

单片机定时计数器设计方案

1 设计方案

1.1 设计要求

用 AT89S52 单片机，在 1602LCD 上显示年月日、星期、时分秒设计数字钟，有 4 组可设置的定时开/关，控制一路开关量输出。定时开关的设置分为单次操作(2011 年 6 月 25 日 8: 00 开 2011 年 6 月 26 日 18: 00 关)，周期操作（如周三 8: 00 开，周四 16: 00 关）。数码管显示时间（单位秒）与设置值（单位分钟），用发光二极管表示周期与单次操作。

1.2 设计方案论证

本设计是纯粹的应用性设计，主要以实现计时、定时、温度显示等功能为最终目的。设计以 Atmel 公司的 AT89S52 为核心，在最小系统的基础上扩展键盘，数字温度计，以及 LCD 显示模块，硬件设计简单节约，其功能的实现主要靠软件设计，所以软件在本设计中是最为重要的，在计时设计思路上有两种方案可以选取。

方案一：选用外部时钟芯片，单片机只负责其数据的存取读写以及外部辅助电路的扩展，这种方案的优点是计时准确，软件编写相对简单，但其要求对其芯片必须要有比较深的了解

方案二：利用单片机的部时钟和定时器进行计数，这种方案电路简单，容易实现，但时间容易丢失，且长时间计数会有误差，而且软件编写也相对复杂。

考虑到本次设计的目的，采用方案二，并加入了年月日的扩展，由于时间和精力有限年月日只是进行简单的模拟，没有精确的进行计算。只是进行适当的扩展。

本设计包括硬件设计和软件设计两部分。主要硬件有 AT89S52 单片机、字符型液晶显示模块 HY1602A 和若干按键等。软件大致思路为：使用 12MHz 的晶振，单片机部的定时器 0 工作在方式 1，每计数 50000 个机器周期（即 50ms）产生一次中断，中断 20 次就是一秒，这样就可以实现精确计时的目的。在把实时数据显示在 LCD1602 上的同时，不断扫描按键，如果有按键按下，则对按键做出相应的响应。

2 硬件设计

2.1 器件选型

本设计选取主控 MCU 为 Atmel 公司的 AT89S52，显示器件选择市面上常见的字符型液晶显示模块 HY1602A。

2.2 器件介绍

2.2.1 单片机 AT89C51

AT89C51 是一种带 4K 字节闪烁可编程可擦除只读存储器（Flash Programmable and Erasable Read Only Memory）的低电压，高性能 CMOS8 位微处理器，俗称单片机。该器件采用 ATMEL 高密度非易失存储器制造技术制造，与工业标准的 MCS-51 指令集和输出管脚相兼容。由于将多功能 8 位 CPU 和闪烁存储器组合在单个芯片中，ATMEL 的 AT89C51 是一种高效微控制器，为很多嵌入式控制系统提供了一种灵活性高且价廉的方案。

