Universidade Presbiteriana Mackenzie

Projeto

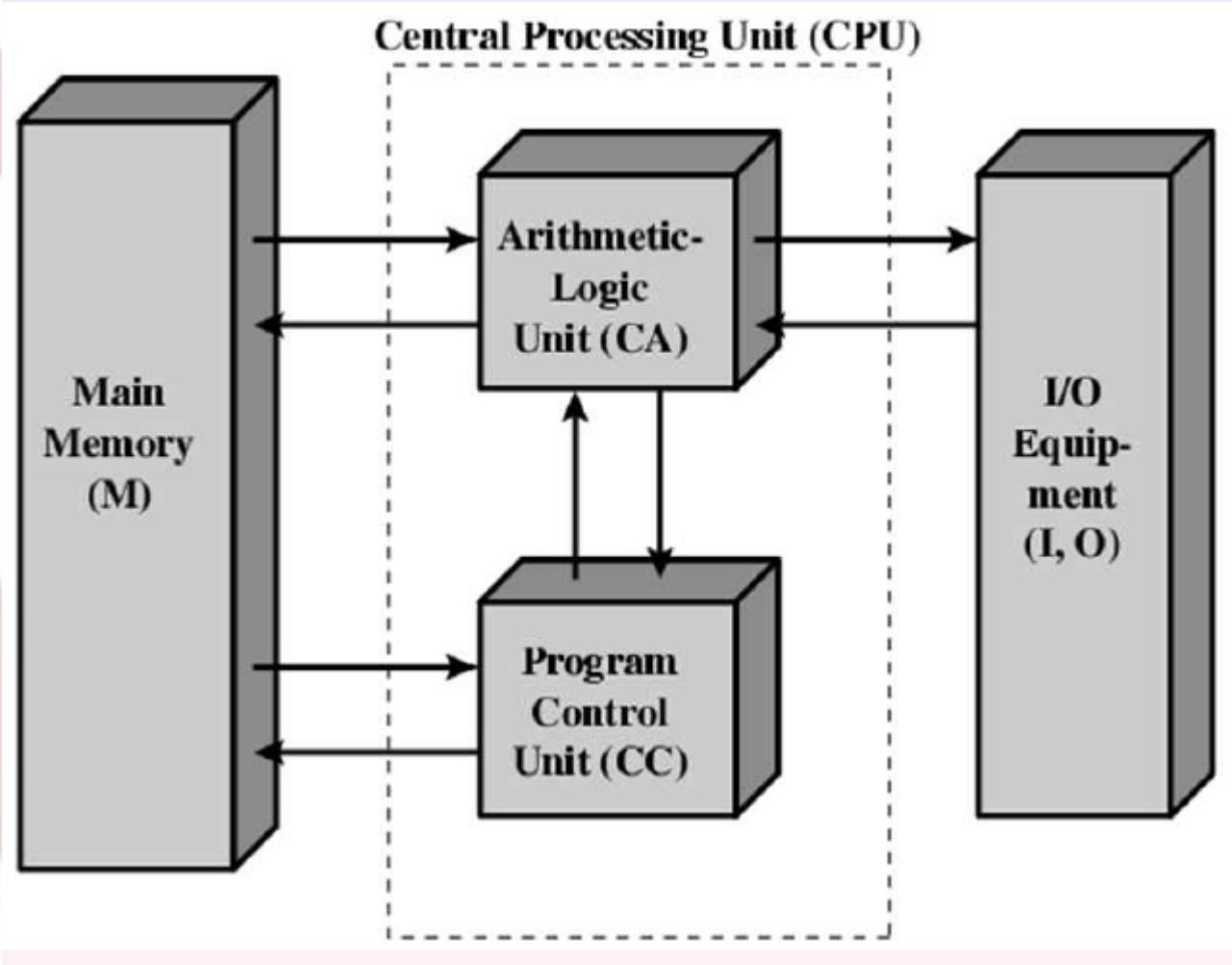
Prof. Fabio Kawaoka Takase

Faculdade de Computação e Informática

Máquina de von Neumann

- Memória principal: armazena dados e instruções.
- ULA: Operações aritméticas com dados binários.
- Unidade de Controle: interpreta instruções na memória e as executa.
- Equipamentos de E/S: operados pela unidade de controle

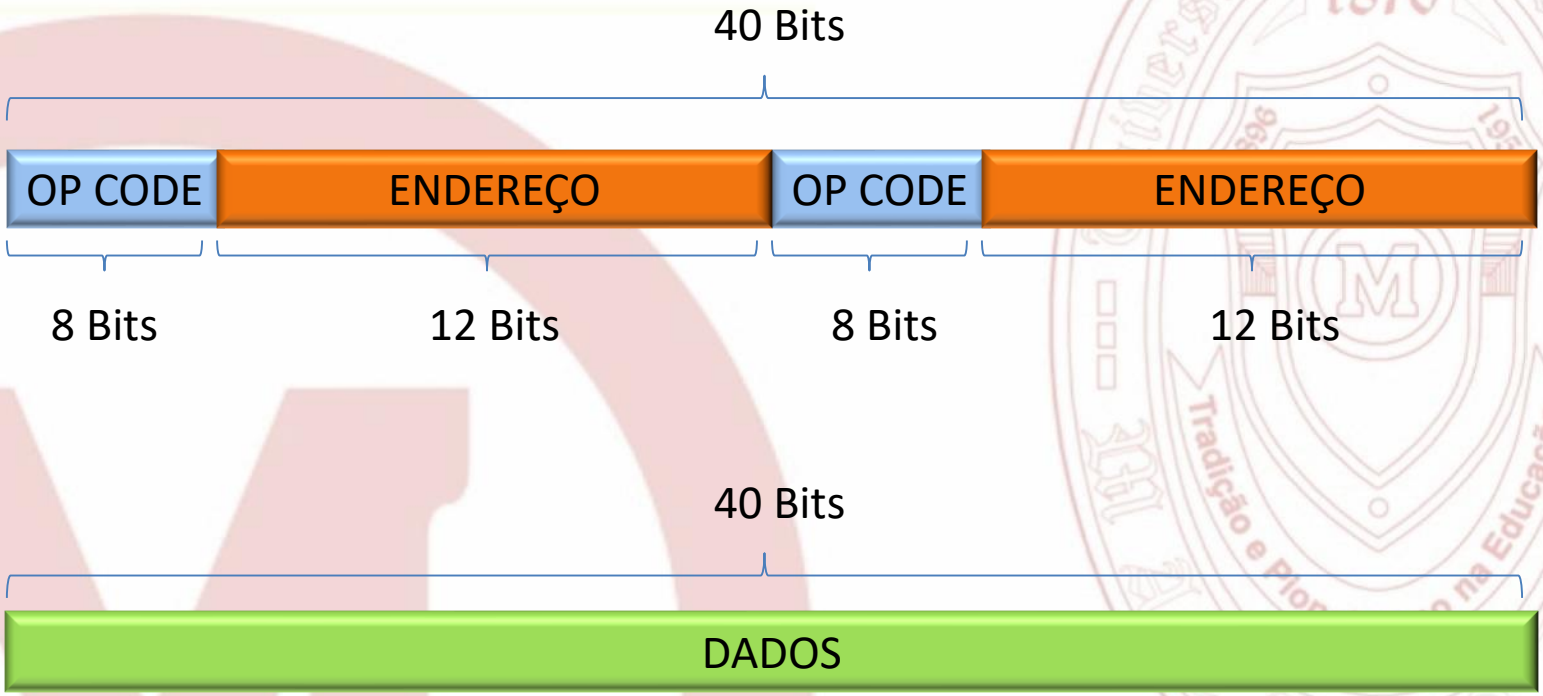
Máquina de von Neumann



IAS – von Neumann

- 1000 palavras de 40 bits cada
- Base binária (números binários)
- Cada palavra de 40 bits continha duas instruções de 20 bits.

Uso de PLD


40 Bits
OP CODE
ENDEREÇO
OP CODE
ENDEREÇO
8 Bits
12 Bits
8 Bits
12 Bits
40 Bits
DADOS

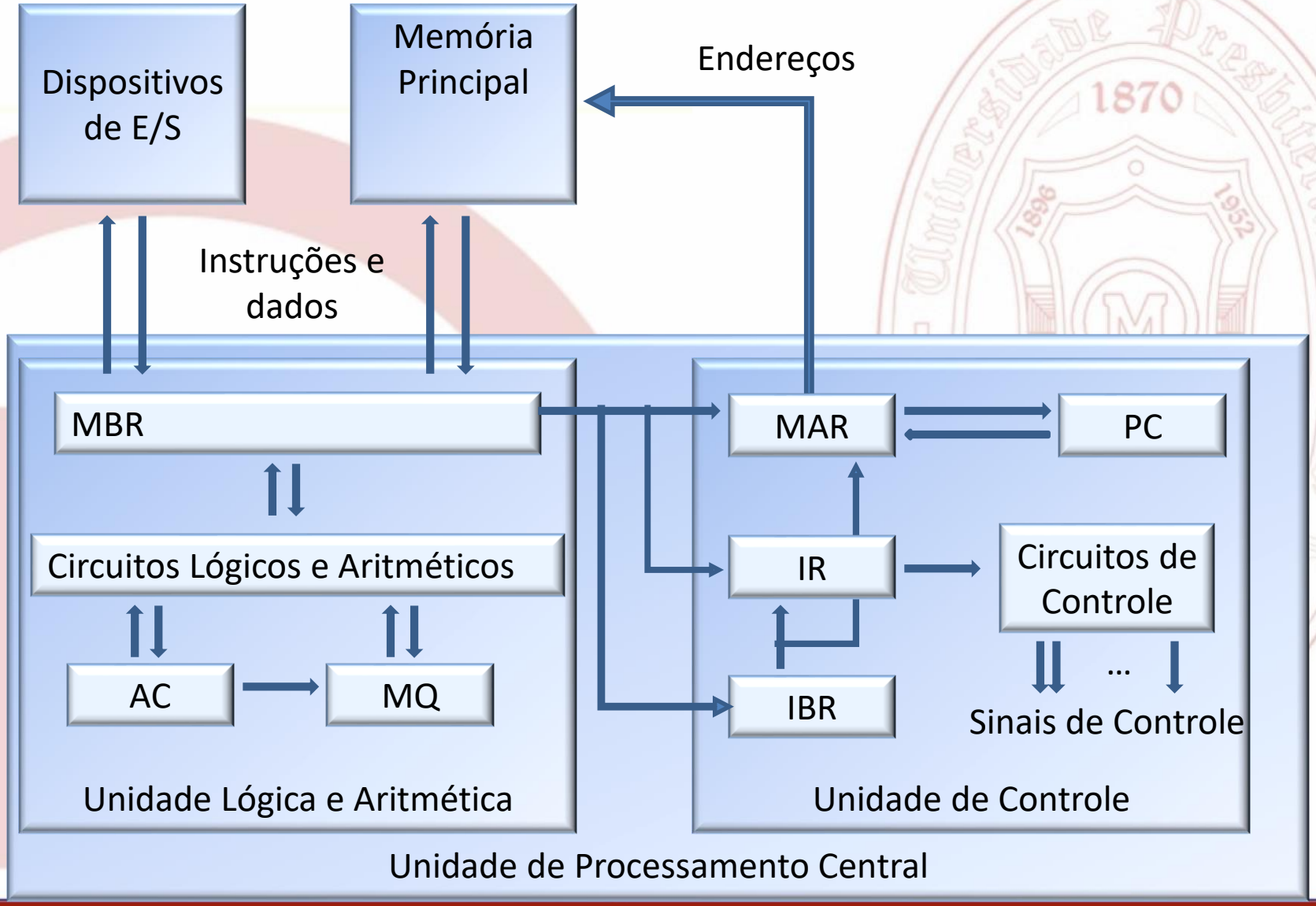
IAS – Registradores

- Armazenamento na UCP
- Memory Buffer Register (MBR): contém uma palavra a ser armazenada em memória ou é utilizado para receber uma palavra da memória.
- Memory Address Register (MAR): Especifica o endereço em memória da palavra que será escrita ou lida no/do MBR
- Instruction Register (IR): contém o código de 8 bits da instrução sendo executada.

