21世纪高等学校规划教材
Textbook Series of 21st Century

单片机原理及应用习题与实验指导书

王义军 主编
刘晓峰 盛文利 副主编

中国电力出版社
http://jc.cepp.com.cn

21世纪高等学校规划教材
Textbook Series of 21st Century

单片机原理及应用习题与实验指导书

主　编　王义军
副主编　刘晓峰　盛文利
编　写　刘彦臣　杨卫华
主　审　任先文

中国电力出版社
http://jc.cepp.com.cn

内 容 提 要

本书为21世纪高等学校规划教材。

本书为MCS-51单片机原理及应用的学习辅导书和相关的实验指导书。

全书共分两篇9章。其主要内容为微型计算机基础知识、MCS-51单片机组成及结构分析、MCS-51单片机汇编语言程序设计、MCS-51单片机内部接口电路、MCS-51系统扩展、指令与寻址方式的验证、单片机内部接口设备的使用、典型子程序设计、外部接口应用。

本书为电类在校本科生和任课教师编写的学习指导和实验指导书，也可作为相关工程技术人员的参考书。

图书在版编目（CIP）数据

单片机原理及应用习题与实验指导书/王义军主编. —北京：中国电力出版社，2006

21世纪高等学校规划教材

ISBN 7-5083-4371-9

Ⅰ.单... Ⅱ.王... Ⅲ.单片微型计算机—高等学校—教学参考资料 Ⅳ.TP368.1

中国版本图书馆CIP数据核字（2006）第045951号

中国电力出版社出版、发行

（北京三里河路6号 100044 http://jc.cepp.com.cn）

北京同江印刷厂印刷

各地新华书店经售

*

2006年7月第一版 2006年7月北京第一次印刷

787毫米×1092毫米 16开本 17.75印张 406千字

印数 0001—3000册 定价 **27.00**元

前　言

自20世纪70年代MCS-51单片机推出以来，以其很高的性能价格比、容易掌握的操作，很快成为电子技术工程师们工程设计时的首选芯片。20多年来，MCS-51单片机的性能不断完善，目前仍保持其强大的生命力。这在大规模、超大规模集成电路发展史上是难能可贵的。因此，国内很多工科院校把“单片机原理及应用”作为必修课程，这门课程的教材也出版了几部，但是关于这门课程学习指导以及实验指导方面的书籍却很少出版，给教学和实验带来了一定的困难。为此，编者总结几年来教学和实践经验，编写了这本教材，供大专院校的老师、学生们以及有关技术人员参考。

本书共分两部分，第一部分是单片机知识的学习辅导，第二部分是实验指导书。为了能够适应各种版本的教科书，本书不参考任何教科书的分章方法，而是从MCS-51单片机本身的特点出发，以MCS-51单片机的知识点为核心，分五部分介绍相应内容的学习方法以及相关习题解答。现有的开发系统大多是基于Windows操作系统的可视化人机界面，功能基本相同。本书以目前使用较广泛的QTH系列单片机仿真开发系统为例，介绍仿真软件的使用方法，另外，还利用MCS-51单片机提供的伪指令解决实验板基本器件种类相同而地址不同的问题。上述问题的解决，使本教材能适应更广泛读者的需求。

本教材由东北电力大学王义军副教授担任主编，刘晓峰、盛文利老师担任副主编。王义军副教授编写了本书的第一部分，刘晓峰、盛文利两位老师编写了本书的第二部分，全书由王义军统稿。在本教材的编写过程中，得到东北电力大学任先文教授的亲切关怀和指导，并且也得到杨卫华老师、刘彦臣老师、李辉老师、石磊老师的友情帮助。本教材由任先文教授主审，并对本教材的编写提出了许多宝贵意见。在此我们一并表示感谢。

