



职业院校教学用书（电子类专业）

电工电子类专业 **技能大赛** 实战丛书

# 单片机控制装置安装与调试

下册

雷林均 主编

MCU control device installation and commissioning



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

职业院校教学用书（电子类专业）  
电工电子类专业技能大赛实战丛书

# 单片机控制装置安装与调试

## 下册

雷林均 主编

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书通过大量的实例,由简到繁、由易到难地实践单片机控制装置电路、编程、调试等技术;内容丰富,讲解深入,紧扣全国中职学生单片机技能大赛而不同于大赛。

上册主要包括:学习单片机所需的硬件和软件、AT89S51/52 单片机的特性、并行 I/O 口的结构及编程,通过发光二极管、开关、按钮、矩阵键盘、百变流水灯等基础应用讲解 C 语言编程知识,详细介绍数码管、点阵、液晶等显示技术与编程技巧,还有大量关于定时/计数器、中断系统的应用示例。

下册主要包括:单片机的特殊功能,如节电控制寄存器、看门狗,扩展单片机的 I/O 口、存储器、AD/DA、定时器的方法,温度、压力等传感器技术及数字滤波编程方法,智能控制装置的应用,大量串行通信实例,基于 RTX-51 操作系统的多任务编程,物料搬运、微波炉、电梯等典型控制系统及近两年全国单片机大赛试题解析。

本书配套光盘提供全部实验的源程序和 188 个 Proteus ISIS 仿真电路,特别是提供了智能物料搬运装置、电梯等复杂系统的动画仿真,让没有实验条件的读者也能做单片机实验。

本书既是一本大赛训练宝典,又是一本单片机爱好者入门和提高的好教程;既适合单片机初学者用做教材,也适合有一定基础的单片机爱好者用做参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

### 图书在版编目(CIP)数据

单片机控制装置安装与调试.下册/雷林均主编.一北京:电子工业出版社,2011.2

(电工电子类专业技能大赛实训丛书)

职业院校教学用书·电子类专业

ISBN 978-7-121-12884-4

I. ①单… II. ①雷… III. ①单片微型计算机—计算机控制系统—安装—专业学校—教学参考资料  
②单片微型计算机—计算机控制系统—调试—专业学校—教学参考资料 IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2011)第 016577 号

策划编辑:杨宏利 yhl@phei.com.cn

责任编辑:徐 萍

印 刷:北京市李史山胶印厂

装 订:

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编:100036

开 本:787×1092 1/16 印张:12.75 字数:327 千字

印 次:2011 年 2 月第 1 次印刷

印 数:4 000 册 定价:28.00 元(含光盘 1 张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

# 前 言

单片机技术的应用越来越广泛，电子技术的应用领域中，几乎可以说“想得到就用得上”。为了推广单片机技术，推动电工电子类专业课程改革，全国技能大赛电工电子类设置了“单片机控制装置安装与调试”竞赛项目。为了满足广大师生对单片机教学及参加技能大赛训练的需要，特编撰此书奉献给读者。

本书以行动导向的思想为指导，通过大量实例，围绕单片机控制装置，由浅入深、由简到繁地安排训练项目。将单片机硬件知识和 C 语言编程知识分解到实例中，通过“做中学”来教授单片机硬件知识和编程知识，特别适合“做、学、教”一体化教学方案。书中既有基础知识和基本技能的训练，又有满足大赛需要的高级技术、技巧。

单片机编程有汇编语言和 C 语言两种。采用 C 语言编程具有代码短、可读性强、可移植性强、开发周期短的优点，特别是在较大规模的单片机运用场合及比较复杂的算法实现上，C 语言明显优于汇编语言。因此本书采用 C 语言进行编程。

笔者在编写过程中，针对每一个模块，总是尽量采用多种程序算法和列举较多的应用实例，以拓展读者思维。例如，对一个简单的“开关控制 LED”实验，介绍 if 语句法、问号语句法、赋值法、函数法等；矩阵键盘取键值的函数，介绍扫描法、反转法、单键判断法等；流水灯效果，介绍十余种程序算法；各种显示器，不但介绍基本的显示技术，还介绍许多显示特效的程序算法；对定时器的应用，给出计数器、频率计、脉宽测量、交通灯、音乐演奏、电子钟、闹钟、农历日历钟等大量实例。

