

Fundamentos de Arquitetura de Computadores

Tiago Alves

Faculdade UnB Gama
Universidade de Brasília



Definição

Ligaçāo elétrica (ou caminho) compartilhada(o) por múltiplos dispositivos.

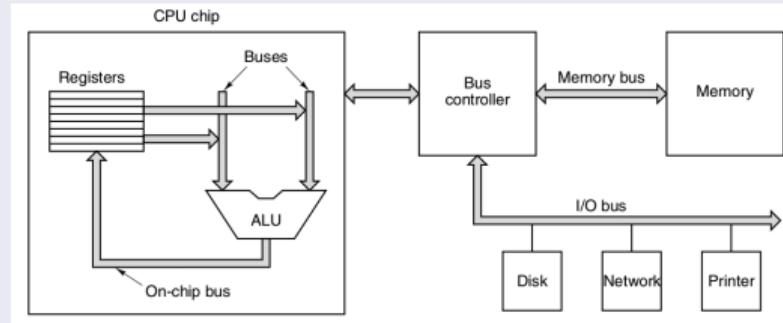
Categorias:

- internos à CPU, usado na troca de dados com a ULA.
- externos à CPU, usado para ligá-la à memória ou a dispositivos de I/O.



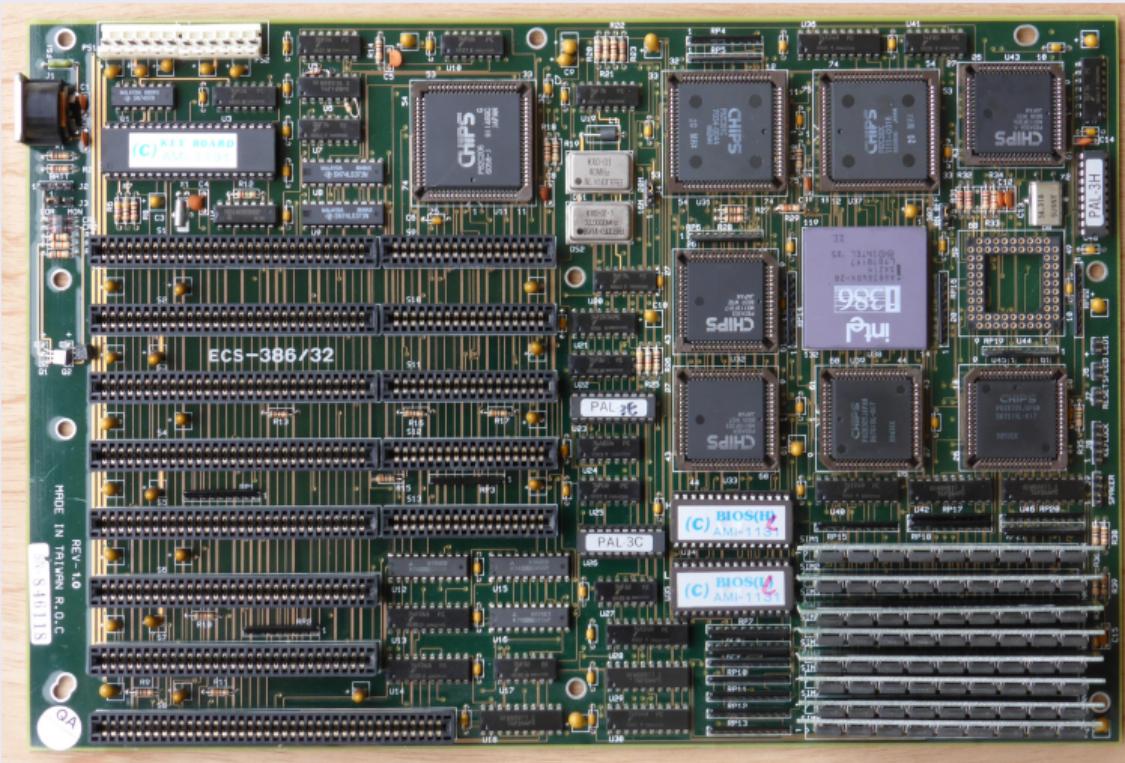
Primórdios da Estrutura em Sistemas Eletrônicos Computacionais

- Computadores pessoais primitivos possuíam apenas 1 barramento externo
- 50 a 100 linhas paralelas de cobre repousadas na placa-mãe
- Conectores eram espaçados a distâncias regulares. Eram substrato para a conexão de memórias e de placas/cartões de I/O
- Computadores modernos possuem barramento de propósito especial entre a CPU e a memória principal e outros barramentos para I/O.



