

万年历设计报告

学院：武夷学院

班级：09 电信 1 班

组员：林巧文

一、设计要求与方案论证 2

1.1 设计要求：2

1.1.1 根本要求 2

1.1.2 发挥局部 2

1.2 系统根本方案选择和论证 2

1.2.1 单片机芯片的选择方案和论证：2

1.2.2 显示模块选择方案和论证：2

1.2.3 时钟芯片的选择方案和论证：3

1.3 电路设计最终方案决定 3

二、理论分析与计算 3

2.1, 秒数的产生由定时器 T0 产生：3

三、系统的硬件设计与实现 4

3.1 电路设计框图：4

3.2 系统硬件概述：4

3.3 主要单元电路的设计 5

3.3.1 单片机主控制模块的设计 5

3.3.2 显示模块的设计 5

3.3.3 闹钟模块的设计 7

3.3.4 电源稳压模块 7

四、系统的软件设计 7

4.1 程序流程框图 7

4.2 闹钟模块流程图：7

4.3 按键调整模块流程图：7

五、测试方案与测试结果分析 7

5.1 测试仪器..... 7

5.2 软件测试平台 Keil C518

5.3 模块测试 8

5.3.1 显示模块测试 8

5.4 测试结果分析与结论 8

5.4.1 测试结果分析 8

5.4.2 测试结论..... 8

六、作品总结 9

参考文献 9

附录一：系统电路图 9

附录三：系统 C 程序 10

一、设计要求与方案论证

1.1 设计要求:

根本要求

- (1) 准确显示: 时、分、秒 (24 小时制)
- (2) 显示星期
- (3) 显示公历
- (4) 时间、日期、星期可调节
- (5) 断电记忆功能

发挥局部

- (1) 闹钟功能
- (2) 显示阴历
- (3) 显示 24 节气
- (4) 其他

1.2 系统根本方案选择和论证

单片机芯片的选择方案和论证:

方案一:

采用 89C51 芯片作为硬件核心, 采用 Flash ROM 内部具有 4KBROM 存储空间, 能于 3V 的超低压工作, 而且与 MCS-51 系列单片机完全兼容, 但是运用于电路设计中时由于不具备 ISP 在线编程技术, 当在对电路进展调试时, 由于程序的错误修改或对程序的新增功能需要烧入程序时, 对芯片的屡次拔插会对芯片造成一定的损坏。

方案二:

采用 AT89S52, 片内 ROM 全都采用 Flash ROM; 能以 3V 的超底压工作; 同时也与 MCS-51 系列单片机完全该芯片内部存储器为 8KB ROM 存储空间, 同样具有 89C51 的功能, 且具有在线编程可擦除技术, 当在对电路进展调试时, 由于程序的错误修改或对程序的新增功能需要烧入程序时, 不需要对芯片屡次拔插, 所以不会对芯片造成损坏。所以选择采用 AT89S52 作为主控制系统。

显示模块选择方案和论证:

方案一:

采用点阵式数码管显示，点阵式数码管是由八行八列的发光二极管组成，对于显示文字比拟适合，如采用在显示数字显得太浪费，且价格也相对较高，所以不用此种作为显示。

方案二：

采用 LED数码管动态扫描,LED数码管价格适中,对于显示数字最适宜,但无法显示图形文字,在显示星期是也只能用数字表示,而且采用动态扫描法与单片机连接时,在编程时比拟复杂。所以也不采用了 LED数码管作为显示。

方案三：

采用 LCD液晶显示屏,液晶显示屏的显示功能强大,可显示文字,图形,显示多样,清晰可见,所以在此设计中采用 LCD液晶显示屏。

时钟芯片的选择方案和论证：

方案一：

采用 DS1302时钟芯片实现时钟，DS1302芯片是一种高性能的时钟芯片，可自动对秒、分、时、日、周、月、年以及闰年补偿的年进展计数，而且精度高,位的 RAM做数据暂存区，工作电压 2.5V~5.5V*围内，2.5V 时耗电小于 300nA.但在，题目中明确规定不能选用不可以使用任何专用的时钟芯片和模块，因此不采用此方案。

