

**课 程 实 验 报 告**

**课程名称： 汇编语言程序设计实验**

**实验名称：** 实验六 输入输出与中断处理程序设计

**实验时间： 2016-4-26，14：30-17：30 实验地点： 南一楼804室102号实验台**

**指导教师： 张勇**

**专业班级：计算机科学与技术201409班**

**学 号： U2014414808 姓 名： 王林**

**同组学生： 无 报告日期： 2016年 4 月 26日**

**原创性声明**

  本人郑重声明：本报告的内容由本人独立完成，有关观点、方法、数据和文献等的引用已经在文中指出。除文中已经注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品或成果，不存在剽窃、抄袭行为。特此声明！

学生签名： 王林

日期： 2016.4.26

成绩评定

|  |  |  |
| --- | --- | --- |
| 实验完成质量得分（70分）（实验步骤清晰详细深入，实验记录真实完整等） | 报告撰写质量得分（30分）（报告规范、完整、通顺、详实等） | 总成绩（100分） |
|  |  |  |

指导教师签字：

日期：

**目录**

[实验目的与要求 1](#_Toc449813799)

[实验过程 1](#_Toc449813800)

[6.1任务1 1](#_Toc449813801)

[6.1.1设计思想及存储单元分配 1](#_Toc449813802)

[6.1.2流程图 2](#_Toc449813803)

[6.1.3源程序 2](#_Toc449813804)

[6.1.4实验步骤 3](#_Toc449813805)

[6.1.5实验记录 4](#_Toc449813806)

[6.2任务2 5](#_Toc449813807)

[6.2.1设计思想及存储单元分配 5](#_Toc449813808)

[6.2.2流程图 5](#_Toc449813809)

[6.2.3源程序 6](#_Toc449813810)

[6.2.4实验步骤 7](#_Toc449813811)

[6.2.5 实验记录 7](#_Toc449813812)

[6.3 任务3 8](#_Toc449813813)

[6.3.1设计思想及存储单元分配 8](#_Toc449813814)

[6.3.2 流程图 8](#_Toc449813815)

[6.3.3 源程序 8](#_Toc449813816)

[6.3.4 实验步骤 10](#_Toc449813817)

[6.3.5 实验记录 10](#_Toc449813818)

[体会 10](#_Toc449813819)

[参考文献 11](#_Toc449813820)

实验目的与要求

(1) 掌握中断矢量表的概念；

(2） 熟悉I/O访问，BIOS功能调用方法；

(3) 掌握实方式下中断处理程序的编制与调试方法。

实验内容

任务1：用三种方式获取中断类型码10H对应的中断处理程序的入口地址。

要求：首先要进入虚拟机状态，然后

（1） 直接运行调试工具（TD.EXE），观察中断矢量表中的信息。

（2） 编写程序，用 DOS系统功能调用方式获取，观察功能调用相应的出口参数与“（1）”看到的结果是否相同 （使用TD观看出口参数即可）。

（3） 编写程序，直接读取相应内存单元，观察读到的数据与“（1）”看到的结果是否相同 （使用TD观看程序的执行结果即可）。

任务2：编写一个接管键盘中断的中断服务程序并驻留内存，要求在程序返回DOS操作系统后，键盘的按键A变成了按键B，而B变成了A。

要求：

（1）在 DOS虚拟机或DOS窗口下执行程序，中断服务程序驻留内存。

（2）在DOS命令行下按键A，屏幕显示为B，按B时屏幕显示为A。执行TD，在代码区输入指令“MOV AX,0”看是否能发生变化。

（3）选作：另外编写一个中断服务程序的卸载程序，将键盘中断服务程序恢复到原来的状态（也就是还原中断矢量表的信息）。

任务3：读取CMOS内指定单元的信息，按照16进制形式显示在屏幕上。

要求：

(1)先输入待读取的CMOS内部单元的地址编号（可以只处理编号小于10的地址单元）。再使用IN/OUT指令，读取CMOS内的指定单元的信息。

（2） 将读取的信息用16进制的形式显示在屏幕上。若是时间信息，可以人工判断一下是否正确。

实验过程

6.1任务1

用三种方式获取中断类型码10H对应的中断处理程序的入口地址。

6.1.1设计思想及存储单元分配

分别采用三种方式来获取中断类型码10H对应的处理程序入口地址，1、直接在TD窗口观察10H入口地址，2、编写一段程序，系统调用10H对应的程序，通过TD单步运行来观察，3、编写一段小程序，直接获取对应的地址值。

