CESAR
Características Gerais e Modos de Endereçamento

Prof. Sérgio Luis Cechin

.inf INSTITUTO DE INFORMÁTICA UFRGS

Características Gerais

- Não é código-compatível com o RAMSES nem com o NEANDER
- É um processador de 16 bits
- Incorpora todos os recursos já vistos nos processadores anteriores
  - A forma, no entanto, é diferente
- Tudo o que se pode fazer com o CESAR pode ser feito com o RAMSES ou o NEANDER...
- ...Entretanto, é mais eficiente

Características Específicas (1)

- Largura dos dados e endereços: 16 bits
  - Tamanho da memória: 65.536 bytes
- Representação dos dados em complemento de 2
  - Afeta a forma com a ULA efetua seus cálculos
- Modos de endereçamento: 8
- Processamento de pilha
- Código das Instruções ocupando 1 ou 2 bytes (1 ou 2 endereços de memória)

Características Específicas (2)

- Registradores
  - 8 registradores de 16 bits, chamados R0, R1, ..., R7
  - 1 registrador de estado
- Registradores de uso geral
  - R0, R1, ..., R5
- Registradores de uso específico
  - R6: apontador de pilha (SP - **Stack Pointer**)
  - R7: apontador de programa (PC - **Program Counter**)

Características Específicas (3)

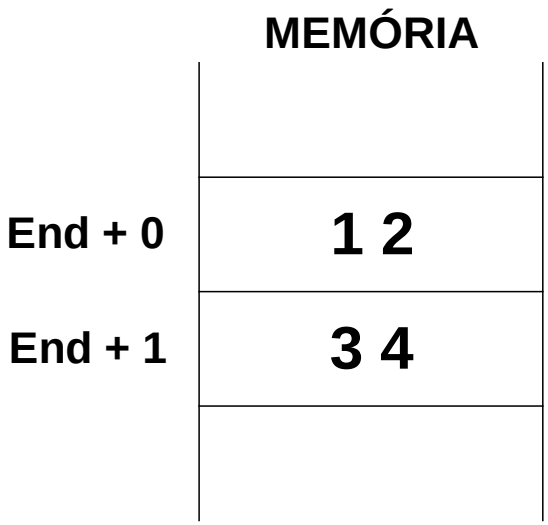
- Registrador de estado
  - Código de condição N (Negativo)
    - Se 1, indica valores negativos (complemento de 2)
  - Código de condição Z (Zero)
    - Se 1, indica valor zero
  - Código de condição C (Carry)
    - Se 1, significado dependente da instrução geradora
  - Código de condição V (Overflow)
    - Se 1, indica estouro de representação (complemento de 2)

Estrutura da Memória

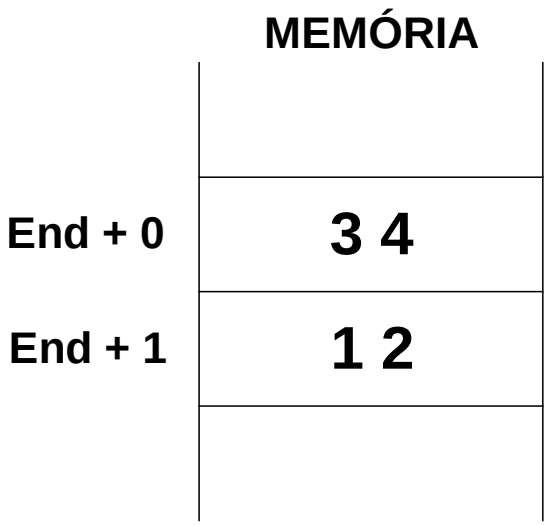
- Organizada em bytes
  - Cada endereço corresponde a 1 byte da memória
  - Porque...
    - ...permite acomodar, de forma eficiente, instruções com 1 byte;
    - ...é utilizada na maioria dos processadores comerciais;
    - ...em geral, dados mais largos são sempre múltiplos de 8 bits.
- Armazenamento de valores com 16 bits
  - Little endian ou Low-high: Intel
  - **Big endian ou High-low**: Motorola, **Cesar**

Armazenamento do valor 1234H em...

Big endian ou High-low



Little endian ou Low-high



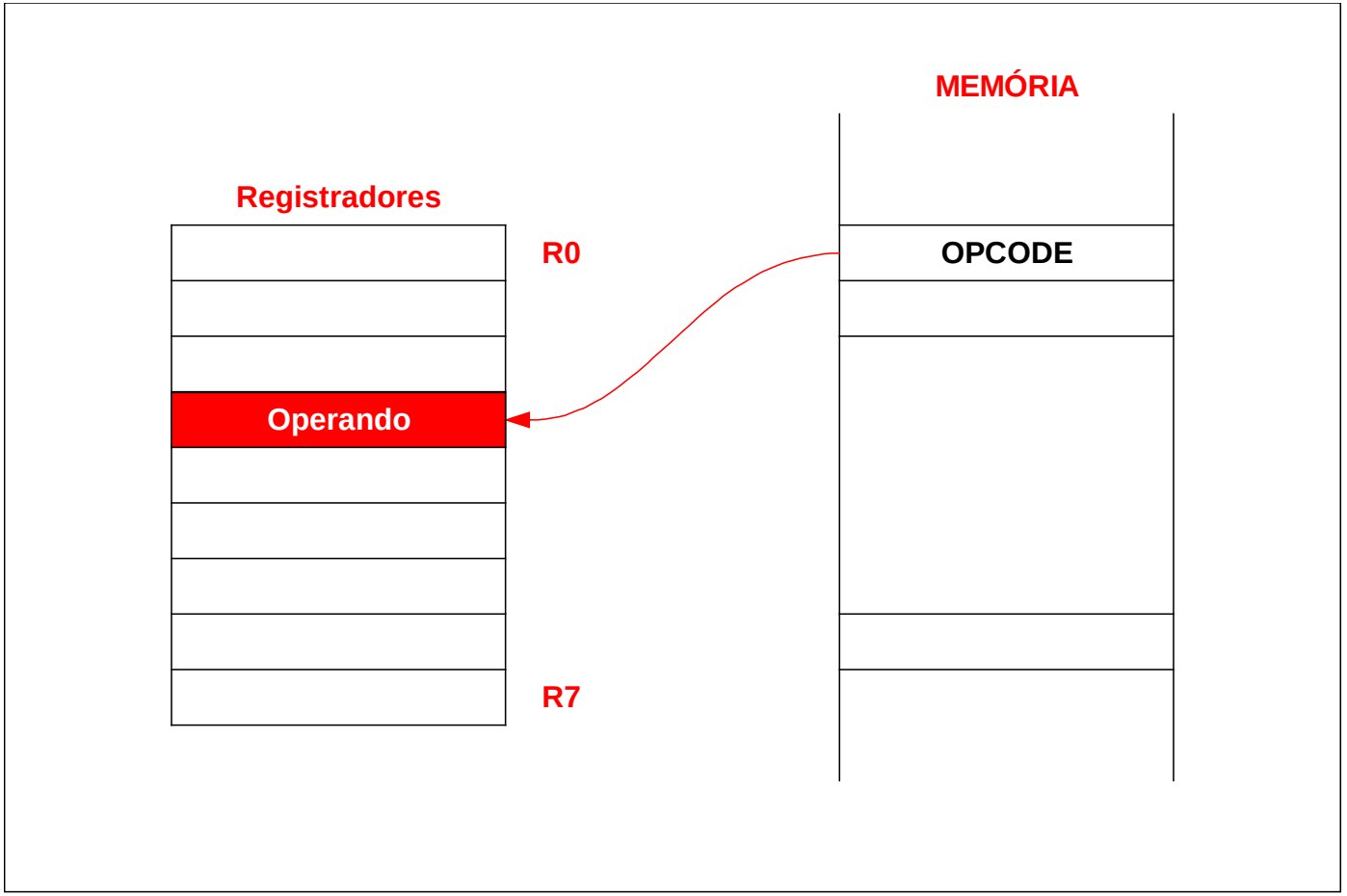
Modos de Endereçamento

- Oito modos de endereçamento
- Grupo 1
  - Registrador
  - Registrador pós-incrementado
  - Registrador pré-decrementado
  - Indexado
- Grupo 2
  - Registrador **Indireto**
  - Registrador pós-incrementado **Indireto**
  - Registrador pré-decrementado **Indireto**
  - Indexado **Indireto**