IAS – Registradores

- Instruction Buffer Register (IBR): armazena temporariamente a instrução do lado direito de uma palavra.
- Program Counter (PC): contém o endereço do próximo par de instruções a ser lido da memória no ciclo de busca.
- Accumulator (AC) e Multiplier Quotient (MQ): utilizados temporariamente pela ULA em suas operações.

IAS – Estrutura



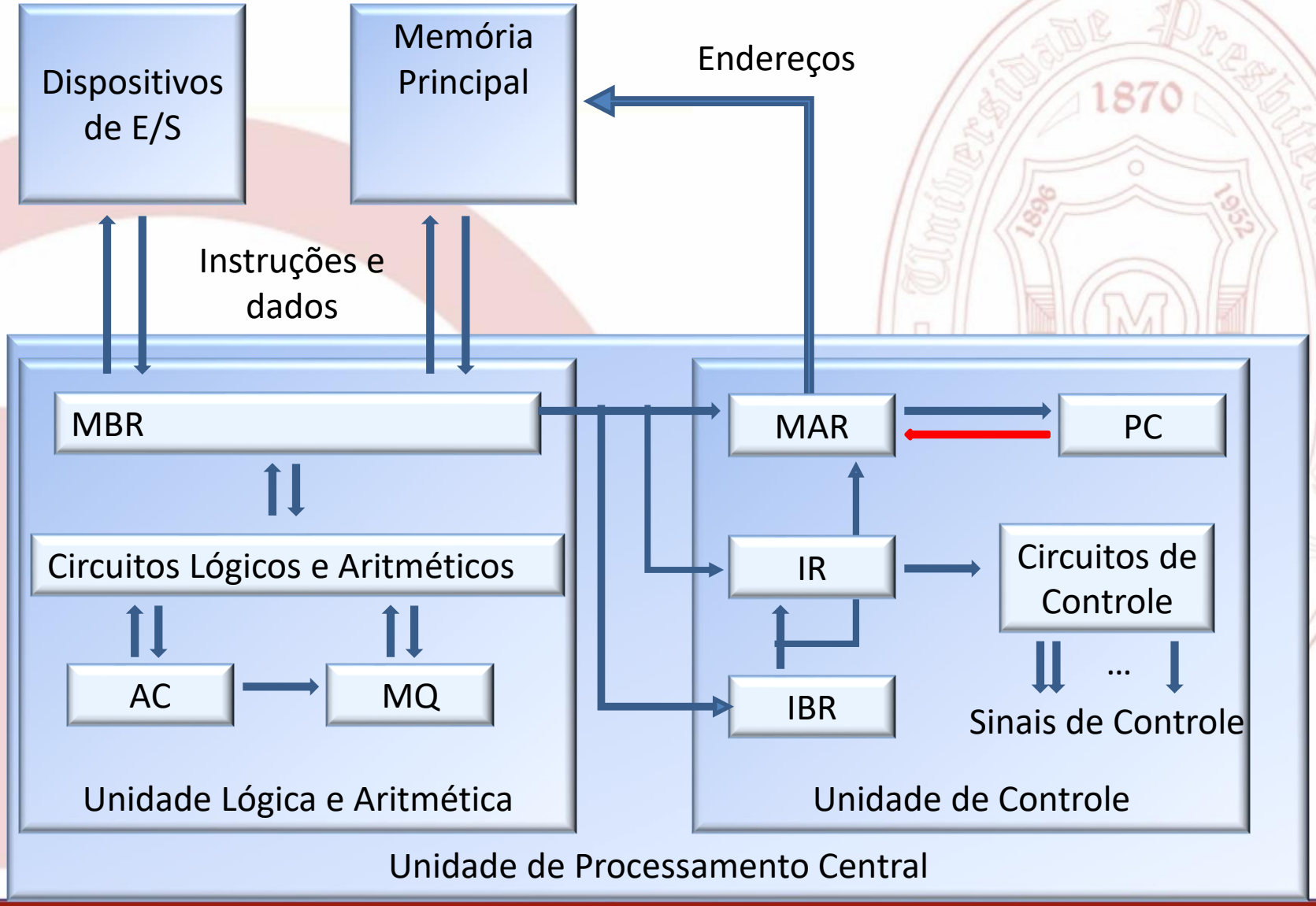
IAS – Operação

- Execução repetida de um Ciclo de Instrução.
- Cada Ciclo de Instrução consiste em dois subciclos:
  - Ciclo de Busca (fetch cycle).
  - Ciclo de Execução (execute cycle).

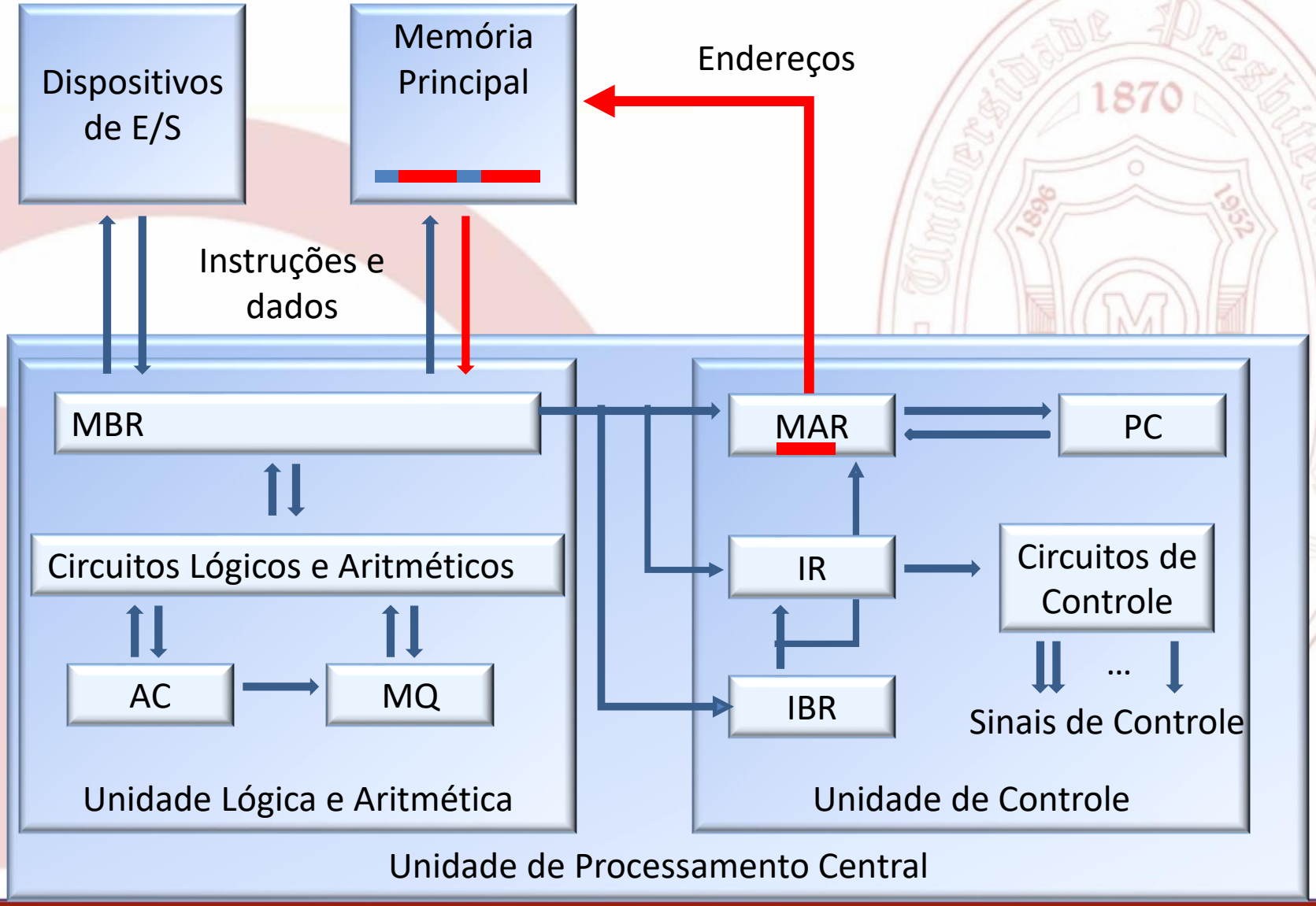
IAS – Ciclo de Busca

- O código da operação da próxima instrução é carregado no IR e a parte do endereço é carregada no MAR.
- A instrução é carregada do IBR ou é obtida da memória carregando uma palavra no MBR, e depois para o IBR, IR e MAR.
- Quando o código da operação da instrução está no IR o Ciclo de Execução se inicia.

