



PROCESSADOR

SUMÁRIO

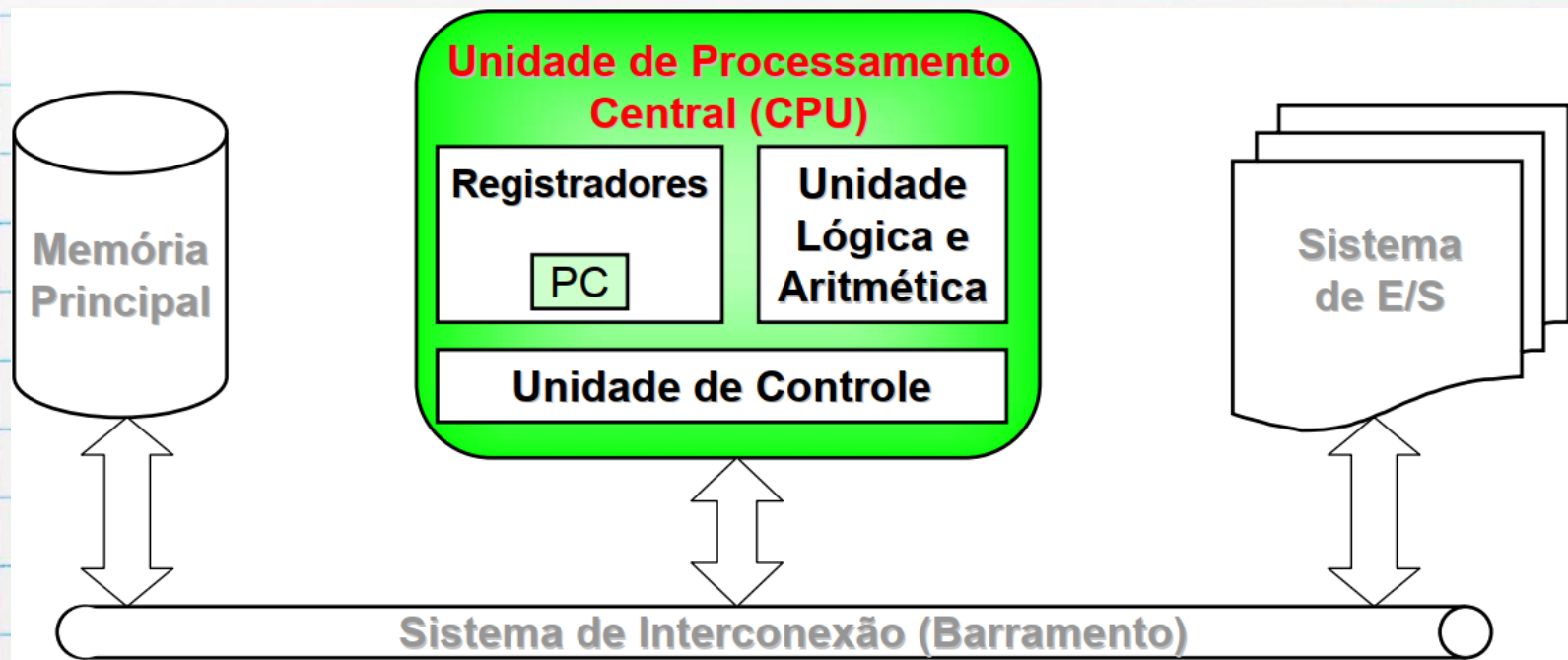
2

- *Introdução*
- *Função*
- *Elementos*

- *Introdução*

INTRODUÇÃO

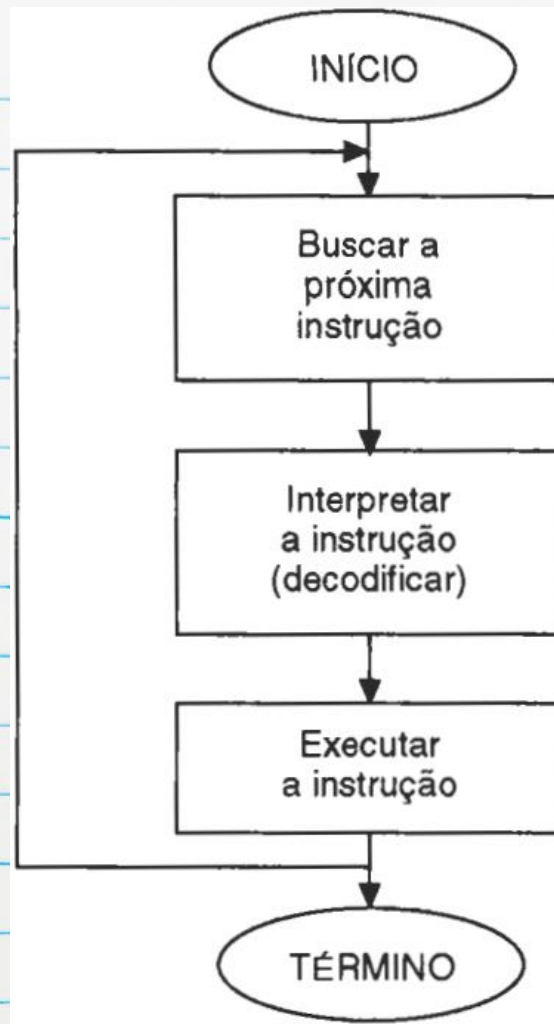
4



INTRODUÇÃO

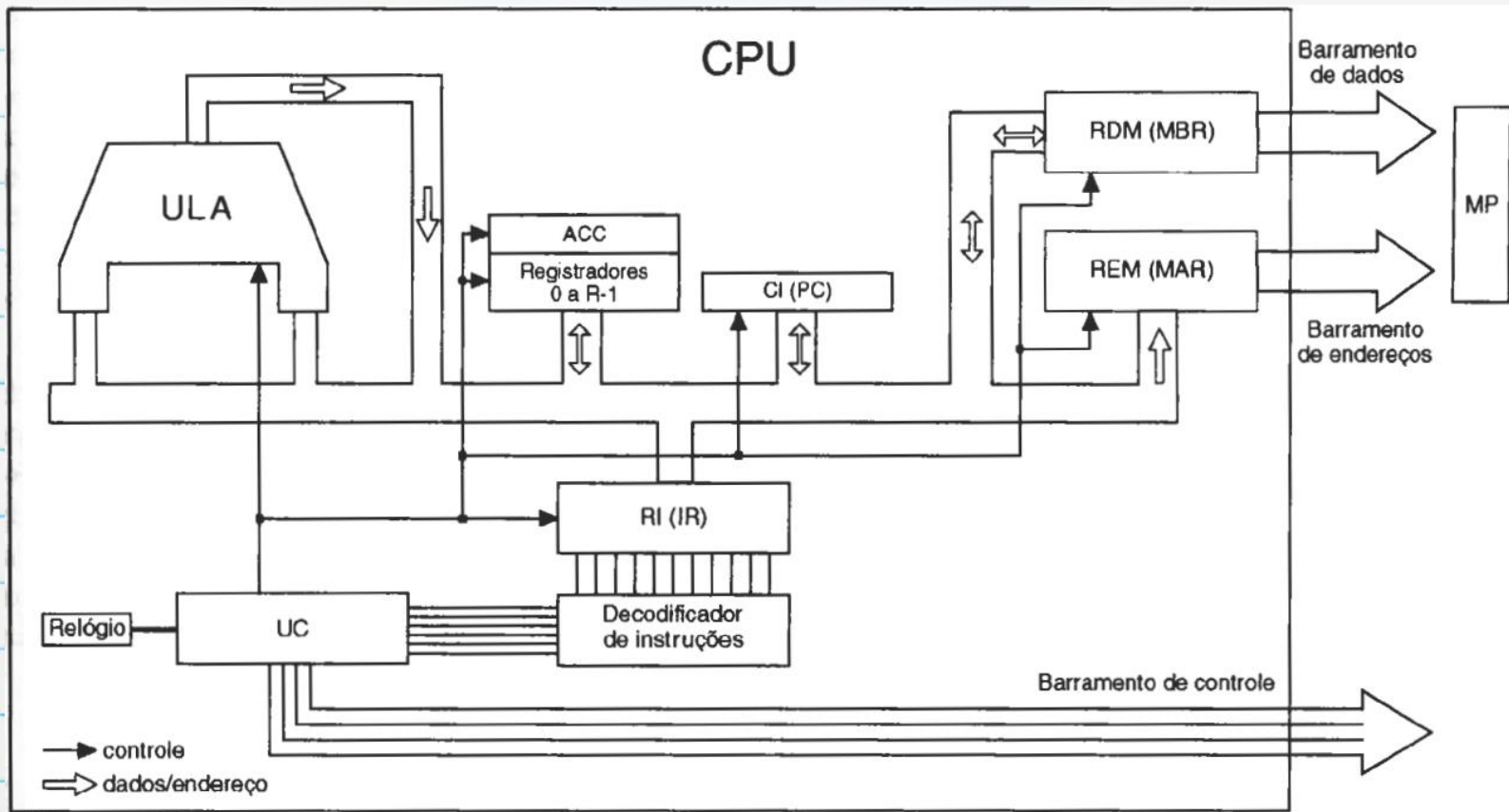
- “Cérebro” do computador.
- Todo processador tem um conjunto de operações primitivas.
 - ❖ Representadas por uma sequencia de sinais elétricos (0's e 1's).
 - ❖ Instrução de máquina
 - Sequencia de 0's e 1's que formaliza uma determinada operação a ser realizada pelo processador.
 - ❖ Programa executável
 - Conjunto de instruções de máquinas sequencialmente ordenadas.
 - Pré-requisitos para executar
 - Armazenar em células sucessivas da MP as instruções a serem executadas;
 - Armazenar na CPU o endereço da primeira instrução a ser executada.

