

9.3 测试应用程序

Main 函数

- “main” 是所有 C 程序的入口点
- 不要加入参数，也不要返回值 • 语法:

```
void main(void)
{
    /* 代码 */
}
```

访问外围

- 所有 I/O 寄存器在头文件里都被定义为特殊功能寄存器
- 象普通变量一样访问

```
#include <io8515.h>    /* 定义 8515 */
void main(void)
{
    DDRD = 0xFF;      /* Port D 输出 */
}
```

9.3.1 读/写口

```
#include <io8515.h>    /* 定义 AT90S8515 */
void main(void)
{
    char c;
    DDRB = 0xFF;      /* PortB 输出 */
    for(;;)          /* 死循环 */
    {
        c = PIND;    /* 读 Port D */
        PORTB = c;  /* 回写到 Port B */
    }
}
```

9.3.2 延时函数

```
#include <io8515.h>    /* 定义 8515 */
void delay(unsigned int delayValue)
{
    unsigned int i;

    for(i=0;i<delayValue;i++) /* 循环 */
        ;                    /* Do nothing */
}
```

9.3.2A: 延时函数

```

void main(void)
{
    unsigned char runner = 0x01;

    DDRB = 0xff;          /* Port B 输出 */
    for (;;)              /* 死循环 */
    {
        if (runner) runner <<= 1;
        else runner = 0x01;
        PORTB = runner;   /* 设置 LED */
        delay(100);      /* 调用延时函数 */
    }
}

```

9.3.3 读/写 E2PROM

```

/* 利用 IAR 标准 I/O 函数来读/写 E2PROM */
#include <io8515.h>
#include <ina90.h>
void main(void)
{
    char temp = 0;
    _EEDPUT(0x10,temp); /* 写 E2PROM 地址: 0x10 */

    _EEDGET(temp,0x10); /* 读 E2PROM 地址: 0x10 */
}

```

9.3.4 AVR 的 PB 口变速移位

```

/* 文件名:SLAVR934.ASM */
/* 位运算符: ~ 按位取反; << 左移; >> 右移; & 按位与; | 按位或; ^ 按位异或;
; i++相当于 i=i+1; i--相当于 i=i-1 */
#include <io8515.h> ; /* 器件配置文件 */
#define BIT(x)(1 << (x)); /* 左移 */
void delay(void)
{
    unsigned char i,j;
    for (i=1;i;i++)
        for(j=1;j;j++);
}
void led_pb(void)
{
    unsigned char i;
    DDRB=0xff; /* 设 PB 口输出 */
    for (i=0;i<8;i++) /* 硬件设定低电平灯亮,LED 的 1 位亮灯从 B 口 PB0→PB7 */
}

```

```

    {
        PORTB=~BIT(i);    /* LED 亮灯 1 位 */
        delay( );        /* 延时 */
    }
}
void main (void)        /* 主函数 */
{
    while (1)          /* 循环 */
        led_pb( );
}

```

9.3.5 4 个口 LED 亮灯变速移位

```

/* 文件名:SLAVR935.ASM */
/* 请修改程序,改变移位方向,2 位或 3 位或一隔一亮灯移位等 */
#include <io8515.h>    /* 预处理命令,头文件 */
#define BIT(x) (1 << (x))    /* 定义位函数,可修改移位方向 */
void delay(unsigned char t); /* 延时函数 */
{
    unsigned char i;
    unsigned char j;
    for (i=0;i<t;i++)
        for(j=1;j;j++);
}
void led_pb(unsigned char t); /* LED 移位函数 */
{
    unsigned char i;
    DDRB=0xff;    /* 设 PB 口为输出 */
    for (i=0;i<8;i++) /* 硬件设定低电平灯亮,LED 的 1 位亮灯从 B 口 PB0→PB7 */
    {
        PORTB=~BIT(i); /* LED 亮灯 1 位 */
        delay(t);    /* 延时 */
    }
    PORTB=0xff;    /* 关 PB 口 */
}
void led_pd(unsigned char t); /* LED 的 1 位亮灯从 D 口 PD0→PD7 移位函数 */
{
    unsigned char i;
    DDRD=0xff;
    for (i=0;i<8;i++) /* LED 的 1 位亮灯从 D 口 PD0→PD7 */
    {
        PORTD=~BIT(i);
        delay(t);
    }
    PORTD=0xff;
}

```

```

}
void led_pc(unsigned char t); /* C 口 PC0→PC7 移位函数 */
{
    unsigned char i;
    DDRC=0xff;
    for (i=0;i<8;i++) /* LED 的 1 位亮灯从 C 口 PC0→PC7 */
    {
        PORTC=~BIT(i);
        delay(t);
    }
    PORTC=0xff;
}
void led_pa(unsigned char t); /* A 口 PA7→PA0 移位函数 */
{
    unsigned char i;
    DDRA=0xff;
    for (i=8;i>0;i--) /* LED 的 1 位亮灯从 A 口 PA7→PA0 */
    {
        PORTA=~BIT(i-1);
        delay(t);
    }
    PORTA=0xff;
}
void main (void) ; /* 主函数 */
{
    unsigned char dt;
    while (1) /* 循环 */
    {
        for (dt=5;dt<200;dt+=25)
        {
            led_pb(dt); /* LED 发光二极管一亮灯沿四个口移位变速循环 */
            led_pd(dt);
            led_pc(dt);
            led_pa(dt);
        }
    }
}

