

Universidade Federal Rural de Pernambuco  
Sistemas de Informação



Princípios de Software Básico

Lhaíslla Eduarda Cavalcanti Rodrigues da Silva

RECIFE-PE

## Exercícios referente ao capítulo - Entrada e Saída

1. Avanços na tecnologia de chips tornaram possível colocar o controlador inteiro, incluindo toda a lógica de acesso do barramento, em um chip barato. Como isso afeta o modelo da Figura 1.6?

Na figura pode-se observar o controlador do dispositivo como unidades separadas.

O motivo é permitir que um controlador manipula vários dispositivos e, assim, elimine a necessidade de ter um controlador por dispositivos.

Se os controladores ficarem quase livres, será mais simples incorporar o controlador no próprio dispositivo. Esse projeto também permitirá várias transferências em paralelo e, portanto, oferece um melhor desempenho.

O avanço da tecnologia possibilita aumentar o desempenho e a capacidade de processamento das informações, além do armazenamento. Por meio dos drives de dispositivos é possível realizar a integração entre software e hardware, além dos tratamentos das interrupções e erros que podem vir a ocorrer.

2. Dadas as velocidades listadas na Figura 5.1, é possível escanear documentos de um scanner e transmiti-los através de uma rede de 802,11 g na velocidade máxima? Defenda sua resposta. Sim, é possível, pois os drives têm a capacidade de prever a existência de interrupções externas e agir para evitar paralisações durante a execução, neste caso mesmo que o tempo de retorno para obtenção de resultado seja longo e hajam interrupções o computador dará um retorno do processo esperado.
3. A Figura 5.3(b) mostra uma maneira de conseguir a E/S mapeada na memória mesmo na presença de barramentos separados para dispositivos de memória e, E/S, a saber, primeiro tentar o barramento de memória e, se isso falhar, tentar o barramento de E/S. Um estudante de computação inteligente pensou em uma melhoria para essa ideia: tentar ambos em paralelo, a fim de acelerar o processo de acessar dispositivos de E/S. O que você acha dessa ideia?

Para cada registrador de controle é designado um endereço de memória único para o qual nenhuma memória é designada. Esse sistema é chamado de E/S mapeada na memória. Na maioria dos sistemas, os endereços designados estão no — ou próximos do — topo do espaço de endereçamento, com a E/S mapeada na memória, nenhum mecanismo de proteção especial é necessário para evitar que processos do usuário realizem E/S. Tudo o que o sistema operacional precisa fazer é deixar de colocar aquela porção do espaço de endereçamento contendo os registros de controle no espaço de endereçamento virtual de qualquer usuário. Melhor ainda, se cada dispositivo tem os seus registradores de controle em uma página diferente do espaço de endereçamento, o sistema operacional pode dar a um usuário controle sobre dispositivos específicos, mas não dar a outros, ao simplesmente incluir as páginas desejadas em sua tabela de páginas. Esse esquema pode permitir que diferentes drivers de dispositivos sejam colocados em diferentes espaços de endereçamento, não

apenas reduzindo o tamanho do núcleo, mas também impedindo que um driver interfira nos outros

4. Explique os ganhos e perdas entre interrupções precisas e imprecisas em uma máquina superescalar.

Uma vantagem de interrupções precisas é a simplicidade do código no sistema operacional, pois o estado da máquina está bem definido. Por outro lado, em interrupções imprecisas, os escritores do S.O precisam descobrir quais instruções foram parcialmente executadas e até que ponto. No entanto, interrupções precisas aumentaram a complexidade do projeto e da área do chip, o que resultou em uma CPU mais lenta.

5. Um controlador de DMA tem cinco canais. O controlador é capaz de solicitar uma palavra de 32 bits a cada 40 ns. Uma resposta leva um tempo igualmente longo. Quão rápido o barramento precisar ser para evitar ser um gargalo?

O barramento é o gargalo do sistema, uma vez que não pode ser utilizado para realizar mais de uma transferência de forma simultânea, neste caso é necessário esperar que a solicitação seja resolvida.

6. Suponha que um sistema usa DMA para a transferência de dados do controlador do disco para a memória principal. Além disso, presume-se que são necessários  $t_1$  ns em média para adquirir o barramento e  $t_2$  ns para transferir uma palavra através do barramento ( $t_1 \gg t_2$ ). Após a CPU ter programado o controlador de DMA, quanto tempo será necessário para transferir 1.000 palavras do controlador do disco para a memória principal, se (a) o modo uma-palavra-de-cada-vez for usado, (b) o modo de surto for usado? Presuma que o comando do controlador de disco exige adquirir o barramento para enviar uma palavra e o reconhecimento de uma transferência também exige adquirir o barramento para enviar uma palavra

Ignorando o uso de instruções de leitura/escrita e assumindo que o processador somente busca (fetch) instruções. Então o processador precisará acessar a memória principal somente uma vez a cada microsegundo. O módulo DMA transferirá caracteres numa taxa de 1200 caracteres por segundo, ou 1 a cada 833  $\mu$ s. O DMA, desta forma, roubará (steals) um ciclo a cada sequência de 833 ciclos. Assim, reduzirá a taxa de acesso à memória pelo processador de  $1/833 \times 100\% = 0,12\%$

7. Um modo que alguns controladores de DMA usam é o controlador do dispositivo enviar a palavra para o controlador de DMA, que então emite uma segunda solicitação de barramento para escrever para a memória. Como esse modo pode ser usado para realizar uma cópia memória para memória? Discuta qualquer vantagem ou desvantagem de usar esse método em vez de usar a CPU para realizar uma cópia memória para memória.

