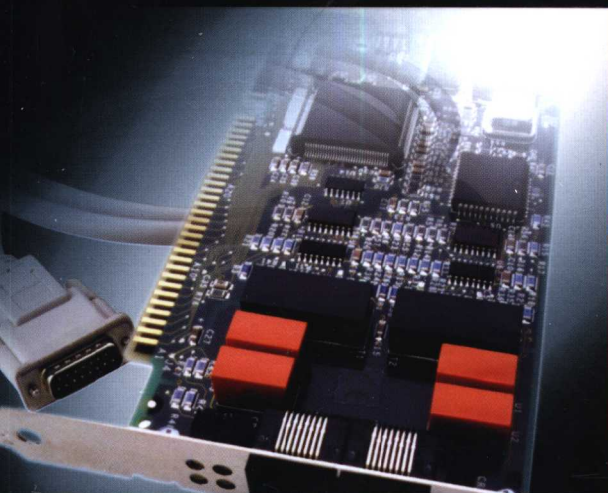


嵌入式系统与单片机系列丛书

# 基于单片机的 智能系统设计与实现

沈红卫 著



“十五”国家重点图书  
出版规划项目



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

“十五”国家重点图书出版规划项目  
嵌入式系统与单片机系列丛书

# 基于单片机的智能系统设计与实现

沈红卫 著

電子工業出版社

**Publishing House of Electronics Industry**

北京·BEIJING

## 内 容 简 介

本书按照“完整性、公开性、实践性、典型性”原则，主要讨论了远程监测系统、数字化调节器、汽车组合仪表指针压装控制器等 5 个智能系统的设计，其中 4 个来自生产实践。所有系统均以 51 系列单片机为基础，主要涉及 ATME1 89 系列和 Philips 89LPC 两个主流系列。介绍的实例，从单片机结构来说，既有单 CPU 系统，又有双 CPU 系统；从通信总线来说，既有 RS-232 串行总线，又有 CAN 现场总线；从开发语言来说，所有系统均以 ASM 51 和 C51 两种语言分别实现，并提供了完整的源程序。本书内容既有典型 A/D、D/A、显示等功能部件的设计与应用问题，又有许多作者的实践经验的体现及对可靠性等普遍性问题的理解。

本书内容丰富、取材典型，具有很好的实用价值，适合于从事基于单片机的智能仪器仪表系统开发的科技人员及高校师生阅读。

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

基于单片机的智能系统设计与实现/沈红卫著. —北京: 电子工业出版社, 2005.1  
(嵌入式系统与单片机系列丛书)

ISBN 7-121-00738-X

I. 基… II. 沈… III. 单片微型计算机—系统设计 IV. TP368.1

中国版本图书馆 CIP 数据核字 (2004) 第 137571 号

责任编辑: 张 榕 (zr@phei.com.cn)

印 刷: 民族印刷厂

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

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 22.25 字数: 570 千字

印 次: 2005 年 1 月第 1 次印刷

印 数: 5 000 册 定价: 32.00 元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系。联系电话: (010) 68279077。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。

# 前 言

微处理器早已广泛应用于多种领域，尤其是在智能仪器仪表中的应用更是如此，不仅引起了产品本身的变革，也深深地影响了设计理念的变革。智能仪器仪表作为一种智能系统，其核心在于微处理器。基于微处理器的智能系统设计，已经成为广大电子设计工程师或相关领域设计者关注的热点。由于智能系统面向不同的应用领域，采用不同的实现途径，使用不同的开发技术，涉及不同的学科背景，因此其设计调试过程往往是复杂而且痛苦的。尽管智能系统的设计没有普遍的规律可以遵循，但还是具有某些共性的问题值得探讨，还具有某些共性的实现理念值得相互借鉴，这正是作者写作此书的动因所在。

智能系统是一个复杂系统，一般包含微处理器、按键与显示人机界面、A/D 转换、D/A 转换等基本功能部件，同时也可能包含与应用领域相关的其他特殊部件。由于具体的系统对每一种部件有不同的要求，决定了其实现形式的多样性。智能系统一般需要在恶劣工况下长期连续运行，因此在满足功能的基础上，其可靠性也成为一個需要重点关注的问题。在多年的设计实践中，作者积累了许多可以引以为鉴的经验，摸索出了许多在智能系统设计中的共性问题，按照“完整性、公开性、实践性、典型性”的成稿原则，力图将这些毫无保留地体现在著作中。作者相信，书中涉及的许多设计理念、实现方法以及具体方案必将对阅读者产生重要影响。

全书共为 5 章，分别讨论了 5 个智能系统的设计问题，完全是作者近年来的开发实践总结和提升。由于所使用的微处理器均为目前最流行的应用最广泛的 51 系列及其兼容系列，主要是 ATMEL 89 系列和 Philips 的 89LPC932 系列。每一章按照“设计要求→方案确定→硬件设计→软件设计→系统调试→可靠性或值得进一步探讨的问题→系统样机”等环节进行阐述。其中，软件设计部分分别基于 ASM51 汇编语言和 C51 语言讨论实现方法。第 1 章讨论了电子闹钟的设计问题，尽管所讨论的系统的实际意义不大，但其阐发的设计思想和实现途径，具有重要参考意义。书中讨论的其他 4 个智能系统，均来自生产实践，具有很强的应用价值。全书给出了 5 个智能系统的所有硬件资料，以及全部程序代码。

