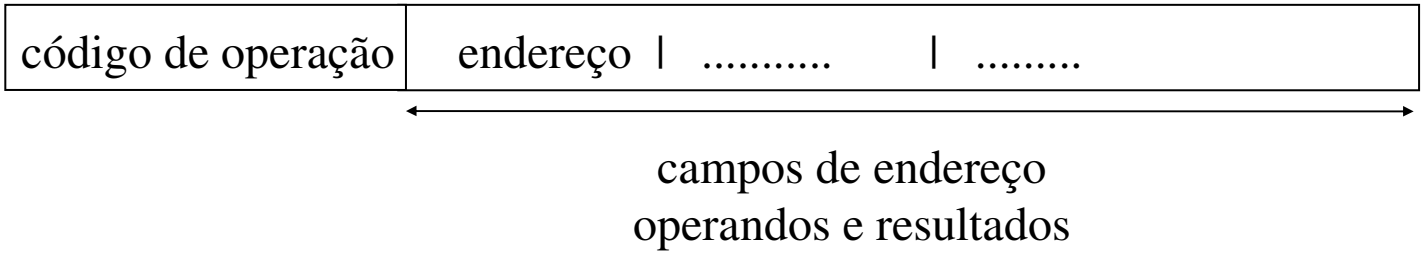
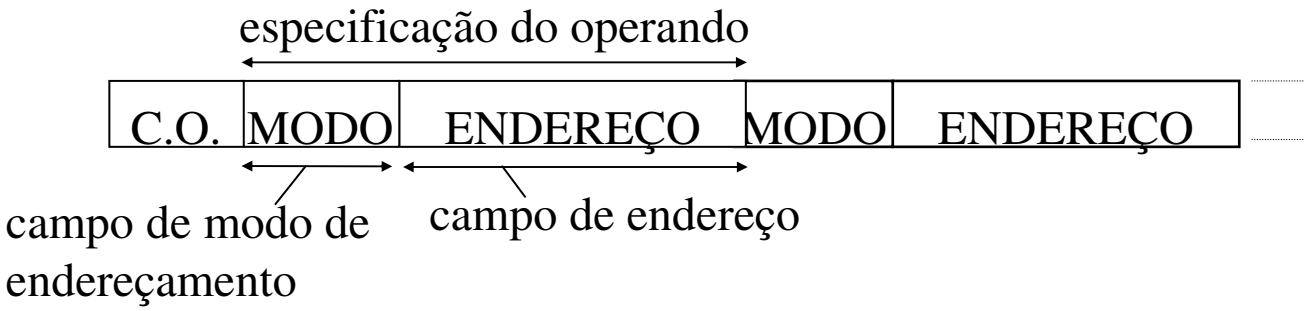
5. Endereçamento
(capítulo 4 do livro texto)

Instruções em linguagem de máquina têm, como vimos, o seguinte formato:



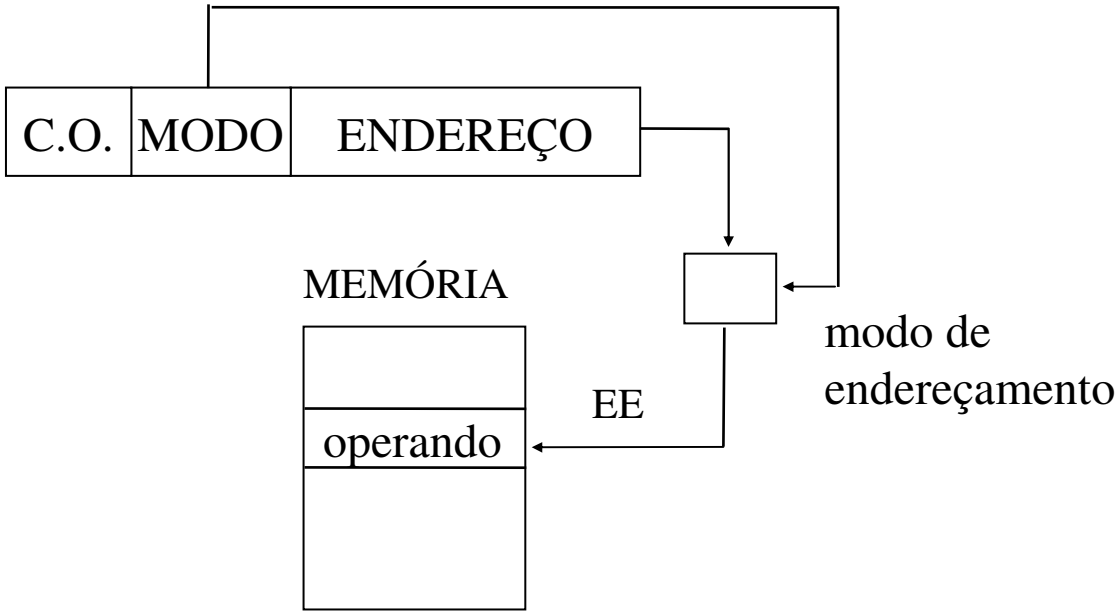
Neste capítulo usaremos a seguinte notação:



5.1 Modos básicos de endereçamento

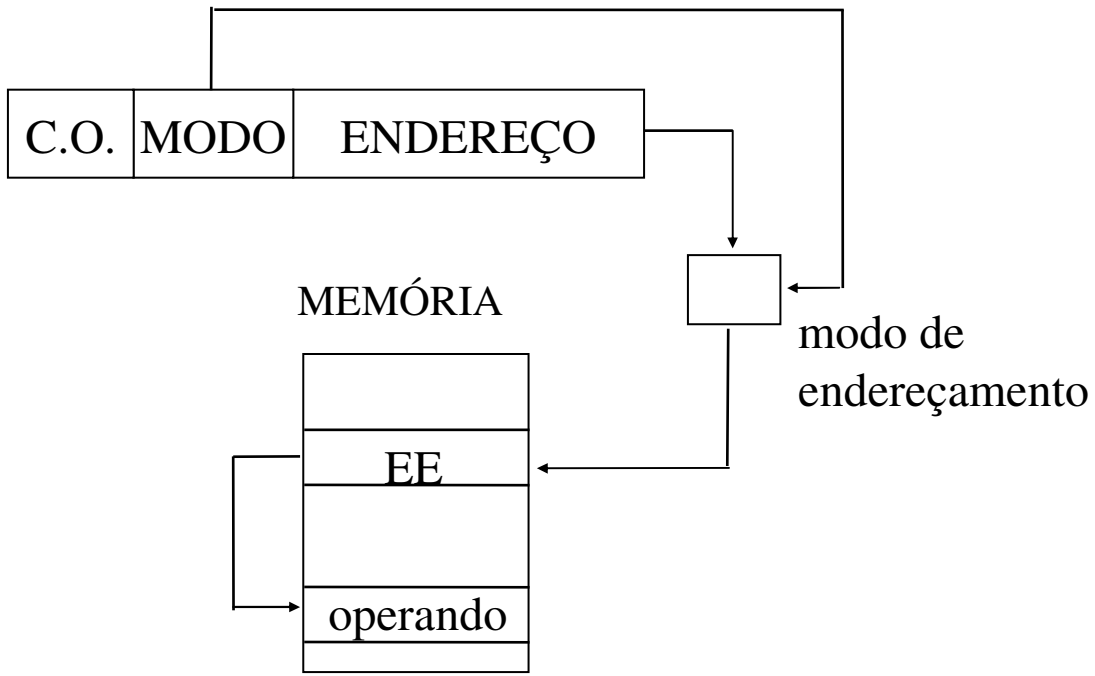
Definição: **endereço efetivo – EE –** é o endereço real ou ainda final de um dado a ser manipulado, de uma instrução a ser executada.

