Mecanismo de Interrupção

Mecanismo de Interrupção

Constitui a base de operação de um sistema de multiprogramação.

É um sinal de hardware que informa a ocorrência de um evento no sistema, tal como o término de uma operação de E/S.

Provoca uma mudança no fluxo de controle, o qual é transferido para uma rotina de tratamento da interrupção correspondente.

Interrupções de software: trap e SVC.

Inibição de Interrupções

O núcleo (*kernel*) do S.O. algumas vezes previne a ocorrência de interrupções durante atividades críticas, que poderiam resultar em dados corrompidos se estas fossem permitidas (ex: quando manipulando listas endadeadas). Assim, certas instruções (ditas *privilegiadas*) permitem colocar o processador em um certo nível de execução em que ele mascara (inibe) certos valores de interrupção.

Níveis Típicos de Interrupção

Erros de Máquina

Relógio

Disco

Interface de Rede

Terminal

Interrupção de SW



Modos de Operação (1)

O compartilhamento de recursos requer do S.O. garantias de que um programa com comportamento incorreto (deliberadamente ou não) possa causar danos ao sistema ou aos outros programas.

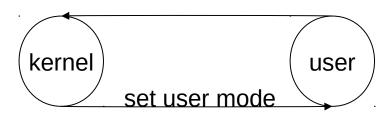
Nesse sentido, o *hardware* fornece suporte para que a execução de processos num dado sistema operacional se diferencie pelo menos entre dois modos de operação: *modo usuário* ("*user mode*") e *modo supervisor* ("*kernel mode*").

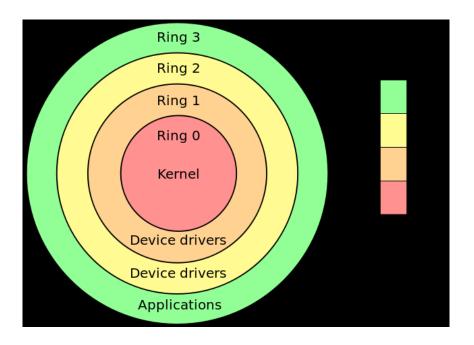
Para isso, um "Mode bit" é adicionado ao hardware do computador para indicar o modo corrente: supervisor (0) ou usuário (1).

Processos executando em modo usuário podem acessar suas próprias instruções e dados mas não as instruções e dados do *kernel* (i.e., do núcleo do sistema operacional) ou mesmo de outros processos.

Modos de Operação (2)

Interrupt/fault





Modos de Operação (3)

Processos em modo supervisor não possuem esta limitação, podendo acessar endereços de usuário e do próprio *kernel* do sistema.

Quando um processo executa uma chamada de sistema, isto é, quando ele faz uma SVC ("supervisor call"), o modo de execução muda de usuário para supervisor. O sistema operacional sempre roda no estado supervisor.

Com o sistema no estado supervisor:

- interrupções podem ser inibidas ou novamente habilitadas;
- As proteções estão desabilitadas.
- Qualquer instrução pode ser executada.

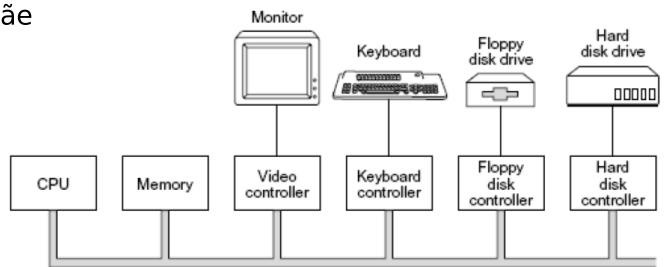
Instruções privilegiadas só podem ser executadas em modo supervisor. Sua tentativa de execução em modo usuário resulta em erro.

