**湖北汽车工业学院实验报告**

班 号 计算机222 学 号 202202296  姓 名 章崇文

选课班中的序号  完成日期 年 月 日 至 节

实验五 **8255和LED数码管显示实验**

**一、实验目的**

1．掌握并行接口8255A的工作原理及使用方法。

2．了解七段数码管显示数字的原理。

3．掌握多位数码显示的接口技术。

**图表, 图示, 箱线图

描述已自动生成二、实验电路**

**三、实验内容**

1．静态显示：按图3连接好电路，将8255A的A口PA0～PA6分别与七段数码管的段码驱动输入端a～g相连，位码驱动输入端S1接+5V（选中），S0、S2、S3、dp接地（关闭）。编程从键盘输入一位十进制数字（0～9），在七段数码管上显示出来。

2．动态显示：按图4连接好电路，七段数码管段码连接不变，位码驱动输入端S1、S0接8255C口的PC1、PC0，S2、S3接地（关闭）。编程在两个数码管上显示56。

**四、编程提示**

1．实验台上的七段数码管为**共阴极**，码段采用**同相驱动**，输入端加高电平，选中的数码管亮，位码加**反相驱动器**，位码输入端高电平选中。

表格

低可信度描述已自动生成

2.

**五、实验程序（必须加注释）和结果**

**实验一**

DATA SEGMENT

*; 定义一个字节数组LED，存储0~9对应的7段数码管段码值*

      LED  DB 3FH,06H,5BH,4FH,66H,6DH,7DH,07H,7FH,6FH

*; 定义一个字符串MSG，包含要显示的个人信息，以'$'结尾*

      MSG  DB 0DH,0AH,'zhangchongwen 202202296 lizihang 202202297$',0DH,0AH,'$'

DATA ENDS

CODE SEGMENT

            ASSUME CS:CODE, DS:DATA

      START:

*; 初始化数据段寄存器DS*

            MOV    AX, DATA

            MOV    DS, AX

*; 初始化8255芯片，假设8255的控制端口地址是28BH*

            MOV    DX, 28BH

            MOV    AL, 80H

            OUT    DX, AL

*; 输出个人信息*

*; 加载字符串MSG的地址到DX寄存器*

            LEA    DX, MSG

*; 设置功能号为09H，用于显示字符串*

            MOV    AH, 09H

*; 调用DOS中断21H，显示字符串*

            INT    21H

      SSS:

*; 从键盘读取一个字符，结果存储在AL中*

            MOV    AH, 01

            INT    21H

*; 检查输入的字符是否小于'0'*

            CMP    AL, '0'

            JL     EXIT

*; 检查输入的字符是否大于'9'*

            CMP    AL, '9'

            JG     EXIT

*; 将字符转换为对应的数字（减去30H）*

            SUB    AL, 30H

*; 加载LED表的起始地址到BX寄存器*

            LEA    BX, LED

*; 使用AL中的数字作为索引，从LED表中获取对应的段码*

            XLAT

*; 从8255的A口输出段码*

*; 假设8255的A口地址是288H*

            MOV    DX, 288H

            OUT    DX, AL

*; 跳转回SSS标签，等待下一次输入*

            JMP    SSS

      EXIT:

*; 程序结束，返回DOS*

            MOV    AH, 4CH

            INT    21H

CODE ENDS

END START

* 程序首先会显示字符串zhangchongwen 202202296 lizihang 202202297。
* 然后进入一个循环，等待用户输入数字字符（'0'到'9'）。
* 如果输入的是数字字符，程序会将对应的段码输出到8255的A口，控制7段数码管显示相应的数字。
* 如果输入的不是数字字符，程序会结束并返回DOS。

**实验二**

显示56

DATA SEGMENT

*; 定义一个字节数组LED，存储0~9对应的7段数码管段码值*

       LED     DB 3FH,06H,5BH,4FH,66H,6DH,7DH,07H,7FH,6FH

*; 定义要打印的字符串MESSAGE，并以$结尾表示字符串结束*

       MESSAGE DB 'zhangchongwen 202202296 lizihang 202202297 $',0

DATA ENDS

CODE SEGMENT

              ASSUME CS:CODE, DS:DATA

       START:

*; 初始化数据段寄存器*

              MOV    AX, DATA

              MOV    DS, AX

*; 打印字符串*

*; 将字符串MESSAGE的地址加载到DX寄存器*

              LEA    DX, MESSAGE

*; 设置AH为09H，准备打印字符串*

              MOV    AH, 09H

*; 调用DOS中断21H，打印字符串*

              INT    21H

*; 配置8255芯片*

*; 写控制字使8255的A口、C口为输出方式*

              MOV    DX, 28BH

              MOV    AL, 80H

              OUT    DX, AL

       LOOP1:

*; 清除8255 C口*

*; 将AL设置为00H，清除C口*

              MOV    AL, 00H

              MOV    DX, 28AH

              OUT    DX, AL

*; 显示数字5*

*; 将AL设置为5*

              MOV    AL, 5

*; 将LED表的起始地址加载到BX寄存器*

              MOV    BX, OFFSET LED

*; 查表求出相应的段码给AL*

              XLAT

*; 将段码输出到8255的A口*

              MOV    DX, 288H

              OUT    DX, AL

*; 选择8255 C口位2*

*; 将AL设置为02H，选择C口位2*

              MOV    AL, 02H

              MOV    DX, 28AH

              OUT    DX, AL

*; 延时*

*; 设置CX为3000，进行延时*

              MOV    CX, 3000

       DELAY:

*; 循环延时*

              LOOP   DELAY

*; 清除8255 C口*

*; 将AL设置为00H，清除C口*

              MOV    AL, 00H

              MOV    DX, 28AH

              OUT    DX, AL

*; 显示数字6*

*; 将AL设置为6*

              MOV    AL, 6

*; 将LED表的起始地址加载到BX寄存器*

              MOV    BX, OFFSET LED

*; 查表求出相应的段码给AL*

              XLAT

*; 将段码输出到8255的A口*

              MOV    DX, 288H

              OUT    DX, AL

*; 选择8255 C口位1*

*; 将AL设置为01H，选择C口位1*

              MOV    AL, 01H

              MOV    DX, 28AH

              OUT    DX, AL

*; 延时*

*; 设置CX为3000，进行延时*

              MOV    CX, 3000

       DELAY1:

*; 循环延时*

              LOOP   DELAY1

*; 检查键盘输入*

*; 检查是否有键盘输入*

              MOV    AH, 01H

              INT    16H

*; 如果有输入，跳转到EXIT*

              JNZ    EXIT

*; 如果没有输入，继续循环*

              JMP    LOOP1

       EXIT:

*; 清除8255 A口*

*; 将AL设置为0H，清除A口*

              MOV    AL, 0H

*; 将0H输出到8255的A口*

              MOV    DX, 288H

              OUT    DX, AL

*; 返回DOS*

*; 设置AH为4CH，准备返回DOS*

              MOV    AH, 4CH

*; 调用DOS中断21H，结束程序*

              INT    21H

CODE ENDS

END START

* 程序首先会显示字符串zhangchongwen 202202296 lizihang 202202297。
* 然后进入一个循环，依次显示数字5和数字6，并通过8255的C口选择不同的位。
* 每次显示后进行延时。
* 检查是否有键盘输入，如果有输入则退出循环。
* 退出循环后，清除8255的A口，并结束程序。

**六、实验总结和体会**

通过本次实验，不仅掌握了汇编语言的基本编程技巧，还深入理解了硬件与软件的交互过程。汇编语言虽然复杂，但其直接操作硬件的能力使其在某些应用场景中具有不可替代的优势。实验过程中积累的经验和技巧，对于后续的硬件编程和系统开发具有重要的指导意义。

**体会**

汇编语言编程需要对硬件有深入的理解，尤其是对寄存器、端口地址和控制字的配置。代码的每一行都需要精确控制，任何小的错误都可能导致程序无法正常运行。通过本次实验，深刻体会到硬件与软件的紧密结合。汇编语言直接操作硬件，能够实现高效的控制和响应。8255芯片的配置和使用，展示了如何通过软件控制硬件设备，实现特定的功能。调试汇编程序需要耐心和细致，通过逐步执行和观察寄存器和端口的变化，可以找到问题的根源。使用调试工具（如DEBUG）可以帮助定位问题，理解程序的执行流程。DOS中断INT 21H提供了丰富的功能，如显示字符串和读取键盘输入，简化了程序的编写。中断的使用需要了解其功能号和参数，合理调用可以提高程序的效率和可读性。延时是通过循环实现的，延时时间取决于循环次数和CPU的时钟频率。延时的精确控制对于硬件设备的正常工作非常重要，需要根据实际需求进行调整。在循环中检查键盘输入，确保程序能够响应用户的操作，提高了程序的健壮性。通过条件跳转和循环控制，实现了程序的灵活性和可扩展性。

**得 分\_\_\_\_\_\_\_\_\_\_\_\_\_**

**评阅日期\_\_\_\_\_\_\_\_\_\_\_\_\_**

**教师签名\_\_\_\_\_\_\_\_\_\_\_\_\_**