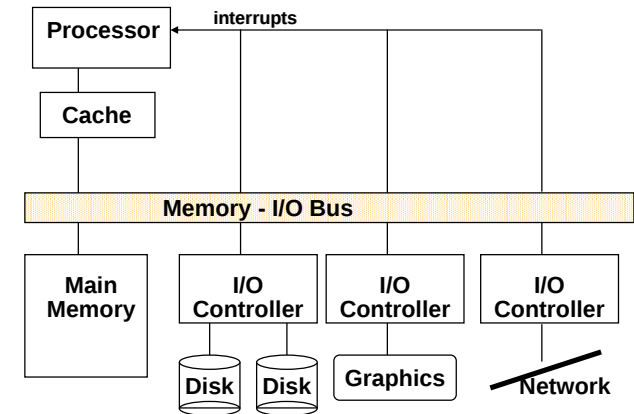
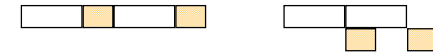


Introdução a Teoria das Filas e Interfaces de I/O

Sistemas de I/O

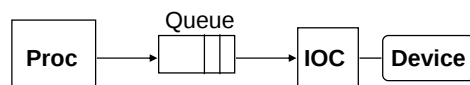
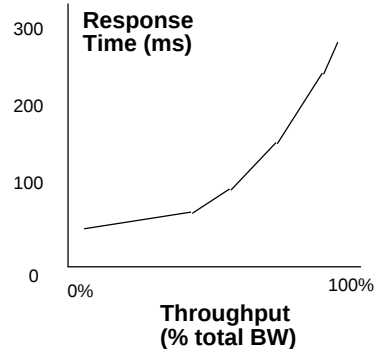


$$\text{Time(workload)} = \text{Time(CPU)} + \text{Time(I/O)} - \text{Time(Overlap)}$$



Performance do Sistema de Disco

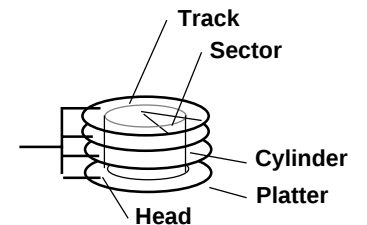
Métricas: Tempo de resposta e “throughput”



$$\text{Response time} = \text{Queue} + \text{Device Service time}$$

Dispositivos: Discos Magnéticos

- Objetivos:
 - Armazenamento não-volátil por tempo longo
 - Alta capacidade
- Características:
 - Tempo de acesso alto
 - Seek time, rotational delay
 - Transfer rate (1 setor/ms)
- Capacidade
 - Gigabytes



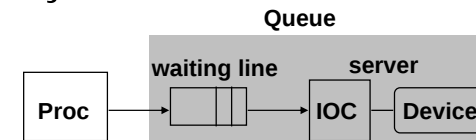
3600 RPM = 60 RPS $\Rightarrow$ 16 ms per rev
 ave rot. latency = 8 ms
 32 sectors per track $\Rightarrow$ 0.5 ms per sector
 1 KB per sector $\Rightarrow$ 2 MB / s
 32 KB per track
 20 tracks per cyl $\Rightarrow$ 640 KB per cyl
 2000 cyl $\Rightarrow$ 1.2 GB

$$\text{Tempo de resposta} = \text{Fila} + \text{Controlador} + \text{Seek} + \text{Rot} + \text{Xfer}$$

Exemplo de Performance de Disco

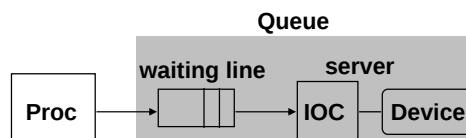
- Parâmetros:
 - Tamanho de transferência é de 8K bytes
 - Tempo médio de seek é de 12 ms
 - Disco roda a velocidade de 7200 RPM
 - Taxa de transferência é de 4 MB/sec
- Overhead do controlador é de 2 ms
- Pressuponha disco está parado (fila está inicialmente vazia)
- Qual é o tempo médio para acesso de disco p/ setor?
 - Seek + rot. Delay + taxa transferência + overhead do controlador
 - $12 \text{ ms} + 0.5 / (7200 \text{ RPM} / 60) + 8 \text{ KB} / 4 \text{ MB/s} + 2 \text{ ms}$
 - $12 + 4.15 + 2 + 2 = 20 \text{ ms}$
- Tempo de seek não assume nenhuma localidade: tipicamente 1/4 a 1/3 de redução 20 ms => 12 ms

Introdução a Teoria das Filas



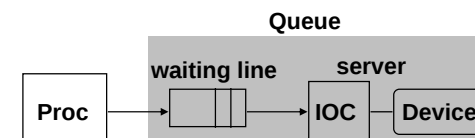
- Na verdade tempo de serviço deve considerar também período de espera devido a requisições chegando ao controlador aleatoriamente
- Modelo utilizado: **single server queue** - combinação de modelo contendo um servidor e uma área de espera: juntos chamados de fila
- Servidor gasta tempo aleatório com clientes. Como podemos caracterizar a variabilidade?

