智能仪器原理与设计技术

实 验 指 导 书

东南大学仪器科学与工程学院

2007 年

前　　言

“XYZ22 型综合实验仪”适用于测控类及弱电类专业大学本科生以下课程的实验和实践教学：《MCS-51 单片机原理、接口及应用》、《检测（测控）技术及系统设计》、《智能仪器原理与设计技术》等；也适用于测控类及弱电类专业硕士研究生以下课程的实验和实践教学：《智能仪器设计》、《智能测控系统设计》等；也可以被相关专业的大专生选用。

本实验装置采用模块化设计，将有相互联系的各专业课程的实验组合起来，包含这些专业课程要求的若干单个实验，最后形成“从传感器→前置调理、接口电路→A/D 转换电路→计算机（单片机）→后置接口电路→测控对象”接近实际应用系统的软硬件设计调试综合实验。训练和提高学生在检测（测控）技术及智能技术应用系统方面的设计调试能力。同时，帮助学生将各专业课程内容综合起来，融会贯通，形成系统的概念，迅速迈过从理论到实际的门槛。

大多数实验的实验内容都设置了基本要求和高级要求，能够满足不同层次学生的需要。

仪器仪表的发展可以简单的划分为三代：第一代为指针式仪器仪表，如指针式万用表、功率表等。它们的基本结构是电磁式的，基于电磁测量原理使用指针来显示最终的测量结果。第二代为数字式仪器仪表，如数字电压表、数字功率计、数字频率计等。它们的基本结构中离不开模/数转换环节，并以数字方式显示或打印测量结果。第二代仪表响应速度较快、测量精度较高。第三代就是智能式仪器仪表，这类仪器仪表的主要特征是内含微处理器，因此，通常具有信息采集、数据处理、输出控制以及自动进行测试过程和测试结果显示、记录、传输等功能。有的智能仪器还能辅助专家进行推理、分析或决策。

祝学云

目　　录

第一章 实验设备介绍..... 1
一、实验设备的联接..... 1
二、伟福（WAVE）G6W 型单片机仿真开发器介绍..... 1
三、XYZ22 型综合实验仪介绍..... 2
第二章　显示器及键盘实验..... 11
实验一　显示器及键盘实验..... 11
第三章　程控放大器及程控信号发生器实验..... 16
实验二　程控放大器实验..... 16
实验三　程控信号发生器实验..... 18
第四章　键盘键值远距离传输实验..... 23
实验四　键盘键值远距离传输实验..... 23
第五章　四位半数字电压表实验..... 28
实验五　四位半数字电压表实验..... 28
第六章　数字滤波程序设计实验..... 33
实验六　数字滤波程序设计实验..... 33
第七章　非线性校正及标度变换程序设计实验..... 39
实验七　非线性校正及标度变换程序设计实验..... 39