由于我们水平有限，难免有不足之处，请读者谅解，并提出宝贵意见。

编者

2005.12

目　录

第二篇 实 验 指 导

第一篇　学习指导和习题解答

在以往的教学过程中发现学生对于学习这门课程存在很多障碍，经常会问一些无言以对、莫名其妙的问题，这些现象反映出学生对所学知识的渴望和对课程特点的不明了。这本教材就是教会学生学习这门课程的方法。总结成一句话就是“单片机原理及应用”这门课程自成系统，对它的学习是在记忆基础上的理解和灵活应用。

所要记忆的内容包括：单片机的内部资源的使用方法、硬件的连线规则、指令系统等，了解这些知识后，如何把这些资源协调使用还要靠技巧和经验。技巧和经验要通过实际动手才能得到，因此，要想学好单片机必须做实验，并且保证上机时间和做好实验的预习准备工作，在每次实验中要验证的内容和改变方法的再重新尝试都是非常重要的。

第1章　微型计算机基础知识

1.1　计算机中的数和编码系统

计算机的最基本功能是进行数的计算和处理加工。数在微型计算机中是以器件的物理状态来表示的，为了使其表示得更为方便和可靠，在计算机中采用二进制数字系统。计算机处理的所有的数，都要用二进制数字系统来表示。因此，研究计算机基础要从讨论二进制计数系统开始。

1.1.1　进位计数制

在日常生活中，可遇到各种进位计数制，如：十进制、十二进制、十六进制、六十进制等。最常用的是十进制，而计算机中表示数的方法使用的是二进制数。可以把上述几种计数制的特点概括为以下两点。

(1) 每一种计数制都有一个固定的基数 r，它的每一位可能取 $0 \sim r-1$，r 个不同的数值。

(2) 具有 n 位整数和 m 位小数的 r 进制数 D 按权展开

$$D = d_{n-1} \times r^{n-1} + d_{n-2} \times r^{n-2} + \cdots + d_1 \times r^1 + d_0 \times r^0 + d_{-1} \times r^{-1} + d_{-2} \times r^{-2} + \cdots + d_{-m} \times r^{-m}$$

$$D = \sum_{i=-m}^{n-1} d_i \times r^i \tag{1-1}$$

它是逢“r”进位的。因此，它的每一个数位 i 对应一个固定的值 r^i，r^i 就称为该位的“权”，小数点左面各位的权依次是基数 r 的正整数次幂；而小数点右面各位的权依次是基

数 r 的负整数次幂。与此相关，若小数点向左移一位（或数向右移一位），则等于减小了 r 倍；若小数点向右移一位，则等于增加了 r 倍。

1.1.2 进位制数之间的转换❶

1. 任意进制数转换成十进制数

这比较方便，只要将它按式（1-1）展开相加即可。

2. 十进制数转换成 r 进制数

（1）十进制整数转换为 r 进制整数。可概括出把十进制整数转换为 r 进制整数的方法：用 r 不断地去除要转换的十进制数，直至商为 0。每次的余数即为 r 进制数码，最初得到的为整数的最低有效数（LSD——Least Significant Digit）d_0，最后得到的为最高有效数（MSD——Most Significant Digit）d_{n-1}。

（2）十进制小数转换为 r 进制小数。可概括出十进制小数转换为 r 进制小数的办法：不断用 r 去乘要转换的十进制小数，将每次所得的整数（$0\sim r-1$），依次记为 d_{-1}、d_{-2}、…。若乘积的小数部分最后能为 0，那么最后一次乘积的整数部分记为 K_{-m}。则 $0.d_{-1}d_{-2}$、…、d_{-m} 即为十进制小数的 r 进制表达式。但十进制小数，并不都是能用有限位的 r 进制小数精确表示的。可根据精度要求取 m 位，得到十进制小数的 r 二进制的近似表达式。

3. 十六进制与二进制之间的转换

由于 1 位二进制数只能有两种状态，通常把四位二进制数当成一位数来对待，就构成了一种新的进位计数制——十六进制。在微型计算机中，目前通用的字长为 8 位（也有 16 位、32 位等），则可用两位十六进制数表示，故十六进制在微型机中应用十分普遍。