Registrador (1)

- Código: 000
- Simbólico
  - Ri
- Operação de endereçamento
  - Operando := Ri
- Corresponde ao acesso aos conteúdos dos registradores R0, R1, ..., R7
  - Da mesma forma que o acesso aos registradores A, B e X, no RAMSES

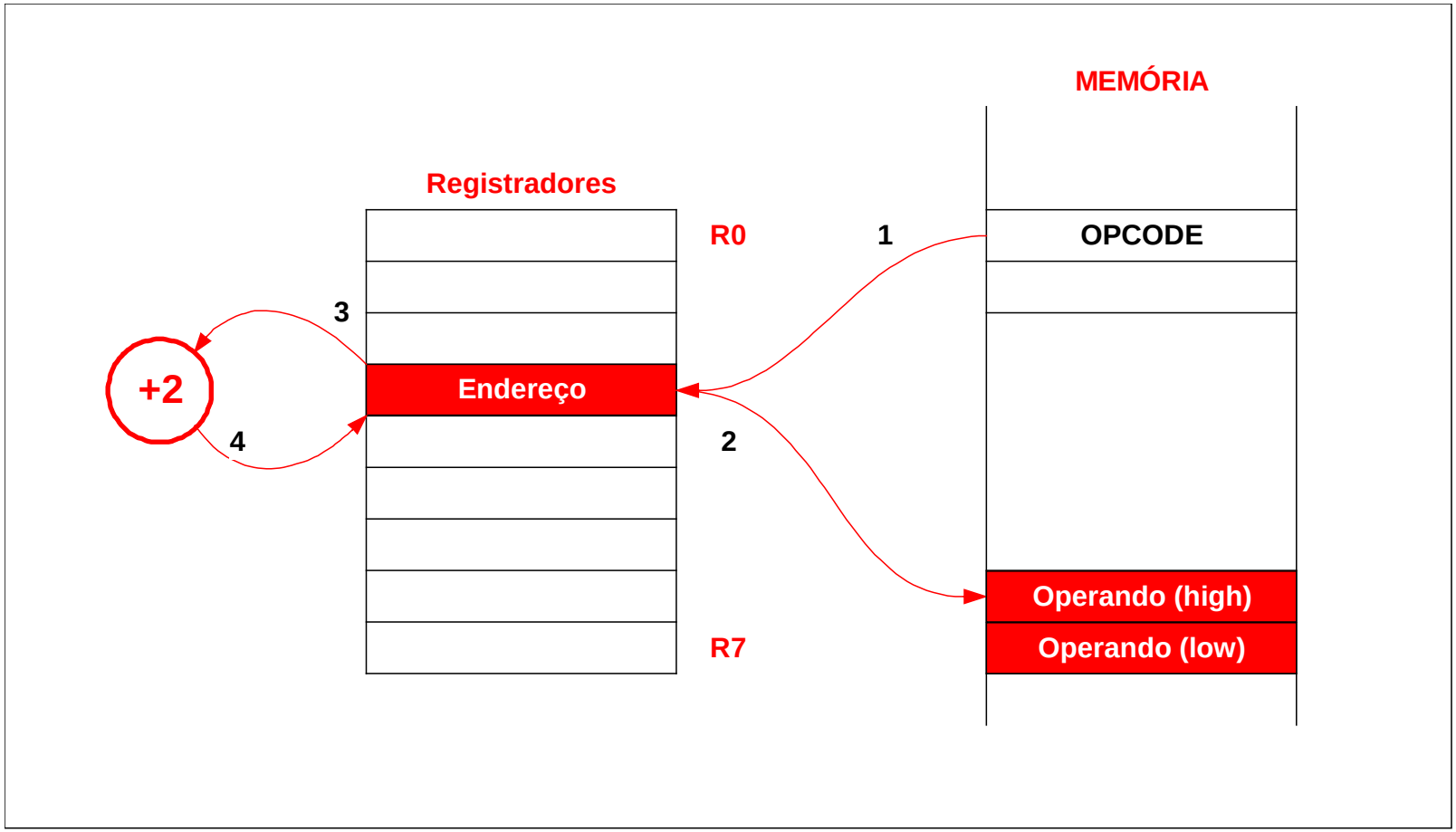
Registrador (2)



Registrador pós-incrementado (1)

- Código: 001
- Simbólico
  - (Ri)+
- Operação de endereçamento
  - Operando := MEM(Ri) e MEM(Ri+1)
  - Ri := Ri+2
- Corresponde ao modo de endereçamento direto do RAMSES.
  - Entretanto, utiliza o conteúdo dos registradores como endereço
  - Auto-incrementa o conteúdo do registrador usado como endereço.

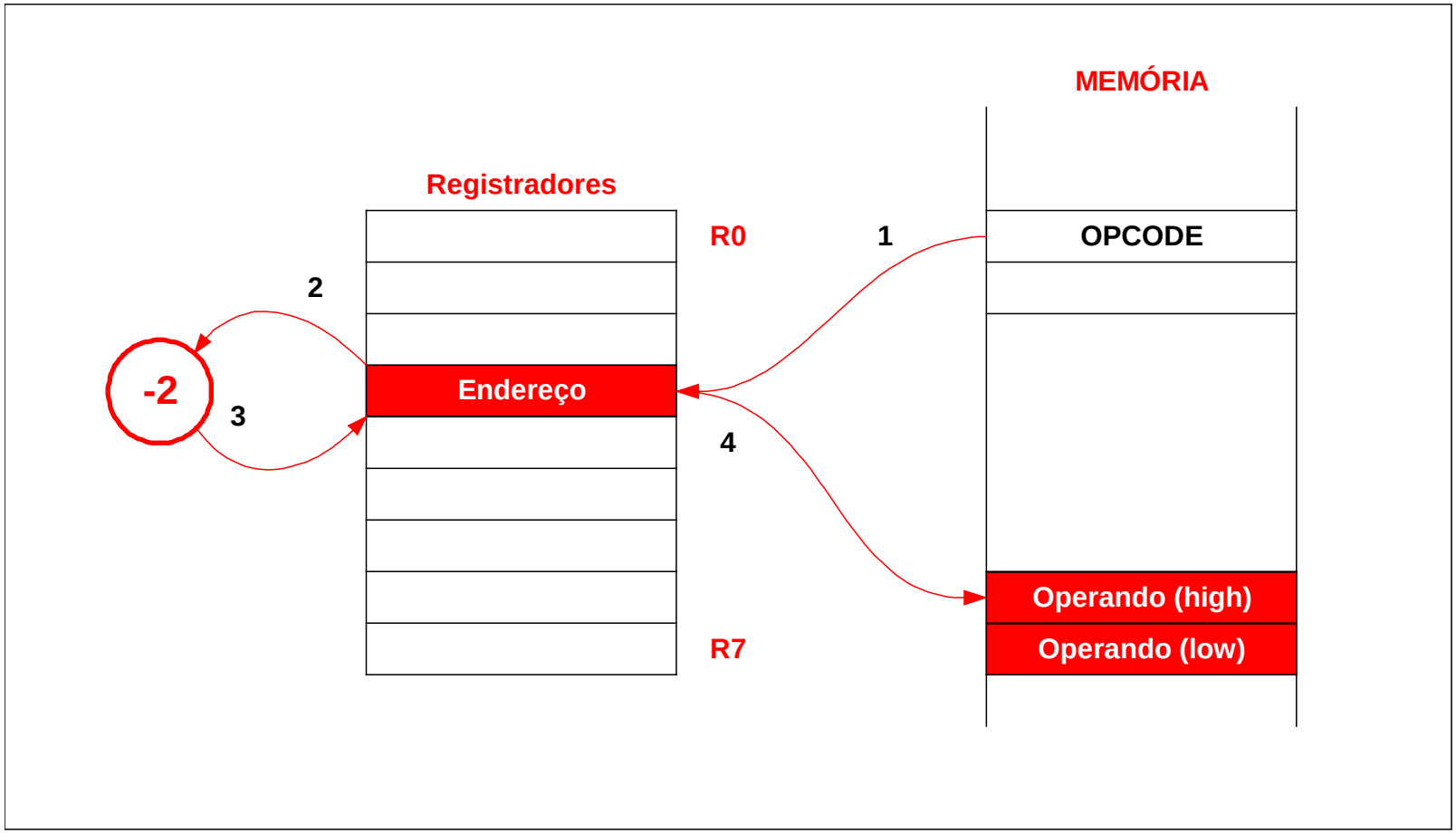
Registrador pós-incrementado (2)



