

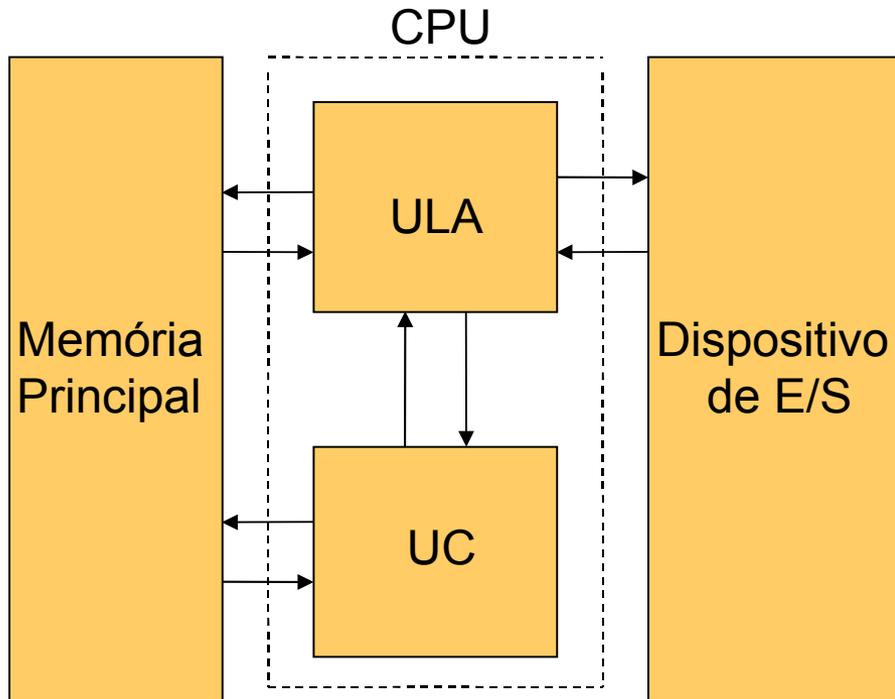
Arquitetura de Computadores

- Princípio de Funcionamento da CPU

Por

Helcio Wagner da Silva

Arquitetura de Von Neumann



ULA : Unidade Lógica Aritmética

UC : Unidade de Controle

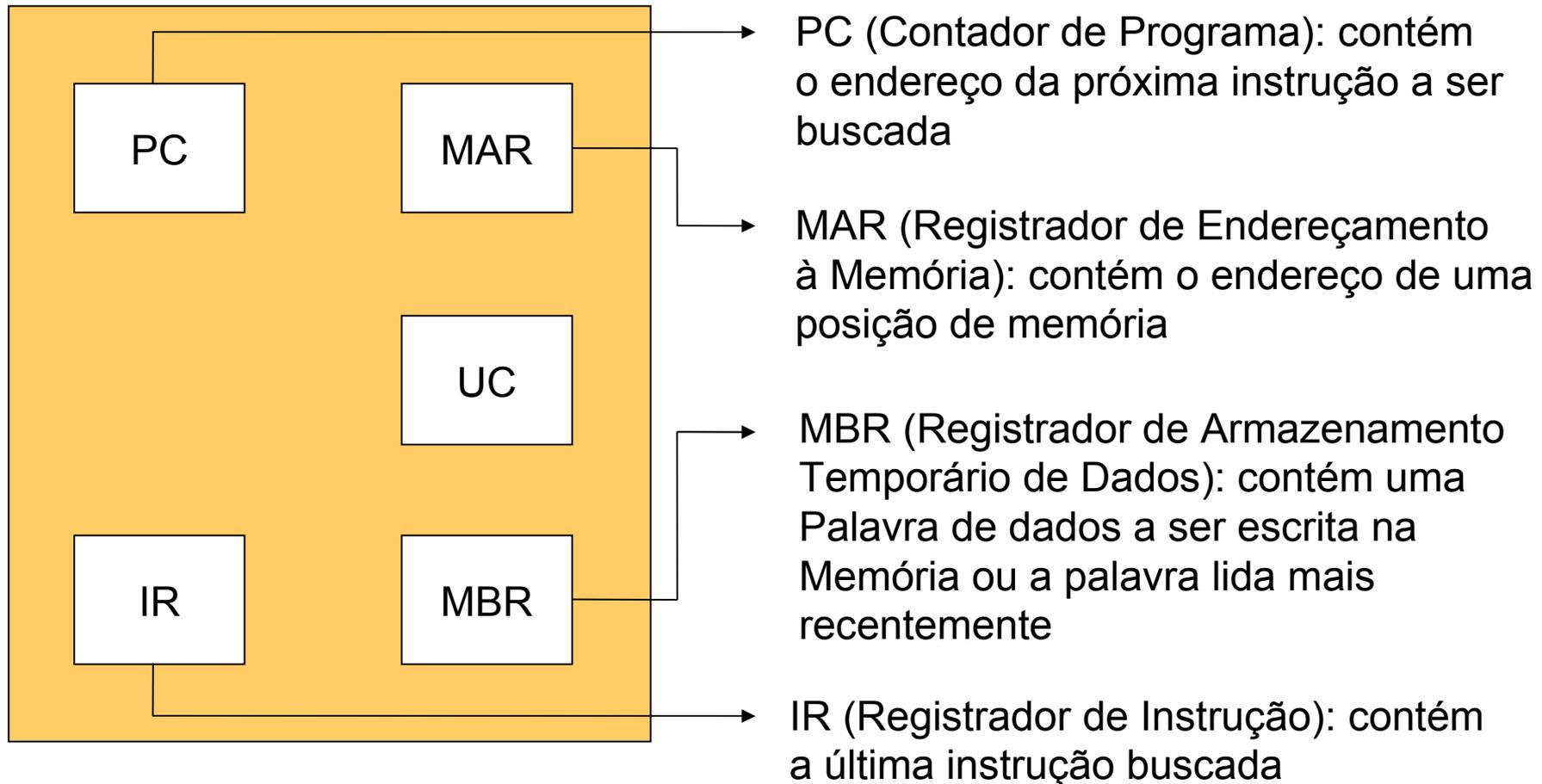
CPU: Unidade Central de Processamento

- Os dados e as instruções são armazenados em uma única memória de leitura e escrita
- O conteúdo da memória é endereçado pela sua posição, independentemente do tipo de dados nela contidos
- A execução de instruções ocorre, via de regra, de modo seqüencial

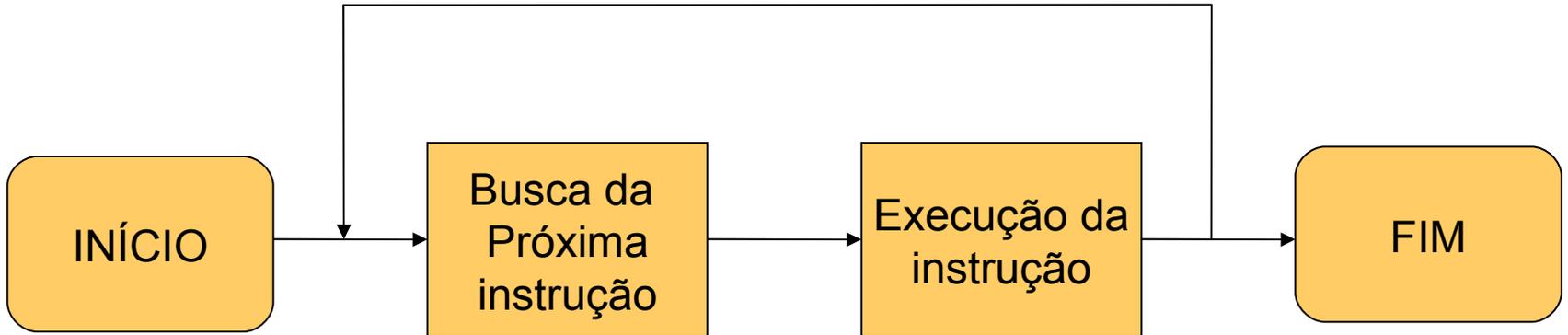
Visão geral de uma CPU

- Além das ULA e UC, os processadores contam com uma pequena quantidade de memória – são os **registradores**
- Os registradores formam o topo da hierarquia de memória de um computador
- Eles se dividem em:
 - Registradores visíveis ao usuário
 - Registradores de controle e estado

Alguns registradores de controle e estado



Ciclo de Instrução Básico



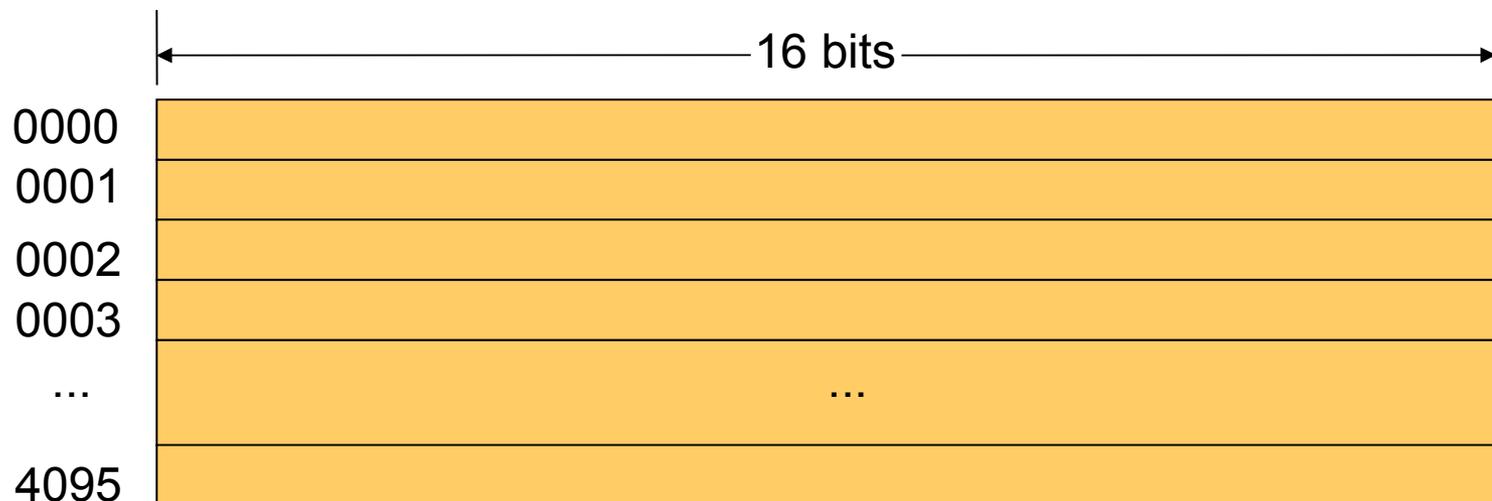
Exemplo de Aplicação do Ciclo de Instrução Básico

- Suposições:
 - A CPU possui apenas um registrador de armazenamento de dados visível ao usuário, denominado AC (Acumulador)
 - Instruções e dados possuem, ambos, 16 bits