作者在撰写过程中得到了电子工业出版社领导，尤其是赵丽松女士的大力支持与帮助，谨此致以深深的谢意！

本书由绍兴文理学院出版基金资助。

沈红卫  
于绍兴文理学院风则江畔

# 目 录

<b>第 1 章 基于 AT89C2051 的电子闹钟</b> .....	1
1.1 电子闹钟的功能与设计方案 .....	1
1.1.1 电子闹钟的功能与设计的要求 .....	1
1.1.2 设计方案的确定 .....	2
1.2 电子闹钟的硬件系统设计 .....	3
1.2.1 电子闹钟的硬件系统框架 .....	3
1.2.2 电子闹钟的主机电路设计 .....	3
1.2.3 电子闹钟的显示电路设计 .....	5
1.2.4 电源设计 .....	6
1.2.5 硬件电原理图 .....	7
1.3 电子闹钟的软件系统设计 .....	7
1.3.1 软件系统中的主模块设计 .....	8
1.3.2 基本显示模块设计 .....	11
1.3.3 当前编辑位闪烁功能的实现 .....	13
1.3.4 时间设定模块设计 .....	15
1.3.5 秒脉冲发生器原理与走时处理 .....	21
1.3.6 12 小时制与 24 小时制的实现 .....	27
1.3.7 闹铃功能的实现 .....	29
1.3.8 基于 ASM51 汇编的源程序 .....	30
1.3.9 基于 C51 的源程序 .....	50
1.4 系统的组装与调试 .....	61
1.4.1 硬件系统的组装与调试 .....	61
1.4.2 软件调试 .....	62
1.4.3 系统实物 .....	65
1.5 电子闹钟系统中可以进一步探讨的问题 .....	66
1.5.1 温度计功能的实现 .....	67
1.5.2 基于 TC77 的温度计的硬件设计 .....	67
1.5.3 温度计的软件设计 .....	68
1.6 低功耗问题及其解决思路 .....	71
<b>第 2 章 基于 AT89S8252 的远程监测系统</b> .....	74
2.1 远程监测系统的设计要求 .....	74
2.1.1 概述 .....	74
2.1.2 远程监测系统的功能与设计的要求 .....	74

2.2	远程监测系统下位机的硬件设计	75
2.2.1	下位机的硬件框架	75
2.2.2	CPU 模块设计	76
2.2.3	人机界面设计	76
2.2.4	模拟量输入通道和开关量输入通道设计	77
2.2.5	MODEM 通信接口设计	78
2.2.6	硬件系统电原理图	79
2.3	远程监测系统下位机的软件设计	81
2.3.1	下位机的软件框架	81
2.3.2	软件主模块设计	81
2.3.3	采用字符点阵液晶模块的显示模块设计	86
2.3.4	基于 AD7705 的 A/D 转换模块设计	87
2.3.5	MODEM 通信模块设计	89
2.3.6	基于软硬件看门狗的容错性设计	91
2.3.7	源程序清单	91
2.4	远程监测系统的下位机样机实物	151
2.5	远程监测系统的上位机软件设计	152
2.5.1	上位机软件的设计要求	152
2.5.2	基于 VB6.0 的上位机软件的设计	153
2.6	串口调试软件及其应用	158
<b>第 3 章</b>	<b>数字化调节器</b>	<b>159</b>
3.1	数字化调节器的应用与设计要求	159
3.1.1	数字化调节器的应用	159
3.1.2	数字化调节器的设计要求	159
3.2	数字化调节器控制系统设计	160
3.2.1	数字化调节器控制系统硬件设计	160
3.2.2	数字化调节器控制系统软件设计	164
3.3	数字化调节器显示系统设计	184
3.3.1	数字化调节器显示系统的设计要求	184
3.3.2	数字化调节器显示系统的硬件设计	185
3.3.3	数字化调节器显示系统的软件设计	187
3.3.4	数字化调节器显示系统源程序	190
3.4	数字化调节器实物	199
3.5	值得进一步探讨的问题	200
<b>第 4 章</b>	<b>基于 CAN 总线的汽车组合仪表指针压装控制器</b>	<b>201</b>
4.1	概述	201
4.2	汽车组合仪表指针压装控制器的设计要求	201
4.2.1	功能与性能设计要求	201
4.2.2	界面设计要求	202

