

文章编号: 1671-7848(2007)04-0419-04

## 单片机系统中标志位的应用

杨晓玲<sup>1,2</sup>, 王克明<sup>1</sup>, 苏 玮<sup>1</sup>, 范同顺<sup>1</sup>

(1. 北京联合大学 自动化学院, 北京 100101; 2. 北京化工大学 信息科学与技术学院, 北京 100101)



**摘 要:** 简要介绍了 MCS-51 系列单片机中的位寻址区和位操作指令。位寻址区和位操作指令构成了 MCS-51 系列单片机独有的布尔处理机。布尔处理机实际上是一个完整的一位微计算机, 在开关决策、逻辑电路仿真和实时控制方面非常有效。位寻址区和位操作指令的一个最重要的应用就是用于设置标志位。这一特点使得单片机系统在开关决策、逻辑电路仿真、智能检测和实时控制方面的到了非常广泛的应用。通过一个实例给出了利用标志位来判断按键是否被按下的算法。

**关键词:** 位操作指令; 位寻址区; 标志位; 单片机系统  
中图分类号: TP 368 文献标识码: A

## Application of Bit-flags in Single-chip Microcomputer System

YANG Xiao-ling<sup>1,2</sup>, WANG Ke-ming<sup>1</sup>, SU Wei<sup>1</sup>, FAN Tong-shun<sup>1</sup>

(1. School of Automation, Beijing Union University, Beijing 100101, China;

2. School of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100101, China)

**Abstract:** The bit operation instruments and bit addressable locations in MCS-51 series microcontroller are introduced. Bit addressable locations and bit operation instruments compose the boolean processor of MCS-51 series microcontroller. The boolean processor is an integrated 1-bit-micro computer. An important use of them is set bit-flags, and this characteristic makes single-chip microcomputer system a wide application in switch-decision making, logical circuit simulation, intelligent detection and real-time control. An algorithm of using bit-flags is presented to detect if a key is pressed or not with an actual example.

**Key words:** bit operation instruments; bit addressable locations; bit-flag; single-chip microcomputer system

### 1 引言

随着微电子技术的不断发展, 微处理器芯片的集成度越来越高, 已经可以在一块芯片上同时集成 CPU、存储器、定时器/计数器、并行和串行接口, 甚至 A/D 转换器等。人们把这种超大规模集成电路芯片称作“单片微控制器”(Single Chip Microcontroller), 简称单片机。

单片机的出现引起了仪器仪表结构的根本性变革。以单片机为主体取代传统仪器仪表的常规电子线路可以容易地将计算技术与测量控制技术结合在一起, 组成新一代的所谓“智能化测量控制仪表”。

在测量控制仪表中采用单片机技术使之成为智能仪表后能够解决许多传统仪表不能或不易解决的难题, 同时还能简化仪表电路, 提高仪表的可靠性, 降低仪表的成本以及加快新产品的开发速度。因此以单片机为核心的智能化测量控制仪表得到了越来越广泛的应用。

### 2 MCS-51 系列单片机简介

MCS-51 系列单片机是上世纪 80 年代由美国 Intel 公司推出的一种高性能的 8 位单片机。其片内集成了并行 I/O 口、16 位定时器/计数器、片内有 256B 的 RAM 和 4~8 KB 的 ROM, 尤其适合于低成本、大批量生产的场合。目前 MCS-51 系列单片机已有许多品种, 典型的为 8051 和 8071 等。

单片机的应用特点是“面向测控”。MCS-51 系列单片机共有 7 种寻址方式, 可寻址 64 KB 的程序存储器空间和 64 KB 的数据存储器空间, 特别是在 MCS-51 系列单片机还集成了一个由直接可寻址位组成的布尔处理机(位处理机), 并且提供了 128 Bit 的位寻址区以及可进行位寻址的特殊功能寄存器(SFR), 结合其指令系统中的位处理指令可对各位进行布尔处理, 能够实现置位、清零、求反、传送、判跳和逻辑运算等操作。特别是对逻辑表达式直接转换成程序软件提供了极为简便的方法, 不需

要大范围地传送数据，屏蔽某些字节或位、检测和转移的树形网络就可解决复杂的逻辑运算问题。位处理指令为测控系统的应用提供了最佳代码和速度，大大增强了实时性。特别适用于测控领域和解决各种逻辑问题。本文主要针对单片机中的位寻址区和位操作指令，结合作者多年从事单片机系统开发的经验，介绍位操作指令在单片机中的应用，并且给出了一个工程实例。