书中实例涵盖全国单片机技能大赛指定设备的所有模块，但不限于大赛模块，还介绍了并行存储器、I<sup>2</sup>C 存储器、I<sup>2</sup>C 实时时钟芯片、压力传感器、PS2 键盘、非接触式 IC 卡、USB 接口等大量其他常用模块的应用。

本书还特别介绍了基于 RTX-51 Tiny 多任务操作系统的编程，使读者面对复杂应用系统的开发也能应对自如。

作为综合应用系统设计示例，书中介绍了物料搬运、微波炉、电梯等典型控制系统，还对全国单片机大赛试题进行了详细解析。

即使您是一个单片机“外行”，相信通过本书的学习和实战训练，也一定能快速成长，成为能独立运用单片机技术解决实际问题的技术人才。

本书配套光盘中提供书中所有示例的源程序，同时提供 Proteus ISIS 仿真电路，方便读者进行实验。

在本书的编写过程中，王莉负责大量文稿处理和校对工作，雷磊负责源程序验证实验工作，在此表示感谢！

由于作者水平有限，书中难免存在一些错误和不足之处，希望广大读者批评指正。

编 者  
2011 年 1 月

# 目 录

|                                    |    |
|------------------------------------|----|
| <b>第 6 章 AT89S52 特殊功能</b>          | 1  |
| 6.1 节电控制                           | 2  |
| 6.1.1 节电控制寄存器 PCON                 | 2  |
| 6.1.2 节电模式实验                       | 3  |
| 6.1.3 掉电检测与后备电源                    | 4  |
| 6.2 看门狗 WDT                        | 5  |
| 6.2.1 看门狗的作用                       | 5  |
| 6.2.2 硬件看门狗                        | 5  |
| 6.2.3 软件抗干扰                        | 6  |
| 6.2.4 内置看门狗                        | 6  |
| 6.2.5 掉电和空闲方式下的 WDT                | 7  |
| 6.2.6 辅助寄存器 AUXR                   | 7  |
| <b>第 7 章 扩展单片机的功能</b>              | 9  |
| 7.1 扩展 I/O 口                       | 10 |
| 7.1.1 串—并转换扩展输出口                   | 10 |
| 7.1.2 并—串转换扩展输入口                   | 11 |
| 7.1.3 扩展 74LS245                   | 13 |
| 7.1.4 74LS373 及其应用                 | 15 |
| 7.1.5 8255 及其应用                    | 18 |
| 7.2 扩展存储器                          | 23 |
| 7.2.1 扩展程序存储器                      | 23 |
| 7.2.2 扩展数据存储器                      | 25 |
| 7.2.3 I <sup>2</sup> C EEPROM 及其应用 | 28 |
| 7.3 扩展 A/D 与 D/A                   | 39 |
| 7.3.1 ADC0809 及其应用                 | 39 |
| 7.3.2 DAC0832 及其应用                 | 44 |
| 7.4 定时器 8253 及其应用                  | 47 |
| 7.4.1 8253 简介                      | 47 |
| 7.4.2 8253 应用示例                    | 49 |
| <b>第 8 章 检测技术</b>                  | 50 |
| 8.1 传感器                            | 51 |