第一章　实验设备介绍

一．实验设备的联接

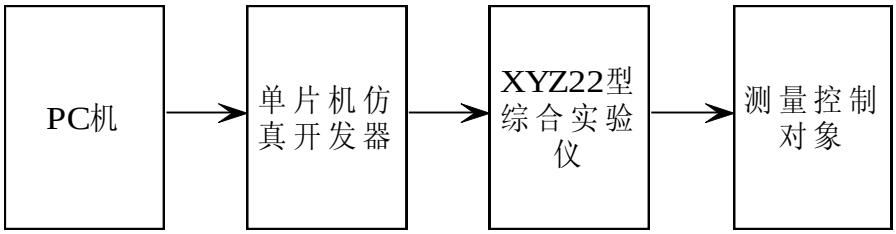


图 1　设备联接

二．伟福（WAVE）G6W 型单片机仿真开发器介绍

G6W 型单片机仿真开发器含有 WINDOWS、DOS 版本双平台；

可选择不同配置的仿真头（仿真不同的 CPU）；

仿真头上有一个晶振跳线器（选用内、外晶振），出厂设置选用内晶振；

打开电源时，先接通仿真器电源，再接通外部用户板（实验仪）的电源。防止 CMOS 型的 CPU 芯片过流，造成芯片温度过高。

1．安装及基本参数设置

1）安装

进入 C:\ 盘符下，建立子目录 C:\51,将编译器安装到 C:\51 目录下即可。

2) 基本参数设置

选择伟福汇编器；

选择 G6W 及 POD-51 型仿真头、8031 CPU；

选择程序空间在仿真器上、数据空间在用户板上；

选择仿真器与 PC 机连接的串行端口。

2．编辑、编译、执行及修改文件

1）编辑及修改

新建一个文件或打开一个文件，对此文件进行编辑及修改，最后将其保存。汇编文件的文件名后缀为 .ASM。

2）编译

对当前窗口文件进行编译、校验，如果程序文件有错，信息窗口将提示出错信息，以便用户修改。

3）执行与退出

包括全速、跟踪、单步、执行到光标处等几种程序执行方式。

3. 在窗口中检查 RAM 及特殊功能寄存器的内容

全速运行程序后，要先“暂停”再检查存储器内容。

CPU 窗口包含源程序的反汇编代码、特殊功能寄存器状态等。

数据窗口包含内部 RAM 和外部 RAM 的状态等。

三．XYZ22 型综合实验仪介绍

一）面板功能块分布图

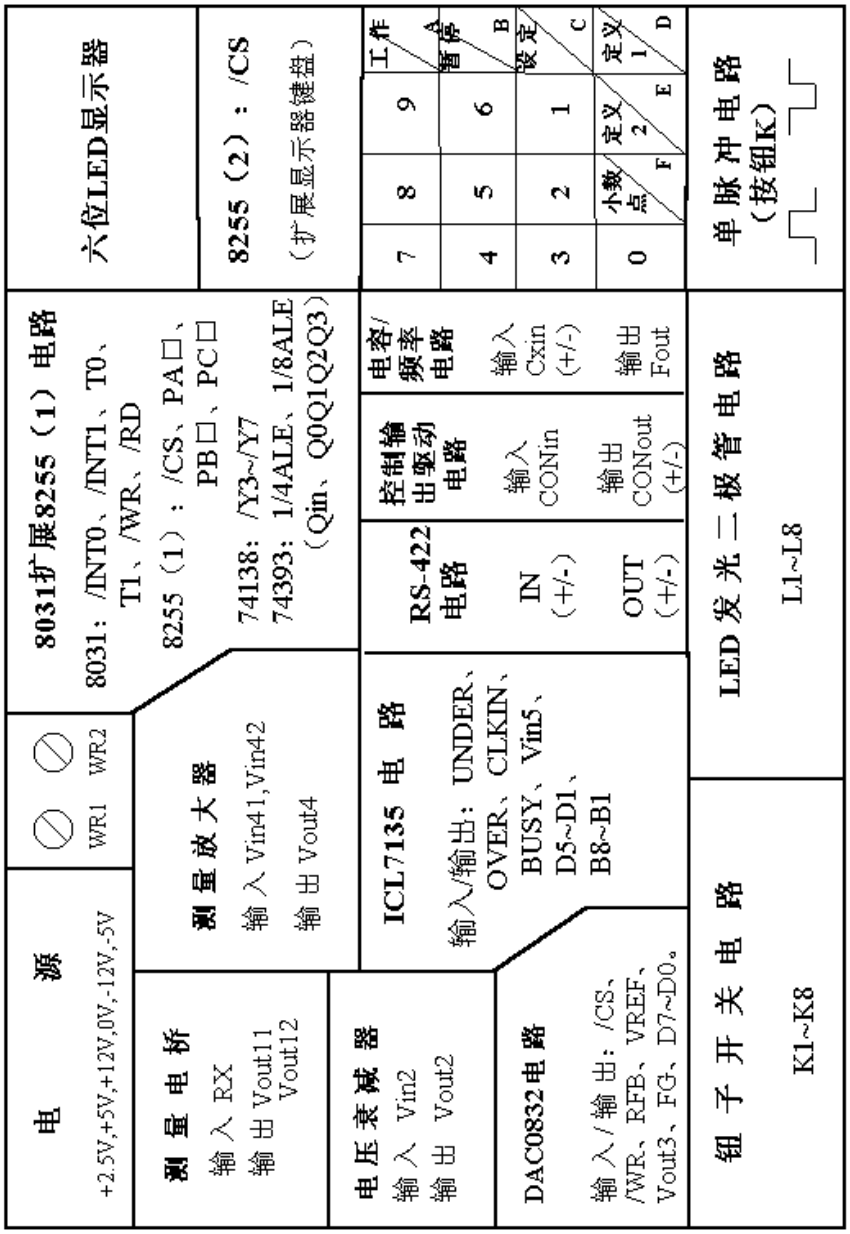


图 2　面板功能块分布图

二）功能块分类

1. 电源部分

外接+5V，+12V，-12V 电源。

实验板上有以下电源插孔：+2.5V, +5V, +12V, -5V, -12V 及 0V（GND）。

实验板上每一功能块电源独立控制，由短路块连通。

2. 测量电桥

电源+2.5V。

Rx 为热敏电阻 Pt100(调试时接电阻箱)。

输出 Vout11、Vout12。

WR3 为调零电位器。

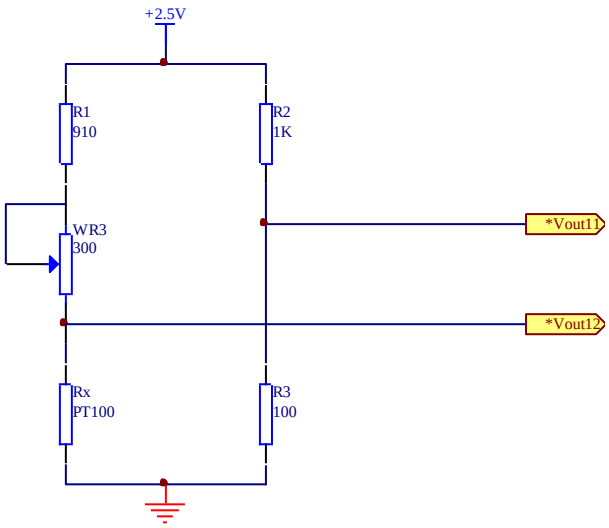


图 3　测量电桥

3. 测量放大器

电源+12V，-12V。

输入 Vin41、Vin42，输出 Vout4。

WR4 为增益调整电位器，WR5 为调零电位器。

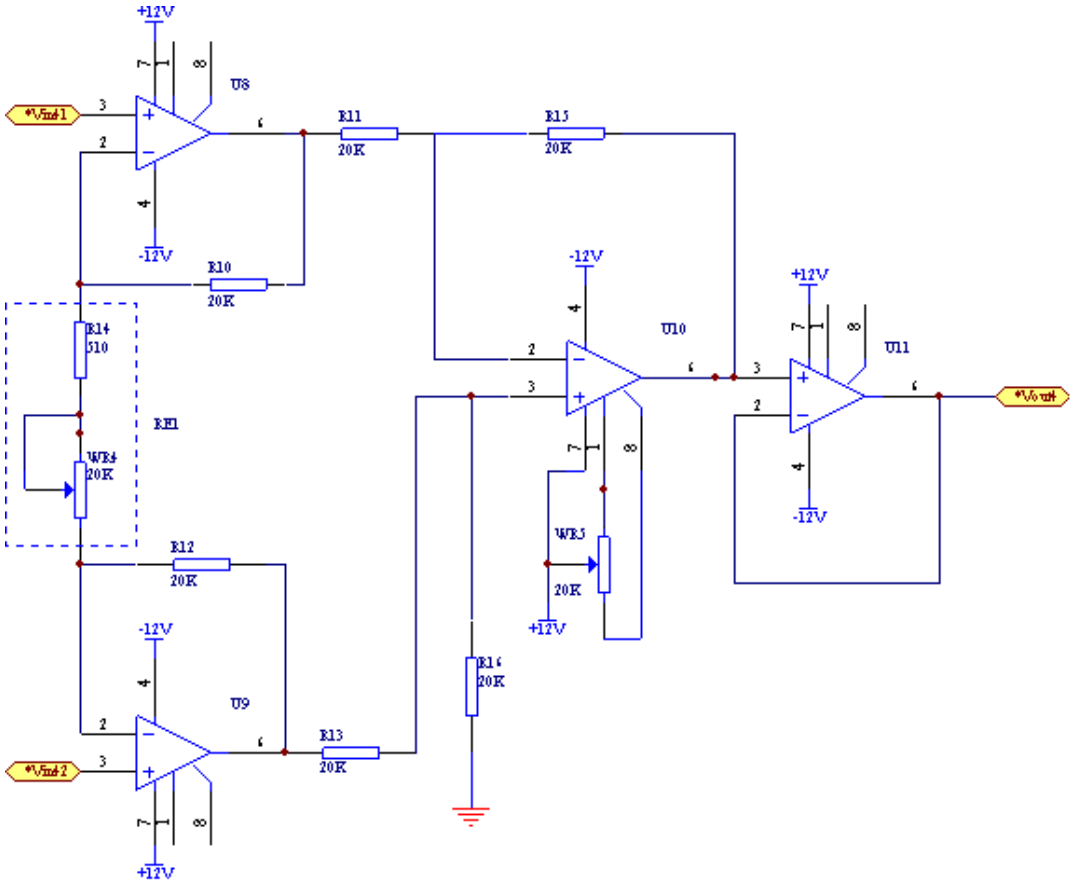


图 4　测量放大器

4.电压衰减器

电源+12V，-12V。

输入 Vin2，输出 Vout2。

衰减倍数：　1/6　（J01 通）

1/3　（J02 通）

1/2　（J03 通）

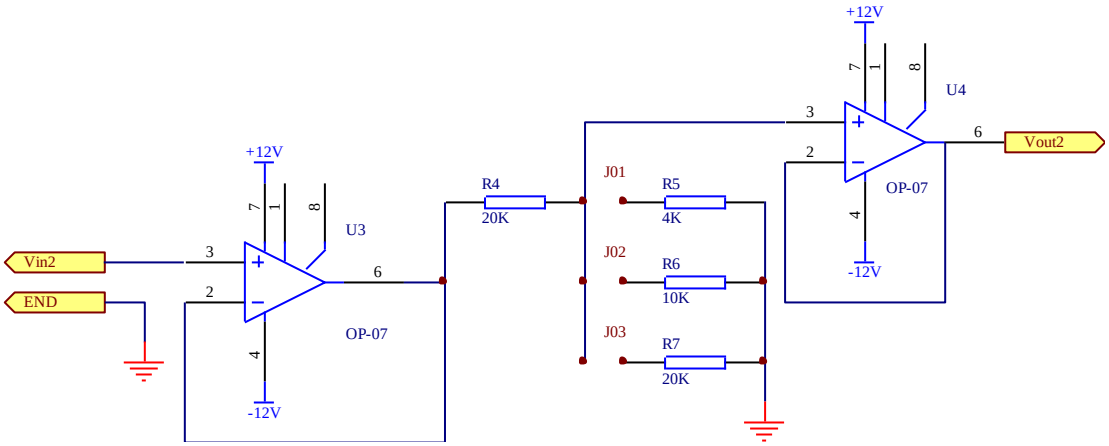


图 5　电压衰减器

5. 四位半 A/D 转换电路(ICL7135)

电源+2.5V, 5V, -5V。

输入 Vin5。使用单端输入 INH1，INH0（负端输入）已接地。输入范围为 0-1.999V，对应的输出数据为 00000-19999。

转换状态信号 BUSY（A/D 转换时为高电平，转换结束时为低电平）。可供查询或中断申请用。

时钟 CLKIN 接 1/8 ALE，单片机晶振频率为 12MHZ，则 1/8 ALE 为 250KHZ，转换频率约为 6 次/秒。转换控制端 R/H 已接高电平，表示 ICL7135 将连续自动转换。

当输入信号超过 1.999V 时，过量程标志 OVER 输出高电平。

当输入信号小于量程的 9%（0.1800V）时，欠量程标志 UNDER 输出高电平。但在单端输入时，输入信号在 0V-0.1800V 时并不影响 A/D 转换器的正常工作。

位状态输出 D5、D4、D3、D2、D1，分别表示现时输出的数据为万、千、百、十、个位。

BCD 码数据输出为 B8、B4、B2、B1。

WR6 调整 ICL7135 第 2 脚（标准电压端 VREF）电压为 1.000V。

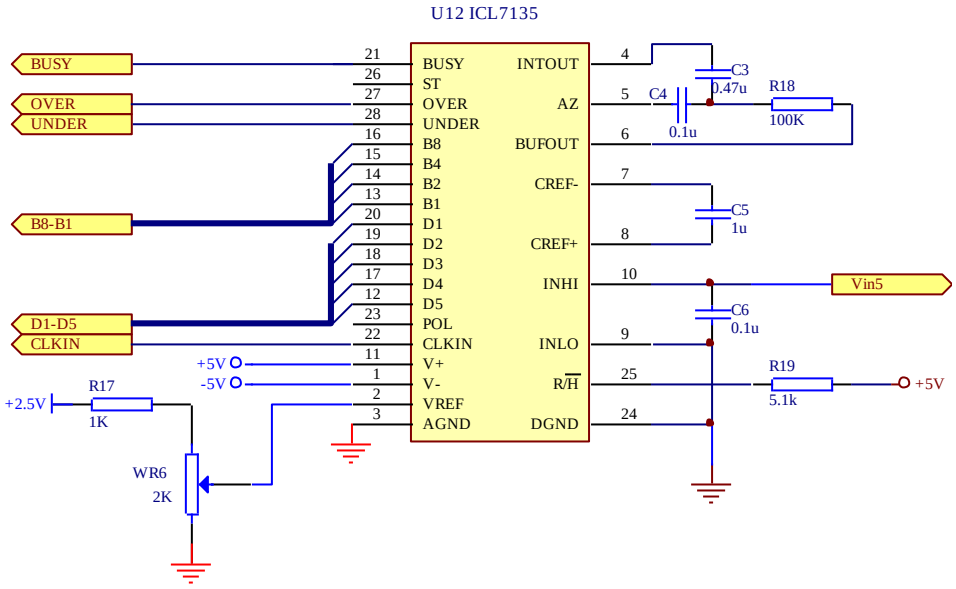


图 6　A/D 转换电路(ICL7135)

