**深 圳 大 学 实 验 报 告**

**课程名称：­ 微型计算机技术**

**实验项目名称： 模块化程序设计**

**学院： 电子与信息工程学院**

**专业： 通信工程**

**指导教师： 张力**

**报告人： 王俊彬 学号： 2020282017 班级： 通信04**

**实验时间： 2022年5月19日**

**实验报告提交时间： 2022年5月20日**

**教务处制**

|  |
| --- |
| 实验目的与要求：   1. 掌握8086汇编，能够进行模块化程序设计 2. 完成Lab2汇编程序，包括判断奇偶的方法和堆栈的使用 |
| 实验原理与内容：  根据微机原理课程第三、四章内容，使用相应的8086汇编在EMU8086编写程序。  给定汇编程序lab2.asm，请添加汇编代码完成以下要求。请在EMU8086中测试你的代码。   1. 接收键盘输入的一组个位数（不超过20个），以回车符结束，将输入的数按次序存入内存并在屏幕上显示输入的数。 2. 对输入的数进行测试，分别统计出奇数、偶数的个数，并分别求出奇数之和、偶数之和，将个数与和以十进制数的形式在屏幕上显示。   ; 数据段  data segment  buffer db 20 dup(0) ;存储输入数据  data ends  ; 堆栈段  stack segment  data\_ss\_w dw 10 dup(0)  stack ends  ; 附加数据段  extra segment  data\_es\_b db 20 dup(0)  extra ends  ; 代码段  code segment  assume cs:code, ds:data, ss:stack, es:extra  start:  ; 设置DS, SS和ES段寄存器, 不需要设置CS段寄存器  mov ax, data  mov ds, ax    mov ax, stack  mov ss, ax  mov sp, 20 ; 设置堆栈指针SP寄存器    mov ax, extra  mov es, ax    ;请在此处添加你的代码    ;返回到DoS操作系统  mov ax, 4c00h  int 21h  code ends   1. end start ;结束汇编程序，并指明程序入口在start标号处 |
| 设计思想（需绘制实验原理图或流程图）： |
| 汇编程序清单（程序中需有必要注释）：  DATA SEGMENT      BUFFER DB 100 DUP('$') ;存储输入      DECIMAL DB 3 DUP('$'),'$' ;存储十进制输出，最多三位数      ODD\_NUM DW 1 DUP(0),'$$'  ;奇数个数      EVEN\_NUM DW 1 DUP(0),'$$' ;奇数之和      ODD\_SUM DW 1 DUP(0),'$$'  ;偶数个数  EVEN\_SUM DW 1 DUP(0),'$$' ;偶数之和      ;一些字符串      ILLEGAL\_WARN DB "Illegal input (only space and digits are allowed), try again!",'$'      ODD\_INFO DB "The number of odd digits is ",'$'      EVEN\_INFO DB "The number of even digits is ",'$'      SUM\_INFO  DB "The sum of these digits is ",'$'  DATA ENDS  STACK SEGMENT      DATA\_SS\_W DB 5 DUP(0)  STACK ENDS  CODE SEGMENT      ASSUME CS:CODE,DS:DATA,SS:STACK  START:      ;初始化      MOV AX, DATA      MOV DS, AX      MOV AX, STACK  MOV SS, AX      ;存储输入      MOV BUFFER,40H      LEA DX,BUFFER      MOV AH,0AH  INT 21H      ;循环变量      XOR CX,CX      MOV CL,BUFFER+1      MOV SI,02H      MOV BL,02H  ;判断是否有效  VAILD:      XOR AX,AX  MOV AL,BUFFER[SI]      CMP AL,' '  JZ NEXT      CMP AL,'0'      JB ILLEGAL      CMP AL,'9'  JA ILLEGAL      CMP BUFFER[SI+1],'0'      JB JUDGE\_ODD\_OR\_EVEN      CMP BUFFER[SI+1],'9'  JA JUDGE\_ODD\_OR\_EVEN      JMP ILLEGAL  ;通过最后一位为0或1判断奇偶  JUDGE\_ODD\_OR\_EVEN:      SUB AL,'0'      MOV BL,01H  AND BL,AL      CMP BL,00H      JZ IS\_EVEN  ;得到奇数  IS\_ODD:      INC ODD\_NUM      ADD ODD\_SUM,AX      JMP NEXT  ;得到偶数  IS\_EVEN:      INC EVEN\_NUM      