Registrador pré-decrementado (1)

- Código: 010
- Simbólico
  - -(Ri)
- Operação de endereçamento
  - Ri := Ri-2
  - Operando := MEM(Ri) e MEM(Ri+1)
- Corresponde ao modo de endereçamento direto do RAMSES.
  - Entretanto, utiliza o conteúdo dos registradores como endereço
  - **Auto-decrementa** o conteúdo do registrador usado como endereço.

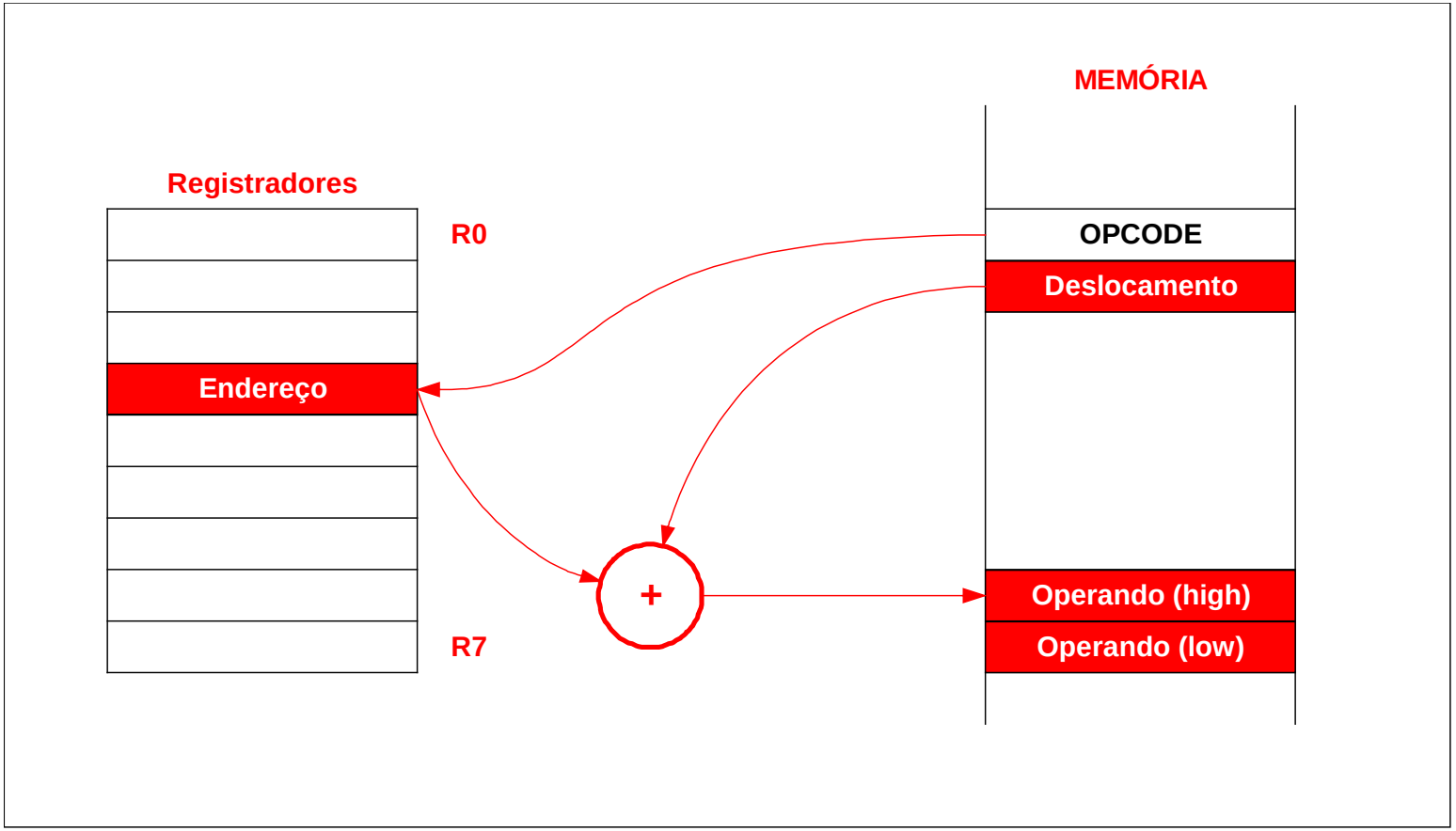
Registrador pré-decrementado (2)



Indexado (1)

- Código: 011
- Simbólico
  - ddd(Ri)
  - **ddd** está representado em complemento de 2
- Operação de endereçamento
  - Operando := MEM(ddd+Ri) e MEM(ddd+Ri+1)
- Corresponde ao modo de endereçamento indexado do RAMSES.
  - Entretanto, utiliza o conteúdo de qualquer dos registradores como índice

Indexado (2)