1. 主要特性：

与 MCS-51 兼容

4K 字节可编程闪烁存储器

寿命：1000 写/擦循环

数据保留时间：10 年

全静态工作：0Hz-24Hz

三级程序存储器锁定

128*8 位部 RAM

32 可编程 I/O 线

两个 16 位定时器/计数器

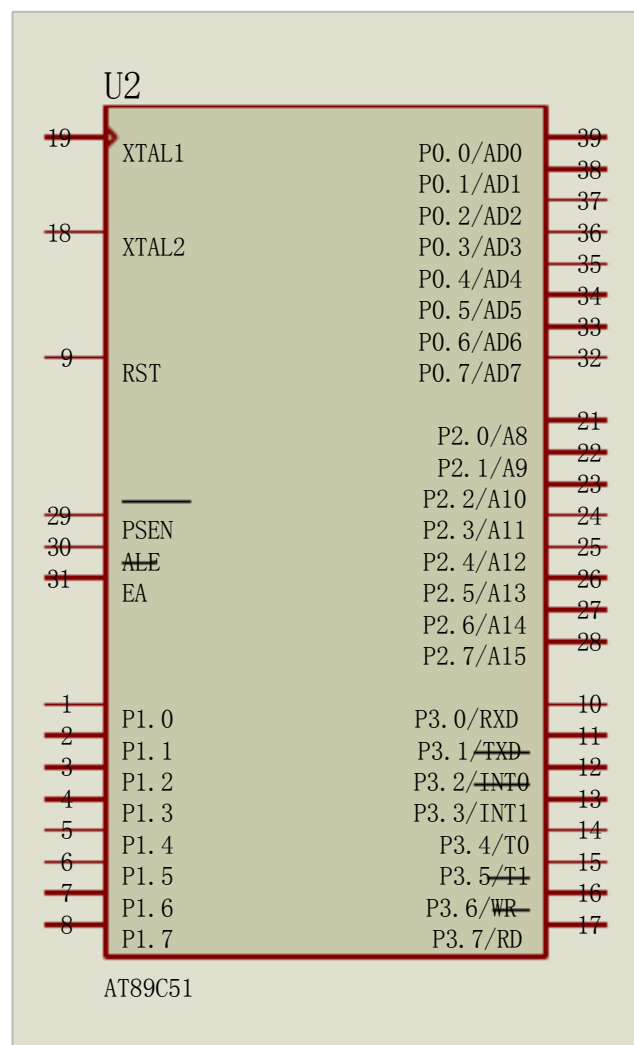
5 个中断源

可编程串行通道

低功耗的闲置和掉电模式

片振荡器和时钟电路

2. 管脚说明：



VCC：供电电压。

GND：接地。

P0 口：P0 口为一个 8 位漏级开路双向 I/O 口，每脚可吸收 8TTL 门电流。当 P1 口的管脚第一次写 1 时，被定义为高阻输入。P0 能够用于外部程序数据存储器，它可以被定义为数据/地址的第八位。在 FIASH 编程时，P0 口作为原码输入口，当 FIASH 进行校验时，P0 输出原码，此时 P0 外部必须被拉高。

P1 口：P1 口是一个部提供上拉电阻的 8 位双向 I/O 口，P1 口缓冲器能接收输出 4TTL 门电流。P1 口管脚写入 1 后，被部上拉为高，可用作输入，P1 口被外部下拉为低电平时，将输出电流，这是由于部

上拉的缘故。在 FLASH 编程和校验时，P1 口作为第八位地址接收。

P2 口：P2 口为一个部上拉电阻的 8 位双向 I/O 口，P2 口缓冲器可接收，输出 4 个 TTL 门电流，当 P2 口被写“1”时，其管脚被部上拉电阻拉高，且作为输入。并因此作为输入时，P2 口的管脚被外部拉低，将输出电流。这是由于部上拉的缘故。P2 口当用于外部程序存储器或 16 位地址外部数据存储器进行存取时，P2 口输出地址的高八位。在给出地址“1”时，它利用部上拉优势，当对外部八位地址数据存储器进行读写时，P2 口输出其特殊功能寄存器的容。P2 口在 FLASH 编程和校验时接收高八位地址信号和控制信号。

P3 口：P3 口管脚是 8 个带部上拉电阻的双向 I/O 口，可接收输出 4 个 TTL 门电流。当 P3 口写入“1”后，它们被部上拉为高电平，并用作输入。作为输入，由于外部下拉为低电平，P3 口将输出电流（ILL）这是由于上拉的缘故。

P3 口也可作为 AT89C51 的一些特殊功能口，如下表所示：

备选功能

P3.0 RXD（串行输入口）

P3.1 TXD（串行输出口）

P3.2 /INT0（外部中断 0）

P3.3 /INT1（外部中断 1）

P3.4 T0（记时器 0 外部输入）

P3.5 T1（记时器 1 外部输入）

P3.6 /WR（外部数据存储器写选通）

P3.7 /RD（外部数据存储器读选通）

P3 口同时为闪烁编程和编程校验接收一些控制信号。

RST：复位输入。当振荡器复位器件时，要保持 RST 脚两个机器周期的高电平时间。

ALE/PROG：当访问外部存储器时，地址锁存允许的输出电平用于锁存地址的地

位字节。在 FLASH 编程期间，此引脚用于输入编程脉冲。在平时，ALE 端以不变的频率周期输出正脉冲信号，此频率为振荡器频率的 1/6。因此它可用作对外部输出的脉冲或用于定时目的。然而要注意的是：每当用作外部数据存储器时，将跳过一个 ALE 脉冲。如想禁止 ALE 的输出可在 SFR8EH 地址上置 0。此时，ALE 只有在执行 MOVX，MOVC 指令是 ALE 才起作用。另外，该引脚被略微拉高。如果微处理器在外部执行状态 ALE 禁止，置位无效。