### 3 MCS-51 系列单片机的可位寻址特殊功能寄存器及可位寻址区

MCS-51 系列单片机的存储器结构如图 1 所示。

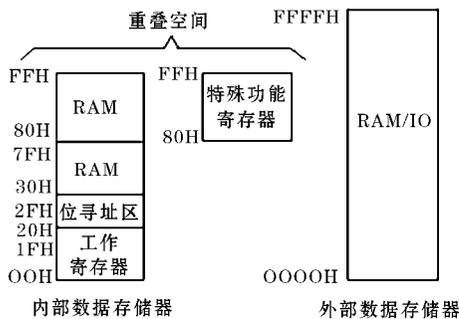


图 1 单片机内部存储器结构

Fig.1 The memory structure of mcu

由图可见，MCS-51 系列单片机的内部 RAM 有两个可直接位寻址的区域：

- ①20H-2FH，共 16 个内部 RAM 单元中的每一位都可单独寻址，共 128 位。
- ②某些特殊功能寄存器(SFR)，其特点是它们的单元地址能够被 8 整除(其地址尾数为 0 或 8)。

在 MCS-51 系列单片机中，这样的特殊功能寄存器有 11 个。8051 单片机中可位寻址的特殊功能寄存器，见表 1。

表 1 可以位寻址的特殊功能寄存器

Table 1 Bit addressable SFR

寄存器	名称	单元地址	符号	位地址
P0	I/O 端口 0	80H	P0.0-P0.7	80H-87H
TCON	定时/计数器控制寄存器	88H	TCON.0-TCON.7	88H-8FH
P1	I/O 端口 1	90H	P1.0-P1.7	90H-97H
SCON	串行控制寄存器	98H	SCON.0-SCON.7	98H-9FH
P2	I/O 端口 2	A0H	P2.0-P2.7	A0H-A7H
IE	中断允许控制寄存器	A8H	IE.0-IE.7	A8H-AFH
P3	I/O 端口 3	B0H	P3.0-P3.7	B0H-B7H
IP	中断优先级控制寄存器	B8H	IP.0-IP.7	B8H-BFH
PSW	程序状态字	D0H	PSW.0-PSW.7	D0H-D7H
ACC	累加器	E0H	ACC.0-ACC.7	E0H-E7H
B	累加器 B	FOH	B.0-B.7	FOH-F7H

对于 20H-2FH 的位寻址区，除了可作为一般的 RAM 单元进行读写之外，还可对各个字节的每一

位进行操作，并且对这些位都规定了相应的位地址，见表 2。

表 2 内部 RAM 中 20H-2FH 的位地址

Table 2 Bit address of 20H-2FH in RAM

字节地址	位地址							
D7	D6	D5	D4	D3	D2	D1	D0	D0
2F	7F	7E	7D	7C	7B	7A	79	78
2E	77	76	75	74	73	72	71	70
2D	6F	6E	6D	6C	6B	6A	69	68
2C	67	66	65	64	63	62	61	60
2B	5F	5E	5D	5C	5B	5A	59	58
2A	57	56	55	54	53	52	51	50
29	4F	4E	4D	4C	4B	4A	49	48
28	47	46	45	44	43	42	41	40
27	3F	3E	3D	3C	3B	3A	39	38
26	37	36	35	34	33	32	31	30
25	2F	2E	2D	2C	2B	2A	29	28
24	27	26	25	24	23	22	21	20
23	1F	1E	1D	1C	1B	1A	19	18
22	17	16	15	14	13	12	11	10
21	0F	0E	0D	0C	0B	0A	09	08
20	07	06	05	04	03	02	01	00

