



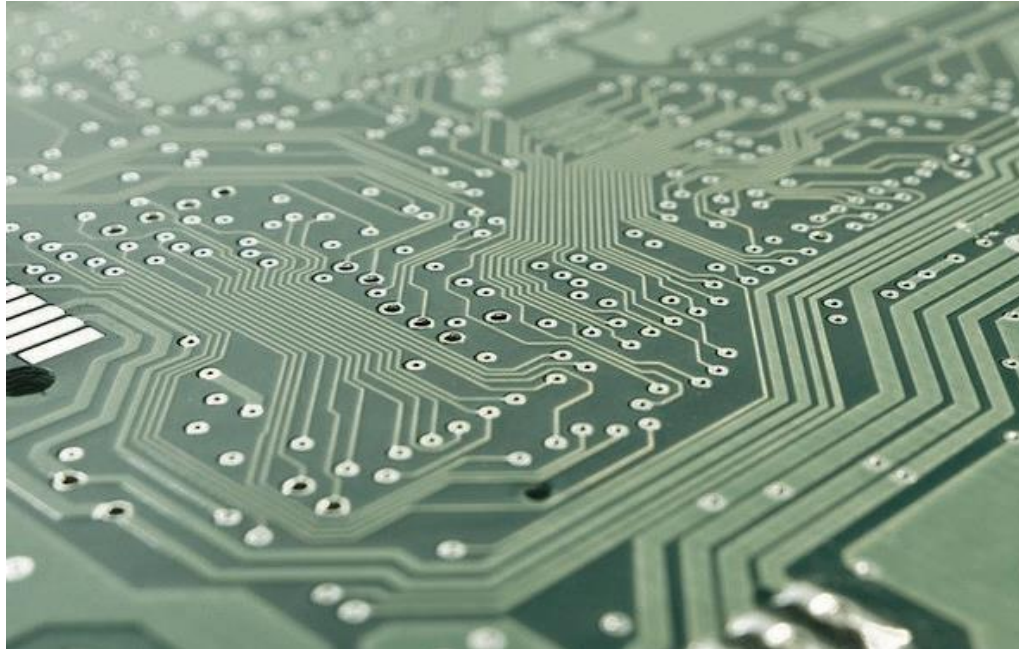
UNIDADE IV

Organização de Computadores

Prof. Dr. Alexandre Bozolan

Barramentos

- Um barramento é definido como um conjunto de fios de cobre, que tem como função realizar um “caminho” elétrico entre os dispositivos internos ou externos do computador.

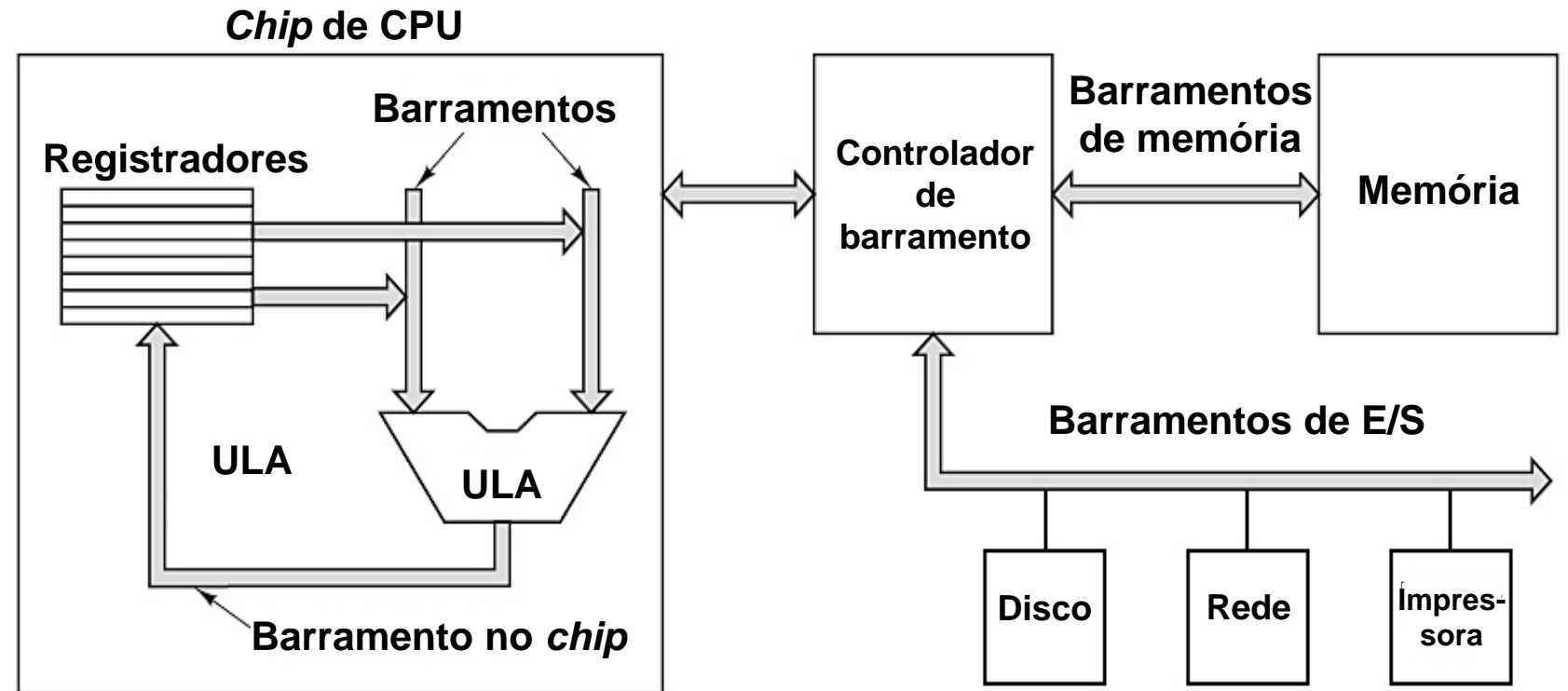


Fonte: pixcove.com

- Os barramentos são diferenciados de acordo com sua finalidade como no transporte de dados entre a ULA e a UC, a CPU e a memória principal ou mesmo entre outros dispositivos de E/S como disco rígido, CD, DVD, impressoras, *pendrive* etc.

Barramentos

- Os barramentos geralmente são representados por setas largas e sombreadas, indicando o fluxo para os dados.



