



**实 验 报 告**

**课 程 名 称 微机原理与接口技术**

**专 业 班 级 计算机科学与技术18-3 班**

**学生姓名及学号 2018211991余梓俊**

**指 导 教 师 毕 翔**

**2019 ～ 2020 学年第 二 学期**

《微机原理与接口技术》课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | 响铃程序 | | |
| 实验日期 | 2020.05.17 | 实验成绩 |  |
| **一、实验目的和要求**  目的：掌握响铃符的使用。  要求：编写程序，从键盘接受输入字符，如是数字N，则响铃N次，如不是数字或数字0，则不响。 | | | |
| **二、实验内容（包括实验步骤和代码）**    ; multi-segment executable file template.  data segment      ; add your data here!      pkey db "press any key from 1~9", 0dh, 0ah, "$"  ends  stack segment      dw   128  dup(0)  ends  code segment  start:  ; set segment registers:      mov ax, data      mov ds, ax      mov es, ax  startOfMain:      lea dx, pkey      mov ah, 9      int 21h        ; output string at ds:dx        ; wait for any key....      mov ah, 1      int 21h        push ax      mov ah, 02h      mov dl, 0dh      int 21h      mov dl, 0ah      int 21h      pop ax        ; 回车或空格      cmp al, 0dh      je endOfMain      cmp al, 20h      je endOfMain        ; 比1小或比9大      cmp al, 31h      jb startOfMain      cmp al, 39h      ja startOfMain        and al, 0fh      mov ah, 00h      mov bp, ax      call ring      jmp startOfMain    endOfMain:      mov ax, 4c00h ; exit to operating system.      int 21h  ends      proc ring      push ax      push cx      push dx      printRing:      MOV dl, 07      MOV ah, 02      int 21h      ; 延迟1s      ; 1,000,000 = f4240h      mov cx, 000fh      mov dx, 4240h      mov ah, 86h      int 15h      dec bp      cmp bp, 0      jne printRing        pop dx      pop cx      pop ax      ret  endp  end start ; set entry point and stop the assembler. | | | |
| **三、实验结果及其分析（可附实验结果图片）**  输入4时产生四下蜂鸣    蜂鸣结束后输入0和a均立即提示再次输入    输入空格后程序结束    *遗留问题：*  *使用*  *Mov ah, 86h*  *Int 15h*  *进行延时操作，有时第一次蜂鸣和第二次蜂鸣之间的时间间隔会小于指定的间隔时长*。 | | | |

《微机原理与接口技术》课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | 交通灯 | | |
| 实验日期 | 2020.05.17 | 实验成绩 |  |
| **一、实验目的和要求**  运行emu8086自带交通灯示例程序，熟悉emu8086软件的操作以及emu8086指示灯的编码方式。 | | | |
| **二、实验内容（包括实验步骤和代码）**  ; controlling external device with 8086 microprocessor.  ; realistic test for c:\emu8086\devices\Traffic\_Lights.exe  #start=Traffic\_Lights.exe#  name "traffic"  mov ax, all\_red  out 4, ax  mov si, offset situation  next:  mov ax, [si]  out 4, ax  ; wait 5 seconds (5 million microseconds)  mov cx, 4Ch ; 004C4B40h = 5,000,000  mov dx, 4B40h  mov ah, 86h  int 15h  add si, 2 ; next situation  cmp si, sit\_end  jb next  mov si, offset situation  jmp next  ; FEDC\_BA98\_7654\_3210  situation dw 0000\_0011\_0000\_1100b  s1 dw 0000\_0110\_1001\_1010b  s2 dw 0000\_1000\_0110\_0001b  s3 dw 0000\_1000\_0110\_0001b  s4 dw 0000\_0100\_1101\_0011b  sit\_end = $  all\_red equ 0000\_0010\_0100\_1001b | | | |
| **三、实验结果及其分析（可附实验结果图片）**    *由示例代码和运行结果可以知道，emu8086指示灯的编码方式为：图中灯的十六进制编号对应2字节数据中从低到高位的十六进制位置编号，1为亮，0为灭。*  *由示例代码还能看出，emu8086指示灯的数据端口的地址为4。* | | | |

《微机原理与接口技术》课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | LED显示 | | |
| 实验日期 | 2020.05.17 | 实验成绩 |  |
| **一、实验目的和要求**  运行emu8086自带LED数字显示示例程序，掌握emu8086的LED数字显示外设的控制方法。 | | | |
| **二、实验内容（包括实验步骤和代码）**  ; this example shows how to access virtual ports (0 to 65535).  ; these ports are emulated in this file: c:\emu8086.io  ; this technology allows to make external add-on devices  ; for emu8086, such as led displays, robots, thermometers, stepper-motors, etc... etc...  ; anyone can create an animated virtual device.  ; c:\emu8086\devices\led\_display.exe  #start=led\_display.exe#  #make\_bin#  name "led"  mov ax, 1234  out 199, ax  mov ax, -5678  out 199, ax  ; Eternal loop to write  ; values to port:  mov ax, 0  x1:  out 199, ax  inc ax  jmp x1  hlt | | | |
| **三、实验结果及其分析（可附实验结果图片）**        *实例代码功能为先显示1234，再显示-5678，然后0开始不停自增并显示。*  *由示例代码可以看出，LED外设的数据端口地址为199。* | | | |

《微机原理与接口技术》课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | 流水灯控制程序 | | |
| 实验日期 | 2020.05.17 | 实验成绩 |  |
| **一、实验目的和要求**  要求：编写汇编程序，实现对仿真程序中指示灯的流水控制。  具体的流水控制过程是：绿灯，黄灯，红灯，然后顺时针方向依次点亮其余三组灯。  目的：掌握IO端口的操作方法以及指示灯显示码的编码方法。 | | | |
| **二、实验内容（包括实验步骤和代码）**    ; controlling external device with 8086 microprocessor.  ; realistic test for c:\emu8086\devices\Traffic\_Lights.exe  #start=Traffic\_Lights.exe#  name "traffic"    ; 210 ba9 876 543  mov ax, 0000\_0000\_0000\_0100b  ; 换灯时右移一位（除了210和ba9之间）  mov cx, 12  l1:      out 4, ax      push ax      push cx      ; 1,000,000 = f4240h      mov cx, 000fh      mov dx, 4240h      mov ah, 86h      int 15h      pop cx      pop ax      shr ax, 1      cmp ax, 0 ; 等于0时说明显示完了第一组的三次      jnz continue      mov ax, 0000\_1000\_0000\_0000b      continue:  loop l1  hlt | | | |
| **三、实验结果及其分析（可附实验结果图片）**      中间省略。。。  最后两次：    *实验结果与预期相同* | | | |