最高位为符号位，最高位为“0”表示正数，为“1”表示负数，这种把符号数值化的数称为机器数，而原来的数值称为机器数的真值。
- 有符号二进制数有三种表示法，即原码、反码和补码。

1. 原码

- 原码的表示方法：
正数的符号位用“0”表示。
负数的符号位用“1”表示。
- 0的原码有两种表示形式：
 $[+0]_{\text{原}} = 00000000$
 $[-0]_{\text{原}} = 10000000$
- 原码的特点：
原码表示简单直观，但0的表示不唯一，加减运算复杂。
- 8位二进制数的原码能表示的范围为
 $11111111\text{B} \sim 01111111\text{B}$
 $(-127 \sim +127)$ 即 $(2^n > X > -2^n)$

原码加减

原码加、减是指操作数与运算结果均用原码表示，运算时令尾数进行加、减，符号位单独处理。因此它实际上执行的操作不仅取决于指令操作码(加或减)，还与两操作数的符号有关，运算结果的符号位形成逻辑也比较复杂。我们将两操作数加、减运算的八种可能归并为四类实际操作，若A、B均为绝对值，则有：

$$(+A) + (+B) = (+A) - (-B)$$

$$(-A) + (-B) = (-A) - (+B)$$

$$(+A) + (-B) = (+A) - (+B)$$

$$(-A) + (+B) = (-A) - (-B)$$

前两类是同号相加或异号相减，它们需要做的操作是绝对值(即尾数)相加，结果符号位取被加(减)数的数符。后两类是异号相加或同号相减，它们需做的实际是绝对值相减，按是否够减决定结果数符。

减法的实现可采用将负数变反相加的方法实现，规则如下：若实际操作为两异号数相加，则正数取其原码，负数取其反码，然后两数相加；相加后若最高位有进位产生，则末位再加1，结果为正；相加后若最高位无进位产生，则将结果变反，数符为负。

例:

$$9 + 3 = 12$$

1001

+ 0011

1100

加符号位01100

$$(-9) + (-3) = -12$$

1001

+ 0011

1100

加符号位11100

$$9 - 3 = 6$$

1001

+ 11100(-3变反)

10101(有进位)

+ 1

0110

00110 加符号

$$3 - 9 = -6$$

0011

+ 10110(-9变反)

1001(无进位)

0110(结果变反)

10110 加符号

根据上述运算规则，可推导出操作命令(+A、 $+\overline{A}$ 、+B、 $+\overline{B}$ 、末位加1，取反)，以及结果数符形成逻辑，从而构成原码加法器硬件逻辑。由于原码加、减比补码加、减复杂，用得较少，我们重点放在补码加减法。

2. 反码

- 反码的表示方法：
正数的反码与原码表示形式相同。
负数的反码的符号位为1，数值位按位取反。
- 0的反码有两种表示形式。
 $[+0]_{\text{反}} = 00000000$
 $[-0]_{\text{反}} = 11111111$
- 8位二进制数的反码能表示的范围为
 $10000000\text{B} \sim 01111111\text{B}$
 $(-127 \sim +127)$

3. 补码

- 补码的表示方法：
正数的补码表示与原码表示形式相同。
负数的补码的符号位为1，数值位按位求反加1。
- 0的补码只有一种表示形式：
 $[+0]_{\text{补}} = [-0]_{\text{补}} = 00000000$
- 补码的特点：
0的表示唯一，加减运算方便。
- 8位二进制数的补码能表示的范围为
 $10000000\text{B} \sim 01111111\text{B}$
 $(-128 \sim +127)$ 即 $(2^n > X \geq -2^n)$

4. 有符号数的运算

- 补码的加法运算规则:

$$[X+Y]_{\text{补}} = [X]_{\text{补}} + [Y]_{\text{补}}$$

- 补码的减法运算规则:

$$[X-Y]_{\text{补}} = [X]_{\text{补}} - [Y]_{\text{补}} = [X]_{\text{补}} + [-Y]_{\text{补}}$$

- 在计算机中有符号数一般用补码表示，无论是加法还是减法都可采用加法运算，而且是连同符号位一起进行运算的，运算的结果仍为补码。

$[X+Y]_{\text{补}} = [X]_{\text{补}} + [Y]_{\text{补}}$ 它表明:

当操作码要求做加法运算时, 可直接将补码表示的两操作数相加, 不必考虑它们的数符是正或负, 所得结果即为补码表示的和。

$[X-Y]_{\text{补}} = [X]_{\text{补}} - [Y]_{\text{补}} = [X]_{\text{补}} + [-Y]_{\text{补}}$ 表明:

当操作码要求做减法运算时, 可转换为与减数的负数相加, 从而化“减”为“加”。 $(-Y)_{\text{补}}$ 是 $Y_{\text{补}}$ 的机器负数。由于 $Y_{\text{补}}$ 本身可正可负, $(-Y_{\text{补}})$ 也可能为负或正。由 $Y_{\text{补}}$ 求 $(-Y)_{\text{补}}$, 称为对 $Y_{\text{补}}$ 变补, 即将 $Y_{\text{补}}$ 连同符号位一起变反加1。如果减数 Y 为正, 则变补后可转换为加一个负数。如果减数 Y 为负, 则变补后可转换为加一个正数。

由此可归纳出补码加减法的运算规则：

- （1）参加运算的操作数用补码表示，符号位作为数的一部分直接参与运算，所得即为补码表示的运算结果；
- （2）若操作码为加，则两数直接相加；
- （3）若操作码为减，则将减数变补后再与被加数相加。

$$9 + 3 = 12$$

$$\begin{array}{r} 01001 \\ + \underline{00011} \\ \hline 01100 \end{array}$$

$$(-9) + (-3) = -12$$

$$\begin{array}{r} 10111 \text{ (-9补码)} \\ + \underline{11101} \text{ (-3补码)} \\ \hline 110100 \text{ (-12补码)} \end{array}$$

$$9 - 3 = 6$$

$$\begin{array}{r} 01001 \\ + \underline{11101} \text{ 对3变补} \\ 100110 \end{array}$$

$$9 - (-3) = 12$$

$$\begin{array}{r} 01001 \\ + \underline{00011} \text{ 对 } (-3) \text{ 变补} \\ 01100 \end{array}$$

$$3 - 9 = -6$$

$$\begin{array}{r} 00011 \\ + \underline{10111} \text{ 对9变补} \\ 11010 \end{array}$$

注意：在做减法运算时，我们把已知 $【Y】_{\text{补}}$ ，
求 $【-Y】_{\text{补}}$ 的过程叫做对“ $【Y】_{\text{补}}$ ”变补；

而把已知的某个数用它的补码形式表示叫做对它求补。

“变补”：将 $【Y】_{\text{补}}$ 连同符号位一起变反加1。

“求补”：符号位不变，数值位变反加1。

4. 有符号数的运算

【例8】 真值 $X1=+17$ ， $X2=+25$ ，求 $X1-X2$ 的值。

解： $[X1]_{\text{补}}=00010001$ $[-X2]_{\text{补}}=11100111$

$$[X1-X2]_{\text{补}}=[X1]_{\text{补}}+[-X2]_{\text{补}}=11111000$$

用二进制加法运算如下：

$$\begin{array}{r} 00010001 \\ + 11100111 \\ \hline 11111000 \end{array}$$

$$X1-X2=[[X1-X2]_{\text{补}}]_{\text{补}}=-8D$$

1. 3. 3 二进制编码

- 在计算机中对数字、字母和字符用二进制代码进行编码的方法很多，二进制数的位数越长，所能编码的数字、字母和字符就越多。
- 常用的二进制编码有BCD码、ASCII码等。

1. BCD码

- 用二进制编码表示的十进制数称为二-十进制数，简称BCD(Binary Coded Decimal)码。
- BCD码保留了十进制的权，用四位二进制数给0~9这10个数字编码。
- BCD码种类较多，如有8421码、2421码和余3码等。
- 最常用的是 8421BCD码(以后简称BCD码)，组成它的4位二进制数码的权分别是8、4、2、1。

8421BCD码与十进制数的对应关系表

十进制数	8421码	十进制数	8421码	十进制数	8421码
0	0000	6	0110	12	00010010
1	0001	7	0111	13	00010011
2	0010	8	1000	14	00010100
3	0011	9	1001	15	00010101
4	0100	10	00010000	16	00010110
5	0101	11	00010001	17	00010111