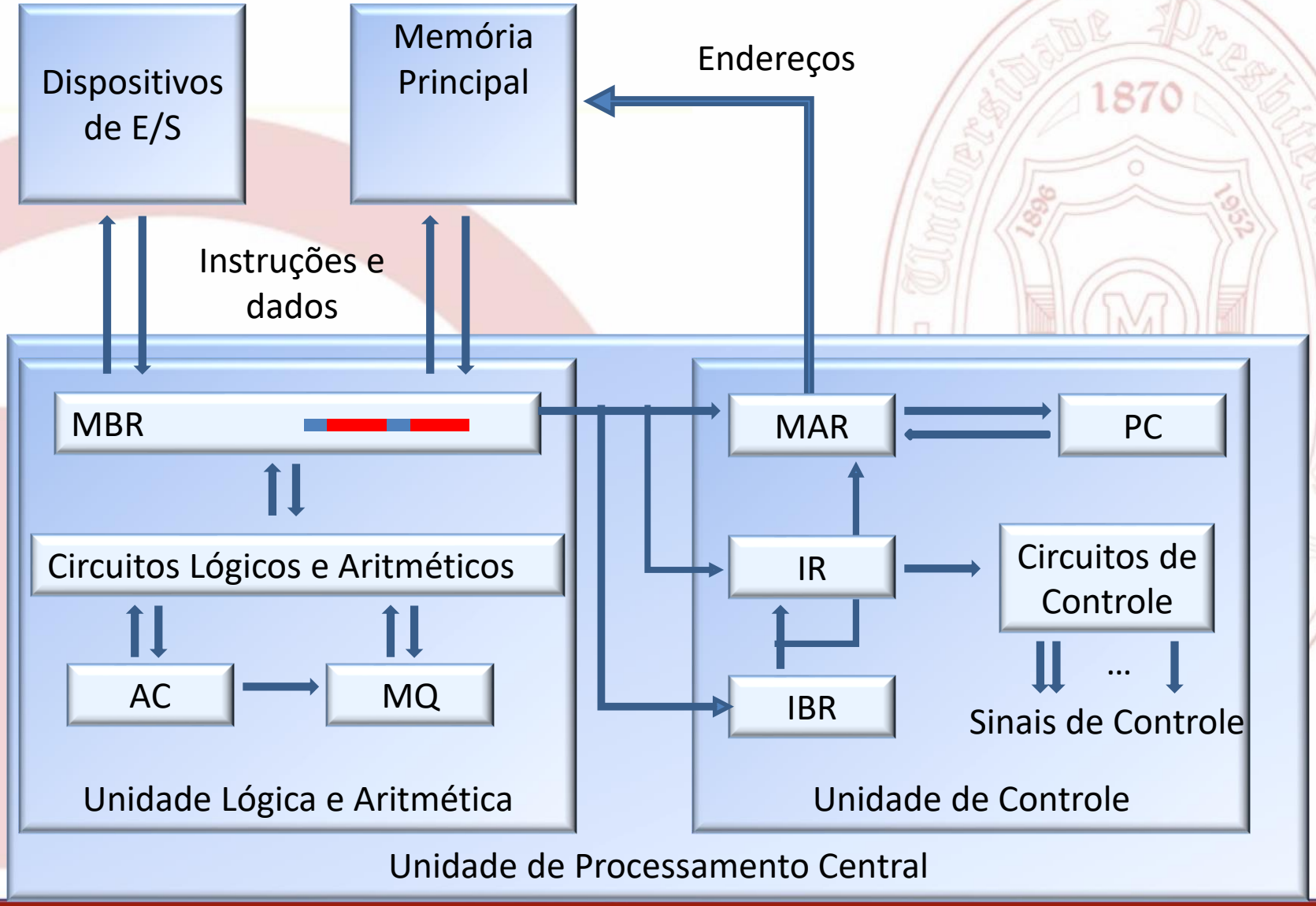
IAS – Ciclo de Busca



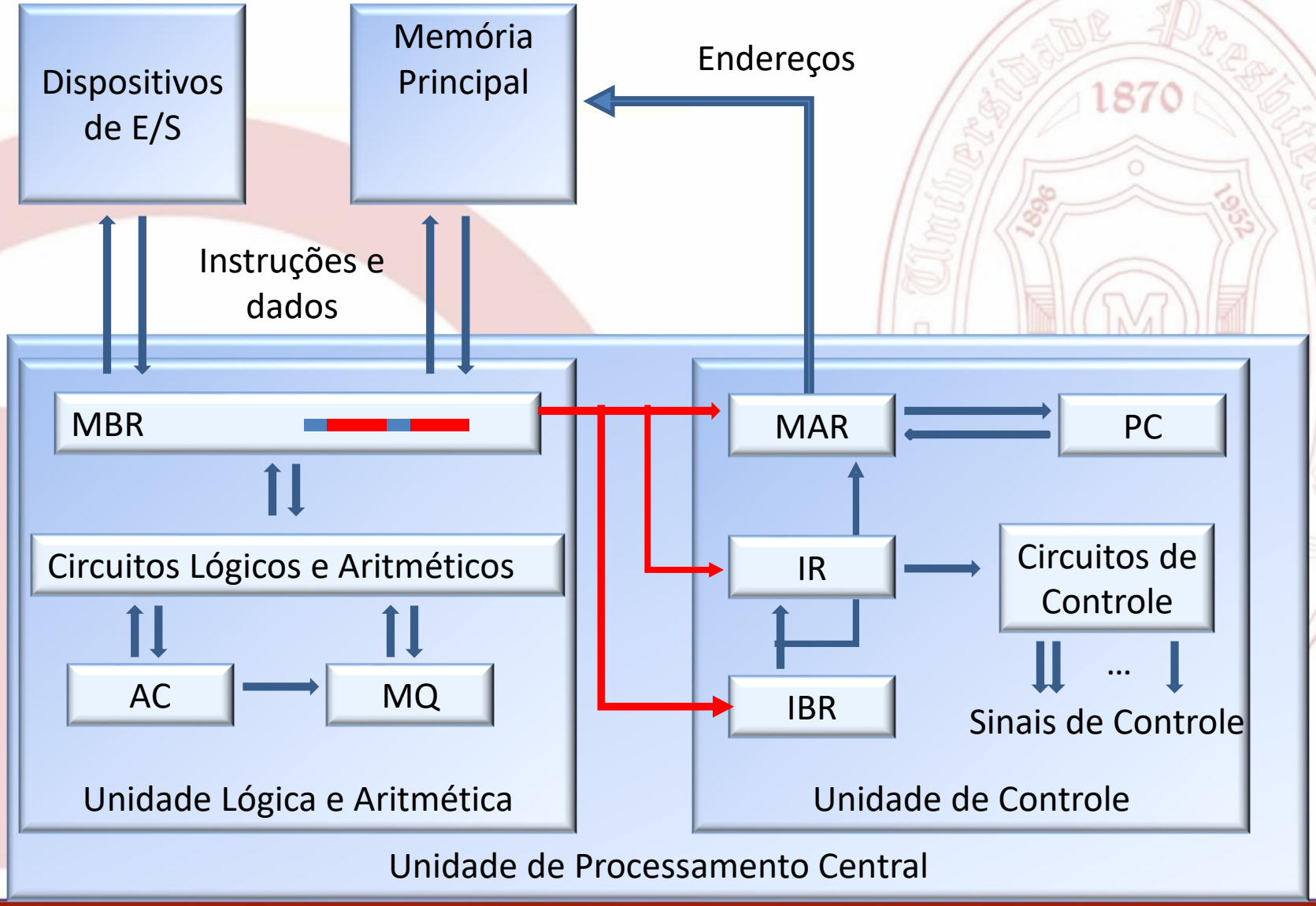
IAS – Ciclo de Busca



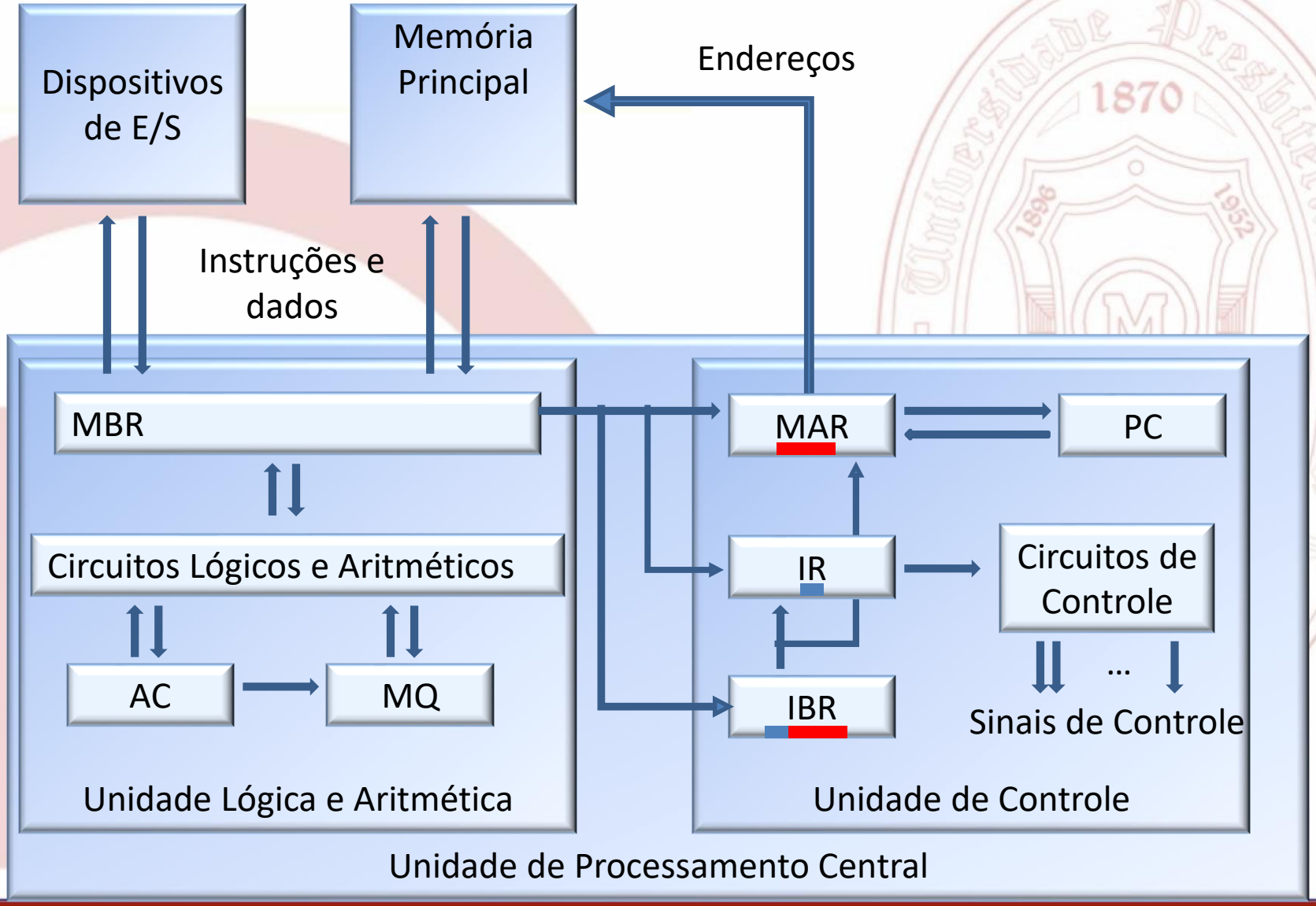
IAS – Ciclo de Busca



IAS – Ciclo de Busca



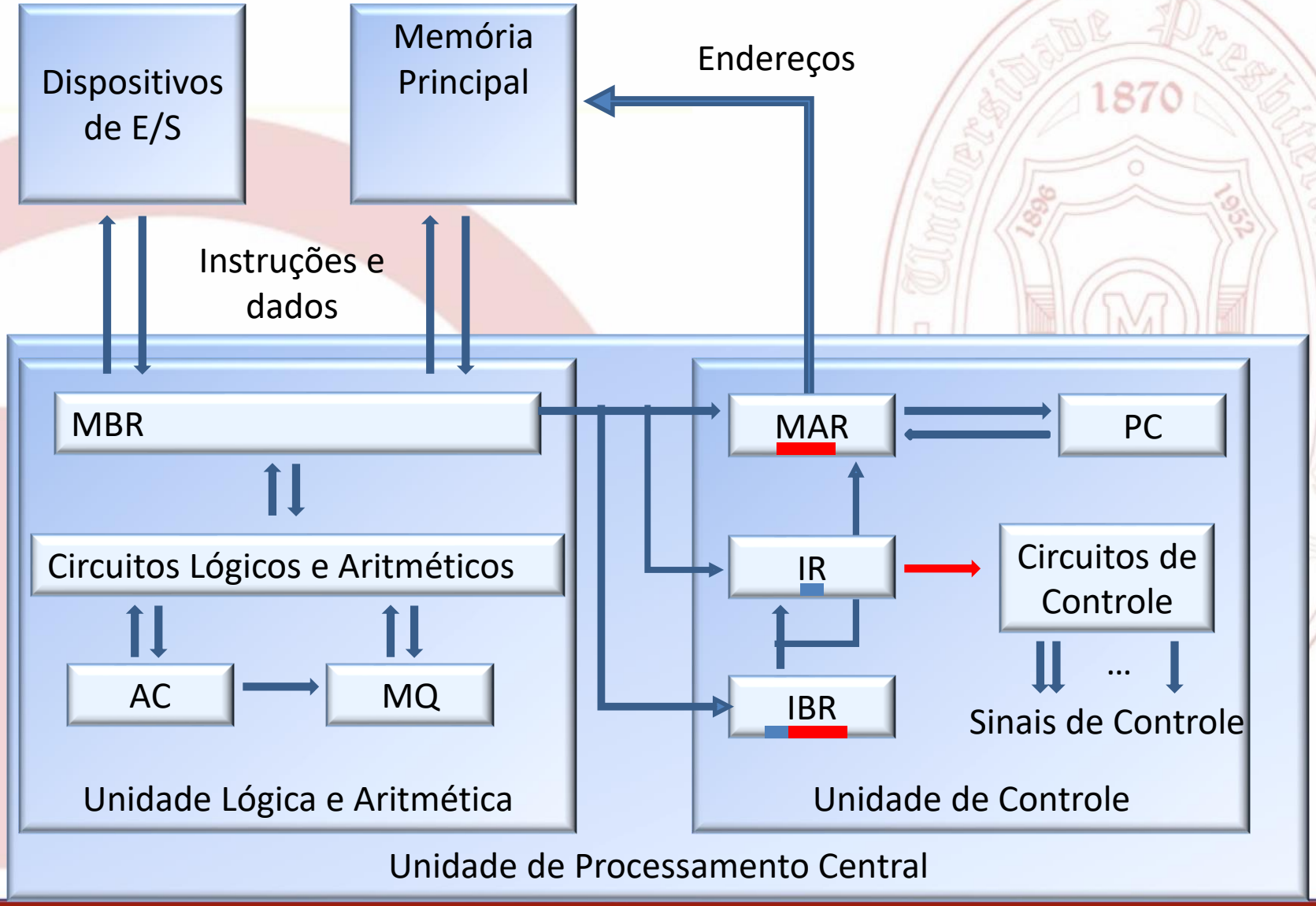