Barrantos

Interfaceamento



Interfaceamento

- Fabricantes tem liberdade de definir a arquitetura das interconexões internamente a um sistema eletrônico (CPU)
- Para interligar esses sistemas eletrônicos na solução de um problema da vida real, um conjunto de regras de construção e sinalização dos barramentos deve existir: protocolos de barramentos
- São considerados aspectos mecânicos (físicos), elétricos e lógicos
- Alguns barramentos:
 - IBM PC bus (PC/XT)
 - ISA bus (PC/AT)
 - Microchannel (PS/2)
 - PCI bus
 - SCSI bus
 - USB
 - FireWire
 - PCI Express

Operação de Barramentos

Agentes

- ativos ou mestres
- passivos ou escravos

Exemplos

- troca de dados entre CPU e disco
- CPU (mestre) ordena o controlador de disco (escravo) a leitura ou escrita de um bloco
- na resposta à requisição, os papéis poderão mudar: controlador (mestre) envia à memória principal as words lidas do disco.

Intermediário elétrico: **bus drivers, receivers e transceivers**. Reforço dado ao dispositivo na interação com o barramento.



Operação de Barramentos

Master	Slave	Example
CPU	Memory	Fetching instructions and data
CPU	I/O device	Initiating data transfer
CPU	Coprocessor	CPU handing instruction off to coprocessor
I/O device	Memory	DMA (Direct Memory Access)
Coprocessor	CPU	Coprocessor fetching operands from CPU



Esquema lógico básico de um Barramento

Lembra uma CPU: possui endereços, dados e linhas de controle.

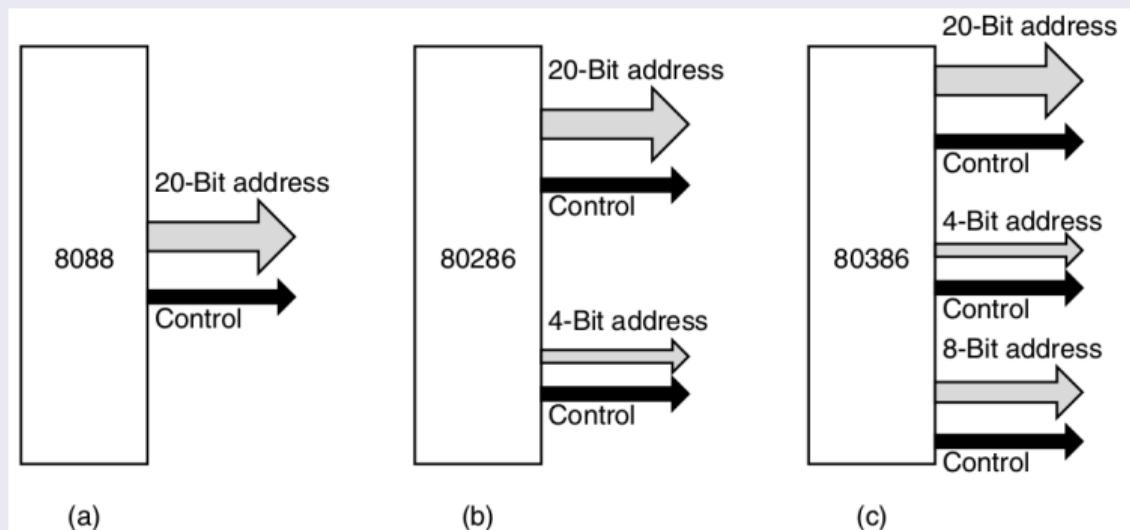
Largura de Barramento:

- Regra básica: quanto mais linhas de endereço possuir um barramento, mais memória poderá ser endereçada pela CPU
- Mais linhas de endereços implica mais linhas elétricas. Que demandam mais espaço físico e que, ao final, resultam em conectores grandes. Isso encarece projetos eletrônicos!



Largura de Barramento

Projeto da Intel de linhas de endereço e suas idiossincrasias: retrocompatibilidade vs. aumento de desempenho.



Largura de Barramento

O aumento de linha de dados tende a acompanhar o aumento no número de linhas de endereços

Busca por desempenho: aumento da largura de banda do barramento

- Estratégia 1: diminuir o período de relógio do barramento (mais transferências por segundo)
- Estratégia 2: aumentar a largura do barramento de dados (mais bits por transferência).