1. BCD码

- **BCD码的加减法运算与十进制运算规则相同，加法为逢十进一、减法为借一为十。**
- **BCD码加法运算的修正原则：**
若和的低4位大于9或低4位向高4位有进位，则低4位加6；若高4位大于9或高4位向最高位有进位，则高4位加6。
- **BCD码减法运算的修正原则：**
若差的低4位大于9或低4位向高4位有借位，则低4位减6；若高4位大于9或高4位向最高位有借位，则高4位减6。

2. ASCII码

- **ASCII码 (American Standard Coded for Information Interchange)** 是“美国信息交换标准代码”的简称，已成为国际通用的标准编码。
- **ASCII码采用7位二进制编码，可为128个字符编码，这128个字符分为两类。**
 - **图形字符——包括10个十进制数符、52个大小写英文字母和34个其它字符，共计96个。图形字符具有特定的形状，可以在显示器上显示。**
 - **控制字符——包括回车、换行、退格等，共32个。控制字符没有特定的形状但有一定的控制作用，不能在显示器上显示。**

ASCII码字符表

字 符 位 低 高		0	1	2	3	4	5	6	7
		000	001	010	011	100	101	110	111
0	0000	NUL	DLE	SP	0	@	P	、	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	“	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	\$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	‘	7	G	W	g	w
8	1000	BS	CAN	(	8	H	X	h	x
9	1001	HT	EM	)	9	I	Y	j	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[	k	{
C	1100	FF	FS	,	<	L	\	l	
D	1101	CR	GS	-	=	M	]	m	}
E	1110	SO	RS	.	>	N	↑	n	~
F	1111	SI	US	/	?	O	←	o	DEL

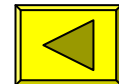
ASCII码		字符	控制字符	意义	ASCII码		字符	控制字符	意义
十进制	十六进制				十进制	十六进制			
000	00		NULL		016	10	▶	DLE	
001	01	☺	SOH		017	11	◀	DC1	
002	02	☹	STX		018	12	↕	DC2	
003	03	♥	ETX		019	13	❗	DC3	
004	04	♦	EOT		020	14	⌋	DC4	
005	05	♣	ENQ		021	15	§	NAK	
006	06	♠	ACK		022	16	—	SYN	
007	07	•	BELL	振铃	023	17	⌚	ETB	
008	08	◻	BS	退格键	024	18	↑	CAN	
009	09		HT	定位键	025	19	↓	EM	
010	0A		LF	line feed	026	1A	→	SUB	档案结束
011	0B	♂	VT	home	027	1B	←	ESC	escape
012	0C	♀	FF	form feed	028	1C	└	FS	向右键
013	0D		CR	carriage return	029	1D	↔	GS	向左键
014	0E	🎵	SO		030	1E	▲	RS	向上键
015	0F	☀	SI		031	1F	▼	US	向下键

ASCII 码		字符	ASCII 码		字符	ASCII 码		字符	ASCII 码		字符
十进制	十六进制		十进制	十六进制		十进制	十六进制		十进制	十六进制	
032	20		056	38	8	080	50	P	104	68	h
033	21	!	057	39	9	081	51	Q	105	69	i
034	22	"	058	3A	:	082	52	R	106	6A	j
035	23	#	059	3B	;	083	53	S	107	6B	k
036	24	\$	060	3C	<	084	54	T	108	6C	l
037	25	%	061	3D	=	085	55	U	109	6D	m
038	26	&	062	3E	>	086	56	V	110	6E	n
039	27	'	063	3F	?	087	57	W	111	6F	o
040	28	(	064	40	@	088	58	X	112	70	p
041	29	)	065	41	A	089	59	Y	113	71	q
042	2A	*	066	42	B	090	5A	Z	114	72	r
043	2B	+	067	43	C	091	5B	[	115	73	s
044	2C	,	068	44	D	092	5C	\	116	74	t
045	2D	-	069	45	E	093	5D	]	117	75	u
046	2E	.	070	46	F	094	5E	^	118	76	v
047	2F	/	071	47	G	095	5F	_	119	77	w
048	30	0	072	48	H	096	60	`	120	78	x
049	31	1	073	49	I	097	61	a	121	79	y
050	32	2	074	4A	J	098	62	b	122	7A	z
051	33	3	075	4B	K	099	63	c	123	7B	{
052	34	4	076	4C	L	100	64	d	124	7C	
053	35	5	077	4D	M	101	65	e	125	7D	}
054	36	6	078	4E	N	102	66	f	126	7E	~
055	37	7	079	4F	O	103	67	g	127	7F	☐