/PSEN：外部程序存储器的选通信号。在由外部程序存储器取指期间，每个机器周期两次/PSEN 有效。但在访问外部数据存储器时，这两次有效的/PSEN 信号将不出现。

/EA/VPP：当/EA 保持低电平时，则在此期间外部程序存储器（0000H-FFFFH），不管是否有外部程序存储器。注意加密方式 1 时，/EA 将部锁定为 RESET；当/EA 端口保持高电平时，此间外部程序存储器。在 FLASH 编程期间，此引脚也用于施加 12V 编程电源（VPP）。

XTAL1：反向振荡放大器的输入及内部时钟工作电路的输入。

XTAL2：来自反向振荡器的输出。

3. 振荡器特性：

XTAL1 和 XTAL2 分别为反向放大器的输入和输出。该反向放大器可以配置为片振荡器。石英振荡和瓷振荡均可采用。如采用外部时钟源驱动器件，XTAL2 应不接。有余输入至内部时钟信号要通过一个二分频触发器，因此对外部时钟信号的脉宽无任何要求，但必须保证脉冲的高低电平要求的宽度。

4. 芯片擦除：

整个 PEROM 阵列和三个锁定位的电擦除可通过正确的控制信号组合，并保持 ALE 管脚处于低电平 10ms 来完成。在芯片擦操作中，代码阵列全被写“1”且在任何非空存储字节被重复编程以前，该操作必须被执行。

此外，AT89C51 设有稳态逻辑，可以在低到零频率的条件下静态逻辑，支持两种软件可选的掉电模式。在闲置模式下，CPU 停止工作。但 RAM，定时器，计数器，串口和中断系统仍在工作。在掉电模式下，保存 RAM 的容并且冻结振荡器，禁止所用其他芯片功能，直到下一个硬件复位为止。