Ao realizar armazenamento interno, o controlador de disco pode conferir a soma de verificação antes de começar uma transferência. Se a soma de verificação estiver incorreta, um erro é sinalizado e nenhuma transferência é feita. Uma vez inicializada uma transferência de disco os bits continuam chegando do disco a uma taxa constante, não importa se o controlador estiver pronto para eles ou não.

8. Suponha que um computador consiga ler ou escrever uma palavra de memória em 5 ns. Também suponha que, quando uma interrupção ocorrer, todos os 32 registradores da CPU, mais o contador do programa e PSW são empurrados para a pilha. Qual é o número máximo de interrupções por segundo que essa máquina pode processar? Muitas instruções estão em vários estágios de execução. Quando ocorre a interrupção, o valor do contador do programa pode não refletir o limite correto entre as instruções executadas e as não executadas. Na realidade, muitas instruções talvez tenham sido parcialmente executadas, com diferentes instruções estando mais ou menos completas. Nessa situação, o contador do programa provavelmente refletirá o endereço da próxima instrução a ser buscada e colocada no pipeline em vez do endereço da instrução que acabou de ser processada pela unidade de execução.
9. Projetistas de CPUs sabem que projetistas de sistemas operacionais detestam interrupções imprecisas. Uma maneira de agradar ao pessoal do SO é fazer que a CPU pare de emitir novas instruções quando uma interrupção é sinalizada, mas permitir que todas as instruções que atualmente estão sendo executadas sejam concluídas, então forçar a interrupção. Essa abordagem tem alguma desvantagem? Explique a sua resposta.

Uma interrupção que deixa a máquina em um estado bem definido. O contador do programa (Program Counter — PC) é salvo em um lugar conhecido. Todas as instruções anteriores àquela apontada pelo PC foram completadas. Nenhuma instrução posterior à apontada pelo PC foi concluída. O estado de execução da instrução apontada pelo PC é conhecido. se ela é uma interrupção de software ou uma falta de página, então o PC geralmente aponta para a instrução que causou a falta para que ele possa ser reinicializado mais tarde.. Máquinas com interrupções imprecisas despejam em geral uma grande quantidade de estado interno sobre a pilha para proporcionar ao sistema operacional a possibilidade de descobrir o que está acontecendo. O código necessário para reiniciar a máquina costuma ser muito complicado. Também, salvar uma quantidade grande de informações na memória em cada interrupção torna as interrupções lentas e a recuperação ainda pior. Isso gera a situação irônica de termos CPUs superescalares muito rápidas sendo, às vezes, inadequadas para o trabalho em tempo real por causa das interrupções lentas
10. Na Figura 5.9(b), a interrupção não é reconhecida até depois de o caractere seguinte ter sido enviado para a impressora. Ele poderia ter sido reconhecido logo no início da rotina do serviço de interrupção? Se a resposta for sim, dê uma razão para fazê-lo ao fim, como no texto. Se não, por quê? Sim, para salvar as informações do bloco de

caracteres em uma tabela confiável, e colocando a nova informação em um pipeline.

11. Um computador tem um pipeline de três estágios como mostrado na Figura 1.7(a). Em cada ciclo do relógio, uma instrução nova é buscada da memória no endereço apontado pelo PC, colocado no pipeline e o PC incrementado. Cada instrução ocupa exatamente uma palavra de memória. As instruções que já estão no pipeline são avançadas em um estágio. Quando ocorre uma interrupção, o PC atual é colocado na pilha, e o PC é configurado para o endereço do tratador da interrupção. Então o pipeline é deslocado para a direita e a primeira instrução do tratador da interrupção é buscada no pipeline. Essa máquina tem interrupções precisas? Defenda sua resposta.

Sim, pois deixa a máquina em um estado bem definido e o contador do programa está salvo em um lugar conhecido.

12. Uma página de texto impressa típica contém 50 linhas de 80 caracteres cada. Imagine que uma determinada impressora possa imprimir 6 páginas por minuto e que o tempo para escrever um caractere para o registrador de saída da impressora é tão curto que ele pode ser ignorado. Faz sentido executar essa impressora usando a E/S orientada pela interrupção se cada caractere impresso exige uma interrupção que leva ao todo 50  $\mu$ s para servir? Não, pois a operação realizada é assíncrona — a CPU inicializa a transferência e vai fazer outra coisa até a chegada da interrupção. Programas do usuário são muito mais fáceis de escrever se as operações de E/S forem bloqueadas — após uma chamada de sistema read, o programa é automaticamente suspenso até que os dados estejam disponíveis no buffer. Fica a cargo do sistema operacional fazer operações que são realmente orientadas à interrupção parecerem bloqueantes para os programas do usuário.