Exemplo de Aplicação do Ciclo de Instrução Básico

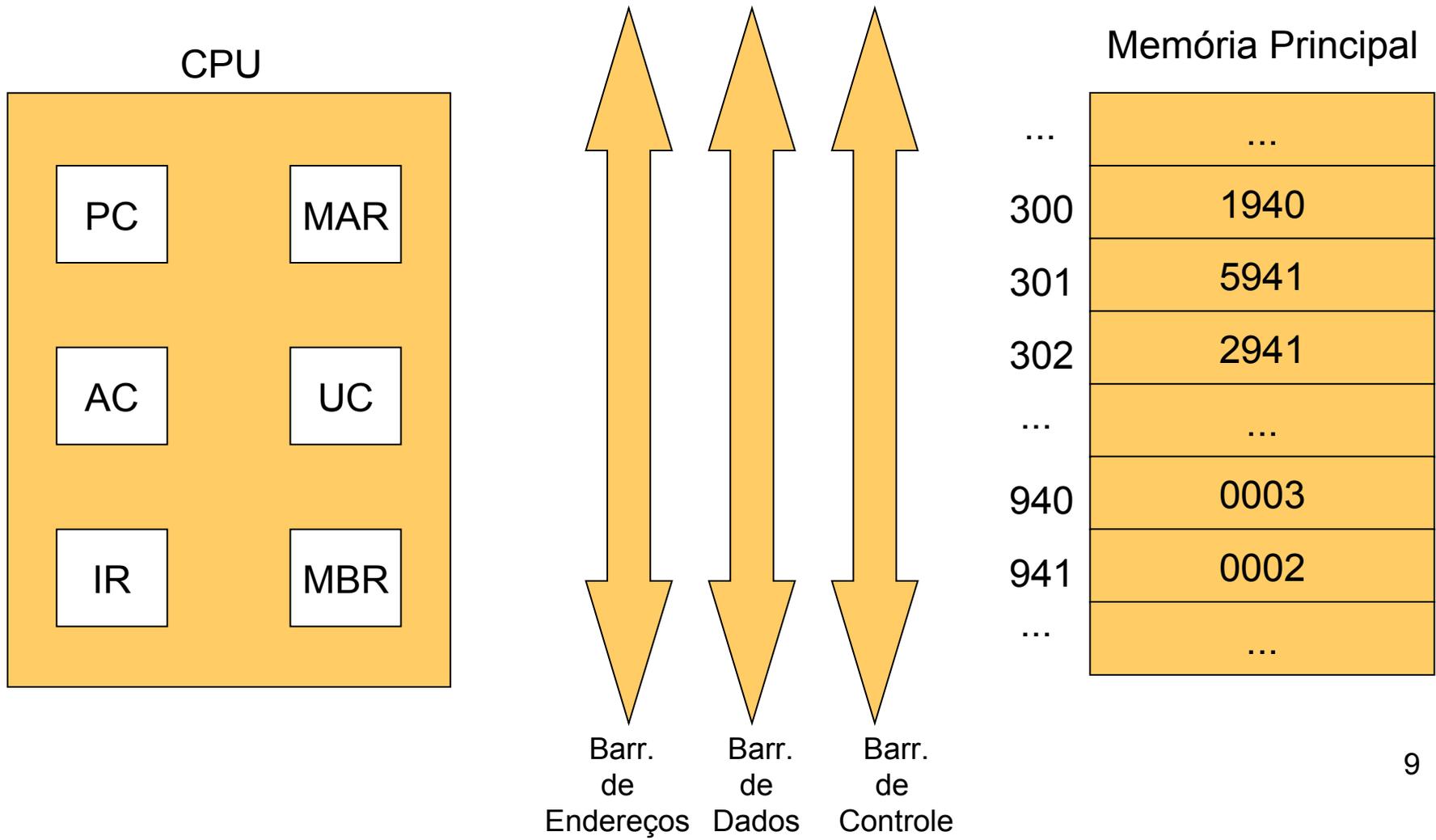
- Suposições (cont.):
 - A memória é organizada em palavras de 16 bits
 - a quantidade de palavras endereçadas diretamente é igual a $2^{12} = 4.096$ (4 K)



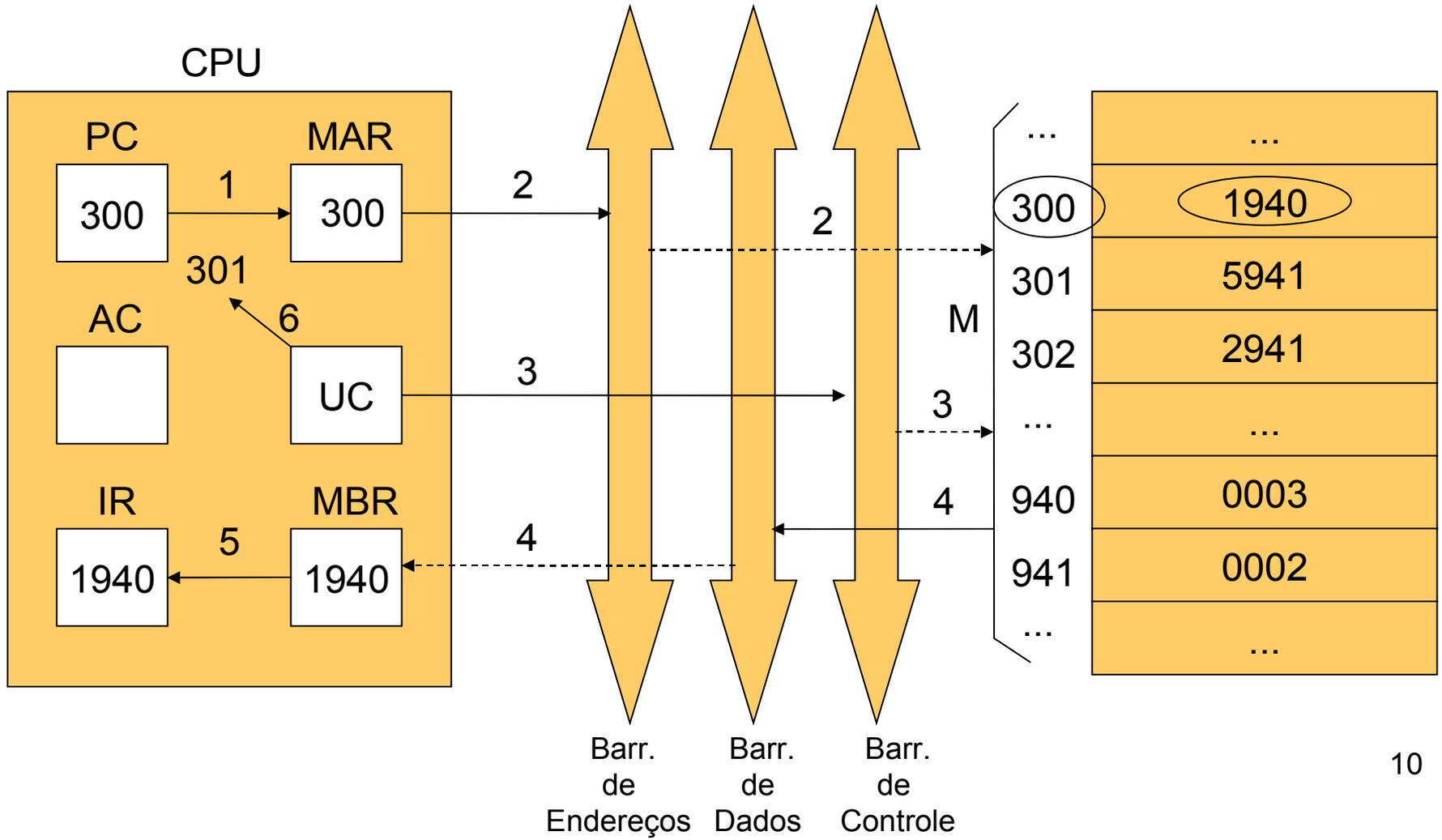
Exemplo de Aplicação do Ciclo de Instrução Básico

- Instruções a serem buscadas e executadas:
 - 1940 : o dado na posição de memória 940 deve ser posto em AC
 - 5941: o dado na posição de memória 941 deve ser somado ao dado em AC, e o resultado armazenado em no AC
 - 2941: o dado localizado em AC deve ser escrito na memória na posição 941

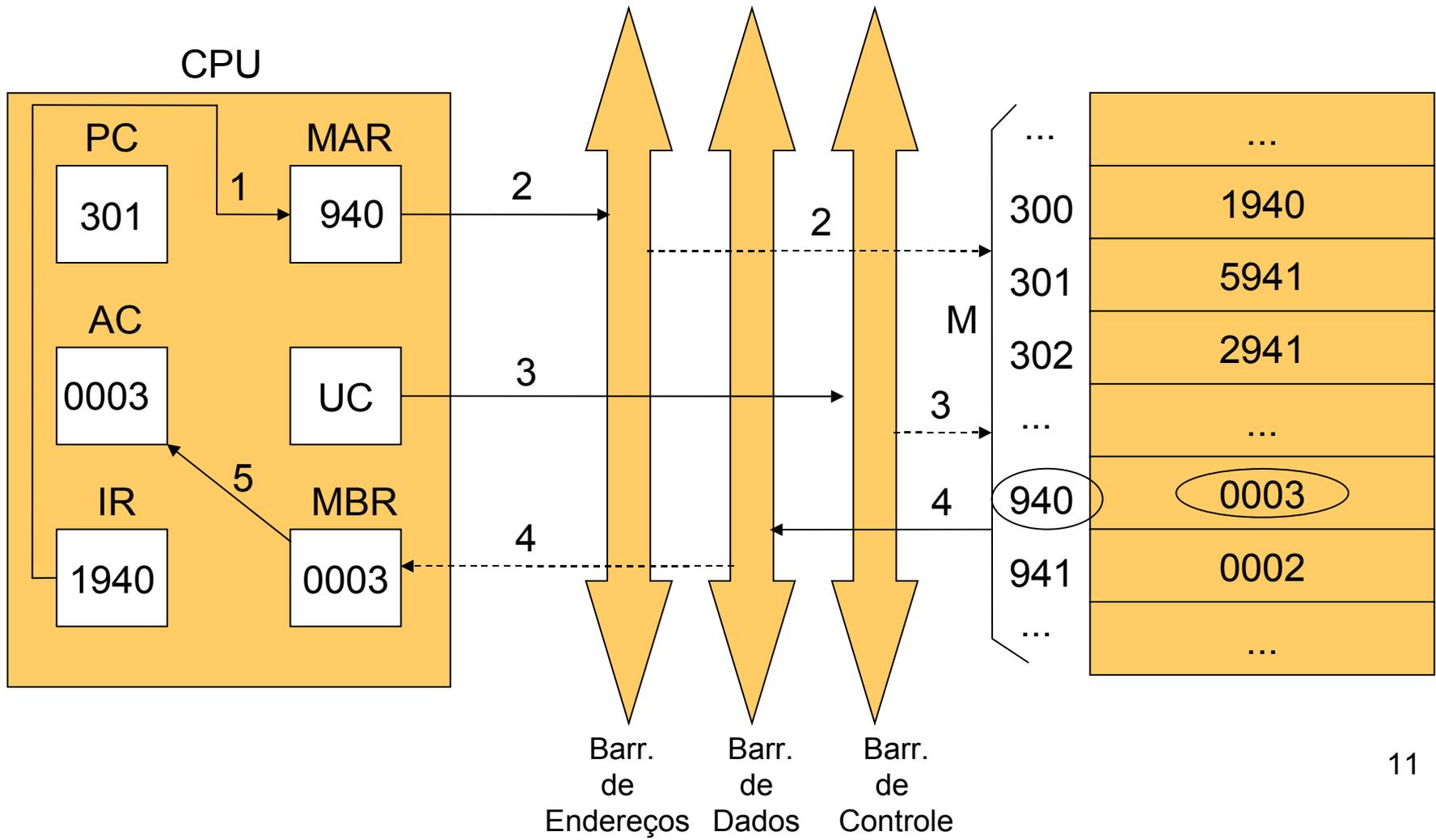
Exemplo de Aplicação do Ciclo de Instrução Básico