6. D/A 转换电路（DAC0832）

电源+5V，+12V，-12V。

输入 D7~D0(00H-FFH)，输出 Vout3(0-5V)。

写信号/WR，片选信号/CS。

标准电压端 VREF，反馈电阻端 RFB，反馈端 FG。

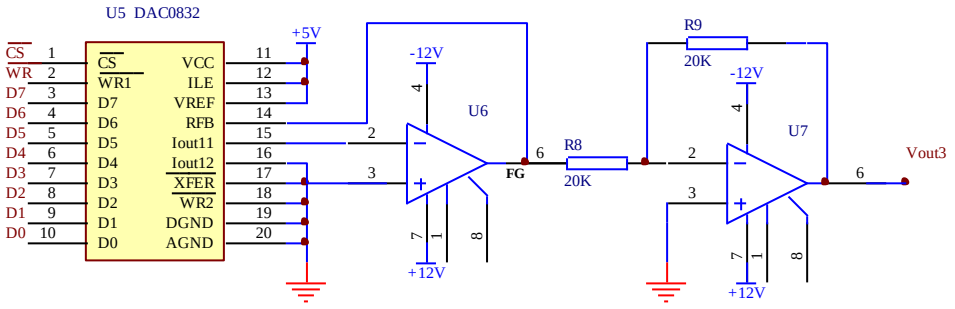


图 7　D/A 转换电路（DAC0832）

7. 钮子开关电路

电源+5V。输出为 K1~K8。

钮子开关拨向上方输出为低“0”，拨向下方输出为高“1”。

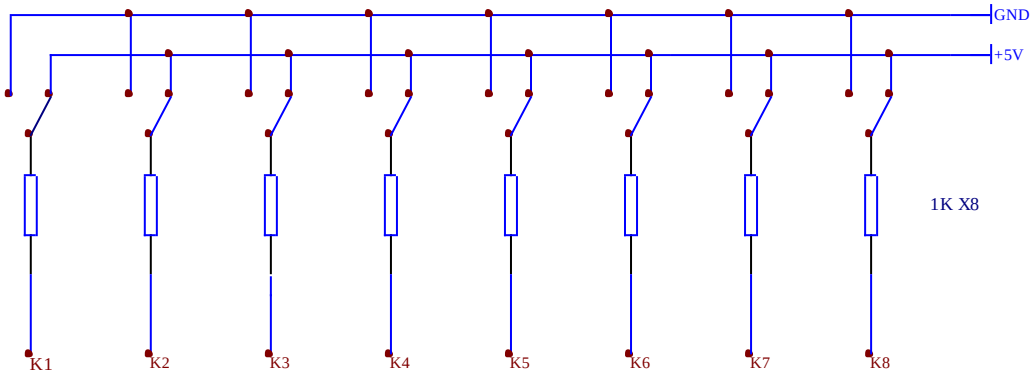


图 8　钮子开关电路

8. LED 发光二极管电路

电源+5V。输入为 L1~L8。

输入为高时发光二极管亮，输入为低时发光二极管灭（7406 为反向驱动器）。

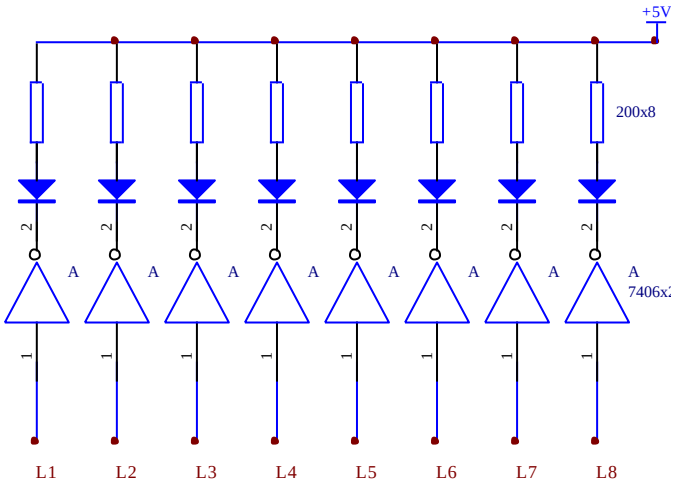


图 9　LED 发光二极管电路

9. 单脉冲输出电路

电源+5V。K 为按钮。

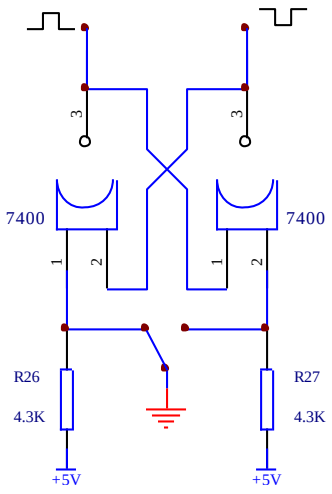


图 10　单脉冲输出电路

10. 8031 扩展 8255（1）

电源+5V。

8031 引出端 /INT0、/INT1、T0、T1、/WR、/RD。

8255（1）片选信号为 /CS，PA 口为 PA7~PA0，PB 口为 PB7~PB0，PC 口为 PC7~PC0。

译码电路 74138 输出 /Y3 ~ /Y7，地址范围为 6000H~FFFFH。

双分频电路 74393 一路输出为 1/4 ALE、1/8 ALE（输入已接 ALE）；另一路输入为 Qin，输出为 Q0~Q3（2 分频~16 分频）。

原理图见下页（图 11）。

11. 利用 8255（2）扩展 LED 显示器及行列式键盘

电源+5V。

8255（2）片选信号为 /CS。

LED 显示器为共阴极，MC1413 为反向驱动器，三极管 9013 为正向驱动。

原理图见下页（图 12）。

12. RS-422 串行口

电源+5V。 输入为 IN（+/-），输出为 OUT(+/-)。

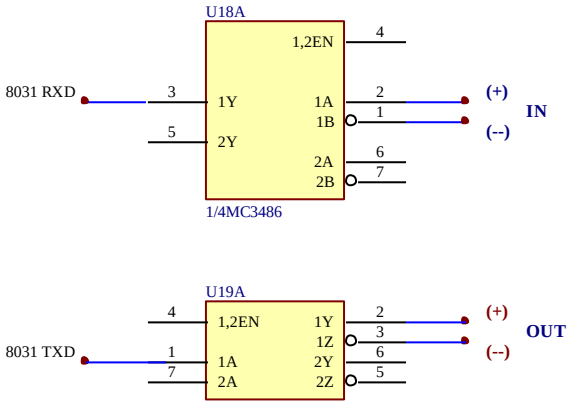


图 13　RS-422 串行口

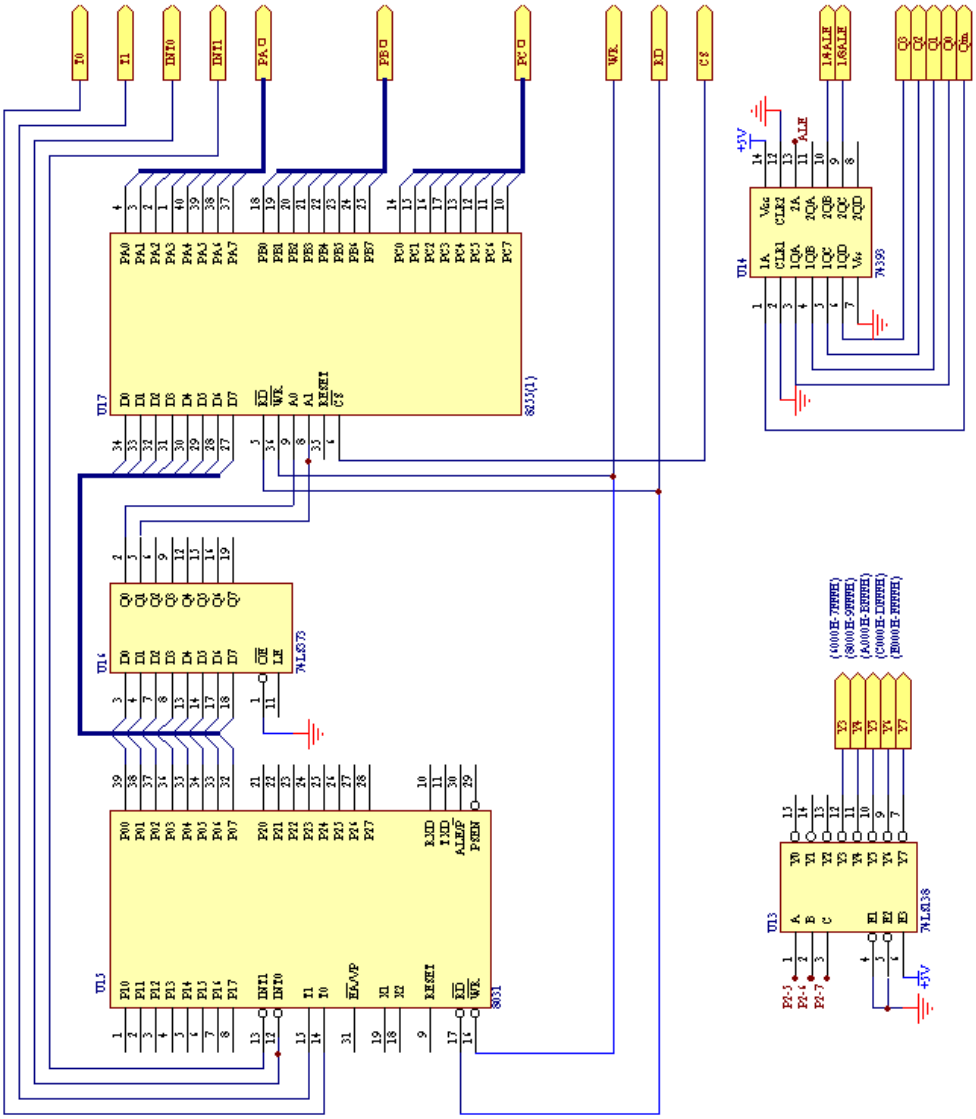


图 11　8031 扩展 8255（1）原理图

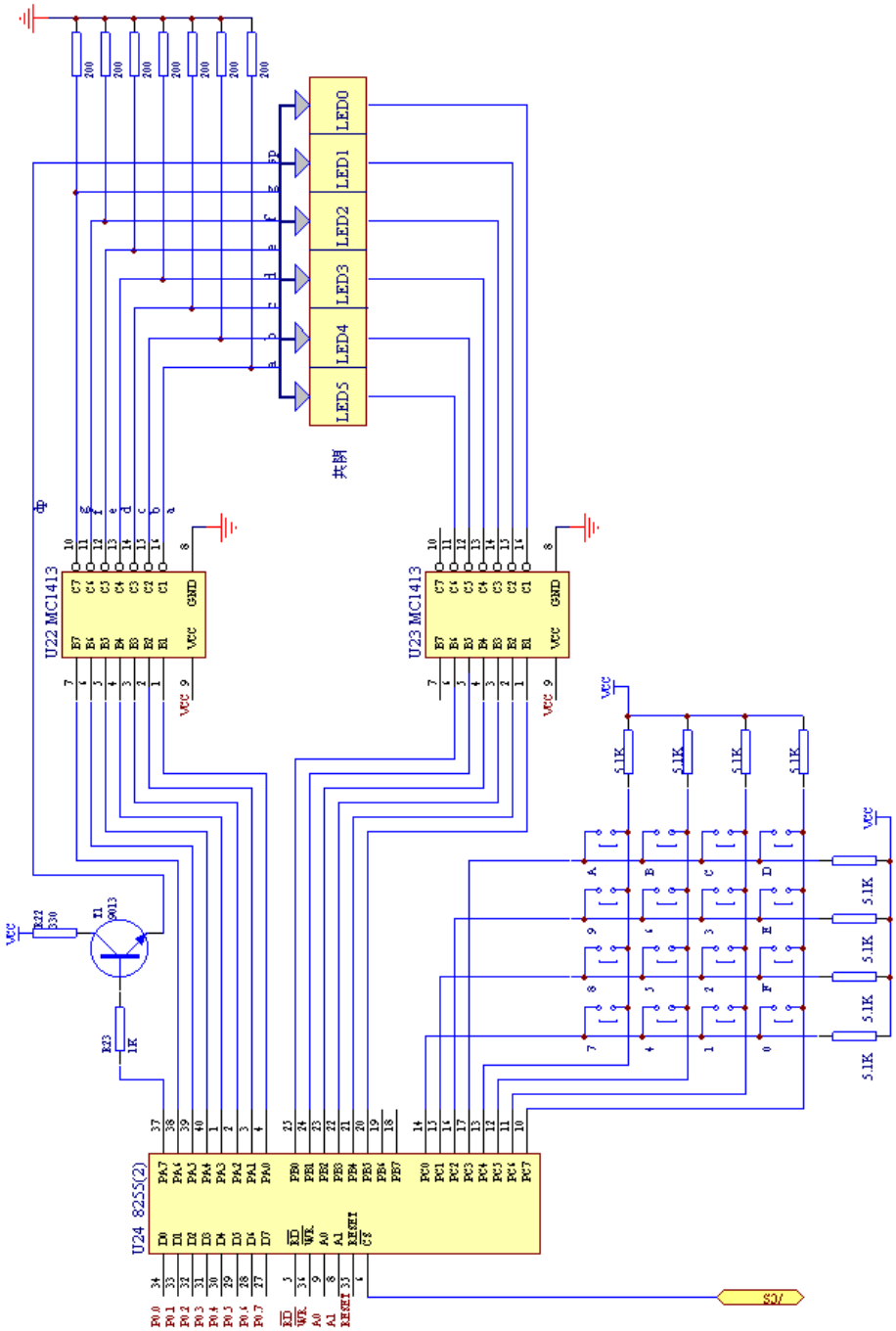


图 12　8255（2）扩展 LED 显示器及行列式键盘原理图

13. 固态继电器（负载）控制电路

电源+5V。

输入为 CONin，输出为 CONout(+/-)。

控制信号 CONin 为高时，负载接通电源。

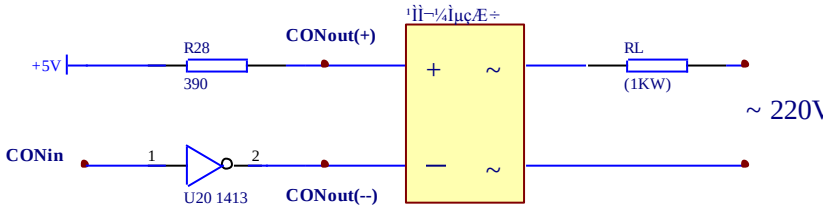


图 14　固态继电器控制电路

14. 555（电容/脉冲）电路

电源+5V。

Cxin（+/-）为电容，输出为 Fout。

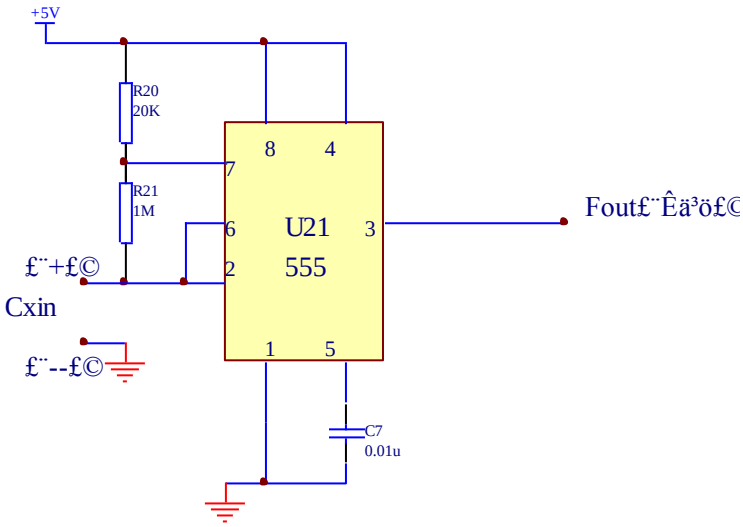


图 15　555 脉冲电路

第二章　显示器及键盘实验

实验一　显示器及键盘实验

一．目的：

1、 了解行列式键盘及 LED 显示器的扩展方法。
2、 熟悉行列式键盘输入程序及 LED 显示器显示程序的设计方法。

二．内容：

从键盘上输入 0~9 及 A~F，在 LED 显示器上显示。

三．硬件连接框图（图 16）：

键盘显示器原理图见第一章图 12。

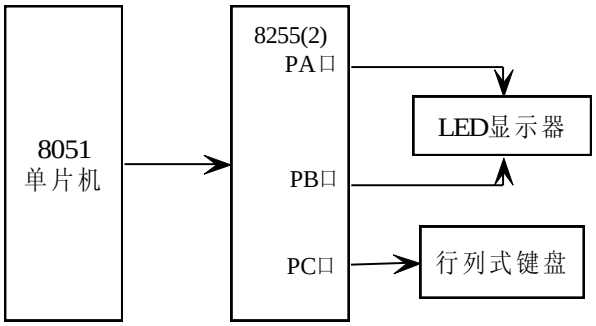


图　16

四．实验步骤：

1、 硬件连接：
8255（2）片选信号 /CS 连接 /Y3~/Y7（任选其一）；
将相关功能块的电源接通。
2、 软件设计：
1） 8255（2）PA 口 PA7 控制小数点，PA6~PA0 控制字形（经反向驱动）。
8255（2）PB 口 PB5~PB0 控制字位（经反向驱动）。
LED 显示器为共阴极。
8255（2）PC 口 PC7~PC4 为行列式键盘的行线，PC3~PC0 为行列式键盘的列线。行线、列线均由上拉电阻接+5V。
2） 行线输出全零时由列线输入低 4 位；列线输出全零时由行线输入高 4 位。
3） 键值表（0~9 及 A~F 键按下时，PC7~PC4 、PC3~PC0 输入合并后的数值）：

```
DB  7EH, 0BEH, 0BDH, 0BBH
DB  0DEH, 0DDH, 0DBH, 0EEH
DB  0EDH, 0EBH, 0E7H, 0D7H
DB  0B7H, 77H, 7BH, 7DH
```