Modo Direto



Modos básicos de endereçamento

Modo Indireto



5.2 Tipos de endereçamento:

A. Inerente ou implícito
B. Por registrador – direto / indireto
C. Absoluto – direto / indireto
D. Imediato
E. Relativo
F. Auto-incremento / auto-decremento
G. Pilha
H. Paginado: por página base / corrente / registrador de página
I. Indexado – direto / indireto
J. Baseado
K. Baseado - Indexado

A. Endereçamento Inerente ou implícito

No modo de endereçamento **Inerente** ou **Implícito** o próprio código de operação identifica o endereço dos dados a serem manipulados.

O endereço do operando não é especificado, é implícito.

Exemplo :

CLR ; Acc ← "0"

NEG ; Acc ← $(\overline{Acc})$

No exemplo: **EE = Acc**.

B.1 Endereçamento por Registrador Direto

O **endereço efetivo** é o próprio registrador referenciado no campo de endereço da instrução; o registrador contém o próprio operando.

$EE=R_i$

Exemplo : para instrução de 2 operandos, ambos no modo registrador direto.

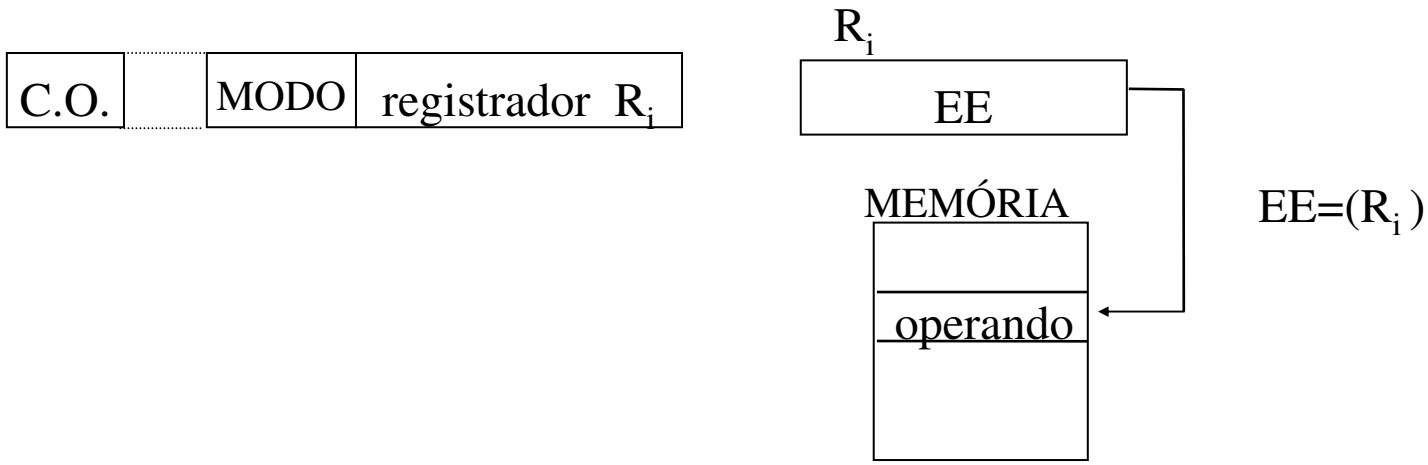
ADD R0,R3 ; R3 ← (R0) + (R3)

Utilização:

É o modo de endereçamento apropriado quando os registradores da UAL estão sendo utilizados como armazenadores temporários de dados.

B.2 Endereçamento por Registrador Indireto

O **endereço efetivo** está contido no registrador referenciado no campo de endereço do operando.



Exemplo : para instrução de 2 operandos, sendo o primeiro no modo registrador indireto e o segundo registrador direto.

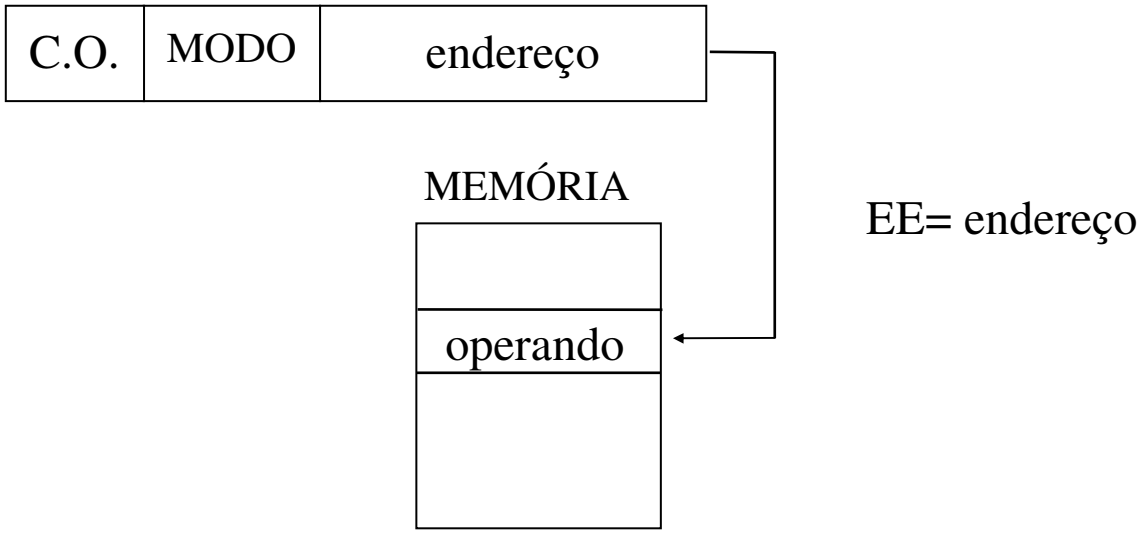
MOVE (R1),R2 ; R2 ← ((R1))

Utilização:

Este modo de endereçamento é utilizado, por exemplo, quando armazenamos o endereço de uma área de dados em um registrador da UAL.

C.1 Endereçamento Absoluto Direto

O **endereço efetivo** é o endereço referenciado no campo de endereço do operando.



Exemplo: para instrução de 2 operandos, sendo o 1° operando no modo absoluto direto e o 2º no modo registrador.