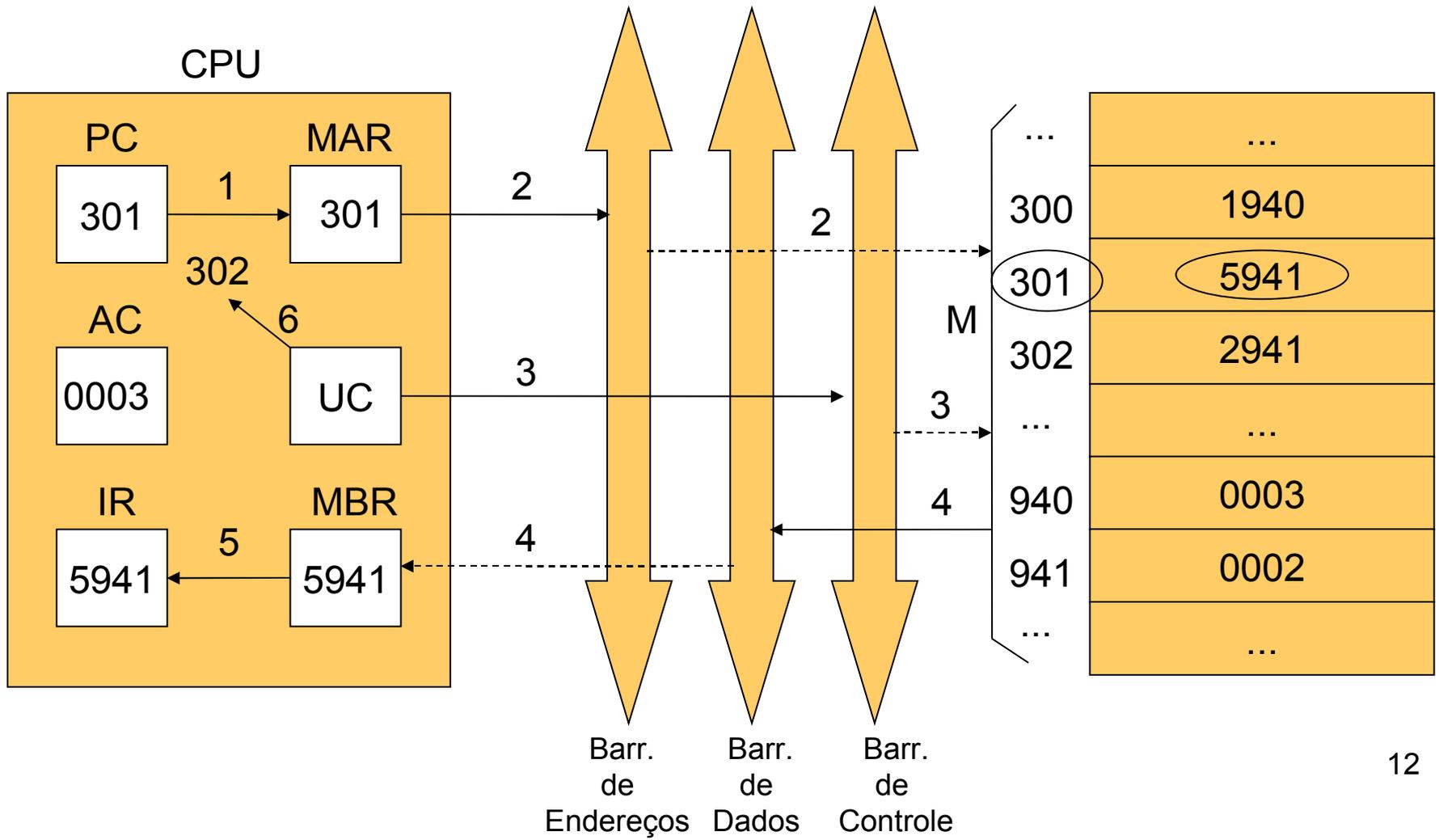
Busca da 1ª Instrução



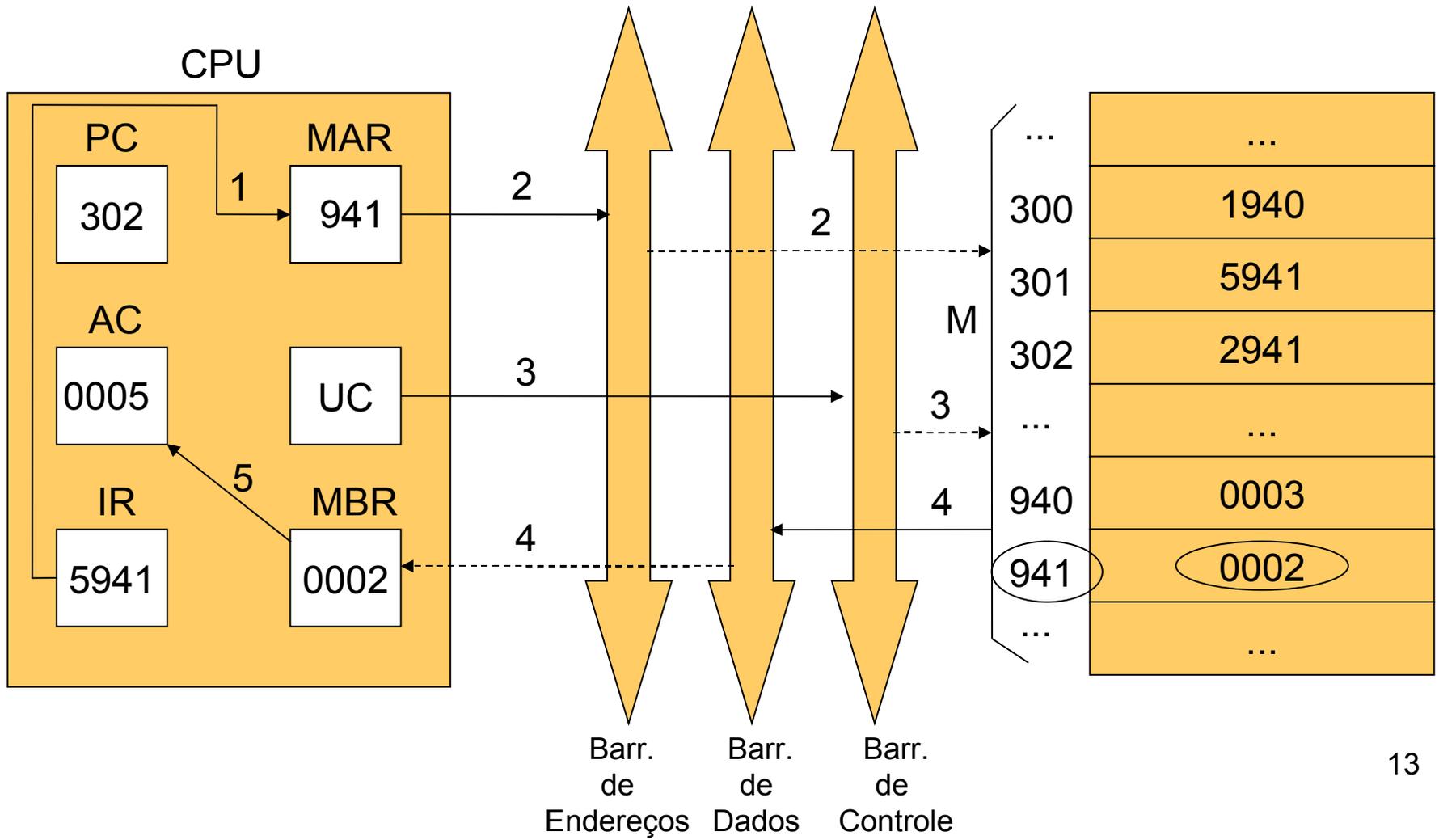
Execução da 1ª Instrução



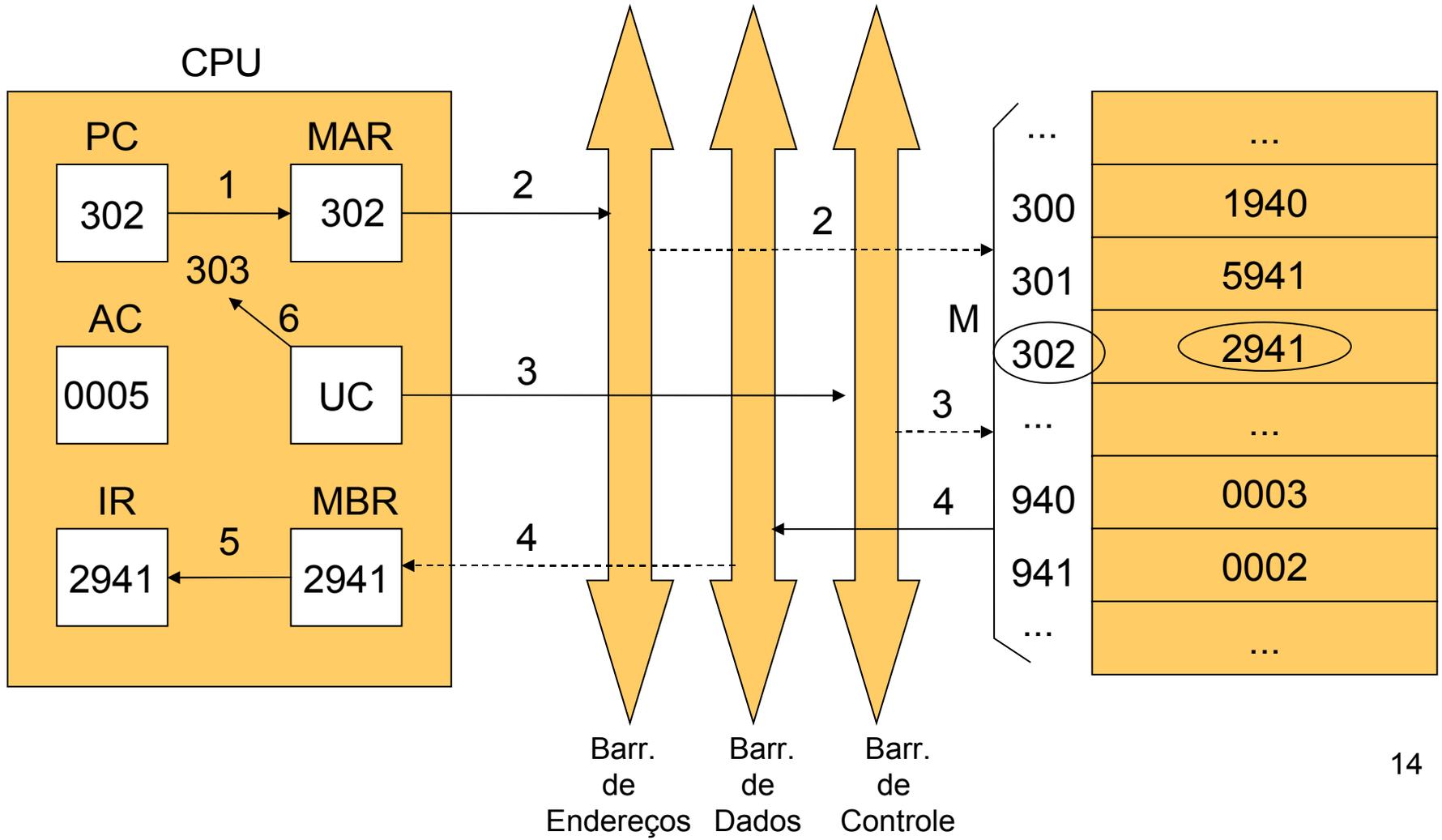
Busca da 2ª Instrução



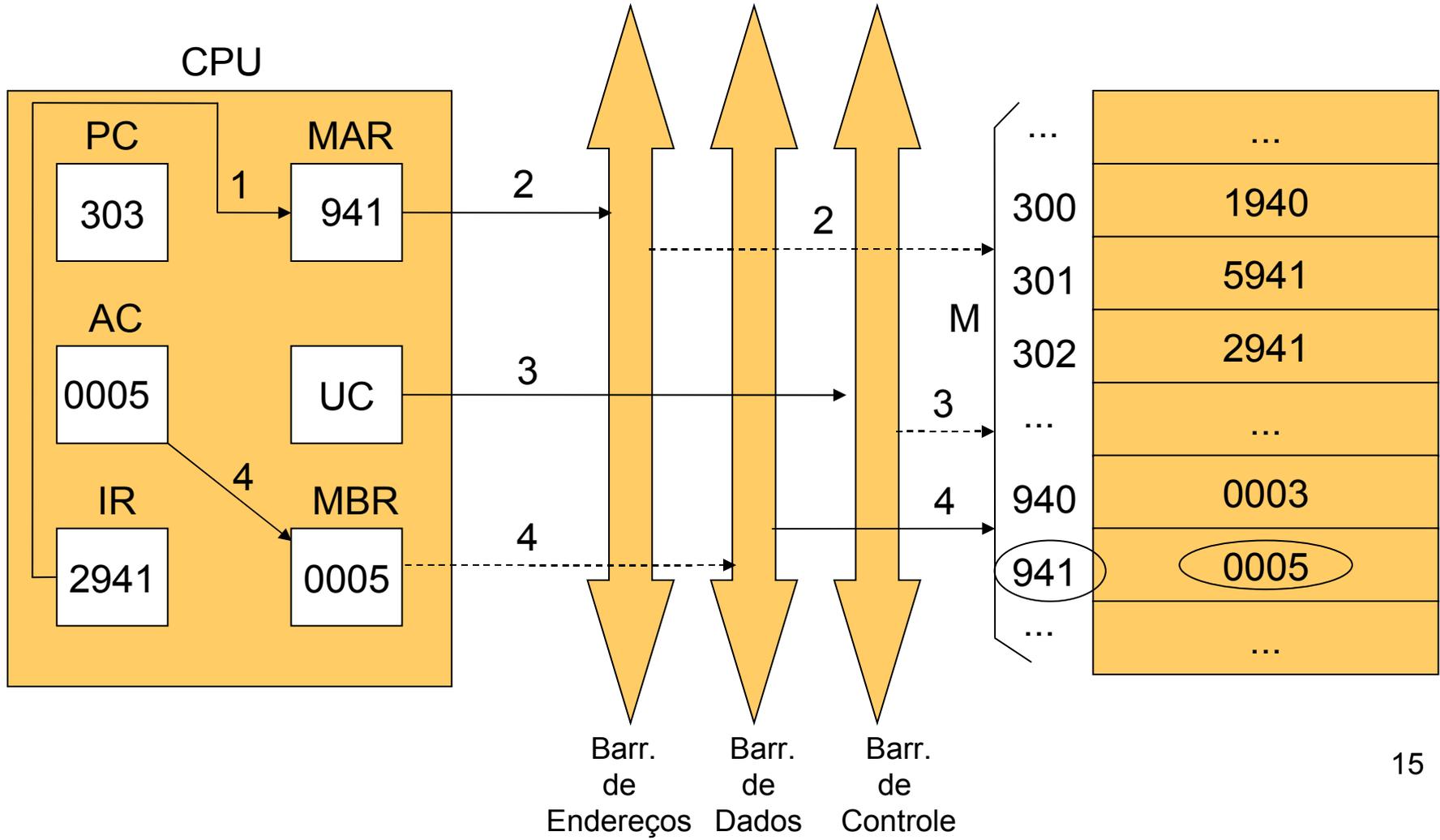
Execução da 2ª Instrução



Busca da 3ª Instrução



Execução da 3ª Instrução



Modos de Endereçamento

- Os campos de endereços das instruções são relativamente pequenos
- Para possibilitar o acesso a uma grande quantidade de posições da memória, foram criados vários *modos de endereçamento*
- Quase todas as arquiteturas de computadores fornecem mais de um desses modos de endereçamento

Modos de Endereçamento mais comuns

- Endereçamento Imediato
- Endereçamento Direto
- Endereçamento Indireto
- Endereçamento de Registrador
- Endereçamento Indireto via Registrador
- Endereçamento por Deslocamento
 - Endereçamento Relativo
 - Endereçamento via Registrador Base
 - Indexação
- Endereçamento à Pilha

Endereçamento Imediato

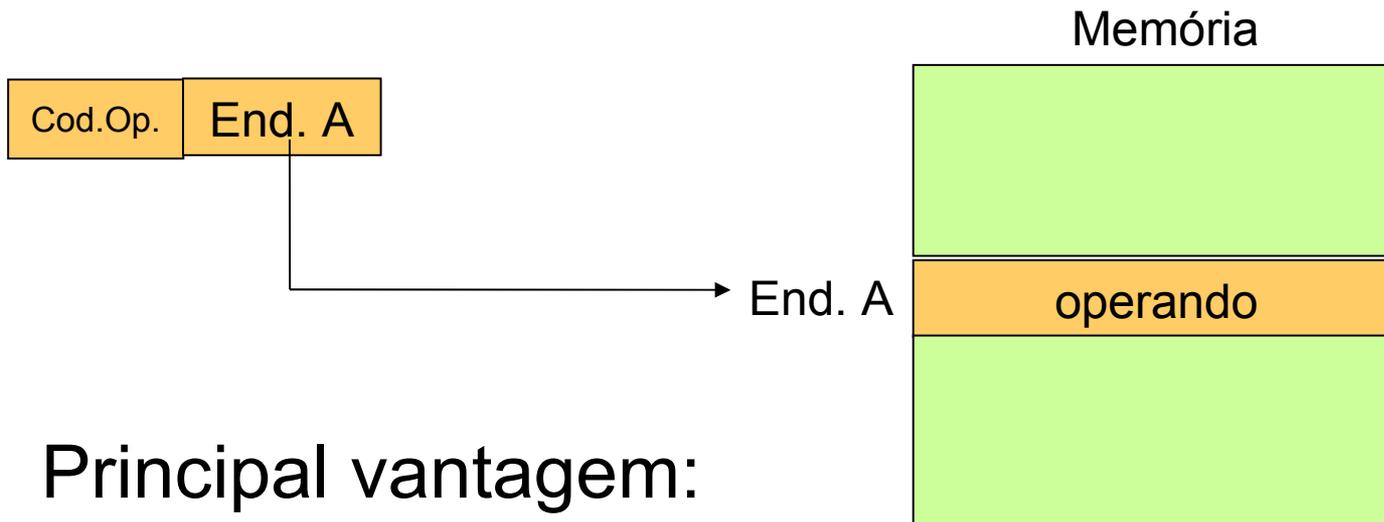
- O valor do operando é especificado diretamente na instrução



- Principal vantagem:
 - Nenhum acesso à memória (economia de tempo)
- Principal desvantagem:
 - O tamanho do operando é limitado pelo tamanho do campo de endereço da instrução⁸

Endereçamento Direto

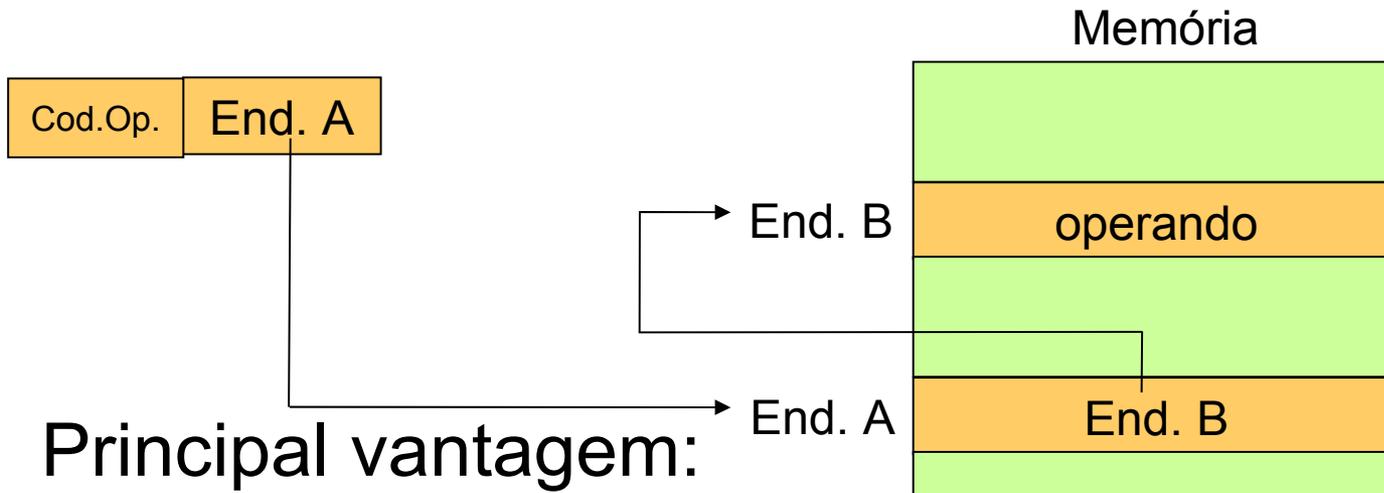
- O campo de endereço da instrução contém o endereço do operando na memória



- Principal vantagem:
 - Simplicidade
- Principal desvantagem:
 - Espaço de endereçamento limitado

Endereçamento Indireto

- O campo de endereço da instrução contém um endereço da memória cujo conteúdo é o endereço do operando na memória



- Principal vantagem:
 - Espaço de endereçamento grande
- Principal desvantagem:
 - Acessos múltiplos à memória

Endereçamento de Registrador

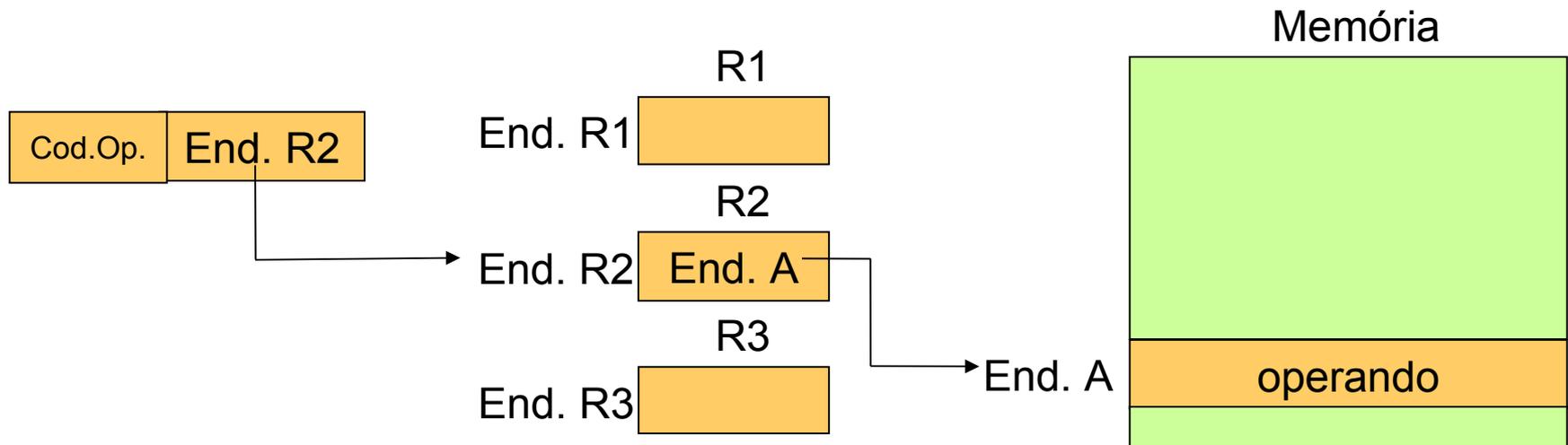
- Semelhante ao endereçamento direto



- Principal vantagem:
 - Nenhum acesso à memória
- Principal desvantagem:
 - Espaço de endereçamento limitado

Endereçamento Indireto via Registrador

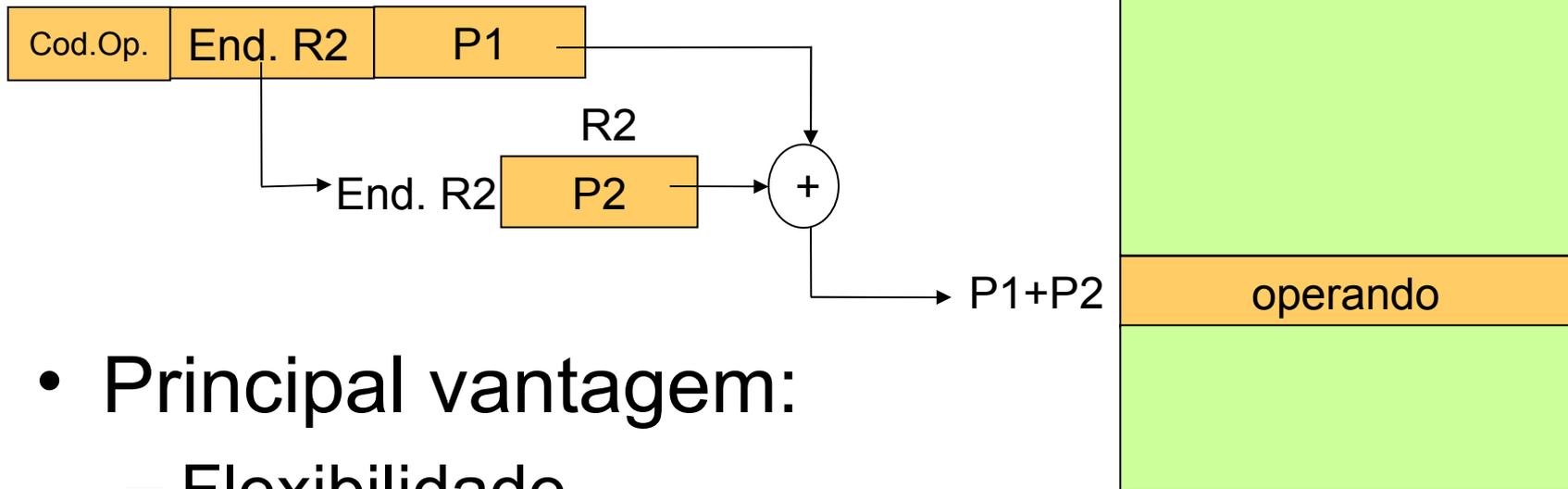
- Semelhante ao endereçamento indireto



- Principal vantagem:
 - Espaço de endereçamento grande
- Principal desvantagem:
 - Acesso extra à memória

