

大连理工大学本科设计报告

(计算机原理实验综合设计)

题目： 基于 ADC 和 LCD 的报警系统

课程名称： 计算机原理实验

学院（系）： 信息与通信工程学院

专 业： 电子信息工程（英语强化）

班 级： 电英 2201

学 号： 20221071227

学生姓名： 乔百合

成 绩：

2024 年 12 月 27 日

题目：

1 设计要求

要求设计一个报警系统，当 ADC0809 转换结果在给定范围内时，绿色 LED 灯亮起，表示该系统处于正常状态。当 ADC0809 转换结果小于给定下限时，黄色 LED 亮，表示当前电压过小。当 ADC0809 转换结果大于上限时，红色 LED 亮，并在 LCD 上有所显示，自行设计 LCD 界面。ADC0809 的上限和下限可以在代码中设计初值，也可以使用按键等进行更改，提供用户交互功能。

2 设计分析及系统方案设计

程序采用顺序结构，采用 JMP ADCC 跳转到 ADC 采集程序。

(1) ADC0809 模数转换，将滑动变阻器的电压与 IN0 相连，选择端三位均接地，表示选中 000 即通道 0，ADC0809 可以进行采集滑动变阻器的电压，并转化为数字量存在 AL 内，当 AL 大于上限 60H 时，8255PB 口输出 01H，LED7 红灯亮，同时蜂鸣器响，LCD 第三行显示‘-HIGH---’；当 AL 在下限和上限之间时，8255PB 口输出 04H，LED5 绿灯亮；8255PB 口输出 02H，LED6 黄灯亮。

(2) LCD 显示模块，第二行显示‘ALARM---’，当 ADC 采集的电压值经转换后高于 60H 时，在第三行显示‘-HIGH---’，提示高于上限。

(3) ADC 采用软件延时法编程，首先向 CS 端写一个 AX 的值启动 ADC 转换，然后转向状态口，查询是否转换完成，状态最低为 1 时表示转换完成，从数据口读取数据到 AL 中便得到了转换结果，而这个过程需要消耗一定的时间，因此采用软件延时给其转换留有足够时间，