Gerência de Entrada e Saída

Dispositivos de Entrada e Saída (1)

- Constituídos de 2 partes:
 - Mecânica
 - Eletrônica Controladora ou Adaptadora
- Controladora

 Placa ligada a um slot livre, ou inserida diretamente na placa-mãe
Monitor

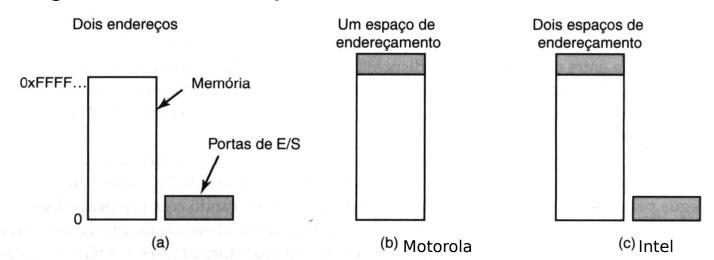


Dispositivos de Entrada e Saída (2)

- Controladora (cont.)
 - Implementa um conjunto de funções básicas para o dispositivo
 - Também tratar o acesso do dispositivo ao barramento
 - Tarefas típicas
 - Converter fluxo serial de bits em bloco de bytes
 - Correção de erros
 - Tornar o bloco disponível para ser copiado para a memória principal
 - Possui registradores usados para comunicar com o SO
 - Seqüência e/ou valores armazenados nestes registradores determina a operação sendo realizada
 - Tipicamente tem memória interna (buffer)
 - Compatibilizar velocidades

Mapeamento de Endereços

- Mapeamento em espaço de entrada e saída
 - Instruções especiais da CPU para E/S
 - Opcodes separados (instruções IN e OUT)
 - Cada registrador está associado a uma porta de E/S
- Mapeamento em espaço de memória
 - Associa um espaço de endereços de memória aos registradores
 - Leitura/Escrita neste espaço realizam a operação sobre os registradores (instruções load/store)

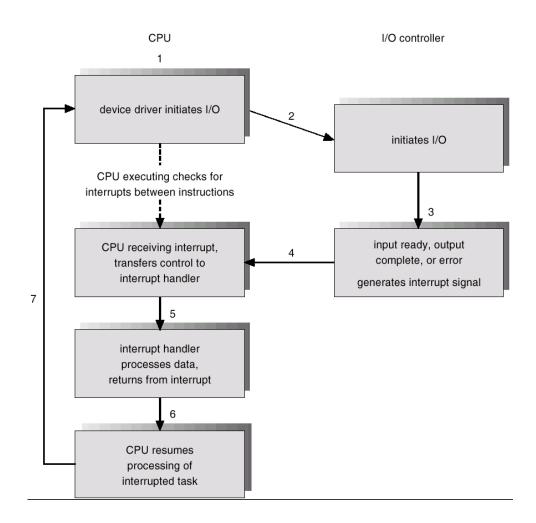


Técnicas para realização de E/S (1)

- Determina a forma de interação entre dispositivos e processador
- Três técnicas usadas:
 - E/S Programada
 - Interrupção
 - Acesso Direto à Memória (DMA)

Técnicas para realização de E/S (4)

E/S Orientada a Interrupção (cont.)

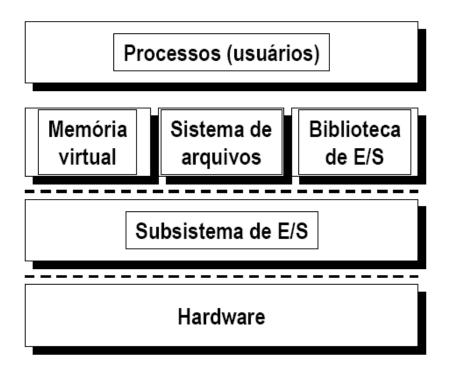


Subsistema (Software) de Entrada e Saída (1)