2.2.2 LCD1602 显示模块

工业字符型液晶，能够同时显示 16x02 即 32 个字符。（16 列 2 行）

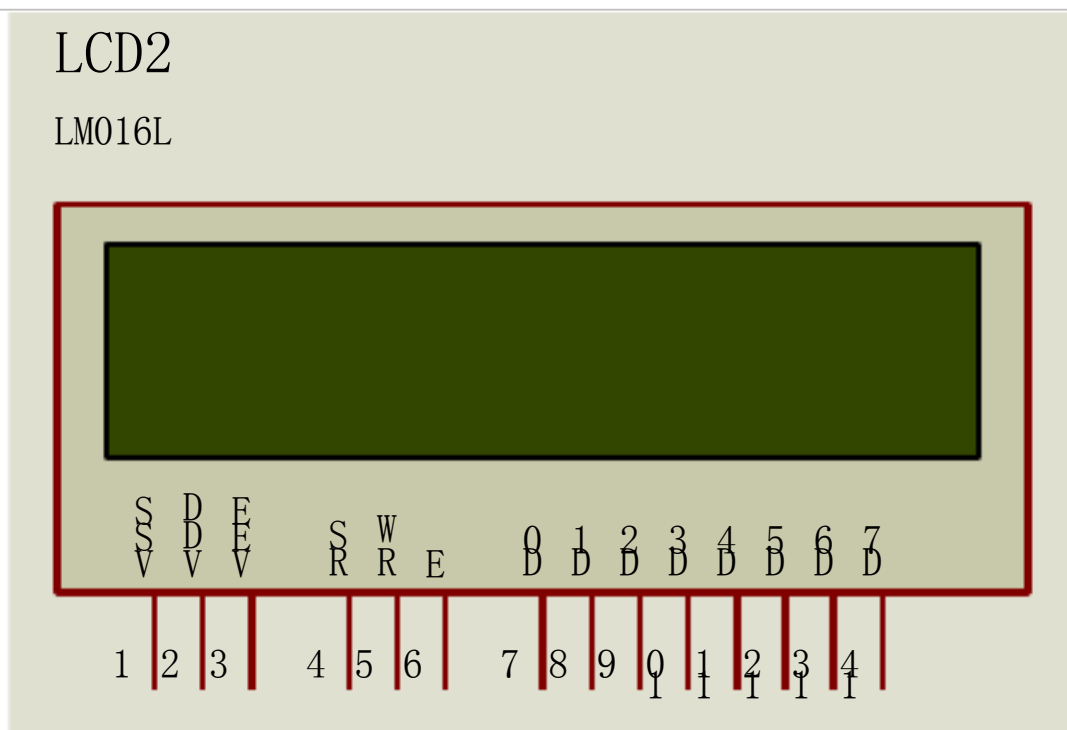
注：为了表示的方便，后文皆以 1 表示高电平，0 表示低电平。

1602 液晶也叫 1602 字符型液晶，它是一种专门用来显示字母、数字、符号等的点阵型液晶模块。它由若干个 5X7 或者 5X11 等点阵字符位组成，每个点阵字符位都可以显示一个字符，每位之间有一个点距的间隔，每行之间也有间隔，起到了字符间距和行间距的作用，正因为如此所以它不能很好地显示图形（用自定义 CGRAM，显示效果也不好）。

1602LCD 是指显示的容为 16X2, 即可以显示两行，每行 16 个字符液晶模块（显示字符和数字）。

市面上字符液晶大多数是基于 HD44780 液晶芯片的，控制原理是完全相同的，因此基于 HD44780 写的控制程序可以很方便地应用于市面上大部分的字符型液晶屏。

管脚功能



1602 采用标准的 16 脚接口，其中：

第 1 脚：VSS 为电源地

第 2 脚：VCC 接 5V 电源正极

第 3 脚：V0 为液晶显示器对比度调整端，接正电源时对比度最弱，接地电源时对比度最高（对比度过高时会产生“鬼影”，使用时可以通过一个 10K 的电位器调整对比度）。

第 4 脚：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。

第 5 脚：RW 为读写信号线，高电平 (1) 时进行读操作，低电平 (0) 时进行写操作。

第 6 脚：E(或 EN) 端为使能(enable) 端，高电平 (1) 时读取信息，负跳变时执行指令。

第 7~14 脚：D0~D7 为 8 位双向数据端。

第 15~16 脚：空脚或背灯电源。15 脚背光正极，16 脚背光负极。

3.3V 或 5V 工作电压，对比度可调

含复位电路

提供各种控制命令,如:清屏、字符闪烁、光标闪烁、显示移位等多种功能
有 80 字节显示数据存储器 DDRAM

建有 192 个 5X7 点阵的字型的字符发生器 CGROM

8 个可由用户自定义的 5X7 的字符发生器 CGRAM

3 软件设计

3.1 软件设计思路

3.1.1 实现功能

用 AT89S52 单片机,在 1602LCD 上显示年月日、星期、时分秒。有 4 组可设置的定时开/关,控制一路开关量输出。定时开关的设置分为单次操作(2011 年 6 月 25 日 8:00 开 2011 年 6 月 26 日 18:00 关),周期操作(如周三 8:00 开,周四 16:00 关)。

具备闹钟和时间段的设定功能,闹铃方式为蜂鸣器持续响 10 秒钟,或者用按键手动关闭响铃。可以用键盘调整时间、设定开关时间和方式。

复位时间是 2011 年 1 月 1 日 00:00:00。

3.1.2 编程思路

1. 给以下变量赋初值:年(Year)、月(Month)、日(Day)、时(Hour)、分(Min)、秒(Sec)。

2. 因为晶振频率为 12MHz,所以计数频率为 $12\text{MHz}/12=1\text{MHz}$,这就需要计数器 0 计数 1000000 次才可以达到一秒钟。我们用定时器 0 工作在方式 1 状态,每次计数 50000 次,如此进行 20 次,即可凑够一秒钟。这样可以得到初始化定时器的控制字: $\text{TMOD}=0\text{x}01$,计数初值 $65536-50000=15536$ 即 $0\text{x}3\text{cb}0$,这样就可以设计一个计数一秒的函数。