（1）十六进制转换为二进制。不论是十六进制的整数或小数，只要把每 1 位十六进制的数用相应的 4 位二进制数代替，就可以转换为二进制数。

（2）二进制转换为十六进制。二进制的整数部分由小数点向左，每 4 位一分，最后不足 4 位的前面补 0；小数部分由小数点向右，每 4 位一分，最后不足 4 位的后面补 0。然后把每 4 位二进制数用相应的十六进制数代替，即可转换为十六进制数。

1.1.3 二进制编码

同其他进位计数制一样，二进制数除了可以用来计数，还可以用特定的二进制码来表示其他的数、字母、符号等，这就是二进制编码。

1. 二进制编码的十进制数

1 位十进制数用 4 位二进制编码来表示，表示的方法可以很多，较常用的是 8421BCD（BCD—Binary Codeed Decimal）码，表 1-1 列出了一部分 8421BCD 码编码关系。

2. 字母与字符的编码

如上所述，字母和各种字符也可以按特定的规则用二进制编码在微型计算机中表示。编码也可以有各种方式——即规定。目前在微型机中最普遍的是采用 ASCII 码（American

---

❶ 以后用数字后面加字母 B（Binary）表示为二进制数；以字母 D（Decimal）或不加字母表示为十进制数；用字母 H（Hexadecimal）表示为 16 进制数。

Standard Code for Information Interchange 美国标准信息交换码)，编码表如附录五所示。它是用七位二进制对最多 128 个字符或符号进行编码，在计算机中表示时，常用一个字节来表示一个 ASCII 字符。例如 0～9 的 ASCII 码为 30H～39H；大写字母 A～Z 的 ASCII 码为 41H～5AH。

表 1-1　　8421BCD 码 编 码 表

十进制数	8421BCD 码	十进制数	8421BCD 码
0	0000	8	1000
1	0001	9	1001
2	0010	10	0001 0000
3	0011	11	0001 0001
4	0100	12	0001 0010
5	0101	13	0001 0011
6	0110	14	0001 0100
7	0111	15	0001 0101

3. 汉字编码

根据我国的国情，要求计算机能够对汉字进行处理。同样，汉字在计算机中也必须依靠二进制编码来实现。我国规定的中华人民共和国国家标准信息交换用汉字编码即国标码，采用两个 7 位的二进制来表示一个图形、符号和汉字。国标码中共收录汉字和图形、符号 7445 个，包括标点、运算符等一般符号 202 个、序号 60 个、数字 0～9、大小写英文字母、汉字拼音 26 个、汉字 6763 个、还有日文假名、希腊字母及俄文字母等。

1.1.4 二进制数的基本运算

1. 二进制加法

二进制加法的规则为

0+0=0

0+1=1+0=1

1+1=0　有进位 1

2. 二进制减法

二进制减法的运算规则为

0−0=0

1−1=0

1−0=1

0−1=1　有借位 1

3. 二进制乘法

二进制乘法的运算规则为

0×0=0

0×1=0

1×0=0

1×1=1

多位二进制的乘法也与十进制的类似，即

```
被乘数      1111
乘  数    ×1101
            1111
           0000
          1111
         1111
      11000011
```

用乘数的每一位去乘被乘数，乘得的中间结果的最低有效位与相应的乘数位对齐，最后把这些中间结果相加。在做相加运算时，如果最后一起相加不仅会出现向前一位进位的现象还有可能出现向前两位进位的现象，因此，为了不出现丢进位的现象经常采用边乘边相加的办法来求部分积的和。其方法为

```
被乘数      1111
乘  数    ×1101
            1111
          +0000
部分积      1111
         +1111
部分积    1001011
        +1111
乘  积  11000011
```