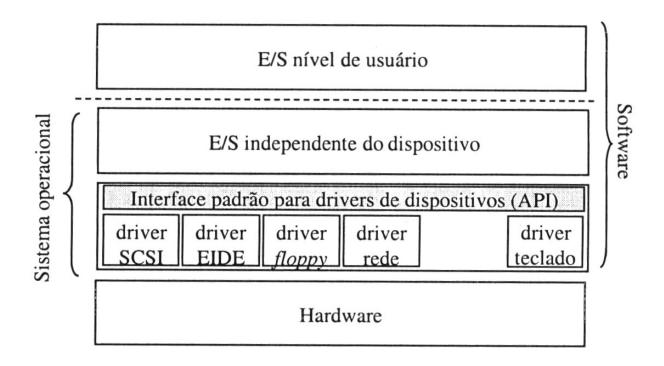
Objetivos

- Fornecer uma interface "amigável" para utilização dos dispositivos de E/S
 - Uniformizar o tratamento dos dispositivos
 - "Esconder" detalhes de mais "baixo nível"
- Permitir a inclusão de novos dispositivos
- Facilitar a correção de erros gerados pelo dispositivo
- Explorar eficientemente os dispositivos de E/S
 - Desempenho
- Compartilhamento dos dispositivos de E/S
- Alocação dos dispositivos
- Escalonamento de requisições

Subsistema (Software) de Entrada e Saída (2)



Subsistema (Software) de Entrada e Saída (3)



Subsistema (Software) de Entrada e Saída (4)

- Software de E/S de usuário:
 - Realiza chamadas de alto nível ao software de E/S independente de dispositivo:
 - Wrappers para chamadas de sistema
 - Abrir/fechar arquivo, ler/escrever dados, etc.

Subsistema (Software) de Entrada e Saída (5)

- E/S independente de dispositivo (a seguir..)
- Interface do subsistema de E/S (API)
 - dispositivos "abstratos" de E/S
 - Cada representa uma classe de dispositivos de E/S
- Dispositivos "abstratos"
 - Orientado a bloco (block device, buffered, random access)
 - Orientado a caractere (stream, character device, unbuffered)
 - Rede (não estão mapeados no sistema de arquivos, usam drivers do kernel, packets)

Linux kernel map functionalities human interface system processing networking storage memory layers interfaces core processes memory access HI char devices sockets access files & directories kernel/signal.c svs fork user svs brk sys_kill access System Call Interface system files sys_vfork cdev add sys open svs socketcall sys_mmapshm vm ops sys_execve svs signal linux/syscalls.h sys_clone sys_socket do_sigaction sys_connect space svs shmctl /proc /sysfs /dev sys_write sys_tee sys_poll linux/uaccess h sys_accept svs_shmat sys_pipe sys_select do_patt copy_from_user sysfs_ops sys_bind /proc/net/ interfaces sys_flock sys_read sys_splice sys_sendfile sys_listen input fops sys futex si meminfo sys_gettimeofday tcp4_seq_show and fops svs ioctl sys_sendmsg sys_mincore so proc sea show dev cdev_add sys_recvmsg system calls /dev/mem console foos svs times notify change sys setsockopt rt cache seq show and system files mem_fops svs newfstat cdev map sys epoll create sys_capset fb_fops /proc/meminfo sys reboot cdevsys_msync sys mount sys_init_module /proc/self/maps Virtual File System threads virtual memory protocol families sock create socket **Device Model** vmalloc_init vfs fsync security security queue work/ vfs read vfs fstat find_vma_prepare vfs_write vfs_create linux/security.h linux/kobject.h vfs_getattr inet_family_ops security_capset may_open kobject work struct virtual kset inet_create inode security socket create bus register vmalloc linux/device.h unix family ops inode_permission vfree bus_type inode operations kernel thread vmlist security_inode_create device_create proto ops security_ops vm struct file_system_type current device selinux ops do fork thread info inet_dgram_ops inet_stream_ops device type virt to page super_block generic_file_aio_read do_splice_direct socket file ops driver register ramfs_fs_type debugging synchronization memory device driver page cache networking socket log_buf mapping address_space bdi writeback thread bridges sys ptrace storage splice probe sock sendpage nfs_file_operations register kprobe add_timer do_mmap_pgoff load_module si_swapinfo cross-functional timer_list down interruptible tcp_sendpage kmem_cache_alloc smb fs type module up semaphore handle_sysrq oprofile start kobiect uevent init vma_link udp_sendpage swap_info kswapd wait event cifs file ops module_param kobiect uevent wake_up spin_lock_irqsave spin_unlock_irqrestore mm_struct sock splice read kodb breakpoint oprofile init do swap page iscsi_tcp_transport kernel param vm area struct tcp_splice_read wakeup kswapd HI subsystems Scheduler logical protocols system run logical memory boot, shutdown kernel/sched.c physically mapped memory file systems power management udp_prot "tcp "prot logical task struct oss init/main.c alsa tcp_v4_rcv ext4 file operations schedule timeout udp rcv kmalloc schedule implementations do_initcalls video device ext4_get_sb kfree kernel restart pgd_t pmd_t ext4 readdir mousedev handler run_init_process kernel_power_off ip_route_input alloc skb context_switch machine ops ip_rcv sk buff generic HW access **Page Allocator** abstract devices interrupts core block devices network interface and request\irq gendisk block/ pci driver request region HID class drivers __free_pages dev_queue_xmit block device operations netif receive skb device request_mem_region pci_register_driver kmem cache iiffies 64++ register netdev drivers/input/ tasklet action pci_request_regions kmem_cache_init kmem_cache_alloc _free_one_page net device ush driver control setup ira scsi device console tick periodic _get_free_pages scsi driver kbd usb submit urb timer interrunt alloc pages sd fops fb ops do softira usb hod giveback urb ether_setup ieee80211_rx do IRQ irq_desc mousedev ioremap netif carrier on ieee80211 xmit drm_driver softirg init totalram_pages try_to_free_pages drivers/net CPU specific HI peripherals device access physical memory disk controller network and bus drivers device drivers operations drivers device drivers hardware trap_init arch/x86/mm/ get_page_from_freelist usbnet probe ac97 driver native_init_IRQ switch to interfaces vga cor sosi host alloc ipw2100 pci init one ehci urb enqueue set_intr_gate atkbd dry free_area free_list system call drivers, registers and interrupts e1000 xmit frame i8042_driver usb hcd ira interrupt out of memory ahci_pci_driver e1000 intr aic94xx init pt_regs atomic_t pci_write num physpages do page fault user peripherals CPU disk controllers network controllers memorv electronics keyboard graphics card © 2007, 2010 Constantine Shulyupin www.Makel inux.net/kernel mai