IAS – Ciclo de Busca



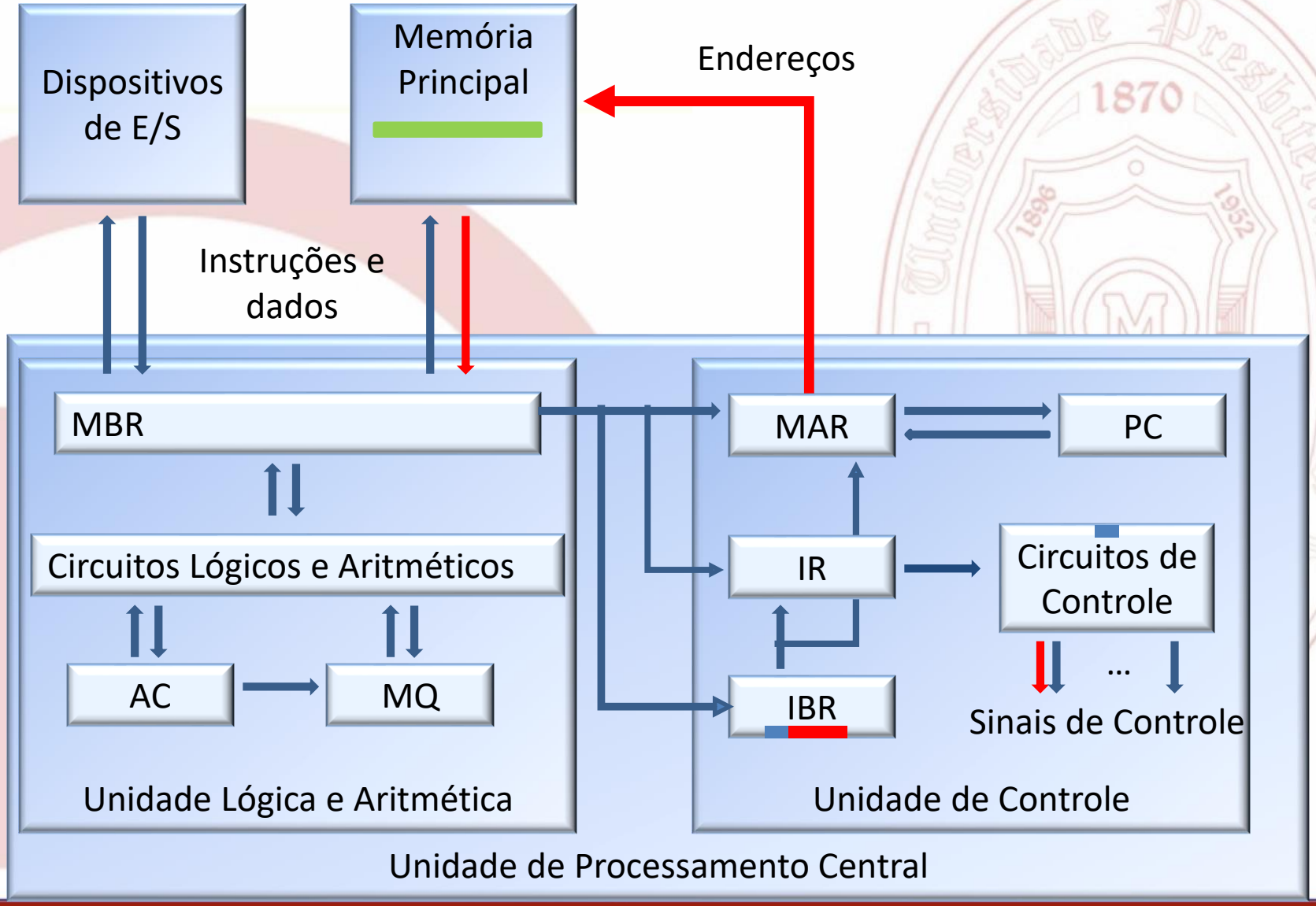
IAS – Ciclo de Execução

- O circuito de controle interpreta o código da operação e executa a instrução enviando os comandos de controle corretos para movimentar dados ou executar operações na ULA.