4.2.3	压装系统的原理示意图 .....	202
4.3	汽车组合仪表指针压装控制器的设计方案 .....	202
4.3.1	控制器总体框架 .....	203
4.3.2	人机子系统和控制子系统的通信方式 .....	203
4.4	汽车组合仪表指针压装控制器的人机子系统设计 .....	204
4.4.1	人机子系统的硬件设计 .....	204
4.4.2	人机子系统的软件设计 .....	207
4.5	基于 89C52 的控制子系统设计 .....	267
4.5.1	控制子系统的硬件设计 .....	268
4.5.2	控制子系统的软件设计 .....	269
4.6	系统调试 .....	291
4.6.1	分系统和分模块调试技术 .....	291
4.6.2	CAN 通信调试 .....	292
4.7	汽车组合仪表指针压装控制器的样机 .....	303
<b>第 5 章</b>	<b>基于 89LPC932 的超声波测距仪</b> .....	<b>305</b>
5.1	概述 .....	305
5.2	超声波测距仪的设计要求 .....	306
5.2.1	设计要求 .....	306
5.2.2	设计方案的确定 .....	306
5.3	超声波测距仪的硬件设计 .....	306
5.3.1	超声波发送器的设计 .....	307
5.3.2	超声波接收器的设计 .....	307
5.3.3	基于 DS18B20 的温度测量电路 .....	308
5.3.4	基于 P89LPC932 的主机电路设计 .....	309
5.4	超声波测距仪的软件设计 .....	311
5.4.1	超声波测距仪的软件规划 .....	311
5.4.2	DS18B20 的接口程序设计 .....	311
5.4.3	显示程序设计 .....	315
5.4.4	距离计算及其程序实现 .....	318
5.4.5	主程序与 P89LPC932 的初始化程序 .....	326
5.4.6	源程序 .....	332
5.5	超声波测距仪的样机 .....	344
<b>参考资料</b>	.....	<b>345</b>

# 第 1 章 基于 AT89C2051 的电子闹钟

本章详细讨论了基于 AT89C2051 的电子闹钟的设计。该闹钟可设定起始走时时间和闹铃时间，采用数码管显示。该章的主要内容包括：亮度可调串行显示电路的设计、具有编辑位闪烁功能的编辑软件的设计、闹铃功能的软件设计、TC77 串行温度传感器的接口设计，以及系统相关的软硬件调试实践等。

## 1.1 电子闹钟的功能与设计方案

### 1.1.1 电子闹钟的功能与设计要

#### 1. 电子闹钟的实现形式

电子闹钟既可以通过纯硬件实现，也可以通过软硬件结合实现，根据电子时钟的核心部件——秒信号的产生原理，通常有三种形式，如下所述。

##### 1) 采用 NE555 时基电路的实现形式

采用 NE555 时基电路或其他振荡电路产生秒脉冲信号，作为秒加法电路的时钟信号或微处理器的外部中断输入信号，可构成电子钟。由 555 构成的秒脉冲发生器电路如图 1-1 所示。输出的脉冲信号  $V_O$  的频率  $f = 1.443 / (R_A + 2R_B) \times C$ ，可通过调节这 3 个参数，使输出  $V_O$  的频率为精确的 1Hz。

##### 2) 采用石英钟专用芯片的实现形式

采用石英钟专用计时芯片实现的电子钟，具有实现简单、计时精度高的特点。石英计时芯片（简称“机芯”）比较多，常见的型号有 STP5512F、SM5546A 和 D60400 等。现结合康巴丝石英钟常用的 5512F 型为例做一简单介绍。利用 5512F 的 2 秒输出信号作为秒加法电路的计数脉冲，可实现电子时钟。5512F 的引脚图如图 1-2 所示。

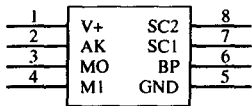


图 1-2 5512F 的引脚图

##### 3) 采用基于单片机的实现形式

利用单片机的智能性，可方便地实现具有智能的电子钟设计。由于微处理器均具有时钟

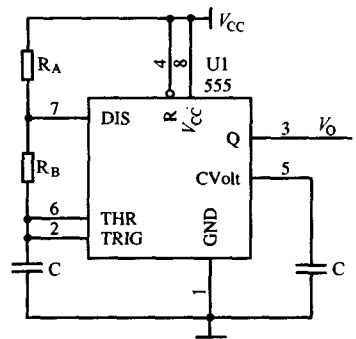


图 1-1 基于 555 的秒脉冲发生器

其中，引脚 7、8 为外接晶振及振荡电路，引脚 1 接电源正极，电源为 1.5V，引脚 3、4 原为指针用步进电机线圈的输出驱动端，这里可用 3 脚作为脉冲输出，频率决定于外接晶振的频率。



振荡系统，利用系统时钟借助微处理器的定时器/计数器可实现电子钟功能。虽然系统时钟的误差较大，电子钟的累积误差也可能较大，但可以通过误差修正软件加以修正。本章讨论的电子钟就是采用这种形式。