Introdução a Teoria das Filas



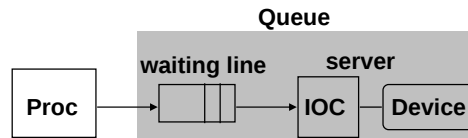
- Servidor gasta tempo aleatório com clientes
 - T_i – Tempo para execução da tarefa i
 - f_i – Frequência de execução da tarefa i
 - Média ponderada $m = (f_1 \times T_1 + f_2 \times T_2 + \dots + f_n \times T_n) / F$ ($F = f_1 + f_2 + \dots$)
 - Variância = $(f_1 \times T_1^2 + f_2 \times T_2^2 + \dots + f_n \times T_n^2) / F - m^2$
 - Squared coefficient of variance: $C = \text{variância} / m^2$**

Introdução a Teoria das Filas



- Squared coefficient of variance: $C = \text{variância} / m^2$**
- Distribuição exponencial **$C = 1$** : maioria dos tempos menores que a média; 90% < 2.3 x média, 63% < média
- Distribuição hipo-exponencial **$C < 1$** : tempos próximos à média, para $C=0.5 \Rightarrow 90\% < 2.0 \times \text{média}$, somente 57% < média
- Distribuição hiper-exponencial **$C > 1$** : tempos mais afastados da média, para $C=2.0 \Rightarrow 90\% < 2.8 \times \text{média}$, 69% < média

Introdução a Teoria das Filas : Tempo de Serviço Variável

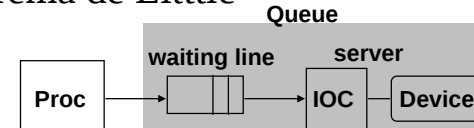


- Tempo de resposta de discos tem $C = 1.5$ (maioria dos tempos < média)
- Usualmente utiliza-se $C = 1.0$ (por simplicidade)
- Valor médio do tempo esperado para terminar a tarefa atual $m(z)$:
 - Se tempo de serviço das tarefas, m , fosse constante, esse tempo médio seria:

$$m(z) = 1/2 \times m$$

- Mas o tempo de serviço das tarefas não é constante
- Podemos derivar que este tempo médio de serviço da tarefa correntemente sendo servida (Average Residual Service Time) $m(z) = 1/2 \times m \times (1 + C)$
- Se não houver variância $\Rightarrow C = 0 \Rightarrow m(z) = 1/2 \times m$

Introdução a Teoria das Filas: Teorema de Little



- Modelo de filas assume estado de equilíbrio: taxa de entrada de requisições = taxa de atendimento

- Notação:

r número médio de chegada de requisições/segundo

T_s tempo médio de serviço de um cliente

u utilização do servidor (0..1): $u = r \times T_s$

T_w tempo médio/requisição na fila de espera

T_q tempo médio/requisição na fila: $T_q = T_w + T_s$

L_w tamanho médio da fila de espera: $L_w = r \times T_w$

L_q tamanho médio da fila: $L_q = r \times T_q$

- Lei de Little: $r = L_q / T_q = L_w / T_w = u / T_s$
Número médio de requisições = taxa de chegada x tempo médio de serviço

Introdução a Teoria das Filas : Tempo Médio de Espera

- Calculando tempo médio de espera T_w
 - Se algo estiver sendo executado no servidor, levará na média $m(z)$ para completar
 - Chance do servidor estar ocupado = u ; logo tempo médio será $u \times m(z)$
 - Depois, todas as outras requisições na fila de espera deverão completar, cada uma levando na média T_s

$$T_w = u \times m(z) + L_w \times T_s = 1/2 \times u \times T_s \times (1 + C) + L_w \times T_s$$

$$T_w = 1/2 \times u \times T_s \times (1 + C) + r \times T_w \times T_s$$

$$T_w = 1/2 \times u \times T_s \times (1 + C) + u \times T_w$$

$$T_w \times (1 - u) = T_s \times u \times (1 + C) / 2$$

$$T_w = T_s \times u \times (1 + C) / (2 \times (1 - u))$$

- Notação:

r número médio de chegada de requisições/segundo

T_s tempo médio de serviço de um cliente

u utilização do servidor (0..1): $u = r \times T_s$

T_w tempo médio/requisição na fila de espera

L_w tamanho médio da fila de espera: $L_w = r \times T_w$

Introdução a Teoria das Filas : Um Exemplo

- Processador envia 10 requisições/segundo de acesso a disco de 8KB, exponencialmente distribuídas, com tempo de serviço de disco = 20 ms
- Na média, como o disco é utilizado?
 - Número médio de requisições na fila de espera?
 - Tempo médio gasto na fila de espera?
 - Tempo de resposta médio para uma requisição?

- Notação:

r número médio de chegada de requisições/segundo = 10

T_s tempo médio de serviço de um cliente = 20 ms

u utilização do servidor (0..1): $u = r \times T_s = 10/s \times 0.02s = 0.2$

T_w tempo médio/requisição na fila de espera = $T_s \times u / (1 - u)$

$$= 20 \times 0.2 / (1 - 0.2) = 20 \times 0.25 = 5 \text{ ms}$$

T_q tempo médio/requisição na fila: $T_q = T_w + T_s = 5 \text{ ms} + 20 \text{ ms} = 25 \text{ ms}$

L_w tamanho médio da fila de espera: $L_w = r \times T_w = 10/s \times .005s = 0.05$

L_q tamanho médio da fila: $L_q = r \times T_q = 10/s \times .025s = 0.25$

Introdução a Teoria das Filas : Outro Exemplo

- Processador envia agora 20 requisições/segundo de acesso a disco de 8KB, exponencialmente distribuídas, com tempo de serviço = 12 ms
- Na média, como o disco é utilizado?
 - Número médio de requisições na fila de espera?
 - Tempo médio gasto na fila de espera?
 - Tempo de resposta médio para uma requisição?
- Notação:

r número médio de chegada de requisições/segundo = 20
 T_s tempo médio de serviço de um cliente = 12 ms
 u utilização do servidor (0..1): $u = r \times T_s = 20/s \times 0.012s = 0.24$
 T_w tempo médio/requisição na fila de espera = $T_s \times u / (1 - u)$
 $= 12 \times 0.24 / (1 - 0.24) = 12 \times 0.32 = 3.8$ ms
 T_q tempo médio/requisição na fila: $T_q = T_w + T_s = 3.8ms + 12$ ms = 16 ms
 L_w tamanho médio da fila de espera: $L_w = r \times T_w = 20/s \times .0038s = 0.076$
 L_q tamanho médio da fila: $L_q = r \times T_q = 20/s \times 0.016s = 0.32$

Introdução a Teoria das Filas : Mais Outro Exemplo

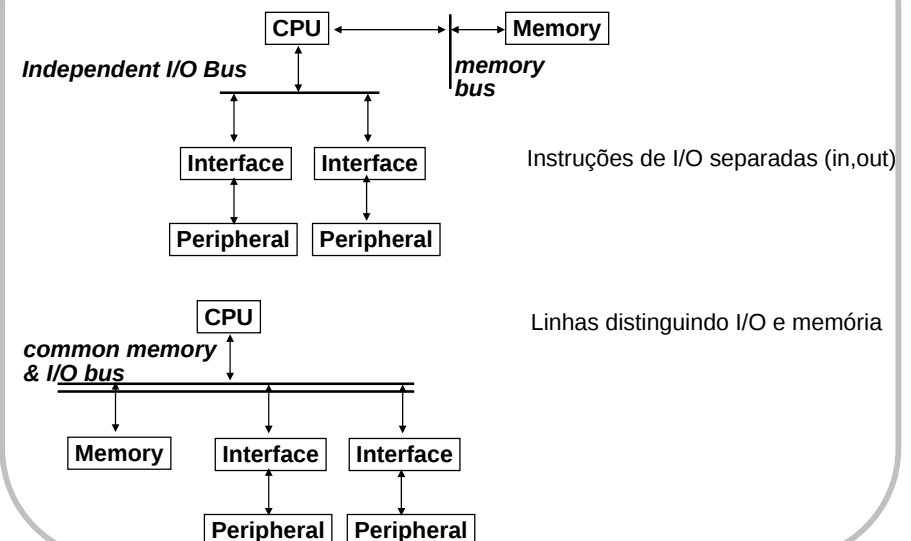
- Processador envia 10 requisições/segundo de acesso a disco de 8KB, *squared coef. var.* = 1.5, tempo de serviço de disco = 20 ms
- Na média, como o disco é utilizado?
 - Número médio de requisições na fila de espera?
 - Tempo médio gasto na fila de espera?
 - Tempo de resposta médio para uma requisição?
- Notação:

r número médio de chegada de requisições/segundo = 10
 T_s tempo médio de serviço de um cliente = 20 ms
 u utilização do servidor (0..1): $u = r \times T_s = 10/s \times 0.02s = 0.2$
 T_w tempo médio/requisição na fila de espera = $T_s \times u \times (1 + C) / (2 \times (1 - u))$
 $= 20 \times 0.2(2.5) / (2(1 - 0.2)) = 20 \times 0.32 = 6.25$ ms
 T_q tempo médio/requisição na fila: $T_q = T_w + T_s = 6.25ms + 20$ ms = 26 ms
 L_w tamanho médio da fila de espera: $L_w = r \times T_w = 10/s \times .006s = 0.06$
 L_q tamanho médio da fila: $L_q = r \times T_q = 10/s \times 0.026s = 0.26$

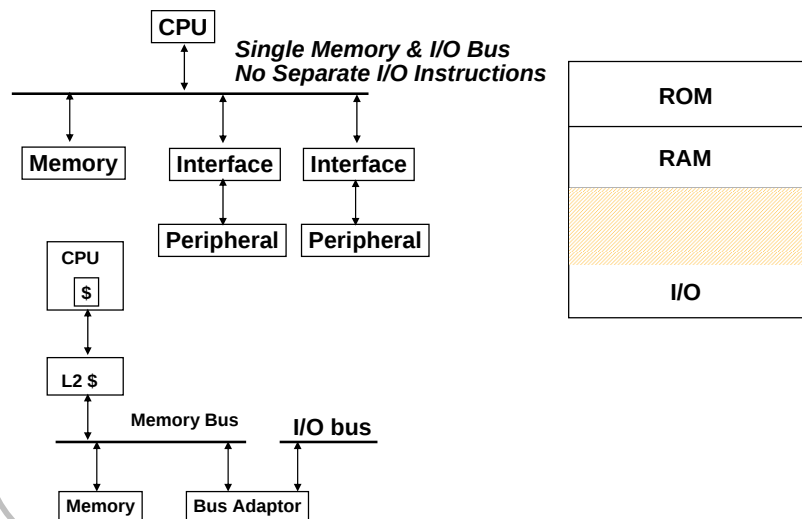
Interfaces com o Processador

- Interconexões
 - Barramentos
- Interface do Processador
 - Interrupções
 - I/O mapeado em memória
- Estruturas de Controle de I/O
 - Polling
 - Interrupções
 - DMA
 - Controladores de I/O
 - Processadores de I/O
- Capacidade, Tempo de acesso, BW