Subsistema (Software) de Entrada e Saída (6)

- Interface do subsistema de E/S (cont.)
 - Dispositivos Orientado a bloco
 - Organiza dados em blocos de tamanho fixo
 - Acessa diretamente um bloco de dados
 - Blocos são identificados por endereços (número do bloco)
 - ex: blocos de disco
 - Operações típicas:
 - open(), read(), write() e close()
 - Disponibilizadas aos usuários via sistema de arquivos, permite leitura/escrita de tamanho arbitrário e acesso randômico

Subsistema (Software) de Entrada e Saída (7)

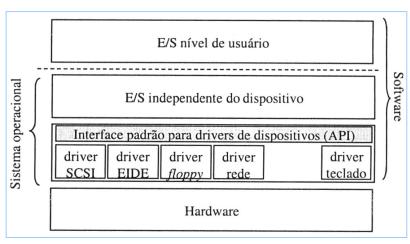
- Interface do subsistema de E/S (cont.)
 - Dispositivos Orientado a caractere
 - Controla dispositivos que produzem ou consomem conjunto de dados de tamanho arbitrário em formato de stream
 - Operações típicas:
 - put() e get()
 - ex: teclado, vídeo, mouse, impressora, etc...
 - Operações específicas (inversão de cores, bip, inicialização, etc) são fornecidas por uma função genérica
 - io control()

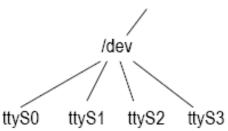
Subsistema (Software) de Entrada e Saída (8)

- Interface do subsistema de E/S (cont.)
 - Dispositivos Orientado a rede
 - Controle e acesso a dispositivos que estão fisicamente instalados em outros equipamentos
 - Necessário estabelecimento de conexões
 - Interface típica são sockets
 - Operações típicas:
 - open(), close(), create()
 - Suporte a serviços:
 - orientado a conexão: connect(), accept(), read(), write()
 - sem conexão: send() e recv()

Subsistema (Software) de Entrada e Saída (9)