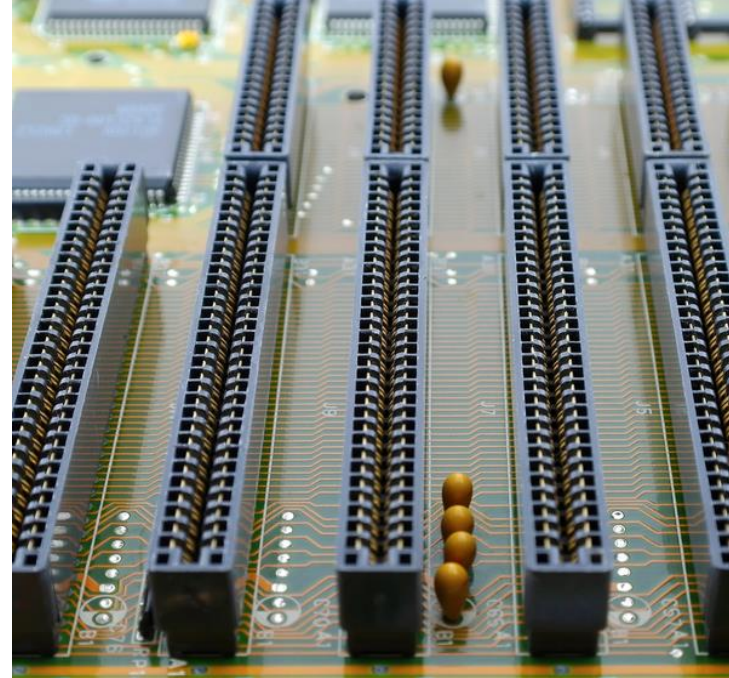
Fonte: TANENBAUM e AUSTIN (2011, p. 148).

Barramentos

- Os barramentos seguem uma certa padronização de protocolos que determinam especificações mecânicas (tensões mecânicas) e elétricas (temporização, largura de banda, frequência de operação) na fabricação das placas.
- Existem variedades de barramentos para uso em computadores, dentre os mais conhecidos estão: Omnibus (PDP-8), Unibus (PDP-11), Multibus (8086), barramento IBM PC (PC/XT), barramento ISA (PC/AT), barramento PCI, barramento AGP, entre outros.

Barramentos

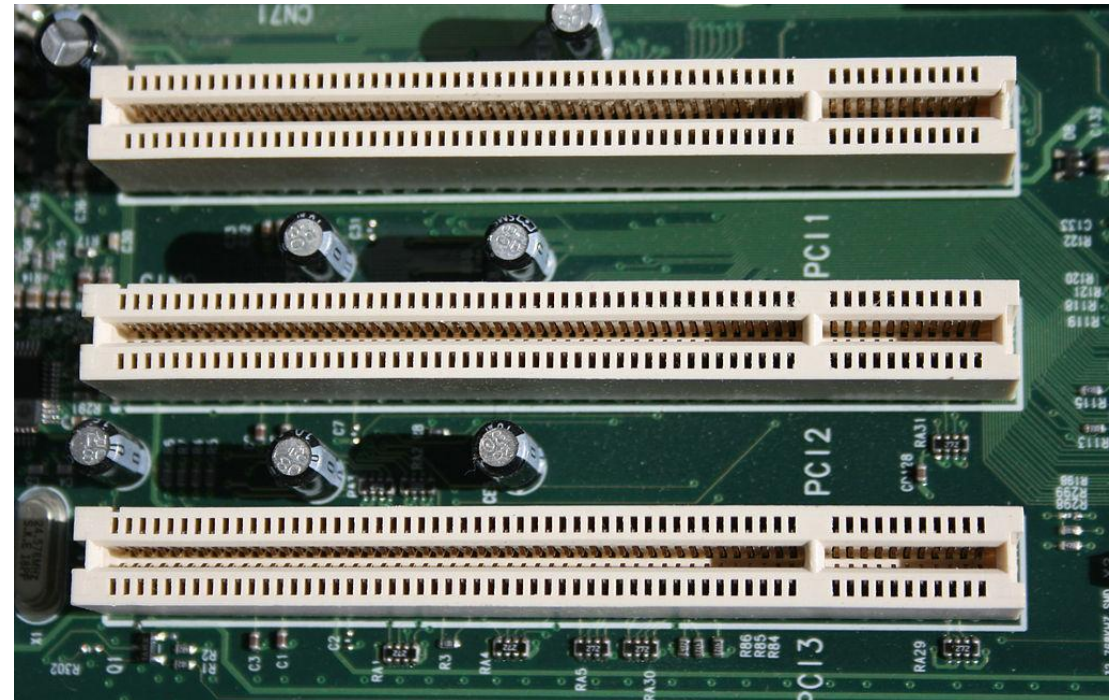
- O barramento ISA (*Industry Standard Architecture*) foi desenvolvido em meados dos anos 1980 para funcionar em computadores pessoais do tipo IBM PC/AT, equipados com processadores Intel 80286.
- Esse barramento possuía uma estrutura típica de 8 *bits* que foi projetada para substituir o padrão já ultrapassado VESA (*Video Electronics Standards Association*).



Fonte:
smallformfactors.mil-embedded.com

Barramentos

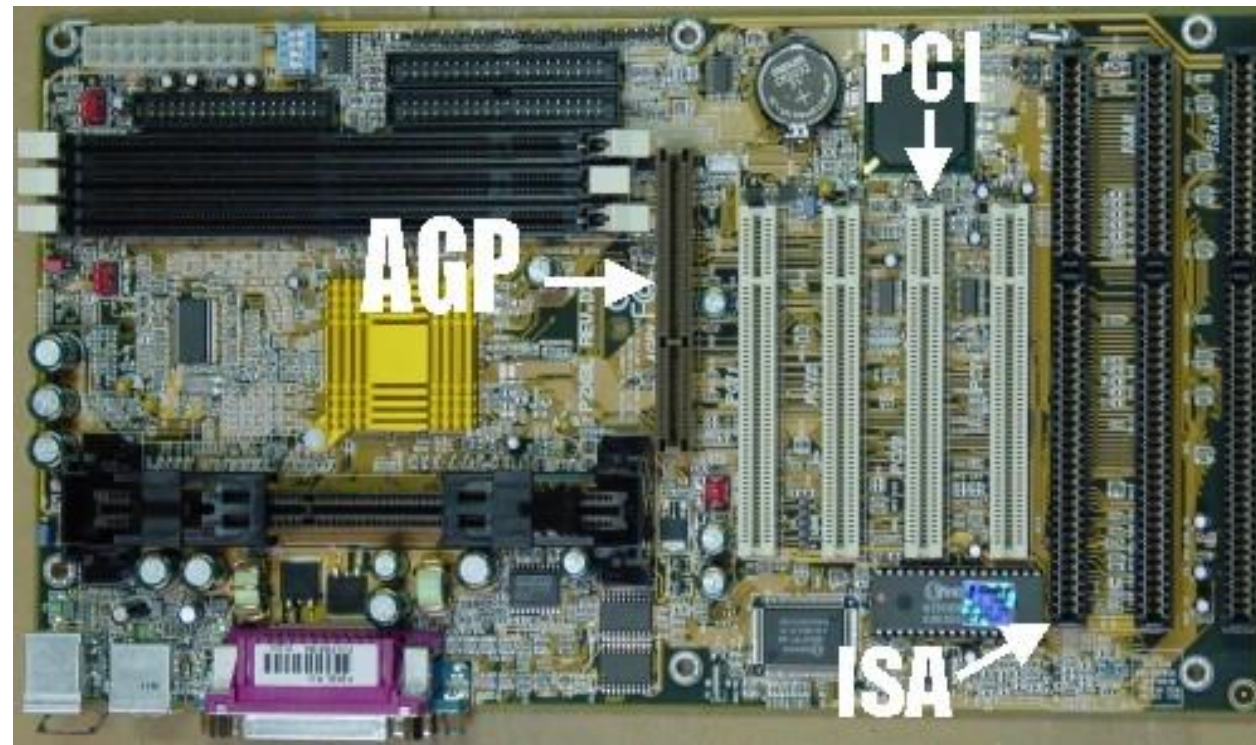
- O barramento PCI (*Peripheral Interconnect Bus* ou barramento de interconexão de componente periférico) foi desenvolvido em 1990 em substituição ao já ultrapassado barramento EISA (*Extended Industry Standard Architecture*).
- Originalmente, o PCI possuía a capacidade de transferência de instruções de 32 *bits*, barramento sincronizado por um *clock* de 33 MHz e largura de banda de 133 MB/s.



