****

保护模式中断实验

班级：07111505

小组成员：1120151825郭佳楠

1120151828兰天

1120151839徐宇恒

1120151858 王阁元

目录

**第一章 实验目的**1

**第二章 实验内容**1

**第三章 实验环境**1

**第四章 程序素材解析**1

4.1 程序与系统的关系2

4.2 预处理过程2

4.3 中断处理过程2

**第五章 程序素材修改及方式**3

5.1 填充IDT3

5.2 填充键盘中断处理程序4

5.3 开中断4

5.4 设置循环结束条件4

**第六章 实验心得**5

程序语言认知实验

**一、实验目的**

通过实验，了解保护模式下，中断的处理流程。掌握保护模式下可编程中断控制器8259A的编程方式，并为指定端口添加中断处理程序。

**二、实验内容**

① 初始化8259芯片，并设置键盘中断向量号为21H。

② 在保护模式32位代码段中，添加指定键盘驱动程序。

③ 在保护模式下，按Esc键后，返回实模式，并结束程序。

键盘驱动程序：

|  |
| --- |
| KeyboardHandler:  in al, 60h  cmp al, 1  jnz notEsc  ; 如果出发ESC则将ecs标志为1  mov ecx, 1  notEsc:  ; 屏幕第 1 行, 第 70 列，按键就将ascll加1  inc byte ptr gs:[((80 \* 1 + 70) \* 2)]  mov al, 20h  ;发送EOI  out 20h, al  Iretd |

**三、实验环境**

DosBox-0.74、MASM、LINK、EDIT、debug

**四、程序素材解析**

**4.1 程序与系统的关系**

程序并没有真正调用系统的描述符和选择符等结构，而是在程序中建立虚拟的STRUC结构，模拟系统TSS描述符、任务状态段TSS、门描述符、全局描述符表寄存器GDTR、全局描述符表GDT以及中断描述符表IDT。并将自己建立的IDT加载在系统上，通过8259中断调用IDT，进而实现调用自己的中断服务程序。

**4.2 预处理过程**

① 将中断服务程序写入同一个段，并加载至内存。

② 实模式下填充GDT以及IDT表。由于中断处理程序位于同一段，因此GDT内秩序添加此段信息即可。设计IDT表，分别将键盘中断处理程序以及时钟中断处理程序的中断向量设置为0x21以及0x20。

③ 初始化可编程中断控制器8259，设置为两片级联工作方式，并仅开启IR0和IR1。

**4.3 中断处理过程**

偏移

TSS基址

TSS限长

TSS选择符

门选择符

GDT

JMP/CALL

IDT

任务门

X

Y

TSS段描述符

新任务TSS

LDTR

**五、程序素材修改及方式**

**5.1 填充IDT**

|  |
| --- |
| ; IDT  ALIGN 32  IDT LABEL BYTE  IDT\_00\_1F Gate 32 dup (<offset SpuriousHandler, Code32\_Sel, 0, DA\_386IGate, 0>)  IDT\_20 Gate 1 dup (<offset ClockHandler, Code32\_Sel, 0, DA\_386IGate, 0>)  ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*在0x21-0x22间添加键盘驱动程序门描述符\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  IDT\_21 Gate 1 dup (<offset KeyboardHandler, Code32\_Sel, 0, DA\_386IGate, 0>)  ;95修改为94保持80H中断  IDT\_21\_7F Gate 94 dup (<offset SpuriousHandler, Code32\_Sel, 0, DA\_386IGate, 0>)  ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  IDT\_80 Gate 1 dup (<offset UserIntHandler, Code32\_Sel, 0, DA\_386IGate, 0>) |

在中断描述符表地址0x21处添加门描述符，此任务门偏移为键盘中断处理程序在自身段内的偏移，段选择符指向中断处理程序所在段。

**5.2 填充键盘中断处理程序**

|  |
| --- |
| KeyboardHandler:  in al, 60h  cmp al, 1  jnz notEsc  ; 如果出发ESC则将ecs标志为1  mov ecx, 1  notEsc:  ; 屏幕第 1 行, 第 70 列，按键就将ascll加1  inc byte ptr gs:[((80 \* 1 + 70) \* 2)]  mov al, 20h  ;发送EOI  out 20h, al  Iretd |