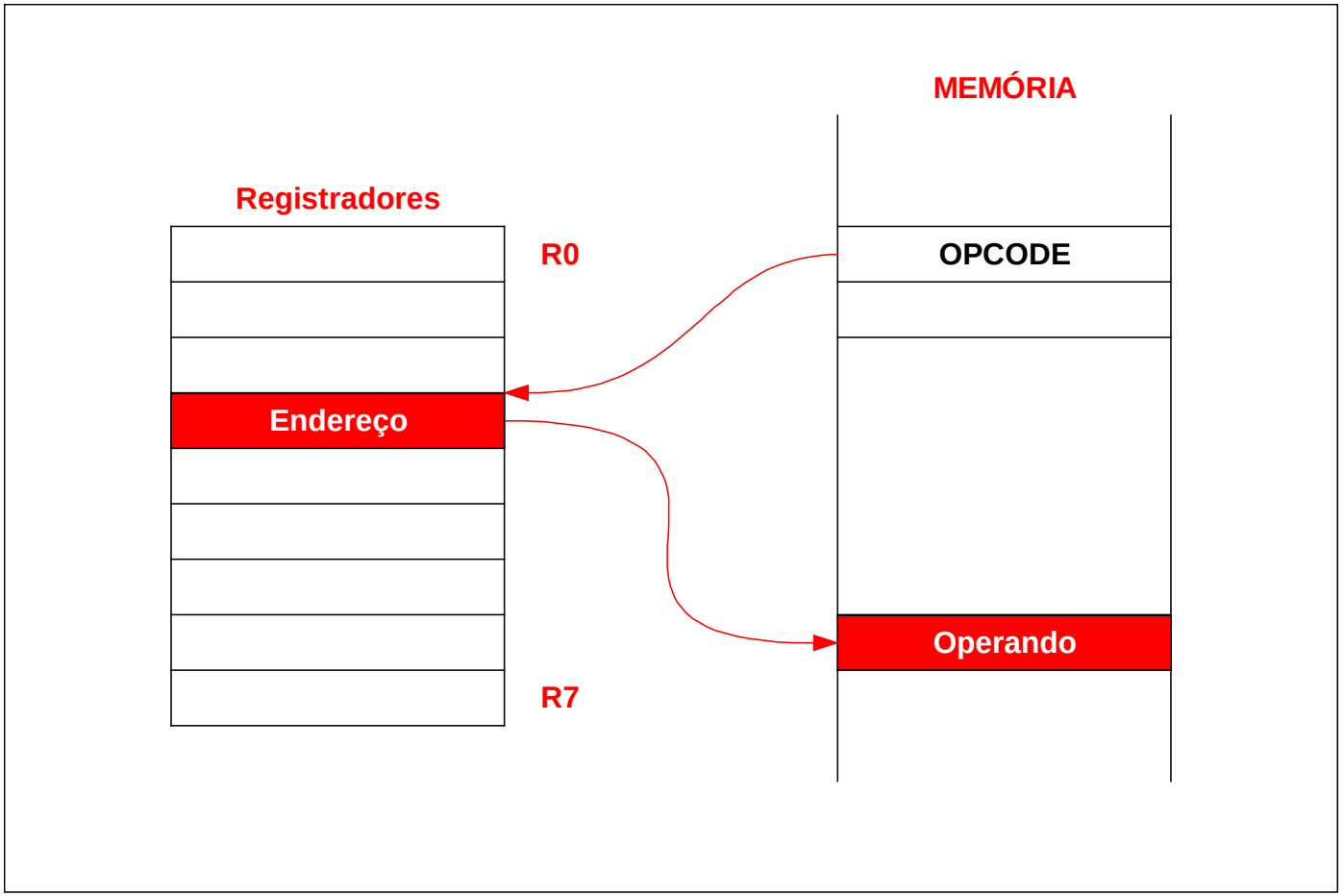


Registrador Indireto

- Código: 100
- Simbólico

  (Ri)
- Operação de endereçamento
  - Operando := MEM(Ri) e MEM(Ri+1)
- Corresponde ao modo de endereçamento direto do NEANDER/RAMSES
  - Entretanto, utiliza o conteúdo dos registradores como endereço

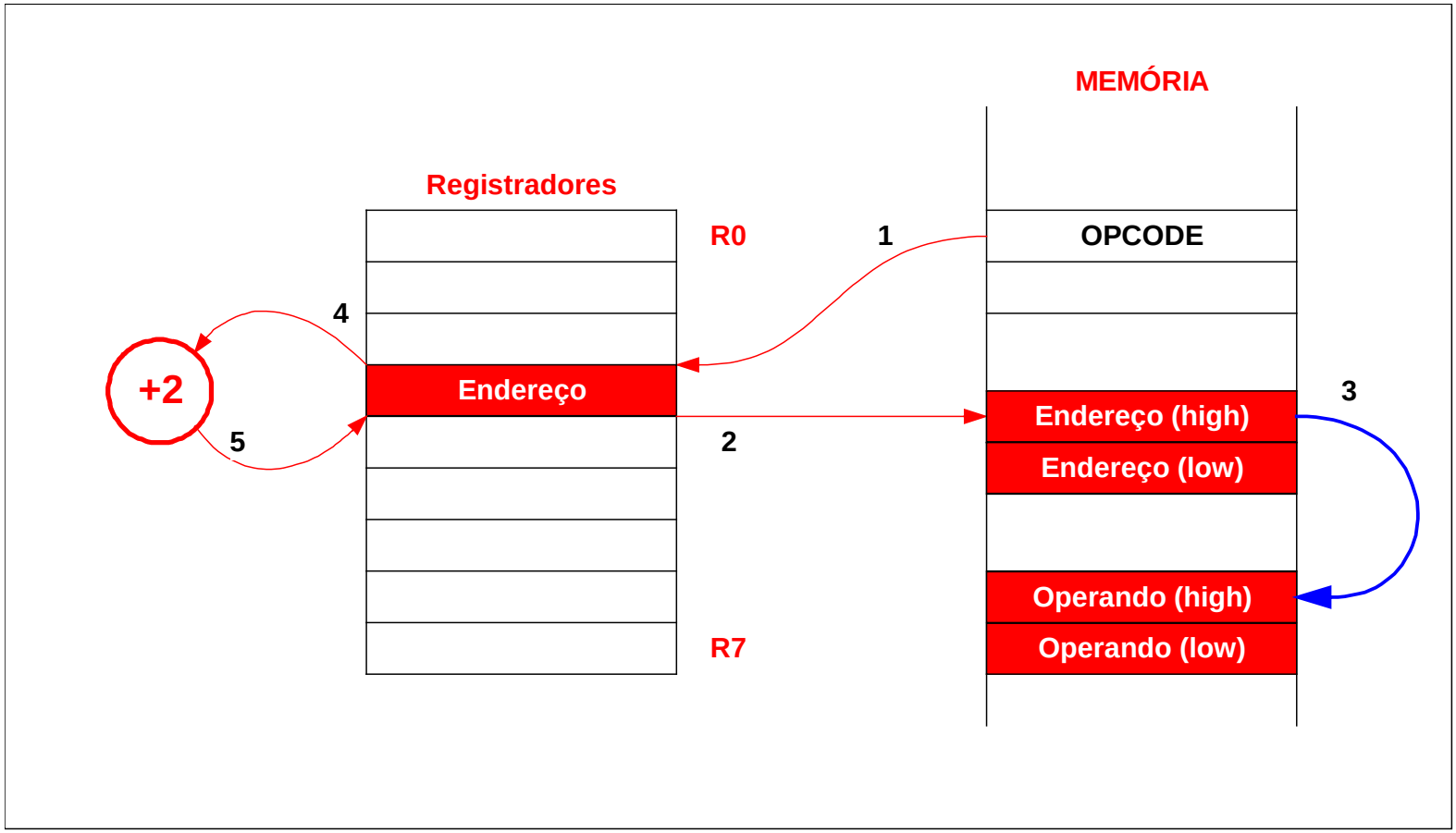
Registrador Indireto



Registrador pós-incrementado Indireto (1)

- Código: 101
- Simbólico
  - ((Ri)+)
- Operação de endereçamento
  - Endereço := MEM(Ri) e MEM(Ri+1)
  - Operando := MEM(Endereço) e MEM(Endereço+1)
  - Ri := Ri+2
- Corresponde ao modo de endereçamento indireto do RAMSES.
  - Entretanto, utiliza o conteúdo dos registradores como referência
  - Auto-incrementa o conteúdo do registrador usado como referência.