Fonte:
clubedohardware.com

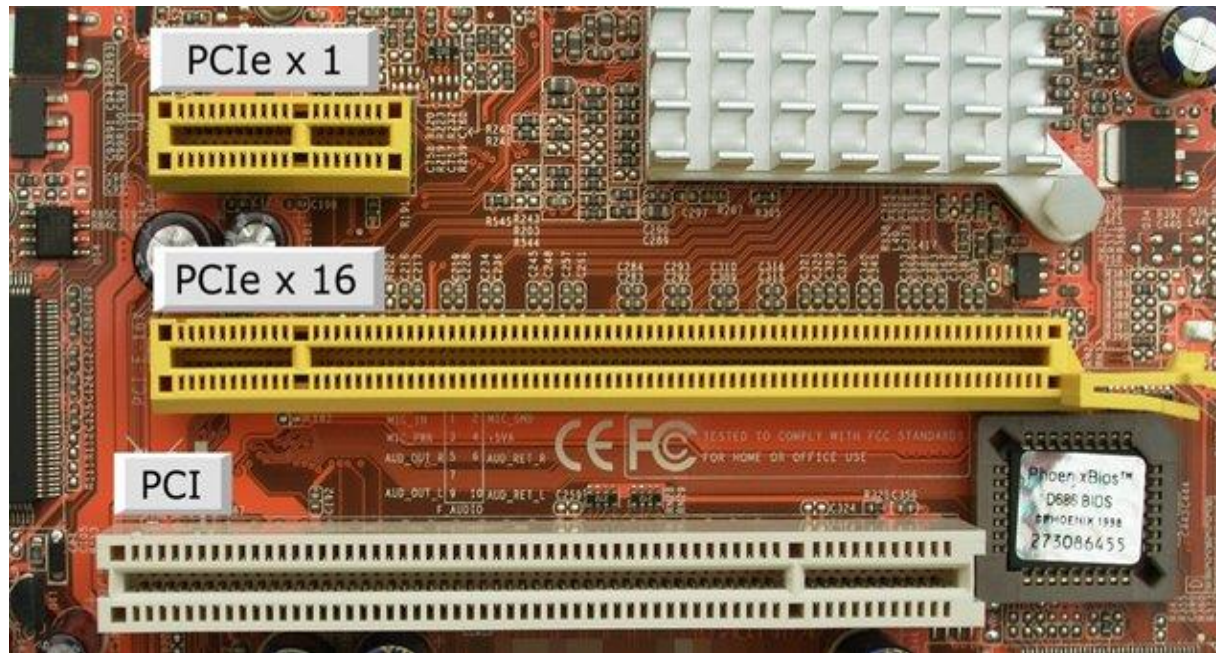
Barramentos

- No final dos anos 1990, o barramento padrão ISA estava praticamente desatualizado, portanto, novos projetos, além do aprimoramento do padrão PCI, estavam em desenvolvimento.
- Um novo tipo de barramento utilizado para comandar a placa gráfica foi fabricado pela Intel, denominado AGP (*Accelerated Graphics Port Bus* ou barramento de porta gráfica acelerada).



Barramentos

- O barramento AGP foi superado em desempenho por um novo tipo de barramento, denominado PCI Express ou PCIe, que possui a incrível capacidade de largura de banda de 16 GB/s para dados baseado em enlaces de barramento serial de alta velocidade.
- Uma das implementações PCIe foi eliminar o barramento paralelo constituído de muitos mestres e escravos e utilizar um projeto baseado em conexões seriais ponto a ponto de alto desempenho.



Barramentos

- Outra característica de funcionamento dos barramentos está relacionada à transferência ativa ou passiva de dados/instruções.
- Os barramentos ativos são conhecidos como mestres, enquanto os passivos denominados de escravos.
- Quando o processador ordena a uma controladora de disco que realize a leitura ou escrita de um bloco de palavras, a CPU está agindo como um mestre e o dispositivo que atua obedecendo a operação como escravo.

Barramentos

- Os sinais em binário, emitidos pelos dispositivos do computador, geralmente são fracos e não conseguem energizar o barramento, principalmente se o barramento for relativamente longo e com muitos dispositivos conectados nele.
- Assim, a maioria dos barramentos mestres está conectada por um *chip* denominado controlador de barramento, que opera como um amplificador digital de sinais elétricos.
- De forma semelhante, os dispositivos que operam como escravos estão conectados em um *chip* denominado transceptor de barramento.

Barramentos

- Os barramentos podem ser síncronos ou assíncronos.
- Os síncronos possuem uma linha controlada por um oscilador de cristal de quartzo, responsável pelo sincronismo do *clock*.
- Os assíncronos não são sincronizados por *clock* e são fáceis para operarem devido aos seus intervalos discretos.

Interatividade

Um barramento é um conjunto de fios de cobre que tem como função realizar um “caminho” elétrico entre os vários dispositivos internos ou externos do computador. Tratando-se do funcionamento da transferência de dados, os barramentos possuem quais classificações?

- a) Internos e externos.
- b) Mestres e escravos.
- c) Laterais e opostos.
- d) Individuais e coletivos.
- e) Regulares e irregulares.