INTRODUÇÃO



*Fluxo Básico (resumido)
de um ciclo de instrução*

INTRODUÇÃO



Esquema simplificado de uma CPU

- Função

FUNÇÃO

9

- *Requisitos exigidos:*

- ❖ *Buscar instrução: o processador lê uma instrução da memória (registrador, cache, memória principal).*
- ❖ *Interpretar a instrução: a instrução é decodificada para determinar qual ação é requerida.*
- ❖ *Obter os dados: a execução de uma instrução pode requerer leitura de dados da memória ou um módulo de E/S.*
- ❖ *Processar os dados: a execução de uma instrução pode requerer efetuar alguma operação aritmética ou lógica com os dados.*
- ❖ *Gravar os dados: os resultados de uma execução podem requerer gravar dados para memória ou um módulo E/S.*

FUNÇÃO

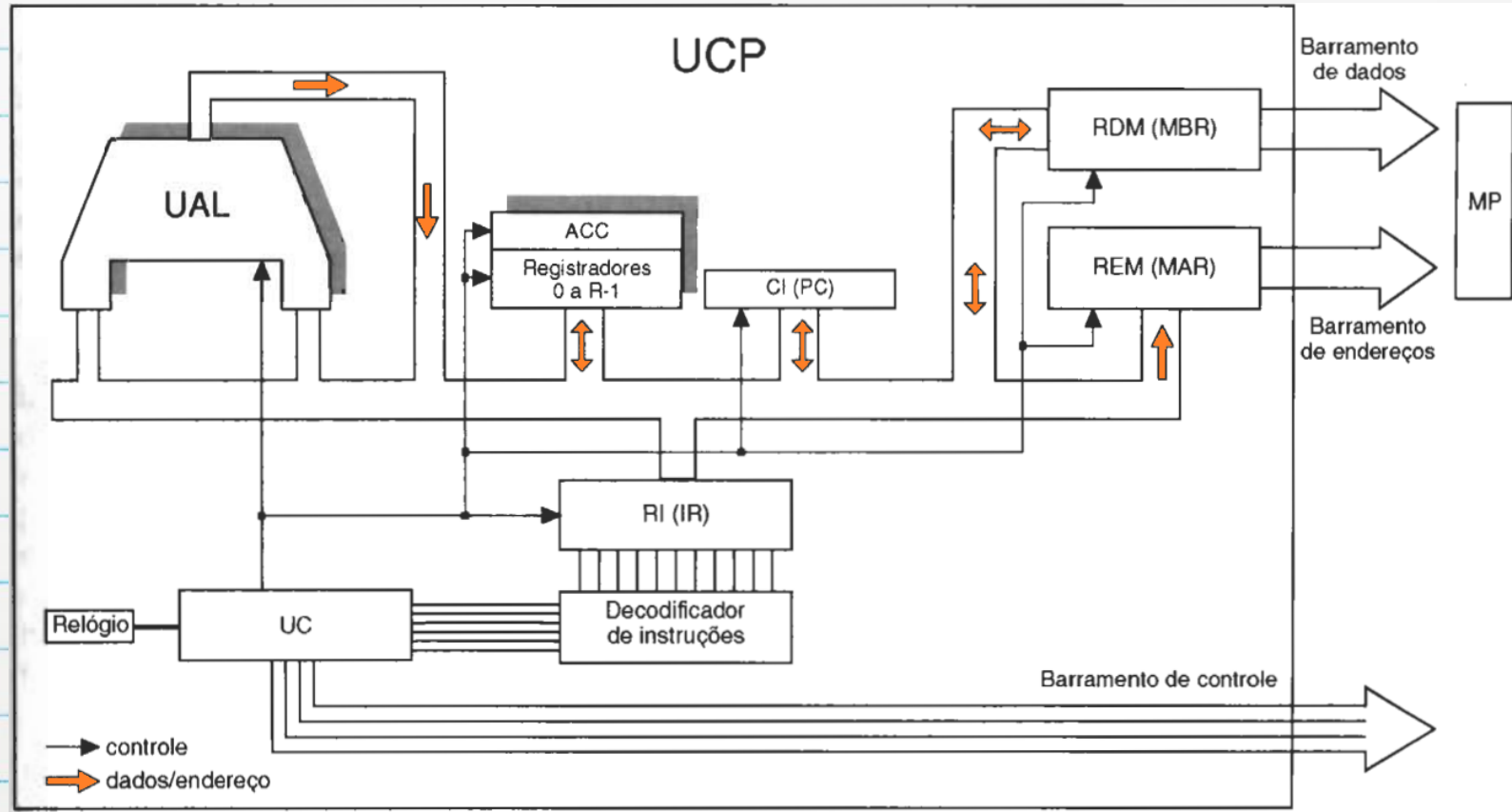
Processamento

10

- Realiza as atividades relacionadas com a efetiva execução de uma operação;
- Tarefas comuns
 - ❖ Operações aritméticas (somar, subtrair, multiplicar, dividir);
 - ❖ Operações lógicas (and, or, xor, etc);
 - ❖ Movimentação de dados (memória - UCP, registrador-registrador, etc);
 - ❖ Desvios (alteração de sequência de execução de instruções);
 - ❖ Operações de entrada e saída.
- Componentes
 - ❖ ULA - Dispositivo principal desta área de atividades;
 - ❖ Registradores - armazenar dados (ou guardar resultados);
 - ❖ Barramento interno - interligação da ULA aos registradores.

FUNÇÃO

Processamento



Esquema simplificado de uma CPU com realce para os elementos que contribuem para a realização da função de processamento

FUNÇÃO

Controle

12

- Projetada para entender o que fazer, como fazer e comandar quem vai fazer no momento adequado.
 - ❖ Controle das ação dos demais componentes do sistema de computação
 - Atividades de busca, interpretação e controle da execução das instruções
- O conhecimento das funções exercidas pela área de controle responde:
 - ❖ O que é e como funciona uma instrução de máquina?
 - ❖ Como a referida instrução de máquina é movimentada da memória para a CPU?
 - ❖ Onde a instrução de máquina será armazenada na CPU?
 - ❖ Como será identificada e controlada a ação (a operação) que deve ser realizada?

FUNÇÃO

Controle

13

- Parte funcional que realiza as atividades de:
 - ❖ Busca da instrução que será executada
 - Armazenando-a em um registrador projetado para esta finalidade;
 - ❖ Interpretação das ações a serem desencadeadas com a execução da instrução (*fetch cycle* - ciclo de busca da instrução);
 - ❖ Geração dos sinais de controle apropriados para ativação das atividades requeridas para a execução propriamente dita da instrução identificada (*execute cycle* - ciclo de execução da instrução)
 - Eles são enviados aos diversos componentes do sistema, sejam internos da CPU (como a ULA) ou externos (como a memória ou E/S)