Essa busca esbarra em problemas de retrocompatibilidade e de compatibilidade eletromagnética

Solução de compromisso em algumas tecnologias: barramento multiplexado. Ora trafegam endereços, ora trafegam dados.



Sincronização de Barramento

Categorias de barramento:

- Síncronos: provê linha de sinal de relógio. Oscilador a cristal.

Todas as interações com o barramento devem respeitar a frequência de relógio.

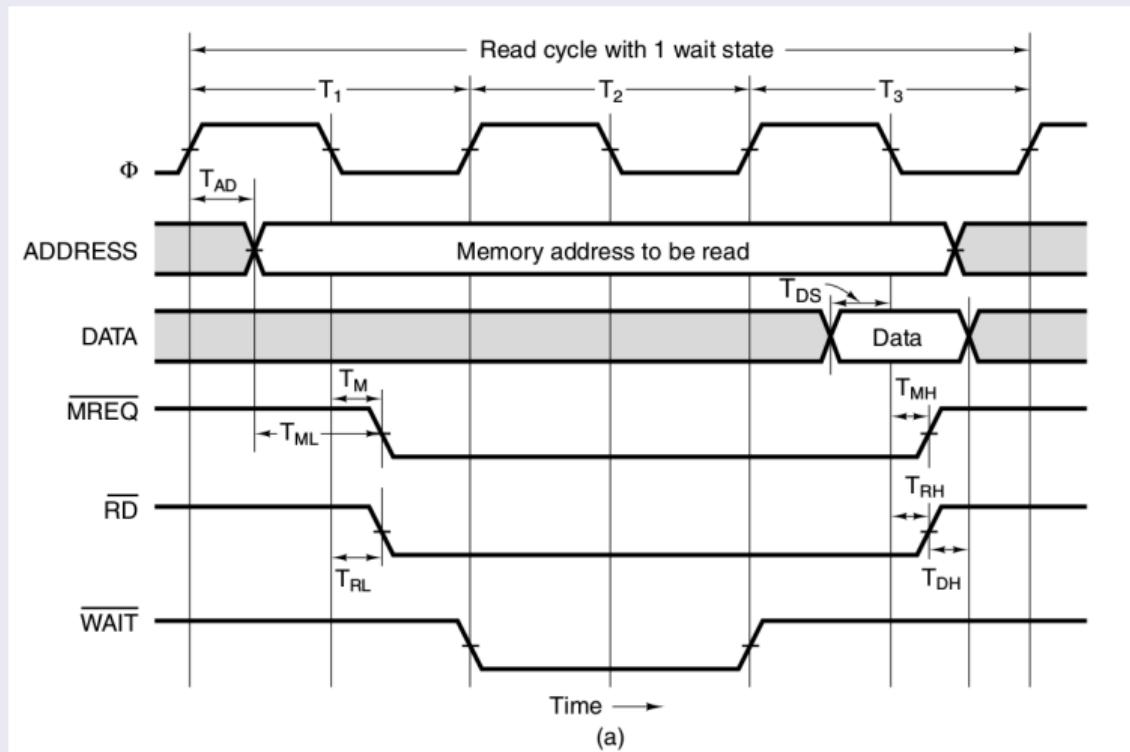
- Assíncronos: não possuem um sinal de relógio mestre/de referência.

Sinais de relógio poderão ser de qualquer comprimento e não precisam ser os mesmos entre todos os pares de dispositivos



Sincronização de Barramento

Síncrono (operação de leitura):



Sincronização de Barramento

Síncrono:

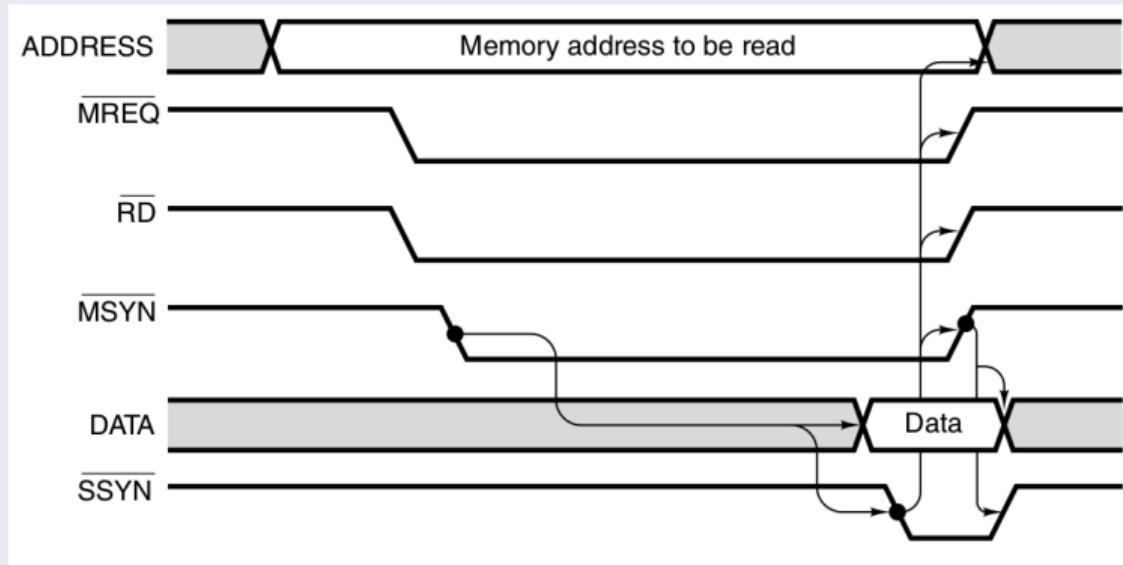
Symbol	Parameter	Min	Max	Unit
T_{AD}	Address output delay		4	nsec
T_{ML}	Address stable prior to \overline{MREQ}	2		nsec
T_M	\overline{MREQ} delay from falling edge of Φ in T_1		3	nsec
T_{RL}	RD delay from falling edge of Φ in T_1		3	nsec
T_{DS}	Data setup time prior to falling edge of Φ	2		nsec
T_{MH}	\overline{MREQ} delay from falling edge of Φ in T_3		3	nsec
T_{RH}	\overline{RD} delay from falling edge of Φ in T_3		3	nsec
T_{DH}	Data hold time from negation of \overline{RD}	0		nsec

(b)



Sincronização de Barramento

Assíncrono:



Controle (Arbitragem) de Barramento

O modelo em que há um mestre e um escravo é fácil de se analisar e de se implementar

Porém, a arquitetura mais frequente permite com que haja troca de papéis entre dispositivos a depender das transações:

- chips de IO necessitam se tornar mestres para lerem e escreverem na memória principal e para lançar interrupções
- coprocessadores (FPU) podem também precisar se tornarem mestres

Dois esquemas para controle de barramento no cenário em que mais de um dispositivo necessitar se tornar mestre:

- centralizado
- descentralizado



Controle (Arbitragem) de Barramento: Centralizado

Há um árbitro/juiz que decide quem controlará o barramento em determinado momento

Embutido em CPU ou, às vezes, acompanhante da CPU

Linha única de pedido de controle de barramento usada para sinalizar o pedido para a CPU (OR)

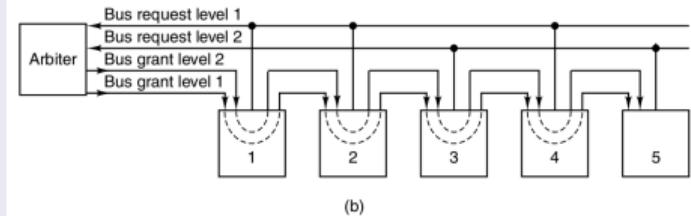
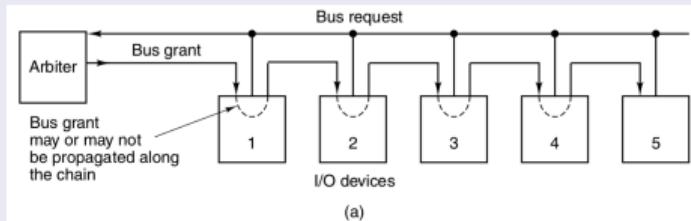
- múltiplos pedidos podem se sobrepor!
- duas opções/estados: sem pedidos ou com pedidos.



Controle (Arbitragem) de Barramento: Centralizado

Árbitro conectado aos dispositivos para os quais cederá controle (do barramento) através de uma linha serial interconectada