方案二：

直接采用单片机定时计数器提供秒信号，使用程序实现年、月、日、星期、时、分、秒计数。采用此种方案可以减少芯片的使用，也可以节约本钱。所以采用此方案。

1.3 电路设计最终方案决定

综上各方案所述,对此次作品的方案选定：采用 AT89S52 作为主控制系统;采用单片机定时器提供时钟；液晶显示屏作为显示。

二、理论分析与计算

2.1, 秒数的产生由定时器 T0 产生：

T0 置 10ms 定时初值 dc00H(2^16=65536D,dc00H=56320D)

定时时间=(65536-56320)*(1/11.0592)*12=10ms (单片机晶振 f0=11.0592Mhz)

用累加的方法让定时时间累加一百次即可产生一秒时间：100*10ms=1000ms=1s

2.2. 总天数的算法： 首先用 if 语句判断定义年到输入年之间每一年是否为闰年，

是闰年，该年的总天数为 366，否则，为 355。然后判断输入的年是否为定义年，假设是，令总天数 S=1，否则，用累加法计算出定义年到输入年之间的总天数，再把输入年的一月到要输出的月份之间的天数累加起来，假设该月是闰年中的月份并且该月还大于二月，再使总

天数加 1，否则，不加，既算出从定义年一月一日到输出年的该月一日的总天数。2.3 输

出月份第一天为星期几的算法：使总天数除以 7 取余加 2 得几即为星期几，假设是 7，则为星期日。

2.4. 算出输出月份的完整日历算法：算出输出月份第一天为星期几后，把该日期以前的位置用空格补上，并总该日起一次输出天数直到月底，该月中的天数加上该月一日为星期几的数字再除以 7 得 0 换行，即可完整的输出该月的日历。

5. 计算*一天是星期几的算法：蔡勒 (Zeller) 公式

历史上的*一天是星期几. 未来的*一天是星期几. 关于这个问题，有很多计算公式 (两个通用计算公式和一些分段计算公式)，其中最著名的是蔡勒 (Zeller) 公式。即：

$$w=y+[y/4]+[c/4]-2c+[26(m+1)/10]+d-1$$

公式中的符号含义如下，w: 星期；c: 世纪-1；y: 年 (两位数)；m 月 (m 大于等于 5，小于等于 14，即在蔡勒公式中，*年的 1、2 月要看作上一年的 13、14 月来计算，比方 2003 年 1 月 1 日要看作 2002 年的 13 月 1 日来计算)；d: 日；[] 代表取整，即只要整数局部。(C 是世纪数减一，y 是年份后两位，M 是月份，d 是日数。1 月和 2 月要按上一年的 13 月和 14 月来算，这时 C 和 y 均按上一年取值。) 算出来的 W 除以 7，余数是几就是星期几。如果余数是 0，则为星期日。以 2049 年 10 月 1 日 (100 周年国庆) 为例，用蔡勒 (Zeller) 公式进展计算，过程如下：

$$\text{蔡勒 (Zeller) 公式: } w=y+[y/4]+[c/4]-2c+[26(m+1)/10]+d-1$$

$$=49+[49/4]+[20/4]-2 \times 20+[26 \times (10+1)/10]+1-1$$

$$=49+[12.25]+5-40+[28.6]$$

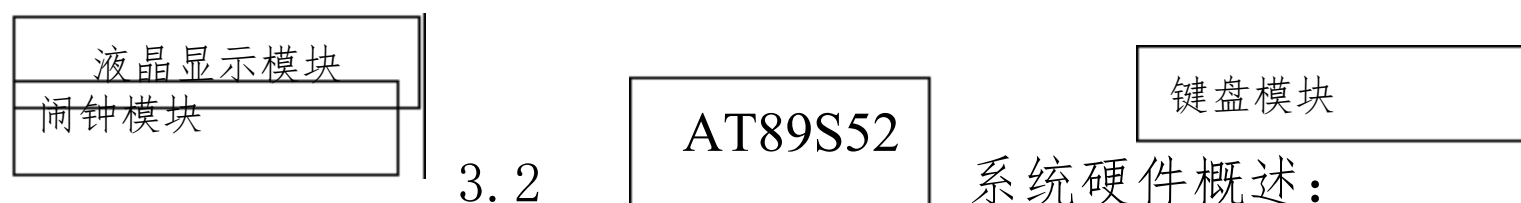
$$=49+12+5-40+28$$

$$=54 \text{ (除以 7 余 5)}$$

即 2049 年 10 月 1 日 (100 周年国庆) 是星期 5。

三. 系统的硬件设计与实现

3.1 电路设计框图：



本电路是由 AT89S52 单片机为控制能在 3V 超低压工作；显示部份即 32 个字符；闹钟局部由蜂鸣器构子讯响器，采用直流电压供电，广报警器、电子玩具、汽车电子设备、件。