3 系统电路图

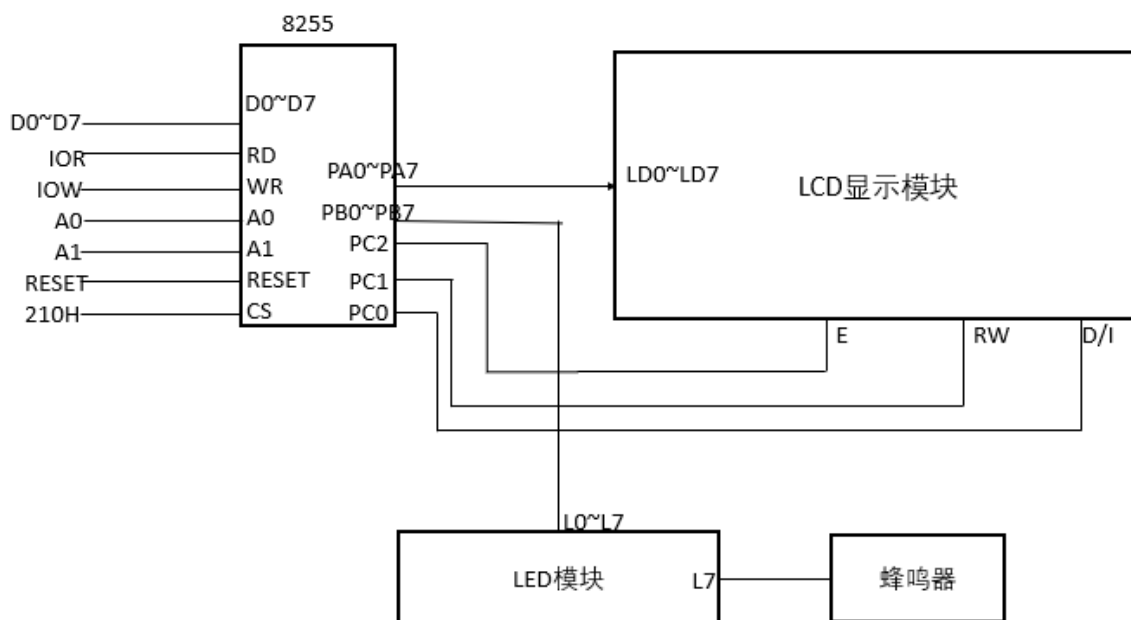


图 3.1 系统电路图

4 外围接口模块硬件电路功能描述

(1) ADC0809 模数转换，将滑动变阻器的电压与 IN0 相连，选择端三位均接地，表示选中 000 即通道 0，ADC0809 可以进行采集滑动变阻器的电压，并转化为数字量存在 AL 内，当 AL 大于上限 60H 时，8255PB 口输出 01H，LED7 红灯亮，同时蜂鸣器响，LCD 第三行显示‘-HIGH--’；当 AL 在下限和上限之间时，8255PB 口输出 04H，LED5 绿灯亮；8255PB 口输出 02H，LED6 黄灯亮。

(2) 128*64 字符 LCD 点阵显示模块电路，模块内置 128 个 ASCII 字符库，模块设计为八位并行模式，只需要通过数据线向模块写入对应命令、数据即可。4 号引脚 D/I 高电平表示数据，低电平表示命令；5 号引脚 R/W 高电平是读，低电平是写；6 号 E 引脚使能高电平有效。此电路有“独立”和“CPU（总线）”方式，由 SW9 开关设置，“独立”模式可以将 8255 的端口与 JP9 插座相连，以及对应的控制引脚（EN、W/R、D/I、RST）连接，此实验就是把 8255 的 PA 口与 JP9 相连，PA 口配置为方式 0 输出。总线模式下，模块自动定义器端口地址位 400H 且不用连接模块数据连线。CPU 在向 LCD 写命令之前，先查询 BF=1 才可以进行发送命令，此实验采用延时操作，简化代码。下表是相关汉字与 LCD 模块 DDRAM 地址对应关系，此实验只显示英文，因为和 ASCII 相关，显示比较方便。

表 4.1 128*64 字符 LCD 屏汉字显示坐标与 LCD 模块 DDRAM 地址的对应关系

Y 坐标	X 坐标							
Line1	80H	81H	82H	83H	84H	85H	86H	87H
Line2	90H	91H	92H	93H	94H	95H	96H	97H
Line3	88H	89H	8AH	8BH	8CH	8DH	8EH	8FH
Line4	98H	99H	9AH	9BH	9CH	9DH	9EH	9FH

(3) 可编程并行接口 8255A，连接 A0,A1 与总线 A0,A1，片选 I8255A 接 210H，则 I8255K 控制端位 23H，本实验将 I8255K 设为 1000 0000，，方式 0 状态下，表示 PAPBPC 三个端口均为输出模式，无其他联络信号。具有锁存功能，可以保持输出的数据，具有缓冲功能，输入无锁存功能。主机与外设之间采用“无条件输入输出”，不考虑两者速度平衡问题。

(4) LED 模块，8255A 的 PB 口输出接 L0~L7，对应位是高电平就亮灯，低电平不亮，因此当 ADC 转换结果变化时，根据转换结果 AL 在哪个范围内，亮对应颜色的灯。

(5) 蜂鸣器模块，高电平会发生响声，电路连接简单，但是具有实际意义，当电压高于设定值时需要报警器来提醒人们注意安全。

5 主程序中主要变量说明

变量名称	内存单元	功能
AL	大小为 DB	存储 ADC0809 转换后数据(AL 功能不唯一)
I8255A	210H	8255 起始地址，用于方式 0 输出，连接 LCD
I8255B	211H	8255PB 口用于方式 0 输出，连接 LED 提示灯
I8255C	212H	8255PC 口
I8255K	213H	8255 控制字
IOAD_S	220H	ADC