ASCII码		字符	ASCII码		字符	ASCII码		字符	ASCII码		字符
十进制	十六进制		十进制	十六进制		十进制	十六进制		十进制	十六进制	
128	80	Š	160	A0	á	192	C0	Ł	224	E0	α
129	81	ŭ	161	A1	í	193	C1	ł	225	E1	β
130	82	é	162	A2	ó	194	C2	ŧ	226	E2	γ
131	83	â	163	A3	ú	195	C3	ţ	227	E3	π
132	84	ä	164	A4	ñ	196	C4	—	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	†	229	E5	σ
134	86	â	166	A6	ª	198	C6	‡	230	E6	μ
135	87	ç	167	A7	º	199	C7	‡	231	E7	τ
136	88	ê	168	A8	¿	200	C8	Ł	232	E8	φ
137	89	ë	169	A9	ƒ	201	C9	Œ	233	E9	θ
138	8A	è	170	AA	¬	202	CA	℥	234	EA	Ω
139	8B	ï	171	AB	½	203	CB	₣	235	EB	δ
140	8C	î	172	AC	¼	204	CC	℥	236	EC	∞
141	8D	ì	173	AD	ı	205	CD	=	237	ED	φ
142	8E	Ä	174	AE	«	206	CE	≠	238	EE	€
143	8F	Å	175	AF	»	207	CF	±	239	EF	∩
144	90	É	176	B0	⋮	208	D0	℥	240	F0	≡
145	91	æ	177	B1	⋮	209	D1	₣	241	F1	±
146	92	Æ	178	B2	⋮	210	D2	₣	242	F2	≥
147	93	ô	179	B3		211	D3	℥	243	F3	≤
148	94	ö	180	B4	†	212	D4	℥	244	F4	∫
149	95	ò	181	B5	‡	213	D5	₣	245	F5	∫
150	96	û	182	B6	‡	214	D6	₣	246	F6	÷
151	97	ù	183	B7	₣	215	D7	≠	247	F7	≈
152	98	ÿ	184	B8	₣	216	D8	≠	248	F8	°
153	99	Ö	185	B9	‡	217	D9	┘	249	F9	•
154	9A	Ü	186	BA		218	DA	┐	250	FA	·
155	9B	Ŧ	187	BB	₣	219	DB	■	251	FB	√
156	9C	£	188	BC	┘	220	DC	■	252	FC	ⁿ
157	9D	¥	189	BD	┘	221	DD	■	253	FD	²
158	9E	₪	190	BE	┘	222	DE	■	254	FE	■
159	9F	ƒ	191	BF	┘	223	DF	■	255	FF	