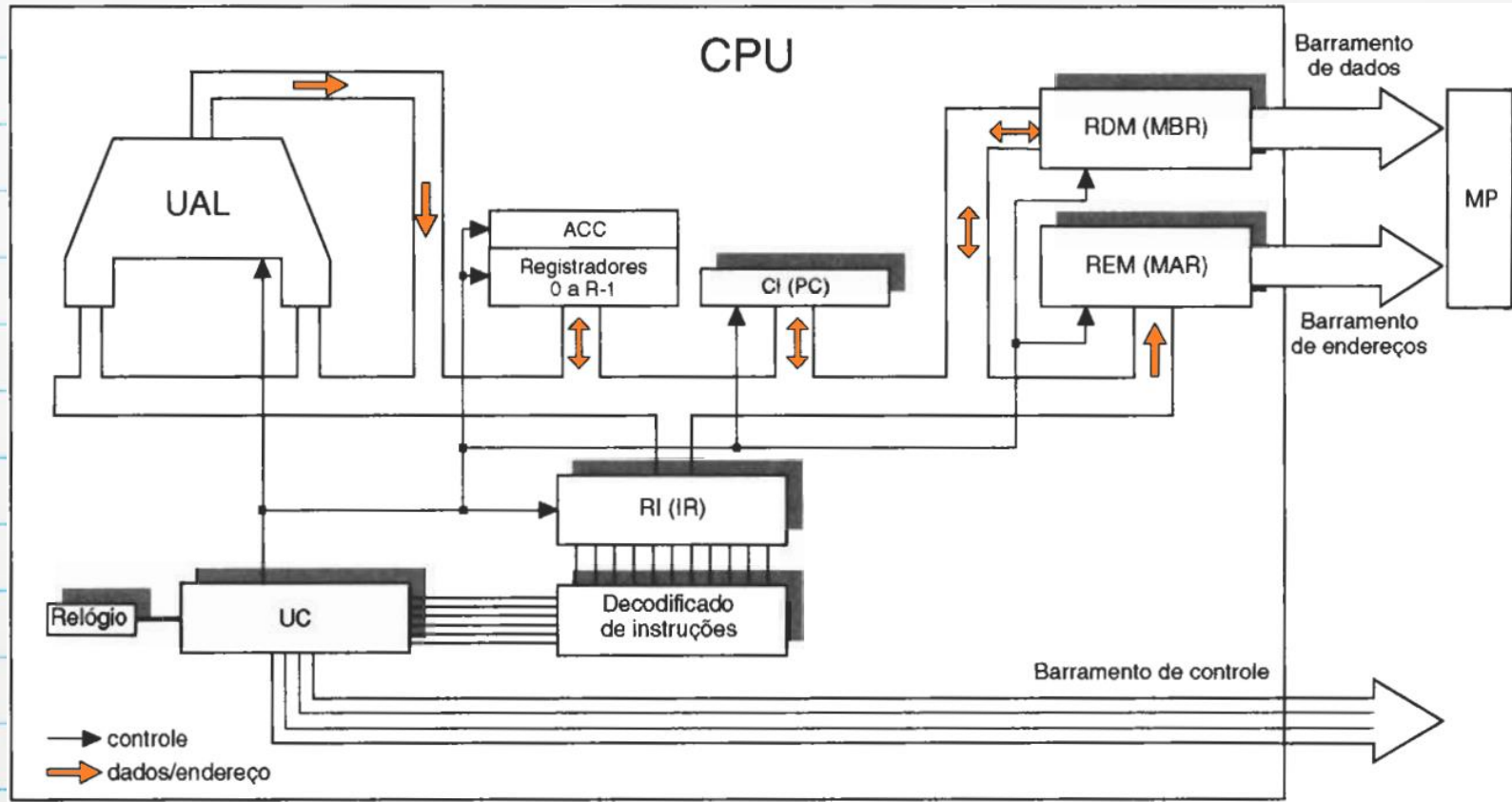
FUNÇÃO

Controle

14

- Dispositivos básicos que devem fazer parte da área funcional;
 - ❖ Unidade de controle;
 - ❖ Decodificador;
 - ❖ Registrador de instrução (RI ou IR - instruction register);
 - ❖ Contador de instrução (CI ou PC - program counter);
 - ❖ Relógio ou clock;
 - ❖ Registradores de endereço de memória (REM); e
 - ❖ Registradores de dados da memória (RDM)
- A quantidade, a complexidade e a disposição dos componentes que realizam as funções de controle variam consideravelmente de CPU para CPU.

FUNÇÃO Controle



Esquema simplificado de uma CPU com realce para os elementos que contribuem para a realização da função de controle

- Elementos

ULA

ELEMENTOS

ULA

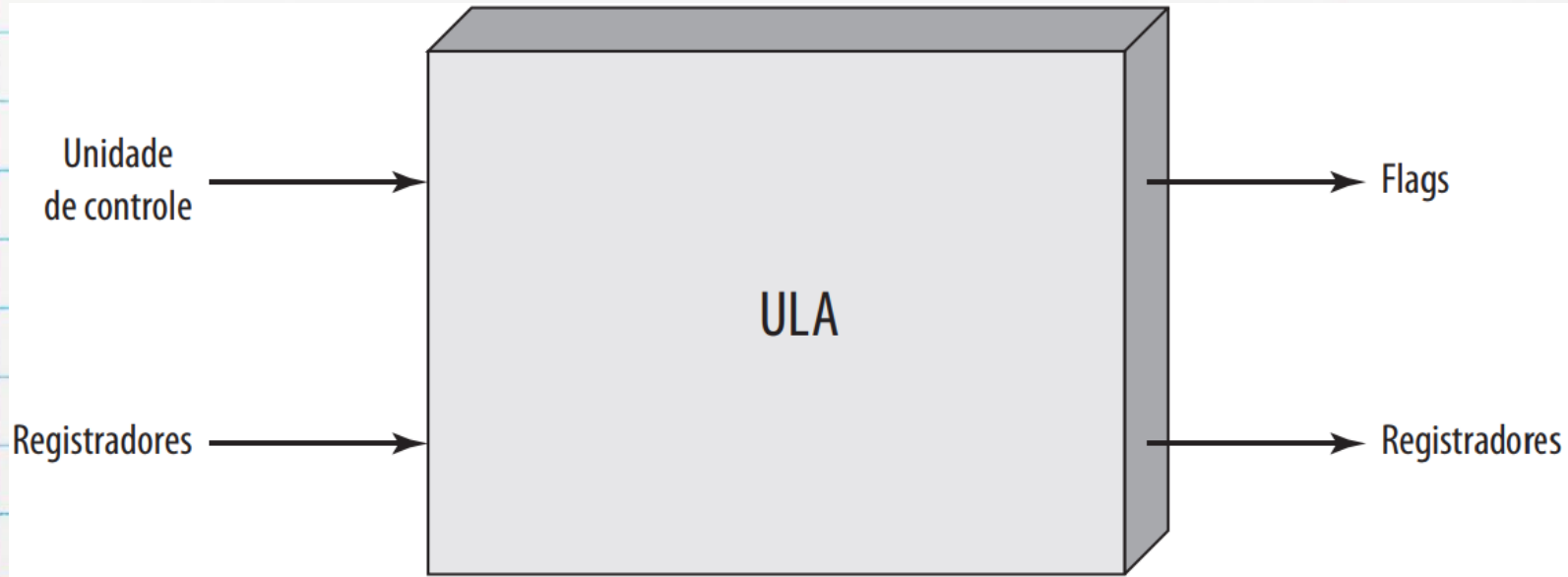
17

- “Núcleo” da CPU.
 - ❖ Contribui para a realização da função de processamento
 - ❖ Todos os outros elementos do sistema de computação (unidade de controle, registradores, memória, E/S) existem principalmente para trazer dados para a ALU processar, e depois levar os resultados de volta.
- Aglomerado de circuitos lógicos e componentes eletrônicos simples
- Pode ser
 - ❖ Uma pequena parte da pastilha do processador; ou
 - ❖ Um conjunto de componentes lógicos de alta velocidade (mais de uma ULA nos processadores modernos);