## 2. 电子闹钟的功能与设计要求

就电子闹钟而言，一般应具有以下基本功能要求：

(1) 能随意设定走时起始时间。对电子钟而言，最基本的功能是具有对时功能，即能随意设定走时起始时间。

(2) 能设定闹铃时间。电子钟一般都具有闹铃功能，即预设一个时间，一旦走时到该时间，电子钟能以声或光的形式告警提示（俗称“闹铃”、“打铃”）。

(3) 能指示秒节奏，即秒指示。

(4) 12 小时 / 24 小时两种制式可选，以适应不同的需要。

(5) 采用交直流供电电源。与石英钟不同的是，电子钟一般采用数码管等显示介质，因而必须以交流供电为主，以直流电源为后备辅助电源，并能自动切换。

(6) 具有走时误差修正能力。

### 1.1.2 设计方案的确立

可从以下几个方面来确定电子闹钟的设计方案。

#### 1. 微处理器

采用 ATMEL 的 AT89C2051 微处理器，是基于以下几个因素：

(1) 89C2051 为 51 内核，仿真调试软硬件资源丰富；

(2) 性价比高，货源充足；

(3) DIP20 封装，体积小，便于产品小型化；

(4) 为 E<sup>2</sup>PROM 程序存储介质，1 000 次以上擦/写周期，便于编程调试；

(5) 具有 IDLE 和 POWER-DOWN 两种工作模式，便于进行低功耗设计；

(6) 工作电压范围宽：2.7~6V，便于交直流供电。

#### 2. 显示电路

就时钟而言，通常可采用液晶显示或数码管显示。对于一般的段式液晶屏，需要专门的驱动电路，而且液晶显示作为一种被动显示，可视性相对较差；对于具有驱动电路和微处理器接口的液晶显示模块（字符或点阵），一般多采用并行接口，对微处理器的接口要求较高，占用资源多。另外，89C2051 本身无专门的液晶驱动接口，因此，本时钟采用数码管显示方式。数码管作为一种主动显示器件，具有亮度高、价格便宜等优点，而且市场上也有专门的时钟显示组合数码管。

#### 3. 按键电路

考虑到对时和设定闹铃时间这两种操作的使用频率不是很高，为了精简系统和节省成本，本时钟系统只设两个按键：

(1) SET 键, 对应系统的不同工作状态, 具有 3 个功能:

- 在复位后的待机状态下, 用于启动设定时间参数 (对时或定闹);
- 在设定时间参数状态而且不是设定最低位 (即分个位) 的状态下, 用于结束当前位的设定, 当前设定位下移;
- 在设定最低位 (分个位) 的状态下, 用于结束本次时间设定。

(2) +1 键, 用于对当前设定位 (编辑位) 进行加 1 操作, 根据 12 / 24 小时工作模式和正在编辑的当前位的含义 (时十位、时个位、分十位、分个位) 自动进行数据的上限和下限判断。例如, 对 12 小时制, 小时的十位只能是 0、1, 如果当前值为 0, 则按 +1 键后为 1, 再按 +1 键则又回复到 0。

## 1.2 电子闹钟的硬件系统设计

电子闹钟至少应包括秒信号发生器、时间显示电路、按键电路、供电电源, 以及闹铃指示电路等几部分。

### 1.2.1 电子闹钟的硬件系统框架

电子闹钟的系统框架如图 1-3 所示。在系统中, 除了按键电路以外, 还设计了“是否设定闹铃”、“12 小时 / 24 小时制选择”等按钮电路。

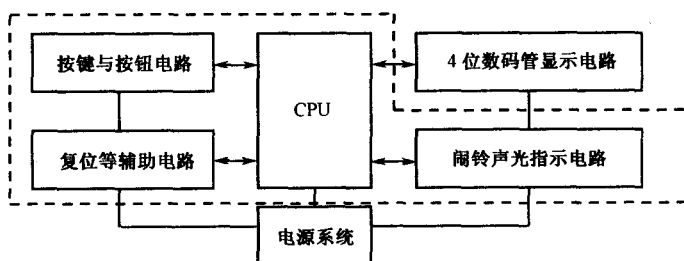


图 1-3 电子闹钟的系统框架

### 1.2.2 电子闹钟的主机电路设计

电子闹钟的主电路指的是图 1-3 中虚线框内部分, 主要涉及到微处理器电路和按键按钮电路。主机的设计具体地说有: (1) 系统时钟电路设计; (2) 系统复位电路设计; (3) 按键与按钮电路设计; (4) 闹铃声光指示电路设计。以下分别讨论:

#### 1. 系统时钟电路设计

系统时钟电路的设计如图 1-4 所示。对于时间要求不是很高的系统, 只要按图进行设计就能使系统可靠起振并稳定运行。但由于图中的 C1、C2 电容起着系统时钟频率微调 and 稳定的作用, 因此, 在本闹钟系统的实际应用中一定要注意正确选择参数 ( $30 \pm 10 \text{ pF}$ ), 并保证电路的对称性 (尽可能匹配), 选用正牌

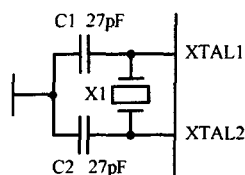


图 1-4 系统时钟电路

厂家生产的瓷片或云母电容，如果可能的话，温度系数要尽可能低。实验表明，这两个电容元件对闹钟的正负走时误差有较大关系。

## 2. 系统复位电路设计

智能系统一般应有手动或上电复位电路。复位电路的实现通常有两种方式：即专用μP 监控电路和 RC 复位电路。前者电路实现简单，成本低，但复位可靠性相对较低；后者成本较高，但复位可靠性高，尤其是高可靠重复复位。对于复位要求高、并对电源电压进行监视的场合，大多采用这种方式。

### 1) 专用μP 监控电路

专用μP 监控电路又称为电源监视电路，具有上电时可靠产生复位信号和电源电压跌落到“门限值”时可靠产生复位信号等功能。按有效电平分，有高电平输出、低电平输出两种；按功能分，有简单的电源监视复位电路、带“看门狗”定时器（WATCH DOG Timer, WDT）的监控电路和 WDT + E<sup>2</sup>PROM 的监控电路等多种类型。比较常见的生产厂家有 MAXIM、Philips、IMP 及 DALLS 等，51 系列微处理器中常用的型号有 MAX813L、MAX809、X25043/5 等。