|                         |                       |           |
|-------------------------|-----------------------|-----------|
| 8.1.1                   | 传感器的种类 .....          | 51        |
| 8.1.2                   | 传感器与单片机 .....         | 52        |
| 8.1.3                   | 传感器的应用 .....          | 53        |
| 8.2                     | 温度传感器 LM35 .....      | 55        |
| 8.2.1                   | LM35 介绍 .....         | 55        |
| 8.2.2                   | 模块电路分析 .....          | 56        |
| 8.2.3                   | 模块连接 .....            | 56        |
| 8.2.4                   | 温度转换编程 .....          | 56        |
| 8.3                     | 数字滤波技术 .....          | 57        |
| 8.3.1                   | 算术平均值法滤波 .....        | 57        |
| 8.3.2                   | 权重计算法滤波 .....         | 58        |
| 8.3.3                   | 滑动平均值法滤波 .....        | 58        |
| 8.3.4                   | 去极值法滤波 .....          | 58        |
| 8.3.5                   | 限幅法滤波 .....           | 59        |
| 8.3.6                   | 中值法滤波 .....           | 59        |
| 8.4                     | 压力传感器 MPX4105 .....   | 60        |
| 8.4.1                   | MPX4105 介绍 .....      | 60        |
| 8.4.2                   | 应用举例 .....            | 60        |
| 8.4.3                   | 程序示例 .....            | 61        |
| 8.5                     | 数字温度传感器 DS18B20 ..... | 61        |
| 8.5.1                   | DS18B20 介绍 .....      | 61        |
| 8.5.2                   | 基本程序 .....            | 64        |
| 8.5.3                   | 防止中断干扰 .....          | 66        |
| <b>第 9 章 智能控制 .....</b> |                       | <b>67</b> |
| 9.1                     | 交、直流电动机模块 .....       | 68        |
| 9.1.1                   | 交、直流电动机模块分析 .....     | 68        |
| 9.1.2                   | 转数控制 .....            | 69        |
| 9.1.3                   | 转速测量 .....            | 70        |
| 9.2                     | 步进电动机模块 .....         | 72        |
| 9.2.1                   | 步进电动机简介 .....         | 72        |
| 9.2.2                   | 驱动编程 .....            | 72        |
| 9.2.3                   | 步进电动机模块 .....         | 73        |
| 9.2.4                   | 初始定位编程 .....          | 76        |
| 9.2.5                   | 定位控制编程 .....          | 77        |
| 9.2.6                   | 速度控制 .....            | 79        |
| 9.3                     | 智能物料搬运装置 .....        | 80        |
| 9.3.1                   | 智能物料搬运装置详解 .....      | 80        |
| 9.3.2                   | 物料搬运装置的调节 .....       | 82        |

|                           |    |
|---------------------------|----|
| 9.3.3 物料搬运装置与单片机的连接 ..... | 82 |
| 9.3.4 基本控制程序 .....        | 83 |

## 第 10 章 串行通信 .....

|                              |     |
|------------------------------|-----|
| 10.1 基本概念 .....              | 87  |
| 10.1.1 通信协议 .....            | 87  |
| 10.1.2 波特率 .....             | 87  |
| 10.2 51 单片机串行口 .....         | 87  |
| 10.2.1 读/写串行口数据 .....        | 87  |
| 10.2.2 串行控制与状态寄存器 .....      | 88  |
| 10.2.3 串行口的工作方式 .....        | 89  |
| 10.2.4 波特率加倍控制 .....         | 90  |
| 10.3 双机通信 .....              | 90  |
| 10.3.1 双机通信电路 .....          | 90  |
| 10.3.2 多路温度采集与串行传输 .....     | 91  |
| 10.4 单片机与 PC 通信 .....        | 96  |
| 10.4.1 单片机与 PC 接口 .....      | 96  |
| 10.4.2 上位机程序 .....           | 96  |
| 10.4.3 下位机程序 .....           | 97  |
| 10.5 多机通信 .....              | 99  |
| 10.5.1 原理 .....              | 99  |
| 10.5.2 实例 .....              | 100 |
| 10.6 特殊串行通信实例 .....          | 103 |
| 10.6.1 日历时钟芯片 PCF8563 .....  | 103 |
| 10.6.2 PS2 键盘接口 .....        | 106 |
| 10.6.3 非接触式 IC 卡及读/写模块 ..... | 110 |
| 10.6.4 USB 接口 CH372 .....    | 119 |