1.4 微型计算机的结构与原理

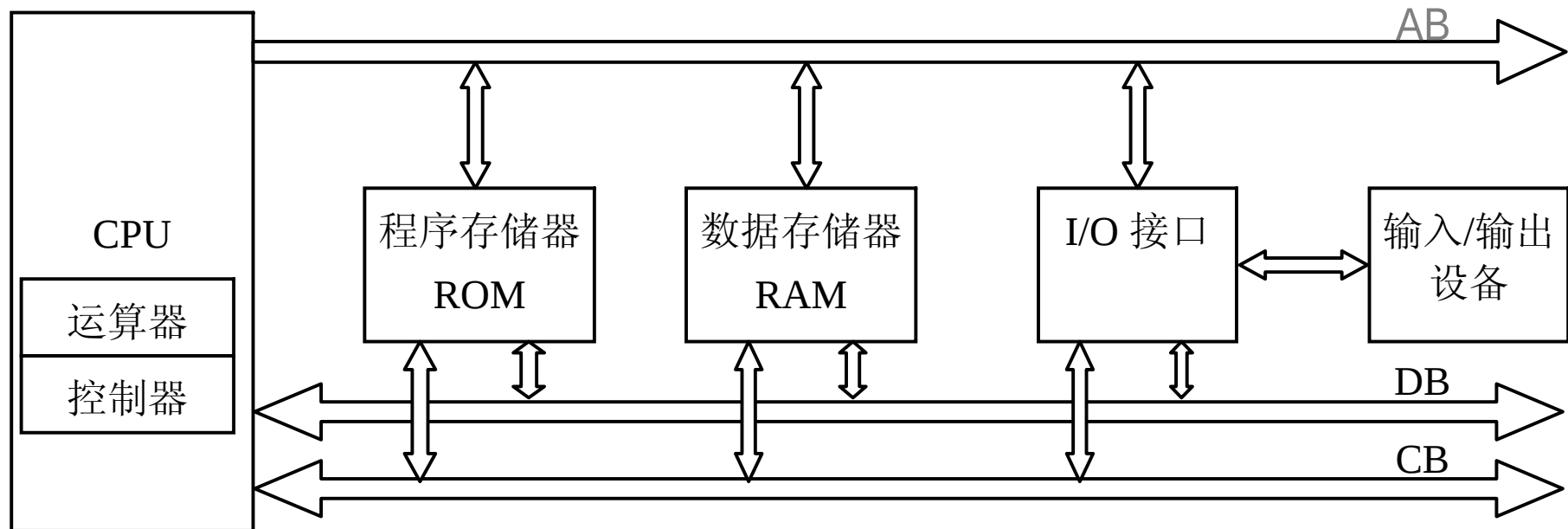
- 1.4.1 微型计算机的基本结构
- 1.4.2 微型计算机的工作原理



1.4.1 微型计算机的基本结构

- 微型计算机由硬件系统和软件系统两大部分组成，一般把二者构成的系统称为微型计算机系统。
- 微型计算机的硬件主要是由CPU(运算器和控制器)、存储器、I/O接口和I/O设备组成，各组成部分之间通过地址总线AB (Address Bus)、数据总线DB (Data Bus)、控制总线CB (Control Bus)联系在一起。
- 微型计算机的软件包括系统软件和应用软件两大类。软件与硬件相辅相成，共同构成微型计算机系统，缺一不可。

微型计算机的系统结构框图



- 微型计算机系统采用总线结构形式。
- 总线结构的主要优点：设计简单、灵活性好、易于扩展、便于故障检测和维修。

1.4.2 微型计算机的工作原理

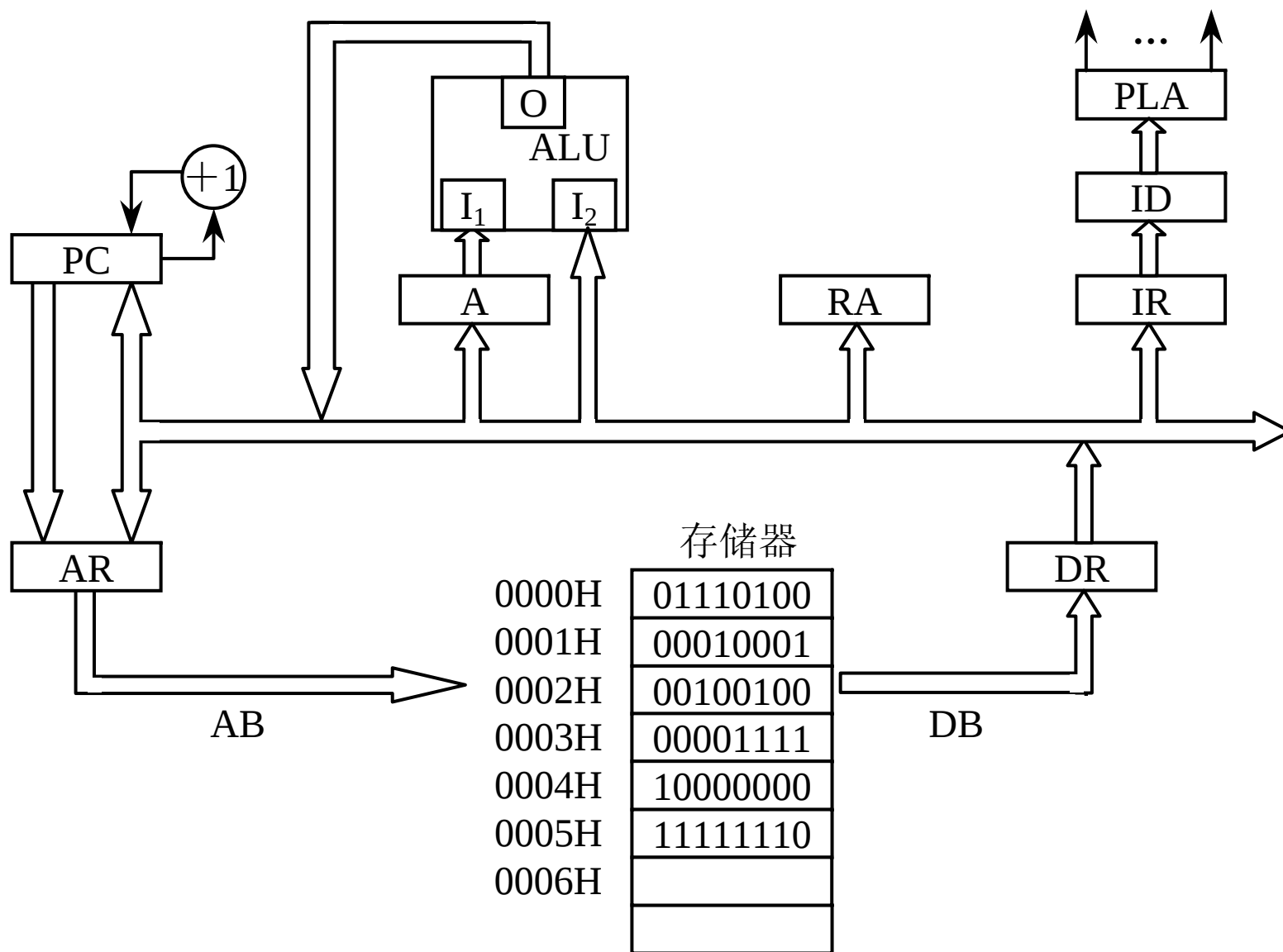
- 指令是对计算机发出的一条条工作命令，命令它执行规定的操作，程序是实现既定任务的指令序列。
- 把程序和数据送到具有记忆功能的存储器中保存起来，计算机工作时只要给出程序中第一条指令的地址，控制器就可依据存储程序中的指令顺序周而复始地取出指令、分析指令、执行指令，直到执行完全部指令为止。