3. 读相应变量的值，送入 LCD 显示时间等信息。

4. 不断的扫描按键，并且响应按键输入。在进入设定模式后，所有时间变量被过渡变量代替，如果确认设置，则过渡变量赋给时间变量，如果取消设置，则还使用原时间变量。

3.1.3 操作及功能介绍

定时器能显示年月日、星期、时分秒。有 4 组可设置的定时开/关，控制一路开关量输出。定时开关的设置分为单次操作(2011 年 6 月 25 日 8:00 开 2011 年 6 月 26 日 18:00 关)，周期操作(如周三 8:00 开，周四 16:00 关)。下面具体介绍按键和 LED 的功能：

1、K1 为时间的小时设定键，按下后将进入时间的设定，按 K1 是对时间小时的设定。

2、K2 为时间的分钟设定键，当处于时间的调整时，按它可以对时间的分钟进行调整，当未处于时间调整时，按下第一次为开关量的开启时间，按下第二次为开关量的关闭时间。

3、K3 为设定时间完成的退出键和进入开关量时间的设定键，但时间设置完成以后，按下 K3 键将进入正常的计时状态；当要进入定时任务时间设定时，按下此键将进入定时任务开始时间的设定，设定完成后退出进入正常计时，然后再按下此键进入定时任务关闭时间的设定。