4） 字形码：

```
DB   40H,79H,24H,30H
DB   19H,12H,02H,78H
DB   00H,10H,08H,03H
DB   46H,21H,0EH,41H
```

5） 2ms 软件延时程序：

```
DELAY2MS:  MOV R3, #04H
   DELAY0:  MOV R4, #0FFH
   DELAY1:  DJNZ R4, DELAY1
            DJNZ R3, DELAY0
            RET
```

五．参考程序清单（见下页附录一）：

六．高级要求：

1、显示小数点。在左边第 3 位上增加小数点的显示，该位字形码如何确定？修改程序，观察结果。

2、查询资料，举出其它形式的键盘显示器设计方法。分析优缺点。

附录一（参考程序清单）：

8255（2）片选信号 /CS 连接 /Y3 (6000H)

```
        ORG 0000H
        AJMP START
        ORG 0100H
START:  MOV A,#06H        ;六位显示缓冲区（75H-7AH）清零
        MOV R0,#75H       ;
CLE:    MOV @R0,#00H;
        INC R0;
        DEC A;
        JNZ CLE;

KEY:    ACALL LED          ;调显示子程序
        MOV DPTR,#6003H    ;扫描键盘。8255（2）初始化
        MOV A,#88H          ;PA 口 PB 口输出（LED 显示器）
        MOVX @DPTR,A      ;PC 口高四位输入低四位输出（行列式键盘）
        MOV DPTR,#6002H    ;PC 口低四位输出零
        MOV A,#00H          ;
        MOVX @DPTR,A      ;
        MOVX A,@DPTR      ;PC 口高四位输入
```

```
        ANL A,#0F0H          ;
        MOV 74H,A           ;键值高四位送 74H
        CJNE A,#0F0H,READ ;键值高四位非全“1”有键按下
        AJMP KEY            ;键值高四位全“1”无键按下

READ:   MOV DPTR,#6003H    ;有键按下。8255（2）初始化（行列输入输出
        MOV A,#81H           ;  交换），PA 口 PB 口输出（LED 显示器），
        MOVX @DPTR,A      ;PC 口高四位输出低四位输入（行列式键盘）
        MOV DPTR,#6002H    ;PC 口高四位输出零
        MOV A,#00H           ;
        MOVX @DPTR,A      ;
        MOVX A,@DPTR      ;PC 口低四位输入
        ANL A,#0FH           ;
        ADD A,74H            ;键值低四位送 74H
        MOV 74H,A           ;
        ACALL DELAY12MS   ;软件延时
KEY1:   ACALL LED
        MOV DPTR,#6003H   ;再扫描一次
        MOV A,#88H          ;
        MOVX @DPTR,A       ;
        MOV DPTR,#6002H     ;
        MOV A,#00H           ;
        MOVX @DPTR,A      ;
        MOVX A,@DPTR      ;
        ANL A,#0F0H          ;
        MOV 73H,A           ;
        MOV DPTR,#6003H   ;
        MOV A,#81H          ;
        MOVX @DPTR,A     ;
        MOV DPTR,#6002H   ;
        MOV A,#00H          ;
        MOVX @DPTR,A      ;
        MOVX A,@DPTR      ;
        ANL A,#0FH           ;
        ADD A,73H            ;

        CLR C               ;两次值相同去再扫描一次（按键未松）
        SUBB A,74H          ;
        JZ KEY1              ;

        MOV R1,#00H       ; 按键已松。R1 作计数器
        MOV R5,#10H       ; 16 个键
SEARCH: MOV DPTR,#KTABLE   ;键值表首址
        MOV A,R1               ;
```

```
        MOVC A,@A+DPTR     ;查键表值
        CLR C                ;
        SUBB A,74H           ;比较读入键值与键值表数据
        JZ FIND              ;相同则键值有效，去处理显示缓冲区
        INC R1               ;不相同比较下一个键值表数据
        DJNZ R5,SEARCH     ;
        AJMP KEY           ;16 个键值表数据比较完

FIND:   MOV A,R1           ;键值有效准备显示
        MOV R5,A           ;R1 数值与键盘标称值（0-F）相同
        MOV R7,#05H        ;
        MOV R0,#79H        ;显示缓冲区（7AH-75H）数据向前移位
        MOV R1,#7AH        ;
ROL:    MOV A,@R0          ;
        MOV @R1,A          ;
        DEC R0             ;
        DEC R1             ;
        DJNZ R7,ROL        ;
        MOV 75H,R5         ;新读键盘标称值（0-F）送显示缓冲区 75H
        AJMP KEY           ;

LED:    MOV DPTR,#6003H    ;显示子程序
        MOV A,#80h         ;8255（2）初始化
        MOVX @DPTR,A       ;
SHOW:   MOV R0,#75h        ;显示缓冲区首地址
        MOV R2,#01h        ;位码
        MOV A,R2           ;
LOOP:   MOV DPTR,#6001H    ;位码送 PB 口
        MOVX @DPTR,A       ;
        MOV DPTR,#6000H    ;PA 口送全黑字形码
        MOV A,#7FH         ;
        MOVX @DPTR,A       ;

        MOV DPTR,#LTABLE   ;字形表首地址
        MOV A,@R0          ;取显示缓冲区值
        MOVC A,@A+DPTR     ;取字形码
        MOV DPTR,#6000H    ;字形码送 PA 口
        MOVX @DPTR,A       ;
        ACALL DELAY1MS     ;软件延时
        INC R0             ;指向下一显示缓冲区地址
        MOV A,R2           ;
        JB ACC.5,LOOP1     ;第六位显示完？
        RL A               ;未完显示下一位
        MOV R2,A           ;
```

```
        AJMP LOOP          ;
 LOOP1: RET                 ;第六位显示完返回

 DELAY12MS: MOV R3,#0FFH   ;
 DLAY0:     MOV R4,#0FFH
 DLAY1:     DJNZ R4,DLAY1
            DJNZ R3,DLAY0
            RET

DELAY1MS: MOV R3,#02H   ;
DELAY0:   MOV R4,#0FFH
DELAY1:   DJNZ R4,DELAY1
          DJNZ R3,DELAY0
          RET

LTABLE: DB 40H,79H,24H,30H,19H     ;字形表
        DB 12H,02H,78H,00H,10H
        DB 08H,03H,46H,21H,06H
        DB 0EH

KTABLE:   DB 7EH,0BEH,0BDH,0BBH   ;键值表
          DB 0DEH,0DDH,0DBH,0EEH
          DB 0EDH,0EBH,0E7H,0D7H
          DB 0B7H,77H,7BH,7DH

          END
```

第三章　程控放大器及程控信号发生器实验

实验二　程控放大器实验

一．　目的：

了解程控放大器的结构，学会程控放大器的设计和使用。

二．　内容：

利用 DAC0832 设计程控放大器，测量出程控数字量与放大倍数的关系。

三．　硬件连接框图（图 17）：

DAC0832 电路图见第一章图 7。

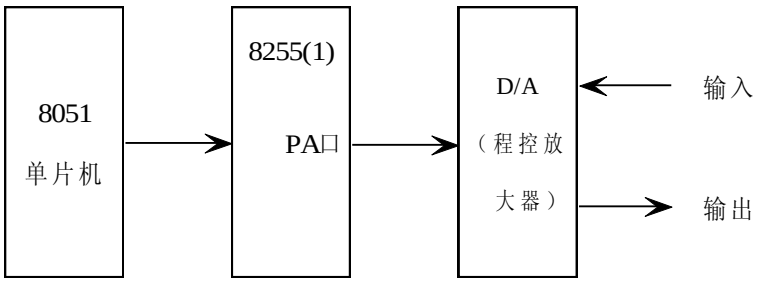


图　17

四．　实验步骤：

1、　硬件连接：

1）　+2.5V 电源接入电位器 WR2 一端，另一端接 0V（GND）。从中间抽头输出 0.1V 电压，将此电压接入 DAC0832 电路的反馈电阻端 RFB（程控放大器输入端）。

2）　DAC0832 电路的片选信号端 /CS 接 /Y3~/Y7（任选其一），写控制端 /WR 接 8031 的 /WR，D7~D0 接 8255（1）PA 口的 PA7~PA0，反馈端 FG 接标准电压端 VREF。

3）　8255（1）片选信号 /CS 接 /Y3~/Y7（任选其一）。

4）　连接相关功能块电源。

2、　软件设计：

1）　8255（1）初始化，PA 口输出。

2）　将数字量（00H~FFH）送入 8255（1）PA 口。再将此数字量送入 DAC0832。

五．　执行程序：用万用表测量 DAC0832 电路的输出端 Vout3 的电压。修改程序中输出的数字量，重复以上过程。找出放大器的饱和区。

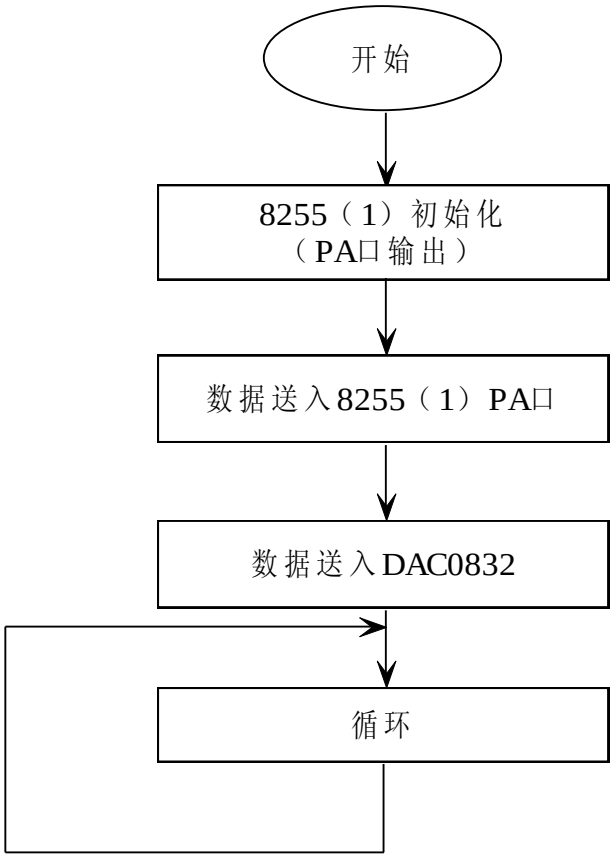
六． 填写下列表格： （输入电压 0.1V）

数字量	输 出 电 压	放 大 倍 数

七． 高级要求：

1、 将 -5V 电源接入电位器 WR2 一端，另一端接 0V（GND）。从中间抽头输出 -0.1V 电压，将此电压接入 DAC0832 电路的反馈电阻端 RFB（程控放大器输入端）。重复以上过程。
2、 分析程控放大器的工作原理，画出程控放大器的原理图。
3、 查询资料，了解专用程控放大器芯片及其使用方法。

八、参考程序框图（图 18 ）：



实验三　程控信号发生器实验

一、目的：

1、了解程控信号发生器的结构。
2、学会程控信号发生器的设计和使用。

二、内容：

利用 DAC0832 的 D/A 转换功能，根据一定的时序（函数）关系，将相应的数字量送入 DAC0832。

使 DAC0832 电路的输出端输出相应的波形。

三、 硬件连接框图（图 19）：

DAC0832 电路图见第一章图 7。

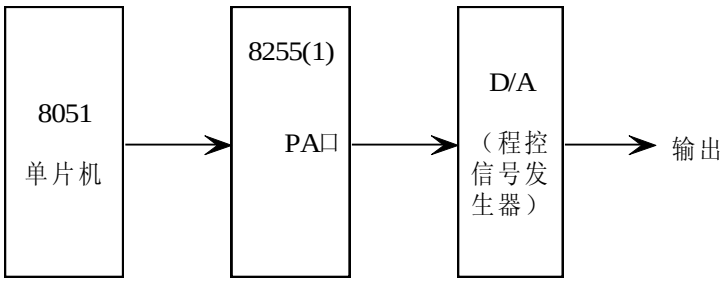


图　19

四、 实验步骤：

1、 硬件连接：
1） DAC0832 电路的标准电压端 VREF 接+5V，片选信号端 /CS 接 /Y3~/Y7（任选其一），写控制端 /WR 接 8031 的 /WR，D7~D0 接 8255（1）PA 口的 PA7~PA0，反馈端 FG 接反馈电阻输入端 RFB。
2） 8255（1）片选信号 /CS 接 /Y3~/Y7（任选其一）。
3） 连接相关功能块电源。
2、 软件设计：
1） 8255（1）初始化，PA 口输出。
2） 根据一定的时序（函数）关系，将相应的数字量送入 8255（1）PA 口。再将此数字量送入 DAC0832。