### 2) RC 复位电路

本系统采用的是 RC 复位方式。RC 复位电路的实质是一阶充放电电路，现结合图 1-5 所示说明这种复位电路的特点。系统上电时该电路提供有效的复位信号 RST（高电平）直至系统电源稳定后撤销复位信号（低电平）。从理论上说，51 系列单片机复位引脚只要外加两个机器周期的有效信号即可复位，即只要保证  $t = RC > 2M$ （机器周期）便可。但在实际设计中，通常 C1 取值为 10μF 以上，R1 通常取值 10kΩ 左右。实践发现，R1

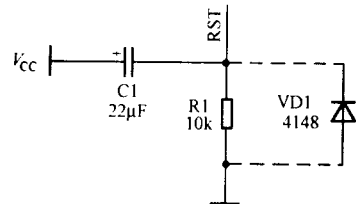


图 1-5 RC 复位电路

如果取值太小，例如 1kΩ，则会导致 RST 信号驱动能力变差而无法使系统可靠复位。另外，从图 1-6 所示的复位信号波形图可以明显看出，图 1-5 中的虚线所接的续流二极管 VD1 对于改善复位性能，起到了重要作用。它的作用是在电源电压瞬间下降时使电容迅速放电，因此一定宽度的电源毛刺（如波形中 A 点）也可令系统可靠复位。图 1-6 为加二极管前后的复位信号特性对比。

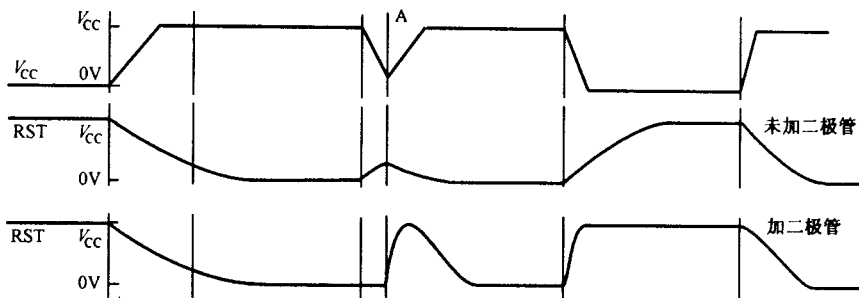


图 1-6 加二极管前后的复位信号特性对比

### 3. 按键与按钮电路设计

按键与按钮电路的设计参见系统电原理图中的 S1、S2 和 S3 对应部分。按键与按钮电路设计中关键要考虑的就是按键去抖动问题（简称“去抖”），一般有硬件去抖和软件去抖两种方式。过去硬件去抖电路通常采用分立元件或触发器实现，目前市场上已有硬件去抖专用接口芯片，例如：MAXIM 公司 MAX6816~6818，均为单电源供电，电压为 2.7~5.5V，分别为单输入、双输入和八输入，输出端具有欠压锁定功能。这里考虑到系统的硬件简化和成本，没有采用硬件去抖，而采用软件去抖。

### 4. 闹铃声光指示电路设计

闹铃指示可以有声或光两种形式，本系统采用声音指示。关键元件是蜂鸣器。蜂鸣器有无源和有源两种，前者需要输入声音频率信号才能正常发声，后者则只需外加适当直流电源电压即可；元件内部已封装了音频振荡电路，在得电状态下即起振发声。市场上的有源蜂鸣器分为 3V、5V、6V 等系列，以适应不同的应用需要。其电路设计参见系统电原理图。其中 PNP 小功率三极管 Q2 采用 9012，其最大集电极电流为 800mA，完全满足蜂鸣器驱动的需要。适当调节基极电阻可改变蜂鸣器的发声功率（即响度）。

### 1.2.3 电子闹钟的显示电路设计

显示采用共阳数码管，其目的是为了简化限流电路的设计和实现亮度可调的要求。4 位共阳数码管显示电路如图 1-7 所示。从图中可知，该显示电路采用了与一般的段电流电阻限流