微型计算机执行程序的过程

- 微型计算机执行程序的过程就是逐条执行指令的过程。由于执行每一条指令，都包括取指令与执行指令两个基本阶段，所以，微机的工作过程，也就是不断地取指令和执行指令的过程。
- **17+15** 的加法运算程序如下：

存储地址	机器语言程序	汇编语言程序	注释
0000H	74 11	MOV A,#11H	;数11H送入累加器A
0002H	24 0F	ADD A,#0FH	;数0FH与累加器A中的 11H相加,和送入累加器A
0004H	80 FE	SJMP \$	;暂停

执行程序的过程示意图



执行第一条指令的过程（一）

- 开始执行程序时，必须先给程序计数器PC赋以第1条指令的首地址0000H，然后就进入第一条指令的取指令阶段。
- 取第一条指令的步骤如下：
 - ① 把程序计数器PC的内容0000H送到地址寄存器AR。
 - ② 程序计数器PC的内容送入AR后，PC自动加1，即由0000H变为0001H。此时AR的内容并没有变化。
 - ③ 把地址寄存器AR的内容0000H通过地址总线AB送至存储器，经地址译码器译码，选中存储器相应的0000H单元。
 - ④ CPU控制器发出读命令。
 - ⑤ 在读命令控制下，把所选中的0000H单元中的内容即第1条指令的操作码74H读到数据总线DB上。
 - ⑥ 把读出的内容74H经数据总线DB送到数据寄存器DR。
 - ⑦ 因取出的是指令的操作码，故把数据寄存器DR的内容74H送到指令寄存器IR，然后再送到指令译码器ID。这就完成了第一条指令的取指令阶段，然后转入第一条指令的执行阶段。