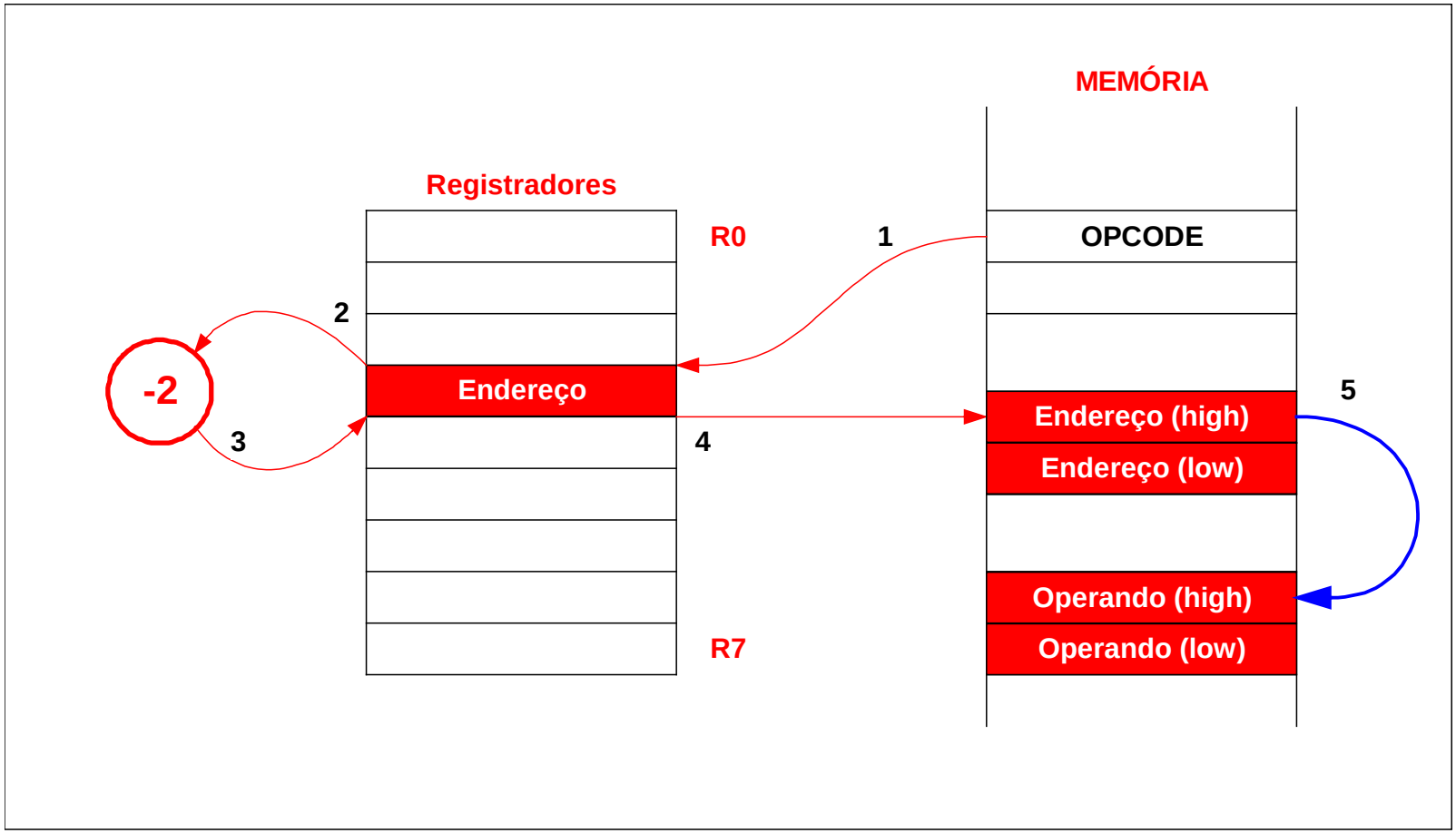
Registrador pós-incrementado Indireto (2)



Registrador pré-decrementado Indireto (1)

- Código: 110
- Simbólico
  - (-(Ri))
- Operação de endereçamento
  - Ri := Ri-2
  - Endereço := MEM(Ri) e MEM(Ri+1)
  - Operando := MEM(Endereço) e MEM(Endereço+1)
- Corresponde ao modo de endereçamento indireto do RAMSES.
  - Entretanto, utiliza o conteúdo dos registradores como endereço
  - **Auto-decrementa** o conteúdo do registrador usado como endereço.

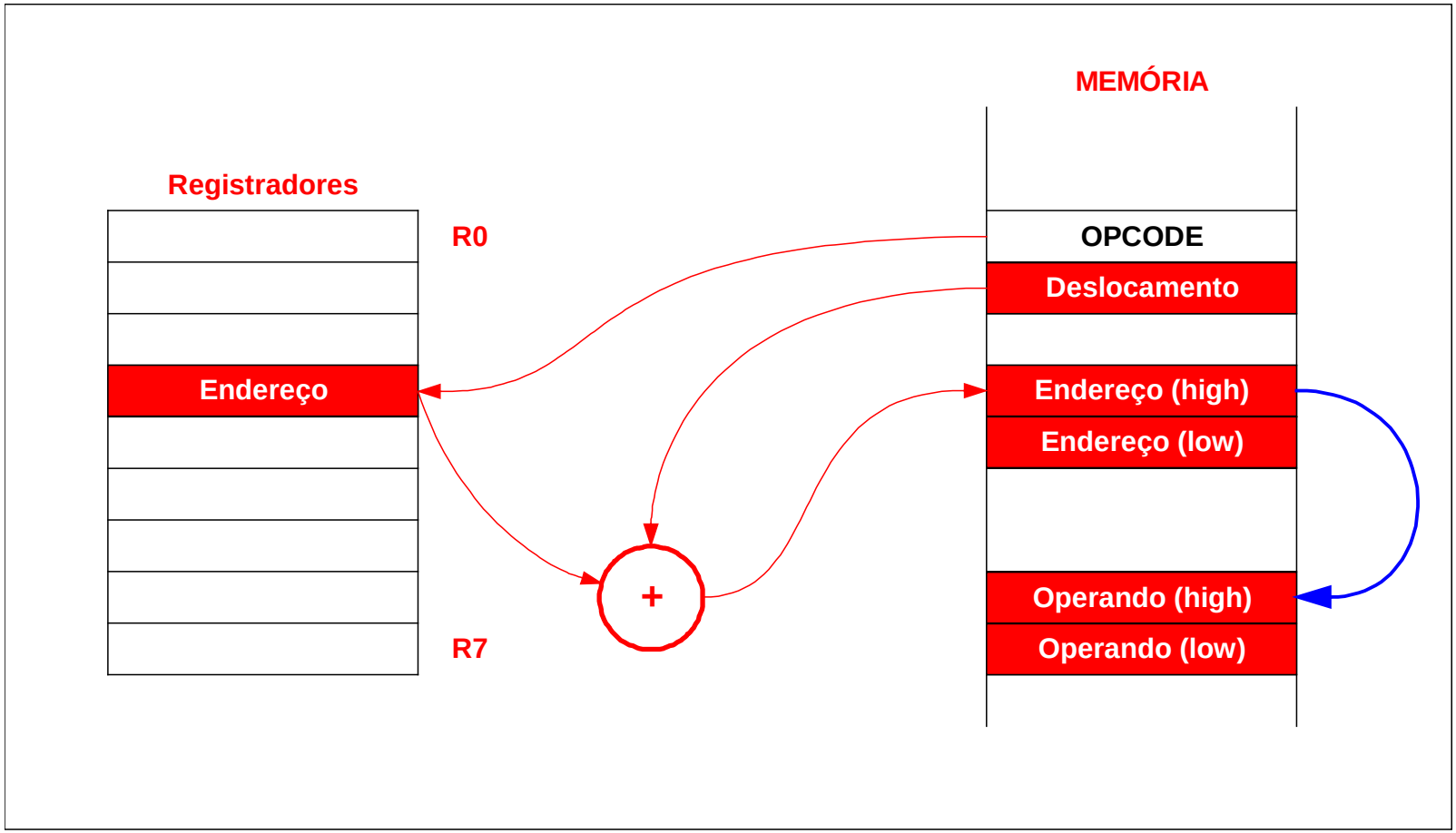
Registrador pré-decrementado Indireto (2)



Indexado Indireto (1)

- Código: 111
- Simbólico
  - (ddd(Ri))
  - **ddd** está representado em complemento de 2
- Operação de endereçamento
  - Endereço := MEM(ddd+Ri) e MEM(ddd+Ri+1)
  - Operando := MEM(Endereço) e MEM(Endereço+1)
- Não tem correspondência com os modos de endereçamento do NEANDER/RAMSES.