Endereçamento por Deslocamento

- É uma combinação dos modos Direto e Indireto via Registrador



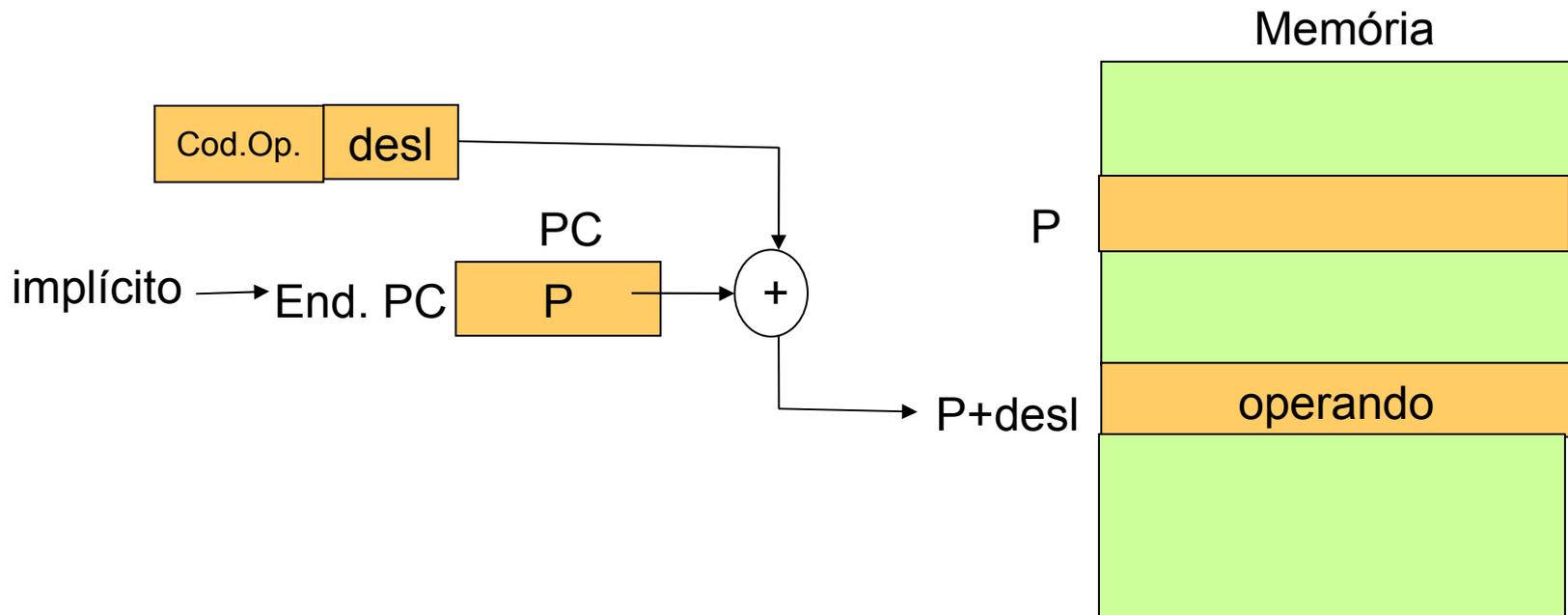
- Principal vantagem:
 - Flexibilidade
- Principal desvantagem:
 - Complexidade

Endereçamento por Deslocamento

- Tipos comuns:
 - Endereçamento Relativo
 - Endereçamento via Registrador Base
 - Indexação

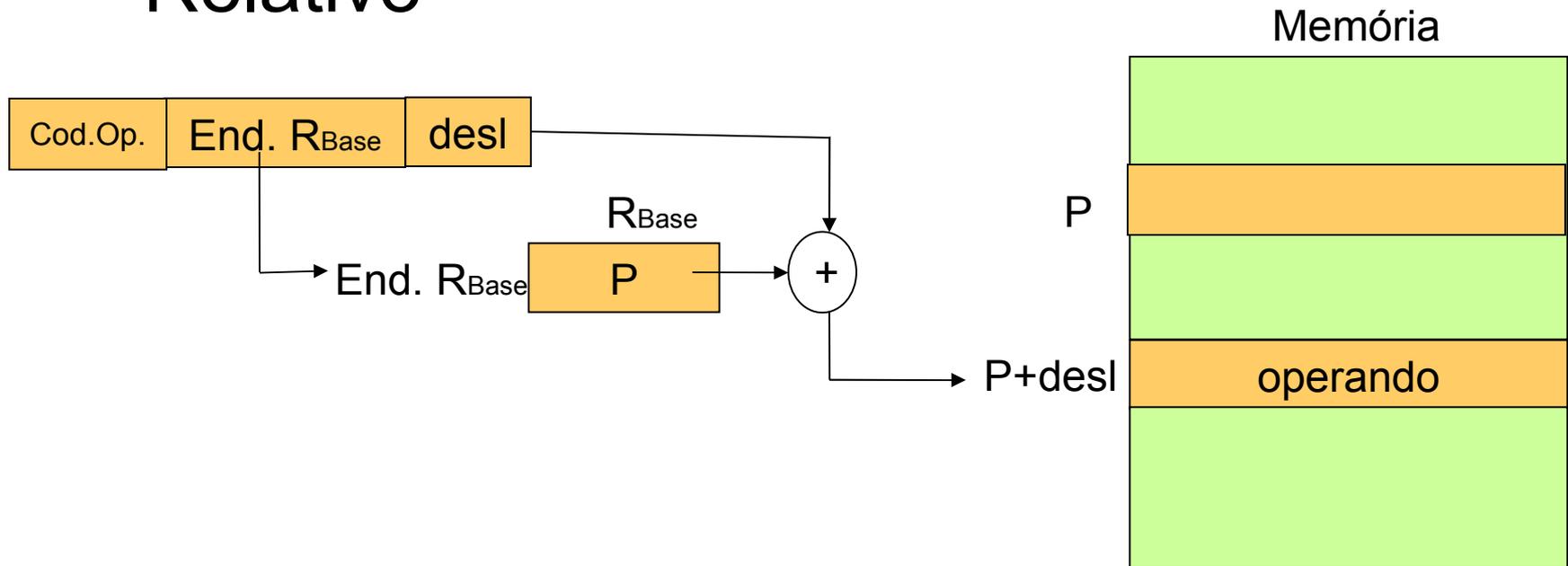
Endereçamento Relativo

- Explora o conceito de localidade, economizando bits no campo de endereço da instrução



Endereçamento via Registrador Base

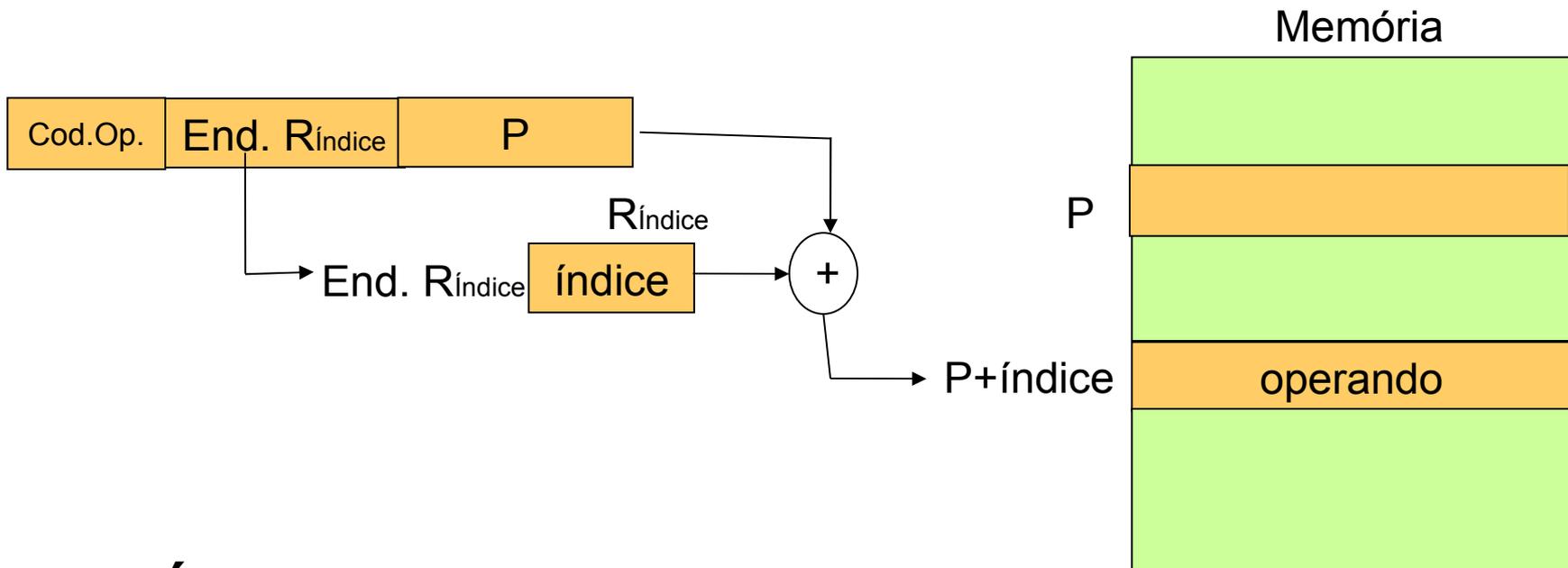
- É uma generalização do Endereçamento Relativo



- Útil na implementação de segmentação de memória

Indexação

- Interpretação oposta ao do Endereçamento via Registrador Base



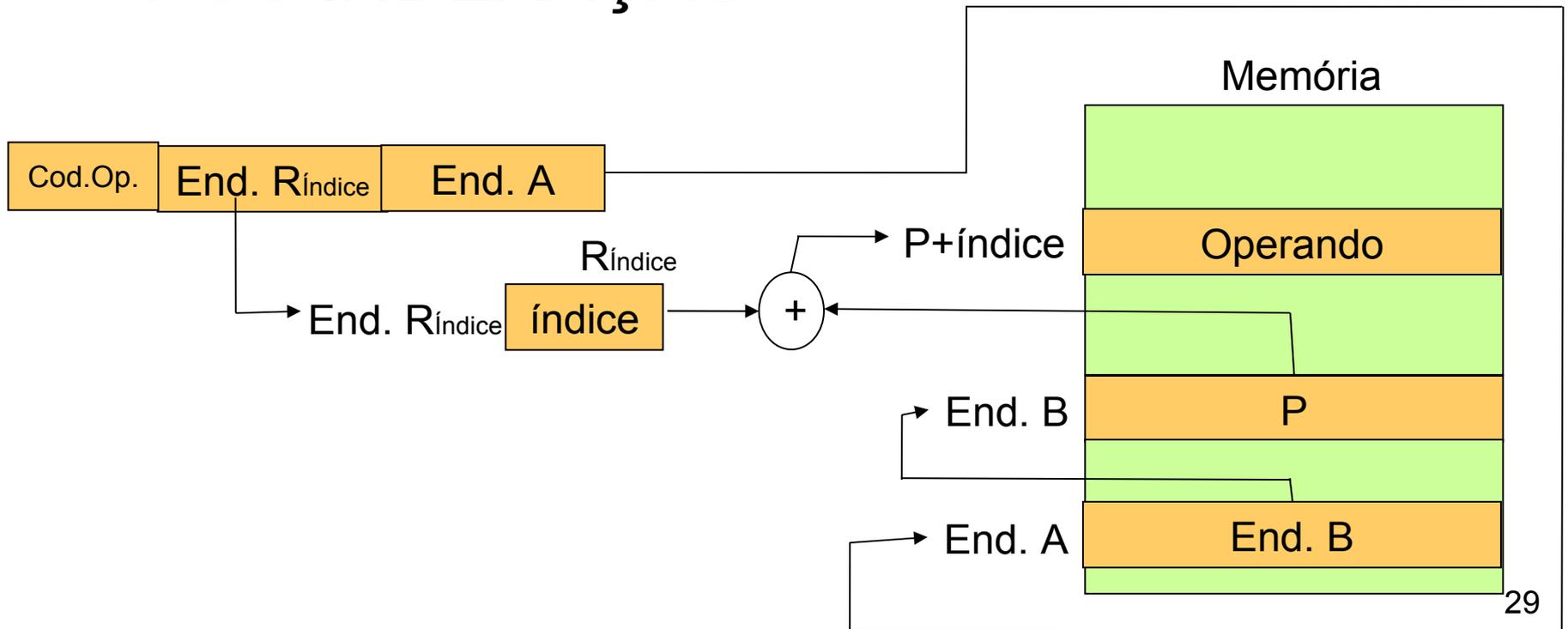
- Útil em operações iterativas

Indexação

- Auto-indexação: incremento do registrador índice no mesmo ciclo de instrução
 - Realizada automaticamente em registradores dedicados exclusivamente à indexação
 - Quando registradores de propósito geral são utilizados, é necessária indicação na instrução de que deva ser realizada
- Algumas máquinas permitem Endereçamento Indireto e Indexação na mesma instrução