4. 二进制除法

除法是乘法的逆运算。与十进制的运算规则类似，从被除数的最高有效位 MSB 开始检查，并找出需要超过除数的位数。找到这个位时商记 1，并把选定的被除数值减除数。然后把被除数的下一位下移到余数上。如果余数不能减去除数（不够减）则商记 0，把被除数的再下一位移到余数上；若余数够减则商记 1，余数减去除数，把被除数的下一位移到余数上。这样继续下去，直到全部被除数的位都下移完为止。然后把余数/除数作为商的分数，表示在商中。例如

```
        000111
   101√100011
      101↓↓
      0111↓
       101↓
        101
        101
          0
```

1.1.5　逻辑运算

除了算术运算之外，微型机还要完成一些逻辑运算，如逻辑“与”、逻辑“或”、逻辑“非”等。逻辑运算都是位对位进行的，不存在进位或借位问题，即位与位之间是独立的互不相关的，因此比算术运算简单。

1. 逻辑“与”运算

逻辑“与”，简称“与”运算。

运算规则为

$0 \wedge 0=0$

$0 \wedge 1=1 \wedge 0=0$

$1 \wedge 1=1$

这里“∧”是“与”运算符号，“与”运算可以用来将一个数的一部分清“0”，而其他位不变。

2. 逻辑“或”运算

逻辑“或”运算简称“或”运算。它的运算规则是

$0 \vee 0=0$

$0 \vee 1=1 \vee 0=1$

$1 \vee 1=1$

这里“∨”是“或”运算符号，“或”运算可以用来将一个数的一部分置“1”，而其他位不变。

3. 逻辑“非”运算

逻辑“非”又称为“求反”。它的运算规则是

$\overline{0}=1$

$\overline{1}=0$

这里“0”或“1”上面的“一横”表示“非”运算，或说“求反”。

4. “异或”运算

“异或”运算又称为按位加，它的运算规则是

$0 \oplus 0=0$

$0 \oplus 1=1 \oplus 0=1$

$1 \oplus 1=0$

即相异出“1”，相同出“0”。这里“⊕”是“异或”运算符号。

“异或”运算可以比较出两个多位数的某位数是否相同。

1.1.6　带符号数的表示法

1. 机器数与真值

在机器中，数的大小用二进制表示，数的正负同样也要用二进制数来表示，那么符号是怎么表示的呢？通常是一个数的最高位为符号位。若字长为8位即 D_7 为符号位，$D_6 \sim D_0$ 为数字位。符号位用0表示正、用1表示负。

这样连同一个符号位在一起作为一个数，就称为机器数；而它的数值称为机器数的真值。在计算机中带符号数有三种表示法——原码、反码和补码。

2. 原码

按上所述，正数的符号位用0表示，负数的符号位用1表示，其他各位为数值位。这种表示法就称为原码。

3. 反码

正数的反码表示与原码相同，最高位为符号位，用“0”表示正，其余位为数值位。而负数的反码表示，即为它的正数按位取反（连符号位）而形成的。

（1）“0”有两种表示法。

（2）8位二进制反码所能表示的数值范围为－127～＋127。

（3）当一个带符号数由反码表示时，最高位为符号位。当符号为0（即正数）时，后面的七位为数值部分；但当符号位为1（即负数）时，一定要注意后面几位表示的不是此负数的数值，一定要把它们按位取反，才表示它的二进制数。

4. 补码

正数的补码表示与原码相同，即最高位为符号位，用“0”表示正，其余位为数值位。而负数的补码为它的反码在最后位加1所形成。表1-2为数的表示法。

表1-2　　数的表示法

二进制数码表示	无符号二进制数	原码	补码	反码
00000000	0	＋0	＋0	＋0
00000001	1	＋1	＋1	＋1
00000010	2	＋2	＋2	＋2
⋮	⋮	⋮	⋮	⋮
01111100	124	＋124	＋124	＋124
01111101	125	＋125	＋125	＋125
01111110	126	＋126	＋126	＋126
01111111	127	＋127	＋127	＋127
10000000	128	－0	－128	－127
10000001	129	－1	－127	－126
10000010	130	－2	－126	－125
⋮	⋮	⋮	⋮	⋮
11111100	252	－124	－4	－3
11111101	253	－125	－3	－2
11111110	254	－126	－2	－1
11111111	255	－127	－1	－0