## 第 11 章 RTX-51 操作系统 .....

|                              |     |
|------------------------------|-----|
| 11.1 RTX-51 概述 .....         | 125 |
| 11.2 简单的多任务系统 .....          | 126 |
| 11.2.1 两个 LED 不等速闪烁 .....    | 126 |
| 11.2.2 开关控制 LED 闪烁 .....     | 127 |
| 11.3 RTX-51 Tiny 原理 .....    | 129 |
| 11.4 配置文件 CONF_TNY.A51 ..... | 131 |
| 11.5 使用 RTX-51 Tiny .....    | 132 |
| 11.6 RTX-51 Tiny 函数参考 .....  | 133 |
| 11.7 超级系统示例 .....            | 137 |
| 11.7.1 任务书 .....             | 137 |

|                  |     |
|------------------|-----|
| 11.7.2 任务分析····· | 137 |
| 11.7.3 程序解析····· | 138 |

## 第 12 章 综合练习····· 146

|                              |     |
|------------------------------|-----|
| 12.1 简易物料传送机·····            | 147 |
| 12.1.1 任务书·····              | 147 |
| 12.1.2 分析·····               | 148 |
| 12.1.3 程序示例·····             | 148 |
| 12.1.4 调试·····               | 149 |
| 12.2 微波炉控制器·····             | 150 |
| 12.2.1 任务书·····              | 150 |
| 12.2.2 电路及分析·····            | 151 |
| 12.2.3 程序示例·····             | 152 |
| 12.2.4 调试·····               | 154 |
| 12.3 电梯仿真实验·····             | 154 |
| 12.3.1 任务书·····              | 154 |
| 12.3.2 设计示例·····             | 155 |
| 12.3.3 程序示例·····             | 156 |
| 12.4 物料搬运加工·····             | 161 |
| 12.4.1 任务书·····              | 161 |
| 12.4.2 分析·····               | 162 |
| 12.4.3 程序示例·····             | 163 |
| 12.5 2009 年全国中职组单片机项目赛题····· | 167 |
| 12.5.1 任务书·····              | 167 |
| 12.5.2 分析·····               | 169 |
| 12.5.3 程序详解·····             | 170 |
| 12.6 2010 年全国中职组单片机项目赛题····· | 173 |
| 12.6.1 任务书·····              | 173 |
| 12.6.2 分析·····               | 178 |
| 12.6.3 程序详解·····             | 180 |

## 附录 A C51 资料及竞赛规则····· 190

## 参考文献····· 194

# 第 6 章

## AT89S52 特殊功能



# 6.1 节电控制

## 6.1.1 节电控制寄存器 PCON

AT89S52 单片机可以通过置位 PCON 的 IDL 及 PD, 来进入空闲工作模式或掉电保持模式, 从而减少单片机的能耗。

表 6.1.1 所示为 PCON 的位定义

表 6.1.1 PCON 的位定义

| 位 号 | 7    | 6 | 5 | 4   | 3   | 2   | 1  | 0   |
|-----|------|---|---|-----|-----|-----|----|-----|
| 位 名 | SMOD | — | — | POF | GF1 | GF0 | PD | IDL |

- IDL: 空闲模式启动位。
- PD: 掉电模式启动位。
- GF0, GF1: 通过标志位。
- POF: 掉电标志。
- SMOD: 波特率加倍控制。使用定时器 1 作为串行通信波特率发生器时, SMOD 置 1, 波特率加倍。

PCON 只能以字节操作, 不能直接进行位操作。例如, 置位 IDL, 可以是  $PCON|=0x01$ , 不能为  $IDL=1$ 。

### 1. 空闲节电模式

IDL 位置 1 时, 立即进入空闲工作模式。空闲模式下, 振荡器继续工作, 中断系统、定时器、串行口有效 (所有片内的外设仍保持激活状态); 但时钟停止提供给 CPU, CPU 保持睡眠状态; 片内 RAM 和所有特殊功能寄存器的内容保持不变。

在空闲模式下, 任何允许的中断请求, 将使 IDL 清零, 退出空闲模式, 同时进入中断服务程序。中断函数返回后, 执行的第一条指令, 是前面进入空闲模式时置位 IDL 指令的下一条指令。