五、 执行程序：用示波器测量 DAC0832 电路的输出端 Vout3 输出的波形。

六、 高级要求：

1、 设计其它波形（如梯形波）的程序。
2、 怎样在程序中处理输出波形的频率？
3、 查询资料，了解专用波形发生器芯片及其使用方法。

七、 参考程序框图（见下页图 20）：

八、　参考程序清单（见下页附录二）：

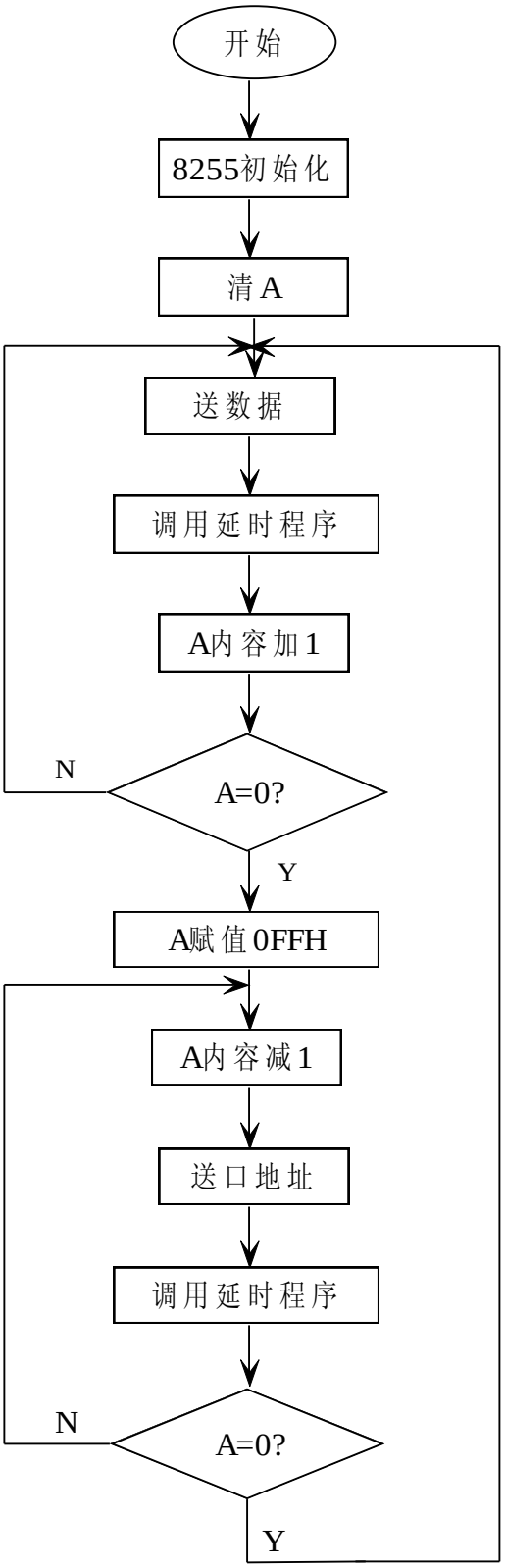


图 20-1　三角波

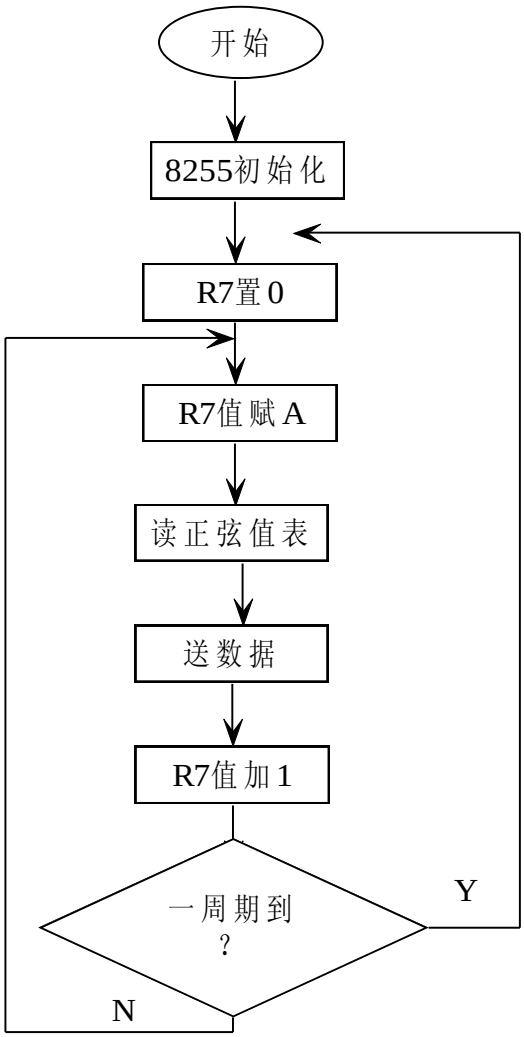


图 20-2　正弦波

附录二（参考程序清单）：

DAC0832 片选信号端 /CS 接 /Y3（6000H）
8255（1）片选信号 /CS 接 /Y4（8000H）

1.　三角波：

```
 ORG  0000H
AJMP MAIN
ORG  0100H
```

```
MAIN:  MOV   DPTR,#8003H      ;8255 初始化
       MOV   A,#80H
       MOVX  @DPTR,A
       CLR   A
  UP:  MOV   DPTR,#8000H      ;线性上升段。数据送 8255
       MOVX  @DPTR,A
       MOV   DPTR,#6000H      ; 数据送 DAC0832
       MOVX   @DPTR,A
       ACALL   DING1
       INC   A
       JNZ   UP

       MOV   A,#0FFH
 DOWN: DEC A                    ; 线性下降段。数据送 8255
       MOV   DPTR,#8000H      ;
       MOVX  @DPTR,A
       MOV   DPTR,#6000H
       MOVX   @DPTR,A         ; 数据送 DAC0832
       ACALL   DING1
       JNZ   DOWN
       AJMP   UP

 DING1:    MOV R3,#01H
 DELAY0:   MOV R4,#80H
 DELAY1:   DJNZ R4,DELAY1
           DJNZ R3,DELAY0
           RET
```

2.　正弦波：

```
          ORG   0000H
          AJMP MAIN

          ORG   0100H
  MAIN:   MOV   DPTR,#8003H     ;8255 初始化
          MOV   A,#80H
          MOVX  @DPTR,A

  SIN0:   MOV   R7,#00H           ;置偏移量
  SIN1:   MOV   A,R7
          MOV   DPTR,#TABLE     ;从表中取值
          MOVC   A,@A+DPTR
          MOV    DPTR,#8000H     ; 数据送 8255
          MOVX   @DPTR,A
```

```
        MOV    DPTR,#6000H      ；数据送 DAC0832
        MOVX @DPTR,A
        INC     R7                ；指向下一个数
        CJNE   R7,#0B5H,SIN1     ；一周期 180 个数（B5H=181D）
        AJMP   SIN0

TABLE:  DB 7FH,83H,87H,8CH,90H,95H,99H,9DH,0A2H;（每 2 度一个数）
        DB 0A6H,0AAH,0AEH,0B2H,0B6H,0BAH,0BEH,0C2H
        DB 0C6H,0C9H,0CDH,0D0H,0D3H,0D7H,0DAH,0DDH
        DB 0E0H,0E3H,0E5H,0E8H,0EAH,0EDH,0EFH,0F1H
        DB 0F3H,0F4H,0F6H,0F7H,0F9H,0FAH,0FBH,0FCH
        DB 0FCH,0FDH,0FDH,0FDH,0FEH,0FDH,0FDH,0FDH
        DB 0FCH,0FCH,0FBH,0FAH,0F9H,0F7H,0F6H,0F4H
        DB 0F3H,0F1H,0EFH,0EDH,0EAH,0E8H,0E5H,0E3H
        DB 0E0H,0DDH,0DAH,0D7H,0D4H,0D0H,0CDH,0C9H
        DB 0C6H,0C2H,0BEH,0BAH,0B6H,0B2H,0AEH,0AAH
        DB 0A6H,0A2H,9DH,99H,95H,90H,8CH,88H,83H
        DB 7FH,7AH,76H,71H,6DH,69H,64H,60H,5CH,57H
        DB 53H,4FH,4BH,47H,43H,3FH,3BH,38H,34H,31H
        DB 2DH,2AH,26H,23H,20H,1DH,1BH,18H,15H,13H
        DB 11H,0EH,0CH,0BH,09H,07H,06H,04H,03H,02H,01H
        DB 01H,00H,00H,00H,00H,00H,00H,00H,01H,01H
        DB 02H,03H,04H,06H,07H,09H,0AH,0CH,0EH,10H
        DB 13H,15H,18H,1AH,1DH,20H,23H,26H,29H,2DH
        DB 30H,34H,37H,3BH,3FH,43H,46H,4BH,4FH,53H
        DB 57H,5BH,5FH,64H,68H,6CH,71H,75H,7AH,7FH
```

第四章　键盘键值远距离传输实验

实验四　键盘键值远距离传输实验

一．目的：

利用串行口实现数值的远距离传输。

二．内容：

1、将 A 单片机的键值通过串行口传输到 B 单片机中，并在 B 单片机的 LED 显示器上显示。
2、将 B 单片机的键值通过串行口传输到 A 单片机中，并在 A 单片机的 LED 显示器上显示。

三．硬件连接框图（图 21）：

相关电路原理图见第一章图 12、图 13。

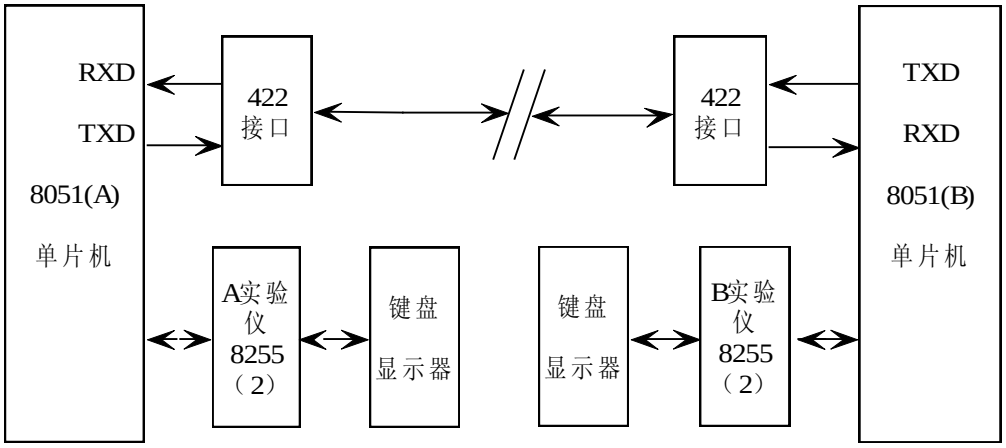


图　21

四．实验步骤：

1、 硬件连接：
1） A 单片机实验仪 RS-422 接口输入端 IN（+，-）连接 B 单片机实验仪 RS-422 接口输出端 OUT（+，-）。
2） B 单片机实验仪 RS-422 接口输入端 IN（+，-）连接 A 单片机实验仪 RS-422 接口输出端 OUT（+，-）。
3） 8255（2）片选信号 /CS 接 /Y3~/Y7（任选其一）。
4） 连接相关功能块电源。