ELEMENTOS

ULA

18



Entradas e saídas da ULA

ELEMENTOS

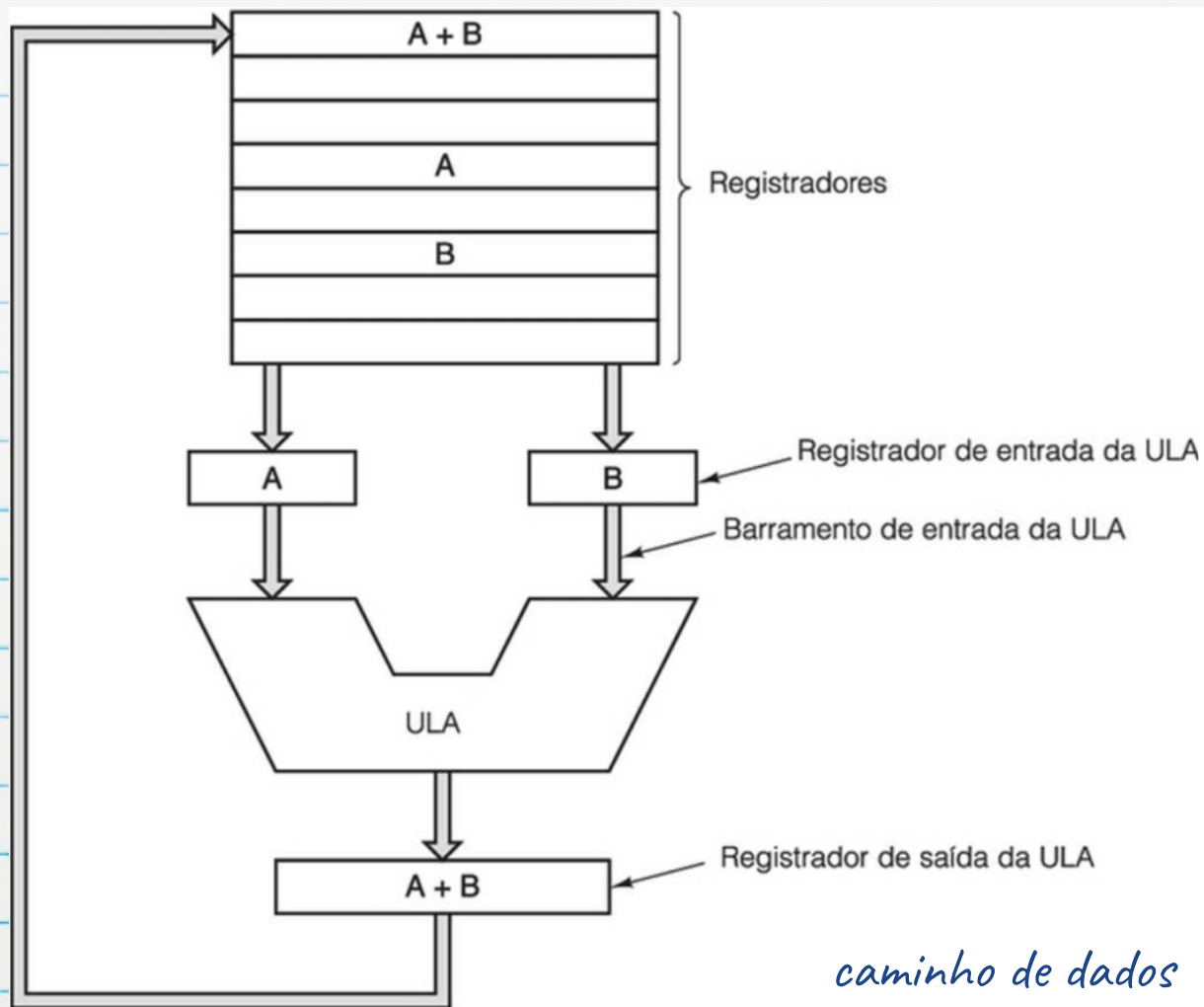
ULA

19

- Os dados são apresentados à ULA em registradores.
- Os resultados de uma operação são armazenados nos registradores.
 - ❖ Esses registradores são locais de armazenamento temporários dentro do processador, que são conectados por meio de sinais à ULA.
- Ativa bits especiais (flags) como resultado de uma operação
 - ❖ Armazenados nos registradores dentro do processador;
 - ❖ Ex: operação nula, operação negativa, overflow, etc.
- A unidade de controle oferece sinais que controlam a operação da ULA e o movimento dos dados para dentro e fora da ULA.

ELEMENTOS

Processamento - ULA



ELEMENTOS

ULA

21

- Executa as operações de processamento de dados.
 - ❖ Podem ser diferentes para cálculos com inteiros e ponto flutuante.
- Algumas operações matemáticas
 - ❖ Soma
 - ❖ Multiplicação
 - ❖ Operação lógica AND
 - ❖ Operação lógica XOR
 - ❖ Deslocamento a direita
 - ❖ Incremento
 - ❖ Subtração
 - ❖ Divisão
 - ❖ Operação lógica OR
 - ❖ Operação complemento
 - ❖ Deslocamento a esquerda
 - ❖ Decremento

ELEMENTOS

ULA

22

- Nos processadores modernos, o barramento interno conduz os bits de dados de e para a memória cache
- Está conectada a um grupo de registradores pelo barramento interno, formando o caminho de dados.
 - ❖ Nem todos os projetos têm os registradores A, B e de saída.
 - ❖ Pode-se utilizar um conjunto de ULAs para a execução paralela de instruções

- *Elementos*

Registradores

ELEMENTOS

Registradores

24

- Pequenas unidades de memória com alta velocidade.
 - ❖ Mais rápido que as memórias principal e cache.
 - ❖ Utilizam o barramento interno da CPU.
 - ❖ Possuem largura igual ao tamanho estabelecido pelo fabricante pela palavra do referido processador.
- Armazenamento temporário de dados, instruções e endereços em utilização pelo processador.
 - ❖ Memória interna do processado
 - ❖ Possibilitam operações de tanto de leitura quanto de escrita.
 - Alguns permitem acesso indireto no nível ISA

ELEMENTOS

Registradores

25

- Possuem diferentes funções, mas têm um uso bem definido dentro da arquitetura.
 - ❖ Contribui para a realização tanto da função de processamento quanto da função de controle.
- Os registradores no processador desempenham dois papéis:
 - ❖ Registradores visíveis ao usuário: possibilitam que o programador de linguagem de máquina ou assembly minimize as referências à memória, pela otimização do uso de registradores.
 - ❖ Registradores de controle e estado: usados pela UC para controlar a operação do processador e por programas privilegiados do SO para controlar a execução de programas

Registradores Visíveis ao Usuário

- Pode ser referenciado pelos recursos da linguagem de máquina que o processador executa.
- Pode ser dividido nas seguintes categorias:
 - ❖ Uso geral.
 - ❖ Dados.
 - ❖ Endereços.
 - ❖ Códigos condicionais

Obs.: a especialização limita a flexibilidade do programado

- Mais registradores requerem mais bits para especificadores de operandos;
- Menos registradores resultam em mais referências de memória.

Registradores Visíveis ao Usuário

Registradores de propósito geral

- ❖ Atribuídos para uma variedade de funções pelo programador.
- ❖ Qualquer registrador de uso geral pode conter um operando para qualquer opcode.
- ❖ No entanto, frequentemente existem restrições.
 - Ex.: registradores dedicados para ponto flutuante e operações de pilha.
- ❖ Em alguns casos, são usados para funções de endereçamento
 - Ex.: indireto por registradores, deslocamento.
- ❖ Em outros casos, existe uma separação clara ou parcial entre os registradores de dados e os de endereços

Registradores Visíveis ao Usuário

Registradores de dados

- ❖ Usados apenas para guardar dados e não podem ser empregados para calcular o endereço de um operando.

Registradores de endereços

- ❖ Uso geral ou dedicados para um modo de endereçamento em particular.
- ❖ Ex.:
 - ponteiros de segmento: em uma máquina com endereçamento segmentado, um registrador de segmento guarda o endereço base do segmento.
 - Registradores de índice: estes são usados para indexar endereços e podem ser auto indexados.
 - Ponteiros de pilha: se houver endereçamento de pilha visível ao usuário, haverá um registrador dedicado que aponta para o topo da pilha.

Registradores Visíveis ao Usuário

Registradores de códigos condicionais

- ❖ Parcialmente visível ao usuário
- ❖ Guarda códigos condicionais (ou flags).
 - Bits definidos pelo hardware do processador como resultado das operações.
 - Ex.: uma operação aritmética pode produzir um resultado positivo, negativo, zero ou fora da capacidade.
- ❖ Coletados em um ou mais registradores.
 - Normalmente eles fazem parte do registrador de controle.
- ❖ Geralmente, as instruções de máquina permitem que esses bits sejam lidos por referência implícita, mas o programador não pode alterá-los.

Registradores Visíveis ao Usuário

Vantagens	Desvantagens
1. Como os códigos condicionais são definidos por instruções normais aritméticas ou de movimentação de dados, eles devem reduzir o número de instruções de comparação e teste (COMPARE, TEST) necessárias.	1. Códigos condicionais acrescentam complexidade, tanto para hardware como para software. Os bits dos códigos condicionais são frequentemente modificados de maneiras diferentes por instruções diferentes, tornando a vida do microprogramador e do projetista de compiladores mais difícil.
2. Instruções condicionais, como BRANCH, são simplificadas em relação a instruções compostas como TEST AND BRANCH.	2. Códigos condicionais são irregulares; normalmente eles não fazem parte do caminho principal de dados e, por isso, requerem conexões extras de hardware.
3. Códigos condicionais facilitam desvios múltiplos. Por exemplo, uma instrução TEST pode ser seguida de dois desvios, um para menor ou igual a zero e outro para maior que zero.	3. Frequentemente, máquinas com códigos condicionais precisam adicionar instruções especiais que não usam códigos condicionais para situações especiais de qualquer forma, como verificação de bits, controle de laços e operações atômicas de semáforos.
	4. Em uma implementação de pipeline, códigos condicionais requerem sincronização especial para evitar conflitos.

Códigos condicionais

Registradores de controle e estado

- *Controla a operação do CPU e as trocas de informações com a MP.*
 - ❖ *Normalmente não é visível ao usuário*
 - ❖ *Alguns podem ser visíveis às instruções da máquina executadas no modo de controle ou de sistema operacional.*
- *Quatro registradores são essenciais para execução das instruções:*
 - ❖ *Registrador de instrução - RI*
 - ❖ *Contador de programa - PC*
 - *Ou contador de instrução (CI) ou ponteiro de instrução (EIP)*
 - ❖ *Registrador de endereço de memória (REM); e*
 - ❖ *Registrador de dados de memória (RDM).*
 - *Ou Registrador de buffer de memória (MBR)*

ELEMENTOS

32

Registadores de controle e estado

Registrador de instrução - RI

- ❖ Armazena a instrução a ser executada pela CPU.
- ❖ Ao iniciar um ciclo de instrução, a UC emite sinais de controle para processar um ciclo de leitura para buscar a instrução na memória.
- ❖ Ao término do ciclo de leitura, a instrução desejada está armazenada no RI via barramento de dados e RDM.

Contador de programa - PC

- ❖ Armazena o endereço da próxima instrução a ser executada.
- ❖ Após o início da busca da instrução, o sistema automaticamente efetiva a modificação do conteúdo do PC
 - Ele passa a armazenar o endereço da próxima instrução na sequência.