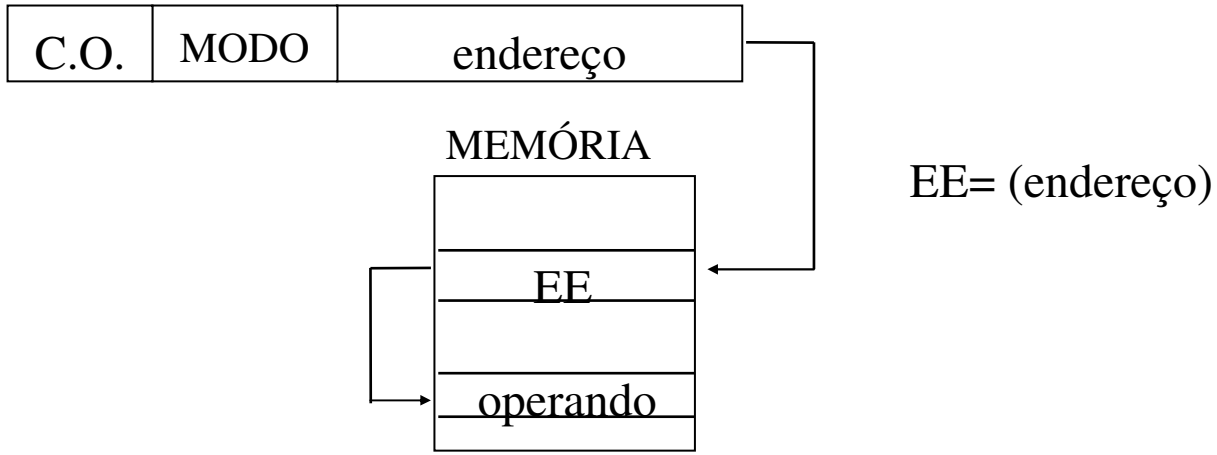
MOVE end,R1 ; R1 ← (end)

Utilização:

É utilizado para acessar o dado diretamente na memória.

C.2 Endereçamento Absoluto Indireto

O **endereço efetivo** está contido na posição de memória cujo endereço é referenciado no campo do operando.



Exemplo: para instrução de 2 operandos, sendo o 1° operando no modo absoluto indireto e o 2° no modo registrador.

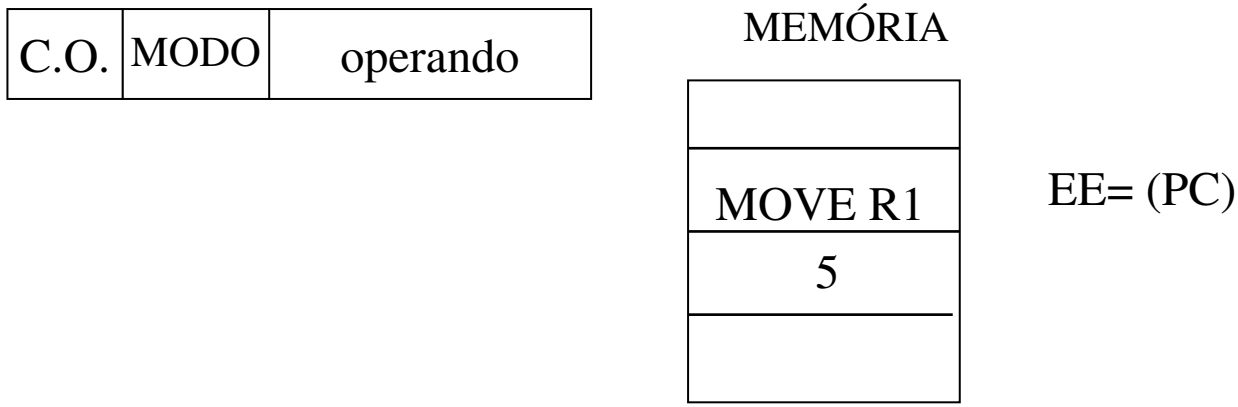
MOVE (end),R1 ; R1 ← ((end))

Utilização: o endereço do operando é acessado via um armazenador (no caso uma posição de memória); esta posição de memória que contém o endereço pode ser operada, incrementada ou decrementada, permitindo o acesso aleatório a uma área de dados.

D. Endereçamento Imediato

Neste tipo de endereçamento, o operando, uma constante ou literal, é especificado diretamente no campo de endereço da instrução. Assim o operando faz parte da instrução.

O **endereço efetivo** é o conteúdo do próprio registrador contador de programas - PC.



Exemplo: para instrução de 2 operandos, sendo o 1° operando no modo imediato e o 2º no modo registrador.

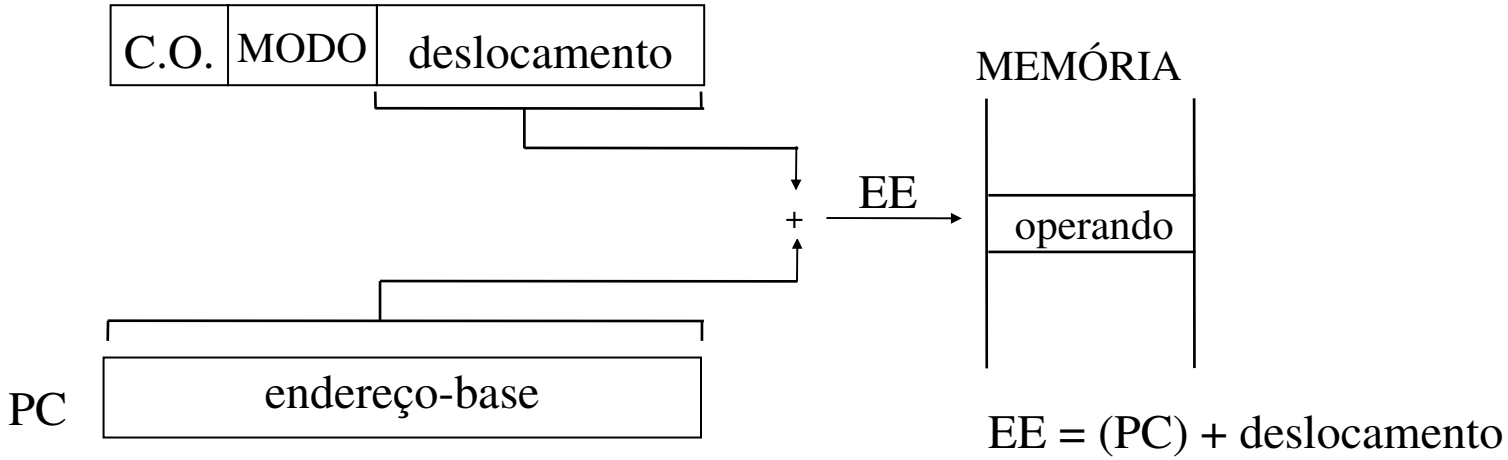
MOVE #5,R1 ; R1 ← 5

Utilização:

O endereçamento imediato é utilizado quando a operação não envolve uma variável e sim uma constante.

E. Endereçamento Relativo

O **endereço efetivo** é calculado pela adição do conteúdo do registrador contador de programa (PC) com um deslocamento contido no campo de endereço da instrução.



Exemplo: para instrução de 2 operandos, sendo o 1º operando no modo relativo e o 2º no modo registrador.