4、K4 为定时任务是否开启的设定键，若 LED 的黄灯是亮的说明任务是开启的，若没有亮则是关闭的。

5、K5 为定时任务运行的方式：周期与单次的切换，若绿灯亮则为周期，没亮则为单次。

6、K6、K7、K8 依次为年月日的设定键。

3.2 主程序流程图

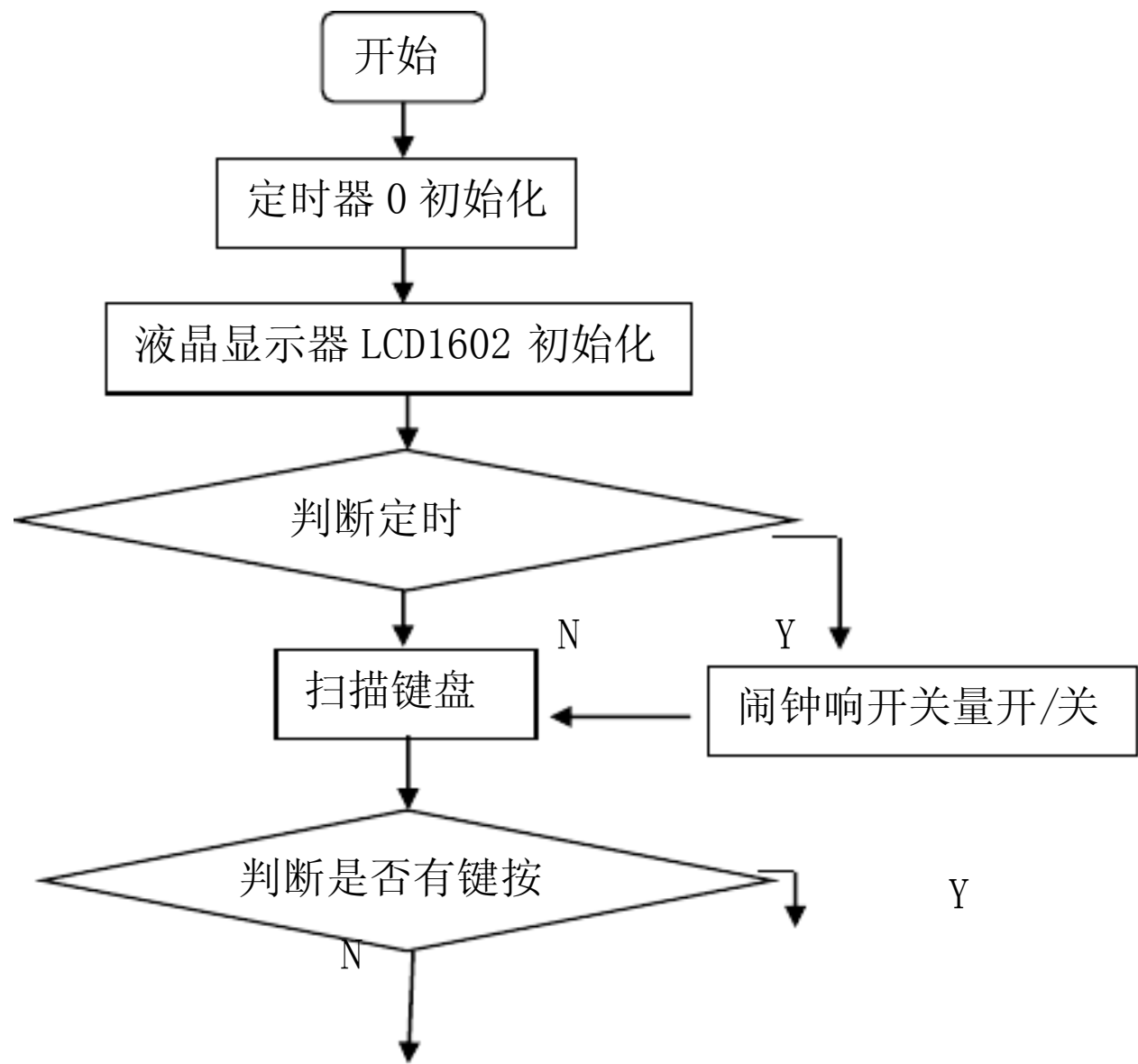


图 3-1 主程序流程图

4 仿真图

5 设计体会及小结

本设计在 Proteus 仿真软件测试成功，但是，此电子钟有自身的先天缺陷。首先，由于采用单片机部定时器产生时间信号，所以如果时钟意外掉电，则会丢失当前的时间和闹钟等数据，需要重新设定。由于本人能力和时间有限，只能暂时实现本既定方案。

课设也大大锻炼了我们的自学能力。比如我们要用到的很多芯片，不可能样样都在课本上学习过的，学过的毕竟是少数，在以后的学习实践和工作中，要用的东西五花八门，怎么可能四年都学遍呢？而且像我们学的这门迅速发展中的学科，知识更新一日千里，要准备好学习一辈子的，在大学里课堂上所学的只是方法，至于其中的容，还要靠自己去挖掘，所谓“师傅领进门，修行在

个人”就是这个道理，这才是成年人的学习生活，因此，通过课设，我们掌握了一项很基本但对于我们来讲很重要且具有十分深刻意义的技能，就是学会自学，这是前所未有的。

参考文献

- [1]王质朴，吕运鹏. MCS-51 单片机原理接口与应用，理工大学，2009.
- [2]学坚；周斌. 微型计算机原理及应用[M].：清华大学，2001
- [3]谭浩强. C 程序设计（第三版）. 清华大学，2007.
- [4]康光华. 电子技术基础（模拟部分）.：高等教育，2004.
- [5] 康光华. 电子技术基础[M].：高等教育，2006：124-135.
- [6] 谭浩强. C 程序设计[M].：清华大学，2005：82-135.

附录 1 硬件电路设计图

附录 2 主程序

```
#include<reg52.h>
#define uint unsigned int
#define uchar unsigned char

uint
miao, fen, fen1, shi, shi1, flag1, flag2, flag3, aa, K1num, K2num, K3num, K4num, N, i, da
n1, k, fen2, shi2, a, m, Month=1, Year=2011, Day=1, K6num, num, Year1, Year2, Day1, Day2
, Month1, Month2;          //K1num-K1 按键被按下的标记变量, K2num-K2 按键被按
下的标记变    量, K3num=0-K3 按键被按下的标记变量
sbit K1=P1^0;  //四个按键
sbit K2=P1^1;
sbit K3=P1^2;
sbit K4=P1^3;
sbit K5=P1^4;
sbit K6=P1^5;
sbit K7=P1^6;
sbit K8=P1^7;
sbit dan=P2^6;
sbit dan2=P2^7;
sbit rs=P2^0;
sbit lcden=P2^2;
sbit led=P2^3; //发光二极管控制端
sbit beep=P2^4; //蜂鸣器控制端
sbit relay=P2^5; //继电器控制端
void delay(uint z) //延时函数
{
    uint x, y;
    for(x=z; x>0; x--)
        for(y=110; y>0; y--);
}
```

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/078116077047007006>