**西安电子科技大学**

**计算机组成与结构课程设计（II） 课程实验报告**

**实验名称 基本IO口扩展实验**

计算机科学与技术 学院 2203051 班

成 绩

姓名 陈璐 学号 22009201353

同作者 无

实验日期 2024 年 11 月 21 日

|  |
| --- |
| 指导教师评语：  指导教师：  年 月 日 |
| **实验报告内容基本要求及参考格式**  一、实验目的  二、实验所用仪器（或实验环境）  三、实验基本原理及步骤（或方案设计及理论计算）  四、实验数据记录（或仿真及软件设计）  五、实验结果分析及回答问题（或测试环境及测试结果） |

实验地点EII-312实验批次第02班

实验一 汇编语言编程实验

1. 实验目的

1. 掌握汇编语言的编程方法

2. 掌握DOS功能调用的使用方法

3. 掌握汇编语言程序的调试运行过程

1. 实验所用仪器

星研集成编译软件、硬件系统（实验箱）

1. 实验基本原理及步骤

1. 将指定数据区的字符串数据以ASCII码形式显示在屏幕上，并通过DOS功能调用完成必要提示信息的显示。

2. 在屏幕上显示自己的学号姓名信息。

3. 循环从键盘读入字符并回显在屏幕上，然后显示出对应字符的ASCII码，直到输入”Q”或“q”时结束。

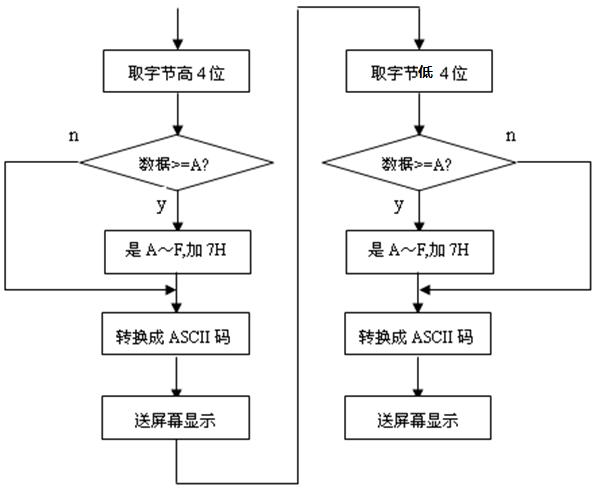
4. 自主设计输入显示信息，完成编程与调试，演示实验结果。

实验步骤：

1. 运行QTHPCI软件，根据实验内容，参考程序流程图编写程序。

2. 选择“项目”菜单中的“编译”或“编译连接”对实验程序进行编译连接。

3. 选择“调试”菜单中的“进行调试”，进入Debug调试，观察调试过程中传输指令执行后各寄存器及数据区的内容。按F9连续运行。



四、实验数据记录

实现代码：

.model small

.stack 100h

.data

hello\_msg db 'Hello, world!', 0Dh, 0Ah, '$'

name\_prompt db 'Enter your name: $'

id\_prompt db 'Enter your student ID: $'

name db 50 dup('$')

id db 20 dup('$')

newline db 0Dh, 0Ah, '$'

ascii\_msg db 'ASCII code: $'

quit\_msg db 'Program ended. Press any key to exit.$'

.code

main proc

mov ax, @data

mov ds, ax

; Display "Hello, world!"

mov ah, 09h

lea dx, hello\_msg

int 21h

; Prompt for name

mov ah, 09h

lea dx, name\_prompt

int 21h

; Read name

mov ah, 0Ah

lea dx, name

mov byte ptr [name], 48

int 21h

; Print newline

mov ah, 09h

lea dx, newline

int 21h

; Prompt for student ID

mov ah, 09h

lea dx, id\_prompt

int 21h

; Read student ID

mov ah, 0Ah

lea dx, id

mov byte ptr [id], 18

int 21h

; Print newline

mov ah, 09h

lea dx, newline

int 21h

; Echo name and ID

mov ah, 09h

lea dx, name+2

int 21h

mov ah, 09h

lea dx, newline

int 21h

mov ah, 09h

lea dx, id+2

int 21h

mov ah, 09h

lea dx, newline

int 21h

input\_loop:

; Read a character

mov ah, 01h

int 21h

; Check if 'q' or 'Q' was entered

cmp al, 'q'

je end\_program

cmp al, 'Q'

je end\_program

; Print ASCII code

push ax

mov ah, 09h

lea dx, ascii\_msg

int 21h

pop ax

mov ah, 0

call print\_number

mov ah, 09h

lea dx, newline

int 21h

jmp input\_loop

end\_program:

; Display quit message

mov ah, 09h

lea dx, quit\_msg

int 21h

; Wait for keypress

mov ah, 01h

int 21h

; Exit program

mov ax, 4C00h

int 21h

main endp

; Subroutine to print a number in AX

print\_number proc

push ax

push bx

push cx

push dx

mov bx, 10

xor cx, cx

divide\_loop:

xor dx, dx

div bx

push dx

inc cx

test ax, ax

jnz divide\_loop

print\_loop:

pop dx

add dl, '0'

mov ah, 02h

int 21h

loop print\_loop

pop dx

pop cx

pop bx

pop ax

ret

print\_number endp

end main

实验二 数码转换实验

1. 实验目的

1. 掌握不同进制数及编码相互转换的程序设计方法。

2. 掌握运算类指令编程及调试方法。

3. 掌握循环程序的设计方法

1. 实验所用仪器

星研集成编译软件、硬件系统（实验箱）

1. 实验基本原理及步骤

实验原理：

十进制数可以表示为：Dn\*10n+Dn-1\*10n-1+…+D0\*100= Di\*10i

其中Di代表十进制数1、2、3、…、9、0。

上式可以转换为： Di\*10i=（（（Dn\*10+Dn-1）\*10+ Dn-2）\*10+…+ D1）\*10+ D0

由上式可归纳出十进制数转换为二进制数的方法：从十进制数的最高位Dn开始做乘10加次位的操作，依此类推，则可求出二进制数结果。

实验内容：

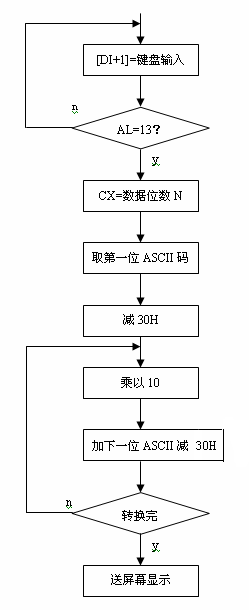
1. 重复从键盘输入不超过5位的十进制数，按回车键结束输入；

2. 将该十进制数转换成二进制数；结果以2进制数的形式显示在屏幕上；

3. 如果输入非数字字符，则报告出错信息，重新输入；

4. 直到输入“Q”或‘q’时程序运行结束。

5. 键盘输入一字符串，以空格结束，统计其中数字字符的个数，在屏幕显示



四、实验数据记录

实现代码：

.MODEL SMALL

.STACK 100H

.DATA

PROMPT DB 'Enter a decimal number (1-99999) or Q/q to quit: $'

ERROR DB 13,10,'Invalid input! Please enter numbers only.$'

BIN\_MSG DB 13,10,'Binary: $'

NEWLINE DB 13,10,'$'

NUMBER DW 0 ; 存储输入的十进制数

BUFFER DB 6 ; 缓冲区大小

DB ? ; 实际输入字符数

DB 6 DUP(?) ; 输入缓冲区

.CODE

MAIN PROC

MOV AX, @DATA

MOV DS, AX

INPUT\_LOOP:

; 显示提示信息

LEA DX, PROMPT

MOV AH, 9

INT 21H

; 读取输入

LEA DX, BUFFER

MOV AH, 0AH

INT 21H

; 检查是否输入Q/q

MOV BL, BUFFER[2]

CMP BL, 'Q'

JE EXIT

CMP BL, 'q'

JE EXIT

; 检查输入长度

MOV BL, BUFFER[1]

CMP BL, 0

JE INVALID\_INPUT

CMP BL, 5

JA INVALID\_INPUT

; 转换为数字

XOR AX, AX ; 清零AX

MOV CX, 0 ; CX用作计数器

MOV CL, BUFFER[1]

LEA SI, BUFFER[2]

CONVERT:

MOV BL, [SI]

CMP BL, '0'

JB INVALID\_INPUT

CMP BL, '9'

JA INVALID\_INPUT

; 乘以10

PUSH CX

MOV CX, 10

MUL CX

POP CX

; 加上当前数字

SUB BL, '0'

XOR BH, BH

ADD AX, BX

INC SI

LOOP CONVERT

MOV NUMBER, AX

; 显示二进制结果

LEA DX, BIN\_MSG

MOV AH, 9

INT 21H

; 转换为二进制并显示

MOV CX, 16 ; 16位二进制

MOV BX, NUMBER

PRINT\_BINARY:

ROL BX, 1 ; 循环左移

JC PRINT\_ONE ; 如果进位标志为1

MOV DL, '0'

JMP PRINT\_BIT

PRINT\_ONE:

MOV DL, '1'

PRINT\_BIT:

MOV AH, 2

INT 21H

LOOP PRINT\_BINARY

; 打印换行

LEA DX, NEWLINE

MOV AH, 9

INT 21H

JMP INPUT\_LOOP

INVALID\_INPUT:

LEA DX, ERROR

MOV AH, 9

INT 21H

LEA DX, NEWLINE

MOV AH, 9

INT 21H

JMP INPUT\_LOOP

EXIT:

MOV AH, 4CH

INT 21H

MAIN ENDP

END MAIN

实验三 基本IO口扩展实验

1. 实验目的

1. 了解TTL芯片扩展简单I/O口的方法。

2. 掌握数据输入输出程序编制的方法。

1. 实验所用仪器

星研集成编译软件、硬件系统（实验箱）

1. 实验基本原理及步骤

实验原理：

74LS244是一种三态输出的8总线缓冲驱动器，无锁存功能，当G为低电平，Ai信号传送到Yi，当为高电平时，Yi处于禁止高阻状态;

  74LS273是一种带清除功能的8D触发器， 1D～8D为数据输入端，1Q～8Q为数据输出端，正脉冲触发，低电平清除，常用作8位地址锁存器。

实验内容：

用74LS244作为输入口，读取开关状态，并将此状态通过74LS273连到发光二极管显示。具体实验内容如下：

1.开关Yi为低电平时对应的发光二极管亮，Yi为高电平时对应的发光二极管灭。

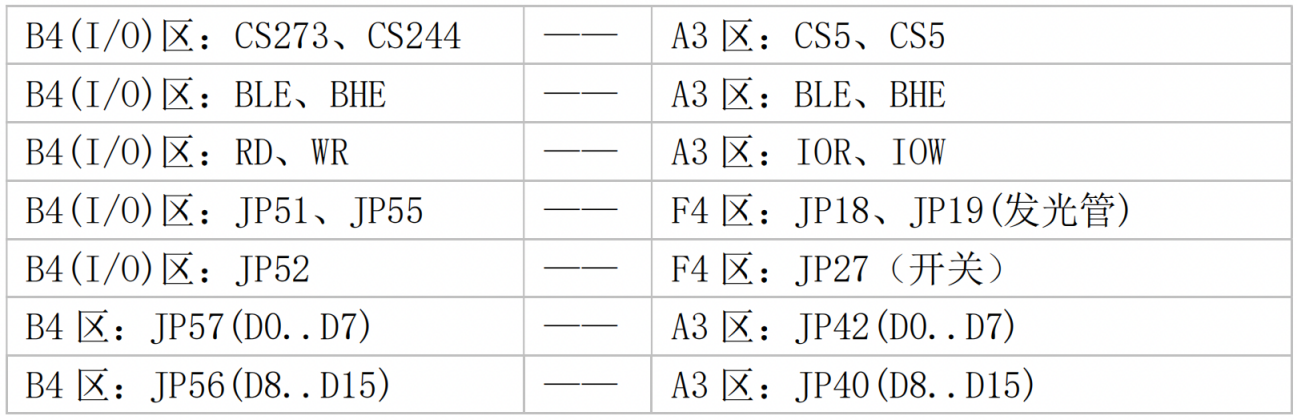
2.当开关Yi全为高电平时，发光二极管Qi从左至右轮流点亮。

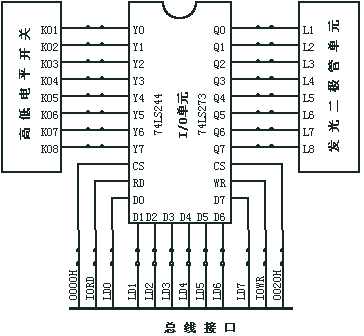
3.当开关Yi全为低电平时，发光二极管Qi从右至左轮流点亮。

4.自主设计控制及显示模式，完成编程调试，演示实验结果。

实验步骤：

1. 按照实验连线图连接：





实验连线图

2.编写实验程序，编译链接，运行程序

3.拨动开关，观察发光二极管的变化。

四、实验数据记录

实现代码：

IO244 EQU 0230H ;74LS244端口地址

IO273 EQU 0230H ;74LS273端口地址

\_STACK SEGMENT STACK

DW 100 DUP(?)

\_STACK ENDS

\_DATA SEGMENT WORD PUBLIC 'DATA'

\_DATA ENDS

CODE SEGMENT

ASSUME CS:CODE, DS:\_DATA, SS:\_STACK

START PROC NEAR

MOV AX, \_DATA

MOV DS, AX

INPUT:

MOV DX, IO244

IN AX, DX

CMP AX, 0FFFFH ;若开关全为低电平

JZ Q1 ;从右往左依次点亮

CMP AX, 0 ;若开关全为高电平

JZ Q2 ;从左往右依次点亮

MOV DX, IO273

OUT DX, AX

JMP INPUT

Q1:

MOV AX, 7FFFH

MOV DX, IO273

R2L:

CALL DELAY ;延时

OUT DX, AX ;送给273,点亮对应的二极管

ROL AX, 1

CMP AX, 7FFFH

JNE R2L ;若相等,说明一轮从右往左已经完成,若不等,则继续循环

JMP INPUT

Q2:

MOV AX, 0FFFEH

MOV DX, IO273

L2R:

CALL DELAY

OUT DX, AX

ROR AX, 1

CMP AX, 0FFFEH

JNE L2R ;若相等,说明一轮从左往右已经完成,若不等,则继续循环

JMP INPUT ;继续读入开关状态

Delay PROC NEAR ;延时子程序

Delay1:

XOR CX,CX ;做一个异或操作将CX清零,仅循环一次

LOOP $

RET

Delay ENDP

START ENDP

CODE ENDS

END START

五、实验结果分析及回答问题

**1、如何实现当开关Yi全为高电平时，发光二极管Qi从左至右轮流点亮、当开关Yi全为低电平时，发光二极管Qi从右至左轮流点亮？**

程序从端口IO244读开关状态至AX，AX为0FFFFH即判开关全低，Q1中先设AX为7FFFH，在R2L循环里延时后输出AX点亮二极管，左移AX改变状态，直至变回7FFFH；AX为0则判开关全高，Q2先设AX为0FFFEH，于L2R循环延时并输出AX点亮二极管，右移AX，直至变回0FFFEH，以此实现对应点亮功能。 **2、如何对 8255 的 PC 口进行位操作？**

首先要对8255进行初始化，设置其工作方式。通过向控制字寄存器写入控制字来确定端口的工作模式等。然后可以使用IN指令从端口读取数据到寄存器，对相应位进行逻辑运算来修改该位的值，再用OUT指令将修改后的数据输出到PC口，从而实现位操作。

实验四 数码转换实验

1. 实验目的

1. 了解可编程并行接口8255的内部结构，

2. 掌握工作方式、初始化编程及应用。

1. 实验所用仪器

星研集成编译软件、硬件系统（实验箱）

1. 实验基本原理及步骤

实验原理：

8255是一个通用可编程并行接口电路。它具有A、B、C三个8位并行口。其中C口也可用作A、B口的联络信号及中断申请信号。通过编程，它可以被设置为基本输入输出、选通输入输出以及双向传送方式。对于C口还具有按位置0、1的功能。

实验内容：

1.流水灯实验：利用8255的A口、B口循环点亮发光二极管。

2.交通灯实验：利用8255的A口模拟交通信号灯。

3.I/O输入输出实验：利用8255的A口读取开关状态，8255的B口把状态送发光

四、实验数据记录

实现代码：

COM\_ADD EQU 0273H

PA\_ADD EQU 0270H

PB\_ADD EQU 0271H

PC\_ADD EQU 0272H

\_STACK SEGMENT STACK

DW 100 DUP(?)

\_STACK ENDS

\_DATA SEGMENT WORD PUBLIC 'DATA'

\_DATA ENDS

CODE SEGMENT

START PROC NEAR

ASSUME CS:CODE, DS:\_DATA, SS:\_STACK

MOV AX,\_DATA ; 将数据段地址加载到AX寄存器

MOV DS,AX ; 将数据段地址传送给数据段寄存器DS

NOP

MOV DX,COM\_ADD ; 将串口地址传送给DX寄存器

MOV AL,82H ; 设置AL寄存器的值为82H

OUT DX,AL

全1：跳转到low1

全0：跳转到high1

其余：直接输出

CMP al, 0FFH ;全1

JZ low1

CMP al, 0 ;全0

JZ high1

MOV DX, PA\_ADD

OUT DX, al

JMP INPUT

low1:

MOV al, 7FH ; 设置AL寄存器的值为7FH

MOV DX, PA\_ADD ; 将并口PA地址传送给DX寄存器

low2:

ROL al, 1 ; 将AL寄存器的值左循环移位1位

OUT DX, al

CALL Delay

CMP al, 7FH

JNE low2

MOV AX, 0FFFFH ; 将AX寄存器的值设为0xFFFF

OUT DX, AX

; 处理输入值为8-15的情况

low3:

MOV al, 7FH

MOV DX, PC\_ADD

low4:

ROL al, 1

OUT DX, al

CALL Delay

CMP al, 7FH

JNE low4

JMP INPUT

high1:

MOV al, 0FEH

MOV DX, PC\_ADD

high2:

ROR al, 1

OUT DX, al

CALL Delay

CMP al, 0FEH

JNE high2

MOV AX, 0FFFFH

OUT DX, AX

; 处理输入值为7-0的情况

high3:

MOV al, 0FEH

MOV DX, PA\_ADD

high4:

ROR al, 1

OUT DX, al

CALL Delay

CMP al, 0FEH

JNE high4