MOVE +8(PC),R1 ; R1 ← ((PC)+8) - é importante notar que o conteúdo inicial do PC é o endereço seguinte à instrução corrente

Utilização:

O endereçamento relativo é muito útil para especificar endereços próximos à posição corrente. É usado para ter-se código "relocável".

F. End. Auto-incremento / decremento

O **endereço efetivo** é alterado durante o ciclo de acesso. A forma como este conteúdo é alterado define o modo de endereçamento incr. ou decr.

É uma forma variante do endereçamento indireto por registrador, sendo muito útil para a manipulação de dados acessados sequencialmente.

Autoincremento: após calculado o **endereço efetivo**, o conteúdo do registrador é incrementado.

Exemplo : para instrução de 2 operandos, sendo o 1º operando no modo registrador e o 2º no modo autoincremento.

MOVE R1, (R2)+ $\equiv$ (R2) $\leftarrow$ (R1) EE = (R2)
R2 $\leftarrow$ (R2) + 1

Auto-incremento / decremento - cont.

Autodecremento: antes de se determinar o endereço efetivo, o conteúdo do registrador é decrementado.

Exemplo : para instrução de 2 operandos, sendo o 1º operando no modo registrador e o 2º no modo autodecremento.

MOVE R1, -(R2) $\equiv$ R2 $\leftarrow$ (R2) - 1 EE = (R2) - 1

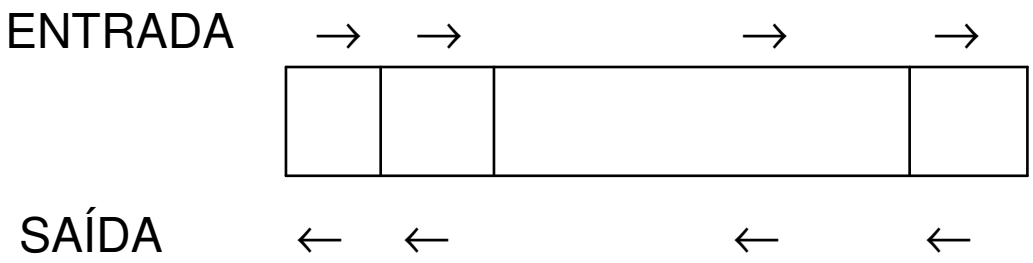
(R2) $\leftarrow$ (R1)

Utilização:

Este modo de endereçamento é utilizado para o acesso sequencial aos elementos de uma área de dados.

G. Endereçamento por Pilha

Pilha ("stack") já foi definida – Capítulo 1 - como uma estrutura de dados onde a entrada e a saída de dados dá-se no mesmo extremo da estrutura, conhecida como o *Topo da Pilha*.
Assim, a estrutura é tratada na forma **Last-In First-Out** (LIFO).



É geralmente implementada alocando-se um conjunto de posições contíguas na memória.

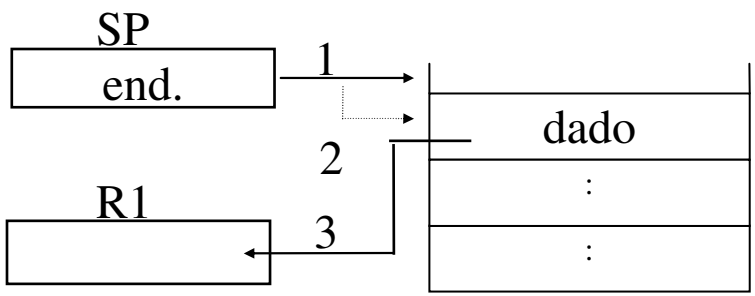
A **inserção** (operação **PUSH**) ou **retirada** (operação **POP**) de um dado a partir do topo da pilha é feita através do registrador **Apontador de Pilha** ("**stack pointer**" - **SP**) que contém o endereço do topo da pilha.

Pilha – cont.

Obs.: em nossa definição o Topo da Pilha é a primeira posição livre da estrutura

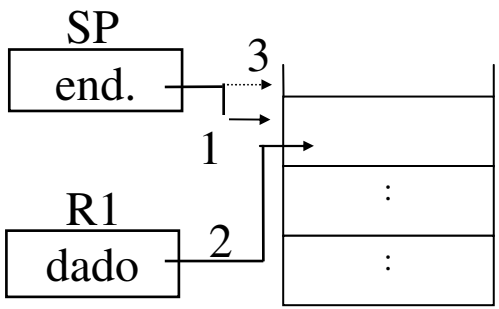
Exemplo:

POP R1 ≡ SP ← (SP) - 1
R1 ← ((SP))



EE = (SP) - 1

PUSH R1 ≡ (SP) ← (R1)
SP ← (SP) + 1



EE = (SP)

A seta tracejada indica o endereço apontado por SP após a execução da operação.

Utilização: Este tipo de endereçamento é muito utilizado para salvar o conteúdo de registradores e posições de memória que posteriormente serão recarregados em suas posições originais. Exemplo típico: subrotinas.

H. Endereçamento Paginado

Neste tipo de endereçamento, o espaço de endereços está logicamente dividido em páginas (blocos de memória) de tamanho fixo.

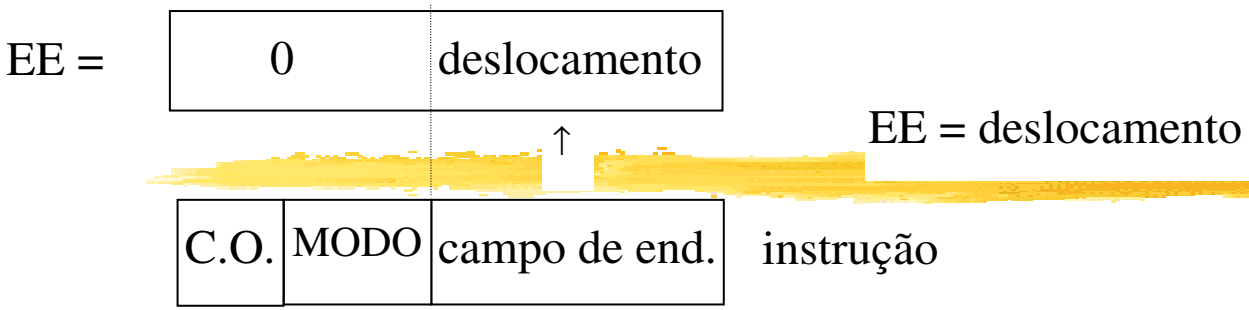
O campo de endereço da instrução conterá a informação da posição (deslocamento) do operando ou instrução dentro de uma página.

O **endereço efetivo** será a concatenação do número da página e do valor do campo de endereço.

Esta técnica pode aliviar o problema de endereçamento de uma memória extensa, mas com palavra curta (poucos bits para muitos endereços).

A referência à página pode ser por **Página Base**, por **Página Corrente** ou por **Registrador de Página**.