8位带符号位的补码表示也列在表1-2中。它有以下特点。

（1）$[+0]_{补}=[-0]_{补}=00000000$。

（2）8位二进制补码所能表示的数值为：＋127～－128。

（3）一个用补码表示的二进制数，最高位为符号位，当符号位“0”（即正数）时，其余七位即为此数的二进制数；但当符号位为“1”（即负数）时，其余几位不是此数的二进制数，把它们按位取反，且在最低位加1，才是它的二进制数。

在计算机中都利用多位二进制数组合起来表示多种信息，比较常用的表示数据的单位有位、字节、字、双字等，一个字节是8位，一个字是两个字节，双字就是四个字节。在MCS-51单片机中存储数据以字节为单位。

由于计算机的字长是一定限制的，所以一个带符号数是有一定的范围的，当运算的结果超出这个表达范围时，结果就不正确了，这种现象称为溢出。这时数就要用更多字节表示，例如用两个或四个字节等来表示这些数。

习题与思考题解题思路与方法

【习题1-1】将下列各二进制数转换为十进制数。

（1）11010101B；（2）11010011B；（3）10101011B；（4）10111101B。

【解答】将任一进制的数据变成十进制数只需代入公式 $D=\sum_{i=-m}^{n-1} D_i r^i$

其中，r 是数制的基数；D_i 可取0～$r-1$，m 为小数位数；n 为整数的位数。

（1）11010101B＝213D；（2）11010011B＝211D；（3）10101011B＝171D；（4）10111101B＝189D。

【习题1-2】将【习题1-1】中各二进制数转换为十六进制数。

【解答】将二进制数以小数点为基准，四位为一个单位整数部分不足四位高位用0代替，小数部分不足四位低位用0代替，然后将每个四位二进制数变成相应的16进制数即可。

（1）11010101B=1101，0101B=D5H；（2）11010011B=1101，0011B=D3H；（3）10101011B=1010，1011B=ABH；（4）10111101B=1011，1101B=BDH。

【习题1-3】将下列各数转换为二进制数。

（1）215D；（2）253D；（3）01000011BCD；（4）00101001BCD。

【解答】若要把十进制整数215转换成二进制数，即要变成如下的形式

$(215)_{10}=(K_{n-1}K_{n-2}\cdots K_1K_0)_2$

问题就是要找到 K_{n-1}～K_0 的值，而这些值不是1就是0，取决于要转换的十进制数。根据二进制的定义

$(K_{n-1}K_{n-2}\cdots\cdots K_1K_0)_2=K_{n-1}\times2^{n-1}+K_{n-2}\times2^{n-2}+\cdots+K_1\times2^1+K_0\times2^0$

$(215)_{10}=K_{n-1}\times2^{n-1}+K_{n-2}\times2^{n-2}+\cdots K_1\times2^1+K_0\times2^0$