6.1.2流程图

无

6.1.3源程序

1、在TD中直接观察，无源代码

2、编写小程序系统调用!0H

【ins12.c】

.386

data segment use16

data ends

stack segment use16 stack

db 200 dup(0)

stack ends

code segment use16

assume cs:code,ds:data,ss:stack

start:

mov ax, data

mov ds, ax

mov ax, 3510h

int 21h

mov ah, 4ch

int 21h

code ends

end start

3、编写程序直接获取

【ins13.c】

.386

data segment use16

data ends

stack segment use16 stack

db 200 dup(0)

stack ends

code segment use16

assume cs:code,ds:data,ss:stack

start:

mov ax, 0

mov ds, ax

mov ax, ds:[10h\*4]

mov es, ds:[10h\*4+2]

mov ah, 4ch

int 21h

code ends

end start

6.1.4实验步骤

1、在TD中观察中断向量表

1. 直接运行TD
2. 使数据区在当前区域
3. 快捷键【ctrl+g】，输入【0:40H】,回车，跳到对应中断向量表10H对应的地方

读取连续的4个字节，前两个为偏移地址，后两个为段值（小端方式存储）

1. 然后到代码区
2. 快捷键【ctrl+g】，输入刚才的到到的值,回车，得到对应的程序

2、编写小程序系统调用10H

1. 按要求编写程序
2. 通过Vmaware 安装DOS系统虚拟机，对DOS系统进行映射，将源程序文件复制进

去

1. 编译、链接、生成可执行程序，调试BUG直到无错
2. 通过TD反汇编，单步执行
3. 观察执行完”int 21h”后的寄存器”ax”、“es”的值是否与（1）中一致

3、编写程序直接获取

1. 按要求编写程序
2. 通过Vmaware 安装DOS系统虚拟机，对DOS系统进行映射，将源程序文件复制进

去

1. 编译、链接、生成可执行程序，调试BUG直到无错
2. 通过TD反汇编，单步执行
3. 观察执行完” mov es, ds:[10h\*4+2]“后寄存器”ax“、”es”的值是否与(1)、（2）中一致

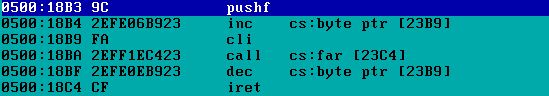
6.1.5实验记录

1、实验环境条件：虚拟机DOS系统，内存:16M，处理器: 1.命令行窗口；Sublime Text 2；MASM.EXE 6.0； LINK.EXE 5.2; TD.EXE 5.0。

2、在第一个小任务中，比较简单，直接运行td，直接找到中断向量表。但是输入的地址要正确，是段值加上偏移地址。找到对应地址后，注意是小端模式，要正确读出对应地址值。

****

**图 6-1 中断向量表观察**

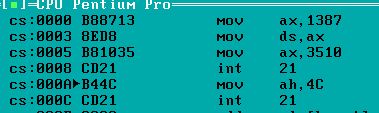
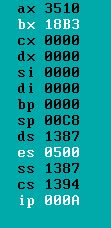
****

**图 6-2 中断向量表观察**

3、中断类型码为10H，则偏移地址为【10H\*4】，段值为【10H\*4+2】.

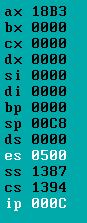
4、在DOS系统中，不能通过鼠标操作，可以通过【Tab】键来改变当前区。

5、第二个小任务中，要编写小程序，通过系统调用，然后再通过TD反汇编观察，在TD直接修改程序是BIOS调用。编程中，“mov ah， 35h” ,“mov al，10h”,通过35h的功能取中断类型信息，al为中断类型编号。

**图 6-3 中断向量表观察**

6、第三个小任务中，直接编写程序访问内存，获取对应地址值信息，观察与(1)、(2)中是否一致。直接对ax和es赋值。

**图 6-4 中断向量表观察**

6.2任务2

编写一个接管键盘中断的中断服务程序并驻留内存，要求在程序返回DOS操作系统后，键盘的按键A变成了按键B，而B变成了A。

6.2.1设计思想及存储单元分配

将原中断地址信息保存在OLD\_INT16H,为DB类型，占4个字节。然后是新的16H。

先取出原中断信息，保存然后将编写的新16H的中断信息保存到对应地址，进行中断信息的替换。

6.2.2流程图



**图 6-5 任务二流程图 新16H中断信息**



**图 6-6 任务二流程图 程序初始化**

6.2.3源程序

【ins2.c】

.386

code segment use16

assume cs:code,ss:stack

OLD\_INT DW ?,?