2、 软件设计：

1） A、B 单片机串行口工作方式为方式 1。定时器 T1 做波特率发生器，波特率为 1200HZ（晶振频率为 12M）。

2）8255（2）初始化（扩展显示器及键盘）；

调用显示子程序；

查询串行口接收标志（查询方式），接收到数据即修改显示缓冲区；

调用扫描键盘子程序，有键盘输入即将键值从串行口发送。

3）8255（2）PA 口 PA7 控制小数点，PA6~PA0 控制字形（经反向驱动）。

8255（2）PB 口 PB5~PB0 控制字位（经反向驱动）。

LED 显示器为共阴极。

4）8255（2）PC 口 PC7~PC4 为行列式键盘的行线，PC3~PC0 为行列式键盘的列线。行线、列线均由上拉电阻接+5V。

行线输出全零时由列线输入低 4 位；列线输出全零时由行线输入高 4 位。

五．参考程序清单（见下页附录三）：

六．高级要求：查询资料，了解其它标准的串行口与 8031 单片机的连接方式，了解其使用方法，了解相关常用芯片。

附录三（参考程序清单）：

8255（2）片选信号 /CS 接 /Y3（6000H）

```
        ORG 0000H
        AJMP START
        ORG 0100H
START:  MOV TMOD,#20H       ;T1 定时工作方式 2
        MOV TH1,#0CCH        ;T1 定时常数(波特率)
        MOV TL1,#0CCH
        MOV SCON,#50H       ;串行口方式 1,允许接收
        MOV PCON,#80H       ;波特率设置(SMOD=1)
        MOV SP,#30H          ;设置堆栈指针
        SETB TR1             ;启动定时器 1
        MOV A,#06H           ;设置显示缓冲区初值
        MOV R0,#75H
STA:    MOV @R0,#10H
        INC R0
        DEC A
        JNZ STA
        LCALL LED           ;调显示子程序
WINA:   JBC RI,CHANGE       ;等待接收
        LCALL KEY           ;调键盘扫描子程序
        LCALL LED
        SJMP WINA
```

```
CHANGE: LCALL LED           ;输入串口数据
        CLR RS1
        SETB RS0
        MOV A,SBUF
        MOV R5,A
        MOV R7,#05H          ;将显示缓冲区值右移一位
        MOV R0,#79H
        MOV R1,#7AH
ROT:    MOV A,@R0
        MOV @R1,A
        DEC R0
        DEC R1
        DJNZ R7,ROT
        MOV 75H,R5              ;将接收值存入 75H
        AJMP WINA
        ORG 1000H
 KEY:   LCALL LED          ;键盘扫描子程序
        SETB RS1
        CLR RS0
        MOV DPTR,#6003H   ;8255（2）控制字(PC7-PC4 输入,PC3-PC0 输出)
        MOV A,#88H
        MOVX @DPTR,A
        MOV DPTR,#6002H   ;PC3-PC0 输出零,读入 PC7-PC4 值
        MOV A,#00H
        MOVX @DPTR,A
        MOVX A,@DPTR
        ANL A,#0F0H
        MOV 74H,A             ;将键值的高四位存入 74H 单元里
        CJNE A,#0F0H,READ     ;无键按下时返回
        RET
READ:   LCALL LED
        SETB RS1
        CLR RS0
        MOV DPTR,#6003H ;8255（2）控制字（PC7-PC4 输出,PC3-PC0 输入）
        MOV A,#81H
        MOVX @DPTR,A
        MOV DPTR,#6002H    ;PC7-PC4 输出零,读入 PC3-PC0 值
        MOV A,#00H
        MOVX @DPTR,A
        MOVX A,@DPTR
        ANL A,#0FH
        ADD A,74H
        MOV 74H,A             ;将键值存入 74H 单元里
        ACALL DELAY12MS   ;调用延时程序
```

```
KEY1:   LCALL LED           ;再扫描
        SETB RS1
        CLR RS0
        MOV DPTR,#6003H
        MOV A,#88H
        MOVX @DPTR,A
        MOV DPTR,#6002H
        MOV A,#00H
        MOVX @DPTR,A
        MOVX A,@DPTR
        ANL A,#0F0H
        MOV 73H,A
        LCALL LED
        SETB RS1
        CLR RS0
        MOV DPTR,#6003H
        MOV A,#81H
        MOVX @DPTR,A
        MOV DPTR,#6002H
        MOV A,#00H
        MOVX @DPTR,A
        MOVX A,@DPTR
        ANL A,#0FH
        ADD A,73H
        CLR C               ;如两次值相同，去再扫描
        SUBB A,74H
        JZ KEY1
        LCALL LED
        SETB RS1
        CLR RS0
        MOV R1,#00H           ;R1 作计数器
        MOV R5,#10H           ;十六个键
SEARCH: MOV DPTR,#TABLE1    ;取键值表的首地址
        MOV A,R1
        MOVC A,@A+DPTR      ;取键值表数据
        CLR C
        SUBB A,74H          ;比较
        JZ SOUT             ;相同转发送程序
        INC R1
        DJNZ R5,SEARCH
        RET                  ;不相同无效
SOUT:   MOV A,R1            ;发送键值（0-F）
        MOV SBUF,A
  TR:   LCALL LED
```

```
        JBC TI,TR
        RET
LED:    SETB RS1                ;显示子程序
        SETB RS0
        MOV DPTR,#6003H
        MOV A,#80H              ;8255（2）PA 口输出 PB 口输出
        MOVX @DPTR,A
SHOW:   MOV R0,#75h
        MOV R2,#01h
        MOV A,R2
LOOP:   MOV DPTR,#6001H
        MOVX @DPTR,A            ;送位选码到 PB 口
        MOV DPTR,#TABLE2        ;取显示表的首地址
        MOV A,@R0
        MOVC A,@A+DPTR          ;取字形码
        MOV DPTR,#6000H
        MOVX @DPTR,A             ;送段码到 PA 口
        ACALL DELAY1MS           ;调用延时程序
        INC R0
        MOV A,R2
        JB ACC.5,LOOP1          ;是否扫描到第 6 位
        RL A
        MOV R2,A
        AJMP LOOP
LOOP1:  RET
DELAY12MS: MOV R3,#0FFH                ;延时 12 毫秒
DLAY0:     MOV R4,#0FFH
DLAY1:     DJNZ R4,DLAY1
           DJNZ R3,DLAY0
           RET
DELAY1MS:  MOV R3,#04H                  ;延时 1 毫秒
DELAY0:    MOV R4,#0FFH
DELAY1:    DJNZ R4,DELAY1
           DJNZ R3,DELAY0
           RET
TABLE1: DB 7EH,0BEH,0BDH,0BBH        ;键值表
        DB 0DEH,0DDH,0DBH,0EEH
        DB 0EDH,0EBH,0E7H,0D7H
        DB 0B7H,77H,7BH,7DH,91H
TABLE2: DB 0C0H,0F9H,0A4H,0B0H       ;字型表
        DB 99H,92H,82H,0F8H
        DB 80H,90H,88H,83H
        DB 0C6H,0A1H,86H,8EH,8CH
        END
```

第五章　四位半数字电压表实验

实验五　四位半数字电压表实验

一．目的：

1．掌握 ICL7135 和 MCS-51 单片机接口设计及其程序设计方法。
2．了解简易数字电压表的设计方法。

二．内容：

将直流电压从 ICL7135 的输入端输入，转换成对应的四位半数字量后，送 LED 显示器显示。显示范围：0.0000V-1.9999V。

三．硬件连接框图（图 22）：

相关电路原理图见第一章图 6、图 11 和图 12。

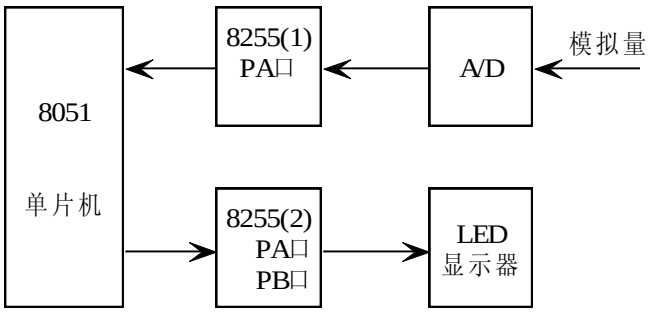


图　22

四．实验步骤：

1．硬件连接：
1）将+2.5V 电压接入电位器 WR2 的一端，0V（GND）接入另一端。中间抽头输出 0V~2.5V，此电压接入 ICL7135 输入端 Vin5。
2）8255（1）PA 口、PB 口接 ICL7135 的数据输出端 B8~B1（BCD 码输出）及 ICL7135 的位状态输出端 D5~D1（万、千、百、十、个位状态）。
3）ICL7135 的时钟输入端 CLKIN 接 1/8ALE，转换状态输出端 BUSY 接 8031 的外部中断输入端 。
4）8255（1）片选信号 /CS 接 /Y3~/Y7（任选其一）。
8255（2）片选信号 /CS 接 /Y3~/Y7（任选其一）。
5）连接相关功能块电源。

2．软件设计：
1）8255（1）初始化（PA 口、PB 口输入，连接 ICL7135）。8255（2）初始化（PA 口、PB 口输出，扩展 LED 显示器）。设置外部中断（BUSY 向 CPU 申请

中断），低电平有效。设置显示缓冲器初值。

2）主程序调用显示程序，并循环等待中断。

3）中断服务子程序分别判断万、千、百、十、个位输出状态，分别读入万、千、百、十、个位数据并保存。修改显示缓冲器内容。

4）8255（2）PA 口 PA7 控制小数点，PA6~PA0 控制字形（经反向驱动），PB 口 PB5~PB0 控制字位（经反向驱动）。LED 显示器为共阴极。

五．参考程序清单（见下页附录四）：

六．高级要求：

修改程序，设计一个量程为 0.00V-12.00V 的简易数字电压表。

利用 +12V 电源，将+12V 接入电压衰减器输入端，电压衰减器衰减倍数设为 1/6。电压衰减器输出端接入电位器 WR2 的一端，0V（GND）接入电位器 WR2 的另一端。电位器 WR2 的中间抽头输出 0V~2V，此电压接入 ICL7135 输入端 Vin5。

电压衰减器电路原理图见第一章图 5。

附录四（参考程序清单）：

8255（1）PA 口的 PA7~PA4 接 ICL7135 数据输出端 B8~B1（BCD 码输出），PA3~PA0 接 ICL7135 位状态输出端 D4~D1（千、百、十、个位输出状态）；