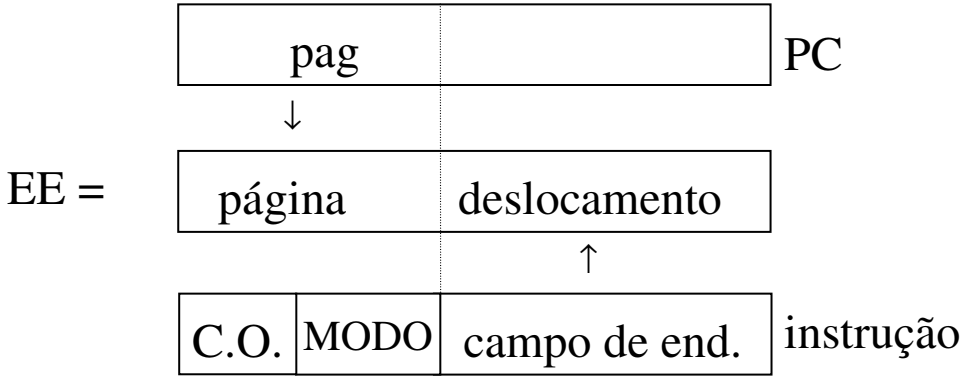
- **Endereçamento Paginado por Página Base**



EE = deslocamento

Neste tipo de endereçamento, a página referenciada é sempre a página zero, definindo o acesso ao primeiro bloco de memória.

- **Endereçamento Paginado por Página Corrente**



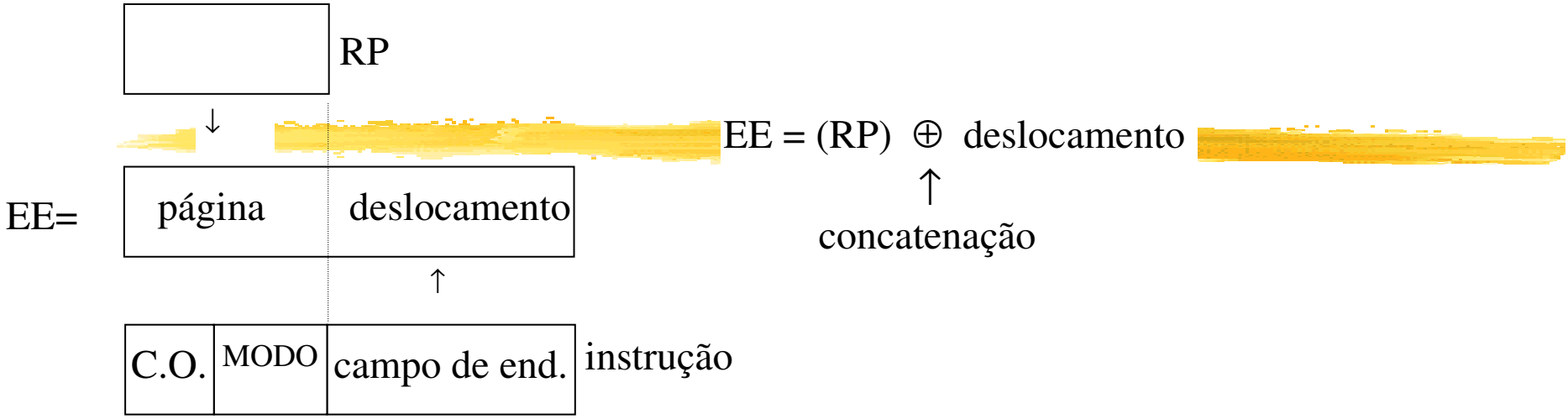
EE = pag ⊕ deslocamento

↑

concatenação

A informação sobre a página está no registrador PC, assim o operando deverá estar na mesma página da instrução que está sendo executada.

- **Endereçamento Paginado por Registrador de Página**



Neste tipo de endereçamento, a página referenciada é definida pelo registrador de página – RP.

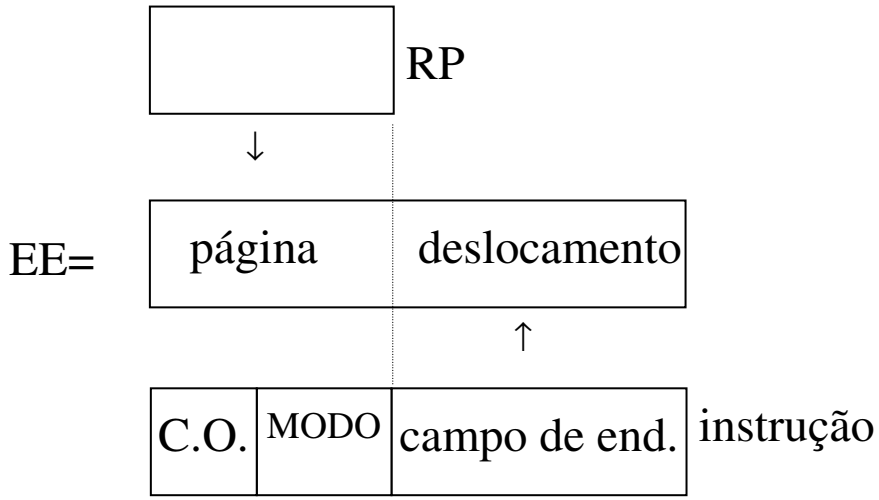
Utilização : o endereçamento paginado é pouco utilizado pelo programador de aplicações; porém, é muito utilizado em Software de Sistema como em Sistemas Operacionais Multiusuários com Gerenciamento de Memória Virtual.

Neste tipo de Sistema, a memória é particionada em páginas com um número fixo de palavras; o programa compilado de cada usuário é também dividido em páginas com o mesmo número de palavras. Cada usuário terá uma ou mais páginas de seu programa carregado na memória.