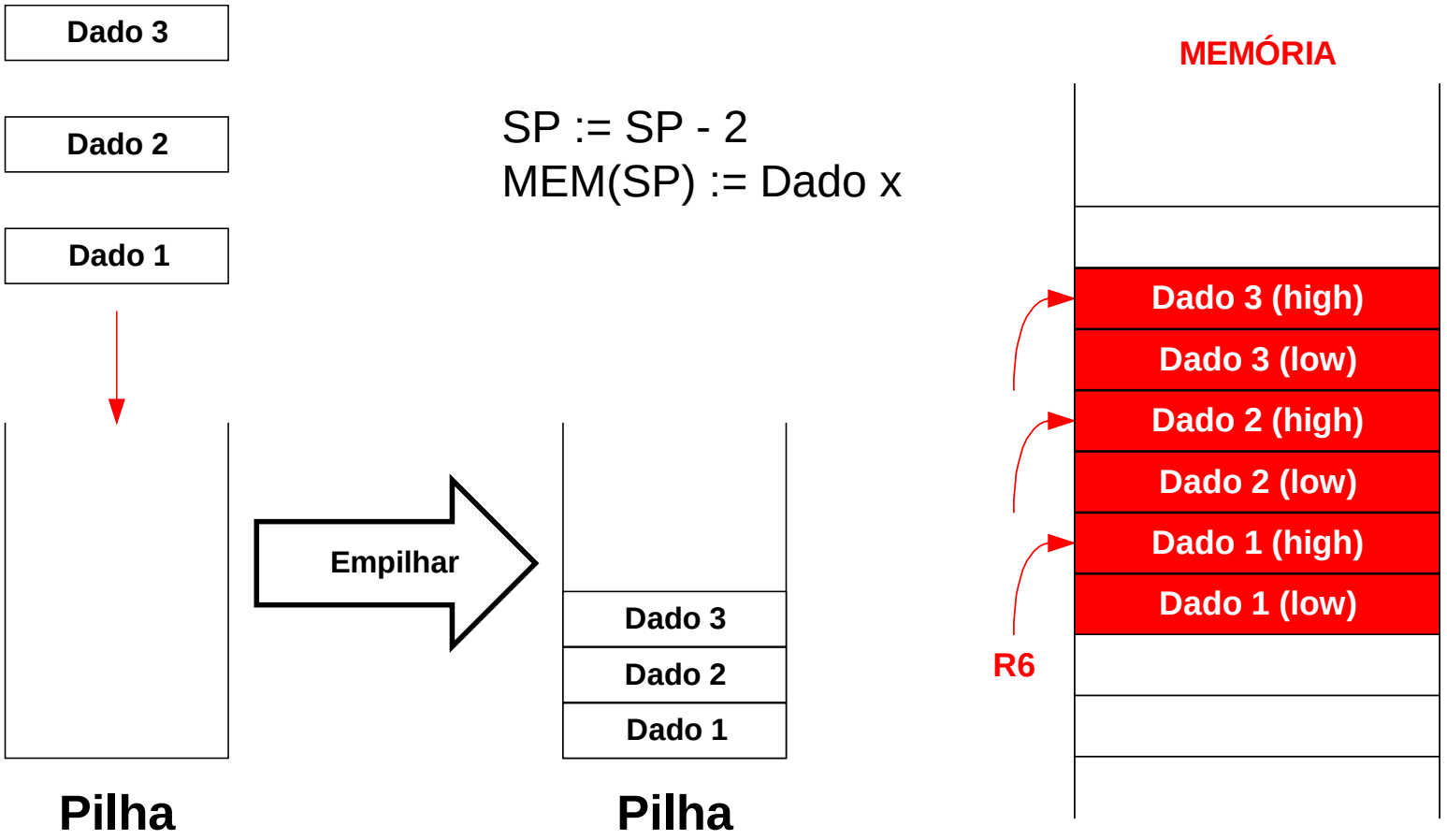
Indexado
Indireto (2)



Pilha do Processador

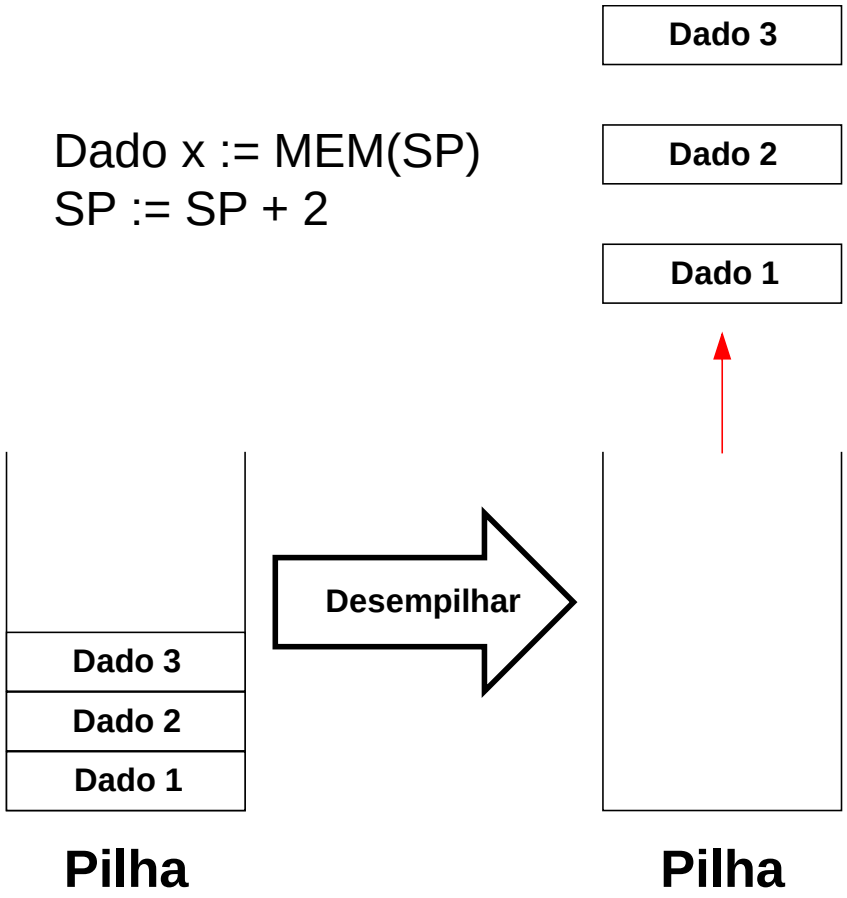
- Utilização do registro R6 como ponteiro de pilha (SP - *Stack Pointer*)
- Na inicialização, R6 recebe 0
- Operações
  - No empilhamento, R6 é decrementado
  - No desempilhamento, é incrementado
  - A pilha cresce com R6 movendo-se em direção aos endereços menores
  - Não existe controle de colisão