无千、百、十、个位输出时，即为万位输出；

ICL7135 的时钟输入端 CLKIN 接 1/8 ALE，转换状态输出端 BUSY 接 8031 的外部中断输入端 /INT0；

8255（1）片选信号 /CS 接 /Y7（E000H）；

8255（2）片选信号 /CS 接 /Y3（6000H）。

```
        ORG 0000H
        AJMP START
        ORG 0003H              ;INT0 中断矢量地址入口
        LJMP INTER
        ORG 0100H
START:  MOV R0,#50H
        MOV R2,#06H
LOOP0:  MOV @R0,#10H           ;给显示缓冲区 50H-55H 赋值
        INC R0                   ;显示全黑
        DJNZ R2,LOOP0
        MOV DPTR,#6003H
        MOV A,#80H
        MOVX @DPTR,A          ;对 8255(2)进行初始化(LED 显示器)
        MOV A,#92H               ;对 8255(1)进行初始化(接 ICL7135)
        MOV DPTR,#0E003H
```

```
        MOVX @DPTR,A
        MOV SP,#70H             ;设置堆栈区
        SETB IT0                 ;/INT0 跳变触发
        SETB EA
        SETB EX0                 ;开中断
MAIN:   ACALL LED               ;调显示子程序
        NOP                      ;等待中断
        NOP
        AJMP MAIN

 LED:   CLR EA                       ;关中断
        MOV R0,#01H                  ;位码
        MOV R1,#50H                  ;显示缓冲区首址
LOOP:   MOV A,R0                     ;送位码
        MOV DPTR,#6001H
        MOVX @DPTR,A

        MOV A,@R1                ;送段码
        MOV DPTR,#SEG
        MOVC A,@A+DPTR
        CJNE R0,#01H,LOOP1         ;左边第一位加小数点
 DOT:   ADD A,#80H
        MOV DPTR,#6000H
        MOVX @DPTR,A

        ACALL DELAY

        MOV A,#7FH            ;清黑
        MOVX @DPTR,A
        AJMP GOON
LOOP1:  MOV DPTR,#6000H
        MOVX @DPTR,A

        ACALL DELAY

        MOV A,#7FH            ;清黑
        MOVX @DPTR,A
GOON:   INC R1                ;指向下一位
        MOV A,R0
        RL A
        MOV R0,A
        CJNE R0,#40H,LOOP            ;六位显示完返回
        SETB EA                      ;开中断
        RET
```

```
 DELAY: MOV R7,#02H
DELAY0: MOV R6,#0FFH
DELAY1: DJNZ R6,DELAY1
        DJNZ R7,DELAY0
        RET

        ORG 1000H
 INTER: PUSH PSW                    ;中断子程序
        PUSH DPH                    ;现场保护
        PUSH DPL
        PUSH ACC
        CLR EA                      ;关中断

   D5:  MOV DPTR,#0E000H
        MOVX A,@DPTR            ; 等待万位输出
        MOV R3,A
        ANL A,#0FH
        JNZ D5
        MOV A,R3
        ANL A,#0F0H                ;万位 BCD 码放入 50H
        SWAP A
        MOV 50H,A

   D4:  MOV DPTR,#0E000H
        MOVX A,@DPTR
        MOV R3,A
        ANL A,#08H              ;等待千位输出
        JZ D4
        MOV A,R3
        ANL A,#0F0H
        SWAP A
        MOV 51H,A               ;千位 BCD 码放入 51H

   D3:  MOV DPTR,#0E000H        ;等待百位输出
        MOVX A,@DPTR
        MOV R3,A
        ANL A,#04H
        JZ D3
        MOV A,R3
        ANL A,#0F0H
        SWAP A
        MOV 52H,A                ;百位 BCD 码放入 52H
```

```
D2:     MOV DPTR,#0E000H        ;等待十位输出
        MOVX A,@DPTR
        MOV R3,A
        ANL A,#02H
        JZ D2
        MOV A,R3
        ANL A,#0F0H
        SWAP A
        MOV 53H,A               ;十位 BCD 码放入 53H

D1:     MOV DPTR,#0E000H        ;等待个位输出
        MOVX A,@DPTR
        MOV R3,A
        ANL A,#01H
        JZ D1
        MOV A,R3
        ANL A,#0F0H
        SWAP A
        MOV 54H,A               ;十位 BCD 码放入 54H
RETURN: POP ACC                 ;恢复现场
        POP DPL
        POP DPH
        POP PSW
        SETB EA                 ;开中断
        RETI

SEG:    DB 40H,79H,24H,30H,19H,12H,02H,78H
        DB 00H,10H,08H,03H,46H,21H,06H,0EH
        DB 7FH                  ；7FH 为全黑字形码
```

第六章　数字滤波程序设计实验

实验六. 数字滤波程序设计实验

一、实验目的：

了解数据采集中数字滤波的方法，掌握数字滤波的程序设计。

二、实验内容：

采集的数据已存放在指定的存储单元，采用去极值平均滤波法（去掉最小值及最大值，然后求平均值），求得有效值。

三、实验步骤：

1、软件设计说明：

假设已采集到六个数据（六个数据在程序中赋值给出），每个数据为两字节十六进制数（0000H-FFFFH）。

数据存放的地址为：(2BH,2AH)、(29H,28H)、(27H,26H)、(25H,24H)、(23H,22H)、(21H,20H)。高字节在前。

将六个数排序后，去掉最小值及最大值，然后求平均值。将此平均值作为有效值存放在（61H,60H）。

2、执行程序（暂停），检查结果。修改程序中给定的六个数据，重复以上过程。

四、数字滤波简介：

1、程序判断滤波

在生产实践中，很多物理量的变化有一定的规律和需要一定的时间，相邻两次采样值之间的变化也有一个限额。为此，可根据经验数据，规定出一个最大可能的变化范围，每次采样值均与上次的有效采样值进行比较。如果变化幅值不超过规定值限，则本次采样值有效，否则，本次采样值应视为干扰而弃之，以上次采样值为准。

2、中值滤波

对目标参数连续进行若干次采样，然后将这些采样进行排序，选取中间位置的采样值为有效值。本算法为取中值，采样次数应为奇数。常用 3 次或 5 次。对于变化很缓慢的参数，有时也可增加次数。

2、算术平均滤波

对于目标参数进行连续采样，然后求其算术平均值作为有效采样值。该算法适用于抑制随机干扰。采样次数 n 越大，平滑效果越好，但系统的灵敏度要下

降。为方便求平均值，n 一般取 4、8、16 类的 2 的整数幂，以便用位移来代替除法。

3、　去极值平均滤波

连续采样 n 次，去除其中的最大值和最小值，按 n-2 个采样值求平均，即得有效值。为使平均滤波方便，n-2 应为 2、4、8、16，故 n 常取 4、6、10、18。具体做法有两种：对于快变参数，先连续采样 n 次，然后再处理，但要在 RAM 中开辟出 n 个数据的暂存区。对于慢变参数，可一边采样，一边处理，而不必在 RAM 中开辟数据暂存区。

五、高级要求：

如果采集数据为六个 BCD 码数据（数据在程序中赋值给出），每个 BCD 码数据有五位（范围为 00000D-19999D）。存放单元为：（4DH，4CH，4BH，4AH，49H）、（48H，47H，46H，45H，44H，）、（43H，42H，41H，40H，3FH）、（3EH，3DH，3CH，3BH，3AH）、（39H，38H，37H，36H，35H）、（34H，33H，32H，31H，30H）。高字节在前。

要求将 BCD 码转变成二字节十六进制数，再采用去极值平均滤波法（去掉最小值及最大值，然后求平均值），求得有效值。

六、参考程序清单：

赋值子程序“FUZHI”省略，
结果在(23H,22H)。

```
        ORG 0000H
        AJMP MAIN
        ORG 0100H
MAIN:   LCALL FUZHI   ;给(2BH,2AH)-(21H,20H)赋值
        LCALL MAX
        LCALL MIN
        LCALL JIA
        SJMP $

MAX:    CLR C         ;(2BH,2AH)-(21H,20H)最大值放入(21H,20H)
        MOV A,20H
        SUBB A,22H
        MOV A,21H
        SUBB A,23H
        JC APUT1
        LJMP AXX1

APUT1:  MOV A,20H
```

```
        MOV 20H,22H
        MOV 22H,A
        MOV B,21H
        MOV 21H,23H
        MOV 23H,B


AXX1:   CLR C
        MOV A,20H
        SUBB A,24H
        MOV A,21H
        SUBB A,25H
        JC APUT2
        LJMP AXX2

APUT2:  MOV A,20H
        MOV 20H,24H
        MOV 24H,A
        MOV B,21H
        MOV 21H,25H
        MOV 25H,B

AXX2:   CLR C
        MOV A,20H
        SUBB A,26H
        MOV A,21H
        SUBB A,27H
        JC APUT3
        LJMP AXX3

APUT3:  MOV A,20H
        MOV 20H,26H
        MOV 26H,A
        MOV B,21H
        MOV 31H,27H
        MOV 27H,B

AXX3:   CLR C
        MOV A,20H
        SUBB A,28H
        MOV A,21H
        SUBB A,29H
        JC APUT4
        LJMP AXX4
```

```
APUT4: MOV A,20H
       MOV 20H,28H
       MOV 28H,A
       MOV B,21H
       MOV 21H,29H
       MOV 29H,B

 AXX4: CLR C
       MOV A,20H
       SUBB A,2AH
       MOV A,21H
       SUBB A,2BH
       JC APUT5
       LJMP AXX5

APUT5: MOV A,20H
       MOV 20H,2AH
       MOV 2AH,A
       MOV B,21H
       MOV 21H,2BH
       MOV 2BH,B
 AXX5: RET

  MIN: CLR C          ;(2BH,2AH)-(23H,22H)最小值放入(2BH,2AH)
       MOV A,2AH
       SUBB A,22H
       MOV A,2BH
       SUBB A,23H
       JNC PUT11
       AJMP XX1

PUT11: MOV A,2AH
       MOV 2AH,22H
       MOV 2AH,A
       MOV B,2BH
       MOV 2BH,23H
       MOV 23H,B

  XX1: CLR C
       MOV A,2AH
       SUBB A,24H
       MOV A,2BH
       SUBB A,25H
```

```
        JNC PUT22
        AJMP XX2

PUT22:  MOV A,2AH
        MOV 2AH,24H
        MOV 24H,A
        MOV B,2BH
        MOV 2BH,25H
        MOV 25H,B

  XX2:  CLR C
        MOV A,2AH
        SUBB A,26H
        MOV A,2BH
        SUBB A,27H
        JNC PUT33
        AJMP XX3

PUT33:  MOV A,2AH
        MOV 2AH,26H
        MOV 26H,A
        MOV B,2BH
        MOV 2BH,27H
        MOV 27H,B

  XX3:  CLR C
        MOV A,2AH
        SUBB A,28H
        MOV A,29H
        JNC PUT44
        AJMP XX4

PUT44:  MOV A,2AH
        MOV 2AH,28H
        MOV 28H,A
        MOV B,2BH
        MOV 2BH,29H
        MOV 29H,B
  XX4:  RET

  JIA:  MOV A,22H     ;(29H,28H)-(23H,22H)平均值放入(23H,22H)
        ADD A,24H
        MOV 22H,A
        MOV A,23H
```

```
ADDC A,25H
MOV 23H,A

MOV A,22H
ADD A,26H
MOV 22H,A
MOV A,23H
ADDC A,27H
MOV 23H,A

MOV A,22H
ADD A,28H
MOV 22H,A
MOV A,23H
ADDC A,29H
MOV 23H,A

MOV A,23H
RRC A
MOV 23H,A
MOV A,22H
RRC A
MOV 22H,A
CLR C
MOV A,23H
RRC A
MOV 23H,A
MOV A,22H
RRC A
MOV 22H,A
RET
```

第七章　非线性校正及标度变换程序设计实验

实验七　非线性校正及标度变换程序设计实验

一、实验目的：

了解非线性校正及标度变换的方法，掌握非线性校正及标度变换的程序设计。

二、实验内容：

根据表格中给出的数据，采用等节点分段直线插值法，建立非线性校正及标度变换公式（温度与 A/D 数值之间的关系式）。

根据给出的 A/D 采集数据（存放在 61H、60H 单元的两字节 16 进制数，由程序中赋值给出，范围为：0000H – FFFFH ），利用非线性校正及标度变换公式求出温度值（温度范围为 00.0℃—99.9℃）。

将温度值在 LED 显示器上显示。

三、实验步骤：

1、 硬件连接:

8255(2)片选信号 /CS 连接 /Y3—/Y7（任选其一）。

连接相关功能块电源。

LED 显示器电路原理图见第一章图 12。

2. 软件设计说明 :

1） 显示子程序的设计

显示温度值为三位十进制数，范围为 00.0℃—99.9℃(小数点在确定个位数的字型码时给出)。在右边三位 LED 显示器上显示。

8255（2）的 PA 口 PA7 控制小数点， PA6—PA0 控制字形段选（反向驱动），PB 口 PB5—PB0 控制字位（反向驱动）。

共阴极 LED 显示器。

2） 建立表格

已知 11 个测量点的采集数据见表。

温度（℃）	0.0	10.0	20.0	30.0	40.0	50.0
阻值（Ω）	100.0	103.9	107.8	111.7	115.5	119.4
A/D 数值	00195	02300	04319	06355	0302	10285
十六进制	00C3H	08FCH	10DFH	18D3H	206EH	282DH
温度（℃）	60.0	70.0	80.0	90.0	99．9	
阻值（Ω）	123.2	127.1	130.9	134.7	138.1	
A/D 数值	12256	14201	16161	18025	19627	
十六进制	2FE0H	3779H	3F21H	4669H	4CABH	

3） 建立插值公式：

将上表按温度均匀分成十段，以温度值和 A/D 采集数据为函数关系建立十段线性插值公式。

设直线方程 y=ax+b。y 是温度值，x 是 A/D 采集数据。

从表中可以得出每段的（x1,y1）和(x2,y2)。

求解方程 y1=ax1+b　　则 a=(y2-y1)/(x2-x1)

y2=ax2+b　　b=(x1y2-x2y1)/(x1-x2)