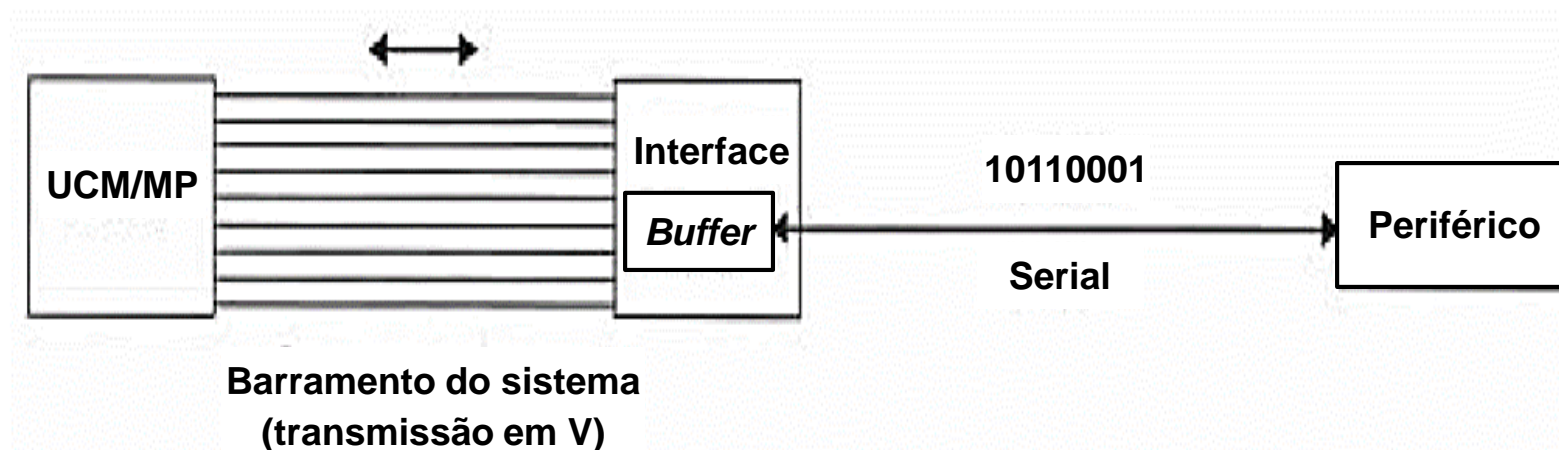
Resposta

Um barramento é um conjunto de fios de cobre que tem como função realizar um “caminho” elétrico entre os vários dispositivos internos ou externos do computador. Tratando-se do funcionamento da transferência de dados, os barramentos possuem quais classificações?

- a) Internos e externos.
- b) Mestres e escravos.**
- c) Laterais e opostos.
- d) Individuais e coletivos.
- e) Regulares e irregulares.

Tipos de transmissão

- Existem dois modos básicos para realizar uma transmissão/recepção entre os periféricos ou interfaces dos computadores ou outros dispositivos eletrônicos denominados: serial e paralelo.
- Na transmissão serial, o periférico é conectado ao módulo de controle ou interface de E/S por uma linha única para realização da transmissão de dados.
- Assim, a transferência será realizada sempre *bit a bit*, sequencialmente.



Tipos de transmissão

- Como a transmissão serial é realizada *bit a bit*, sempre será necessário que o receptor (RX) e o transmissor (TX) estejam sincronizados, ou seja, o transmissor deverá transmitir os *bits* sempre na mesma velocidade, além de possuírem o mesmo tempo de duração.
- Atualmente, a aplicação do cabeamento em série é bastante empregada em conexões de discos rígidos ou *drives* de SSD, utilizando o cabeamento SATA (*Serial Advanced Technology Attachment*).



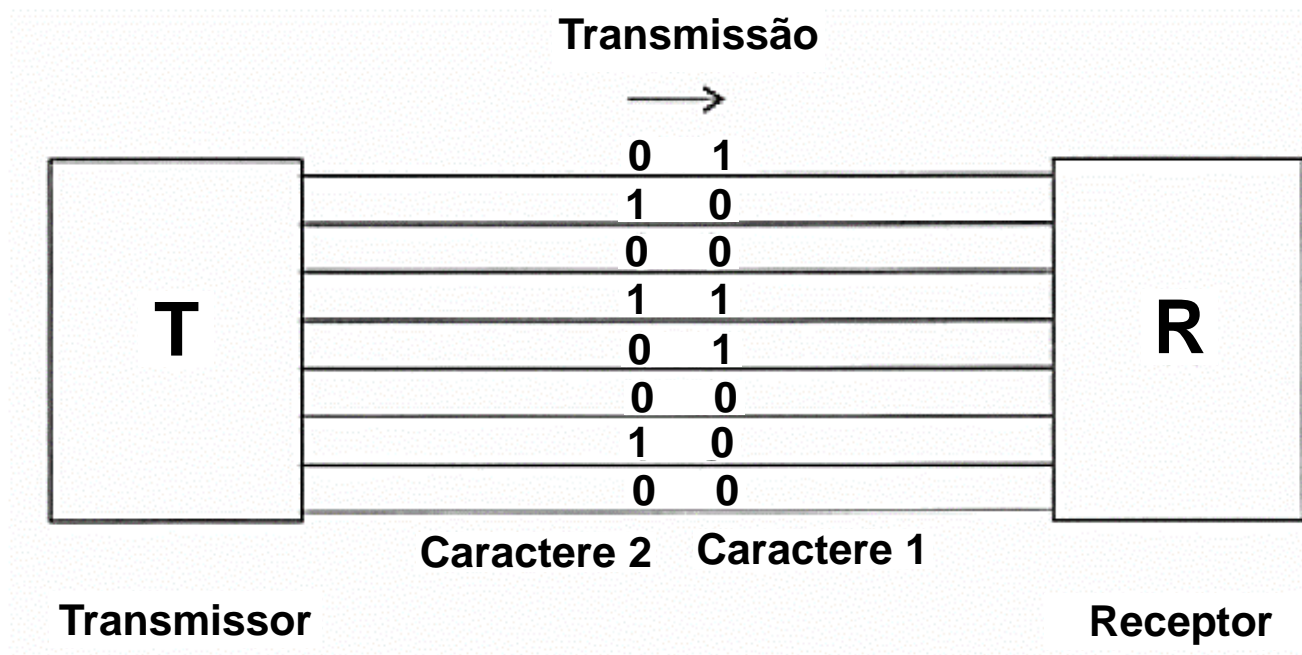
Tipos de transmissão

- Outra aplicação muito importante das transmissões em série foi o desenvolvimento da conexão do tipo USB (*Universal Serial Bus* – barramento serial universal).
- Lançada em 1998 por sete empresas de tecnologia (Compaq, DEC, IBM, Intel, Microsoft, NEC e Northern Telecom) que se juntaram para buscar uma solução de unificação na comunicação de uma gama variada de dispositivos de E/S como impressoras, *tablets*, *smartphones*, *smart TVs* e sistemas embarcados em geral.



Tipos de transmissão

- A transmissão em paralelo possui a capacidade de enviar um grupo de *bits* de cada vez, em que cada *bit* segue uma linha separada de transmissão.
- Outra característica das transmissões em paralelo é que elas costumam ser mais rápidas se comparadas com algumas conexões em série mais antigas, o que permite maiores taxas de transmissões de dados.