Registradores de controle e estado

Registrador de endereço de memória (REM) e dados de memória (RDM)

- ❖ Utilizados pela CPU e memória para comunicação e transferência de informações.
- ❖ RDM
 - Contém uma palavra de dados para ser escrita na memória ou a palavra lida mais recentemente.
 - Tamanho igual ao barramento de dados
 - Normalmente igual ou múltiplo do tamanho da palavra do processador.
- ❖ REM
 - Contém o endereço de uma posição de memória.
 - Tamanho igual ao dos endereços da memória e, conseqüentemente, do barramento de endereço do sistema.

ELEMENTOS

34

Registradores de controle e estado

- Os registradores RI, PC, REM e RDM são usados para movimentar dados entre o processador e a memória.
- Dentro do processador, os dados precisam ser apresentados à ULA para serem processados.
 - ❖ Ela pode acessar direto o MBR e os registradores visíveis ao usuário.
- Pode haver outros registradores de buffer na vizinhança do ULA;
 - ❖ Eles servem:
 - Como registradores de entrada e saída para a ALU
 - Para trocar dados com o MBR e com os registradores visíveis ao usuário.

ELEMENTOS

35

Registradores de controle e estado

- O processador atualiza o PC depois de ler cada instrução para que o PC sempre aponte para a próxima instrução a ser executada.
 - ❖ Uma instrução de desvio ou salto também modifica o conteúdo de PC.
- A instrução lida é colocada em IR, onde o opcode e os especificadores de operando são analisados.
- Os dados são trocados com a memória com o uso de MAR e MBR.
- Em um sistema organizado com barramentos, MAR se conecta diretamente ao barramento de endereços e MBR se conecta diretamente ao barramento de dados.
- Registradores visíveis ao usuário, por sua vez, trocam dados com MBR.

ELEMENTOS

36

Registradores de controle e estado

- Palavra de Estado do Programa (PSW – Program Status Word)
 - ❖ Conjunto de registradores que contém as informações de estado;
 - ❖ Auxilia e completa a realização das operações matemáticas pela ULA;
 - ❖ Indica o estado de vários elementos referentes à operação em si.
 - ❖ Se divide em bits que possuem significado diferente, um por um.

Registradores de controle e estado

- Principais bits de estado do PSW
 - ❖ Sinal: sinal resultante da última operação aritmética realizada;
 - ❖ Overflow: se setado ($=1$), última operação aritmética realizada resultou em um estouro do valor, um erro.
 - ❖ Zero: se setado ($=1$), última operação aritmética realizada resultou no valor zero.
 - ❖ Carry (vai 1): ocorreu "vai 1" para o bit mais à esquerda na última operação de soma realizada.
 - Pode indicar, também, overflow em operações com números sem sinal.
 - ❖ Paridade: setado ($=1$) ou não ($=0$), depende da quantidade de bits 1 no byte recebido.

ELEMENTOS

38

Registradores – Processador 8088/8086

- CPU possui 14 registradores de 16 bits visíveis.
- 4 registradores de uso geral:
 - ❖ **AX** (Acumulador): armazena operandos e resultados dos cálculos aritméticos e lógicos.
 - ❖ **BX** (Base): armazena endereços indiretos.
 - ❖ **CX** (Contador): conta iterações de loops ou especifica o n° de caracteres de uma string.
 - ❖ **DX** (Dados): armazena overflow e endereço de E/S.
 - ❖ Podem ser usados como registradores de 8 bits
 - Ex: AH e AL (byte alto e byte baixo de AX).



ELEMENTOS

39

Registradores - Processador 8088/8086

- 4 registradores de segmento:
 - ❖ **CS** (Segmento de Código): endereço da área com as instruções de máquina em execução.
 - ❖ **DS** (Segmento de Dados): endereço da área com os dados do programa.
 - Geralmente aponta para as variáveis globais do programa.
 - ❖ **SS** (Segmento de Pilha): endereço da área com a pilha
 - A pilha armazena informações importantes sobre o estado da máquina, variáveis locais, endereços de retorno e parâmetros de sub-rotinas.
 - ❖ **ES** (Segmento Extra): permite acesso a alguma área da memória quando não é possível usar os outros registradores de segmento.
 - Ex: transferências de bloco de dados.

ELEMENTOS

40

Registradores - Processador 8088/8086

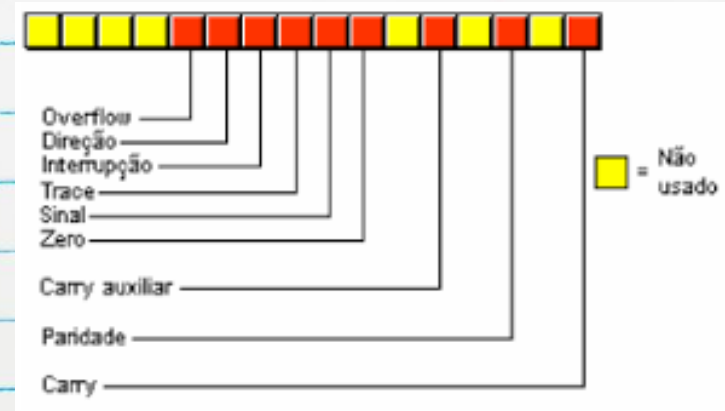
- 5 registradores de offset:
 - ❖ **PC** (Program Counter) ou **IP** (Instruction Pointer): usado em conjunto com o **CS** para apontar a próxima instrução.
 - ❖ **SI** (source index) e **DI** (destiny index): utilizados para mover blocos de bytes de um lugar (**SI**) para outro (**DI**) e como ponteiros para endereçamento (junto com os registradores **CS**, **DS**, **SS** e **ES**).
 - ❖ **BP** (Base Pointer): usado em conjunto com o **SS** para apontar a base da pilha.
 - Similar ao registrador **BX**.
 - Usado para acessar parâmetros e variáveis locais.
 - ❖ **SP** (Stack Pointer): usado em conjunto com o **SS** para apontar o topo da pilha

ELEMENTOS

47

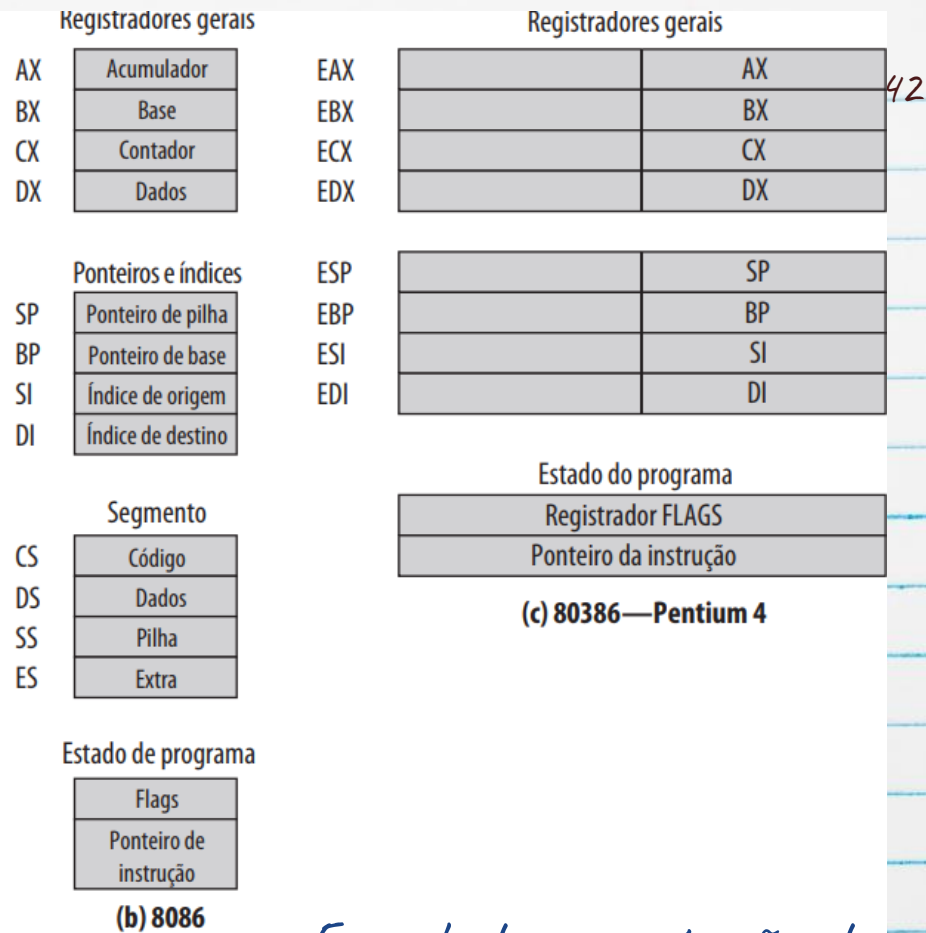
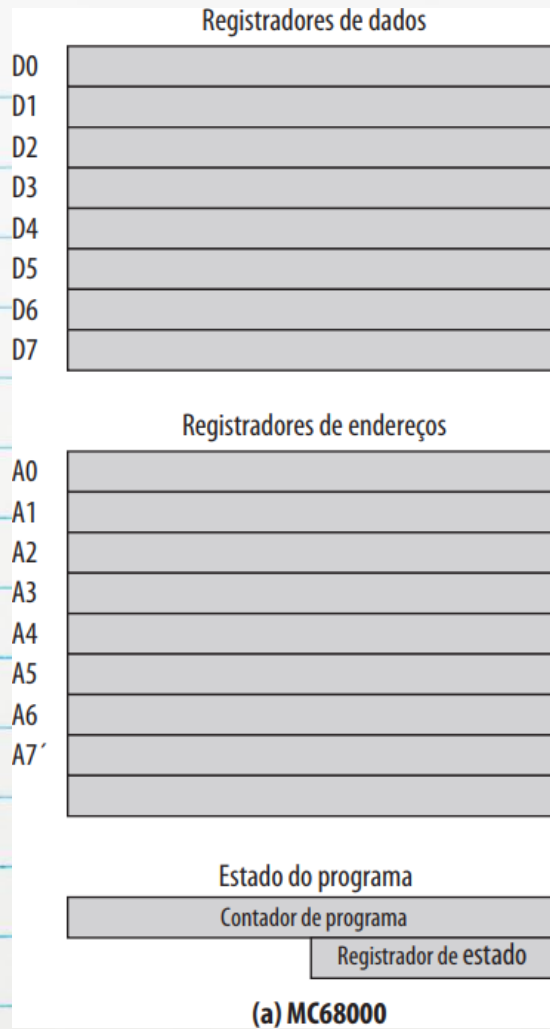
Registradores – Processador 8088/8086