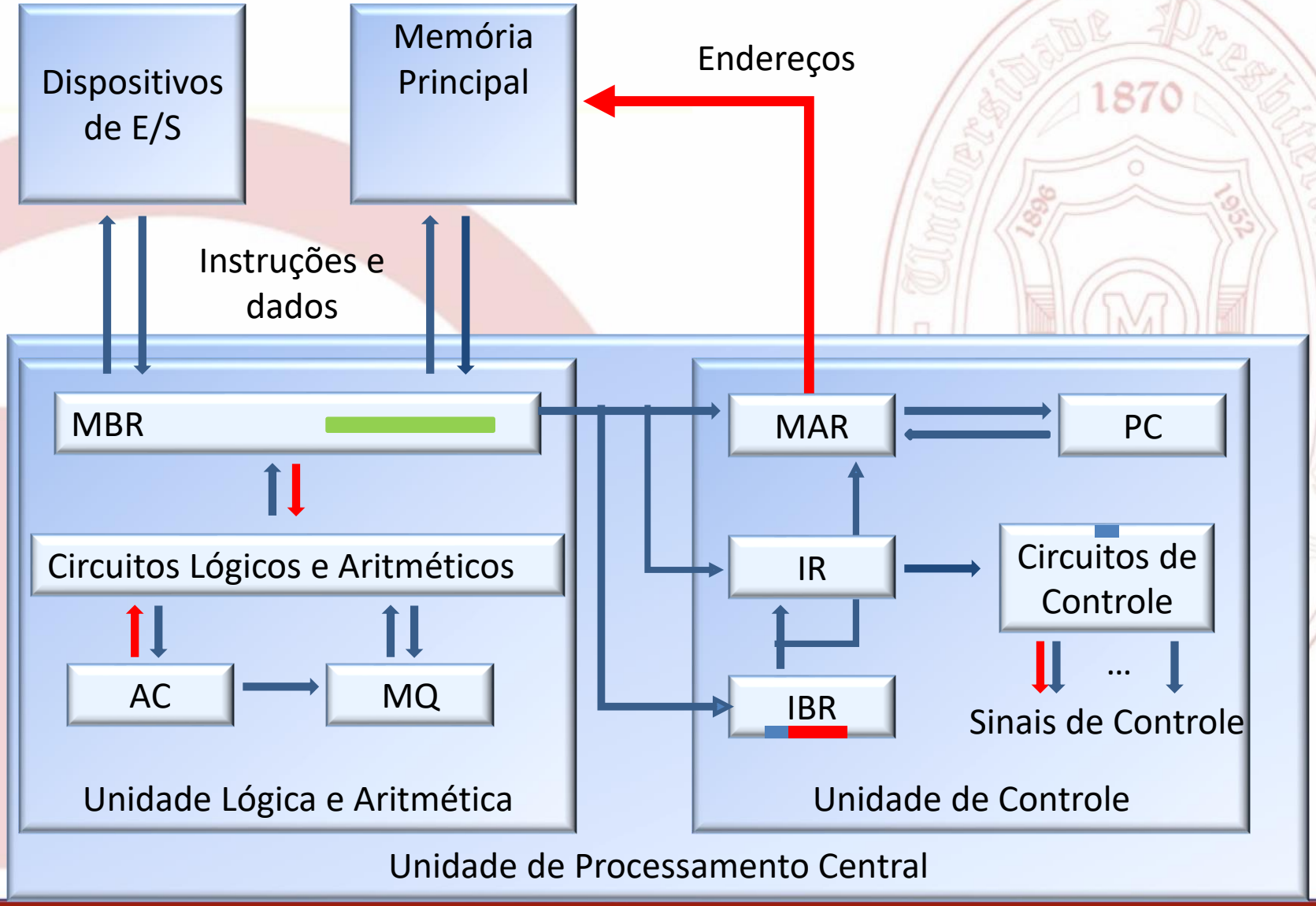
IAS – Ciclo de Execução



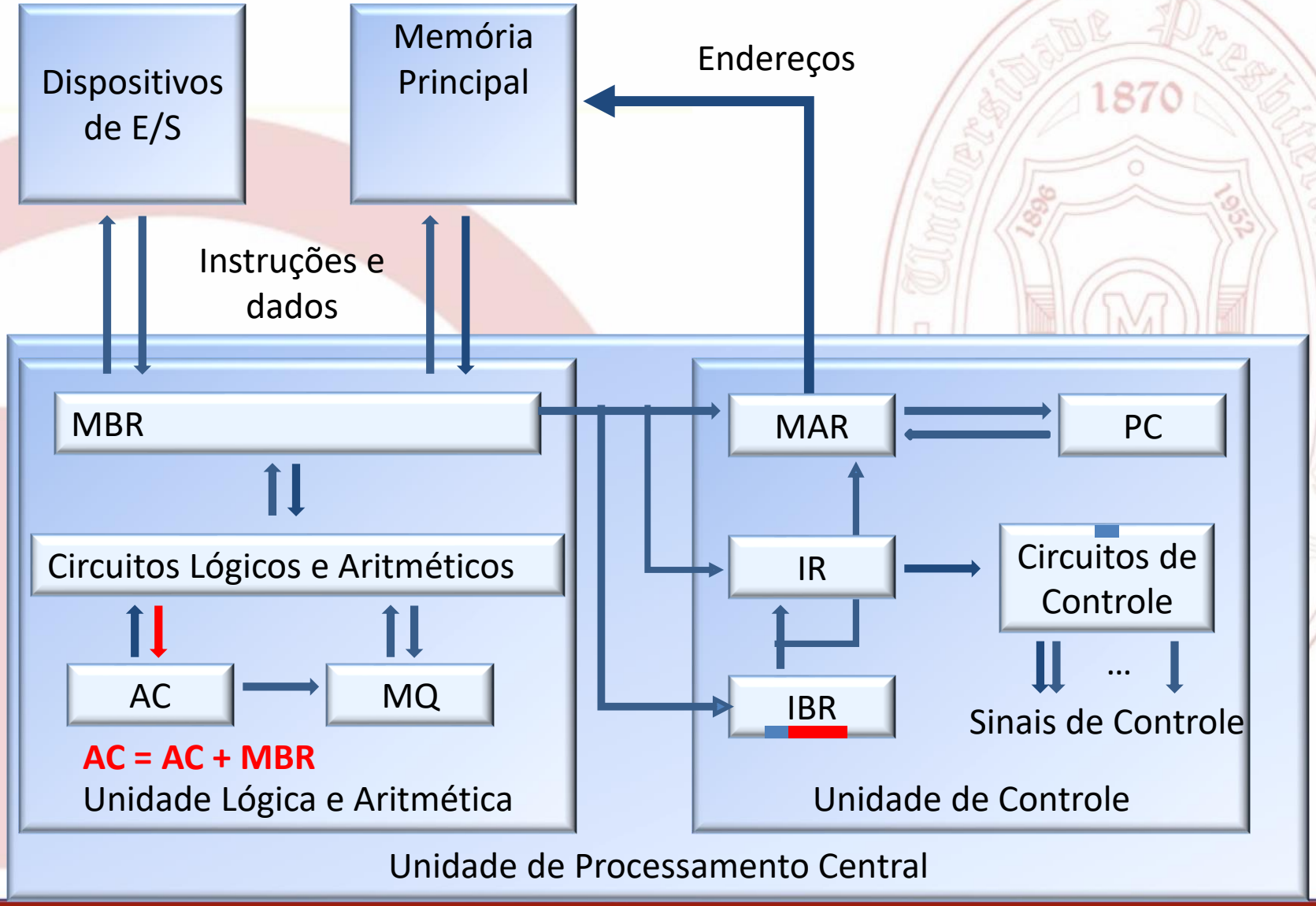
IAS – Ciclo de Execução



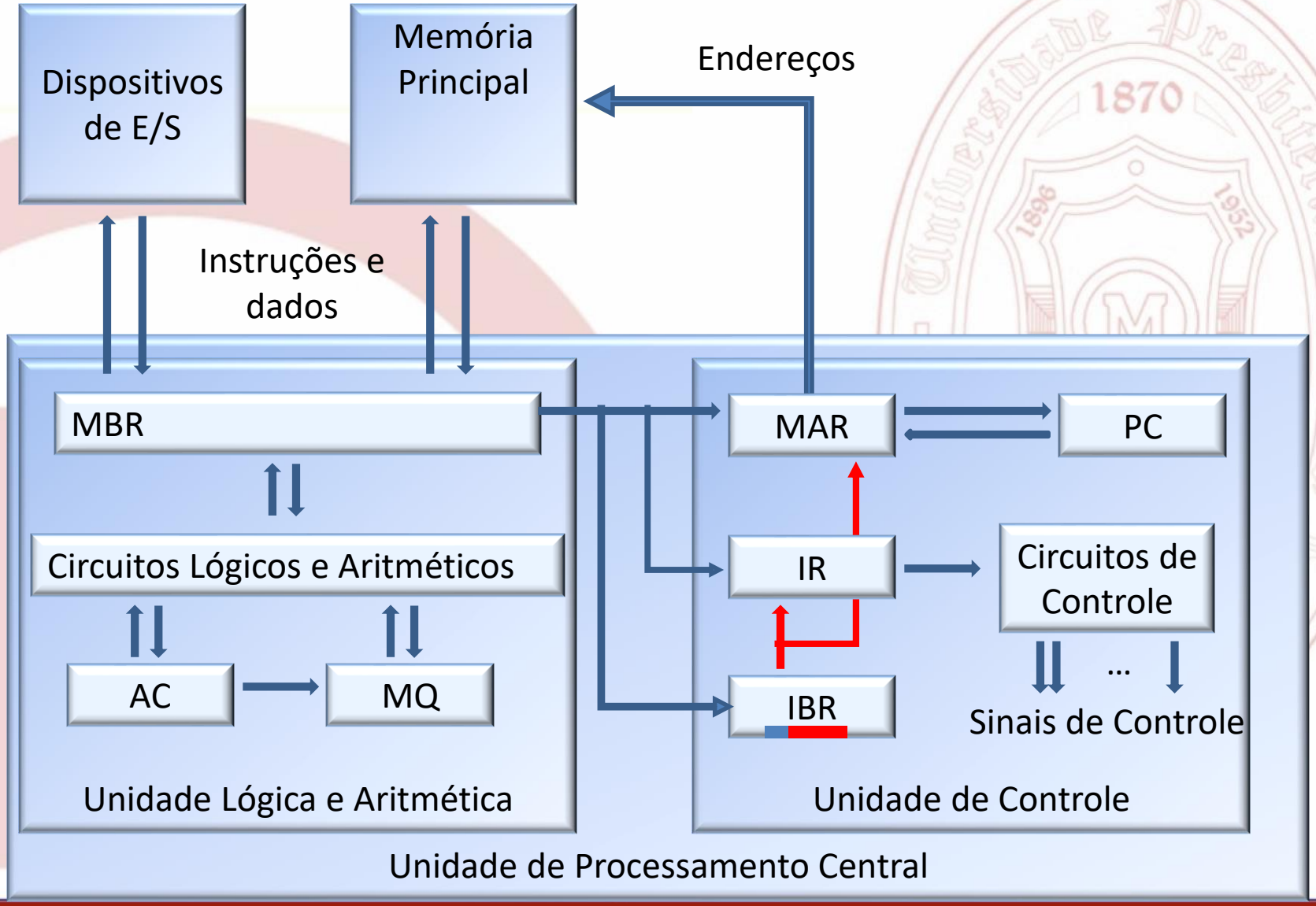
IAS – Ciclo de Execução



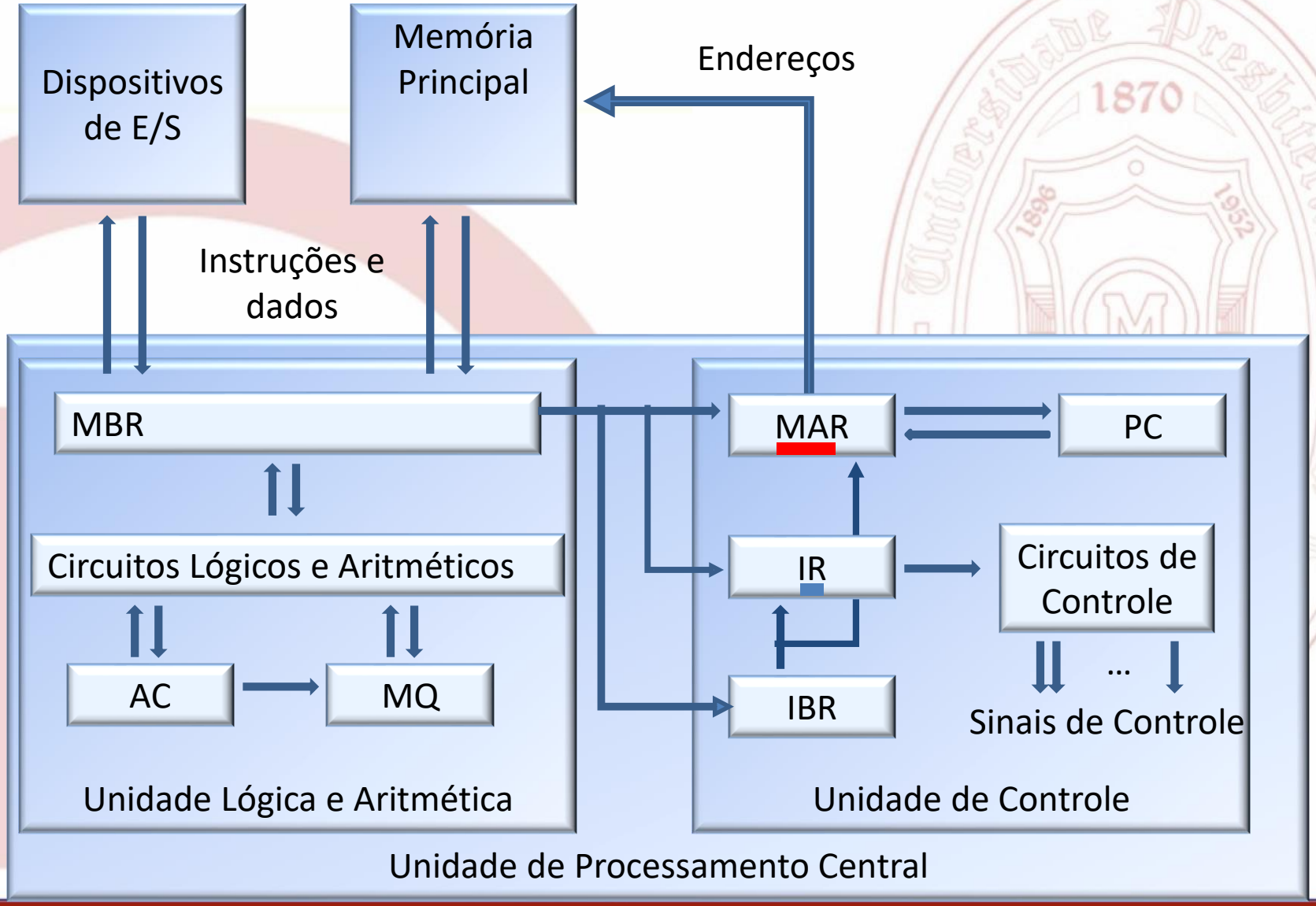
IAS – Ciclo de Execução



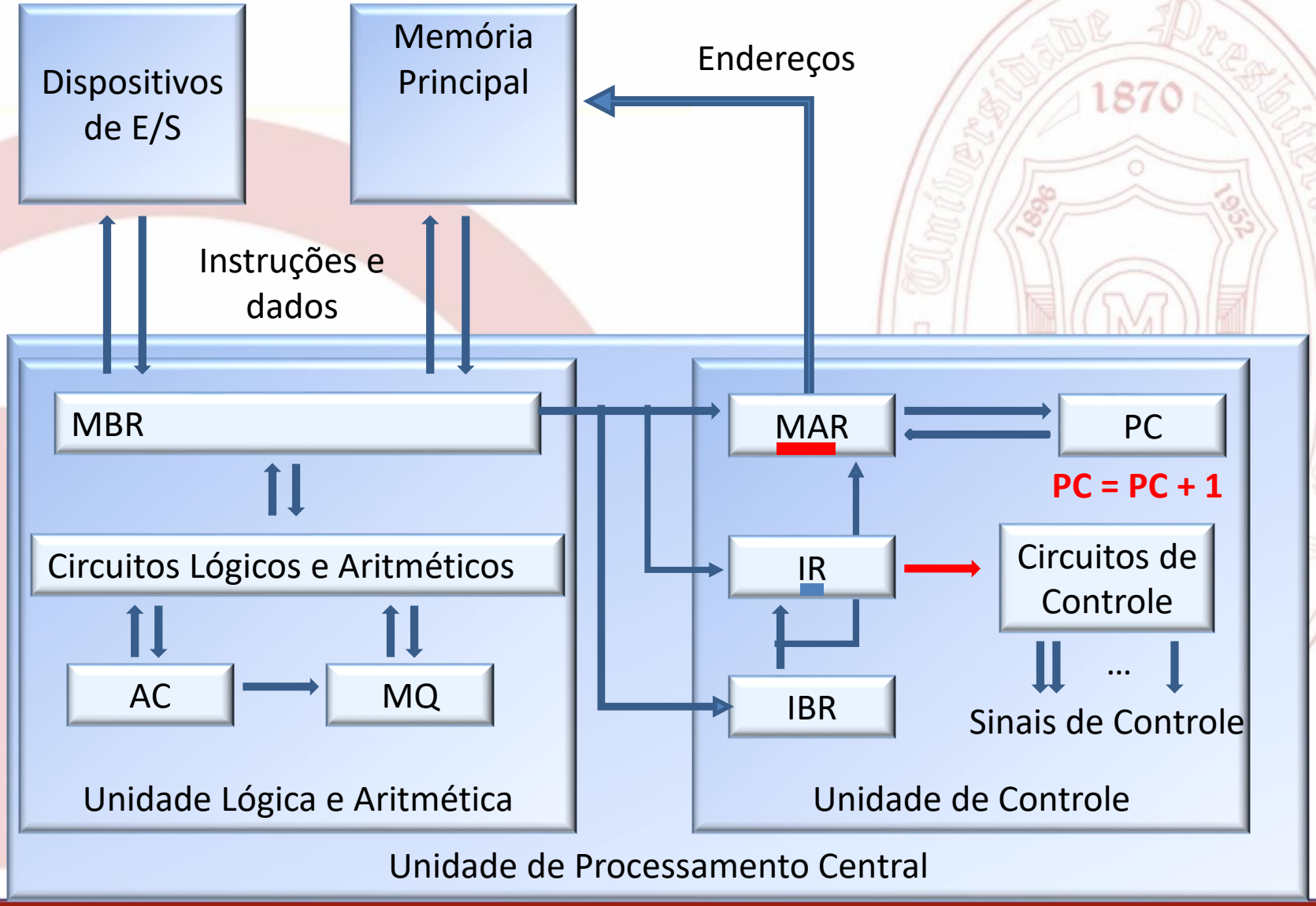
IAS – Ciclo de Execução



IAS – Ciclo de Execução



IAS – Ciclo de Execução



Máquina “Modernas”

- Unidade de controle gera sinais de controle.
- Estes sinais de controle devem ser coerentemente sequenciados para que uma operação possa ser realizada.

Controle de um registrador

- W (Write), R(Read) e Z (Reset) são sinais elementares de controle do registrador.
- Estes sinais elementares são chamados de microcomandos.
- Este microcomandos coexistem para controlar diferentes operações elementares indivisíveis que chamamos de microoperações.
- Somente um microcomando por vez pode estar ativo em cada ciclo do relógio (CLK)

Execução de uma adição

- Passos para realizar uma simples adição
  1. Transferir o conteúdo de R1 para Acc
  2. Transferir o conteúdo de R2 prar TMP.
  3. Somar (TMP) e (Acc).
  4. Armazena a soma em Acc
  5. Armazena o resultado em R1.

Execução de uma adição

Passo	microoperação	CLK	microcomando
MOVE Acc,R1	bus← (R1) Acc← (bus)	1	R_{R1} W_A
MOVE TMP,R2	bus← (R1) TMP← (bus)	2	R_{R2} W_{TMP}
ADD Acc,TMP	saidaSomador ← soma Acc← (bus)	3	R_{TMP} R_{Acc}
STO Acc	Acc← soma	4	W_{Acc}
MOVE R1,Acc	bus← (Acc) R1← (bus)	5	R_{Acc} W_{R1}

Instrução de Máquina

- O computador é projetado para uma determinada quantidade de código de operação.
- Este conjunto de códigos de operação é chamado de repertório de instruções do processador.
- No projeto de uma UCP são definidas as instruções que o processador será capaz de executar: os tipos de instruções do repertório e o comprimento de cada instrução.

Tipos de Instruções

- 5 grandes grupos
  1. Instruções de transferência de dados
  2. Instruções de tratamento de operandos.
  3. Instruções de desvio.
  4. Instruções de entrada e saída.