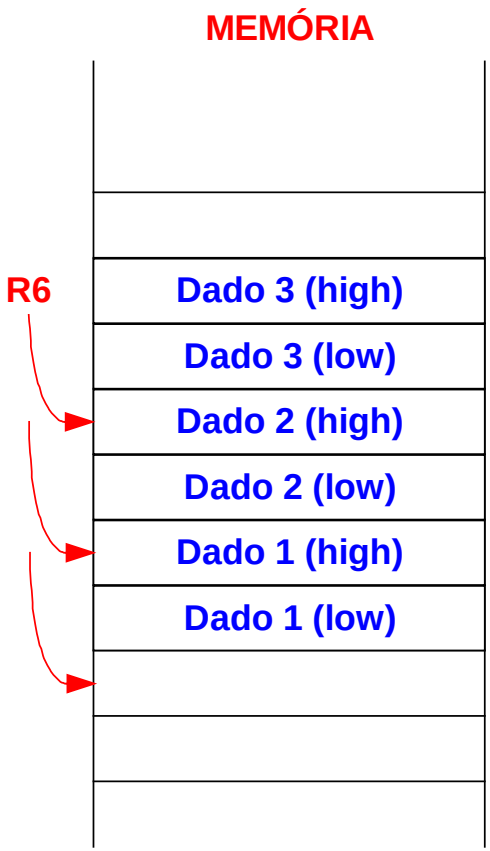
Empilhamento



Desempilhamento

Dado x := MEM(SP)
SP := SP + 2





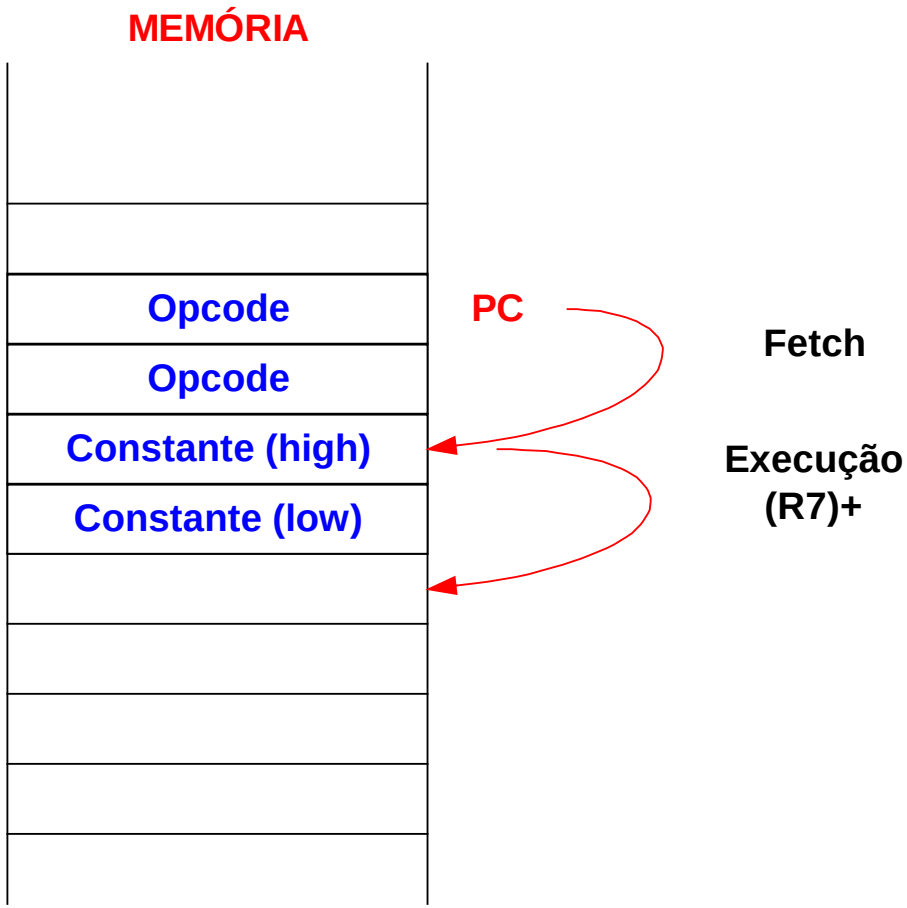
Modos de Endereçamento Usando o R7 (PC)

- O processador permite ao programador o uso do R7 (*Program Counter*) da mesma forma que os outros registradores.
- O programador pode:
  - Ler seu conteúdo
  - Alterar seu conteúdo (o que ocasiona um desvio)
  - Utilizar seu conteúdo para calcular o endereço de um operando
- Criam-se, então, novos modos de endereçamento: absoluto, imediato e relativo.

Modo Imediato – definição

- A palavra que segue o código da instrução é uma constante de 16 bits
- Implementação
  - Através do modo 001 sobre R7
    - Registrador pós-incrementado
    - (R7)+
  - Operação
    - Operando := MEM(R7)
    - R7 := R7 + 2

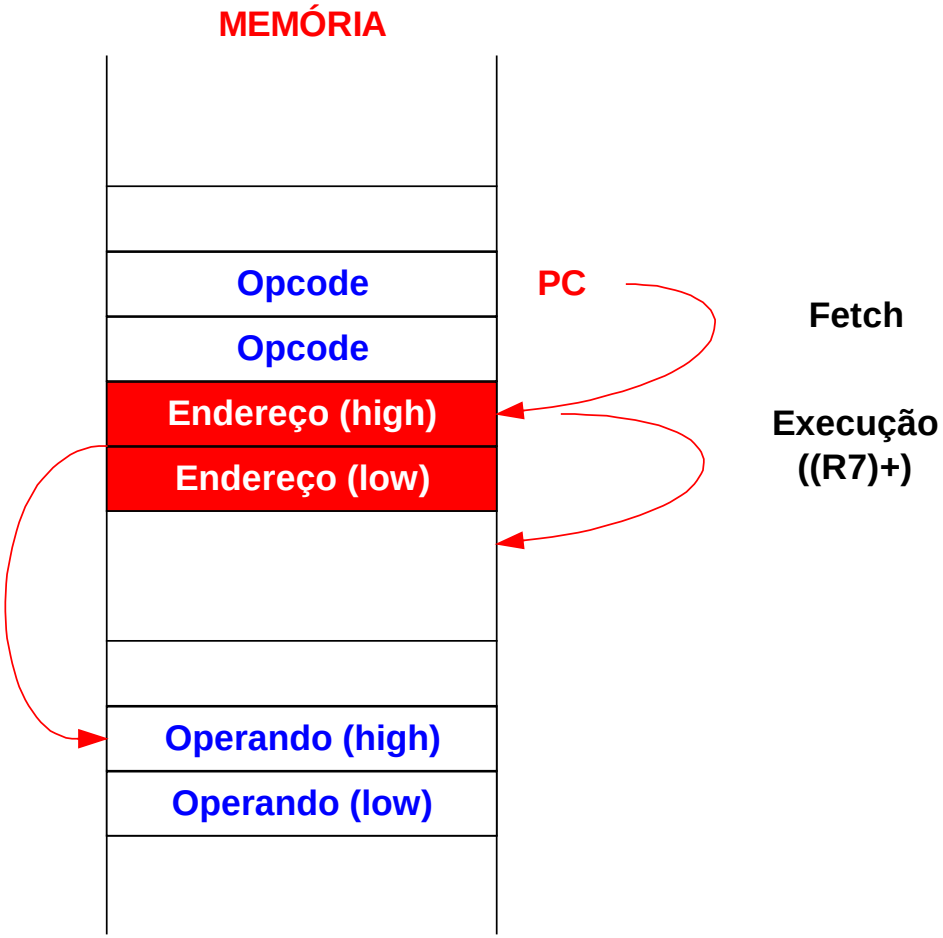
Modo Imediato – operação



Modo Absoluto – definição

- A palavra que segue o código da instrução é o endereço do operando
- Corresponde ao modo direto do NEANDER/RAMSES
- Implementação
  - Através do modo 101 sobre R7
    - Pós-incrementado indireto
    - ((R7)+)
    - Comparando com o modo imediato: apresenta um nível a mais de acesso à memória
  - Operação
    - Operando := MEM(MEM(R7))
    - R7 := R7 + 2