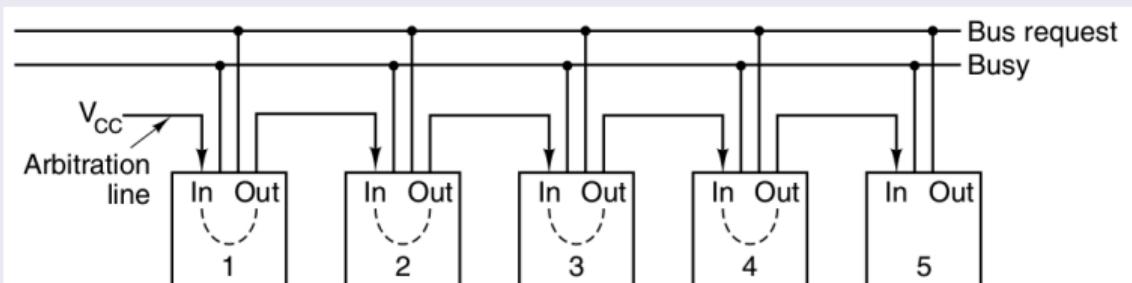
- O dispositivo mais próximo cuja demanda foi enviada recebe o sinal e controla o barramento.
- Se o dispositivo que recebeu o sinal não demandou controle, deve encaminhar o sinal de autorização pela linha serial para seu próximo vizinho.
- É possível alterar o esquema de interconexão para melhorar a distribuição de prioridades entre os dispositivos que compartilham o barramento.



Controle (Arbitragem) de Barramento: Descentralizado

Demanda aumento do número de linhas e uma sistemática de sensibilização dos dispositivos que compartilham o barramento

- Os dispositivos deverão monitorar as linhas e eventuais pedidos, controlando o barramento apenas quando julgarem ser a sua vez.
- Impõe um limite na razão entre o número de dispositivos e o número de linhas de requisição



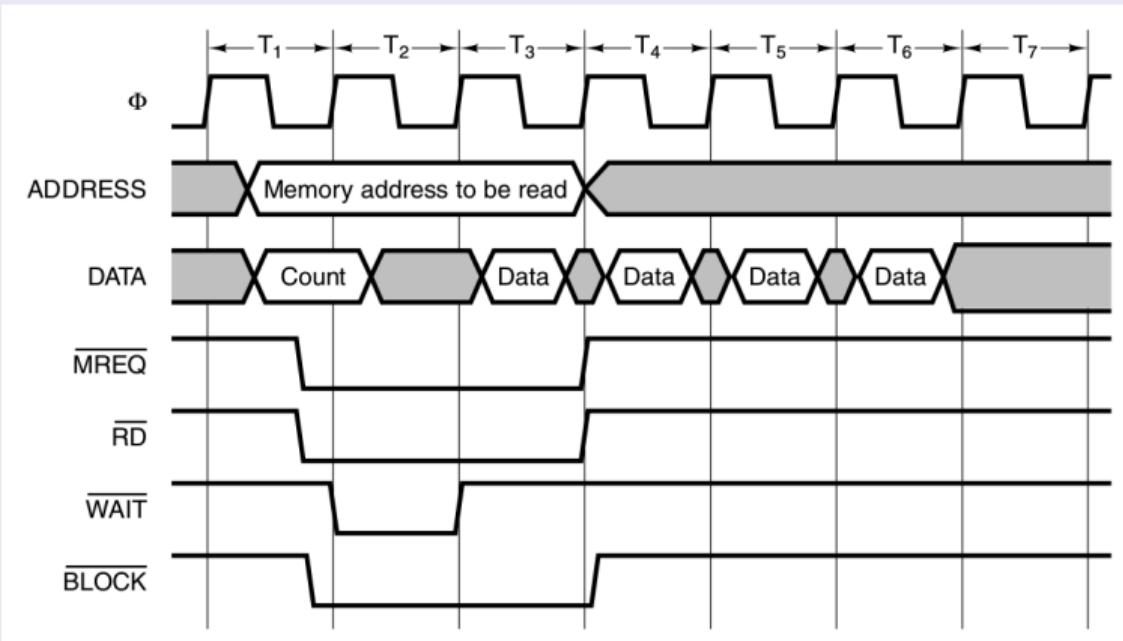
Operações em Barramento

Módulos construtivos que compõem sistemas eletrônicos trocam palavras através de barramentos.

- Quando esquemas de caching são empregados, as trocas costumam ser em blocos de palavras: aumenta a eficiência das transferências diluindo a latência do tempo de aquisição de controle do barramento.
- O mestre costuma indicar quantas palavras irá transmitir pelo barramento.