;判断INT 16H的功能号：00H和10H

NEW16H: cmp ah, 0h

je loap0

cmp ah, 10h

je loap0

jmp dword ptr OLD\_INT

;比较知道输入的是a还是b

loap0: pushf

call dword ptr OLD\_INT

cmp al, 'A'

je newb

cmp al, 'a'

je newb

cmp al, 'B'

je newa

cmp al, 'b'

je newa

iret

;输入a转化为b

newa: mov al, 41h

iret

;输入b转化为a

newb: mov al, 42h

iret

start: xor ax, ax

mov ds, ax

mov ax, ds:[16h\*4]

mov OLD\_INT, ax

mov ax, ds:[16h\*4+2]

mov OLD\_INT+2, ax

CLI

;传送新的中断矢量表

mov word ptr ds:[16h\*4], offset NEW16H

mov ds:[16h\*4+2], cs

STI

mov dx, offset start+15

;驻留内存处理

mov cl, 4

shr dx, cl

add dx, 10h

;退出码为0

mov al, 0

mov ah, 31h

int 21h

code ends

stack segment stack use16

db 200 dup(0)

stack ends

end start

6.2.4实验步骤

1、按照要求编写程序

2、编译、链接、生成可执行程序，调试至没有BUG

3、运行程序，观察是否达到要求。

6.2.5 实验记录

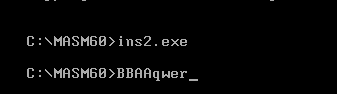
1、实验环境条件：虚拟机DOS系统，内存:16M，处理器: 1.命令行窗口；Sublime Text 2；MASM.EXE 6.0； LINK.EXE 5.2; TD.EXE 5.0。

2、进入原中断服务程序可以通过【CALL】、【JMP】，本次试验中，采用JMP

3、为避免键盘失灵，可以先通过TD调试，观察对应功能入口和出口寄存器的值是否发生改变

4、同时打开另一个窗口，A、B的值没发生改变。

5、运行效果如图：



则如图所示，当按下a键时显示是并、键，按下b键，显示a键，其他键位不变。

6.3 任务3

读取CMOS内指定单元的信息，按照16进制形式显示在屏幕上。

6.3.1设计思想及存储单元分配

首先1号系统功能调用，读入用户输入，作为信息的偏移地址。然后读取信息，并打印。

6.3.2 流程图

无

6.3.3 源程序

【cmos.asm】

.386

data segment use16

buf db 2

data ends

stack segment use16 stack

db 200 dup(0)

stack ends

code segment use16

assume ds:data,ss:stack,cs:code

start:

mov ax, data

mov ds, ax

lopa: mov ah, 1 ;读入输入

int 21h

sub al, 30h ;转化成十进制0~9

out 70h, al ;取出信息

in al, 71h

mov ah, al

and al, 0fh

shr ah, 4

add ax, 3030h ;转化为ascll码

xchg ah, al

mov bx, ax

lea si, buf

mov word ptr [si+1], bx

mov dl, 0ah ;输出换行

mov ah, 2

int 21h

mov dl, 0dh

mov ah, 2

int 21h

mov dl, bl ;输出信息

mov ah, 2

int 21h

mov dl, bh

mov ah, 2

int 21h

mov dl, 0ah ;输出换行

mov ah, 2

int 21h

mov dl, 0dh

mov ah, 2

int 21h

jmp lopa ;循环

mov ah, 4ch

int 21h

code ends

end start

6.3.4 实验步骤

1、按照要求编写程序

2、然后编译、链接、生成可执行程序，知道无错。

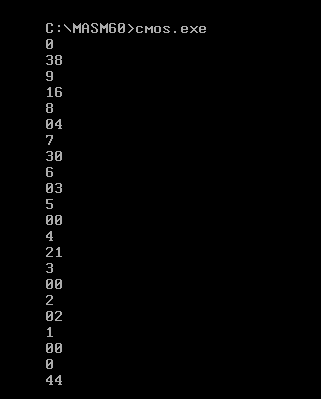
3、依次输入‘9’~‘0’，然后观察屏幕输出信息，验证是否正确。

6.3.5 实验记录

1、实验环境条件：虚拟机DOS系统，内存:16M，处理器: 1.命令行窗口；Sublime Text 2；MASM.EXE 6.0； LINK.EXE 5.2; TD.EXE 5.0。

2、取出的信息按照BCD编码进行压缩，分为两部分，分别存储在AL，AH。比如说当前小时信息是19，则1存储在AH，为0x01,9存储在AL,为0x09。

3、运行效果如图：



如图，依次输入‘9’~‘0’，则屏幕上应当一次输出当前的时间信息，包括年、月、日、星期、小时、分钟、秒。

体会

通过本次上机，熟悉了对中断的处理。了解中断信息的运行机制。

本次上机的具体收获有：（1）通过编写代码或系统调用或TD查看来查看中断向量表（2）通过改变系统的中断向量表来更加理解中断。（3）了解了CMOS李存储的信息

同时，对中断信息程序的执行机理还不够熟悉，需要进一步学习。

# 参考文献

[序号] 作者.书名.版本(第×版).译者.出版地:出版者,出版年：起页-止页

[序号] URL: 网络地址，如URL:http://www.cbs.dtu.dk/services