  5. Instruções outras.

Instruções de transferência de dados

- Instruções que transferem dados, ou blocos de dados, entre diferentes registradores ou posições de memória.

Exemplo: MOVE R1, R2 R2 →(R1)

Instruções de tratamento de operandos

- Instruções que realizam operações aritméticas ou lógicas.

Exemplo: ADD R1, R2 R2 ← (R1) + (R2)

Instruções de desvio

- Instruções que determinam desvio no fluxo do programa, controlando a sequência de execução do programa.

Exemplo: JMP (salto incondicional), Jcond (salto condicional), JSR (salto para subrotina), RET (retorno da subrotina)

JMP end → sempre PC←end

Jcond end → se cond=TRUE PC←end

senão PC←end próxima instrução

Instruções de E/S

- Instruções que realizam a comunicação entre a UCP e as interfaces de entrada e saída.

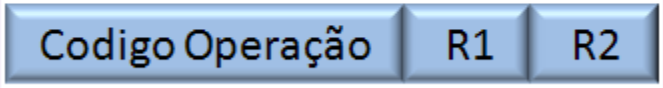
Exemplo: IN port Acc← (port)

Comprimento de instruções

- Depende da quantidade de endereços especificado
- Depende do tipo de ULA utilizada. O que define a quantidade máxima de endereços comportada por uma instrução do processador.
- Depende da quantidade de operandos necessários para a execução da instrução.
- Uma instrução pode ter o comprimento de uma ou mais palavras.

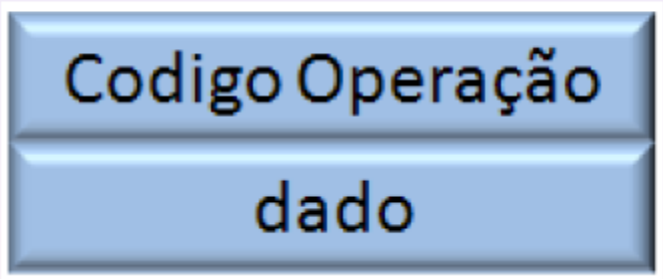
Instruções de 1 palavra

MOVE R1, R2; R2 ← (R1)



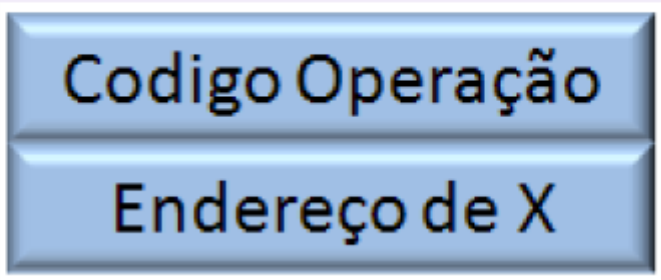
Instruções de 2 palavras

ADI dado; Acc← (Acc) + dado



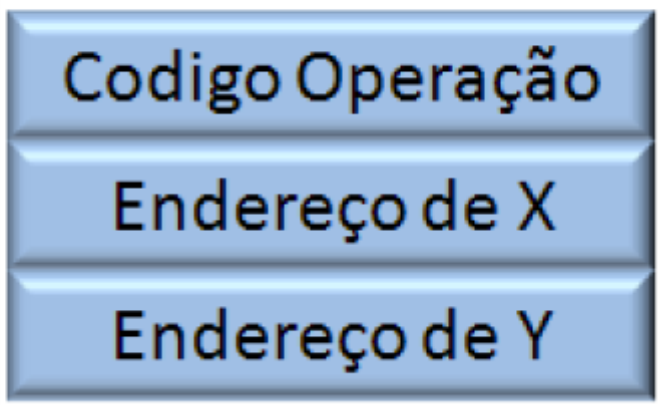
Instruções de 2 palavras

ADD X; Acc← (Acc) + (X)



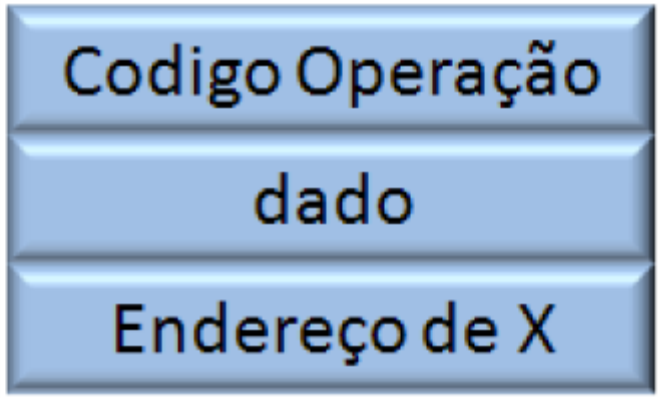
Instruções de 3 palavras

ADD X, Y; Y ← (X) + (Y)



Instruções de 3 palavras

ADD #dado, X; X ← (X) + dado



Execução de Instruções

- Execução de uma instrução dividida em duas fases.
  1. Fase de busca
  2. Fase de execução

Ciclo de máquina

- Sequência de microoperações que são executadas, caracterizando uma operação mais complexa.
- Significado importante no entendimento do funcionamento do computador.
- A execução de instruções é dividida em ciclos de máquina.