13. Explique como um SO pode facilitar a instalação de um dispositivo novo sem qualquer necessidade de recompilar o SO.  
Com a utilização de VFS, virtual file system, que disponibiliza um meio-termo para os dispositivos. Uma interface em comum com unix, fat, ext2/3

14. Em qual das quatro camadas de software de E/S cada uma das tarefas a seguir é realizada:

(a) Calcular a trilha, setor e cabeçote para uma leitura de disco.

Hardware

(b) Escrever comandos para os registradores do dispositivo.

Tratador de interrupções

(c) Conferir se o usuário tem permissão de usar o dispositivo.

Softwares S.O independente do dispositivo

(d) Converter inteiros binários em ASCII para impressão

Drives

15. Uma rede de área local é usada como a seguir. O usuário emite uma chamada de sistema para escrever pacotes de dados para a rede. O sistema operacional então copia

os dados para um buffer do núcleo e, a seguir, copia os dados para a placa do controlador da rede. Quando todos os bytes estão seguramente dentro do controlador, eles são enviados através da rede a uma taxa de 10 megabits/s. O controlador da rede receptora armazena cada bit um microssegundo após ele ter sido enviado. Quando o último bit chega, a CPU de destino é interrompida, e o núcleo copia o pacote recém-chegado para um buffer do núcleo inspecioná-lo. Uma vez que ele tenha descoberto para qual usuário é o pacote, o núcleo copia os dados para o espaço do usuário. Se presumirmos que cada interrupção e seu processamento associado leva 1 ms, que pacotes têm 1024 bytes (ignore os cabeçalhos) e que copiar um byte leva 1  $\mu$ s, qual é a frequência máxima que um processo pode enviar dados para outro? Presuma que o emissor esteja bloqueado até que o trabalho tenha sido concluído no lado receptor e que uma mensagem de reconhecimento retorne. Para simplificar a questão, presuma que o tempo para receber o reconhecimento de volta é pequeno demais para ser ignorado.

Muitas vezes, dados provenientes de um dispositivo não podem ser armazenados diretamente em seu destino final. Por exemplo, quando um pacote chega da rede, o sistema operacional não sabe onde armazená-lo definitivamente até que o tenha colocado em algum lugar para examiná-lo. Também, alguns dispositivos têm severas restrições de tempo real (por exemplo, dispositivos de áudio digitais), portanto os dados devem ser colocados antecipadamente em um buffer de saída para separar a taxa na qual o buffer é preenchido da taxa na qual ele é esvaziado, a fim de evitar seu completo esvaziamento. A utilização do buffer envolve consideráveis operações de cópia e muitas vezes tem um impacto importante sobre o desempenho de E/S. O conceito final que mencionaremos aqui é o de dispositivos compartilhados versus dedicados. Alguns dispositivos de E/S, como discos, podem ser usados por muitos usuários ao mesmo tempo. Nenhum problema é causado por múltiplos usuários terem arquivos abertos no mesmo disco ao mesmo tempo. Outros dispositivos, como impressoras, têm de ser dedicados a um único usuário até ele ter concluído sua operação. Então outro usuário pode ter a impressora. Ter dois ou mais usuários escrevendo caracteres de maneira aleatória e intercalada na mesma página definitivamente não funcionará. Introduzir dispositivos dedicados (não compartilhados) também introduz uma série de problemas, como os impasses. Novamente, o sistema operacional deve ser capaz de lidar com ambos os dispositivos — compartilhados e dedicados — de uma maneira que evite problemas.

16. Por que arquivos de saída para a impressora normalmente passam por um spool no disco antes de serem impressos? Por causa do buffer da impressora, que provê uma estação de espera onde o dado pode ficar armazenado enquanto o dispositivo mais lento fica carregando. Uma estratégia possível para lidar com os caracteres que chegam é fazer o processo do usuário realizar uma chamada de sistema read e bloquear a espera por um caractere. Cada caractere que chega causa uma interrupção. A rotina de tratamento da interrupção passa o caractere para o processo do usuário e o desbloqueia. Após colocar o caractere em algum lugar, o processo lê outro caractere e