Operações em Barramento



Exemplos: PCI (Peripheral Component Interconnect Bus)

Barramento primário para IO em computadores

Antecessores:

- ISA (8.33 MHz, 2B/ciclo, 16.7 MB/s)
- EISA (8.33 MHz, 4B/ciclo, 33.3 MB/s)

Demanda por barramentos de alta velocidade quando o PC passou a ser plataforma para aplicações multimedia:
155 MB/s



Exemplos: PCI (Peripheral Component Interconnect Bus)

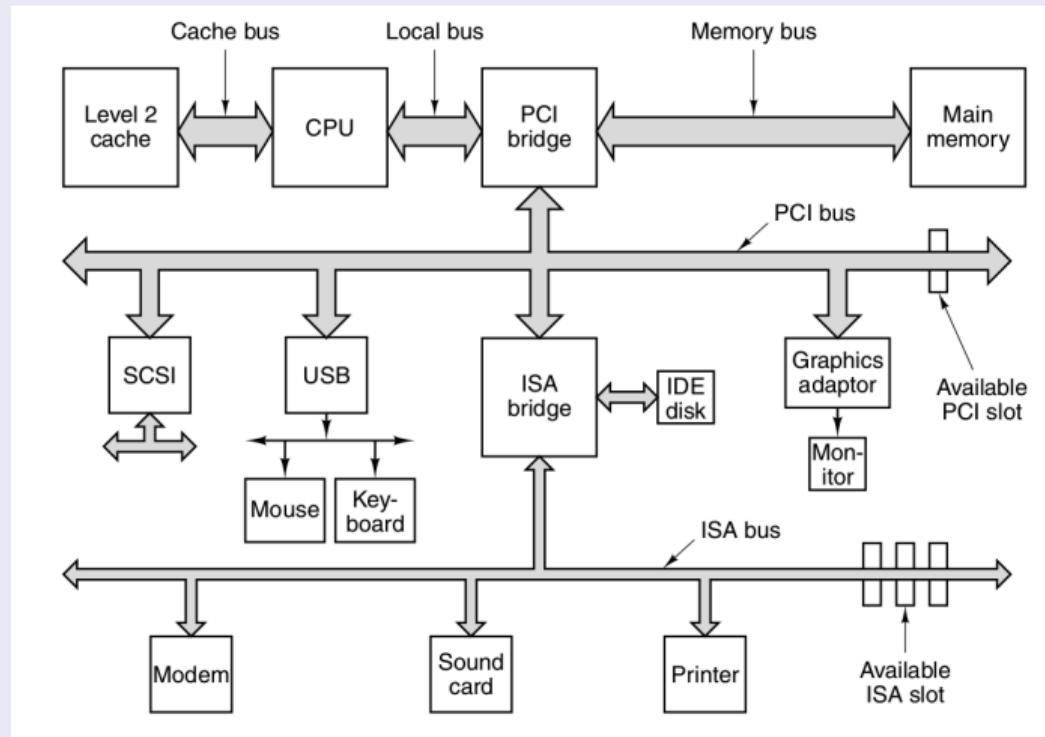
Duas versões: tradicional (1990, Intel) e o PCIe

- padrão aberto: Intel abriu mão das patentes
- 33 MHz, 32 bits (1 word) por ciclo \Rightarrow 133 MB/s
- 66 MHz, 64 bits (1 word) por ciclo \Rightarrow 528 MB/s
- Barramento síncrono



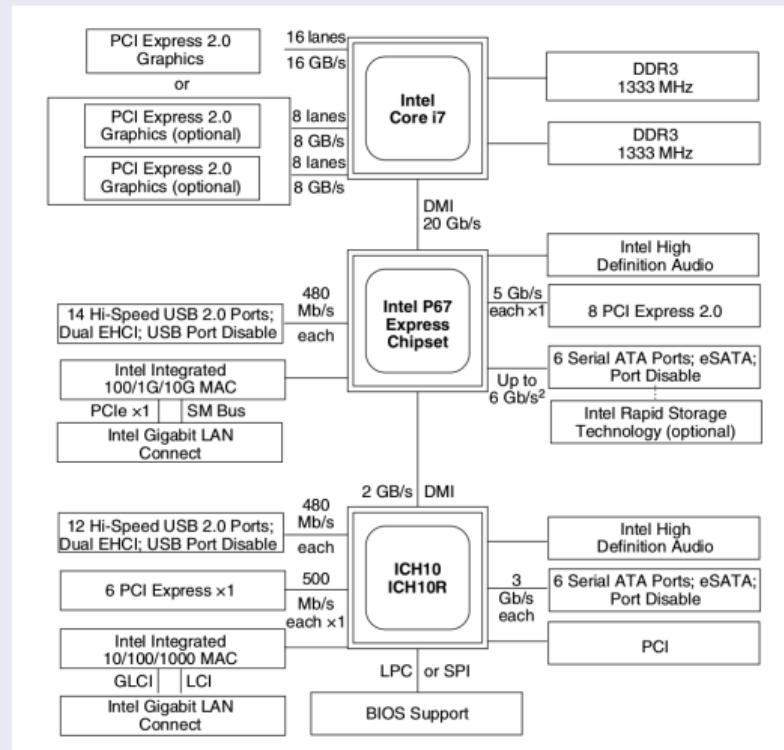
Exemplos: PCI (Peripheral Component Interconnect Bus)