6 系统软件中各个子程序的功能描述

子程序名称	入口参数	出口参数	功能 描述
CLEAR	无	无	清除 LCD 屏幕
FUNCUP	无	无	设置 LCD 显示状态命令
LCD_DISP	CH	无	根据 CH 值显示 HZ_TAB 中对应的行信息，控制 LCD 显示
CMD_SETUP	无	无	设置 LCD 命令模式，控制 8255 端口,启动 LCD 执行命令
DATA_SETUP	无	无	设置 LCD 数据模式，控制 8255 端口,输出汉字编码
DELAY	无	无	软件延时,通过循环实现延迟效果
ADCC	无	无	ADC 转换

7 主程序程序流程图

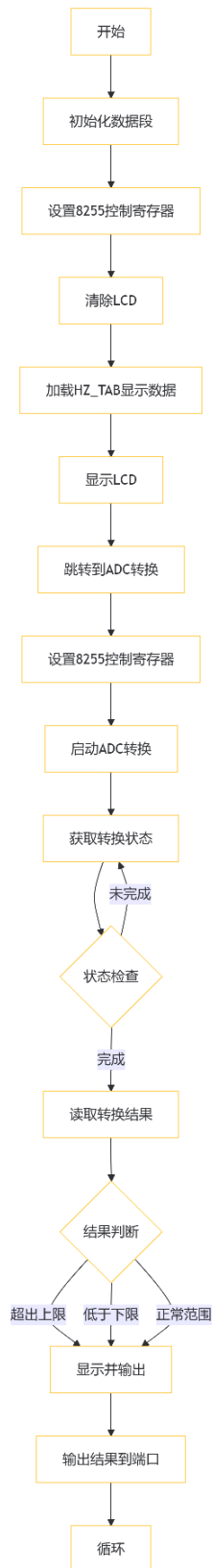


图 7.1 整体流程图

8 程序清单

.MODEL SMALL

.486

STACK SEGMENT STACK

 DB 128 DUP(?)

STACK ENDS

DATA SEGMENT

I8255A EQU 210H

I8255B EQU 211H

I8255C EQU 212H

I8255K EQU 213H

IOAD_S EQU 220H

HZ_TAB DW 0A3C1H,0A3CCH,0A3C1H ,0A3D2H,0A3CDH,0A3ADH,0A3ADH,0A3ADH ;ALARM

 DW 0A3ADH, 0A3C8H,0A3C9H,0A3C7H,0A3C8H,0A3ADH,0A3ADH,0A3ADH ;--HIGH---

 ; DW 0A3ADH,0A3C4H,0A3D5H,0A3D4H,0A3ADH,0A2B4H,0A2B8H,0A2B6H ;---DUT-4.8.6.

 ; DW 0A3ADH, 0A3C8H,0A3C9H,0A3C7H,0A3C8H,0A3ADH,0A3ADH,0A3ADH ;--HIGH---

 ;0A3C1H A ;; 0A3C8H H ;;0A3C9H I ;; 0A3C7H G ;;0A3C8H H

 ;0A3C1H A ;; 0A3C12H L ;;0A3C1H A ;; 0A3C18H R ;; 0A3C13H

M ;;

HZ_ADR DB ? ;存放显示行起始端口地址

DATA ENDS

CODE SEGMENT

 ASSUME CS:CODE,DS:DATA,SS:STACK

START:

;lcd 显示模块，显示系统功能以及报警提示信息！！

LCD:

 MOV AX,DATA ;设数据寄存器的值

 MOV DS,AX

 MOV DX,I8255K ;设 8255 方式 0，A 输出 B 口输出

 MOV AL,80H

```

        OUT    DX,AL                ;写入 8255 控制寄存器
        CALL   CLEAR                ;LCD 清除
        LEA    BX,HZ_TAB
        MOV    CH,2                 ;显示第 2 行信息
        CALL   LCD_DISP

;最重要的部分，这里 LCD 显示第二行结束一定要跳转到 ADC 转换模块!!!
        JMP    ADCC
LL:      JMP    LL

;清屏模块
CLEAR    PROC
        MOV    AL,0CH
        MOV    DX,I8255A
        OUT    DX,AL                ;设置 CLEAR 命令
        CALL   CMD_SETUP            ;启动 LCD 执行命令
        RET
CLEAR    ENDP

FUNCUP    PROC
        MOV    AL,34H                ;LCD 显示状态命令
        OUT    DX,AL
        CALL   CMD_SETUP
        RET
FUNCUP    ENDP

;LCD 显示
LCD_DISP  PROC
        LEA    BX,HZ_TAB
        CMP    CH,2
        JZ     DISP_SEC
        MOV    BYTE PTR HZ_ADR, 88H ;第三行起始端口地址
        ADD    BX,16                 ;指向第二行信息
        JMP    NEXT
DISP_SEC: MOV    BYTE PTR HZ_ADR,90H
NEXT:     MOV    CL,8
CONTINUE: PUSH    CX
        MOV    AL,HZ_ADR

```

```

MOV     DX,I8255A
OUT     DX,AL
CALL    CMD_SETUP           ;设定 DDRAM 地址命令
MOV     AX,[BX]
PUSH    AX
MOV     AL,AH               ;先送编码高位
MOV     DX,I8255A
OUT     DX,AL
CALL    DATA_SETUP         ;输出编码高字节
CALL    DELAY               ;延迟
POP     AX
MOV     DX,I8255A
OUT     DX,AL
CALL    DATA_SETUP         ;输出编码低字节
CALL    DELAY
INC     BX
INC     BX                   ;修改显示内码缓冲区指针
INC     BYTE PTR HZ_ADR     ;修改 LCD 显示端口地址
POP     CX
DEC     CL
JNZ     CONTINUE
RET

LCD_DISP  ENDP

CMD_SETUP PROC
MOV     DX,I8255C           ;指向 8255 端口控制端口
MOV     AL,00000000B        ;PC1 置 0,PC0 置 0 （LCD I 端=0, W 端 = 0）
OUT     DX,AL
NOP
CALL    DELAY
NOP
MOV     AL,00000100B        ;PC2 置 1 （LCD E 端 = 1）
OUT     DX,AL
NOP
CALL    DELAY
MOV     AL,00000000B        ;PC2 置 0, （LCD E 端置 0）
OUT     DX,AL

```



```

        CALL    DELAY
        RET
CMD_SETUP ENDP
DATA_SETUP PROC
        MOV     DX,I8255C           ;指向 8255 控制端口
        MOV     AL,00000001B        ;PC1 置 0, PC0=1 (LCD I 端=1)
        OUT     DX,AL
        NOP
        CALL    DELAY
        MOV     AL,00000101B        ;PC2 置 1 (LCD E 端 = 1)
        OUT     DX,AL
        NOP
        CALL    DELAY
        MOV     AL,00000001B        ;PC2 置 0, (LCD E 端 = 0)
        OUT     DX,AL
        NOP
        CALL    DELAY

```

DATA_SETUP ENDP

;延时函数

```

DELAY    PROC
        PUSH    CX
        PUSH    DX
        MOV     CX,0FFFFH
X1:      LOOP    X1
        POP     DX
        POP     CX
        RET
DELAY    ENDP
        RET