- 1 registrador de estado do processador (PSW):
 - ❖ Registrador especial composto por sinalizadores (flags) que ajudam a determinar o estado atual do processador.
 - Coleção de valores de 1 bit.
 - ❖ Apenas 9 bits são utilizados.
 - 4 mais utilizados:
 - ZF - zero;
 - SF - sinal; e
 - OF - overflow ou underflow
 - CF - carry ("vai um") ou borrow ("vem um");



ELEMENTOS

Registradores - Processador 8088/8086



Exemplo das organizações de registradores do microprocessador

- *Elementos*

Tamanho da Palavra

ELEMENTOS

44

Tamanho da Palavra

- Melhora a capacidade de processamento de uma CPU
 - ❖ A velocidade com que realiza o ciclo de uma instrução;
- Determina o tamanho dos elementos ligados à área de processamento, à ULA e aos registradores de dados.
 - ❖ Atualmente, o barramento de dados tem uma largura maior que o tamanho da palavra
- Influencia no desempenho global de toda a CPU e, por conseguinte, do sistema como um todo.
 - ❖ Um tamanho maior ou menor de palavra acarreta diferenças acentuadas de desempenho da CPU.

ELEMENTOS

45

Tamanho da Palavra

- *Impacta ...*

- ❖ *Tempo maior ou menor na execução de instruções com operações matemáticas na ULA;*
- ❖ *Tamanho escolhido para o barramento interno e externo da CPU;*
- ❖ *Implementação física do acesso à memória.*

- Elementos

Unidade de Controle

ELEMENTOS

47

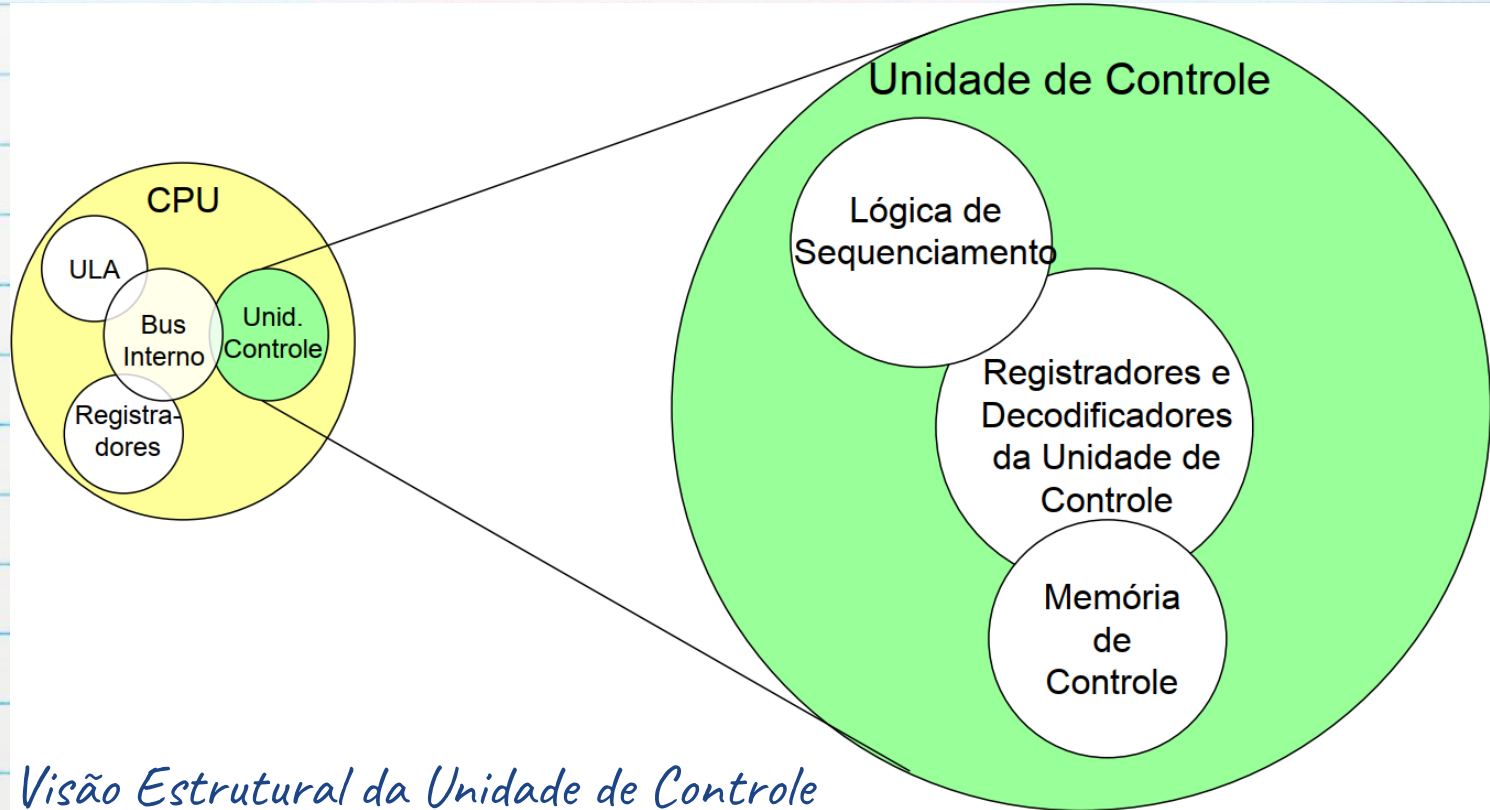
Unidade de Controle

- Dispositivo mais complexo da CPU
 - ❖ Conecta a todos os principais elementos do processador e ao barramento de controle.
- Gerencia os recursos disponíveis e o fluxo de dados entre os componentes
- Controla a execução das instruções pela CPU:
 - ❖ Busca as instruções ISA na memória principal.
 - ❖ Decodificação das instruções (geração dos sinais de controle correspondentes).
 - ❖ Sequenciamento das operações.
 - ❖ Disparo da execução (envio dos sinais de controle).