bloqueia novamente

17. . Quanto deslocamento de cilindro é necessário para um disco de 7200 RPM com um tempo de busca de trilha para trilha de 1 ms? O disco tem 200 setores de 512 bytes cada em cada trilha. o maior disco possível é 31,5 GB
18. Um disco gira a 7200 RPM. Ele tem 500 setores de 512 bytes em torno do cilindro exterior. Quanto tempo ele leva para ler um setor?  
Para esconder os detalhes de quantos setores tem cada trilha, a maioria dos discos modernos tem uma geometria virtual que é apresentada ao sistema operacional. O software é instruído a agir como se houvesse x cilindros, y cabeçotes e z setores por trilha. O controlador então realiza um remapeamento de uma solicitação para (x, y, z) no cilindro, cabeçote e setor real.
19. Calcule a taxa de dados máxima em bytes/s para o disco descrito no problema anterior. O disco tem 200 setores de 512 bytes cada em cada trilha. o maior disco possível é 31,5 GB.
20. RAID nível 3 é capaz de corrigir erros de bit único usando apenas uma unidade de paridade. Qual é o sentido do RAID nível 2? Afinal de contas, ele só pode corrigir um erro e precisa de mais unidades para fazê-lo.  
Diferentemente dos níveis 0 e 1, que trabalham com faixas de setores, o RAID nível 2 trabalha com palavras, talvez mesmo com bytes. Imagine dividir cada byte de um único disco virtual em um par de pedaços de 4 bits cada, então acrescentar um código Hamming para cada um para formar uma palavra de 7 bits, das quais os bits 1, 2 e 4 são de paridade.
21. Um RAID pode falhar se duas ou mais das suas unidades falharem dentro de um intervalo de tempo curto. Suponha que a probabilidade de uma unidade falhar em uma determinada hora é p. Qual é a probabilidade de um RAID com k unidades falhar em uma determinada hora? A probabilidade K/P
22. Compare o RAID nível 0 até 5 em relação ao desempenho de leitura, desempenho de escrita, sobrecarga de espaço e confiabilidade.  
Os níveis RAID 4 e 5 trabalham novamente com faixas, não palavras individuais com paridade, e não exigem discos sincronizados Diferentemente dos níveis 0 e 1, que trabalham com faixas de setores, o RAID nível 2 trabalha com palavras, talvez mesmo com bytes. Imagine dividir cada byte de um único disco virtual em um par de pedaços de 4 bits cada, então acrescentar um código Hamming para cada um para formar uma palavra de 7 bits, das quais os bits 1, 2 e 4 são de paridade.
23. Quantos pebibytes há em um zebibyte?  
Zebibytes para Pebibytes = 1048576

24. Quais são as vantagens e as desvantagens dos discos ópticos versus os discos magnéticos?

A informação gravada em CD tem uma durabilidade superior à dos suportes magnéticos, visto que não é destruída por interferências electromagnéticas.

25. Discos ópticos podem ser facilmente transportados para outros computadores e tem um custo menor

A velocidade de funcionamento dos discos ópticos ainda se mantém inferior à dos discos magnéticos. O problema de apagar e regravar a informação são meios de armazenamento em que a leitura e a escrita são feitas por laser. Ambos os meios de armazenamento tem seus prós e contras. Depende do seu uso final, para que os dados não se percam, precisam de ser gravados num dispositivo de armazenamento como disco rígidos, discos ópticos entre outros.

26. Se um controlador de disco escreve os bytes que ele recebe do disco para a memória tão rápido quanto ele os recebe, sem armazenamento interno, o entrelaçamento é concebivelmente útil? Explique. Não, pois pode ocorrer interrupções no momento da escrita do arquivo e ocasionar perda de informações.

27. Se um disco tem entrelaçamento duplo, ele também precisa de deslocamento do cilindro a fim de evitar perder dados quando realizando uma busca de trilha para trilha? Discuta sua resposta.

Se o controlador tem um buffer de apenas um setor, não importa se a cópia do buffer para a memória principal é feita pelo controlador, a CPU principal ou um chip DMA; ela ainda leva algum tempo. Para evitar a necessidade do entrelaçamento, o controlador deve ser capaz de armazenar uma trilha inteira. A maioria dos controladores modernos consegue armazenar trilhas inteiras.

28. Considere um disco magnético consistindo de 16 cabeças e 400 cilindros. Esse disco tem quatro zonas de 100 cilindros com os cilindros nas zonas diferentes contendo 160, 200, 240 e 280 setores, respectivamente. Presuma que cada setor contenha 512 bytes, que o tempo de busca médio entre cilindros adjacentes é 1 ms, e o disco gira a 7200 RPM. Calcule a (a) capacidade do disco, (b) deslocamento de trilha ótimo e (c) taxa de transferência de dados máxima.

A maioria dos discos usa setores de 512 bytes. O campo ECC contém informações redundantes que podem ser usadas para a recuperação de erros de leitura. O tamanho e o conteúdo desse campo variam de fabricante para fabricante, dependendo de quanto espaço de disco o projetista está disposto a abrir mão em prol de uma maior confiabilidade, assim como o grau de complexidade do código de ECC que o



controlador é capaz de manejar.

29. Um fabricante de discos tem dois discos de 5,25 polegadas, com 10.000 cilindros cada um. O mais novo tem duas vezes a densidade de gravação linear que o mais velho. Quais propriedades de disco são melhores na unidade mais nova e quais são as mesmas? Há alguma pior na unidade mais nova?

O tamanho e o conteúdo desse campo variam de fabricante para fabricante, dependendo de quanto espaço de disco o projetista está disposto a abrir mão em prol de uma maior confiabilidade, assim como o grau de complexidade do código de ECC que o controlador é capaz de manejar. Um campo de ECC de 16 bytes não é incomum. Além disso, todos os discos rígidos têm algum número de setores sobressalentes alocados para serem usados para substituir setores com defeito de fabricação.

30. . Um fabricante de computadores decide redesenhar a tabela de partição de um disco rígido Pentium para gerar mais do que quatro partições. Cite algumas das consequências dessa mudança.