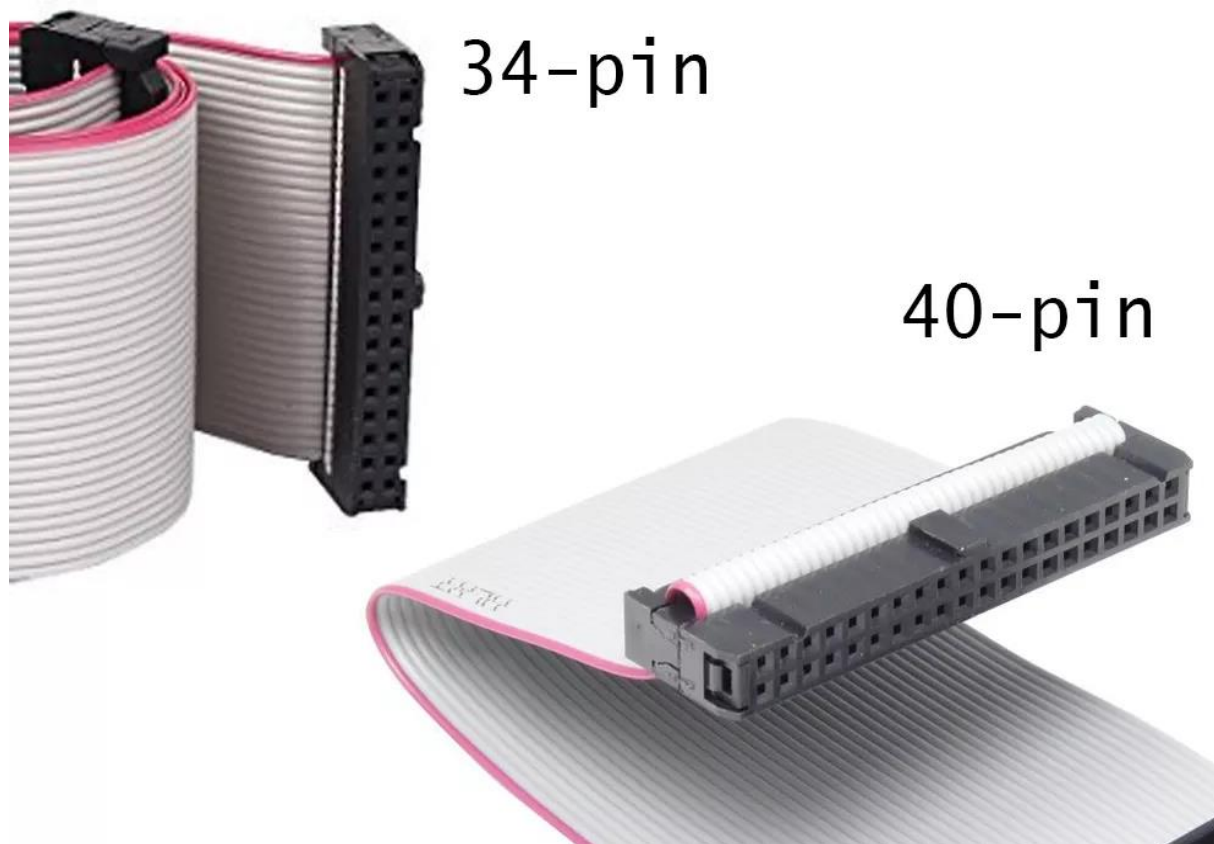


Tipos de transmissão

- Nesse tipo de transmissão, quanto maior a distância entre os dispositivos, maior será o custo, devido à quantidade de linhas utilizadas.
- Um problema aparente com a transmissão em paralelo é que os *bits* enviados podem não chegar ao destino exatamente no mesmo instante ao qual foram programadas para chegarem.
- Isso ocorre devido a algumas diferenças de comprimento de cabos de que alguns canais de comunicação são constituídos.

Tipos de transmissão

- Dentre os dispositivos que utilizavam em larga escala as conexões em paralelo estão as impressoras e as conexões mais antigas entre o disco rígido e a placa-mãe, conhecidos como IDE (*Integrated Drive Electronics*).



Dispositivos de E/S

- Um dos principais dispositivos periféricos do computador é o teclado, pois facilita que o computador possa receber comandos do ser humano.
- O reconhecimento de padrões realizados pelo teclado se baseia na interpretação dos sinais elétricos que cada tecla recebe ao ser pressionada pelo usuário.
- O teclado é um periférico de baixo custo e possui um tempo de vida relativamente longo, com cerca de 20 milhões de pressionamentos em média.



Dispositivos de E/S

- O *mouse* é um periférico utilizado para entrada de dados, cujo propósito principal é o de facilitar o trabalho do usuário final em sua tarefa de comunicação com o computador.
- O *mouse* trabalha basicamente como uma interligação visual do usuário com o sistema de *hardware* e *software* do computador.
- Ele possui alguns tipos de sensores (de acordo com o modelo) como mecânicos ou óticos para realizar a captação do movimento em uma superfície plana e transmitir as informações coletadas desses movimentos ou mesmo do acionamento dos seus botões.



Fonte: computerhope.com

Dispositivos de E/S

As impressoras são dispositivos básicos de saída, em que algumas das informações internas do computador são convertidas em símbolos impressos em um meio externo como o papel. As impressoras possuem algumas características básicas como:

- Volume de impressão que a impressora suporta por unidade de tempo: geralmente, as impressoras possuem “vazão” de impressão e caracteres por segundo (cps), dados em linha por minuto (lpm) ou em páginas por minuto (ppm), de acordo com o tipo e o modelo utilizados.
- Tecnologia para impressão de símbolos: dividida em matricial, jato de tinta, *laser* e térmica.

Dispositivos de E/S

- Impressoras do tipo matricial, jato de tinta, *laser* e térmica.



Fonte: allwhitebackground.com



Dispositivos de E/S

- É difícil de acreditar, mas os primeiros computadores não possuíam monitores de vídeo, mas lâmpadas que acendiam e apagavam na frente do painel da máquina, em um padrão que representava valores em binário.



Fonte: allwhitebackground.com

- Os monitores de vídeo possuem diversas tecnologias e características físicas para exibição de uma imagem ou vídeo, como:
 - Tubo de raios catódicos ou CRT (*cathode-ray tube*).
 - Monitores de cristal líquido ou LCD (*liquid-crystal display*).
 - Diodos emissores de luz ou TV de LED (*light emitting diodes*).
 - Monitores com painel estreito ou VPE (*flat panel display*).

Interatividade

Basicamente, existem dois modos para realizar a transmissão ou a recepção de dados/instruções entre os diferentes periféricos ou interfaces de um computador. Esses modos de transmissão são:

- a) Interno e externo.
- b) Individual e coletivo.
- c) Serial e paralelo.
- d) Regular e irregular.
- e) Côncavo e convexo.

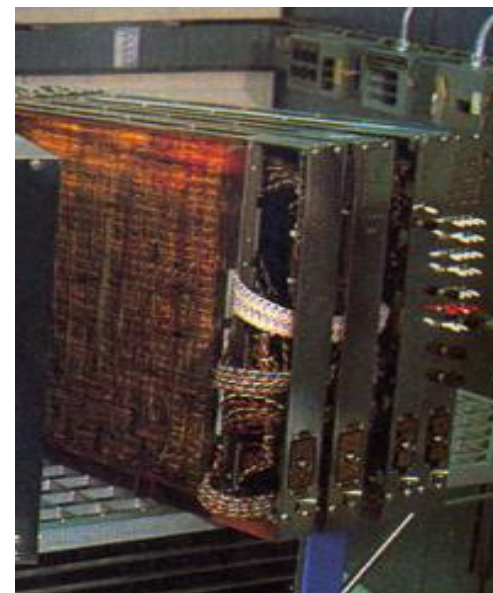
Resposta

Basicamente, existem dois modos para realizar a transmissão ou a recepção de dados/instruções entre os diferentes periféricos ou interfaces de um computador. Esses modos de transmissão são:

- a) Interno e externo.
- b) Individual e coletivo.
- c) **Serial e paralelo.**
- d) Regular e irregular.
- e) Côncavo e convexo.