使用空闲模式时, 通常把大多数的任务都放入中断函数; 主程序只进行必要的初始化, 然后就进入空闲模式; CPU 只在中断函数执行时工作, 大大降低了单片机的功耗。

### 2. 掉电模式

PD 位置 1 时, 立即进入掉电模式。掉电模式下, 振荡器停止工作, 定时器、串行口停止工作; 片内 RAM 和特殊功能寄存器的内容保持不变。

掉电模式下, 原来的 AT89C51 只能通过硬件复位来结束掉电模式状态。AT89S52 进入掉电模式后, 还可以由处于使能状态的外中断  $\overline{INT0}$  和  $\overline{INT1}$  结束。

如果外中断允许, 则外中断  $\overline{\text{INT0}}$  或  $\overline{\text{INT1}}$  的中断请求将清零 PD 位, 退出掉电模式。然后执行中断函数, 中断函数返回时, 接着执行主程序中置位 PD 指令的下一条指令。

使用外中断来唤醒单片机时, 不管是跳变触发还是低电平触发, 均要求  $\overline{\text{INT0}}$  或  $\overline{\text{INT1}}$  低电平能保持足够的时间, 让单片机时钟振荡电路起振并稳定工作。

掉电模式可以使单片机处于最低功耗状态。

空闲模式和掉电模式都可以通过硬件复位结束。复位将重新定义除 POF 位外的全部特殊功能寄存器, 但不改变 RAM 中的内容。复位必须是在  $V_{\text{CC}}$  恢复到正常工作电平后进行, 且必须保持一定时间。

### 3. 掉电标志 POF

当 AT89S52 首次上电时, POF 被硬件置 1。POF 位可以由软件置 1 或清零, 且不受复位影响。利用此标志可以区别初始上电的复位与工作过程中的复位, 以便程序做不同的处理。

## 6.1.2 节电模式实验

### 1. 任务书

通过如图 6.1.1 所示电路, 利用  $\overline{\text{INT0}}$  上的按钮切换, 表现 AT89S52 不同的节电工作模式。

### 2. 分析

正常工作模式、空闲模式、掉电模式分别通过点亮 D1、D2、D3 来指示。空闲模式与掉电模式都能被外部中断唤醒, 所以可以实现按钮 K 对模式的切换。

为了测试 POF 位, 在程序段中让 D1 通过不同频率的闪烁来表现。

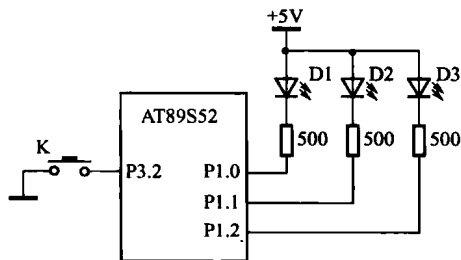


图 6.1.1 节电模式实验

### 3. 程序示例

```
01 #include <REG52.H>
02 sbit D1=P1^0; sbit D2=P1^1; sbit D3=P1^2;
03 sbit P32=P3^2;
04 unsigned char MOD_count;
05 delay(unsigned int i){while(--i);}
06 main() {
07     char i;
08     EA=EX0=1; //开启INT0中断, 默认电平触发
09     while(1){
10         if(PCON&0x10){ //初始上电, POF为1
11             for(i=0; i<4; i++){D1=!D1; delay(60000);} //首次上电
12             PCON&=0xef; //清零标志, POF
13         }else{
14             for(i=0; i<10; i++){D1=!D1; delay(10000);} //非掉电复位
15         }
16         PCON|=0x01; //置位IDL, 进入空闲模式
17         for(i=0; i<10; i++){D2=!D2; delay(30000);} //D2闪烁
18         PCON|=0x02; //置位PD, 进入掉电模式
```

```

19     for(i=0;i<10;i++) {D3=!D3;delay(30000);} //D3闪烁
20   }
21 }
22 /** 外中断0, 什么事也不做, 只为了将CPU从节电模式中唤醒 **/
23 int0_interrupt() interrupt 0
24 { ; }