ELEMENTOS

Unidade de Controle

48



ELEMENTOS

49

Unidade de Controle

- Possui lógica necessária para realizar a movimentação de dados e instruções de e para CPU
 - ❖ Usa sinais de controle emitidos em instantes de tempo programados.
 - Ocorrem em vários instantes durante o período de realização de um ciclo de instrução
 - De modo geral, todos possuem uma duração fixa e igual, originada em um gerador de sinais denominado *clock*.
- Representa uma das partes mais difíceis de ser projetada em um computador, devido à complexidade dos processadores

ELEMENTOS

Unidade de Controle

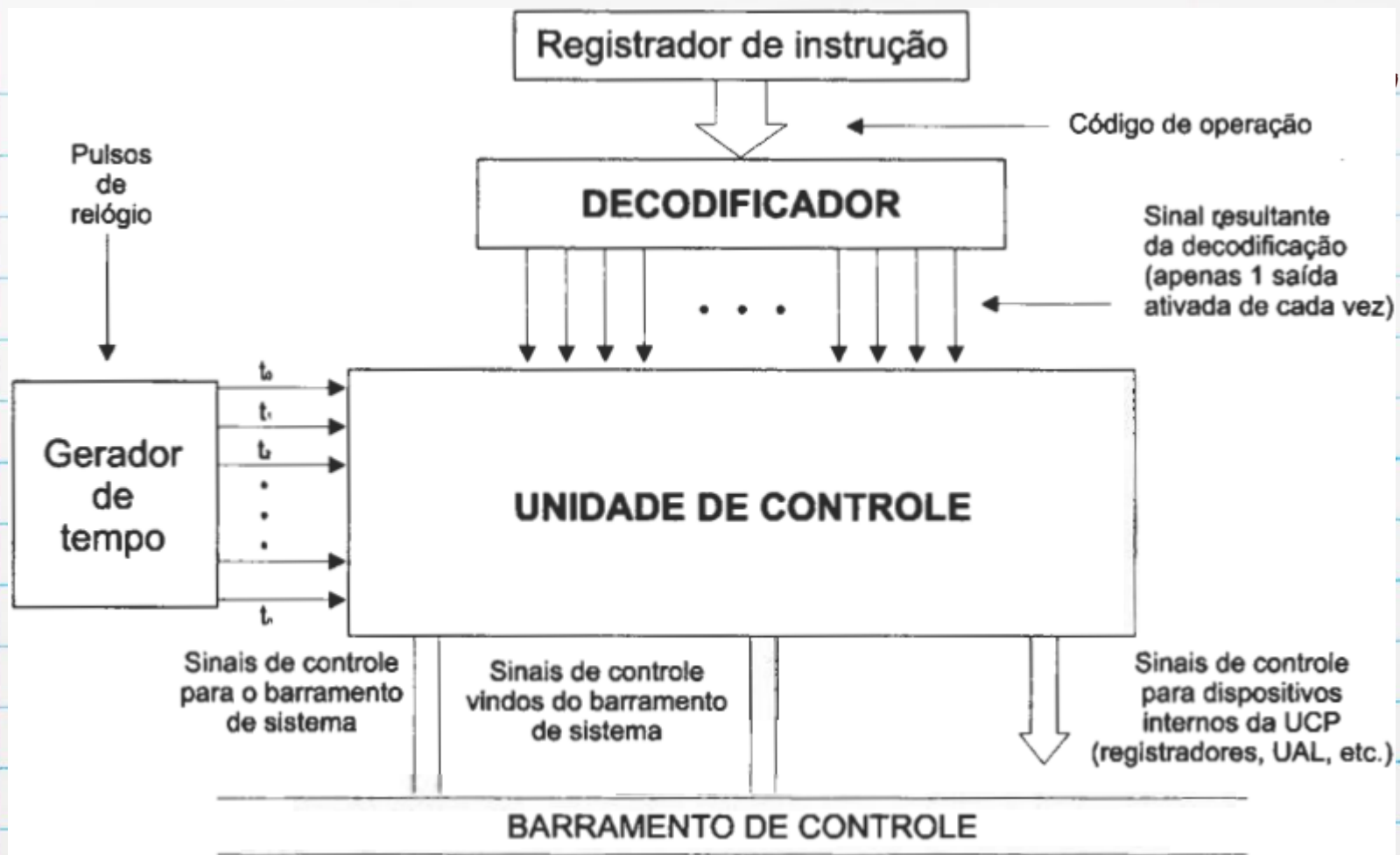


Diagrama em bloco simplificado da função controle.

ELEMENTOS

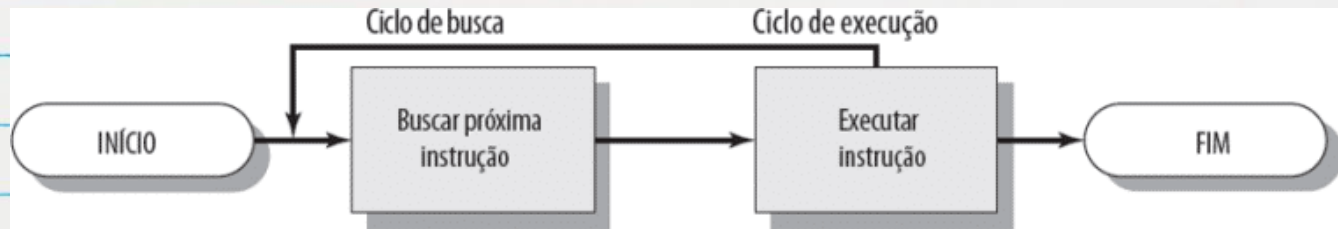
51

Unidade de Controle

- *Microeventos ou microoperações*
 - ❖ *Menor parte individualmente executável pelo processador.*
 - ❖ *Realizada por iniciativa de um pulso originado na UC em decorrência de uma prévia programação*
 - ❖ *podem ser iniciados segundo um de dois princípios de arquitetura*
 - *por microprogramação; ou*
 - *por programação prévia diretamente no hardware.*

Unidade de Controle – Ciclo de Instrução

- *Ciclo de Busca-Execução (fetch-execute)*
 1. *Busca a instrução (memória → IR);*
 2. *Altera PC para indicar a próxima instrução;*
 3. *Decodifica a instrução atual;*
 4. *Determina o endereço e busca o operando na memória;*
 5. *Executa a operação (sinais de controle);*
 6. *Armazena os resultados;*
 7. *Repete passos anteriores*



Unidade de Controle – Ciclo de Instrução

```
1.  while CPU = ativa do
2.    MAR ← PC
3.    MBR ← MEMÓRIA[MAR]
4.    IR ← MBR:OpCode
5.    PC ← PC + incremento
6.    DECODIFICA INSTRUÇÃO
```

Ciclo de busca da instrução
(*fetch*)

```
7.  if IR = ADD then
8.    MAR ← MBR:Endereço
9.    MBR ← MEMÓRIA[MAR]
10.   ACC ← ACC + MBR
11.  else if IR = JUMP then
12.    PC ← MBR:Endereço
    •
    •
    •
```

Ciclo de execução da instrução
(*execute*)

Comportamento do CPU

Unidade de Controle

- *Modo como o sistema conduz a execução das instruções resulta em diversos tipos de arquitetura de processadores, tais como:*
 - ❖ *Processadores que executam instruções de modo exclusivamente sequencial ou serial;*
 - ❖ *Processadores que executam instruções de modo concorrente, ou tipo pipeline ou por linha de montagem;*
 - ❖ *Processadores que executam várias instruções simultaneamente ou por processamento paralelo;*
 - ❖ *Processadores que realizam processamento vetorial.*

- *Elementos*

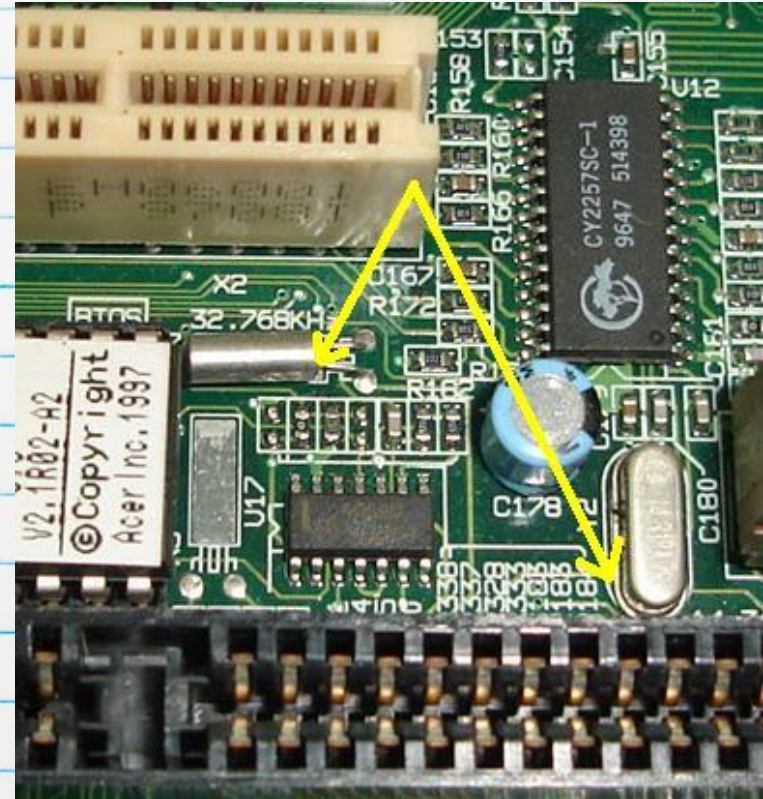
Relógio

ELEMENTOS

Relógio

56

- Dispositivo gerador de pulsos
 - ❖ Gerador de cristal de quartzo
 - ❖ A duração é chamada de ciclo
- Frequência
 - ❖ Unidade de medida do relógio
 - Usada para definir velocidade na CPU
 - ❖ Quantidade de vezes em que este pulso básico se repete em um segundo