AT89S52
主控制
模块

系统硬件概述：

核心，具有在线编程功能，低功耗，LCD1602 实现，能够同时显示 16*02 成，蜂鸣器是一种一体化构造的电泛应用于计算机、打印机、复印机、机、定时器等电子产品中作发声器件。

3.3 主要单元电路的设计

单片机主控制模块的设计

AT89S52 单片机为 40 引脚双列直插芯片,有四个 I/O 口 P0,P1,P2,P3, MCS-51单片机共有 4 个 8 位的 I/O 口 (P0、P1、P2、P3), 每一条 I/O 线都能独立地作输出或输入。单片机的最小系统如下列图所示,18 引脚和 19 引脚接时钟电路,*TAL1 接外部晶振和微调电容的一端,在片内它是振荡器倒相放大器的输入,*TAL2 接外部晶振和微调电容的另一端,在片内它是振荡器倒相放大器的输出. 第 9 引脚为复位输入端,接上电容,电阻及开关后够上电复位电路,20 引脚为接地端,40 引脚为电源端. 如图-1 所示

图-1 主控制系统

显示模块的设计

LCD1602液晶显示屏 工业字符型液晶,能够同时显示 16*02 即 32 个字符。(16 列 2 行),有 16 个引脚,通过 D0~D7 的 8 位数据端传输数据和指令,引脚图如图 6-5。

图 6-5LCD1602引脚示意图

LCD1602液晶显示屏引脚功能如表 6-5 所示:

表 6-5 LCD1602 引脚功能

管脚 1	V_{ss}	一般接地
管脚 2	V_{dd}	接电源 (+5V)
管脚 3	V_0	液晶显示器比照度调整端,接正电源时比照度最弱,接地电源时比照度最高
管脚 4	RS	RS 为寄存器选择,高电平 1 时选择数据寄存器、低电平 0 时选择指令寄存器
管脚 5	R/W	R/W 为读写信号线,高电平 (1) 时进展读操作,低电平 (0) 时进展写操作。
管脚 6	E	E(或 EN)端为使能(enable) 端,下降沿使能。
管脚 7	DB_0	底 4 位三态、双向数据总线 0 位
管脚 8	DB_1	底 4 位三态、双向数据总线 1 位
管脚 9	DB_2	底 4 位三态、双向数据总线 2 位
管脚 10	DB_3	底 4 位三态、双向数据总线 3 位
管脚 11	DB_4	高 4 位三态、双向数据总线 4 位
管脚 12	DB_5	高 4 位三态、双向数据总线 5 位
管脚 13	DB_6	高 4 位三态、双向数据总线 6 位
管脚 14	DB_7	高 4 位三态、双向数据总线 7 位
管脚 15	BLA	背光电源正极
管脚 16	BLK	背光 电源负极

1602 液晶模块内部的控制器共有 11 条控制指令,如表 10-14 所示:

序号	指令	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0
1	清显示	0	0	0	0	0	0	0	0	0	1
2	光标返回	0	0	0	0	0	0	0	0	1	*
3	置输入模式	0	0	0	0	0	0	0	1	I/D	S
4	显示开/关控制	0	0	0	0	0	0	1	D	C	B
5	光标或字符移位	0	0	0	0	0	1	S/C	R/L	*	*
6	置功能	0	0	0	0	1	DL	N	F	*	*
7	置字符发生存贮器地址	0	0	0	1	字符发生存贮器地址					
8	置数据存贮器地址	0	0	1	显示数据存贮器地址						
9	读忙标志或地址	0	1	BF	计数器地址						
10	写数到 CGRAM 或 DDRAM	1	0	要写的数字内容							
11	从 CGRAM 或 DDRAM 读数	1	1	读出的数字内容							

