	COLÉGIO ESTADUAL PROTÁSIO ALVES ENSINO HÍBRIDO - 2021	
	PROFESSORA: Maria Helena - maria-hsilva368@educar.rs.gov.br DISCIPLINA: Sistemas Operacionais CONTEÚDO: Gerência de Dispositivos	Material de Consulta
		DATA: 29 a 02/07
OBSERVAÇÕES: Gerência de Dispositivos - Forma de conexão, transmissão e gerenciamento do sistema operacional com os dispositivos.		

GERÊNCIA DE DISPOSITIVOS

Objetivo: Compreender o funcionamento dos sistemas operacional no gerenciamento dos dispositivos, quanto ao modo de comportamento e forma de relacionar-se com CPU.

Conteúdo

- Introdução
- Drives
- Interrupções
- técnicas
- formas de conexões
- Classificação quanto ao tipo

INTRODUÇÃO

A gerência do dispositivo é uma das principais e mais complexas funções dos sistemas operacionais. Cada periférico do computador possui suas peculiaridades, assim, o procedimento de interação com uma placa de rede é completamente diferente da interação comum disco rígido SCSI. Todavia, existem muitos problemas e abordagens em comum para o acesso aos periféricos. Por exemplo, é possível criar uma abstração única para a maioria dos dispositivos de armazenamento como pen-drives, discos SCSI ou IDE, disquetes e etc, na forma de um vetor de blocos de dados. A função da gerência de dispositivos (também conhecida como gerência de entrada/saída) é implementar a interação com cada dispositivo por meio de drivers e criar modelos abstratos que permitam agrupar vários dispositivos distintos sob a mesma interface de acesso.

Dispositivos de entrada e saída

Os dispositivos de entrada e saída (E/S) são utilizados para permitir a comunicação entre o computador e o mundo externo. Através desses dispositivos a UCP e a memória principal podem se comunicar tanto com usuários quanto com memórias secundárias, a fim de realizar qualquer tipo de processamento.

Os dispositivos de E/S podem ser divididos em duas categorias:

os que são utilizados como memória secundária e os que servem para a interface homem-máquina.

Os Dispositivos utilizados como memória secundária como discos se caracterizam por armazenar grande volume de informações, seu custo é relativamente baixo e seu tempo de acesso é maior que o acesso à memória principal.

Alguns dispositivos servem para a comunicação homem-máquina, como teclados, monitores de vídeo, impressoras, entre

outros. Com o avanço no desenvolvimento de aplicações de uso cada vez mais geral procura-se aumentar a facilidade de comunicação entre o usuário e o computador. A implementação de interfaces mais amigáveis permite cada vez mais que pessoas sem conhecimento específico sobre informática possam utilizar o computador. Scanner, caneta óptica, mouse, dispositivos sensíveis à voz humana e ao calor do corpo humano são alguns exemplos desses tipos de dispositivos.

Device drivers

Device driver é o comando de um dispositivo ou programa. É a forma a partir da qual uma unidade periférica cria uma interface com o sistema operacional para se conectar com o dispositivo do hardware. Vejamos um exemplo prático: quando se conecta uma impressora a um computador, esta impressora requer a instalação do "driver" (que é instalado a partir de um CD ou de um disquete que vem junto com o equipamento), sem o qual ela não conseguirá fazer a interface com o Computador. O "driver" é o elemento que faz esse comando. É, literalmente, o dirigente.

Foi a solução encontrada para que os Sistemas Operacionais sejam compatíveis com diferentes tipos de equipamentos. Cada impressora, por exemplo, tem suas peculiaridades de hardware, logo torna-se inviável que o Sistema Operacional tenha conhecimento sobre todos os equipamentos disponíveis. O Sistema Operacional disponibiliza bibliotecas de programação, para que o fabricante possa criar uma interface entre seu equipamento e o software.

Acesso Direto à Memória (DMA)