ELEMENTOS

Relógio

57

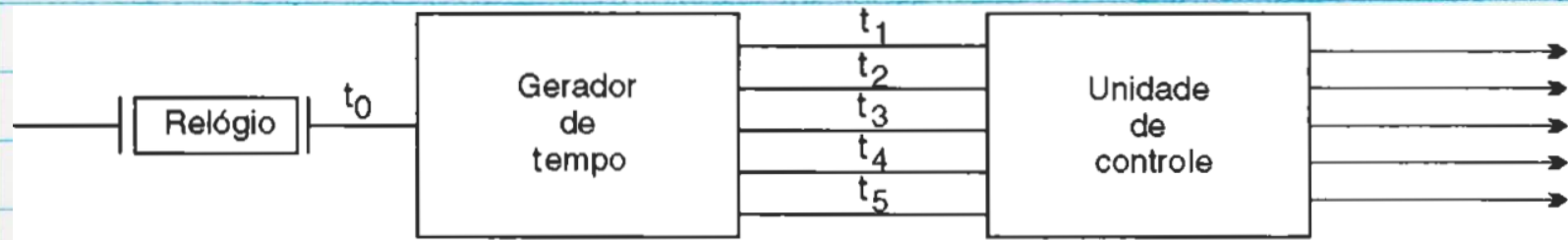


Diagrama em bloco do conjunto de tempo da área de controle

ELEMENTOS

Relógio

58

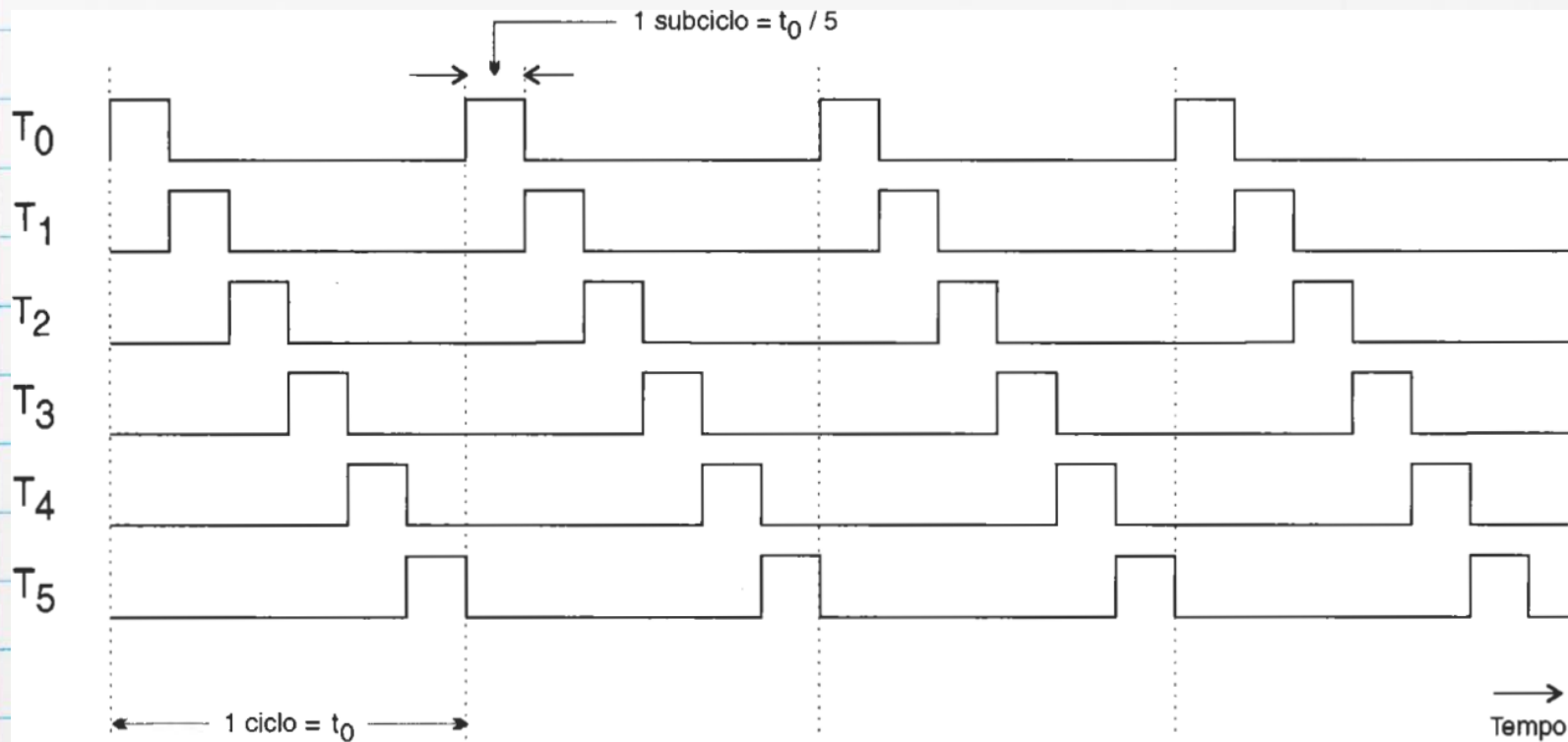


Diagrama de tempo do ciclo do processador (t_0) e seus 5 subciclos

- *Ciclo de relógio ou de máquina (machine cycle)*
 - ❖ *Intervalo de tempo entre o início de um pulso e o início do seguinte.*
 - ❖ *Relacionado à realização de uma operação elementar, durante o ciclo de uma instrução.*
 - *Novas operações básicas são iniciadas em um novo ciclo.*
 - ❖ *Utilizado para atender as relações de tempo requeridas nas operações (sincronismo)*
 - ❖ *Micro operações*
 - *Ciclo de máquina dividido em ciclos menores (subciclos);*
 - *Cada subciclo aciona um passo diferente da operação elementar.*

ELEMENTOS

Relógio

60

- A execução de uma instrução consome um certo número de ciclos.
 - ❖ Varia de acordo com o número de operações básicas requeridas e o tempo de execução de cada uma delas.
- O tamanho do ciclo é um dos fatores que determinam o desempenho de um processador.
 - ❖ Considerado como um indicador menos técnico de desempenho
 - ❖ tamanho do ciclo menor \Rightarrow tempo de execução menor \Rightarrow maior quantidade de instruções/seg.

ELEMENTOS

Relógio

61

- Hertz (Hz)

- ❖ 1 ciclo por segundo

- Medida utilizada para a frequência dos relógios

- Mesma unidade de medida de frequência de sinais analógicos

- ❖ A duração de um ciclo (seu período) é o inverso da frequência

- Exemplo

- ❖ Processador com frequência de operação de 25MHz

- Relógio oscilando 25 milhões de vezes por segundo

- ❖ Duração de ciclo $1/25,000,000$

- Medido em tempo: 0.00000004 ou 40 nano segundos.

- *Elementos*

Decodificador de Instruções

ELEMENTOS

63

Decodificador de Instruções

- *Identifica a operação que será realizada*
 - ❖ *Correlacionada à instrução cujo código de operação foi decodificado.*

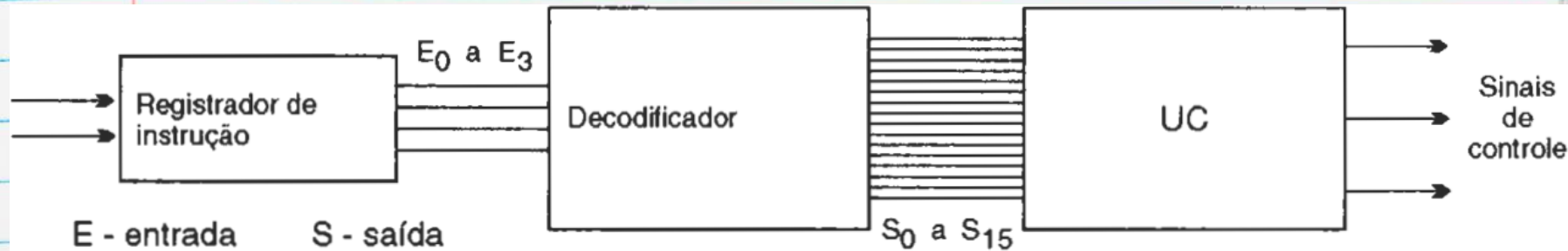


Diagrama em bloco da decodificação de uma UCP

ELEMENTOS

64

Decodificador de Instruções

- Entrada
 - ❖ Um conjunto de N bits previamente escolhido e específico para identificar uma instrução de máquina
- Saída
 - ❖ 2^N linhas de saída
 - ❖ Cada linha aciona de modo diferente a UC.
 - ❖ A UC emite sinais de controle por diferentes caminhos, conforme a linha de saída decodificada

ELEMENTOS

65

Decodificador de Instruções

E ₀	E ₁	E ₂	E ₃	S ₀	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉	S ₁₀	S ₁₁	S ₁₂	S ₁₃	S ₁₄	S ₁₅
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

4 Linhas de entrada e 16 linhas de saída

- Referências

REFERÊNCIAS

- MONTEIRO, Mário A. *Introdução à Organização de Computadores*. 4ª edição, LTC, 2001.
- STALLINGS, William. *Arquitetura e Organização de Computadores*. 5ª edição, Prentice-Hall Brasil, 2002.