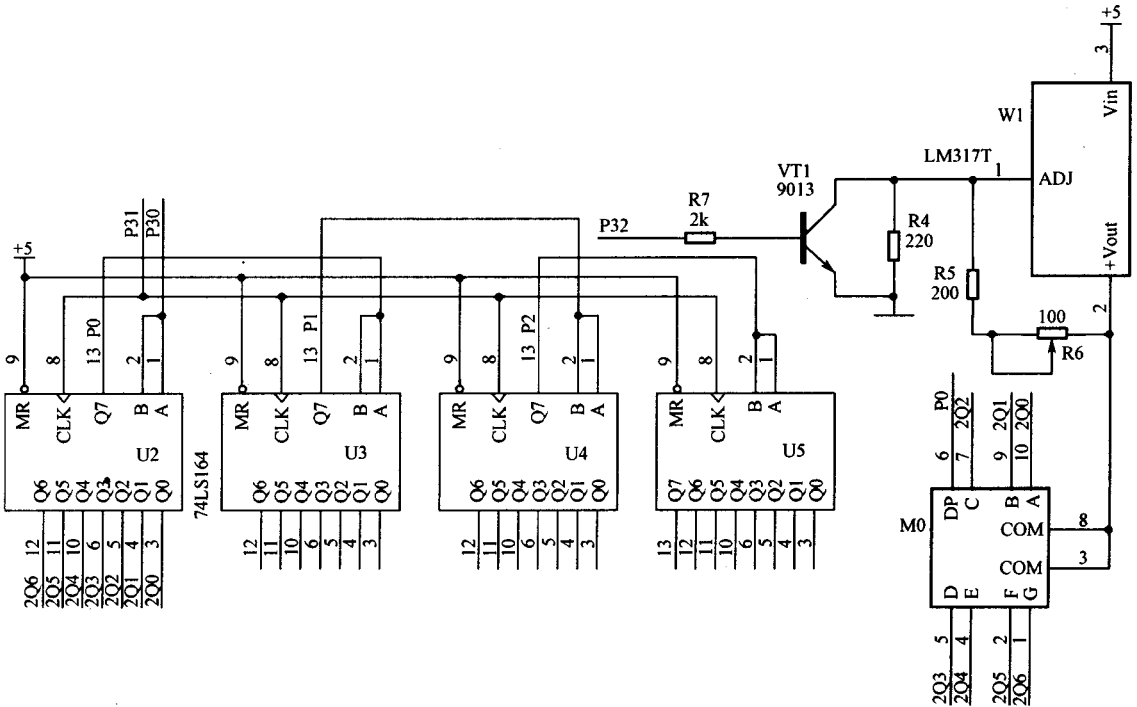


图 1-7 4 位共阳数码管显示电路

方式不同的实现方式，由此减少了  $4 \times 8 = 32$  个限流电阻，简化了硬件系统。每笔画段二极管正常发光时的电流一般为 10mA 左右（当然，电流大小取决于选用的数码管是普亮、高亮还是超高亮类型的不同），其两端压降约为 2.0V，也就是说，只要数码管的公共端（COM）加 +2.0V 以上电压，即可满足每笔画段发光二极管的发光要求，而且适当调节此电压值即可改变发光二极管的电流，从而达到调节亮度的目的。此电压采用三端可调稳压电路 W1（LM317）来实现。其输入为 +5V，按照图中参数，其输出电压由式（1-1）决定：

$$1.25 \times \left( 1 + \frac{R_4}{R_5 + R_6} \right) \tag{1-1}$$

在式（1-1）中， $R_5$  为 200Ω， $R_6$  可调， $R_4$  为 220Ω，因此输出电压为 2.17~2.63V。但由于输入输出压差至少为 2.5V，因此极限电压为 2.5V。

为了节约 CPU 的端口数，显示采用了串行通信口的串行显示接口方式，利用串口的 0 工作方式，在发送 TXD 端口（P31）的时钟信号的作用下，通过接收 RXD 端口（P30）将显示段码串行数据送入 8 位串入并出移位寄存器 74LS164，控制相应的数码管。图 1-7 中所示只画出了一个数码管的连接。端口 P32 的作用是通过 LM317 控制数码管显示的开启与关闭，当 P32 为低电平时，Q1 关断，317 的输出电压低于 1.5V，不足以发光，避免了在显示数据刷新时显示的抖动现象。

### 1.2.4 电源设计

由于 89C2051 通常有 -12 和 -24 两种型号，对应的时钟频率分别为 12MHz 和 24MHz，前者的工作电压为 2~6V，后者的工作电压为 4~6V，这点在设计选用时应当予以注意。考虑到交直流两用的要求和三端稳压电路选用的方便（通常的系列为 5V，6V，…），选择工作电压为 5V。电源系统设计如图 1-8 所示。

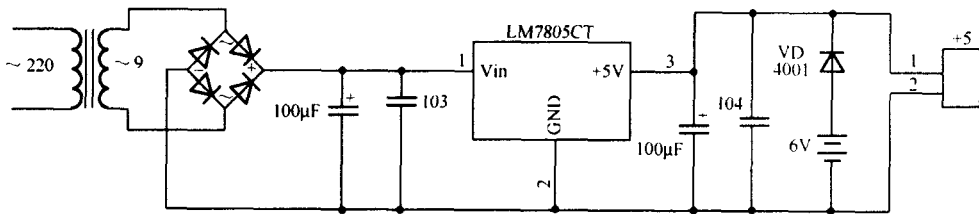


图 1-8 闹钟的电源系统原理图

应当说明的是，尽管有很多型号的 7805 三端稳压集成芯片，其标称最大输出电流均为 1.5A，但在实际应用中，该最大输出电流值往往取决于两个方面：（1）足够的散热面积；（2）不同的生产厂家。按照很多开发者的经验，ST 公司的 7805 三端稳压块能接近标称值。在设计中，必须保证 7805 的输入电压  $V_i$  和输出电压  $V_o$  的压差大于 2.5V，即  $V_i - V_o \geq 2.5V$ ，否则会失去稳压能力。同时考虑到功耗问题，此压差又不宜太大，太大则增加 7805 本身的功率消耗，增加芯片的温升，不利于安全。因此，选为 9V。当交流电源失电或失效时，电压为 6V 的直流电源（电池组或蓄电池）通过二极管投入作用，硅二极管的导通电压降约为 0.2V，因此满足系统的电源要求。