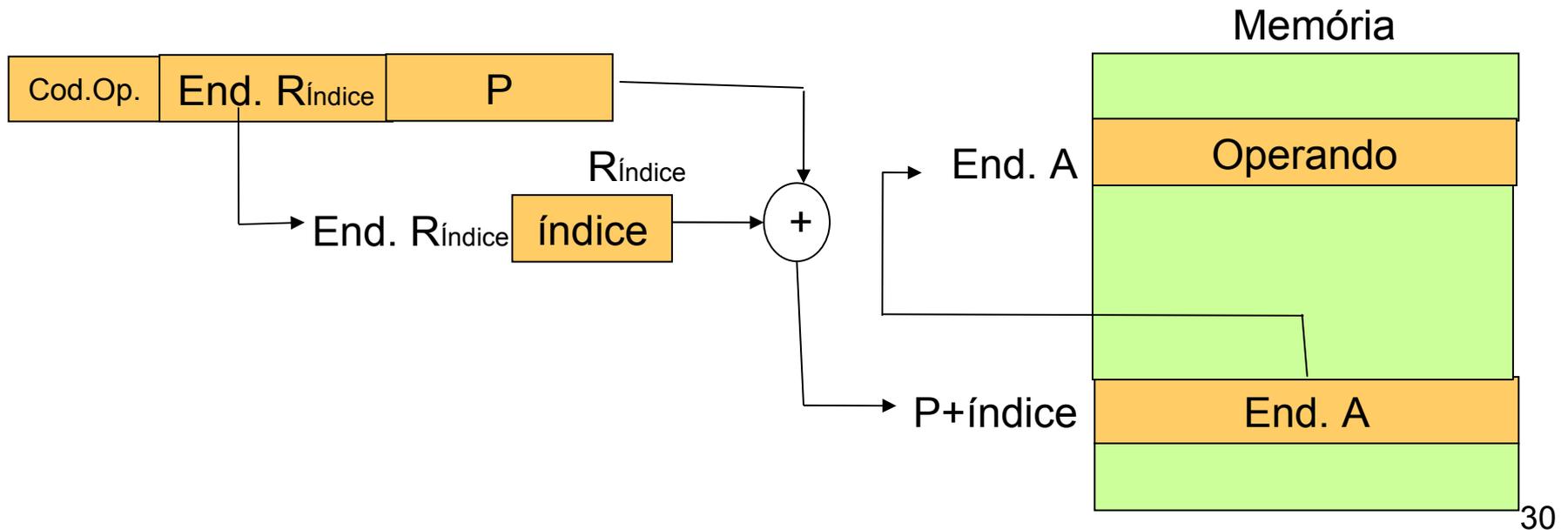
Indexação

- Quando a indexação é feita após o endereçamento indireto, ela é chamada **PÓS INDEXAÇÃO**



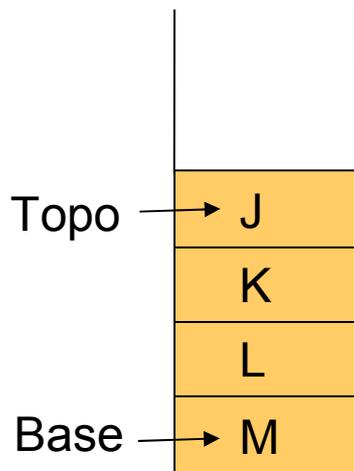
Indexação

- Na PRÉ INDEXAÇÃO, a indexação é feita antes do endereçamento indireto

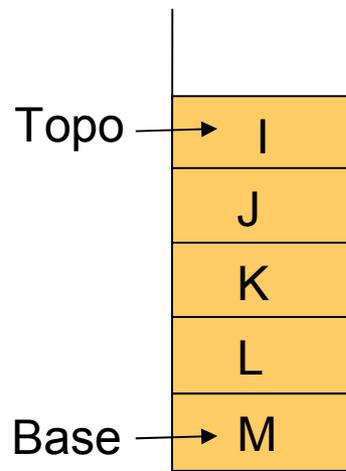


Endereçamento à Pilha

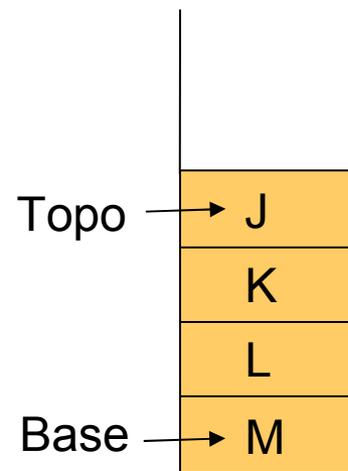
- Pilha: Fila LIFO (*Last In First Out*)
- Operações: PUSH, POP, operações unárias, operações binárias



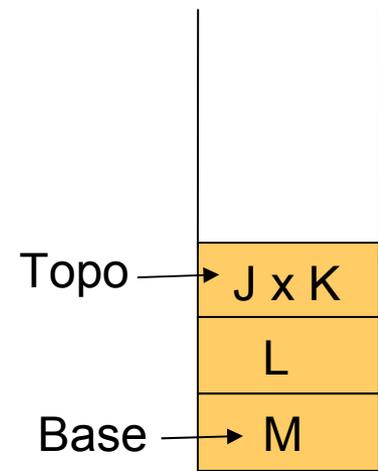
(a)
Início



(b)
Depois do PUSH



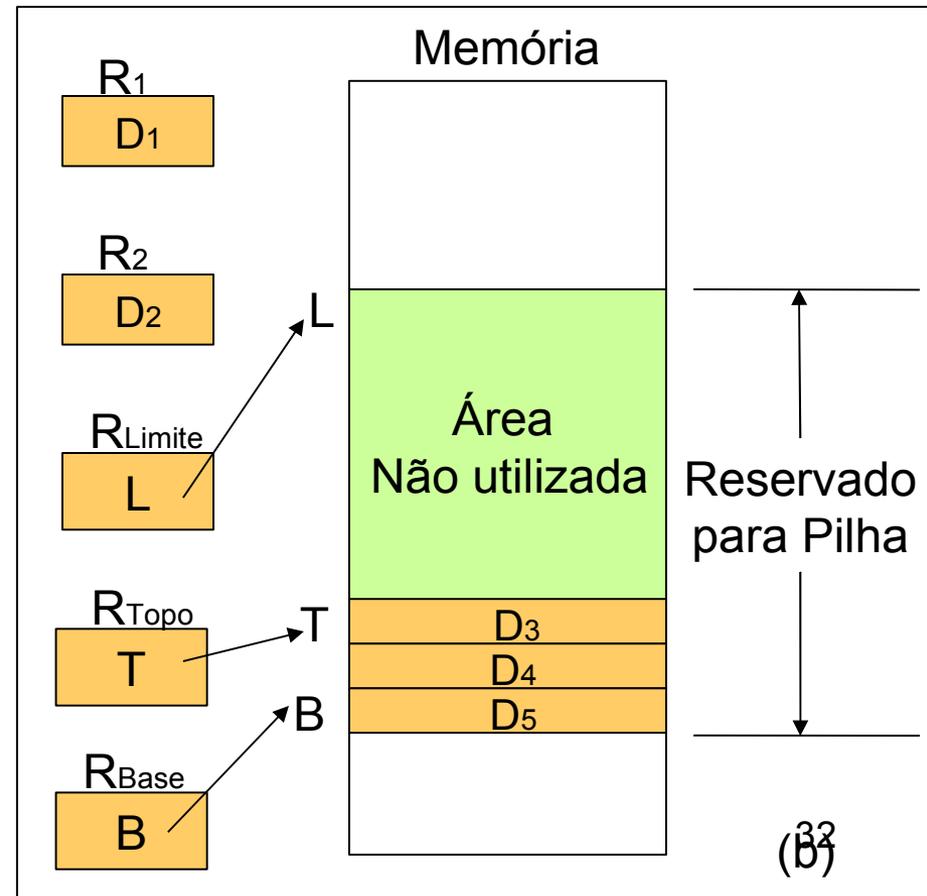
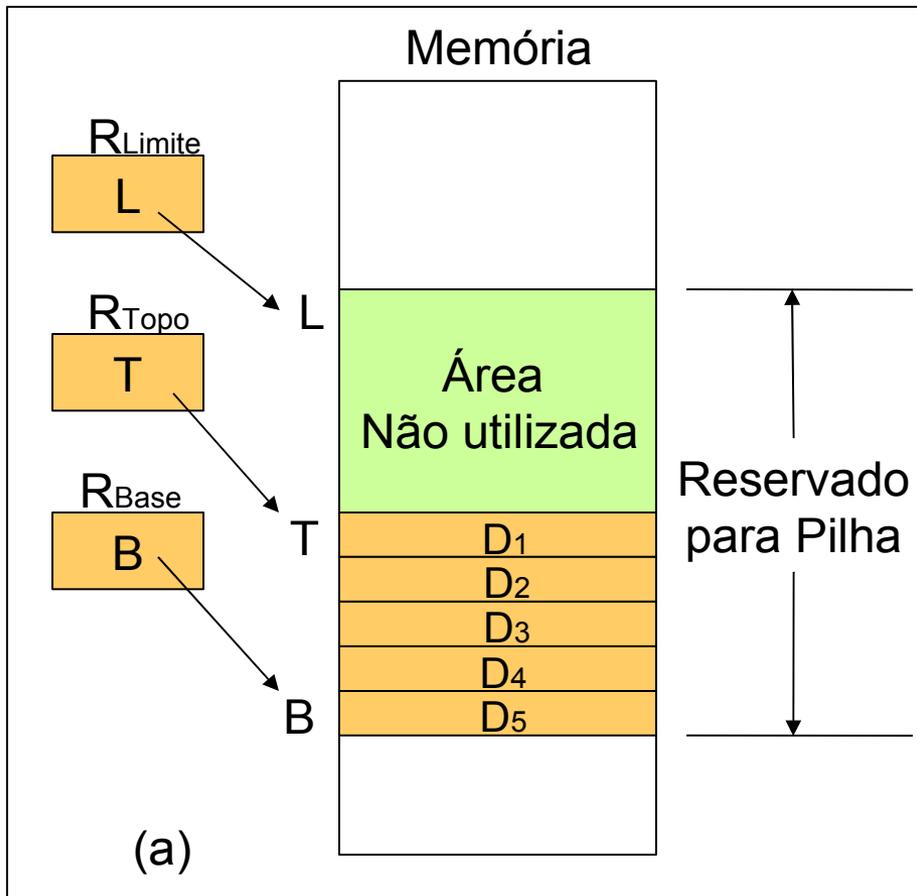
(c)
Depois do POP