表 10-14: 控制命令表

1602 液晶模块的读写操作、屏幕和光标的操作都是通过指令编程来实现的。(说明: 1 为高电平、0 为低电平)

指令 1: 清显示, 指令码 01H, 光标复位到地址 00H 位置。

指令 2: 光标复位, 光标返回到地址 00H。

指令 3: 光标和显示模式设置 I/D: 光标移动方向, 高电平右移, 低电平左移 S: 屏幕上所有文字是否左移或者右移。高电平表示有效, 低电平则无效。

指令 4: 显示开关控制。D: 控制整体显示的开与关, 高电平表示开显示, 低电平表示关显示 C: 控制光标的开与关, 高电平表示有光标, 低电平表示无光标 B: 控制光标是否闪烁, 高电平闪烁, 低电平不闪烁。

指令 5: 光标或显示移位 S/C: 高电平时移动显示的文字, 低电平时移动光标。

指令 6: 功能设置命令 DL: 高电平时为 4 位总线, 低电平时为 8 位总线 N: 低电平时为单行显示, 高电平时双行显示 F: 低电平时显示 5*7 的点阵字符, 高电平时显示 5*10 的点阵字符。

指令 7: 字符发生器 RAM 地址设置。

指令 8: DDRAM 地址设置。

指令 9: 读忙信号和光标地址 BF: 为忙标志位, 高电平表示忙, 此时模块不能接收命令或者数据, 如果为低电平表示不忙。

指令 10: 写数据。

指令 11: 读数据。

将 lcd 的引脚与单片机连接, 通过单片机实现对显示的输出, 电路图实现如下:

闹钟模块的设计

闹钟模块采用蜂鸣器实现，蜂鸣器是一种一体化构造的电子讯响器，采用直流电压供电。通过单片机的 p3.7 引脚输出的电平变化来控制蜂鸣器的导通与否，设计如下：

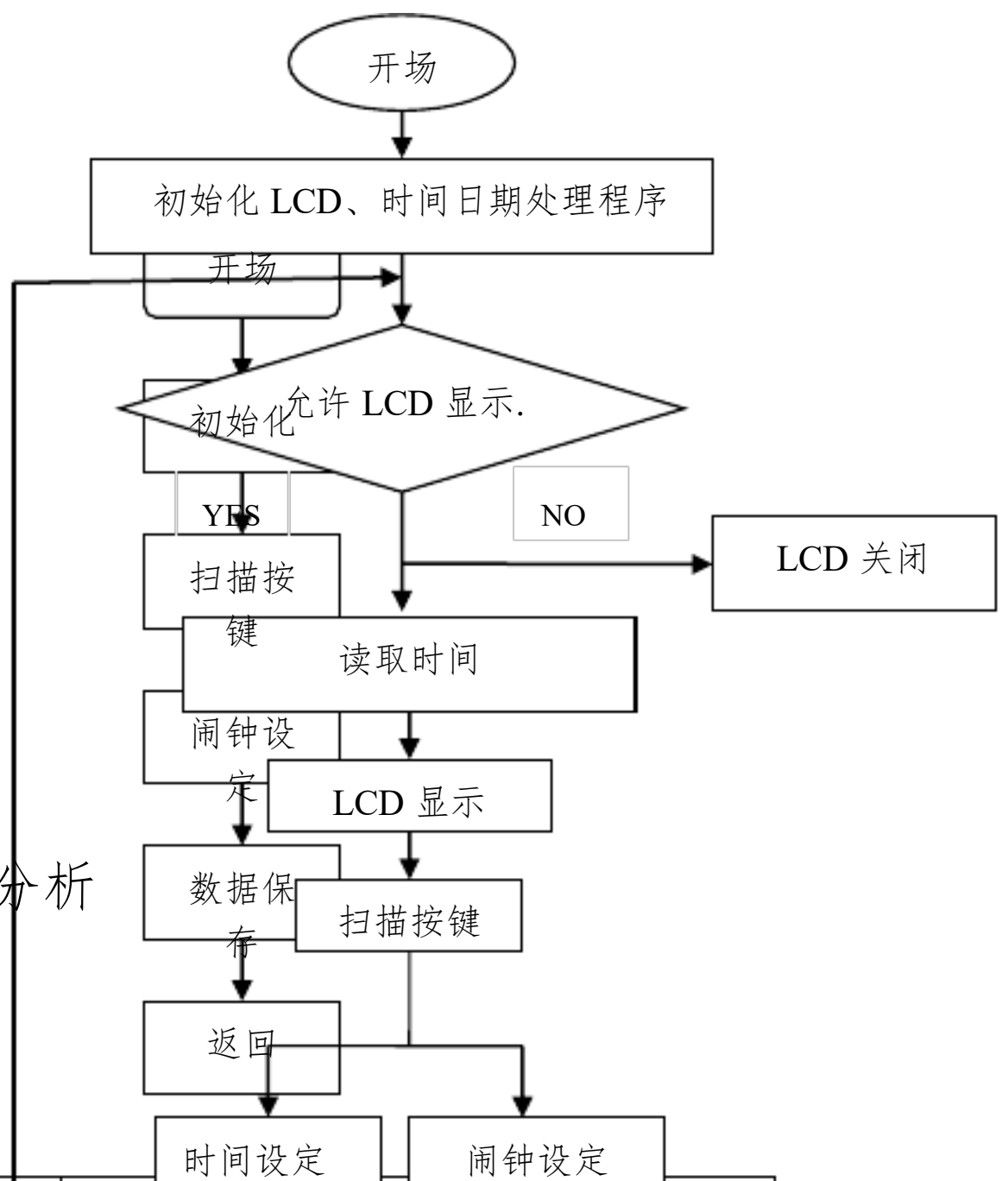
电源稳压模块

方案一：LM7805稳压

方案二：USB接口+5V供电

四、系统的软件设计

4.1 程序流程框图



4.2 闹钟模块流程图：

4.3 按键调整模块流程图：

五、测试方案与测试结果分析

5.1 测试仪器

序号	名称	作用
1	KELL 51 软件	编写调试 C 程序并生成可烧录 he* 文件
2	仿真软件 Proteus	对电路进展焊接前的仿真与测试
3	变压器	提供 5 伏输出电源
4	数字万用表	测试工作电源检测电路运行状况

5.2 软件测试平台 Keil C51

Keil C51 是美国 Keil Software 公司出品的 51 系列兼容单片机 C 语言软件开发系统，与汇编相比，C 语言在功能上、构造性、可读性、可维护性上有明显的优势，因而易学易用。用过汇编语言后再使用 C 来开发，体会更加深刻。Keil C51 软件提供丰富的库函数和功能强大的集成开发调试工具，全 Windows 界面。另外重要的一点，只要看一下编译后生成的汇编代码，就能体会到 Keil C51 生成的目标代码效率非常之高，多数语句生成的汇编代码很紧凑，容易理解。在开发大型软件时更能表达高级语言的优势。下面详细介绍 Keil C51 开发系统各局部功能和使用。Keil C51 工具包的整体构造，其中 uVision 与 Ishell 分别是 C51 for Windows 和 for Dos 的集成开发环境(IDE)，可以完成编辑、编译、连接、调试、仿真等整个开发流程。

5.3 模块测试

显示模块测试

出现的问题：LCD1602 液晶显示屏背景灯亮但不显示内容。

解决问题：初步假设电源已经供电但单片机没有工作。用万用表测试 LCD 使能端 6 引脚和写命令选择端 4 引脚，均有电位变化，推出单片机已经工作。再次假设，LCD1602 的显示屏比照度没调好。让电路工作，慢慢旋转变阻器，但仍无内容显示。三次假设，LCD1602 些引脚短路，经检测各引脚电平，发现 6 脚线路*地方与电源 VCC 相连，导致数据命令无法写入 LCD1602 显示器，导致内容无法显示，改正后，显示正常，问题解决。VCC1602MO**01)