- Software de E/S independente de dispositivo
 - Implementa funções gerais comuns a todos os dispositivos
 - Atribuição uniforme do nome independente do dispositivo
 - O UNIX é um exemplo clássico:
 - Nome do dispositivo é um string



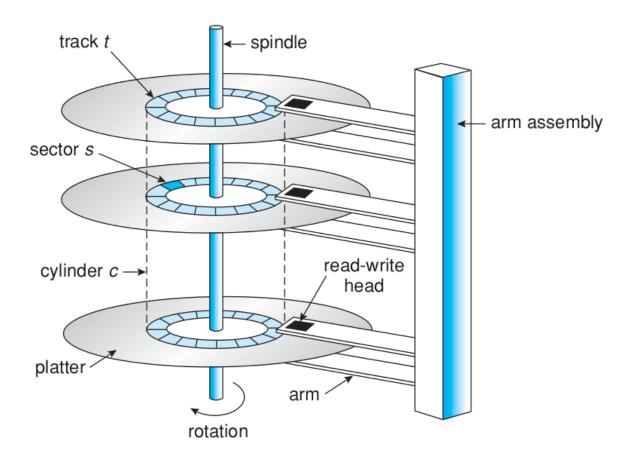


Subsistema (Software) de Entrada e Saída (10)

- Software de E/S independente de dispositivo
 - Implementa os serviços de:
 - Escalonamento de E/S: reordena as requisições para melhorar desempenho
 - Identificação (denominação) : nome lógico no SO
 - Bufferização: área de memória para armazenamento temporário de dados. Ajusta velocidade entre as camadas
 - Direitos de Acesso
 - Tratamento de erros

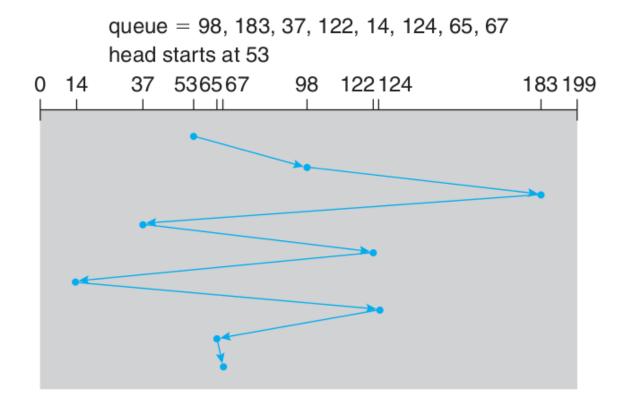
Subsistema (Software) de Entrada e Saída (11)

Mecanismo de um disco



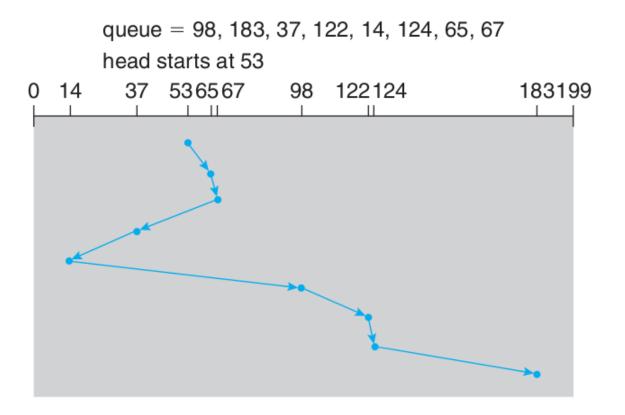
Subsistema (Software) de Entrada e Saída (12)

 Escalonamento de disco – FCFS (first come first served)



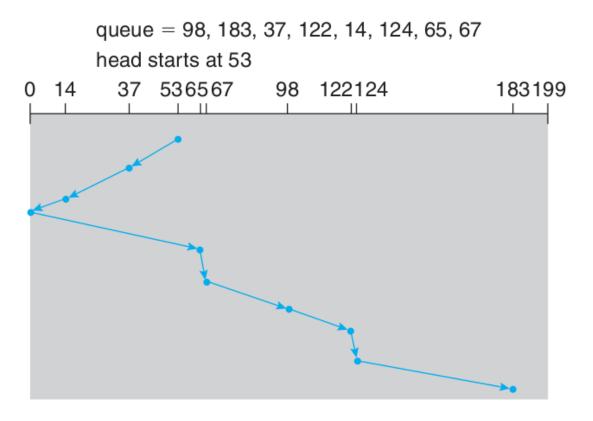
Subsistema (Software) de Entrada e Saída (13)