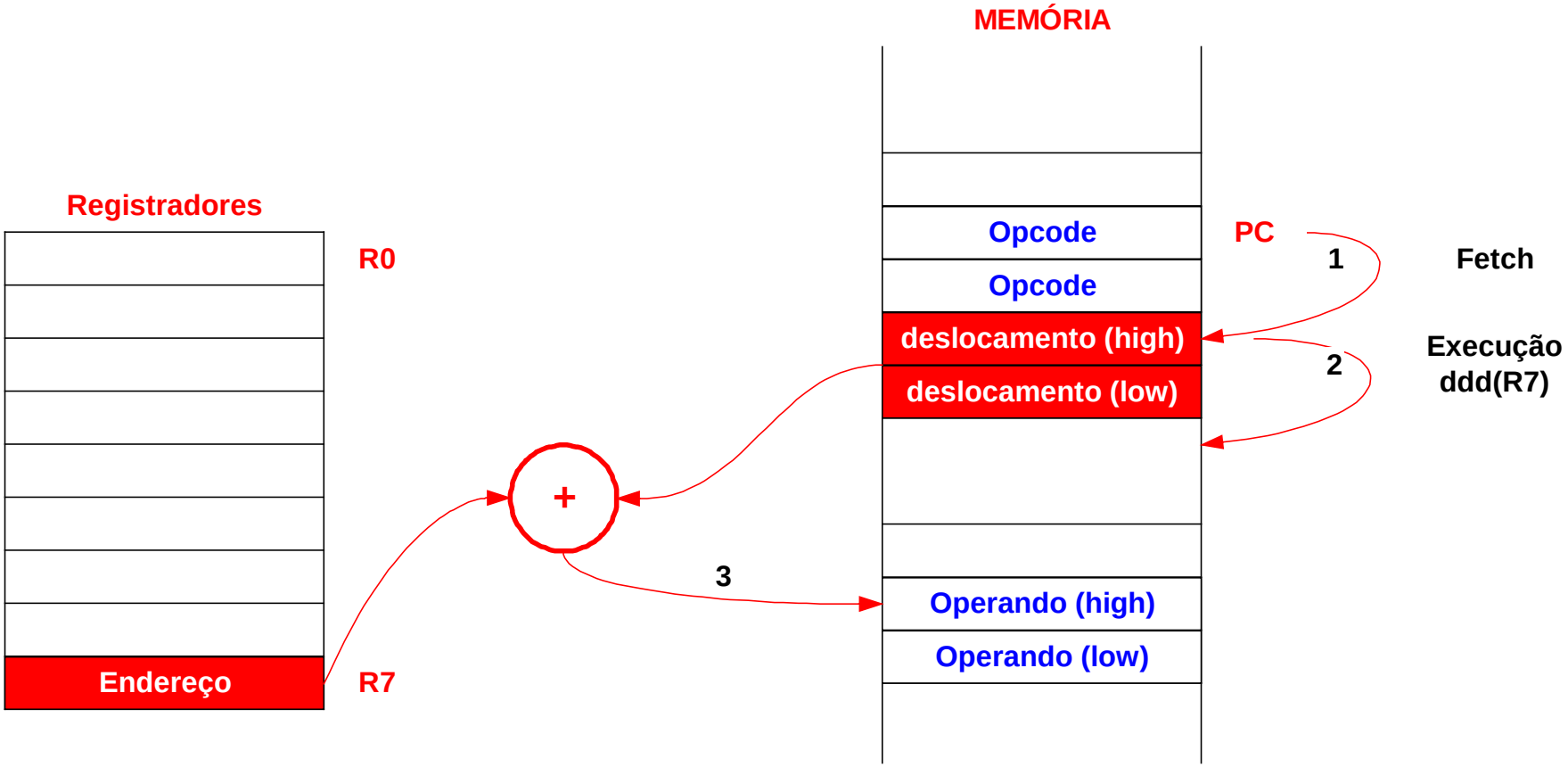
Modo Absoluto – operação



Modo Relativo – definição

- A palavra que segue o código da instrução é o deslocamento da posição do operando, em relação a posição da instrução
- É útil para escrever código relocável
  - Que pode rodar em qualquer endereço
- Implementação
  - Através do modo 011 sobre R7
    - Indexado
    - ddd(R7)
  - Operação
    - Operando := MEM(ddd+R7)

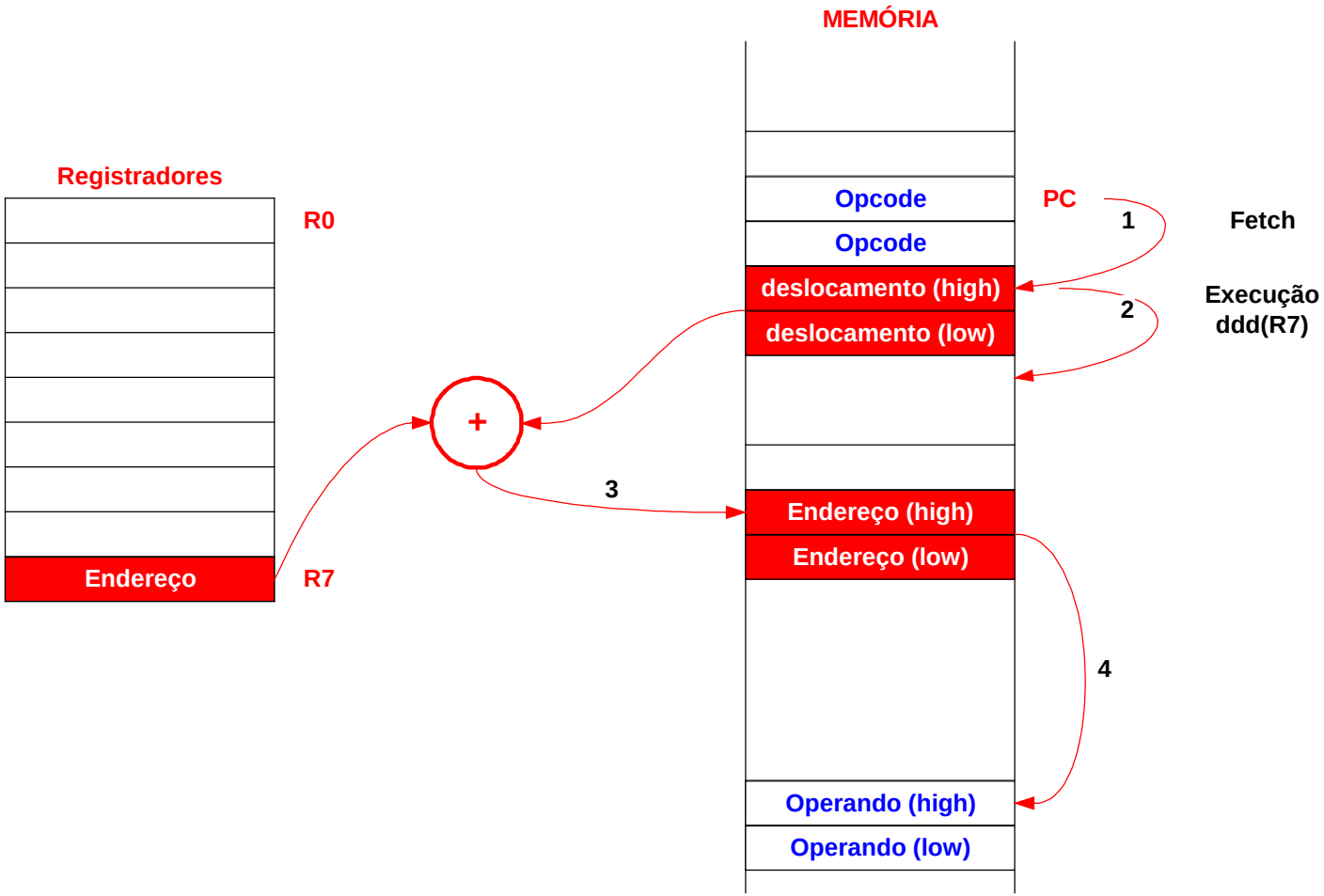
Modo Relativo – operação



Relativo Indireto – definição

- A palavra que segue o código da instrução é o deslocamento da posição do **endereço do** operando, em relação a posição da instrução
- Também é útil para escrever código relocável
  - Que pode rodar em qualquer endereço
- Implementação
  - Através do modo 111 sobre R7
    - Indexado Indireto
    - (ddd(R7))
    - Comparando com o modo relativo: apresenta um nível a mais de acesso à memória
  - Operação
    - Operando := MEM(MEM(ddd+R7))

Relativo Indireto – operação



Resumo

Código	Nome	Simb.	Operação
000	Registrador	Ri	Ri
001	Reg. Pós-incrementado	(Ri)+	MEM(Ri) Ri += 2
010	Reg. Pré-decrementado	-(Ri)	Ri - = 2 MEM(Ri)
011	Indexado	ddd(Ri)	MEM(ddd+Ri)
100	Reg. Indireto	(Ri)	MEM(Ri)
101	Pós-incrementado Indireto	((Ri)+)	MEM(MEM(Ri)) Ri += 2
110	Pré-decrementado Indireto	(-(Ri))	Ri - = 2 MEM(MEM(Ri))
111	Indexado Indireto	(ddd(Ri))	MEM(MEM(ddd+Ri))