```

08 行：开启外部中断 0。第 24 行外部中断函数为“空”，仅用于唤醒 CPU。

10 行：根据掉电标志 POF 来选择 11 行或 14 行。

17 行：从空闲模式唤醒，D2 闪烁 5 次。

19 行：从掉电模式唤醒，D3 闪烁 5 次。

#### 4. 测试

(1) 上电效果观察。首次上电，D1 慢闪 2 次，按复位键，D1 快闪 5 次。

(2) 接着 LED 关闭，说明 CPU 进入空闲模式。

(3) 按一下 K 按钮，D2 闪 5 次，说明空闲模式被  $\overline{\text{INT0}}$  中断结束。

(4) 接着 LED 又关闭，说明 CPU 又进入掉电模式。

(5) 再按一下 K 按钮，D3 闪 5 次，掉电模式被  $\overline{\text{INT0}}$  中断结束。

(6) 接着 D1 快闪 5 次，说明主循环返回。

(7) 无论空闲模式还是掉电模式，按复位键，D1 闪烁 5 次，说明节电模式可以通过复位结束。复位将使程序重新开始。

### 6.1.3 掉电检测与后备电源

工作中突然停电是不可避免的，为了防止突然停电带来的数据丢失，必须对掉电进行检测。一旦掉电，立即启动数据保护程序，在电源降到下限值之前，将数据移至可靠的位置存储。

后备电源的作用，是在市电掉电时维持单片机的全部或部分工作，保护数据如图 6.1.2 所示。

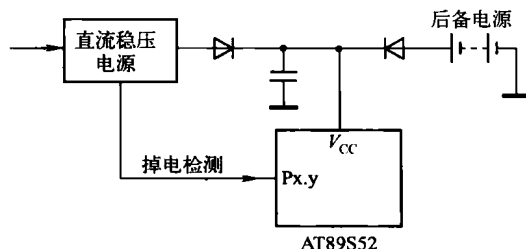


图 6.1.2 后备电源

当市电正常时，通过电源给单片机 I/O 引脚一个信号，可以是市电经隔离、整流、限幅、整形等处理后的脉冲，也可以是某种电平信号。一旦市电掉电，这个信号及时反映到单片机。单片机通过定时查询或中断等方式获取掉电信号，并立即进行掉电处理，比如关闭电源消耗大的外部设备、将单片机切换成空闲工作模式或掉电模式。

# 6.2 看门狗 WDT

## 6.2.1 看门狗的作用

在微型计算机系统中，由于外界电磁场的干扰，单片机运行时可能出现“死机”。这是由于干扰造成程序的跑飞，陷入了死循环，使单片机控制系统无法正常工作，这将发生不可预料的后果。

为了防止程序跑飞出现死机现象，通常利用“看门狗”来监控单片机的工作。看门狗实际上是一个定时装置，看门狗启动后开始定时，一旦定时到达某个阈值，就会输出复位脉冲。看门狗的定时器可以通过程序清零，编程时需要在程序主循环适当位置添加“清零”看门狗定时器的指令。单片机程序正常运行时，看门狗定时器被不断清零达不到阈值，看门狗电路不会输出复位脉冲。一旦程序跑飞、进入某个死循环后，用于清零看门狗的指令不能被执行，看门狗定时器很快达到预设阈值而输出复位脉冲，使单片机复位。单片机因复位而重新开始工作，退出死机状态。

看门狗具有监视单片机工作的作用。把清零看门狗定时值的指令称为“喂狗”。单片机“喂狗”的时间间隔必须比看门狗定时器的阈值时间小，否则看门狗就会复位单片机。

看门狗定时，可以利用单稳态电路来设计，也可以使用专用看门狗芯片，还可以利用单片机定时器设计软件设计看门狗。很多单片机已经内置了看门狗电路。

## 6.2.2 硬件看门狗

对于不带内置看门狗的单片机，可以使用外置硬件看门狗。实现看门狗功能的电路很多，如时基集成电路 555、单稳态触发器 74LS123、带振荡电路的计数器 CC4060 等，都可以设计出看门狗电路。

