南京航空航天大学 计算机科学与技术系学院 计算机组成原理 课程实验

学号：161630220

姓名：赵维康

PA4- 虚实交错的魔法：分时多任务

来自外部的声音

添加时钟中断

思考题：灾难性的后果(这个问题有点难度)

假设硬件把中断信息固定保存在内存地址 0x1000 的位置，AM 也总是从这里开始构造 trap frame。如果发生了中断嵌套，将会发生什么样的灾难性后果?这一灾难性的后果将会以什么样的形式表现出来?如果你觉得毫无头绪，你可以用纸笔模拟中断处理的过程。

答：我们知道发生中断嵌套的时候，第一次中断保存的现场信息将会被优先级高的中断处理过程所覆盖，所以恢复第一次中断保存的现场信息将要付出很大的代价，甚至是难以修复。

首先进入 nemu/include/cpu/reg.h，在 cpu 结构体中添加一个 bool 成员 INTR，如图

```
    uint16_t cs;
    union{
        rtlreg_t eflags_init;
        struct{
            unsigned int CF:1;
            unsigned int ZF:1;
            unsigned int SF:1;
            unsigned int IF:1;
            unsigned int OF:1;
                };
          }eflags;
    struct {
            uint16_t limit;
            uint32_t base;
           }idtr;

    CR0 cr0;

    CR3 cr3;

    bool INTR;
} CPU_state;
```

下面进入 nemu/src/cpu/intr.c，在 dev_raise_intr()中将 INTR 引脚设置为高电平，如图

```
void dev_raise_intr() {
 cpu.INTR = true;
}
```

下面进入 nemu/src/cpu/exec/exec.c，在 exec_wrapper()的末尾添加轮询 INTR 引脚的代码，

每次执行完一条指令就查看是否有 硬件中断到来，如图

```
#define TIMER_IRQ 0x20

extern void raise_intr(uint8_t NO, vaddr_t ret_addr);

if (cpu.INTR && cpu.eflags.IF) {

  cpu.INTR = false;

  raise_intr(TIMER_IRQ, cpu.eip);

  update_eip();

}
```

下面进入 nemu/src/cpu/intr.c，修改 raise_intr()中的代码，在保存 EFLAGS 寄存器后，将其 IF 位置为 0，让处理器进入关中断状态，如图

```
  rtl_push(&cpu.eflags.eflags_init);

  t0 = cpu.cs;

  rtl_push(&t0);

  rtl_push(&ret_addr);


  cpu.eflags.IF = 0;
}
```

下面进入 nexus-am/am/arch/x86-nemu/src/asye.c，在 irq_handle()函数中添加时钟中断的支持,，将时钟中断打包成_EVENT_IRQ_TIME 事件，如图

```
_RegSet* irq_handle(_RegSet *tf) {
  _RegSet *next = tf;
  if (H) {
    _Event ev;
    switch (tf->irq) {
      case 0x80: ev.event = _EVENT_SYSCALL; break;
      case 0x81: ev.event = _EVENT_TRAP;break;
      case 0x20: ev.event = _EVENT_IRQ_TIME; break;
      default: ev.event = _EVENT_ERROR; break;
    }

    next = H(ev, tf);
    if (next == NULL) {
      next = tf;
    }
  }

  return next;
}
```

同时在_asye_init()函数中添加定时钟中断的中断号 0x20（32），如图

```
void _asye_init(_RegSet*(*h)(_Event, _RegSet*)) {
  // initialize IDT
  for (unsigned int i = 0; i < NR_IRQ; i ++) {
    idt[i] = GATE(STS_TG32, KSEL(SEG_KCODE), vecnull, DPL_KERN);
  }

  // -------------------- system call --------------------------
  idt[0x80] = GATE(STS_TG32, KSEL(SEG_KCODE), vecsys, DPL_USER);
  idt[0x81] = GATE(STS_IG32, KSEL(SEG_KCODE), vectrap, DPL_USER);
  idt[0x20] = GATE(STS_TG32, KSEL(SEG_KCODE), vectime, DPL_USER);

  set_idt(idt, sizeof(idt));

  // register event handler
  H = h;
}
```

下面进入 nanos-lite/src/irq.c，Nanos-lite 收到 _EVENT_IRQ_TIME 事件之后，调用 schedule() 进行进程调度，并用 Log()输出一句话，如图

```
static _RegSet* do_event(_Event e, _RegSet* r) {
  _RegSet *ret = NULL;
  switch (e.event) {
    case _EVENT_SYSCALL: do_syscall(r);break;
    case _EVENT_TRAP:
    case _EVENT_IRQ_TIME: Log("irq timer\n");ret = schedule(r); break;
    default: panic("Unhandled event ID = %d", e.event);
  }
    return ret;
}
```