ADD EVEN\_SUM,AX  ;遍历输入  NEXT:      INC SI  LOOP VAILD      CALL END\_LINE  ;打印奇数  PRINT\_ODD:      LEA DX,ODD\_INFO      MOV AH,09H  INT 21H      MOV AX,ODD\_NUM      MOV DX,AX  CALL CONVERT      LEA DX,SUM\_INFO      MOV AH,09H  INT 21H      MOV DX,ODD\_SUM      MOV AX,ODD\_SUM      CALL CONVERT  ;打印偶数  PRINT\_EVEN:      LEA DX,EVEN\_INFO      MOV AH,09H      INT 21H      MOV AX,EVEN\_NUM      MOV DX,EVEN\_NUM  CALL CONVERT      LEA DX,SUM\_INFO      MOV AH,09H  INT 21H      MOV AX,EVEN\_SUM      MOV DX,EVEN\_SUM      CALL CONVERT  ;结束程序  END\_PROGRAM:      MOV AX,4C00H  INT 21H  ;非法输入处理  ILLEGAL:  CALL END\_LINE      LEA DX,ILLEGAL\_WARN      MOV AH,09H      INT 21H        MOV AH,4CH  INT 21H  ;打印回车  END\_LINE PROC NEAR      MOV DX,0AH      MOV AH,02H  INT 21H      MOV DX,0DH      MOV AH,02H  INT 21H      RET  END\_LINE ENDP  ;十六进制转十进制并输出  CONVERT PROC NEAR      XOR SI,SI      XOR DI,DI  MOV BL,0AH      ;判断十进制的位数      CMP DL,64H      JAE GTE\_HUNDRED      CMP DL,0AH      JAE GTE\_TEN      CMP DL,00H      JAE GTE\_ZERO        JMP ILLEGAL        ;百位      GTE\_HUNDRED:      DIV BL      XOR CH,CH      MOV CL,AH      PUSH CX        INC DI      XOR AH,AH      ;十位      GTE\_TEN:      DIV BL      XOR CH,CH      MOV CL,AH      PUSH CX        INC DI      XOR AH,AH      ;个位      GTE\_ZERO:      DIV BL      XOR CH,CH      MOV CL,AH  PUSH CX      INC DI        ;依次弹出栈顶，得到十进制数      POP\_STACK\_TOP:      POP AX      MOV [DECIMAL+SI],AL      ADD [DECIMAL+SI],30H        INC SI      CMP DI,SI      JA POP\_STACK\_TOP        LEA DX,DECIMAL      MOV AH,09H      INT 21H        CALL END\_LINE        ;输出后清空      MOV [DECIMAL],'$'      MOV [DECIMAL+1],'$'      MOV [DECIMAL+2],'$'        RET  CONVERT ENDP  CODE ENDS  END  START |
| 分析实验结果（显示程序运行结果并分析解释）：  样例1：  1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0  运行结果：  Snipaste_2022-05-20_20-30-16  程序先将输入存入到BUFFER中，后续利用最后一位存储为1是奇数，为0是偶数进行奇偶判断并计算个数和总和，因为实际存储的是十六进制，输出时需要十六进制数除十取余，用栈来存储和得到十进制形式，最后打印到屏幕上。  样例2：  1 2 3 4 5  运行结果：  Snipaste_2022-05-20_20-31-16  与样例1同理。  样例3：  1 2 3 4 5 a b  运行结果：  Snipaste_2022-05-20_20-31-55  在VALID代码段验证输入有效性时，发现a不属于空格和‘0’到‘9’范围，无条件跳转到ILLEGAL段，打印错误信息，结束程序。 |

|  |
| --- |
| 实验过程中遇到的问题及解决方案：   1. BUFFER初值为24H即36，最多只能拿到18个数字：   MOV BUFFER,40H  保证最多输入20个数字。   1. 输出要将十六进制转十进制，并且存在最多三位数字：   将待输出的数据与不同位数对应的十六进制数进行比较，跳转到对应段，并且自增DI，用于出栈的执行次数，从而得到对应位数的十进制数。 |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。