5.4 测试结果分析与结论

测试结果分析

在测试过程中遇到 lcd1602 显示不出来，首先使用万用表对电路进展测试，检查工作电压是否正常并查找出电路没正常工作的局部，再次检查各引脚接线情况。

5.4.2 测试结论

经过历次的反复测试与分析，可以对电路的原理及功能更加熟悉，同时提高了设计能力与及对电路的分析能力。同时在软件的编程方面得到更到的提高，对编程能力得到加强。同时对所学的知识得到很大的提高与稳固。

六、作品总结

在整个设计过程中，发挥团队精神，分工合作，我吴余壮健负责电路设计，软件编程，设计报告的编写，李培文负责网上查阅相关有用资料，王文龙负责电路的焊接，充分发挥人的主观能动性，自主学习，学到了许多没学到的知识。较好的完成了作品。到达了预期的目的，在最初的设计中，发挥“三个臭皮匠，顶个诸葛亮”的作用。相互学习、相互讨论、研究。完了最初的设想。在此次设计中，知道了做凡事要有一颗平常的心，不要想着走捷径，一步一脚印。也练就了我们的耐心，做什么事都在有耐心。此次比赛中学到了很多很多东西，这是最重要的。总之，参加电子竞赛我们的能力得到了全方位的提高。

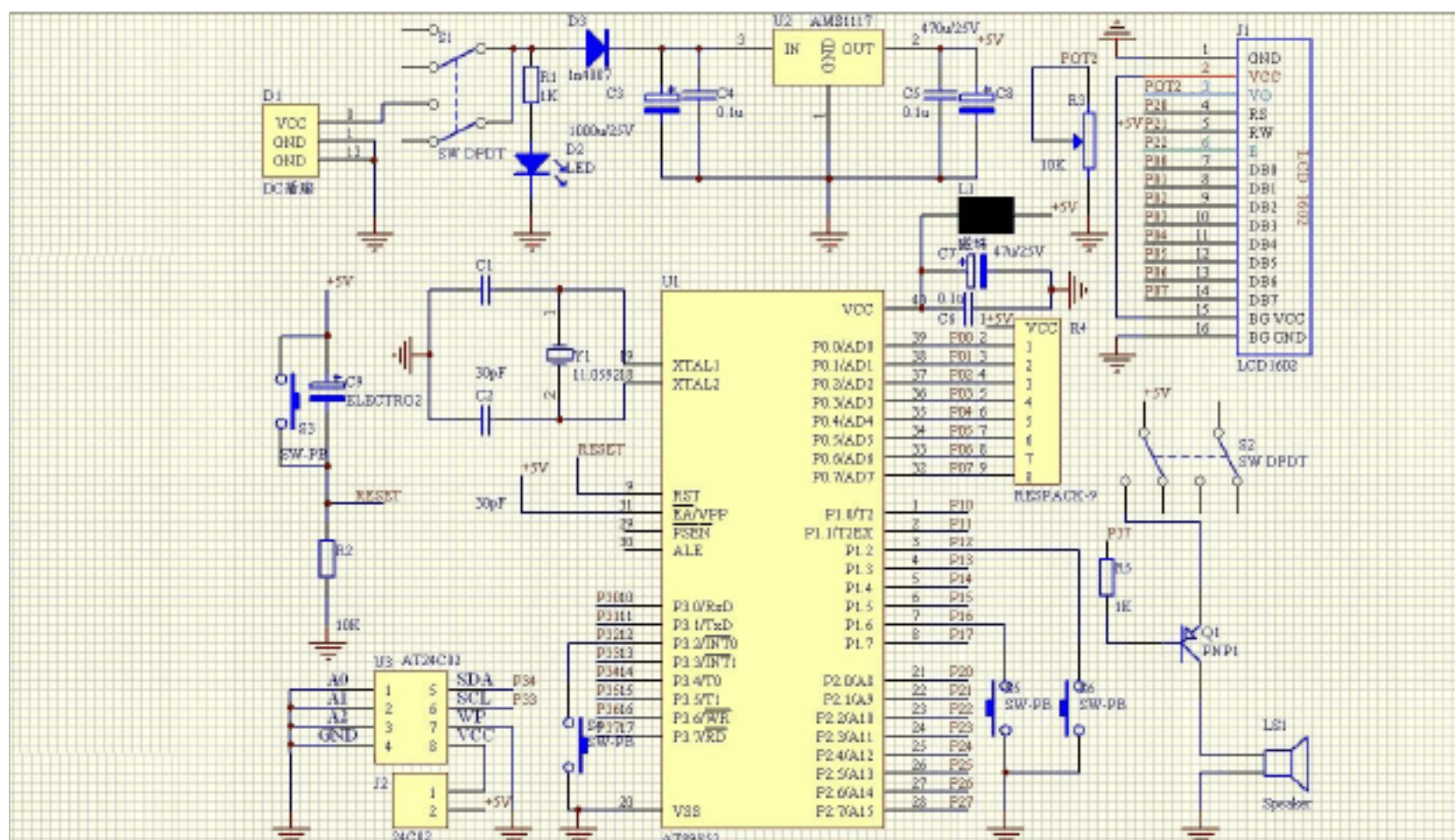
参考文献

新概念 51 单片机 C 语言教程 郭天祥编 电子工业 2009

单片机 C 语言应用一百例 王东锋 王会良 董冠强编电子工业 2009

杨子文 编 单片机原理及应用 **电子科技大学 2006

附录一：系统电路图



附录二：仿真效果图

C 程序

```
*include<reg52.h>
#include<intrins.h>

unsigned char code para_month[13]={0,0,3,3,6,1,4,6,2,5,0,3,5}; //
星期月参变数
unsigned char data dis_buf1[16]; //lcd 上排显示缓冲区
unsigned char data dis_buf2[16]; //lcd 下排显示缓冲区
unsigned char data year,month,date,week;// 年、月、日、星期
unsigned char data armhour,armmin,armsec;// 闹钟时、分、秒
unsigned char data hour,min,sec,sec100; // 时、分、秒、百分之一秒
unsigned char data flag,vkey,skey;// 设置状态计数标志、按键先前值、
按键当前值
bit alarm; // 标识是否启用闹钟, 1--启用, 0--关闭
sbit rs = P2^0; //LCD 数据/命令选择端(H/L)
sbit rw = P2^1; //LCD 读/写选择端(H/L)
sbit ep = P2^2; //LCD 使能控制
sbit PRE = P1^6; // 调整键(k3)
sbit SET = P1^7; // 调整键(k4)