O Acesso Direto à Memória (DMA) é uma das técnicas utilizadas para otimizar o uso de memória por dispositivos. O DMA é um componente de hardware que permite a transferência direta de dados entre dispositivos periféricos e a memória principal, tornando assim dispensável a participação da CPU neste processo. O controlador de DMA é um hardware desenvolvido para desempenhar toda a sequência de transferência de dados acessando diretamente a memória. Ele gerencia vários canais que podem ser programados para realizar a transferência de dados, quer seja de um dispositivo para a memória ou vice-versa. O SO somente pode usar o DMA se o hardware tem o controlador de DMA. O DMA melhora o desempenho do sistema, pois poupa tempo ocioso da CPU, que poderia muito bem executar a tarefa do DMA, porém como o tempo de E/S é grande principalmente grandes quantidades de dados pode fazer com que a CPU fique muito tempo ociosa. Quando a quantidade é pequena, às vezes é até mais viável fazer a transferência direta pela CPU que é um hardware mais rápido que o DMA, isso pode causar concorrência no barramento, pois o barramento utilizado pelo DMA para acessar a memória é o mesmo utilizado pela CPU. Utilizando o DMA, a CPU requisita ao DMA de onde começar a ler os bytes, quantos bytes devem ser lidos/escritos e fica livre para executar outras tarefas que sejam CPU Bound, então quando o DMA termina de realizar sua tarefa, ele transmite um sinal de interrupção para a CPU que simplesmente usa os bytes. Note que a CPU pode fazer exatamente o que o DMA faz, isso fica a cargo do projeto. Coloca uma requisição de leitura no barramento e fica esperando até receber os bytes e assim poder usá-lo, a diferença é que usando a CPU para transferência de uma quantidade maior de dados, poderá ocasionar em CPU ociosa.

Pedido de Interrupção (IRQ)

O Sistema Operacional (SO) chaveia entre os aplicativos ativos para que o usuário tenha a sensação de que estes estão executando em paralelo. O SO permite que um aplicativo utilize o processador durante um determinado período de tempo e então permite que outra aplicação o utilize. Como o chaveamento é feito de uma forma muito rápida temos a impressão de que os aplicativos estão sendo executados ao mesmo tempo. Um pedido de interrupção (abreviação IRQ (em inglês)) é a forma pela qual componentes de hardware requisitam tempo computacional da CPU. Um IRQ é a sinalização de um pedido de interrupção de hardware. Por exemplo: caracteres digitados no teclado, operações de leitura sobre o HD, dados recebidos pelo modem ou mesmo movimentos do mouse devem ser executados mesmo que a máquina esteja processando alguma tarefa. Dessa forma IRQ's são canais de comunicação com o processador. Ao receber um pedido através de algum destes canais, o processador percebe a solicitação de interrompimento de um dispositivo. Quando um programa de usuário emite uma chamada ao sistema, esta é encaminhada ao driver apropriado. Para evitar que a CPU fique ocupada interrogando se o dispositivo terminou a operação de E/S (espera ociosa), o driver pede ao dispositivo que o sinalize quando isso ocorrer. Dessa forma, o Sistema Operacional poderá executar outras tarefas enquanto o programa que o chamou pedindo o serviço se encontra bloqueado. Ao final da operação, o controlador do dispositivo gera uma interrupção, ao chip controlador de interrupção, para sinalizar à CPU. Caso nenhuma outra interrupção esteja pendente ou em tratamento e nenhum outro dispositivo fez uma requisição de maior prioridade no mesmo momento, a interrupção é tratada imediatamente. Caso contrário, ela é ignorada, e o dispositivo continuará emitindo sinal de interrupção. Após executar uma instrução, a CPU verifica as linhas de interrupções para saber se alguma interrupção foi sinalizada. Caso tenha sido sinalizada uma interrupção, o hardware grava os registradores da CPU na pilha do programa que estava em execução e carrega o contador de programa com o endereço da rotina referente à interrupção sinalizada. Interrupções podem ser geradas por hardware ou por software. No caso do hardware, pelos dispositivos periféricos ou pelo relógio (timer). As interrupções geradas por software são conhecidas como system calls (chamadas ao sistema) ou trap (armadilha). A diferença primordial entre as interrupções geradas por software e as geradas por hardware, está no fato de que, conhecendo um programa e seus dados, é possível prever quando as interrupções de software irão acontecer - o que não é possível no caso das interrupções de hardware.