Redução do disco, diminuição de desempenho, aumento do buffer do controlador, perda de performance

31. Solicitações de disco chegam ao driver do disco para os cilindros 10, 22, 20, 2, 40, 6 e 38, nessa ordem. Uma busca leva 6 ms por cilindro. Quanto tempo de busca é necessário para:

(a) Primeiro a chegar, primeiro a ser servido.

12 ms

(b) Cilindro mais próximo em seguida.

38 ms

(c) Algoritmo do elevador (inicialmente deslocando-se para cima). Em todos os casos, o braço está inicialmente no cilindro 20. O algoritmo do elevador exige que o software mantenha 1 bit: o bit de direção atual, SOBE ou DESCE. Quando uma solicitação é concluída, o driver do disco ou elevador verifica o bit. Se ele for SOBE, o braço ou a cabine se move para a próxima solicitação pendente mais alta. Se nenhuma solicitação estiver pendente em posições mais altas, o bit de direção é invertido.

32. Uma ligeira modificação do algoritmo do elevador para o escalonamento de solicitações de disco é sempre varrer na mesma direção. Em qual sentido esse algoritmo modificado é melhor do que o algoritmo do elevador?

Dada qualquer coleção de solicitações, o limite máximo para a distância total é fixo: ele é apenas duas vezes o número de cilindros. Uma ligeira modificação desse algoritmo que tem uma variação menor nos tempos de resposta (TEORY, 1972) consiste em sempre varrer as solicitações na mesma direção. Quando o cilindro com o número mais alto com uma solicitação pendente tiver sido atendido, o braço vai para o cilindro com o número mais baixo com uma solicitação pendente e então continua

movendo-se para cima. Na realidade, o cilindro com o número mais baixo é visto como logo acima do cilindro com o número mais alto.

33. Um vendedor de computadores pessoais visitando uma universidade na região sudoeste de Amsterdã observou durante seu discurso de venda que a sua empresa havia devotado um esforço substancial para tornar a sua versão do UNIX muito rápida. Como exemplo, ele observou que o seu driver do disco usava o algoritmo do elevador e também deixava em fila múltiplas solicitações dentro de um cilindro por ordem do setor. Um estudante, Harry Hacker, ficou impressionado e comprou um. Ele o levou para casa e escreveu um programa para ler aleatoriamente 10.000 blocos espalhados pelo disco. Para seu espanto, o desempenho que ele mensurou foi idêntico ao que era esperado do primeiro a chegar, primeiro a ser servido. O vendedor estava mentindo?

Não, o algoritmo do elevador tem um limite máximo das solicitações fixo, continuam se movimentando na mesma direção até que não existam mais solicitações pendentes naquela direção, então eles trocam de direção. Se duas ou mais solicitações para o mesmo cilindro estiverem pendentes, o driver poderá emitir uma solicitação para o setor que passará sob o cabeçote em seguida.

34. Na discussão a respeito de armazenamento estável usando RAM não volátil, o ponto a seguir foi minimizado. O que acontece se a escrita estável termina, mas uma queda do sistema ocorre antes que o sistema operacional possa escrever um número de bloco inválido na RAM não volátil? Essa condição de corrida arruína a abstração do armazenamento estável? Explique a sua resposta.

Quando um disco escreve um bloco (um ou mais setores), ou a escrita está correta ou ela está incorreta e esse erro pode ser detectado em uma leitura subsequente examinando os valores dos campos ECC. Em princípio, a detecção de erros garantida jamais é possível, pois com um, digamos, campo de ECC de 16 bytes guardando um setor de 512 bytes, há 24096 valores de dados e apenas 2144 valores ECC. Desse modo, se um bloco for distorcido durante a escrita — mas o ECC não —, existem bilhões e mais bilhões de combinações incorretas que resultam no mesmo ECC. Se qualquer uma delas ocorrer, o erro não será detectado.

35. Na discussão sobre armazenamento estável, foi demonstrado que o disco pode ser recuperado para um estado consistente (uma escrita é concluída ou nem chega a ocorrer) se a falha da CPU ocorrer durante uma escrita. Essa propriedade se manterá se a CPU falhar novamente durante a rotina de recuperação? Explique sua resposta.

Qualquer escrita no disco em andamento no momento da falha também para, levando a dados incorretos em um setor e um ECC incorreto que pode ser detectado mais tarde. Com todas essas considerações, o armazenamento estável pode se tornar 100% confiável, no sentido de que as escritas funcionem corretamente ou deixem os dados antigos no lugar. É claro, ele não protege contra desastres físicos, como um terremoto acontecendo e o computador despencando 100 metros em uma fenda e caindo dentro de um lago de lava fervendo. É difícil recuperar de uma situação dessas

por software. O armazenamento estável usa um par de discos idênticos com blocos correspondentes trabalhando juntos para formar um bloco livre de erros. Na ausência de erros, os blocos correspondentes em ambos os dispositivos são os mesmos. Qualquer um pode ser lido para conseguir o mesmo resultado.