```

;ADC 转换模块，采集滑动变阻器电压

ADCC:

```

MOV     AX,DATA
        MOV     DS,AX

        MOV     DX,I8255K
        MOV     AL,10000000B;PA 口方式 0 输出
        OUT DX,AL

LOP1:   MOV     DX,IOAD_S
        OUT DX,AL;启动转换
        MOV     DX,IOAD_S+2
LOP2:   IN      AL,DX;获取状态端口信息
        TEST    AL,01H;判断最低位是不是 1
        JZ     LOP2;不是 1 的话继续获取
        MOV     DX,IOAD_S+1;获取数据端口信息
IN      AL,DX;转换结果存到 AL

;进入判断
;首先判断是否大于上限
CMP AL,60H
JNA NEXTY;小于等于上限
MOV AL,01H;大于 10H 红灯亮
;LCD 显示报警信息
CALL    CLEAR                ;LCD 清除
        LEA     BX,HZ_TAB
        MOV     CH,3          ;显示第 3 行信息:HIGH
        CALL    LCD_DISP
JMP SHUCHU

;电压过低
NEXTY:
CMP AL,02H
JNB NEXT1;大于等于下限
MOV AL,02H;低于下限 04H 黄灯亮
JMP SHUCHU

;安全区

```

```

NEXT1:
MOV AL,04H;在区间内绿灯亮

SHUCHU:
MOV     DX,I8255B
OUT DX,AL;结果送给 8255A 的 PA 口输出
CALL    DELAYY;软件延时法，使延时时间大于转换时间
JMP LOP1;不停转换

DELAYY:PUSH BX;压入堆栈，保护现场
PUSH    CX
MOV     BX,10H
DE2: MOV    CX,800
DE1: LOOP  DE1;循环将 BX-1
DEC BX
JNZ DE2;不为 0 就使 CX=800
POP CX
POP BX
RET

CODE     ENDS
        END     START

```

9 系统调试运行结果说明、分析所出现得问题，设计体会与建议

本次综合设计软件采用 TCP4.8.6 软件，硬件采用 DLUT-DGDZ-JSJ 计算机原理综合实验平台系统。

将 LED 和 LCD 得代码整合到一起的过程中，遇到了问题：两者都需要 8255 的接口，因此在设计 8255 模式的时候需要将 PA,PB 口均设置为输出模式，此时要注意连线。而我遇到最大的困难时无法同时运行两个程序，当先进行 ADC 采集实验的时候，LCD 就没显示；当先进行 LCD 显示时，ADC 无法采集，LED 无法亮。想过引入中断法，但是中断服务程序里一般不写延时函数，这两个均使用延时函数，故不太适合用中断。因此，我将 LCD 显示代码放在前面，当显示结束时，使用 JMP ADCC 语句跳转到 ADC 采集程序里，而当转换结果高于上限时，8255PB 输出 0000 0001,使红灯亮，再加一段 LCD 显示‘HIGH---’的代码，表示此时电压高于上限，蜂鸣器响，实现了报警功能。

此外，我还进行了用户自定义上限下限值的尝试，但是 4*4 小键盘没有成功扫描，和同学交流后没有解决，最终只做了代码段定义的上下限值。

6 结论及设计体会

这在学习了计算机原理理论课后，我们在开始了计算机原理实验课的学习，因此会有一定的理论基础，并且之前也做过汇编语言的作业，所以整体来说难度合适，并且在这个过程中有能力自己去探索并且解决问题，相比于没有任何前置课程会效果更好。

对于本次大作业，想法来源于讲义给定题目 ADC 报警系统设计，我在此基础上添加了 LCD 显示，蜂鸣器等外设，使整体更加丰满。在将 LED 和 LCD 得代码整合到一起的过程中，遇到了问题，最后使用 JMP ADCC 语句跳转到 ADC 采集程序里解决了问题。遇到困难不能逃避，不能总想着换题目，应该想想怎么解决。

本学期的计算机原理实验让我感受到了认真做实验的乐趣，更有底气面对困难，希望以后也能认真对待其他专业课程及实验。

参考文献

《计算机原理综合实验课程》 作者 秦晓美 巢明 大连理工大学电工电子实验中心 2024 年 9 月