### 1.2.5 硬件电原理图

系统的硬件电原理图如图 1-9 所示。

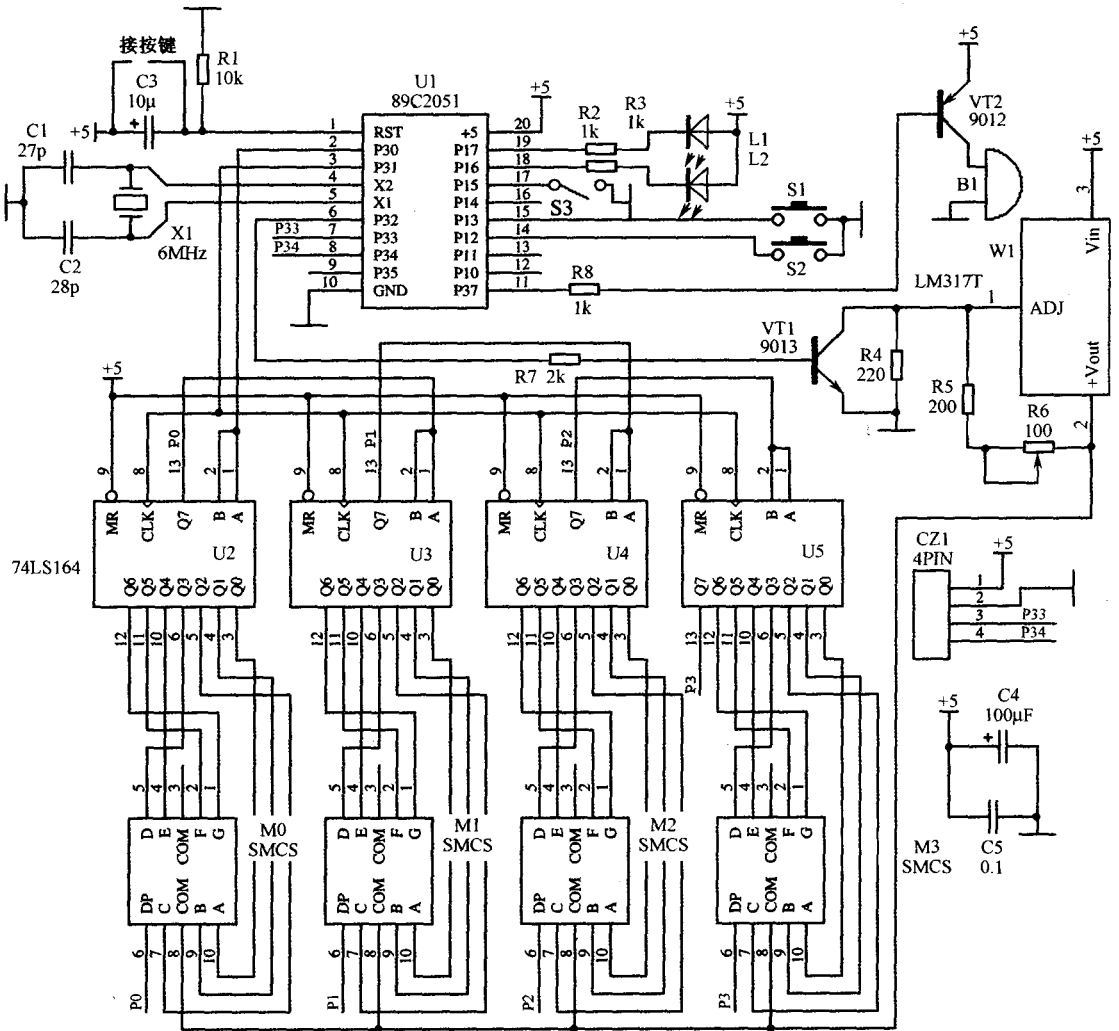


图 1-9 电子闹钟系统的硬件电原理图

### 1.3 电子闹钟的软件系统设计

软件设计的重点在于秒脉冲信号的产生、显示的实现及按键的处理等方面。关于显示实现和按键处理的软件设计参见本节相应小节。这里主要讨论秒脉冲的产生原理。基于软件的秒脉冲信号通常有延时法和定时中断法。延时法一般采用查询方式，在延时子程序前后必然需要查询和处理的程序，导致误差的产生，因此其秒脉冲的精度不高。中断法的原理是，利用单片机内部的定时器溢出中断来实现。例如，设定某定时器每 100ms 中断 1 次，则 10 次的周期为 1s。这种实现法的特点是精度高，秒脉冲的发生和其他处理可以并行进行。本系统

即采用这种方式，实现的关键是定时器工作方式的选择和定时参数的计算设定。具体内容参见源程序中的相关说明。本系统中所使用的晶振频率为 6MHz。

### 1.3.1 软件系统中的主模块设计

主模块是系统软件的主框架。结构化程序设计一般有“自上而下”和“自下而上”两种方式，“自上而下”法的核心就是主框架的构建。它的合理与否关系到程序最终功能的多少和性能的好坏。本系统的主模块的程序框图可用图 1-10 来表示。其具体代码实现以下分别讨论。注意，代码中的有关变量的含义和作用要参阅源程序清单部分的说明。