```

9.3.6 音符声程序

```

/* 源程序 SLAVR936.ASM*/
/* 可改变 t 函数,改变发音快慢 */
#include <io8515.h> /* 预处理命令 */
#define uchar unsigned char
#define uint unsigned int

```

```

void delay(uchar t)
{
    uchar i,j;
    for (i=0;i<t;i++)
        for(j=1;j<150;j++);
}
void sound_pc0(uchar t)
{
    uint i;
    DDRC=0xff;
    PORTC=0xff;
    for (i=0;i<350-t*t;i++) /* 改变发音快慢 ,另见 SLAVR936B.ASM 程序*/
    {
        PORTC^=(1<<0);
        delay(t);
    }
}
void main (void) /* 主函数 */
{
    uchar dt;
    for( ; ; )
        for(dt=1;dt<14;dt++) /* 改变发音数量 */
            sound_pc0(dt);
}

```

9.3.7 8 字循环移位显示程序

```

/* 源程序 SLAVR937.ASM, 在 SL-AVR 开发实验器验证通过*/
/*在 SL-AVR 开发实验器 LED 数码管上,8 字符循环移位显示程序*/
#include <io8515.h> /*器件配置文件*/
#define uchar unsigned char /*定义缩写*/
#define uint unsigned int
void delay(uint t)
{
    uint i;
    for (i=0;i<t;i++);
}
void init_disp(void) /*B 口,D 口初始化*/
{
    DDRB=0xff;
    DDRD=0xff;
    PORTB=0x7f; /*B 口送 8 字符,字形可修改*/
}
void scan(void) /*位选扫描*/

```

```

{
  uchar i,j;
  for (i=0;i<6;i++) /*i++可修改为一位隔一位或隔 2 位显示或改变移位方向*/
  {
    j=150; /*可改变移位速度*/
    do
    {
      PORTD=~(0x01<<i);
      delay(150); /*可改变 LED 显示亮度*/
      PORTD=0xff;
      delay(2100); /*可改变 LED 显示亮度*/
    }
    while(--j);
  }
}
void main(void) /*主程序*/
{
  init_disp(); /*初始化*/
  for( ; ; )
    scan( ); /*位选扫描*/
}

```

9.3.8 按键加 1 计数显示程序

/* 源程序 SLAVR938.ASM, 在 SL-AVR 开发实验器验证通过*/
 /*在 SL-AVR 开发实验器上,用 SHIFT 键,按 1 次键加 1 计数显示程序*/

```

#include <io8515.h> /*头文件*/
#define uchar unsigned char /*缩写定义*/
#define uint unsigned int
flash uchar DATA_7SEG[ ]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,
                           0x7f,0x6f,0x77,0x7c,0x39,0x5e,0x79,0x71};/*LED 字形表*/
uchar led[6]; /*显示缓冲*/
uint count; /*延时子程序*/
void delay(uint t)
{
  uint i;
  for (i=0;i<t;i++);
}
void init_disp(void) /*初始化 B 口,D 口*/
{
  DDRB=0xff;
  DDRD=0x7f;
  PORTD|=0x80;
}

```

```
}
void disp(void)      /*键盘显示*/
{
    uchar i;
    for (i=0;i<6;i++)
    {
        PORTD=~(0x01<<i);
        PORTB=DATA_7SEG[led[i]];
        delay(1000);
    }
    PORTB=0x00;
    PORTD=0xff;
}
void be_pc0(void)    /*发出一声响子程序*/
{
    uint i;
    DDRC|=0x01;
    for (i=0;i<350;i++)
    {
        PORTC^=0x01;
        delay(350);
    }
}
void conv(void)      /*计数值转换成十进制数*/
{
    led[5]=0;
    led[4]=count/10000;
    led[3]=count/1000%10;
    led[2]=count/100%10;
    led[1]=count/10%10;
    led[0]=count%10;
}
void main(void) /*主程序*/
{
    init_disp( ); /*初始化 B 口,D 口*/
    count=0;      /*开始计数值是零*/
    conv();       /*转换*/
    for( ; ; )
    {
        while((PIND&0x80)==0x80) /*没有键按下等待*/
            disp();             /*显示*/
        be_pc0();              /*发出一声响*/
        count++;               /*计数器加 1*/
        conv( );               /*转换成十进制数*/
    }
}
```

```
    while((PIND&0x80)==0) /*有键按下*/  
        disp();          /*显示*/  
    }  
}
```