Arquitetura RISC

- As arquiteturas de computadores do tipo RISC (*Reduced Instruction Set Computer* ou computador com um conjunto reduzido de instruções) foram um grande avanço no desenvolvimento das arquiteturas dos processadores modernos.
- Desenvolvido pela IBM, em 1970, o projeto recebeu o nome de IBM 801 devido ao nome do edifício onde o processador foi construído.
- Foi utilizado inicialmente no gerenciamento de uma central telefônica de grande porte, capaz de manipular, simultaneamente, 300 chamadas por segundo.



Arquitetura RISC

Essa arquitetura trouxe algumas novas questões em seu projeto como:

Menor quantidade de instruções e todas com largura fixa – possui um conjunto de instruções menores se comparado com os das máquinas CISC (*Complex Instruction Set Computer* ou computador com conjunto complexo de instruções) com a mesma configuração de *hardware*.

Arquitetura RISC

- Execução otimizada de chamada de funções – algumas pesquisas mostraram que os programas utilizam muito as chamadas de funções, consumindo um tempo razoável do processador, mas utilizando usualmente poucos dados no processo e consumindo muito processamento na transferência.
- Na arquitetura RISC, essa transferência ocorre basicamente no processador, de forma que, para que não se perca o desempenho, utiliza-se uma quantidade maior de registradores do que a arquitetura CISC.

Arquitetura RISC

- Menor quantidade de modos de endereçamento – outro fator preponderante na arquitetura RISC está relacionado aos modos de endereçamento de memória pelo conjunto de instruções.
- Arquiteturas RISC buscam soluções simples como LOAD/STORE.
- LOAD – transfere o dado oriundo da memória para o registrador específico do processador.
STORE – operação contrária ao LOAD, que irá transferir um dado do registrador específico para algum endereço de memória.
 - Já as máquinas com arquitetura CISC tendem a uma quantidade maior de endereçamentos.

Arquitetura RISC

- Execução de cada instrução em um ciclo de *clock* – computadores do tipo CISC levam um tempo maior para a execução de instruções devido ao uso de um microcódigo, que implicará na interpretação de cada uma das micro-operações envolvidas no processo, o que implicará em um atraso total na execução das instruções.
- Em arquiteturas RISC, por outro lado, ao utilizarem poucas instruções e todas elas mais simples, não haverá a necessidade do uso de microcódigos.
 - Modos de execução baseados no uso de *pipeline* – para que se extraia o melhor desempenho do *pipeline* é preciso que as instruções possuam formato e complexidade similares, ocasionando um tempo redundante em todas as etapas, equilibrando o tempo de execução em cada uma delas.

Pipeline no RISC

- Para máquinas do tipo RISC, a maioria das instruções é do tipo registrador-para-registrador, envolvendo apenas dois ou três estágios de *pipeline*:
- I: busca a instrução;
- E: executa a instrução (calcula um endereço da memória principal);
- D: operação de armazenamento entre o registrador e a memória ou vice-versa.

I	E	D					
	I	E	D				
		I	E				
			I	E			
				I	E	D	
					I	E	
						I	E

Fonte: adaptado
de: STALLINGS
(2018, p. 474).

Pipeline no RISC

- Dois problemas impedem que seja obtido o aumento máximo na velocidade do *pipeline*, sendo um deles em relação ao acesso à memória principal.
- E o segundo problema envolvendo algum desvio na instrução, de modo que o fluxo sequencial de execução pode ser interrompido.
- Assim, como solução utiliza-se a menor quantidade de estágios possível, inserindo-se uma instrução a mais no ciclo de estágios, o que, claramente, resulta em um ciclo de *pipeline* mais longo e demorado.

I	E	D							
	I		E	D					
			I		E				
					I	E	D		
						I		E	
								I	E

Fonte: adaptado de:
STALLINGS (2018, p. 474).

Interatividade

Arquitetura de computador do tipo RISC (*Reduced Instruction Set Computer*) foi um grande avanço no desenvolvimento dos processadores modernos. Essa arquitetura trouxe questões como possuírem um conjunto de instruções menor, execução otimizada de chamada de funções, modos de execução baseado no uso de *pipeline* e execução de cada instrução por ciclo. Além dessas características, qual outra característica relevante possui a arquitetura RISC?

- a) Menor quantidade de memória RAM.
- b) Menor quantidade de memória ROM.
- c) Menor quantidade de espaço em disco rígido.
- d) Menor quantidade de modos de endereçamento.
- e) Menor quantidade de acessos ao barramento.

Resposta

Arquitetura de computador do tipo RISC (*Reduced Instruction Set Computer*) foi um grande avanço no desenvolvimento dos processadores modernos. Essa arquitetura trouxe questões como possuírem um conjunto de instruções menor, execução otimizada de chamada de funções, modos de execução baseado no uso de *pipeline* e execução de cada instrução por ciclo. Além dessas características, qual outra característica relevante possui a arquitetura RISC?

- a) Menor quantidade de memória RAM.
- b) Menor quantidade de memória ROM.
- c) Menor quantidade de espaço em disco rígido.
- d) Menor quantidade de modos de endereçamento.
- e) Menor quantidade de acessos ao barramento.

Processadores Superescalares

- O projeto de processadores Superescalares foi implementado em 1987 logo após o desenvolvimento da arquitetura RISC.
- Um processador Superescalar é constituído por múltiplos e independentes *pipelines* de instruções.
- Ele foi desenvolvido para melhorar o desempenho de processadores que possuem instruções do tipo aritméticas de inteiros, de ponto flutuante, desvios condicionais, realizando tais tarefas simultânea e independentemente.

Processadores Superescalares

- Uma das grandes vantagens da implementação de *pipelines* múltiplos é justamente o aumento no nível de paralelismo de instruções, o que possibilita múltiplos fluxos que podem ser processados simultaneamente.
- Ao invés de buscar somente uma instrução por vez, como nos sistemas convencionais, um processador Superescalar possui como tarefa buscar múltiplas instruções, realizando na sequência uma tentativa de localização das próximas instruções para que, independentemente, possa executá-las em paralelo.

Processadores Superescalares

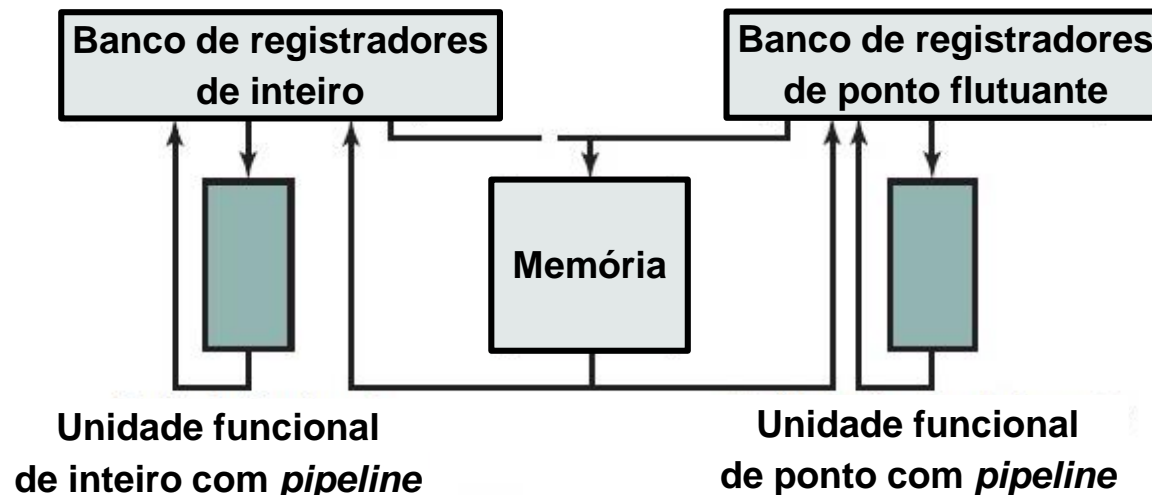
- Se houver dependências, todas devem ser localizadas e o processador poderá executar, e até mesmo completá-las, em uma ordem diferente do código de máquina originalmente programado.
- Uma forma de contornar tal problema seria o processador eliminar algum tipo de dependência de dados ou instruções desnecessárias, utilizando-se para isso registradores adicionais, renomeando assim as referências obtidas dos registradores no código original.