(d)
Depois da
Multiplicação

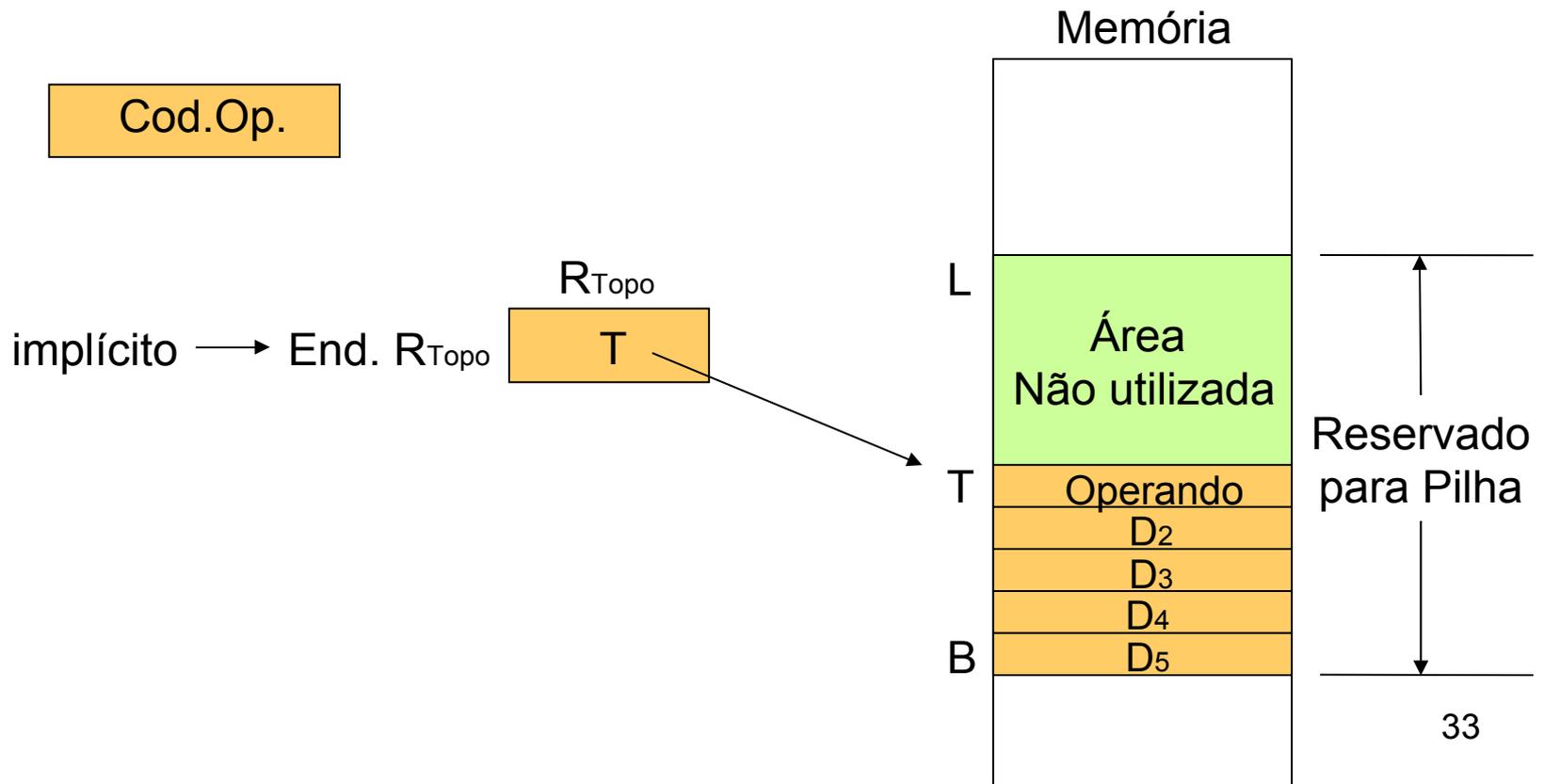
Endereçamento à Pilha

- Implementações típicas da Pilha:



Endereçamento à Pilha

- Trata-se de uma especialização do Endereçamento Indireto via Registrador



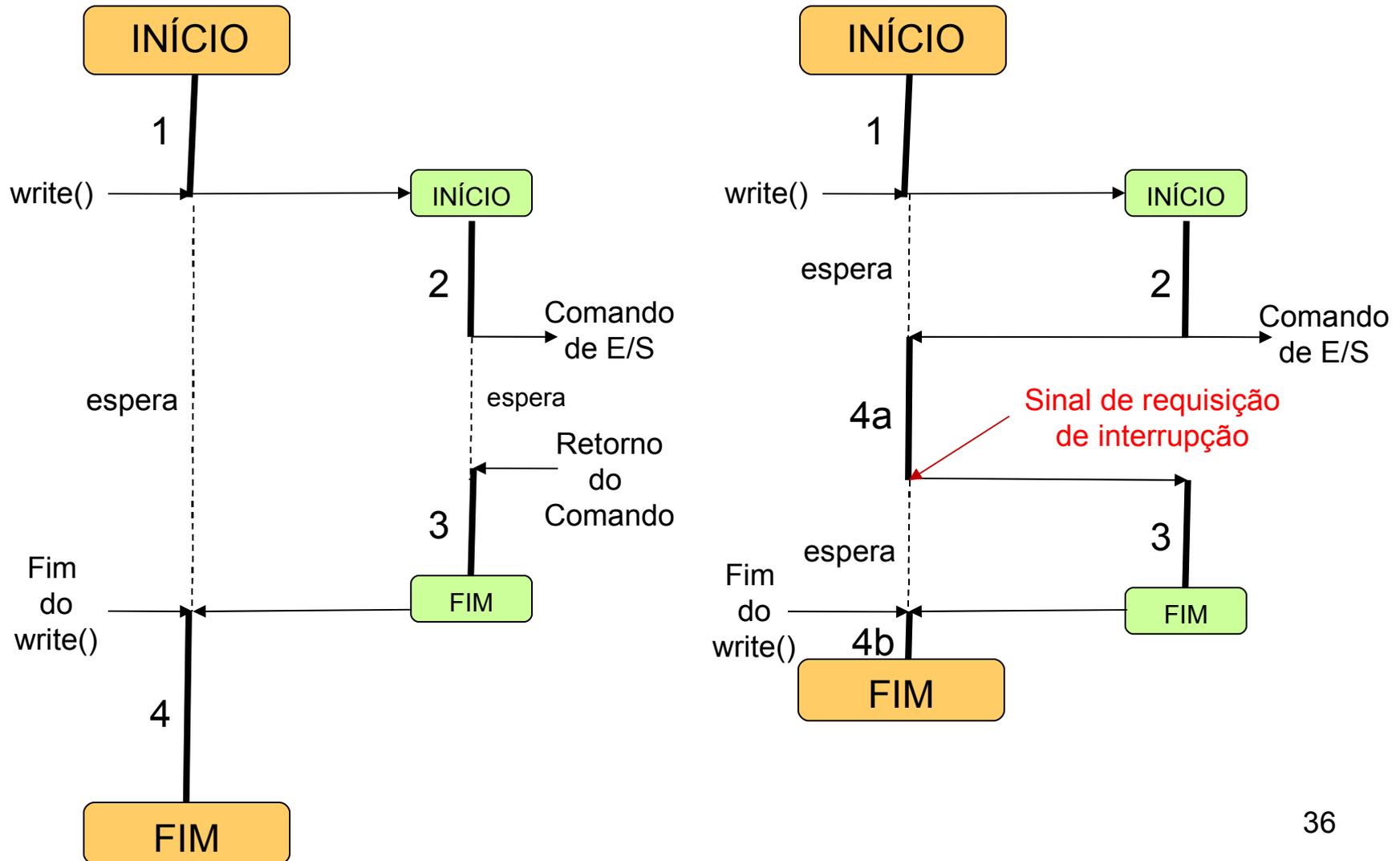
Interrupções

- Mecanismo pelo qual componentes diferentes do processador (E/S, memória) podem interromper a seqüência normal de instruções
- Um dos objetivos das Interrupções é melhorar a eficiência do processamento

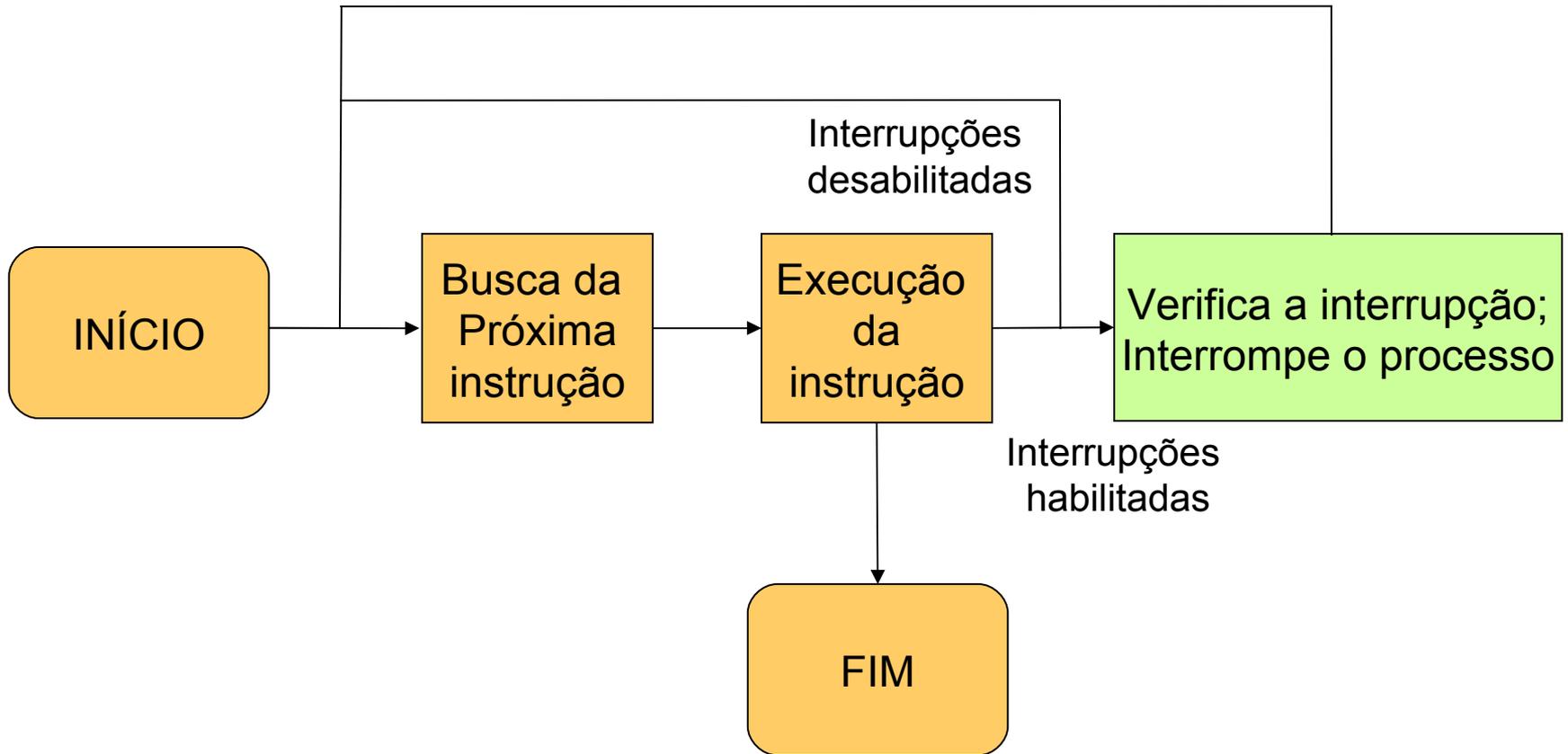
Classes de Interrupções

- Interrupção de SW
 - Gerada por *overflow*, divisão por zero, etc
- Interrupção de Relógio
 - Gerada pelo relógio interno do processador
- Interrupção de E/S
 - Gerada por um controlador de E/S
- Interrupção de falha de HW
 - Gerada em queda de energia, erro de paridade na memória, etc