Arquitetura clássica:



Exemplos: PCI (Peripheral Component Interconnect Bus)

Arquitetura contemporânea:



Exemplos: PCI (Peripheral Component Interconnect Bus)

Problemas:

- Não retrocompatível com antecessores.
- Apesar da velocidade, não era capaz de sustentar o desempenho esperado de um barramento para interligação da CPU com a memória principal.
- Demandou arquitetura multibarramento: um barramento de altíssima velocidade para interligação da CPU com elementos de tempo crítico e outros barramentos de interconexão com periféricos.



Exemplos: PCIe (PCI Express)

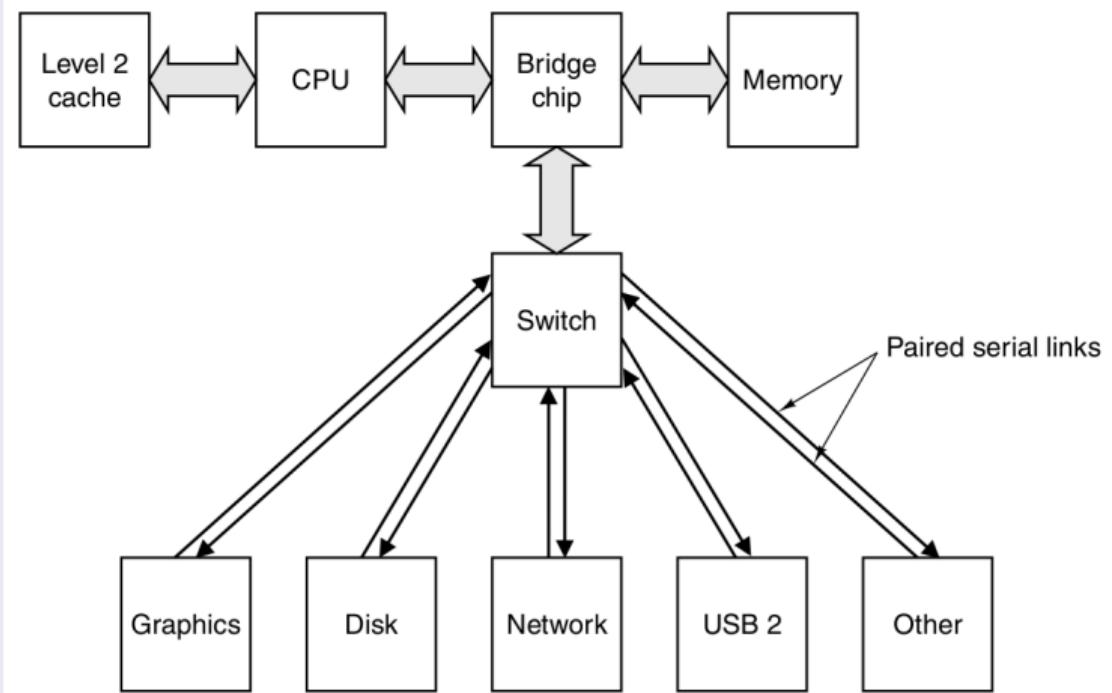
Atender demanda por mais largura de banda para sustentar IO:

Arquitetura

- substituir barramentos paralelos com muitos mestres e escravos (e seus problemas de controle/arbitragem)
- projeto baseado em **linhas de comunicação seriais ponto-a-ponto**: comutador que interliga os periféricos à CPU
- comunicação baseada em comutação de pacotes!
- códigos corretores de erro para proteger os dados trocados através das linhas
- conexão fisicamente curta: compatibilidade/imunidade eletromagnética
- expansível: cascata de switches (comutadores)
- **hot-plug!**
- conectores pequenos



Exemplos: PCIe (PCI Express)



Exemplos: USB (Universal Serial Bus)

Barramentos ora estudados:

- PCI e PCIe: excelentes para dispositivos de elevada velocidade, porém muito caro para dispositivos de IO de baixa velocidade como teclados e mouses.