下面进入 nexus-am/am/arch/x86-nemu/src/trap.S，做如图的修改

```
#----|-------entry-------|-errorcode-|---irq id---|---handler---|
.globl vecsys;    vecsys:  pushl $0;  pushl $0x80; jmp asm_trap
.globl vecnull;  vecnull:  pushl $0;  pushl   $-1; jmp asm_trap
.globl vectrap;  vectrap:  pushl $0;  pushl $0x81; jmp asm_trap
.globl vectime;  vectime:  pushl $0;  pushl $0x20; jmp asm_trap
```

下面进入 nexus-am/am/arch/x86-nemu/src/pte.c， 在_umake()中设置正确的 EFLAGS，如图

```
_RegSet *_umake(_Protect *p, _Area ustack, _Area kstack, void *entry, char *const argv[], char
*const envp[]) {

struct { _RegSet *tf; } *pcb = ustack.start;

 uint32_t *stack = (uint32_t *)(ustack.end - 4);

  // stack frame of _start()

  for (int i = 0; i < 3; i++)

    *stack-- = 0;

  pcb->tf = (void *)(stack - sizeof(_RegSet));

  pcb->tf->eflags = 0x2 | (1 << 9);

  pcb->tf->cs = 8;

  pcb->tf->eip = (uintptr_t)entry;

  return pcb->tf;

}
```

下面执行 make update、make run 命令，如图

```
root@zhaoweikang:/home/zhaoweikang/ics2017/nanos-lite# make update
Building nanos-lite [x86-nemu]
make -s -C /home/zhaoweikang/ics2017/navy-apps ISA=x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/nterm/build/nterm-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/lua/build/lua-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/init/build/init-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/litenes/build/litenes-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/pal/build/pal-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/nwm/build/nwm-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/events/build/events-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/text/build/text-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/bmp/build/bmptest-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/dummy/build/dummy-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/hello/build/hello-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/videotest/build/videotest-x86
root@zhaoweikang:/home/zhaoweikang/ics2017/nanos-lite# make run
Building nanos-lite [x86-nemu]
+ AS src/initrd.S
+ CC src/fs.c
make[1]: Entering directory '/home/zhaoweikang/ics2017/nexus-am'
make[2]: Entering directory '/home/zhaoweikang/ics2017/nexus-am/am'
Building am [x86-nemu]

make -s -C /home/zhaoweikang/ics2017/navy-apps ISA=x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/nterm/build/nterm-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/lua/build/lua-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/init/build/init-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/litenes/build/litenes-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/pal/build/pal-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/apps/nwm/build/nwm-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/events/build/events-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/text/build/text-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/bmp/build/bmptest-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/dummy/build/dummy-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/hello/build/hello-x86
+ LD /home/zhaoweikang/ics2017/navy-apps/tests/videotest/build/videotest-x86
root@zhaoweikang:/home/zhaoweikang/ics2017/nanos-lite# make run
Building nanos-lite [x86-nemu]
+ AS src/initrd.S
+ CC src/fs.c
make[1]: Entering directory '/home/zhaoweikang/ics2017/nexus-am'
make[2]: Entering directory '/home/zhaoweikang/ics2017/nexus-am/am'
Building am [x86-nemu]
make[2]: Nothing to be done for 'archive'.
make[2]: Leaving directory '/home/zhaoweikang/ics2017/nexus-am/am'
```

```
make[1]: Entering directory '/home/zhaoweikang/ics2017/nexus-am/libs/klib'
make[1]: *** 没有指明目标并且找不到 makefile。 停止。
make[1]: Leaving directory '/home/zhaoweikang/ics2017/nexus-am/libs/klib'
/home/zhaoweikang/ics2017/nexus-am/Makefile.compile:86: recipe for target 'klib'
 failed
make: [klib] Error 2 (ignored)
make[1]: Entering directory '/home/zhaoweikang/ics2017/nemu'
+ CC src/cpu/intr.c
fatal: ..: '..' 在仓库之外
Makefile:41: recipe for target 'build/nemu' failed
make[1]: [build/nemu] Error 128 (ignored)
+ LD build/nemu
fatal: ..: '..' 在仓库之外
Makefile:46: recipe for target 'run' failed
make[1]: [run] Error 128 (ignored)
./build/nemu -l /home/zhaoweikang/ics2017/nanos-lite/build/nemu-log.txt /home/zh
aoweikang/ics2017/nanos-lite/build/nanos-lite-x86-nemu.bin
[src/monitor/monitor.c,65,load_img] The image is /home/zhaoweikang/ics2017/nanos
-lite/build/nanos-lite-x86-nemu.bin
Welcome to NEMU!
[src/monitor/monitor.c,30,welcome] Build time: 22:03:10, Jun 30 2018
For help, type "help"
(nemu) c
 [src/mm.c,81,init_mm] free physical pages starting from 0x1d9b000
[src/main.c,41,main] 'Hello World!' from Nanos-lite
[src/main.c,43,main] Build time: 20:21:11, Jun 30 2018
[src/ramdisk.c,26,init_ramdisk] ramdisk info: start = 0x102268, end = 0x1d54f05,
 size = 29699229 bytes
[src/main.c,57,main] Initializing interrupt/exception handler...
[src/loader.c,41,loader] loaded: [52]/bin/pal size:1400608
[src/loader.c,41,loader] loaded: [55]/bin/hello size:21312
[src/irq.c,10,do_event] irq timer

game start!
VIDEO_Init success
loading fbp.mkf
loading mgo.mkf
loading ball.mkf
loading data.mkf
loading f.mkf
loading fire.mkf
loading rgm.mkf
loading sss.mkf
loading desc.dat
PAL_InitGolbals success
PAL_InitFont success
PAL_InitUI success
PAL_InitText success
PAL_InitInput success
PAL_InitResources success
```