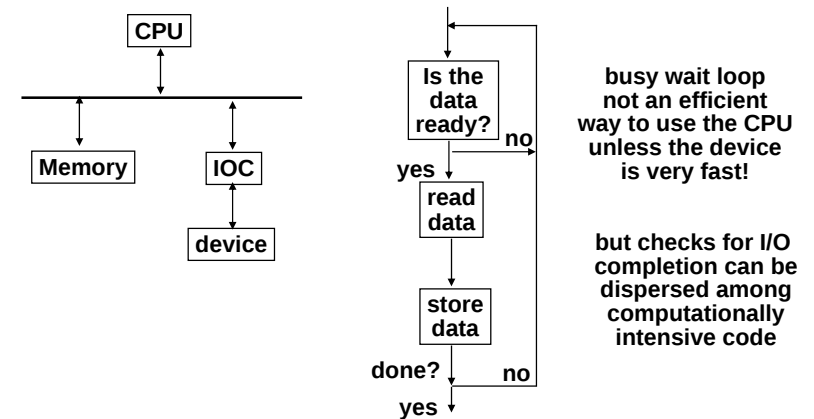
Interface com I/O



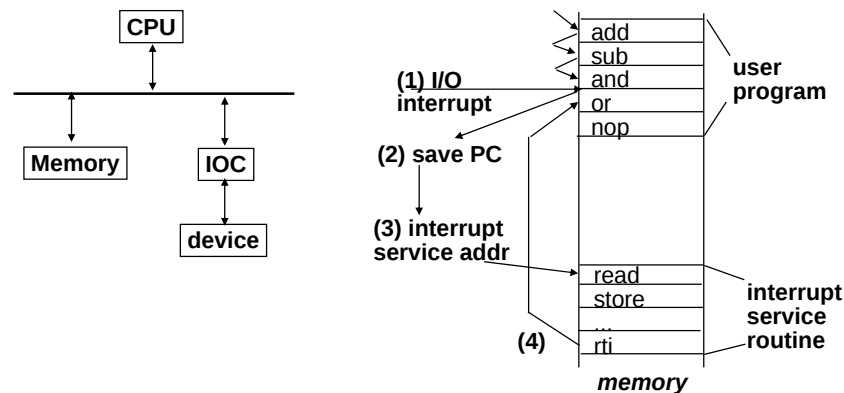
I/O Mapeada em Memória



I/O Programada (*Polling*)



Transferência de Dados por Interrupção



Programa de usuário fica parado durante hora da transferência

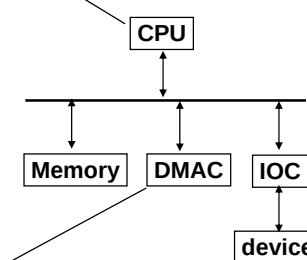
Acesso Direto à Memória

Tempo para fazer 1000 xfers a cada 1 msec cada:

- 1 DMA set-up @ 50 μ sec
- 1 interrupção @ 2 μ sec
- 1 serviço de interrupção @ 48 μ sec

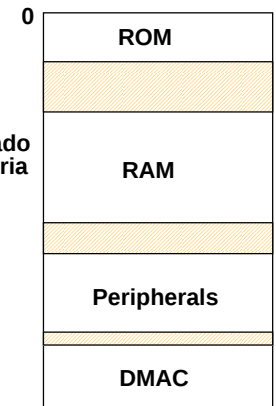
.0001 segundo do tempo de CPU

CPU sends a starting address, direction, and length count to DMAC. Then issues "start".

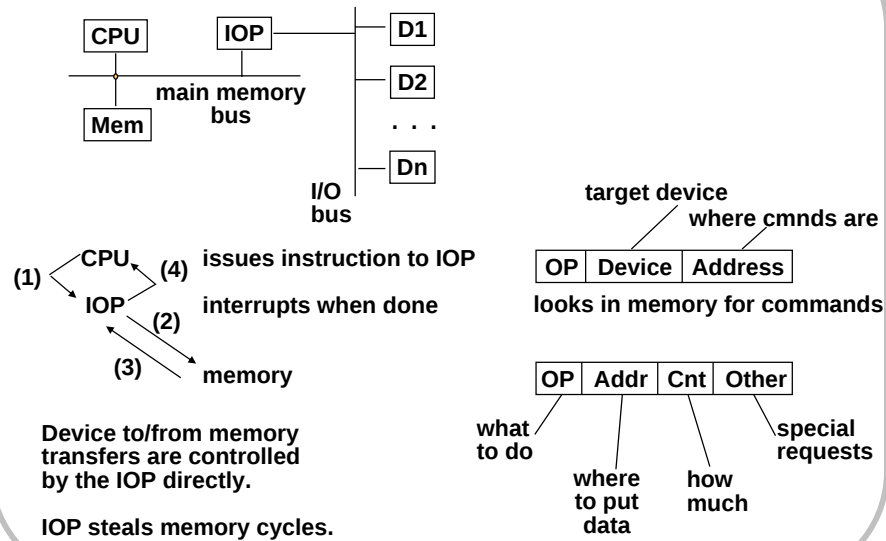


DMAC provides handshake signals for Peripheral Controller, and Memory Addresses and handshake signals for Memory.

I/O mapeado em memória



Processadores de I/O



Relação de I/O com Arquitetura do Processador

- Instruções de I/O especiais e barramentos praticamente desapareceram
- Vetores de interrupção foram substituídos por *jump tables*

$$PC \leftarrow M[IVA + \# \text{ int}]$$

$$PC \leftarrow IVA + \# \text{ int}$$
- Interrupções:
 - Pilha utilizada para guardar contexto
 - Handler* salva registradores e re-habilita interrupções de mais alta prioridade
 - Tipos de interrupção reduzidos em número; *handler* questiona controlador de I/O

Relação de I/O com Arquitetura do Processador

- Caches causam problemas para I/O
 - Flushing é caro, I/O polui cache
 - Solução é emprestada de sistemas multiprocessadores de memória compartilhada
- VM dificulta DMA
- Arquitetura Load/Store necessita de operações atômicas
 - load locked, store conditional
- Muito contexto fica difícil para salvar durante troca de contexto