对于需要进行按位操作的数据即可存放在这个区域。单片机系统正是由于有了这些可寻址位与寄存器才具有了通用计算机系统所不具备的特殊优势，从而成为了测控领域的一支奇葩。

### 4 位地址的表示方式

MCS-51 系列单片机从硬件到软件都设有完整而独立的按位操作的位处理器，位处理器同样由 CPU、程序存储器、累加器、特殊功能寄存器和 I/O 口等组成。

片内 RAM 区和特殊功能寄存器中的位地址可用下列方式表示：

①直接用位地址(00H-FFH)来表示，如 20H 单元的 0-7 位可表示为 00H-07H。

②采用第 n 单元第 n 位(即字节地址加位数)的表示方法，二者之间用“.”分隔。如 25H 单元的第 5 位可表示为 25H.5。

③对于可进行位寻址的特殊功能寄存器，可用该寄存器名称加位数来表示，二者之间也用“.”分隔。如 P1.4, PSW.6 等。

④由用户自己用汇编语言中的伪指令定义。如：

```
CLK BIT P1.5
SETB CLK
```

上述这些表示方式均可被汇编程序所识别。

### 5 位操作指令

在 MCS-51 系列单片机的位处理器中以进位标

志位  $C_y$  作累加器用，共有 17 条位处理指令，指令的格式同字节处理指令，介绍如下：

#### 1) 布尔(位)数据传送指令

```
MOV C, bit
```

```
MOV bit, C
```

以上两条为互为对应的位数据传送指令， $C$  为位处理器的累加器， $bit$  为位直接地址。

#### 2) 布尔(位)操作指令

##### ① 布尔(位)清 0 指令

```
CLR C
```

```
CLR bit
```

这两条指令实现对累加器  $C$  或指定位地址内容的清 0 操作，不影响其他标志位。

##### ② 布尔(位)置 1 指令

```
SETB C
```

```
SETB bit
```

这两条指令实现对累加器  $C$  或指定位地址内容的置 1 操作，不影响其他标志位。

##### ③ 布尔(位)取反指令

```
CPL C
```

```
CPL bit
```

上述两条指令对累加器  $C$  或指定位地址内容取反，不影响其他标志位。

#### 3) 布尔(位)逻辑运算指令

##### ① 布尔(位)逻辑“与”指令

```
ANL C, bit
```

```
ANL C, /bit
```

前一条指令实现对累加器  $C$  内容和指定位地址内容的逻辑“与”操作，结果存在  $C$  中，位地址内容不变。后一条指令是先将位地址内容取反，之后再与累加器  $C$  的内容相“与”，结果存于  $C$  中，位地址内容不变。

##### ② 布尔(位)逻辑“或”指令

```
ORL C, bit
```

```
ORL C, /bit
```

前一条指令实现对累加器  $C$  内容和指定位地址内容的逻辑“或”操作，结果存在  $C$  中，位地址内容不变。后一条指令则是先将位地址内容取反，之后再与累加器  $C$  的内容相“或”，结果存于  $C$  中，位地址内容不变。

#### 4) 布尔(位)条件转移指令

##### ① 判位(bit)变量转移指令

```
JB bit, rel
```

```
JNB bit, rel
```

这两条互为补充的指令对指定位地址内容进行检测，若条件成立则程序转移，否则程序顺序执行。

##### ② 判位(bit)变量并清 0 转移指令

```
JBC bit, rel
```

该条指令对指定位地址内容进行检测，若为 1，则将该位清 0 后程序转向  $(PC) + rel$  目标地址去执行；否则程序顺序执行。

这条指令常用于对某一标志位需要进行多次检测的场合，并且当执行完本指令后自动复位该标志位，恢复原始状态，等待下一次的条件设置和检测，从而可以节省一条清 0 指令。

## 6 单片机中标志位的使用

在 MCS-51 系列单片机中，标志位的使用主要有两种方式：

① 使用由特殊功能寄存器提供的标志位。② 利用系统提供的可位寻址区，由用户自己设置和定义的标志位。

对于第一种方式，使用起来比较简单，用户只需要根据需要读取或设置相应的标志位即可。如在串行通讯中，用户可以通过置位串口控制寄存器  $SCON$  的  $REN$  位 ( $SCON.4$ ) 来确定是否允许单片机串口接收数据。

而对于利用位寻址区来设置标志位，则是仁者见仁，智者见智，而单片机的妙用也正在于此。一般地，位寻址区的每一位都可以当作软件触发器，由程序直接进行位处理，程序设计时通常把各种状态标志位、位控制变量设在此区域。下面以作者研制的“智能电热熔焊机”中的按键处理子程序为例进行说明。

该子程序通过定时器中断方式对“电流设置”、“时间设置”和“开始工作”三个按键进行扫描，若检测出某键被按下，则置位相应的标志位：06H、07H 或 08H。而主程序则可以通过循环检测 06H、07H 或 08H 这三个标志位的值是否为 1 来决定是否执行相应的子程序。

这里对于每一个按键均采用了“原态 AND 现态”的算法，也就是说，将每一个按键的状态分为“现态”和“原态”，分别存在两个位寻址单元中，如 00H 和 01H 中，设其值为 0 时表示该键被按下，则其状态真值见表 3。

表 3 按键状态真值表

Table 3 State value of key-press

00H (现态)	0	0	1	1
01H (原态)	0	1	0	1
00H (现态)	1	1	0	0
原态 AND 现态	0	1	0	0

由表 3 可见，只有当该键原来处于未按下状态（其值为 1），且现在处于按下状态（其值为 0）时，才表示该键被按下。根据这一思想编制的按键扫描中断服务程序如下：（下转第 444 页）