sbit SPK = P3^7;
void delayms(unsigned char ms); // 延时程序
bit lcd_busy(); // 测试 LCD忙碌状态程序
void lcd_wcmd(char cmd); // 写入指令到 LCD程序
void lcd_wdat(char dat); // 写入数据到 LCD程序
void lcd_pos(char pos); //LCD 数据指针位置程序
void lcd_init(); //LCD 初始化设定程序
void pro_timedate(); // 时间日期处理程序
void pro_display(); // 显示处理程序
void pro_key(); // 按键处理程序
void time_alarm(); // 定时报警功能(闹钟)
unsigned char scan_key(); // 按键扫描程序
unsigned char week_proc(); // 星期自动计算与显示函数
bit leap_year(); // 判断是否为闰年
void lcd_sef_chr(); //LCD 自定义字符程序
void update_disbuf(unsigned char t1,unsigned char t2[],unsigned
char dis_h,unsigned char dis_m,unsigned char dis_s); // 更新显示
```

```

// 延时程序
void delay(unsigned char ms)
{ while(ms--)
  { unsigned char i;
    for(i = 0; i < 250; i++)
      {
        _nop_(); // 执行一条_nop_() 指令为一个机器周期
        _nop_();
        _nop_();
        _nop_();
      }
    }
}
//测试 LCD忙碌状态
bit lcd_busy()
{
  bit result;
  rs = 0;
  rw = 1;
  ep = 1;
  _nop_();
  _nop_();
  _nop_();
  _nop_();
  result = (bit)(P0 & 0x80); //LCD 的 D0--D7 中,D7=1 为忙碌,D7=0 为空闲
  ep = 0;
  return result;
}
//写入指令到 LCD
void lcd_wcmd(char cmd)
{
  while(lcd_busy()); // 当 lcd_busy 为 1 时,再次检测 LCD忙碌状态,lcd_busy 为 0 时,开场写指令
  rs = 0;
  rw = 0;
  ep = 0;
}

```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/417011030116006044>