- **Endereçamento Paginado por Registrador de Página – cont.**

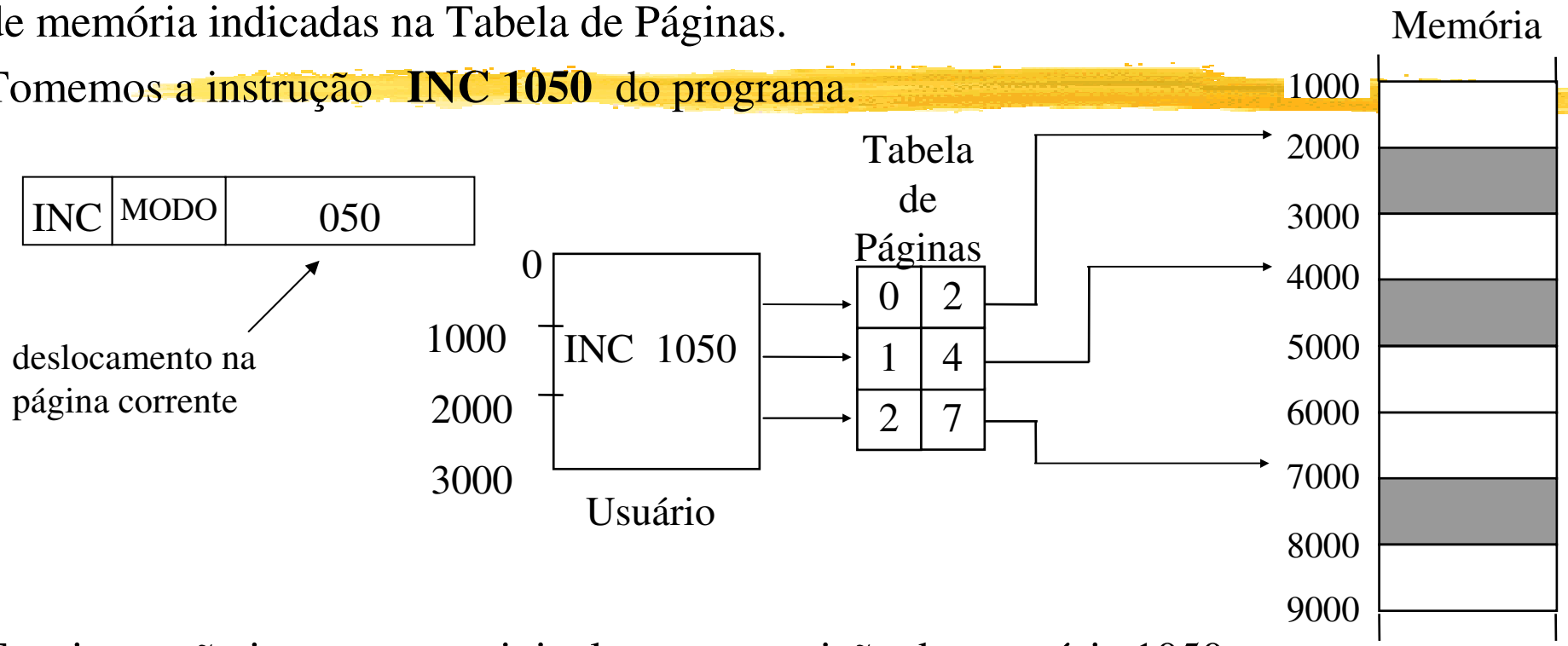
Exemplo (por simplificação as instruções de uma página sempre referenciarão dados na mesma página)

Quando a CPU for executar um determinado programa de usuário, **o Sistema Operacional carregará o Registrador de Páginas (RP) com o número da página de memória onde está armazenado o programa.** Para o gerenciamento das páginas, o Sistema Operacional construirá uma tabela relacionando páginas de memória e páginas de programas.



No exemplo, o programa de um usuário foi compilado utilizando endereçamento paginado. As 3 páginas do programa foram carregadas nas páginas de memória indicadas na Tabela de Páginas.

Tomemos a instrução **INC 1050** do programa.



Esta instrução incrementa originalmente a posição de memória 1050 para o programa iniciando na posição “zero” e será compilada com endereçamento paginado, constando somente o deslocamento no campo de endereço da instrução. Desta forma, ao ser executada a instrução, estando o registrador de páginas com o valor "4", o endereço efetivo obtido será "4050", o que resulta em uma execução correta (lembremos que agora esta instrução está na segunda página).

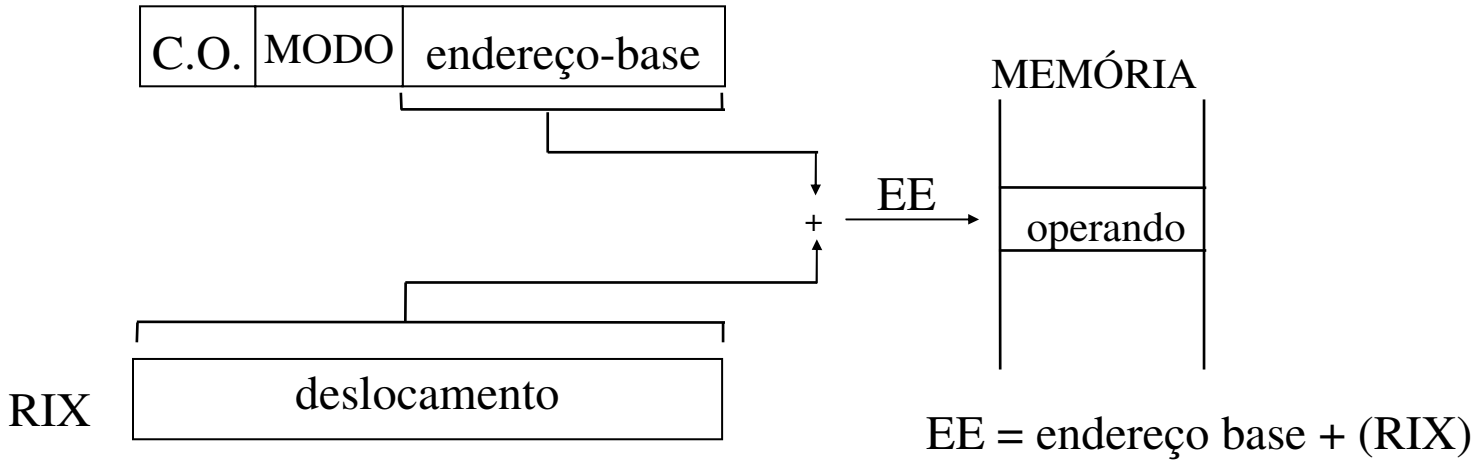
I.1 Endereçamento indexado Direto

O cálculo do **endereço efetivo** envolve o conteúdo de um *Registrador Indexador* (RIX) e de um endereço-base especificado no campo de endereço da instrução. O registrador indexador define um *Deslocamento* com relação ao endereço-base.

Máquinas que possuem o modo de endereçamento indexado possuem um ou mais registradores indexadores que podem ser manipulados (carregados, incrementados, decrementados, etc.) por instruções do repertório de instruções da máquina.

Endereçamento indexado Direto (cont.)

No modo **indexado direto** o conteúdo do registrador indexador é somado ao endereço-base para determinação do endereço efetivo.



Exemplo: para instrução de 2 operandos, sendo o 1º operando no modo indexado e o 2º no modo registrador.