 Escalonamento de disco – SSTF (shortest seek time first)



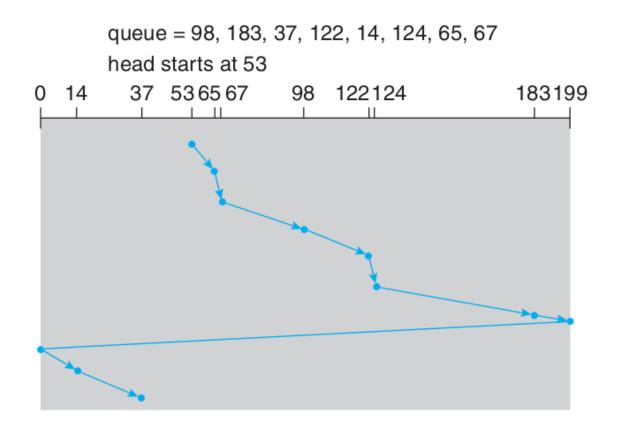
Subsistema (Software) de Entrada e Saída (14)

Escalonamento de disco - SCAN



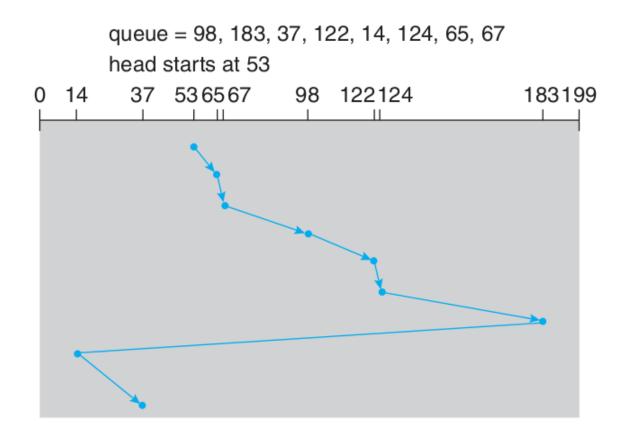
Subsistema (Software) de Entrada e Saída (15)

Escalonamento de disco - C-SCAN



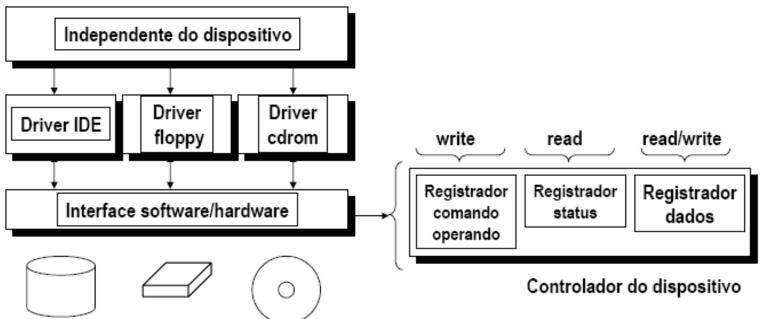
Subsistema (Software) de Entrada e Saída (16)

Escalonamento de disco - C-LOOK



Subsistema (Software) de Entrada e Saída (17)

- Drivers de dispositivo
 - Receber comandos da camada superior (independente de dispositivo) e interagir com os dispositivos
 - Implementa a interface padrão como sequência de acesso aos registradores



Subsistema (Software) de Entrada e Saída (18)

- Drivers de dispositivo (cont.)
 - Em geral, devem rodar no modo kernel
 - Desenvolvidos pelo fabricante do dispositivo
 - Dispositivos mais comuns podem ter o driver incluído no Sistema Operacional
 - Vantagens
 - Isolar o código específico a um dispositivo em um módulo aparte
 - Fabricantes não precisam mexer no kernel
 - Facilidade de adicionar novos drivers