显然，等式右边，除了最后一项 K_0 以外，都包含有2的因子，它们都能被2除尽。故，用2去除十进制数 $(215)_{10}$，则它的余数为 K_0，即

$$2\,|\,\underline{215}$$
$$107\qquad\cdots\cdots\text{余}\ 1=K_0$$

$(215)_{10}$ 被2除且去掉余数后为107

$(107)_{10}=K_{n-1}\times2^{n-2}+K_{n-2}\times2^{n-3}+\cdots K_2\times2^1+K_1$

显然，上面等式的右边除了最后一项 K_1 以外，都包含有 2 的因子，都能被 2 除尽，故若用 2 去除 107，所得的余数必为 K_1，即

2 | 107

53 ……余 $1=K_1$

用这样的方法一直继续下去，直到商为 0，就可以得到 K_{n-1}、K_{n-2}、…、K_1、K_0，即

2 | 215

2 | 107 ……余 $1=K_0$

2 | 53 ……余 $1=K_1$

2 | 26 ……余 $1=K_2$

2 | 13 ……余 $0=K_3$

2 | 6 ……余 $1=K_4$

2 | 3 ……余 $0=K_5$

2 | 1 ……余 $1=K_6$

0 ……余 $1=K_7$

$(215)_{10}=K_7K_6K_5K_4K_3K_2K_1K_0=(11010111)_2$

十进制可以用数字后边的字母“D”表示，在计算机中常用 8421BCD 码表示十进制数。按照上述方法可得

（2）253D = 11111101B；（3）01000011BCD = 43D = 00101011B；（4）00101001BCD = 29D=00011101B。

【习题 1 - 4】 什么叫原码、反码及补码？

【解答】 计算机中的带符号数有三种表示法，即：原码、反码和补码。

正数的符号位用 0 表示，负数的符号位用 1 表示，其他位不变，这种表示法称为原码。反码可由原码得到：如果是正数，则其反码和原码相同；如果是负数，则其反码除符号位为 1 外，其他各数位按位取反。补码可由反码得到：如果是正数，则其补码和反码相同；如果是负数，则其补码为反码加 1。

【习题 1 - 5】 已知原码如下，写出其反码和补码（其最高位为符号位）。

（1）$[X]_{原}$ =01011001B；（2）$[X]_{原}$ =00111110B；（3）$[X]_{原}$ =11011011B；（4）$[X]_{原}$ =11111100B。

【解答】 原码变反码和补码的方法：正数的补码、反码与原码相同；负数的反码是在符号位不变的情况下，其他位按位取反，负数的补码是在反码的基础上加 1。

（1）$[X]_{反}$ =01011001B；（2）$[X]_{反}$ =00111110B；

（3）$[X]_{反}$ =10100100B；（4）$[X]_{反}$ =10000011B；

（1）$[X]_{补}$ =01011001B；（2）$[X]_{补}$ =00111110B；

（3）$[X]_{补}$ =10100101B；（4）$[X]_{补}$ =10000100B。

【习题 1 - 6】 当微机把下列数看成无符号数时，它们相应的十进制数为多少？若把它们看成是补码，最高位为符号位，那么它们相应的十进制数是多少？

（1）10001110B；（2）10110000B；（3）00010001B；（4）01110101B。

【解答】 结合【习题 1 - 1】和【习题 1 - 5】的解题方法可以得到：

把（1）～（4）的数看成无符号数时，它们相应的十进制数为

（1）10001110=142D；（2）10110000=176D；（3）00010001=17D；（4）01110101=117D。

若把它们看成是补码，最高位为符号位，那么它们相应的十进制数为

(1) 10001110=−114D；(2) 10110000=−80D；(3) 00010001=17D；(4) 01110101=117D。

【习题 1-7】 用补码方法计算下列各式（设机器字长为 8 位）

(1) $X=7D$，$Y=8D$，求 $X+Y$。

(2) $X=5D$，$Y=9D$，求 $X-Y$。

(3) $X=6D$，$Y=-7D$，求 $X+Y$。

(4) $X=-11D$，$Y=7D$，求 $X-Y$。

【解答】 用补码方法进行加减运算时，计算规则：$[X]_{补}+[Y]_{补}=[X+Y]_{补}$；$[X]_{补}+[-Y]_{补}=[X-Y]_{补}\ MOD2^8$

(1) $X=7D$，$Y=8D$，$[X]_{补}=7D$，$[Y]_{补}=8D$，$[X+Y]_{补}=[X]_{补}+[Y]_{补}=15D$，因此 $X+Y=15D$。