MOV RIX(100),R1 ; R1 ← (100+(RIX))

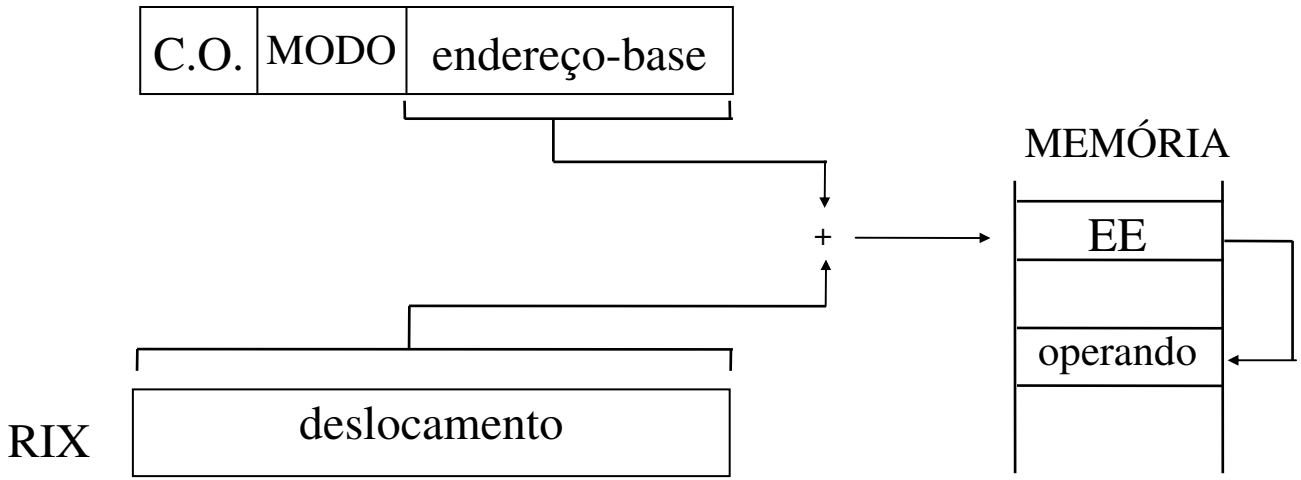
Utilização:

É uma forma de endereçamento muito conveniente para o acesso aleatório a elementos de um vetor.

I.2 Endereçamento indexado Indireto

O modo de endereçamento indexado indireto pode ser **Pré-Indexado** ou **Pós-Indexado**, conforme o conteúdo do registrador indexador seja somado antes ou depois da busca do endereço indireto no cálculo do **endereço efetivo**.

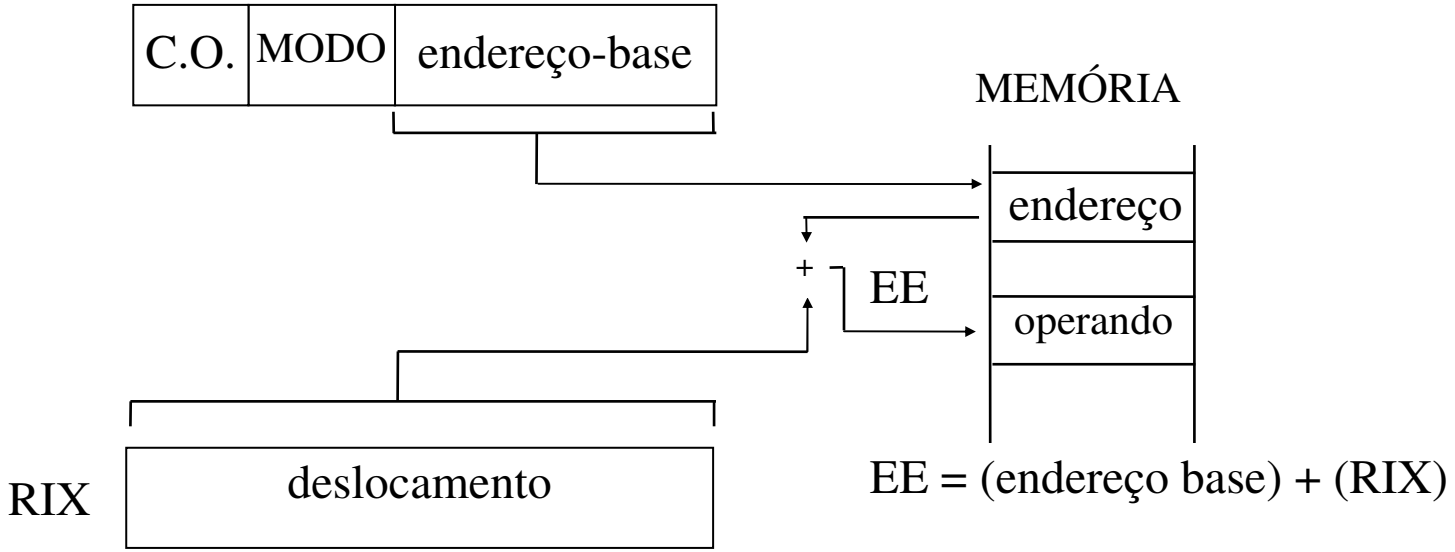
Pré-Indexado: o deslocamento contido no registrador indexador é somado ao endereço base antes da busca do endereço indireto.



EE = (endereço base + (RIX))

Endereçamento indexado Indireto (cont.)

Pós-Indexado: : o deslocamento contido no registrador indexador é somado ao conteúdo da posição de memória dada pelo endereço-base (indireto).



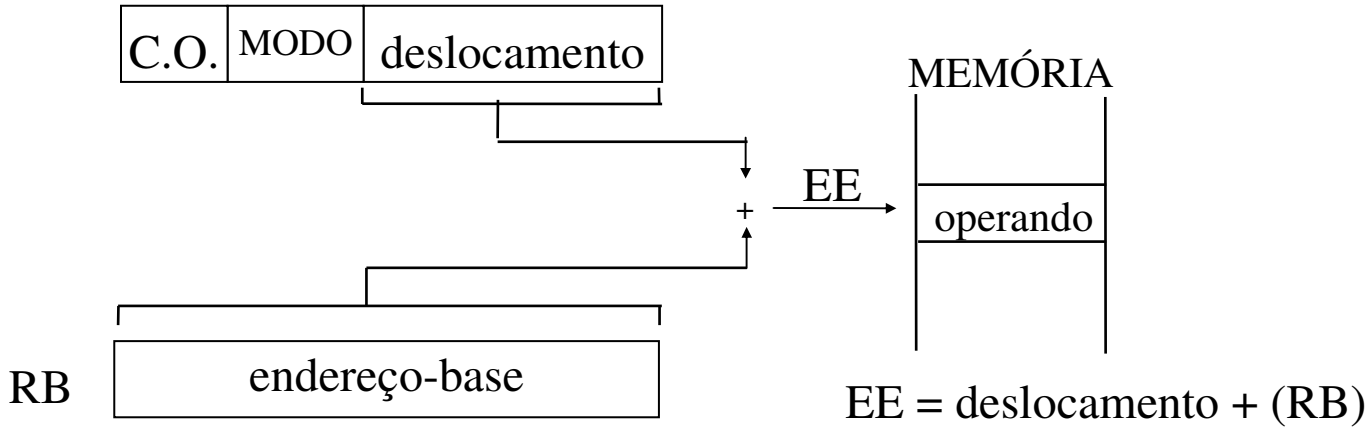
Exemplo: para instrução de 2 operandos, sendo o 1º operando no modo indexado indireto pós-indexado e o 2º no modo registrador.

MOV RIX((100)),R1 ; R1 ← ((100) + (RIX))

Utilização: Acesso a estruturas de dados complexas, como a uma tabela de endereços. Os endereços desta tabela podem ser, por sua vez, endereços de outras estruturas de dados.

J. Endereçamento Baseado

O **endereço efetivo** é calculado com a adição do conteúdo de um Registrador de Base (RB) a um deslocamento contido no campo de endereço da instrução. O deslocamento é interpretado como um número com sinal, permitindo um deslocamento, positivo ou negativo, em relação à posição dada pelo conteúdo do registrador RB.