下面以 555 时基为例，简要说明其设计原理。如图 6.2.1 所示，正常工作时，单片机送来的喂狗脉冲 WDT\_IN 经 C1 使 VT 导通，不断给 C2 充电，555 保持低电平输出；一旦单片机因进入死循环停止喂狗，则 VT 截止，555 开始振荡，RST\_OUT 输出高电平使单片机复位。在 555 输出低电平期间，单片机重新开始工作，喂狗脉冲开始产生，555 被锁定在低电平输出。

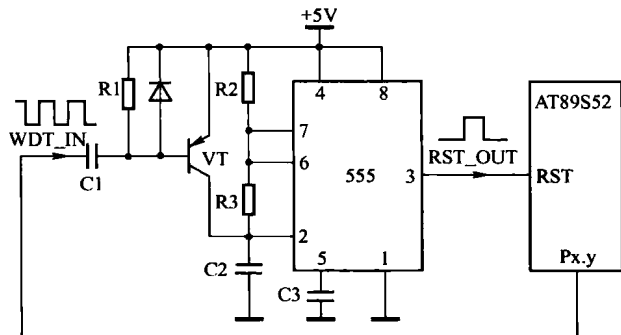


图 6.2.1 时基 555 构成的看门狗复位电路

### 6.2.3 软件抗干扰

#### 1. 端口刷新

特殊功能寄存器，特别是 I/O 寄存器的状态很可能被干扰破坏。程序中通过反复刷新 I/O 寄存器，能使其受干扰破坏的状态及时得到恢复。

#### 2. 程序陷阱

陷阱，就是在程序中间插入一些跳转指令，希望程序跑飞后能落入“陷阱”，实现跳转，进行出错处理或者恢复工作。

##### (1) 指令冗余

51 单片机的指令有单字节、双字节、三字节几种，如果干扰使程序跑飞后，落到 51 单片机的多字节指令中间，程序代码的意义就面目全非了。因此在程序中间嵌入一些“空操作”指令，希望程序跑飞后能遇上空操作指令，使程序回归正轨。

##### (2) 空操作+跳转

这是指在程序中间嵌入“一长串空操作，后面加一条长跳转汇编指令”。正常工作时，这些指令不被执行；程序跑飞后，一旦进入这个区域，则强制程序跳到出错处理及初始化部分。

#### 3. 中断控制

中断控制是指利用定时中断，监视主程序的运行。主程序在定时器没有溢出时，反复清零计数器；如果主程序跑飞，定时器有可能溢出，则通过执行中断程序强制单片机初始化复位。

软件陷阱只能在一定程度上解决程序跑飞问题。与硬件看门狗比起来，软件陷阱程序复杂，降低了程序效率，且效果不佳，特别是在 C 语言编程中实现起来更是不容易。

### 6.2.4 内置看门狗

AT89S51/52 内置有看门狗定时器 (WDT)。它由一个 14 位定时器和看门狗复位 (WDTRST) 电路构成。单片机复位时，WDT 默认为关闭状态。要打开 WDT，只要向看门狗定时器 (地址为 0xA6) 顺序写入 0x1E、0xE1 即可。程序如下：

```
sfr WDTRST=0xA6; //定义
...
WDTRST=0x1E;
WDTRST=0xE1;
```

执行以上两行代码后，WDT 即启动。看门狗定时器会在每个机器周期加 1。除硬件复位或 WDT 溢出复位外，没有办法关闭 WDT。看门狗定时器不能读取，也不能赋值。当 WDT 定时器溢出时，将使 RST 引脚输出持续时间为 98 个晶振周期的高电平复位脉冲，使单片机复位。

程序正常执行时，为了防止 WDT 定时器溢出复位单片机，需在程序适当位置放“喂狗”指令。喂狗也是向 WDTRST 顺序写入 0x1E、0xE1。喂狗的目的是清零看门狗定时器，以避免