36. . Na discussão sobre armazenamento estável, uma suposição fundamental é de que uma falha na CPU que corrompe um setor leva a um ECC incorreto. Quais problemas podem surgir nos cenários de cinco recuperações de falhas mostrados na Figura 5.27 se essa suposição não se mantiver? A falha da CPU acontece após a unidade 1 ter sido escrita, mas antes da unidade 2 ter sido escrita. O ponto em que não há mais volta foi passado aqui: o programa de recuperação copia o bloco da unidade 1 para a unidade 2. A escrita é bem-sucedida. Durante a recuperação, o bloco bom escrever sobre o defeituoso. Novamente, o valor final de ambos os blocos é o novo. O programa de recuperação vê que ambos os blocos são os mesmos, portanto nenhum deles é modificado e a escrita é bem-sucedida aqui também.
37. O tratador de interrupção do relógio em um determinado computador exige 2 ms (incluindo a sobrecarga de troca de processo) por tique do relógio. O relógio executa a 60 Hz. Qual fração da CPU é devotada ao relógio? Uma maneira menos precisa, porém mais simples, para contabilizar o uso da CPU é manter um ponteiro para a entrada da tabela de processos relativa ao processo em execução em uma variável global. A cada tique do relógio, um campo na entrada do processo atual é incrementado. Dessa maneira, cada tique do relógio é “cobrado” do processo em execução no momento do tique. Um problema menor com essa estratégia é que se muitas interrupções ocorrerem durante a execução de um processo, ele ainda será cobrado por um tique completo, mesmo que ele não tenha realizado muito trabalho. A contabilidade apropriada da CPU durante as interrupções é cara demais e feita raramente. Em muitos sistemas, um processo pode solicitar que o sistema operacional lhe dê um aviso após um determinado intervalo. O aviso normalmente é um sinal, interrupção, mensagem, ou algo similar. Uma aplicação que requer o uso desses avisos é a comunicação em rede, na qual um pacote sem confirmação de recebimento dentro de um determinado intervalo de tempo deve ser retransmitido. Outra aplicação é o ensino auxiliado por computador, onde um estudante que não dê uma resposta dentro de um determinado tempo recebe a resposta do computador
38. Um computador usa um relógio programável em modo de onda quadrada. Se um cristal de 500 MHz for usado, qual deveria ser o valor do registrador (holding register) para atingir uma resolução de relógio de
- (a) um milissegundo (um tique de relógio a cada milissegundo)?  $5 \times 10^5$
  - (b) 100 microssegundos?  $5 \times 10^{-1}$
39. Um sistema simula múltiplos relógios encadeando juntas todas as solicitações de relógio pendentes como mostrado na Figura 5.30. Suponha que o tempo atual seja 5000 e existem solicitações de relógio pendentes para o tempo 5008, 5012, 5015,

5029 e 5037. Mostre os valores do cabeçalho do relógio, Tempo atual e próximo sinal nos tempos 5000, 5005 e 5013. Suponha que um novo sinal (pendente) chegue no tempo 5017 para 5033. Mostre os valores do cabeçalho do relógio, Tempo atual e Próximo sinal no tempo 5023.

A vantagem do relógio programável é que a sua frequência de interrupção pode ser controlada pelo software. Se um cristal de 500 MHz for usado, então o contador é pulsado a cada 2 ns. Com registradores de 32 bits (sem sinal), as interrupções podem ser programadas para acontecer em intervalos de 2 ns a 8,6 s. Chips de relógios programáveis costumam conter dois ou três relógios programáveis independentemente e têm muitas outras opções também (por exemplo, contar com incremento em v.

40. Muitas versões do UNIX usam um inteiro de 32 bits sem sinal para controlar o tempo como o número de segundos desde a origem do tempo. Quando esses sistemas entrarão em colapso (ano e mês)? Você acha que isso vai realmente acontecer? Em 19 de Janeiro de 2038, acredito que outros sistemas irão substituir e sanar esse déficit do UNIX, por uma questão de maior disponibilidade de memória.

41. Um terminal de mapas de bits contém 1600 por 1200 pixels. Para deslizar o conteúdo de uma janela, a CPU (ou controlador) tem de mover todas as linhas do texto para cima copiando seus bits de uma parte da RAM do vídeo para outra. Se uma janela em particular tiver 80 linhas de altura e 80 caracteres de largura (6400 caracteres, total), e uma caixa de caracteres tem 8 pixels de largura por 16 pixels de altura, quanto tempo leva para passar a janela inteira a uma taxa de cópia de 50 ns por byte? Se todas as linhas têm 80 caracteres de comprimento, qual é a taxa de símbolos por segundo (baud rate) do terminal? Colocar um caractere na tela leva 5  $\mu$ s. Quantas linhas por segundo podem ser exibidas? Os temporizadores por software (soft timers) evitam interrupções. Em vez disso, sempre que o núcleo está executando por alguma outra razão, imediatamente antes de retornar para o modo do usuário ele verifica o relógio de tempo real para ver se um temporizador por software expirou. Se ele expirou, o evento escalonado (por exemplo, a transmissão de um pacote ou a verificação da chegada de um pacote) é realizado sem a necessidade de chavear para o modo núcleo, dado que o sistema já está ali. Após o trabalho ter sido realizado, o temporizador por software é reinicializado novamente.