Realidade: cada periférico a ser conectado ao PC demandava a abertura do gabinete e configurações (físicas e lógicas): jumpers, dip-switches, bios, drivers, conflitos de IRQ, reboot, ...



Exemplos: USB (Universal Serial Bus)

Objetivos perseguidos com a proposta do USB

- usuários não deverão ajustar dip-switches ou jumpers nas placas ou dispositivos
- usuários não deverão abrir o gabinete para a instalação de novos dispositivos de IO
- deverá haver apenas um tipo de cabo para conectar todos os dispositivos
- os dispositivos de IO deverão receber sinal de força pelo cabo
- até 127 dispositivos poderão ser conectados a um computador
- o sistema deverá suportar dispositivos de tempo-real (telefone, altofalantes, ...)
- os dispositivos poderão ser instalados enquanto o computador estiver em execução
- nenhuma reinicialização será necessária depois das instalação de um novo dispositivo
- o novo barramento e seus dispositivos de IO deverão ser viáveis (em termos de custo) para manufatura.



Exemplos: USB (Universal Serial Bus)

Versões:

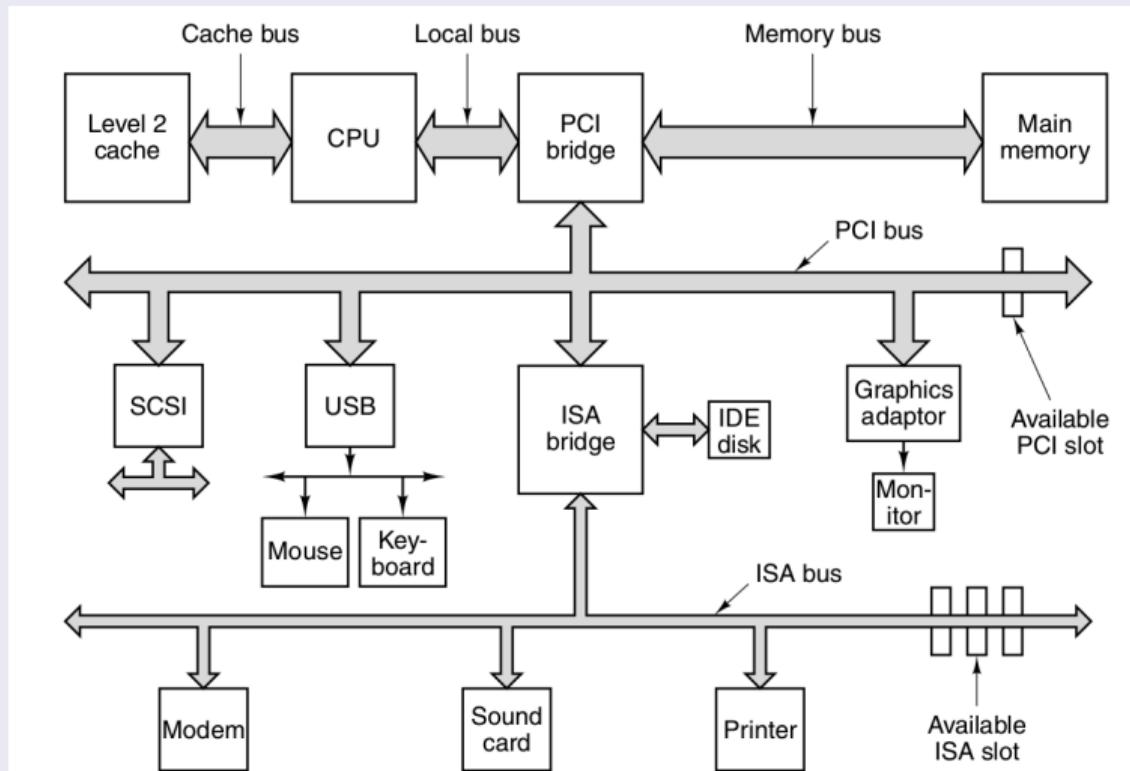
- 1.0: 1.5 Mbps
- 1.1: 12 Mbps
- 2.0: 480 Mbps
- 3.0: 5 Gbps

Arquitetura:

- root hub: conectado ao barramento principal do computador
- o root hub possui conectores para cabos que serão usados na conexão aos dispositivos de IO ou a hubs de expansão.
- conectores dos cabos possuem formatos físicos bem distintos para evitar conexões acidentais



Exemplos: USB (Universal Serial Bus)



Exemplos: USB (Universal Serial Bus)

Cabo:

- 4 linhas
- 2 de dados
- 2 de força/potência: +5 V e GND