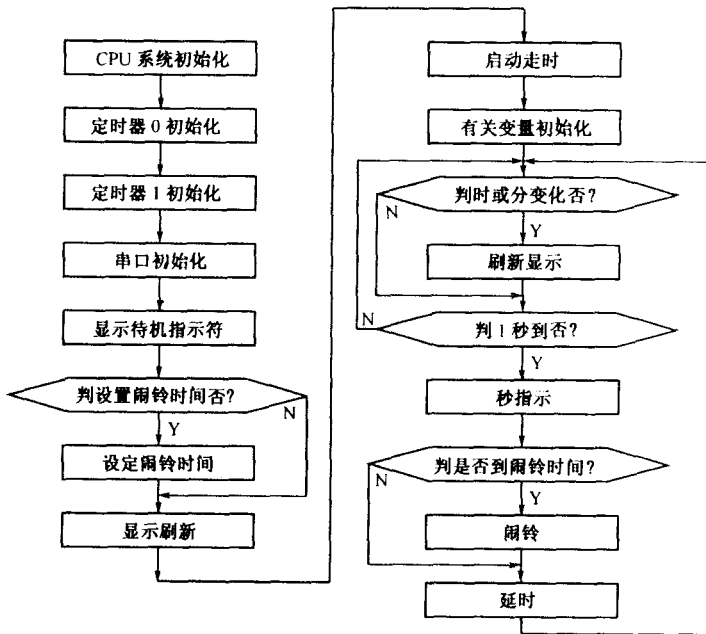


图 1-10 主模块的程序框图

#### 1. 主模块的 ASM51 汇编语言实现

以下程序为上述主模块的 51 汇编语言实现，有关设计思想在程序中均做了详细分析说明。

```

MAIN:
    MOV     SP,    #60H    ; 堆栈设置
    LCALL  SYS_INIT    ; 系统初始化
    LCALL  T0_INIT    ; 定时器初始化

    LCALL  T1_INIT    ; 闪烁用定时器初始化

    LCALL  UART_INIT    ; 串口初始化
    LCALL  DISPLAYP    ; 显示待机符 P.
    LCALL  SETTIME    ; 等待设置当前时间
  
```

```

JB      S3,  MAIN_00    ; 判断是否设置闹铃
LCALL   SETWARN        ; 是, 则设定闹铃时间

      SETB   WARNSETTED ; 设置已设定闹铃时间标志
      MOV    WARNCNT,#0  ; 闹铃次数清 0 (否则会闹 1 分钟)
      CLR    TFIX        ; 误差开始修正标志初始化
      MOV    COUNT, #0

MAIN_00:
      LCALL  DISPLAY1    ; 必须再次刷新, 否则会出现尾位不显示的情况

MAIN_0:
      SETB   TR0         ; 设置完后开始走时, 启动定时器 T0
      SETB   ET0         ; 允许 T0 溢出中断

MAIN_1:
      JB     FRESH,MAIN_2 ; 时(分)有变化, 则刷新显示
      SJMP  MAIN_3        ; 否则, 不更新显示

MAIN_2:
      LCALL  DISPLAY1    ; 先对时分数据进行 HEX2BCD 转换后显示
      SJMP  MAIN_3        ; 进入死循环, 只有复位才能重新设置

MAIN_3: JBC   SEC, MAIN_4 ; 判断秒到否
      SJMP  MAIN_1

MAIN_4: CPL   L1         ; 到则进行秒指示
      CPL   L2
      JNB   WARNING,MAIN_5
      CPL   BEEP         ; 开始闹铃
      INC   WARNCNT      ; 闹铃次数加 1
      MOV   A, WARNCNT
      XRL  A, #08
      JNZ  MAIN_5        ; 到 8 次 (实际是 4 次), 则停止
      CLR   WARNING
      MOV   WARNCNT, #0

MAIN_6: SETB  BEEP
MAIN_5: MOV   R6, #60
      LCALL DEL20_0      ; 点亮时间 (不能太长, 否则影响秒指示, 短则太暗)

      SJMP  MAIN_1

```

## 2. 主模块的 C51 语言实现

以下代码为上述主模块的 C51 语言实现, 该实现中没有闹铃时间设定功能。



```

void main(void)
{
    sys_init();           /* 系统初始化 */
    t0_init();           /* 定时器初始化 */
    t1_init();
    uart_init();         /* 串口初始化 */
    buf[0]=0x0a;         /* “P.”的段码送显示缓冲区 */
    buf[1]=0x0b;
    buf[2]=0x0b;
    buf[3]=0x0b;
    display();           /* 显示待机符 P */
    settime();           /* 等待设置当前时间 */
    display();           /* 带闪烁后要加显，否则会出现尾位不显的现象 */
    TR0=1;               /* 设置完后开始走时，启动定时器 T0 */
    ET0=1;               /* 允许 T0 溢出中断 */
    while(1)
    {
        if(fresh)       /* 判断是否要刷新显示（分或时变化） */
        {
            buf[0]=hourh; /* 是，则新的时间送显示缓冲区 */
            buf[1]=hourl;
            buf[2]=minuh;
            buf[3]=minul;
            display();     /* 显示刷新 */
            if(sec)       /* 判断秒指示要刷新否 */
            {
                sec=0;    /* 是，则刷新（闪烁） */
                L1=!L1;
                L2=!L2;
                delay20ms(); /* 点亮时间不能太长，否则影响秒指示，短则太暗 */
            }
        }
        else
        {
            if(sec)
            {
                sec=0;
                L1=!L1;
                L2=!L2;
            }
        }
    }
}

```