42. Após receber um caractere DEL (SIGINT), o driver do monitor descarta toda a saída atualmente em fila para aquele monitor. Por quê?

Se o formato interno for armazenar ambos (a convenção do Windows), então o driver deve gerar uma alimentação de linha quando recebe um CR e um CR quando recebe uma alimentação de linha. Não importa qual seja a convenção interna, o monitor pode exigir que tanto uma alimentação de linha quanto um CR sejam ecoados a fim de obter uma atualização adequada da tela. Em um sistema com múltiplos usuários como um computador de grande porte, diferentes usuários podem ter diversos tipos de terminais conectados a ele e cabe ao driver do teclado conseguir que

todas as combinações de CR/alimentação de linha sejam convertidas ao padrão interno do sistema e arranjar que todos os ecos sejam feitos corretamente.

43. Um usuário em um terminal emite um comando para um editor remover a palavra na linha 5 ocupando as posições de caractere 7 até e incluindo 12. Presumindo que o cursor não está na linha 5 quando o comando é dado, qual sequência de escape ANSI o editor deve emitir para remover a palavra?

o driver dá um sinal dizendo que o próximo caractere é isento de processamento especial. O caractere LNEXT em si não é inserido na fila de caracteres. Para permitir que os usuários parem uma imagem na tela que está saindo de seu campo de visão, códigos de controle são fornecidos para congelar a tela e inicializá-la mais tarde. No UNIX esses são STOP (CTRL- -S) e START (CTRL-Q), respectivamente. Eles não são armazenados, mas são usados para ligar e desligar um sinal na estrutura de dados do teclado. Sempre que ocorre uma tentativa de saída, o sinal é inspecionado. Se ele está ligado, a saída não ocorre. Em geral, o eco também é suprimido junto com a saída do programa. Muitas vezes é necessário matar um programa descontrolado que está sendo depurado. Os caracteres INTR (DEL) e QUIT (CTRL-\) podem ser usados para esse fim. No UNIX, DEL envia o sinal SIGINT para todos os processos inicializados a partir daquele teclado

44. Os projetistas de um sistema de computador esperavam que o mouse pudesse ser movido a uma taxa máxima de 20 cm/s. Se um mickey é 0,1 mm e cada mensagem do mouse tem 3 bytes, qual é a taxa de dados máxima do mouse presumindo que cada mickey é relatado separadamente? 32 bits

45. As cores primárias aditivas são vermelho, verde e azul, o que significa que qualquer cor pode ser construída a partir da superposição linear dessas cores. É possível que uma pessoa tenha uma fotografia colorida que não possa ser representada usando uma cor de 24 bits inteira? O sistema tenta aproximar ao máximo as cores, caso não consiga ler todos os bits, o sistema redireciona a cores com valores aproximados. Resoluções mais altas são voltadas para os monitores widescreen cuja proporção de perspectiva casa com elas. A uma resolução de apenas  $1920 \times 1080$  (o tamanho dos vídeos HD inteiros), uma tela colorida com 24 bits por pixel exige em torno de 6,2 MB de RAM apenas para conter a imagem, então com 256 MB ou mais, o adaptador gráfico pode conter muitas imagens ao mesmo tempo. Se a tela inteira for renovada 75 vezes/s, a RAM de vídeo tem de ser capaz de fornecer dados continuamente a 445 MB/s.

46. Uma maneira de colocar um caractere em uma tela com mapa de bits é usar BitBlt com uma tabela de fontes. Presuma que uma fonte em particular usa caracteres que são  $16 \times 24$  pixels em cor RGB (red, green, blue) real. (a) Quanto espaço da tabela de fonte cada caractere ocupa? (b) Se copiar um byte leva 100 ns, incluindo sobrecarga, qual é a taxa de saída para a tela em caracteres/s?

Um adaptador gráfico contém uma memória especial chamada RAM de vídeo

que armazena as imagens que aparecem na tela. Adaptadores gráficos muitas vezes têm poderosas CPUs de 32 ou 64 bits e até 4 GB de sua própria RAM, separada da memória principal do computador. Cada adaptador gráfico suporta um determinado número de tamanhos de telas. Tamanhos comuns (horizontal  $\times$  vertical em pixels) são  $1280 \times 960$ ,  $1600 \times 1200$ ,  $1920 \times 1080$ ,  $2560 \times 1600$ , e  $3840 \times 2160$ . Muitas resoluções na prática encontram-se na proporção 4:3, que se ajusta à proporção de perspectiva dos aparelhos de televisão NTSC e PAL e desse modo fornece pixels quadrados nos mesmos monitores usados para os aparelhos de televisão

47. Presumindo que são necessários 2 ns para copiar um byte, quanto tempo leva para reescrever completamente uma tela de modo texto com 80 caracteres  $\times$  25 linhas, no modo de tela mapeada na memória? E uma tela em modo gráfico com  $1024 \times 768$  pixels com 24 bits de cores? .

Um dos itens mais importantes no orçamento de energia de todo mundo é o monitor. Para obter uma imagem clara e nítida, a tela deve ser iluminada por trás e isso consome uma energia substancial. Muitos sistemas operacionais tentam poupar energia desligando o monitor quando não há atividade alguma por um número determinado de minutos. Muitas vezes o usuário pode decidir qual deve ser esse intervalo, devendo escolher entre o desligamento frequente da tela e o consumo rápido da bateria (o que provavelmente o usuário não deseja).