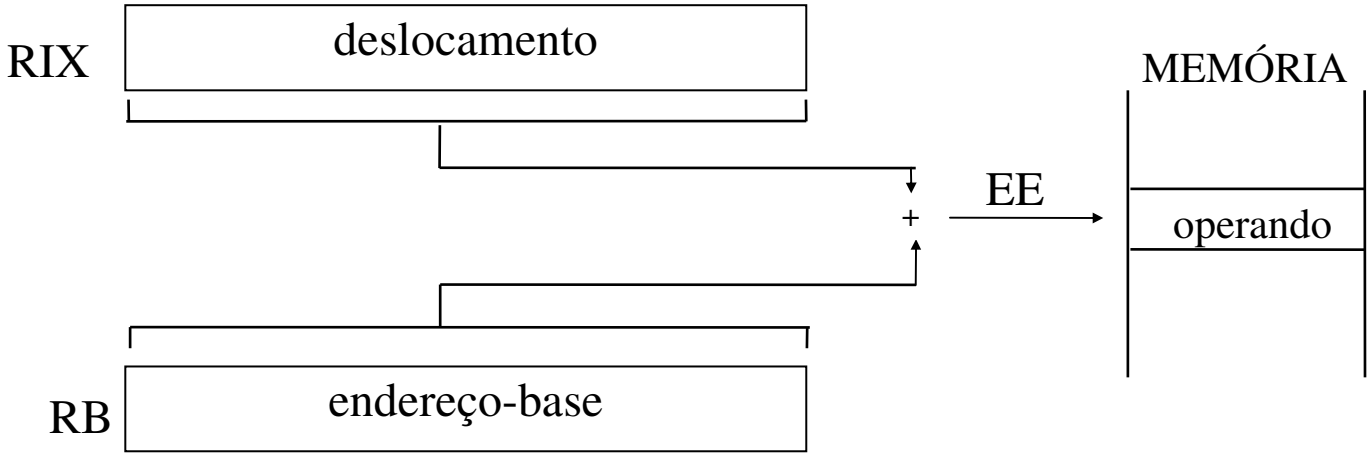
Exemplo: para instrução de 2 operandos, sendo o 1º operando no modo baseado e o 2º no modo registrador.

MOV 8(RB),R1 ; R1 ←((RB)+8)

Utilização: este tipo de endereçamento pode ser utilizado pelo programador no acesso a tabelas. É também muito utilizado em software de sistemas para gerenciamento de memória.

K. Endereçamento Baseado e Indexado

O **endereço efetivo** é calculado como a soma do endereço-base com o deslocamento.



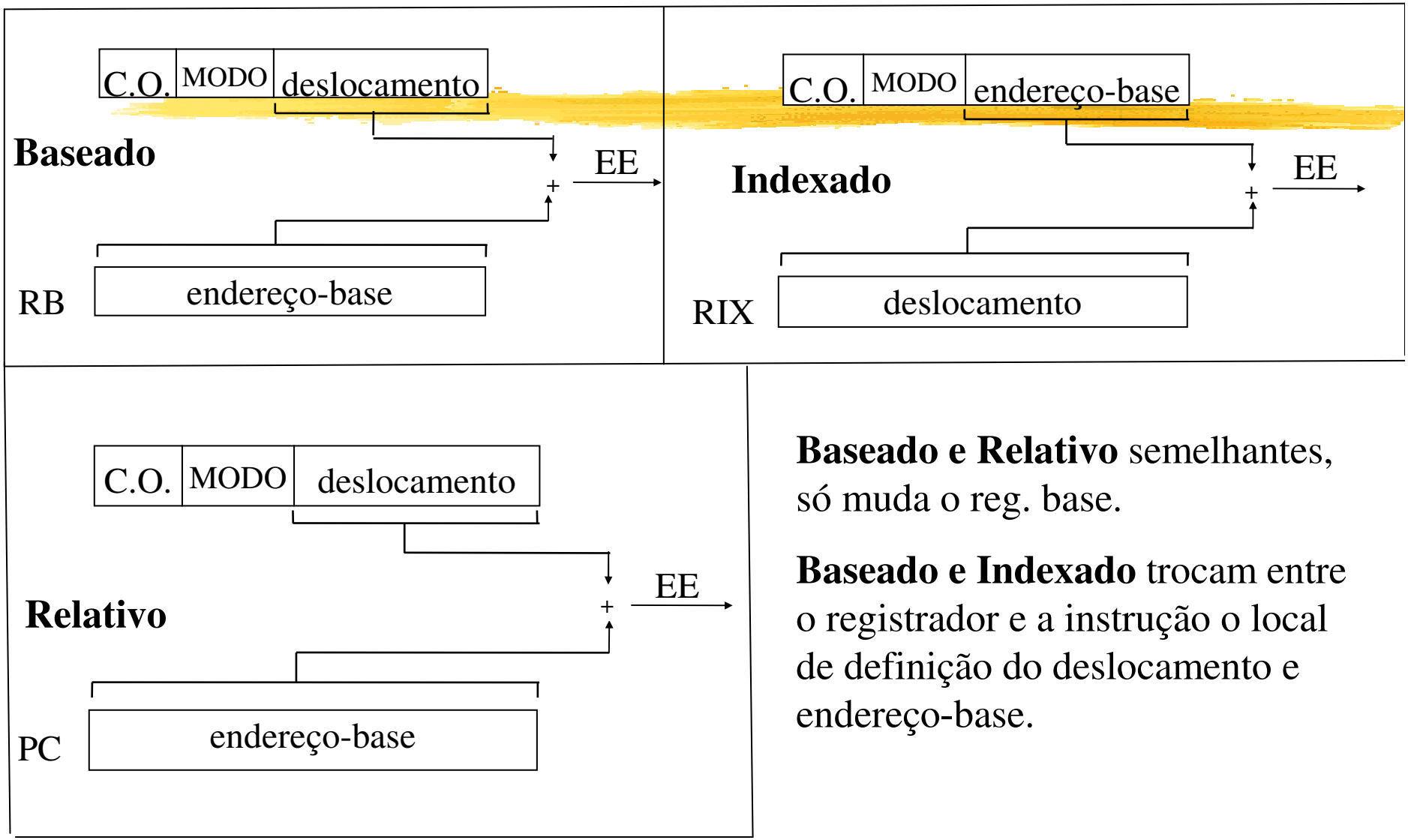
EE = (RB) + (RIX)

Exemplo: em uma instrução, serão referenciados os registradores utilizados como base e índice.

C.O.	MODO	RB	RIX

Utilização: este modo de endereçamento é o mais versátil dos modos de endereçamento. Qualquer um dos exemplos de utilização apresentados pode ser implementado com este modo de endereçamento.

Resumo: Baseado + Indexado + Relativo



Baseado e Relativo semelhantes, só muda o reg. base.

Baseado e Indexado trocam entre o registrador e a instrução o local de definição do deslocamento e endereço-base.

5.3 Exercícios

1. Resolva os exercícios (2) a (8) do livro texto.