执行第一条指令的过程（二）

- 经过对操作码**74H**译码后，**CPU**识别出这个操作码是把下一个存储单元中的操作数送入累加器**A**的双字节指令，所以，执行第一条指令就必须把指令第**2**字节中的操作数取出来。
- 取第一条指令操作数的步骤如下：
 - ① 把程序计数器**PC**的内容**0001H**送到地址寄存器**AR**。
 - ② 程序计数器**PC**的内容送到**AR**后，**PC**自动加1，变为**0002H**。此时**AR**中的内容并没有变化。
 - ③ 把地址寄存器**AR**的内容**0001H**通过地址总线**AB**送至存储器，经地址译码器译码，选中存储器相应的**0001H**单元。
 - ④ **CPU**控制器发出读命令。
 - ⑤ 在读命令控制下，将选中的**0001H**单元的内容**11H**读到数据总线**DB**上。
 - ⑥ 把读出的内容**11H**经数据总线**DB**送到数据寄存器**DR**。
 - ⑦ 因**CPU**已知这时读出的是操作数，且指令要求把它送到累加器**A**，故把数据寄存器**DR**的内容**11H**经内部数据总线送到累加器**A**中，于是第一条指令执行完毕，进入第二条指令的取指令阶段。

本章小结

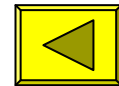
- 计算机按规模、性能、用途和价格来分类，可分为巨、大、中、小、微型计算机。计算机技术已发展成通用计算机系统 and 嵌入式计算机系统两大分支，嵌入式系统一般可分为工业控制计算机、通用CPU模块、嵌入式微处理器和嵌入式微控制器四类。单片机是经典的嵌入式系统，它具有专门为嵌入式应用设计的体系结构与指令系统。
- 单片机主要特点：集成度高、控制功能强、可靠性高、低功耗、低电压、外部总线丰富、功能扩展性强、体积小、性价比高。单片机按数据处理位数可分为4位机、8位机、16位机和32位机，按适用范围可分为通用型和专用型，按并行总线可分为总线型和非总线型。

本章小结

- 在计算机中常用的数制有十进制、二进制和十六进制。不同数制之间的转换都有一定的规则，如二进制数转换成十进制数采用“四位合一位”法，十六进制数转换成二进制数采用“一位分四位”法，十进制整数转换成二进制整数采用“除2取余”法，十进制整数转换成十六进制整数采用“除16取余”法，十进制小数转换成二进制小数采用“乘2取整”法，十进制小数转换成十六进制小数采用“乘16取整”法。
- 有符号二进制数有三种表示法，即原码、反码和补码。在计算机中有符号数一般用补码表示，无论是加法还是减法都可采用加法运算，而且是连同符号位一起进行的，运算的结果仍为补码。
- **BCD**码的加减法运算与十进制运算规则相同，但必须对运算结果进行修正。**ASCII**码是国际通用的标准编码，采用7位二进制编码，分为图形字符和控制字符两类，共**128**个字符。

本章小结

- 微型计算机由硬件系统和软件系统两大部分组成。硬件主要是由**CPU**、存储器、**I/O**接口和**I/O**设备组成，采用总线结构形式。软件包括系统软件和应用软件两大类，程序设计语言分为三级，分别是机器语言、汇编语言和高级语言。
- 微型计算机执行程序的过程就是逐条执行指令的过程。执行每一条指令，都包括取指令与执行指令两个基本阶段。

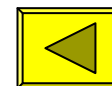


习 题

1. 十进制、二进制、十六进制数各有什么特点?请举例加以说明。
2. 将下列十进制数转换为二进制和十六进制数。
①255 ②127 ③0.123 ④0.1415 ⑤45.32 ⑥1999.08
3. 将下列二进制和十六进制数转换为十进制数。
①1001010B ②0.1010111B ③1010.11B ④1C3H ⑤0.A49H
⑥EB.0AH
4. 将下列二进制数转换为十六进制数。
①1011011B ②0.1010101B ③1010.0101B
④111000010100.1110101B
5. 机器数、真值、原码、反码和补码如何表示?请举例加以说明。
6. 下列是有符号数的原码,请写出其反码和补码。
①01111011B ②10101010B ③10000001B ④11111111B

习 题

7. 把下列数看成无符号数时，对应的十进制数为多少？若把它们看成有符号数的补码，对应的十进制数是多少？
①10101010B ②00110010B ③10000001B
8. 把下列各数用补码表示，再按补码运算规则分别求出 $[X+Y]_{\text{补}}$ 和 $[X-Y]_{\text{补}}$ 。
① $X=+46$ $Y=-78$ ② $X=+112$ $Y=-51$
9. 写出89和157两数的BCD码，并对这两个BCD码进行加法运算。
10. 微型计算机系统的硬件和软件包括哪些部分？
11. 结合图1-2简述微型计算机执行一条指令的过程。



Q & A?

Thanks!