在 Nanos-lite 收到 _EVENT_IRQ_TIME 事件后用 Log()输出了一句话，证明时钟中断确实在工作。

git log 记录

```
zhaoweikang@zhaoweikang:~/ics2017/nanos-lite$ sudo git status
[sudo] zhaoweikang 的密码：
位于分支 pa4
尚未暂存以备提交的变更：
  （使用 "git add <文件>..." 更新要提交的内容）
  （使用 "git checkout -- <文件>..." 丢弃工作区的改动）

        修改：      src/irq.c
        修改：      src/proc.c
        修改：      ../nemu/include/cpu/decode.h
        修改：      ../nemu/include/cpu/reg.h
        修改：      ../nemu/include/cpu/rtl.h
        修改：      ../nemu/src/cpu/decode/decode.c
        修改：      ../nemu/src/cpu/decode/modrm.c
        修改：      ../nemu/src/cpu/exec/all-instr.h
        修改：      ../nemu/src/cpu/exec/arith.c
        修改：      ../nemu/src/cpu/exec/exec.c
        修改：      ../nemu/src/cpu/exec/system.c
        修改：      ../nemu/src/cpu/intr.c
        修改：      ../nemu/src/memory/memory.c
        修改：      ../nemu/src/monitor/monitor.c
```

```
        修改：      ../nexus-am/am/arch/x86-nemu/src/asye.c
        修改：      ../nexus-am/am/arch/x86-nemu/src/pte.c
        修改：      ../nexus-am/am/arch/x86-nemu/src/trap.S

未跟踪的文件:
  （使用 "git add <文件>..." 以包含要提交的内容）

        ../Makefile

修改尚未加入提交（使用 "git add" 和/或 "git commit -a"）
zhaoweikang@zhaoweikang:~/ics2017/nanos-lite$ sudo git add .
zhaoweikang@zhaoweikang:~/ics2017/nanos-lite$ sudo git commit --allow-empty
[pa4 927a75a] fix bug for pa4.3
 2 files changed, 4 insertions(+), 10 deletions(-)
zhaoweikang@zhaoweikang:~/ics2017/nanos-lite$ sudo git log
commit 927a75a97cc76c6e3a236b3fd2dfc19962de4498
Author: 161630220-Zhao Weikang <2875206963@qq.com>
Date:   Sat Jun 30 22:21:38 2018 +0800

    fix bug for pa4.3

commit 927a95712644f32cf904816ddb9bf26659c74ef2
Author: 161630220-Zhao Weikang <2875206963@qq.com>
Date:   Sat Jun 30 17:05:22 2018 +0800

    fix bug for pa4.2

commit 9144289774659c90d26db1783e8d588ae47d35d3
Author: 161630220-Zhao Weikang <2875206963@qq.com>
Date:   Sun Jun 17 16:56:01 2018 +0800

    fix bug for pa4.1

commit 3a55adfe947ada0146f93dc816c4587e0d92b606
Author: 161630220-Zhao Weikang <2875206963@qq.com>
Date:   Sun Jun 10 08:49:55 2018 +0800

    before starting pa4
```