Fase de Busca

- Compreende o ciclo de máquina para a leitura do código de operação.
- É o ciclo de máquina que obtém da memória a primeira palavra da instrução.
- A fase de busca é idêntica para todas as instruções.

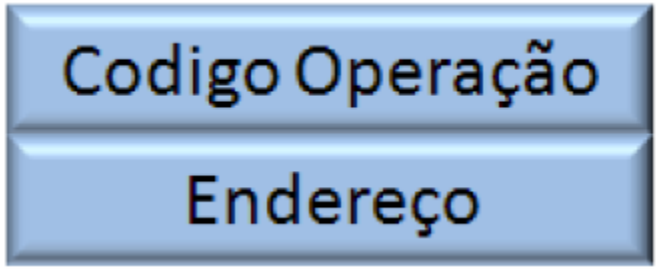
Fase de Execução

- Compreende os ciclos de máquina necessários para a leitura das palavras restantes da instrução, se existirem.
- Compreende os ciclos de máquina necessários para a execução da operação identificada pela instrução.
- É diferente para cada instrução.
- A execução de uma instrução sempre implica na execução de ciclos de máquina que englobam as microoperações necessárias para a execução de cada passo da instrução.

Exemplo 1

Instrução: STA end - transfere o conteúdo do acumulador para a posição de memória end; Utiliza duas palavras de memória.

STA end ⇒ end ← (Acc)

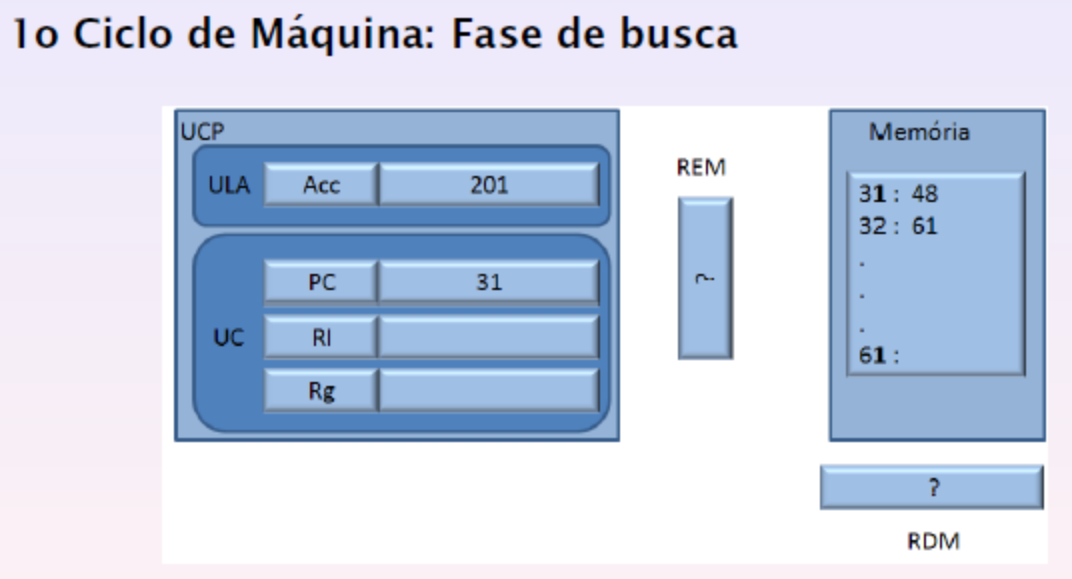


Exemplo 1

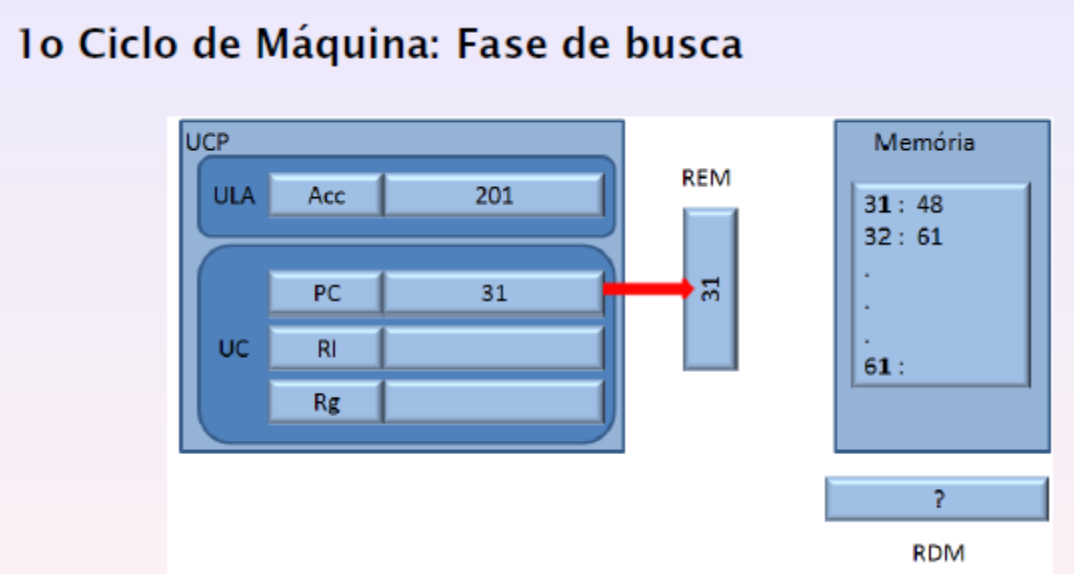
1o Ciclo de Máquina: Fase de busca

CLK	Microoperação
1	REM ← (PC)
2	RDM ← (m) PC ← (PC) + 1
3	RI ← (RDM)

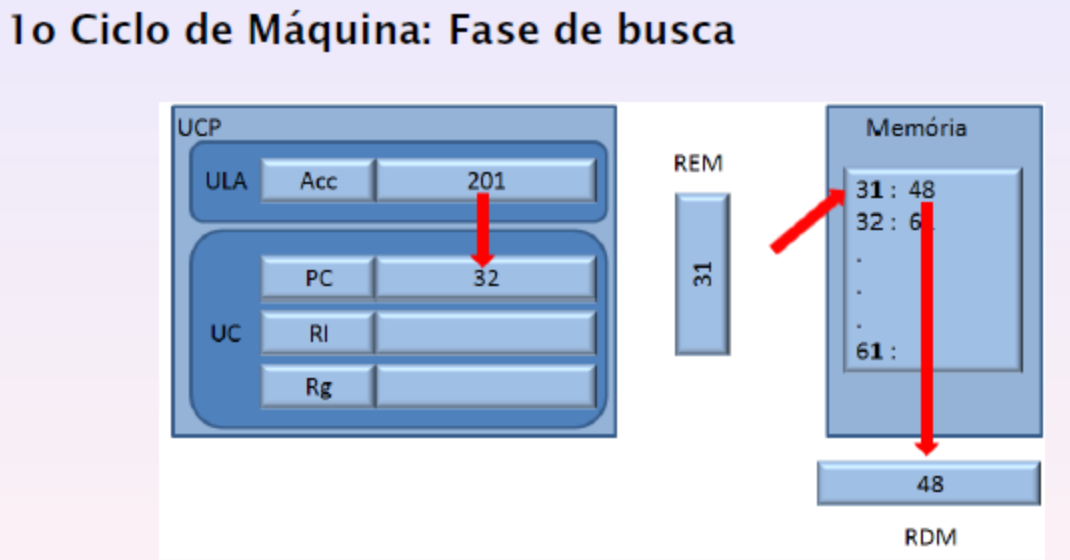
Exemplo 1


1o Ciclo de Máquina: Fase de busca
UCP
ULA
Acc
201
UC
PC
31
RI
Rg
REM
?
Memória
31 : 48
32 : 61
.
.
.
61 :
?
RDM

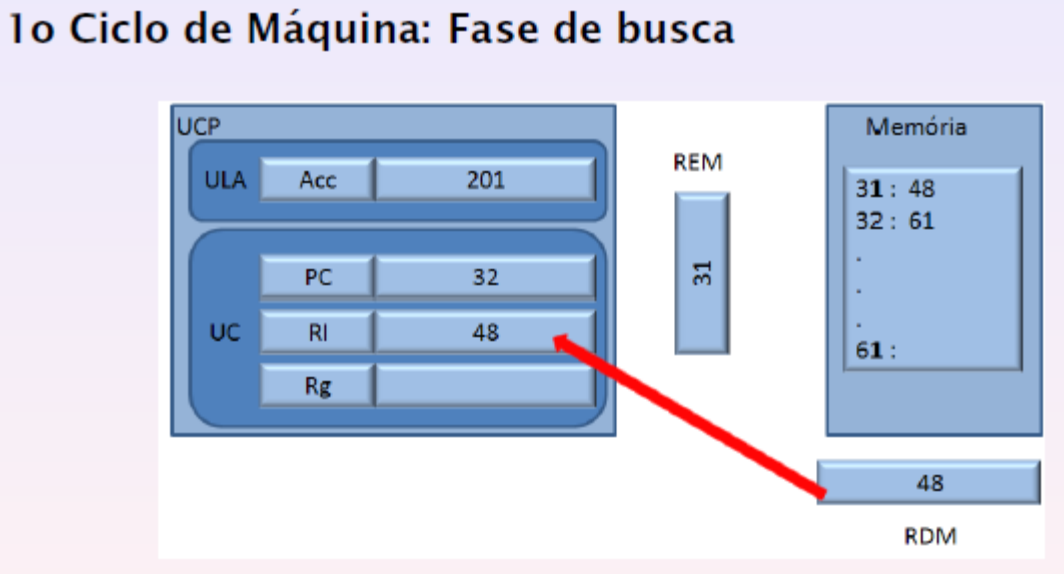
Exemplo 1


1o Ciclo de Máquina: Fase de busca
UCP
ULA
Acc
201
UC
PC
31
RI
Rg
REM
31
Memória
31 : 48
32 : 61
61 :
?
RDM

Exemplo 1


1o Ciclo de Máquina: Fase de busca
UCP
ULA
Acc
201
PC
32
UC
RI
Rg
REM
31
Memória
31 : 48
32 : 6
61 :
48
RDM

Exemplo 1


1o Ciclo de Máquina: Fase de busca
UCP
ULA
Acc
201
UC
PC
32
RI
48
Rg
REM
31
Memória
31 : 48
32 : 61
.
.
.
61 :
48
RDM

Exemplo 1

1o Ciclo de Máquina: Fase de busca

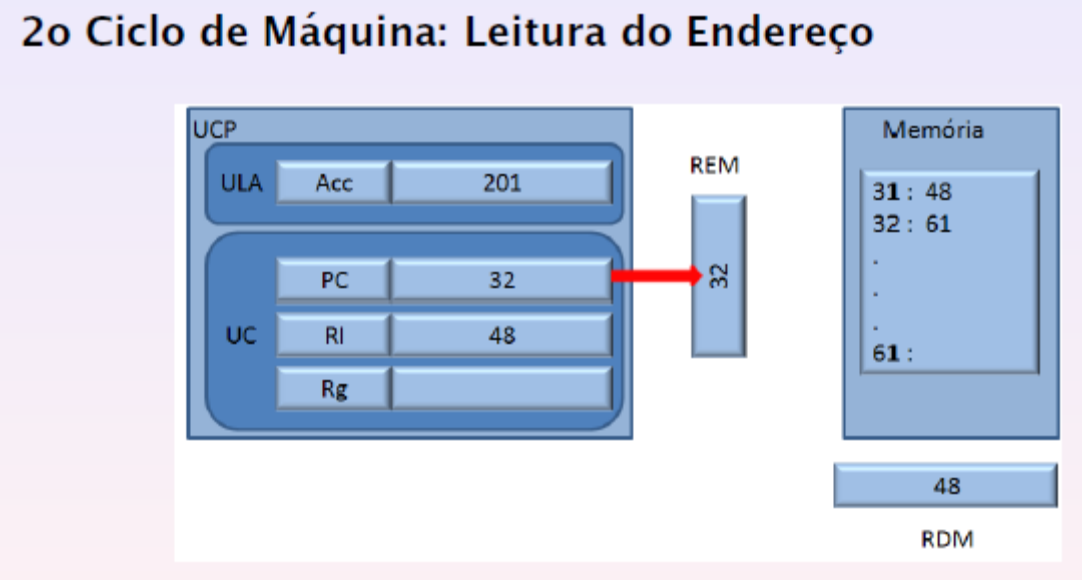
- Com o encerramento do 1o ciclo de máquina, o processador interpreta o código de operação, no exemplo, o código 48 que equivale à operação STA.
- O processador saberá que a instrução STA pede mais um ciclo de máquina para buscar um endereço.