实验中，由于并未修改目标数据段，因此填充中断处理程序时须注意将键盘中断处理程序与原有的时钟中断处理程序等放置在同一段内。

**5.3 开中断**

|  |
| --- |
| mov al, 11111100b ; 仅仅开启定时器中断 \*\*\*\*\*\*\*修改为键盘开中断\*\*\*\*\*\*\*  out 021h, al ; 主8259, OCW1.  call io\_delay |

在中断控制器8259主片上为键盘中断开中断。

**5.4 设置循环结束条件**

|  |
| --- |
| ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*修改循环结束条件\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  xor ecx,ecx;  WaitLoop:  ;cmp ebx, 100;仅循环100次  ;jb WaitLoop  cmp ecx, 1  jnz WaitLoop  ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |

键盘中断服务程序表明，若用户点击Esc键则将ecx寄存器设置为1，在此判断若ecx为1，则停止循环。

**六 实验心得**

**6.1 郭佳楠：**

首先，实验中发现一个有趣的问题，每次点击键盘键盘时，对应输出的ASCII并不是加1，而是加2。也就是说每次点击键盘触发两次键盘中断。通过查询得知，按下键盘和弹起键盘均会出发一次键盘中断。

其次，对实验素材中一段内容不是十分理解。实验素材中为什么设置了输出‘！’的中断处理程序（SpuriousHandler），又在主程序中直接输出‘！’，这样的设置貌似没有什么意义。我的理解是，此中断处理程序用于对IDT表初始化的检测，或者用于中断屏蔽的检测，但是在主程序中直接输出后，使得程序并不能完成检测。

最后，谈谈本次实验的收获。实验过程中，验证了书上的中断处理过程，使我对此有了更深的理解，同时实验中尝试了中断屏蔽等处理方式，感觉还挺有趣的。对于我来说实验的主要难点在于IDT与ICW1中断向量号的对应，可能是上课走神了，实验前我并没有向量号对应的概念。

**6.2 兰天：**

首先，通过本次实验对保护模式下的中断方式有了一个更加清晰的了解，对程序代码中 GDT, IDT 表的构造方式页觉得非常的有意思。老师提供的代码中，简易的模拟了保护模式下的中断的必要的系统表。

通过运行这些虚拟的表实现运行在程序上的保护模式是一件非常有意思的事情。

其次，并且在进行实验的时候，和队友发现了一个有趣的现象，在我们敲击键盘的时候发现ASCII 的输出值并不是加1，而是加2，经过一番查找资料发现，因为键盘中断并不是敲击一次输出一个电平，而是按下按键输出一个电平，松开按键输出一个电平。整个过程实现了完整的键盘中断的触发机制。

最后，在老师的提供的代码框架下成功的实现了标准的保护模式下的中断实验，在实验的过程中也意识到了很多的不足和学习中的缺陷。通过实验这个过程有效的进行了补充和学习。这样的实验非常有必要，收获颇多。

**6.3 徐宇恒：**

本次实验是在我们自己的电脑上通过DOS-BOX模拟出80386和8259的中断控制。通过该实验，我对课本中第三章、第七章的理论知识有了更加深刻的理解。该实验的核心是对中断描述符表的操作。如何设置中断屏蔽开关，如何向中断描述符表中添加新的描述符，实践这些理论知识对我们来说是个不小的挑战。这也使我们对ICW和OCW指令自有了更加深刻的了解。

在实验过程中我们还发现了一个现象。当有键盘输入时，ASCLL表的输出值并不是加1而是加2.这是因为键按下与弹起分别是两个电平。

通过这次实验，我发现如何把课上的理论知识运用到实践中去对我们来说还是一个不小的难题。本次实践课程给了我这样一个动手实践的机会，收获颇丰。

**6.4 王阁元：**

本次实验主要是了解保护模式下的中断处理机制，主要的理论基础与键中断的开启、8259芯片编程，中断服务程序的中断门有关。本次实验算是对课本第3，7章的理论知识进行了综合，最后实现了键盘中断。回顾整个实验过程，一开始设置键盘中断的开启，然后编写中断服务程序，最后添加中断描述符表。整个过程都是基于已给的源代码的提示，然后添加进去的。但是中断描述符表的添加遇到了小问题，最后考虑到IDT表的顺序存储，遂解决问题。