Processadores Superescalares

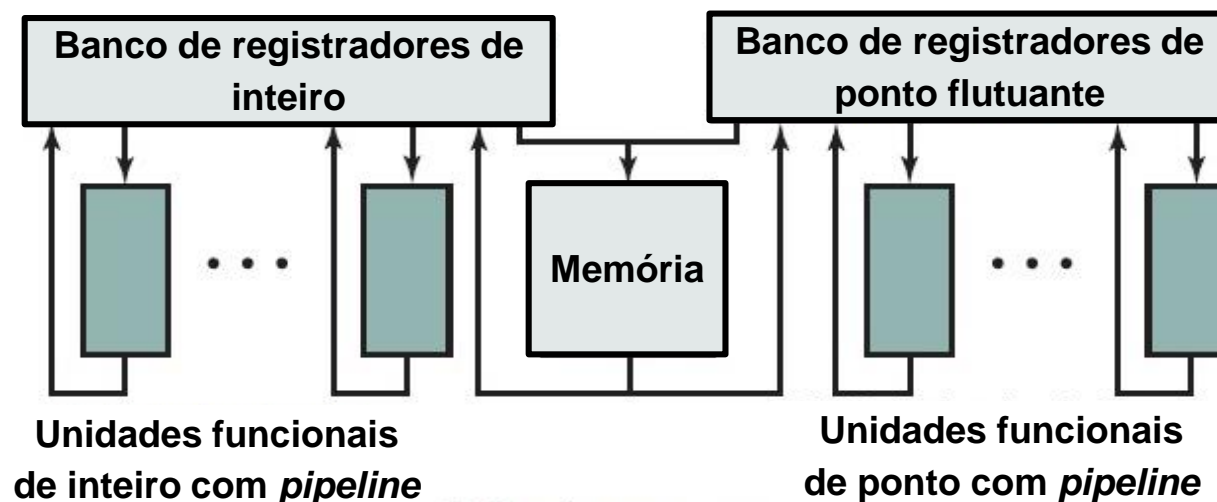
- Em uma organização escalar convencional existe uma unidade funcional em um *pipeline* único para realizar operações inteiras e outro *pipeline* para realizar operações de ponto flutuante.
- Nesse tipo de organização, o paralelismo será obtido quando se habilitam múltiplas instruções em diferentes estágios do *pipeline* de uma só vez.
- Já em uma organização Superescalar, existem várias unidades funcionais, em que cada uma será implementada com um único *pipeline*.
 - Cada uma dessas unidades implementará um grau de paralelismo em função da estrutura com *pipeline*.

Processadores Superescalares

- Comparativo entre organizações.
- Escalar e Superescalar.



(a) Organização escalar



(b) Organização superescalar

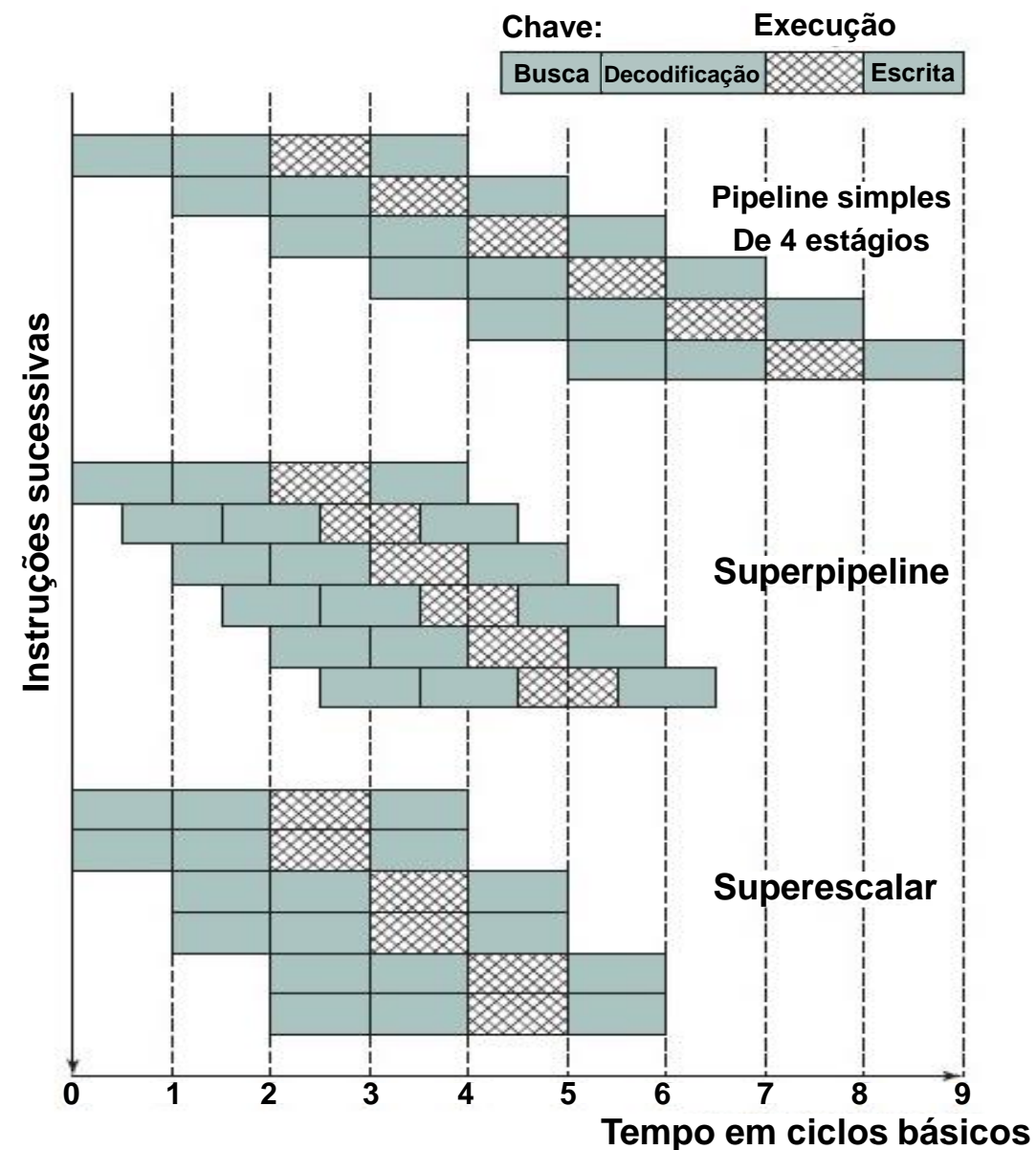
Processamento Superpipeline

- A evolução do *pipeline* simples ficou conhecida como Superpipeline e foi desenvolvida em 1988.
- O Superpipeline trabalha com o fato de que possam se realizar vários estágios de *pipeline*, executando tarefas que irão requerer metade do tempo de um ciclo de *clock*.
- Dessa forma, a velocidade de *clock* possibilitará que duas tarefas sejam realizadas em apenas um único ciclo de *clock*.