Resumo

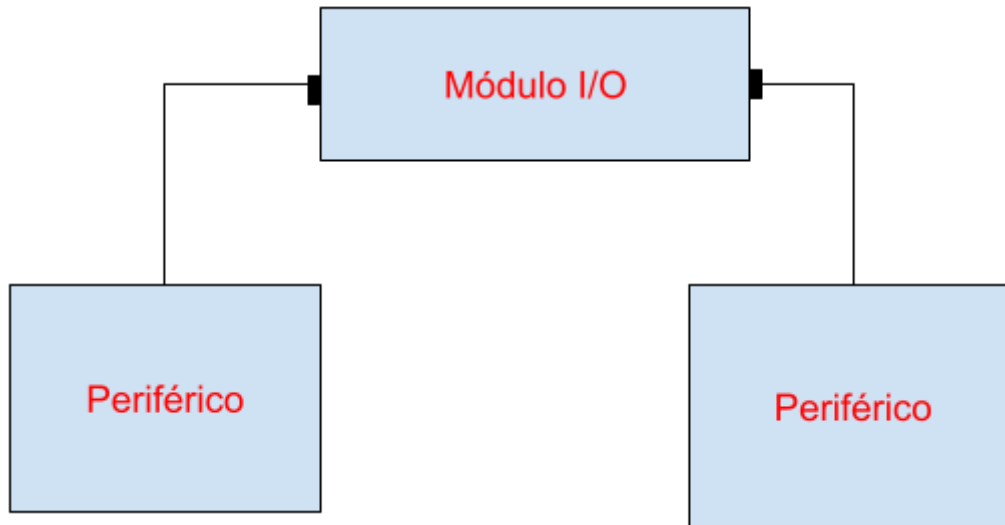
Dentro das funcionalidades dos sistemas operacionais está o gerenciamento de recursos que incluem os dispositivos de E/S, por serem recursos limitados, eles precisam ser devidamente gerenciados e compartilhados entre os processos executando e utilizado concorrentemente. Os Sistemas Operacionais emitem comando de leitura e escrita e sinal de interrupção, esses são emitidos pela CPU, para indicar que esse evento precisa de atenção da CPU ela aciona o SO. Assim sendo, a CPU está seguindo os algoritmos que estavam sendo executados, porém é emitido o sinal de interrupção, que é um sinal de suma importância que precisa ser tratado, pode ser a chegada de dados de leitura. Além disso, tratamentos de erros podem ser ocasionados. Os sistemas de E/S, devem fornecer uma interface única de forma que o Sistema operacional possa prover um conjunto de operações que fornecida para usuários e drives propiciando controle de forma organizada e única para todos os dispositivos de E/S. Sabendo que boa parcela do sistema operacional do Kernel do sistema operacional corresponde aos drives e a maior parte do tempo de um sistema operacional ocorre nos dispositivos de E/S, a computação e processamento são instruções bastantes rápidas pois ocorrem dentro da CPU e em contrapartida os dispositivos de E/S possuem a parte mecânica o disco possui a parte mecânica, portanto há de sincronizar o tempo com a CPU de forma que os dados possam chegar a tempo para serem processados por haver overhead assíncrona em relação

ao tempo dos dispositivos e processamento CPU.

Atividade de Fixação dos Conceitos

1. Quais os comandos que o sistema operacional deve gerenciar no sistemas de dispositivos de E/S?
Emitir instruções de leitura e escrita, interceptar interrupção e tratamentos de erros.
2. Defina interrupção?
Sinal emitido do sistema operacional de alerta a CPU, indicando um evento que ela deverá prestar atenção. Então ele interrompe a sua rotina e executa o procedimento de evento que ele irá tratar, a partir das rotinas a serem realizadas sistematicamente.
3. Escreva um evento que possa gerar uma interrupção:
Instrução de leitura dos dados do disco, a Controladoria do disco no término da leitura dos dados emitirá um sinal de interrupção indicando que o procedimento em forma de solicitação realizada foi concluído.
4. Classifique E/S em tipos de taxonomia
tipo de conexão, transferência de dados e compartilhamento de conexão.
5. Qual é a diferença entre o tipo de conexão seria e paralela?
Serial: apresenta uma única linha de conexão entre o modo e o periférico, os dados são transmitidos um bit após a outros, são mais baratas, lentas e relativamente confiáveis, utilizadas em dispositivos baratos e lentos como por exemplo: impressora.
paralelas. São várias linhas e bits de dados na conexão, conjunto de fios que indicam a capacidade de transmissão paralela onde a conexão, são complexas, rápidas, mais caras e utilizadas em dispositivos mais velozes: altamente confiáveis exemplo disco.
6. Quanto ao tipo transmissão, como é o comportamento dos dispositivos de E/S?
Pode ser em bloco: permite a transmissão um ou mais blocos de tamanho fixo cada um com determinado endereço que pode variar entre 128 e 1048 bytes.
Carácter: fluxo de carácter, não são endereçáveis e não podem fazer acesso aleatório.
7. Conceitue controlador de E/S:
Os controladores são dispositivos de hardware que manipulam diretamente os dispositivos de entrada e saída. O device driver comunica-se com os dispositivos indiretamente através do controlador. Existe assim um controlador para cada tipo de dispositivo físico de entrada /saída (disco, impressora, fita, etc.). Em geral o controlador pode ser um placa independente acoplada a CPU por um slot ou pode ser implementado diretamente na placa do processador.
8. Classificação dos dispositivos de E/S quando ao compartilhamento de conexão: diagrama
ponto a ponto: conexão é mais simples, existe uma linha dedicada entre a conexão e o dispositivo, maior confiabilidade. maior grau de paralelismo. protocolo na rede que utilizam esse mec
multiponto: cabo compartilhado com todos os periféricos não têm a capacidade de realizar o paralelismo

Ponto-a-Ponto



Multiponto