48. . Na Figura 5.36 há uma classe para RegisterClass. No código X Window correspondente, na Figura 5.34, não há uma chamada assim, nem algo parecido. Por que não? O quarto parâmetro, iCmdShow, diz se a janela inicial do programa deve ocupar toda a tela, parte ou nada dela (somente a barra de tarefas). Essa declaração ilustra uma convenção amplamente usada pela Microsoft chamada de notação húngara. O nome é uma brincadeira com a notação polonesa, o sistema pós-fixado inventado pelo lógico polonês J.

49. No texto demos um exemplo de como desenhar um retângulo na tela usando o Windows GDI: Rectangle(hdc, xleft, ytop, xright, ybottom); Existe mesmo a necessidade para o primeiro parâmetro (hdc), e se afirmativo, qual? Afinal de contas, as coordenadas do retângulo estão explicitamente especificadas como parâmetros.

A ação de desenhar para a tela é manejada por um pacote que consiste em centenas de rotinas que são reunidas para formar a interface do dispositivo gráfico (GDI — Graphics Device Interface). Ela pode lidar com texto e gráficos e é projetada para ser independente de plataformas e dispositivos. Antes que um programa possa desenhar (isto é, pintar) em uma janela, ele precisa adquirir um contexto do dispositivo, que é uma estrutura de dados interna contendo propriedades da janela, como a fonte, cor do texto, cor do segundo plano, e assim por diante. A maioria das chamadas para a GDI usa o contexto de dispositivo, seja para desenhar ou para obter ou ajustar as propriedades

50. Um terminal de cliente magro é usado para exibir uma página na web contendo um desenho animado de tamanho  $400 \text{ pixels} \times 160 \text{ pixels}$  executando a 10 quadros/s. Qual fração de um Fast Ethernet de 100 Mbps é consumida exibindo o desenho?

Uma questão relacionada é que os usuários deveriam fazer backups regulares de seus sistemas de arquivos de gigabytes, mas poucos o fazem. Quando acontece um desastre, o que se vê é muita lamentação! Com um sistema centralizado, backups podem ser feitos todas as noites por robôs de fita automatizados. Outra vantagem é que o compartilhamento de recursos é mais fácil com sistemas centralizados. Um sistema com 256 usuários remotos, cada um com 256 MB de RAM, terá a maior parte dessa RAM ociosa a maior parte do tempo. Com um sistema centralizado com 64 GB de RAM, nunca acontece de um algum usuário precisar temporariamente de muita RAM, mas não conseguir porque ela está no PC de outra pessoa. O mesmo argumento se mantém para o espaço de disco e outros recursos.

51. Foi observado que um sistema de cliente magro funciona bem com uma rede de 1 Mbps em um teste. É provável que ocorram problemas em uma situação de múltiplos usuários? (Dica: considere um grande número de usuários assistindo a um programa de TV programado e o mesmo número de usuários navegando na internet.)

web, e talvez nem isso. Dizer que a maioria dos usuários quer uma computação interativa de alto desempenho, mas não quer realmente administrar um computador, provavelmente seja uma conclusão justa. Isso levou os pesquisadores a reexaminar os sistemas de tempo compartilhado usando terminais burros (agora educadamente chamados de clientes magros) que atendem às expectativas de terminais modernos.

52. Descreva duas vantagens e duas desvantagens da computação com clientes magros. Uma vantagem é o compartilhamento de recursos é mais fácil com sistemas centralizados. A desvantagem é que cada PC tem um disco rígido grande e um software complexo que devem ser mantidos .

53. Se a voltagem máxima da CPU,  $V$ , for cortada para  $V/n$ , seu consumo de energia cairá para  $1/n^2$  do seu valor original e sua velocidade de relógio cairá para  $1/n$  do seu valor original. Suponha que um usuário está digitando a 1 caractere/s, mas o tempo da CPU necessário para processar cada caractere é 100 ms. Qual é o valor ótimo de  $n$  e qual é a economia de energia correspondente percentualmente comparada com não cortar a voltagem? Presuma que uma CPU ociosa não consuma energia alguma.

O gerenciamento de energia provoca uma série de questões com que o sistema operacional precisa lidar. Muitas delas relacionam-se com a hibernação de recursos — seletiva e temporariamente desligar dispositivos, ou pelo menos reduzir seu consumo de energia quando estão ociosos.

54. Um notebook é configurado para tirar o máximo de vantagem das características de economia de energia, incluindo desligar o monitor e o disco rígido após períodos de inatividade. Uma usuária às vezes executa programas UNIX em modo de texto e

outras vezes usa o Sistema X Window. Ela ficou surpresa ao descobrir que a vida da bateria é significativamente mais longa quando usa programas somente de texto. Por quê? Por causa do tempo de resposta dos programas que exigem maior desempenho do Sistema operacional, da capacidade de processamento dos drives e dispositivos de entrada e saída em responder as solicitações.