WDT 计数溢出。因 WDT 定时器是 14 位的，所以在每 16 383 个机器周期内必须喂狗一次。

看门狗测试程序如下：

```

01 #include <REGX52.H>
02 sfr WDTRST=0xA6; //看门狗寄存器定义
03 delay(unsigned int i){while(--i);}
04 sbit LED=P2^0; //发光管
05 main(){
06     unsigned char i;
07     for(i=0;i<10;i++){
08         LED=!LED;delay(20000);
09     }
10     while(1){
11         delay(2045); //延时参数≤2 045，主循环正常
12         WDTRST=0x1E; //喂狗
13         WDTRST=0xE1;
14     }
15 }

```

07~09 行：复位时，LED 闪 5 次。

10~14 行：主循环。进入主循环后，LED 停止闪烁。

11 行：延时，代替程序正常工作指令。

12, 13 行：喂狗。每次主循环喂狗一次。

测试：将第 11 行延时函数参数值改为 2 046 及以上，发光管会不停闪烁，说明看门狗复位起了作用，单片机被反复重新启动，第 08 行被执行；如果将延时参数改为 2 045 及以下，则 LED 闪烁 5 次后停止，说明程序正常在 10~14 行循环。

目前 ISIS 仿真不能仿真 AT89S52 的看门狗，请在实验台上实验。

看门狗不是万能的，必须在硬件上进行抗干扰设计，“软硬兼施”提高抗干扰能力。

### 6.2.5 掉电和空闲方式下的 WDT

在掉电模式下，晶体振荡停止，WDT 也停止。掉电模式下，用户不能复位 WDT。有两种方法可以退出掉电模式：通过硬件复位或激活外部中断。当通过硬件复位退出掉电模式时，处理 WDT 可像通常的上电复位一样。而由中断退出掉电模式则有所不同，中断低电平状态持续到晶体振荡稳定，当中断电平变为高即响应中断服务。为防止中断误复位，当器件复位、中断引脚持续为低时，WDT 并未开始计数，直到中断引脚被拉高为止。这为在掉电模式下的中断执行中断服务程序而设置。

为保证 WDT 在退出掉电模式时极端情况下不溢出，最好在进入掉电模式前复位 WDT。

在进入空闲模式前，WDT 打开时，WDT 是否继续计数由特殊功能寄存器 AUXR 的 WDIDLE 位决定，在空闲 (IDLE) 期间 (位 WDIDLE=0) 默认状态是继续计数。为防止单片机在空闲模式中被复位，应通过定时器中断周期性喂狗，然后重新进入空闲模式。

当位 WDIDLE 被置位时，在空闲模式中 WDT 将停止计数，直到从空闲 (IDLE) 模式中退出才重新计数。

### 6.2.6 辅助寄存器 AUXR

表 6.2.1 所示为辅助寄存器 AUXR 的位定义。

表 6.2.1 辅助寄存器 AUXR 的位定义

| 位 号 | 7 | 6 | 5 | 4      | 3      | 2 | 1 | 0      |
|-----|---|---|---|--------|--------|---|---|--------|
| 位 名 |   |   |   | WDIDLE | DISTRO |   |   | DISALE |

**DISALE:** ALE 使能标志位。DISALE=0 时, ALE 以 1/6 晶振频率输出信号; DISALE=1 时, ALE 只在执行 MOVX 或 MOVC 指令(访问外部数据或程序)时激活。

**DISTRO:** 看门狗复位禁止。DISTRO=0 时, 允许看门狗在其定时结束时输出高电平复位信号; DISTRO 置 1 时, 看门狗不输出复位信号。

**WDIDLE:** 空闲模式下看门狗使能标志。空闲模式下, WDIDLE=0 时, WDT 继续计数; WDIDLE=1 时, WDT 停止计数。

AUXR 的地址为 0x8E, 不能直接进行位操作, 只能以字节操作。示例如下:

```
sfr AUXR=0x8E;           //定义 AUXR
#define WDIDLE 0x10       //定义 WDIDLE 位
AUXR |= WDIDLE;          //置 1 WDIDLE 位
AUXR &= (~WDIDLE);       //清零 WDIDLE 位
```

# 第7章

## 扩展单片机的功能