Exemplo 1

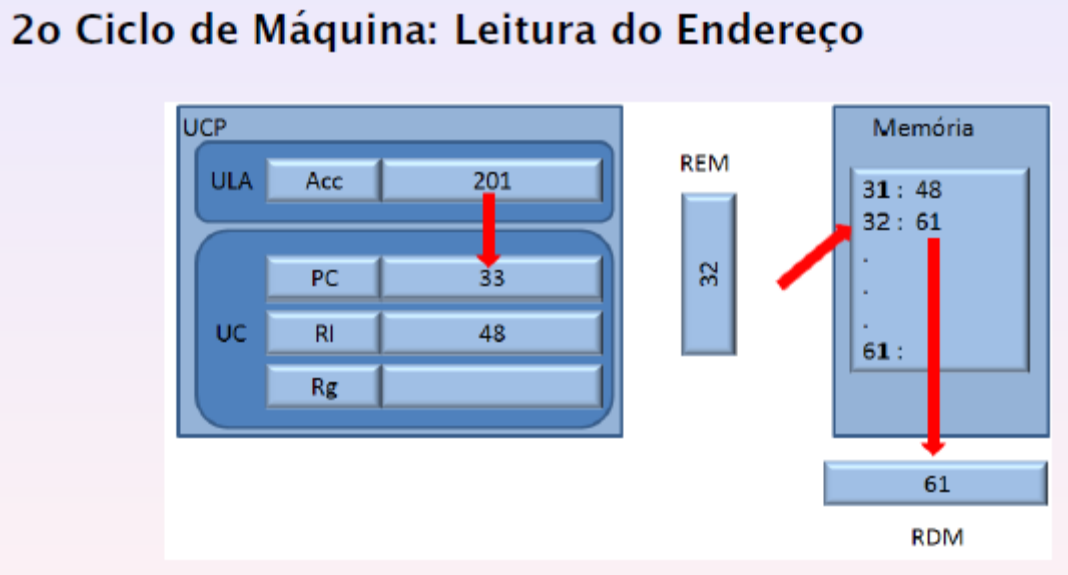
2o Ciclo de Máquina: Leitura do Endereço

CLK	Microoperação
1	REM ← (PC)
2	RDM ← (m) PC ← (PC) + 1
3	Rg ← (RDM)

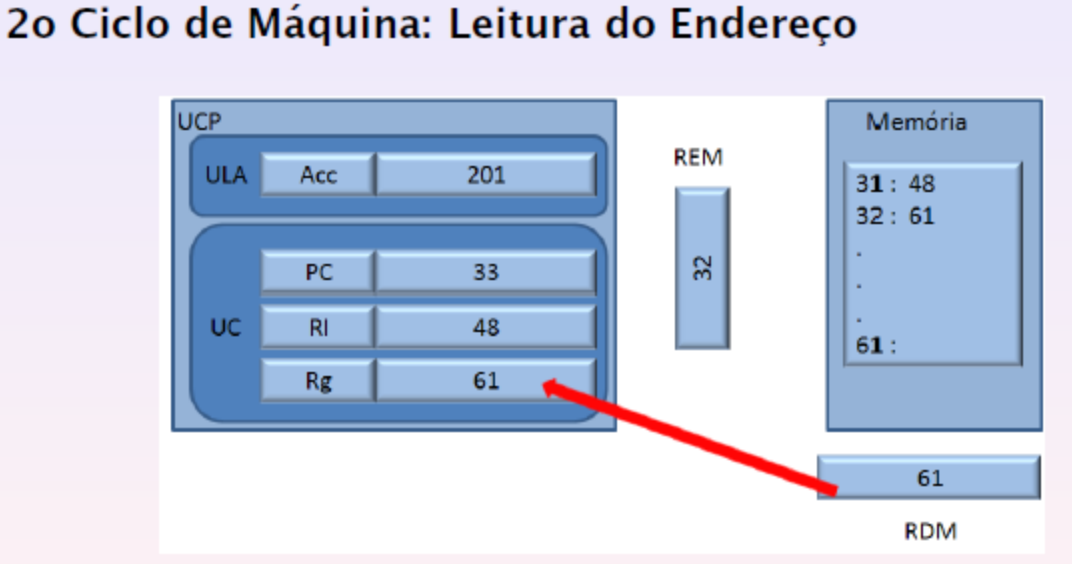
Exemplo 1


2o Ciclo de Máquina: Leitura do Endereço
UCP
ULA
Acc
201
UC
PC
32
RI
48
Rg
REM
32
Memória
31 : 48
32 : 61
.
.
.
61 :
48
RDM

Exemplo 1


2o Ciclo de Máquina: Leitura do Endereço
UCP
ULA
Acc
201
PC
33
UC
RI
48
Rg
REM
32
Memória
31 : 48
32 : 61
61 :
61
RDM

Exemplo 1


2o Ciclo de Máquina: Leitura do Endereço
UCP
ULA
Acc
201
UC
PC
33
RI
48
Rg
61
REM
32
Memória
31 : 48
32 : 61
61 :
61
RDM

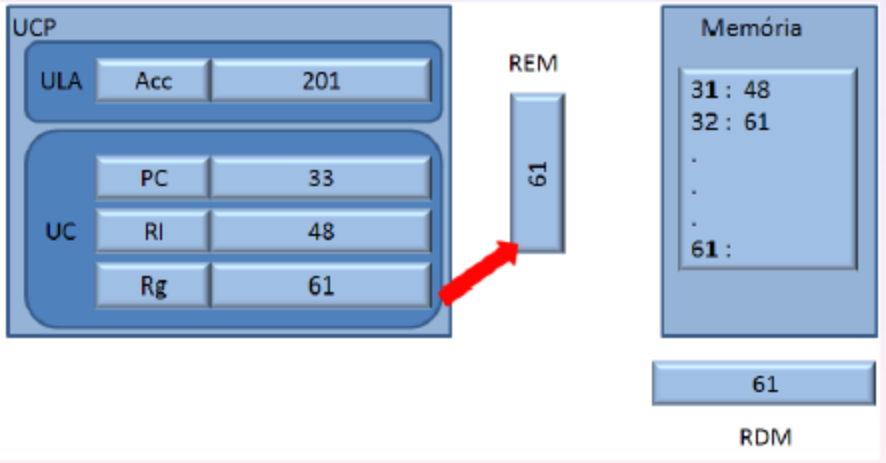
Exemplo 1

3o Ciclo de Máquina: Transferência do conteúdo de Acc para a memória

CLK	Microoperação
1	REM ← (Rg)
2	RDM ← (Acc)
3	(m) ← (RDM)

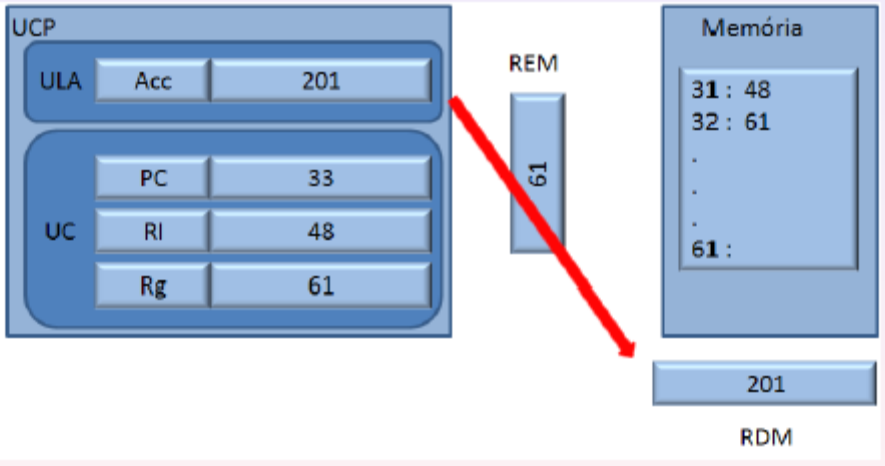
Exemplo 1

3o Ciclo de Máquina: Transferência do conteúdo de Acc para a memória



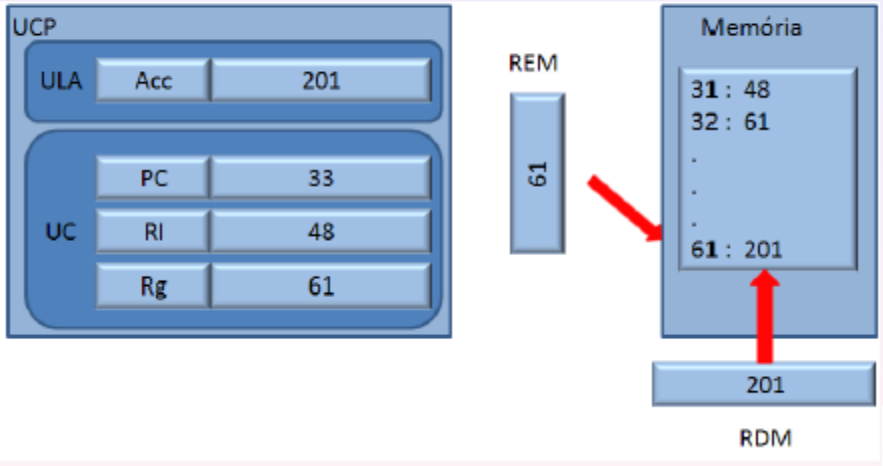
Exemplo 1

3o Ciclo de Máquina: Transferência do conteúdo de Acc para a memória



Exemplo 1

3o Ciclo de Máquina: Transferência do conteúdo de Acc para a memória



Obrigado

Prof. Fabio Kawaoka Takase
fabio.takase@mackenzie.br