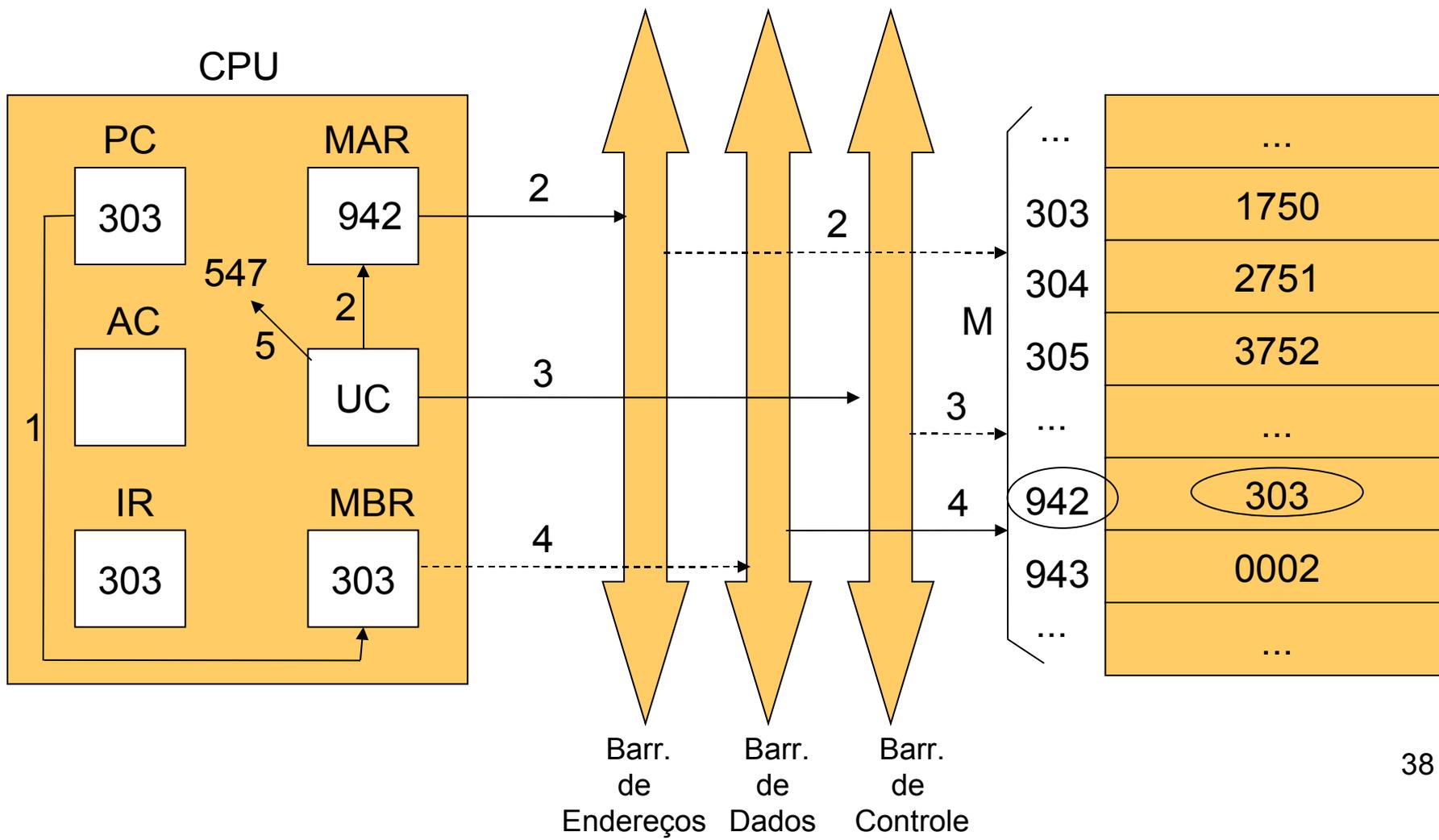
Exemplo de Interrupção



Ciclo de Instrução com Interrupções



Exemplo de Aplicação do Ciclo de Instrução com Interrupções

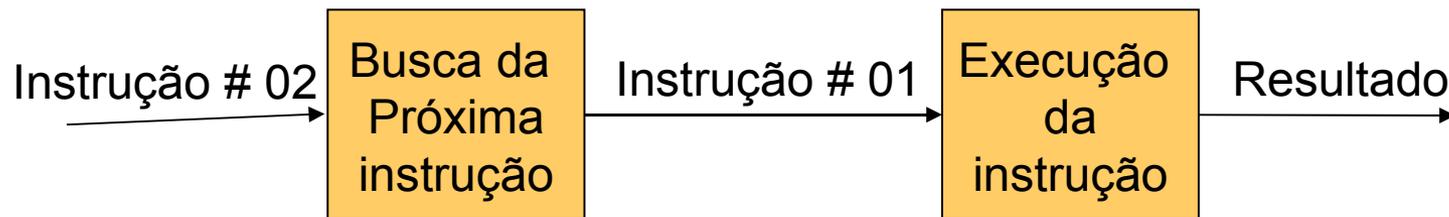


Pipeline de Instruções

- Técnica que objetiva a melhoria do desempenho
- Similar a uma linha de montagem:
 - Uma instrução passa por vários estágios de processamento
 - Instruções em vários estágios podem ser trabalhadas simultaneamente
 - Novas instruções são aceitas numa extremidade antes que instruções aceitas previamente saiam na outra extremidade

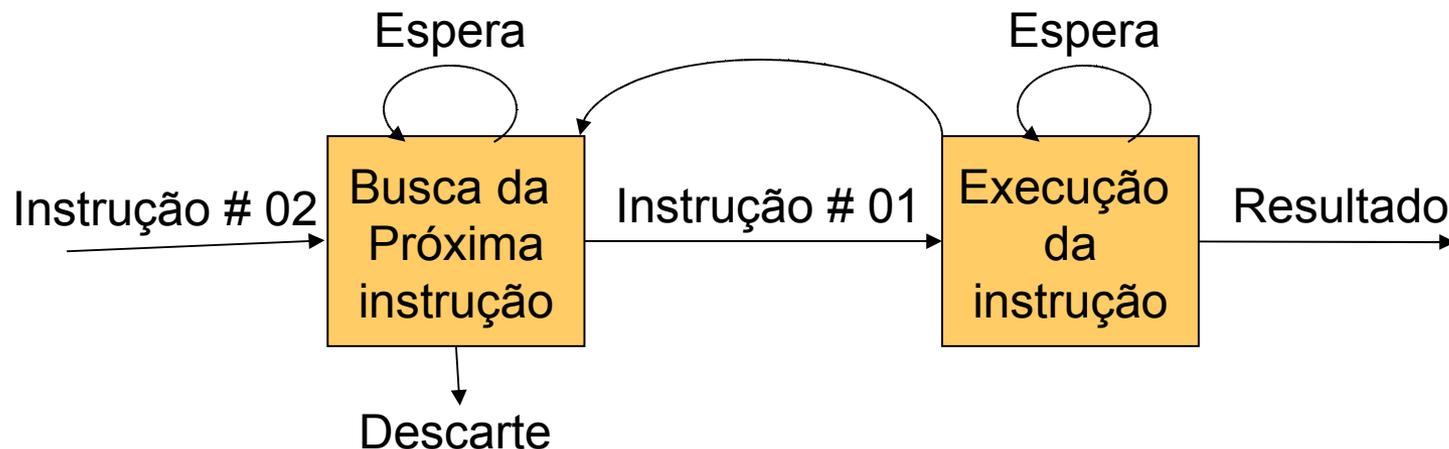
Pipeline de Instruções de Dois Estágios

- Visão idealizada
 - Estágios possuem a mesma duração
 - Eficiência dobrada



Pipeline de Instruções de Dois Estágios

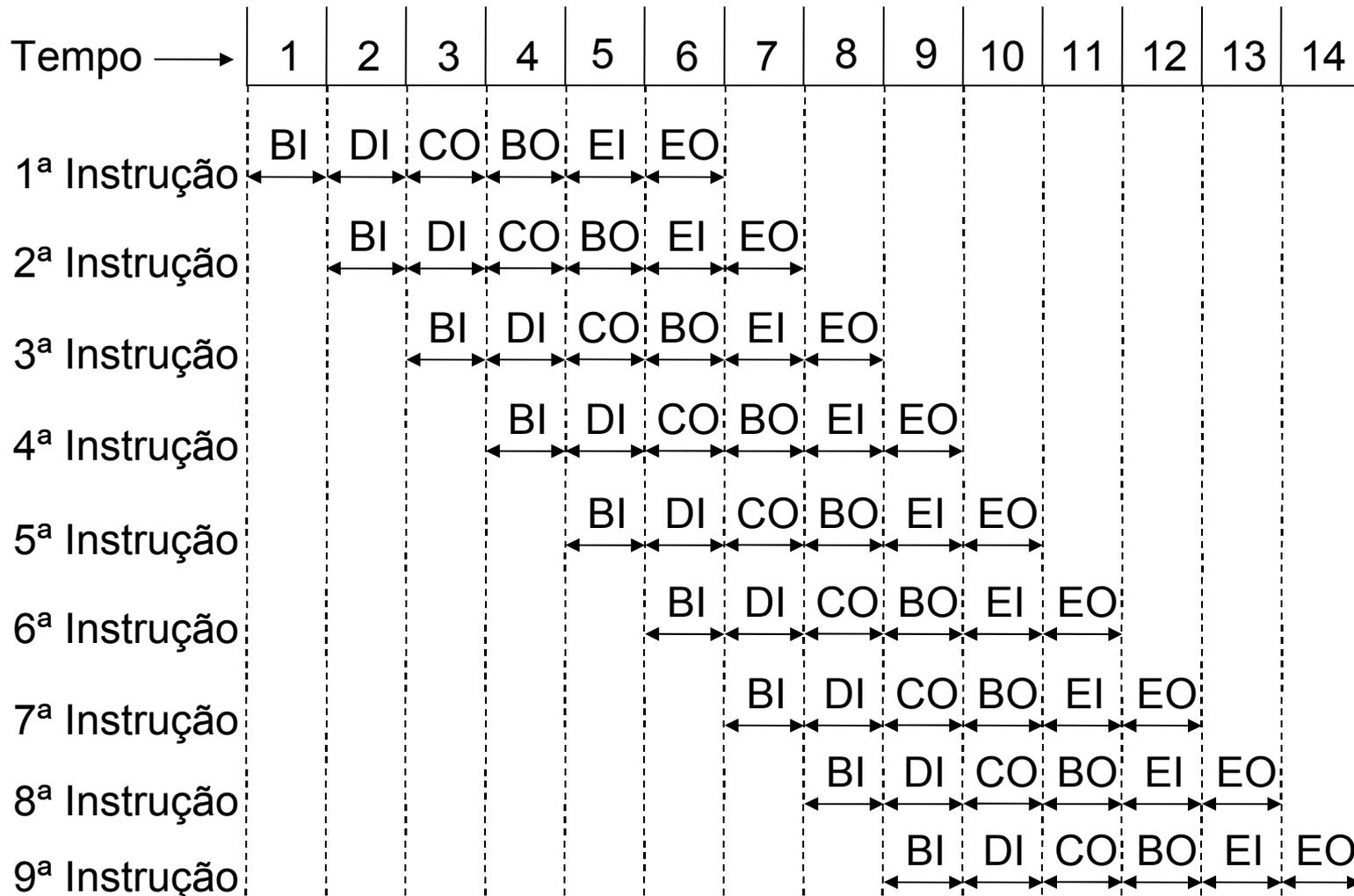
- Visão realista
 - O tempo de execução é maior do que o tempo de busca
 - O estágio de busca espera pelo endereço da próxima instrução em instruções de desvio condicional



Pipeline de Instruções de Seis Estágios

- Para melhor desempenho, é necessário um número maior de estágios
- Decomposição do processamento:
 - BI (Busca da Instrução)
 - DI (Decodificação da Instrução)
 - CO (Cálculo dos Operandos)
 - BO (Busca dos Operandos)
 - EI (Execução da Instrução)
 - EO (Escrita dos Operandos)

Pipeline de Instruções de Seis Estágios



Pipeline de Instruções de Seis Estágios