Superpipeline x Superescalar

- *Pipeline* simples, Superpipeline e Superescalar.

Fonte: STALLINGS (2018, p. 494).



Limitações no processamento paralelo

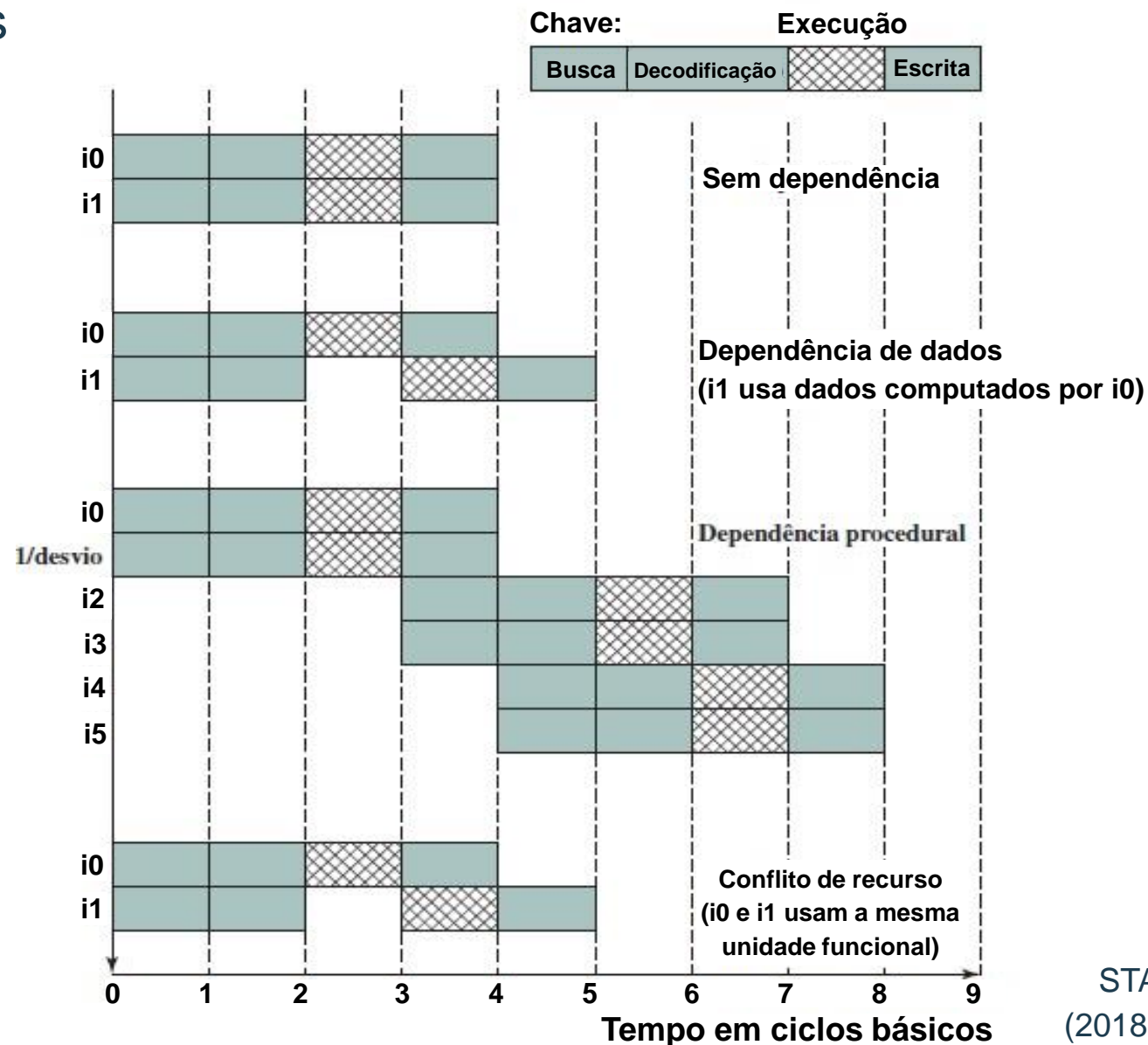
- O paralelismo em nível de instruções também pode conter algumas limitações ao executar suas tarefas.
- Assim, é necessária uma combinação ótima entre as técnicas de implementação de paralelismo em *hardware* em conjunto com o compilador (software), a fim de maximizar o paralelismo das instruções.
- Algumas limitações no paralelismo em nível de instrução podem ser listadas:
 - Dependência de dados verdadeira: quando uma segunda instrução só pode ser executada ao término de execução da primeira instrução, devido ao compartilhamento dos dados.

Limitações no processamento paralelo

- Dependência procedural: quando há algum desvio na sequência de execução das instruções, complicando a operação do *pipeline*.
- Conflitos de recursos: quando há uma competição de duas ou mais instruções pelos mesmos recursos (memória, processador etc.) simultaneamente.
- Dependência de saída: quando duas ou mais instruções necessitam armazenar seu resultado em um mesmo endereço e ao mesmo tempo.
 - Antidependência: quando uma instrução, ao utilizar um local ou operando, entra em conflito com a instrução seguinte que também tenta utilizar o mesmo local ou operando.

Limitações no processamento paralelo

- Comparativo entre os diferentes tipos de dependências em processamento paralelo.



Interatividade

O Superpipeline, desenvolvido em 1988, opera dividindo os ciclos de *clock* em partes menores, não ocorrendo sobreposição entre estágios. Assim, uma das grandes vantagens do Superpipeline consiste em:

- a) Executar duas tarefas em apenas um único ciclo.
- b) Executar tarefas em série ao invés de paralelo.
- c) Executar tarefas diretamente na memória cache.
- d) Executar operações de *pipeline* de números inteiros.
- e) Executar operações de *pipeline* de ponto flutuante.

Resposta

O Superpipeline, desenvolvido em 1988, opera dividindo os ciclos de *clock* em partes menores, não ocorrendo sobreposição entre estágios. Assim, uma das grandes vantagens do Superpipeline consiste em:

- a) Executar duas tarefas em apenas um único ciclo.
- b) Executar tarefas em série ao invés de paralelo.
- c) Executar tarefas diretamente na memória cache.
- d) Executar operações de *pipeline* de números inteiros.
- e) Executar operações de *pipeline* de ponto flutuante.

ATÉ A PRÓXIMA!