(2) $X=5D$，$Y=9D$，$[X]_{补}=5D$，$[-Y]_{补}=256D-9D=F7H$，$[X-Y]_{补}=[X]_{补}+[-Y]_{补}=FCH$，因此，$X-Y=-4D$。

(3) $X=6D$，$Y=-7D$，$[X]_{补}=6D$，$[Y]_{补}=F9H$，$[X+Y]_{补}=[X]_{补}+[Y]_{补}=FFH$，因此 $X+Y=-1D$。

(4) $X=-11D$，$Y=7D$，$[X]_{补}=F5H$，$[-Y]_{补}=F9H$，$[X-Y]_{补}=[X]_{补}+[-Y]_{补}=EEH$（进位舍去，$MOD2^8$），因此 X−Y=−18D。

【习题 1-8】 已知 $X=01011001B$，$Y=00110101B$，用算术运算规则求：

(1) $X+Y$；(2) $X-Y$；(3) $X\times Y$；(4) X/Y。

【解答】 根据 1.1.4 节给出的二进制四则运算规律可以计算出结果

(1) $X+Y=10001110B$；(2) $X-Y=00100100B$。

乘法运算是移位加运算，在做多位二进制乘法运算时需要进行乘数位数次加法。如果最后一起相加，加法容易对错位，故常采用边乘边相加的办法。

在单片机中，除法运算通常只需算出商和余数即可。

(3) $X\times Y=1001001101101B$；(4) X/Y=商是 1，余数是 00100100B。

【习题 1-9】 已知 $X=01111010B$，$Y=10101010B$，用逻辑运算规则求

(1) $X\wedge Y$；(2) $X\vee Y$；(3) $X\oplus Y$；(4) $\overline{X}$。

【解答】 根据 1.1.5 节给出的逻辑运算规律可以计算出结果（与、或和异或运算在后续的字节位操作以及字节拼装中使用较多）：

(1) $X\wedge Y=00101010B$；(2) $X\vee Y=11111010B$；(3) $X\oplus Y=11010000B$；(4) $\overline{X}=10000101B$。

第2章　MCS-51单片机组成及结构分析

MCS-51系列单片机有一定容量的片内RAM，有较强的I/O口功能、系统扩展能力以及CPU处理功能。尤其是MCS-51所特有的布尔处理机，对于逻辑处理与控制具有突出的优点。由于MCS-51单片机具有体积小、功能全、价格低廉、开发应用方便、适合于实时控制等优点，已被广泛应用于工业控制器、智能仪表、智能接口、智能仪器装置以及通用测控单元等领域。

2.1　MCS-51单片机主要功能特点

MCS-51系列单片机分为51子系列和52子系列，在51子系列中，典型代表为8751、8051、8031等，在52子系列中，典型代表为8752、8052、8032等，这两个子系列单片机的指令系统与引脚功能完全兼容，仅在内部功能部件稍有差异，51子系列单片机的主要功能特点如下：

（1）8位CPU。

（2）布尔处理器。

（3）片内128字节RAM（52子系列有256字节RAM）。

（4）片内4KB ROM/EPROM（8051/8751）。

（5）特殊功能寄存器区。

（6）64KB外部数据存储器地址空间。

（7）64KB外部程序存储器地址空间。

（8）2个优先级的5个中断源（52子系列有6个中断源）结构。

（9）4个8位并行I/O口（P0、P1、P2、P3）。

（10）2个16位定时/计数器（52子系列为3个）。

（11）1个全双工串行口。

（12）片内振荡器及时钟电路。

MCS-51系列单片机片内功能配置如表2-1所示。

表2-1　　MCS-51系列单片机片内功能配置

芯片种类 \ 功能配置		ROM	EPROM	RAM	定时器计数器	I/O口		中断源
						并行	串行	
51	8031	—	—	128B	2×16位	4×8位	1	5
	8051	4K	—	128B	2×16位	4×8位	1	5
	8751	—	4K	128B	2×16位	4×8位	1	5