4） 建立插值公式举例：

从表中可知，0℃时 A/D 采集数据为 00195D（00C3H）,10℃时 A/D 采集数据为 02300D (08FCH)。

则 y1=000　　x1=195

y2=100　　x2=2300

（y 为温度值，包含一位小数）

根据以上方法求得 1/a ≈ 21.1D, b ≈ -1D。

为了提高测量精度，设 m = 1/a×10＝211D＝D3H, n = - b = 1D = 01H。

则 0℃—10℃的插值公式 y=ax+b 修改为：y=(10×x)/m - n。

y 是温度值，x 是 A/D 采集数据。

5） 非线性校正及标度变换：

根据 A/D 采集数据 x 的大小（两字节 16 进制数，由程序中赋值给出，存放在 61H、60H 单元）。选择插值公式，求出被测温度值 y。

6） 16 进制数→十进制数转换程序设计

根据插值公式求得的被测温度 y 为两字节 16 进制数，将其转换成十进制数（范围为 000D—999D），再将其送入显示缓冲区。小数点在确定个位数的字型码时给出。显示范围为 00.0℃～99.9℃。

7） 当 x 数值超出 00C3H—4CABH 范围时，给出欠量程或过量程标志（欠量程显示全“－”，过量程显示全“H” ）。

3、执行程序，观察结果。修改程序中的 x 值，重复以上过程。

四、高级要求：了解各种常用非线性校正及标度变换的方法，分析其特点。

五、参考程序清单:

8255(2)片选信号 /CS 连接 /Y3(6000H)

```
        ORG 0000H
        AJMP MAIN
        ORG 0100H
MAIN:   MOV A,#15H          ;给(61H,60H)赋值 (A/D 采集数值)
        MOV 61H,A
        MOV A,#0DFH
        MOV 60H,A
```

```
        MOV R0,#56H
        MOV R2,#03H
PUT1:   MOV @R0,#10H        ;给显示缓冲区(58H-56H)赋值显示全黑
        INC R0
        DJNZ R2,PUT1
        MOV DPTR,#6003H     ;对 8255(2)初始化(接 LED 显示器)
        MOV A,#81H
        MOVX @DPTR,A

PUT2:   ACALL CHANGE
        ACALL LED           ;调显示子程序
        NOP
        NOP
        AJMP PUT2

LED:    MOV R0,#20H                ;显示子程序
        MOV R1,#56H
LOOP:   MOV A,R0                   ;送位码(PB 口)
        MOV DPTR,#6001H
        MOVX @DPTR,A

        MOV A,@R1                  ;送段码(PA 口)
        MOV DPTR,#SEG
        MOVC A,@A+DPTR
        CJNE R0,#10H,LOOP1

DOT:    ADD A,#80H
        MOV DPTR,#6000H           ;个位带小数点
        MOVX @DPTR,A
        AJMP GOON

LOOP1:  MOV DPTR,#6000H           ;非个位不带小数点
        MOVX @DPTR,A

GOON:   ACALL DELAY
        MOV A,#7FH
        MOVX @DPTR,A               ;清黑

        MOV A,R0
        RR A
        MOV R0,A
        INC R1
        CJNE R0,#04H,LOOP          ;三位显示完?
        RET
```

```
 DELAY: MOV R7,#02H
DELAY0: MOV R6,#0FFH
DELAY1: DJNZ R6,DELAY1
        DJNZ R7,DELAY0
        RET

CHANGE: MOV R0,60H     ;非线性校及标度变换(含 16 进制→BCD 码)
        MOV R1,61H     ;按温度分段

        CJNE R1,#00H,L1     ;与 0℃(00C3H)比较。不等转至 L1
        CJNE R0,#0C3H,L2    ;                      不等转至 L2

  CHA1: MOV A,#00H          ;0℃-10℃
        AJMP GO

    L1: CJNE R1,#08H,L3     ;与 10℃(08FCH)比较。不等转至 L3
        CJNE R0,#0FCH,L4    ;                      不等转至 L4

  CHA2: MOV A,#02H          ;10℃-20℃
        AJMP GO

    L2: JC ERROR1           ;即小于 0℃时转 ERROR1
        AJMP CHA1           ;即高位为#00H,必在 0℃-10℃之间

ERROR1: MOV 56H,#11H        ;三位全部显示“-”
        MOV 57H,#11H
        MOV 58H,#11H
        AJMP QU2

    L3: JC CHA1             ;C=1,(61H)<#08H,必在 0℃-10℃
        CJNE R1,#10H,L5     ;与 20℃(10DFH)比较。
        CJNE R0,#0DFH,L6

  CHA3: MOV A,#04H          ;20℃-30℃
        AJMP GO

    L4: JC CHA1             ;C=1,(61H)=08H,(60H)<FCH,必在 0℃-10℃
        AJMP CHA2           ;C=0,(61H)=08H,(60H)>FCH,必在 10℃-20℃

    L5: JC CHA2                 ;C=1,(61H)<10H,必在 10℃-20℃
        CJNE R1,#18H,L7         ;与 30℃(18D3H)比较。
        CJNE R0,#0D3H,L8

  CHA4: MOV A,#06H        ;30℃-40℃
```

```
        AJMP GO

  L6:   JC CHA2          ;C=1,(61H)=10H,(60H)<DFH,必在 10℃-20℃
        AJMP CHA3        ;C=0,(61H)=10H,(60H)>DFH,必在 20℃-30℃

  L7:   JC CHA3
        CJNE R1,#20H,L9        ;与 40℃(206EH)比较。
        CJNE R0,#6EH,L10

CHA5:   MOV A,#08H             ;40℃-50℃
        AJMP GO

  L8:   JC CHA3
        AJMP CHA4

  L9:   JC CHA4
        CJNE R1,#28H,L11       ;与 50℃(282DH)比较。
        CJNE R0,#2DH,L12

CHA6:   MOV A,#0AH             ;50℃-60℃
        AJMP GO

 L10:   JC CHA4
        AJMP CHA5

 L11:   JC CHA5
        CJNE R1,#2FH,L13       ;与 60℃(2FE0H)比较。
        CJNE R0,#0E0H,L14

CHA7:   MOV A,#0CH             ;60℃-70℃
        AJMP GO

 L12:   JC CHA5
        AJMP CHA6

 L13:   JC CHA6
        CJNE R1,#37H,L15       ;与 70℃(3779H)比较。
        CJNE R0,#79H,L16

CHA8:   MOV A,#0EH             ;70℃-80℃
        AJMP GO

 L14:   JC CHA6
        AJMP CHA7
```

```
   L15: JC CHA7
        CJNE R1,#3FH,L17        ;与 80℃(3F21H)比较。
        CJNE R0,#21H,L18

  CHA9: MOV A,#10H              ;80℃-90℃
        AJMP GO

   L16: JC CHA7
        AJMP CHA8

   L17: JC CHA8
        CJNE R1,#46H,L19        ;与 90℃(4669H)比较。
        CJNE R0,#69H,L20

 CHA10: MOV A,#12H              ;90℃-99.9℃
        AJMP GO

   L18: JC CHA8
        AJMP CHA9

   L19: JC CHA9
        CJNE R1,#4CH,L21        ;与 99.9℃(4CABH)比较。
        CJNE R0,#0ABH,L22
        AJMP CHA10             ;恰好为 99.9℃时转 CHA10

   L20: JC CHA9
        AJMP CHA10

   L21: JC CHA10
        AJMP ERROR2

   L22: JC CHA10

ERROR2: MOV 56H,#12H            ;三位全部显示“H”
        MOV 57H,#12H
        MOV 58H,#12H
        AJMP QU2

     GO: MOV R5,A          ;计算 y=(10×x)/m - n
        MOV DPTR,#TABLE
        MOVC A,@A+DPTR
        MOV 37H,A          ;查表将 m 放入 37H(除数)
        MOV A,R5
```

```
        INC A
        MOVC A,@A+DPTR
        MOV 38H,A          ;查表将 n 放入 38H

MULKLM: CLR C
        MOV A,#0AH         ;计算 y=10×x
        MOV B,60H          ;X 为(61H,60H)
        MUL AB
        MOV 67H,A          ;积为(69H,68,67H)
        MOV 68H,B
        MOV A,#0AH
        MOV B,61H
        MUL AB
        ADDC A,68H
        MOV 68H,A
        MOV A,B
        ADDC A,#00H
        MOV 69H,A

        MOV 43H,#00H       ;计算(10×x)/m
        MOV 42H,#00H       ;被除数(69H,68H,67H) 除数(37H)
        MOV 62H,#00H       ;商(43H,42H) 余数(62H)
  DIVD: CLR C
        MOV A,67H
        SUBB A,37H
        MOV B,A
        JC QUIT1          ;不够减的话转至 QUIT1
        MOV 67H,A
        AJMP G            ;够减的话转至 G
 QUIT1: MOV A,68H
        JZ QUIT3   ;最低位不够减且第二位为 0,向最高位借位转 QUIT3
 QUIT2: MOV 67H,B ;最低位不够减但第二位不为 0 时
        MOV A,68H
        SUBB A,#00H
        MOV 68H,A
        AJMP G      ;此部分处理完毕后转至 G
 QUIT3: MOV A,69H
        JZ SUBB1    ;借位但最高位和第二位都为 0 时除法转换结束
        SUBB A,#00H
        MOV 69H,A
        MOV 68H,#0FFH
        MOV A,#0FFH
        SUBB A,37H
        ADD A,67H
```

```
        INC A
        MOV 67H,A

    G:  MOV 62H,67H          ;62H 放余数
        MOV A,42H
        ADD A,#01H
        MOV 42H,A
        MOV A,43H
        ADDC A,#00H
        MOV 43H,A
        AJMP DIVD
SUBB1:  MOV A,62H            ;62H 里放余数
        CJNE A,#5DH,SUBB2    ;不等转 SUBB2,进行下一步判别
        JC SUBB3             ;相等(5 入),跳转至 SUBB3
SUBB2:  JC SUBB4             ;(62H)<5DH,(4 舍)并跳转至 SUBB4
SUBB3:  CLR C
        MOV A,42H
        ADD A,#01H
        MOV 42H,A
        MOV A,43H
        ADDC A,#00H
        MOV 43H,A

SUBB4:  CLR C                ;计算 (10×x/m) - n
        MOV A,42H
        SUBB A,38H
        MOV 42H,A
        MOV A,43H
        SUBB A,#00H
        MOV 43H,A

 HTOB:  MOV 56H,#00H          ;16 进制(43H,42H)→BCD 码
        MOV 57H,#00H
        MOV 58H,#00H          ;BCD 码:5AH(千位)-56H(小数位)
        MOV 59H,#00H          ;实际只有(58H,57H,56H)
        MOV 5AH,#00H
HWWEI:  CLR C                 ;处理千位
        MOV A,42H
        SUBB A,#10H
        MOV B,A
        MOV A,43H             ;2710H=10000B
        SUBB A,#27H
        JC HQWEI
        INC 5AH
```

```
        MOV 43H,A
        MOV 42H,B
        AJMP HWWEI
HQWEI:  CLR C                   ;处理百位
        MOV A,42H
        SUBB A,#0E8H
        MOV B,A
        MOV A,43H               ;03E8H=1000B
        SUBB A,#03H
        JC HBWEI
        INC 59H
        MOV 43H,A
        MOV 42H,B
        AJMP HQWEI
HBWEI:  CLR C                    ;处理十位
        MOV A,42H
        SUBB A,#64H
        MOV B,A
        MOV A,43H                ;64H=100B
        SUBB A,#00H
        JC HSWEI
        INC 58H
        MOV 43H,A
        MOV 42H,B
        AJMP HBWEI
HSWEI:  CLR C                     ;处理个位
        MOV A,42H
        SUBB A,#0AH
        JC HGWEI                  ;0AH=10B
        INC 57H
        MOV 42H,A
        AJMP HSWEI
HGWEI:  MOV 56H,42H               ;处理小数位
  QU2:  RET

TABLE:  DB 0D3H,09H,0CAH,09H,0CCH,0CH,0C3H,1AH
        DB 0C6H,13H,0C5H,16H,0C3H,1EH,0C4H,19H
        DB 0BAH,43H,0A2H,0D6H                          ;参数 m,n

  SEG:  DB 40H,79H,24H,30H,19H,12H,02H,78H,00H
        DB 10H,08H,03H,46H,21H,06H,0EH,7FH,3FH,09H ;段码
```