

MODOS DE ENDEREÇAMENTO E CONJUNTO DE INSTRUÇÕES

Alexandre Lucas Chichosz

Graduando em Engenharia da Computação,
Faculdades Integradas de Três Lagoas – FITL/AEMS

Calwann de Souza Freire

Graduando em Engenharia da Computação,
Faculdades Integradas de Três Lagoas – FITL/AEMS

Myke Albuquerque Pinto de Oliveira

Graduando em Engenharia da Computação,
Faculdades Integradas de Três Lagoas – FITL/AEMS

Oseias Marques Silva

Graduando em Engenharia da Computação,
Faculdades Integradas de Três Lagoas – FITL/AEMS

André Aparecido Leal de Almeida

Docente das Faculdades Integradas de Três Lagoas – FITL/AEMS

Alan Pinheiro de Souza

Mestre em Informática – UFRJ;
Docente das Faculdades Integradas de Três Lagoas – FITL/AEMS

RESUMO

Este artigo científico enuncia detalhadamente como as instruções lógicas da Unidade Central de Processamento interagem com os modos de endereçamento de memória e como são utilizados os campos operandos de uma instrução de máquina. Serão abordados os modos de endereçamento desenvolvidos para processadores, sendo eles: imediato, direto, indireto, por registrador e deslocamento, além de apresentar o desenvolvimento, demonstrando o seu funcionamento, concretizado por exemplos de instruções em linguagem de baixo nível. Por fim, será apresentado um quadro contendo as vantagens e desvantagens de cada tipo de modo de endereçamento.

PALAVRAS-CHAVE: endereçamento de memória; linguagem *Assembly*; instruções de máquina; operadores e operandos.

INTRODUÇÃO

Uma instrução de máquina é um comando para uma operação básica da unidade central de processamento (UCP). Na visão de Stallings (2003), uma instrução deve especificar o código da operação que será realizada, a referência dos operandos fonte e destino e o endereço da próxima instrução. Por exemplo, uma instrução para somar os números 5 e 8 deve informar o código de operação (C. Op.), nesse caso é um código numérico padronizado para a operação de soma, os

operandos, sendo eles 5 e 8, e o local onde será guardado o resultado da operação que pode ser um registrador da UCP ou uma célula da memória principal (MP).

Segundo Stallings (2003), “na maioria dos casos, a próxima instrução é a que segue imediatamente a instrução corrente. Nesses casos, a instrução não inclui uma referência explícita para a próxima instrução”. A instrução de máquina indica em seu campo operando a localização do referido dado, havendo vários modos de endereçamento para efetuar essa busca pelos dados, podendo ser explícita ou implícita. No caso de referência explícita, existem seis principais modos de endereçamento: (i) imediato; (ii) direto; (iii) indireto; (iv) por registrador; (v) indireto via registrador; (vi) por deslocamento.

Este artigo está dividido em sete seções, sendo a primeira referente a conjunto de instruções da UCP. A segunda seção trata do modo de endereçamento imediato. A terceira abrange o modo de endereçamento direto. Na quarta seção será abordado o modo de endereçamento indireto. A quinta seção refere-se ao modo de endereçamento de registrador. A sexta seção detalha o modo de endereçamento indireto via registrador. A sétima seção explica o modo de endereçamento por deslocamento e seus usos: endereçamento relativo, endereçamento baseado em registrador e indexação. Da segunda a sétima seção, há exemplos de código em Linguagem *Assembly* (BERTOGNA, 2015), com base no microcontrolador 8051. Por último, serão apresentadas as considerações finais que condensam as vantagens e as desvantagens de cada modo de endereçamento.

2 CONJUNTO DE INSTRUÇÕES

O processador do computador funciona seguindo ordens simples e básicas através de instruções, as quais podem efetuar a soma de dois números, mover um dado de um local para o outro ou incrementar valor a um número. Para realizar tarefas mais complexas, a UCP utiliza um conjunto de instruções.

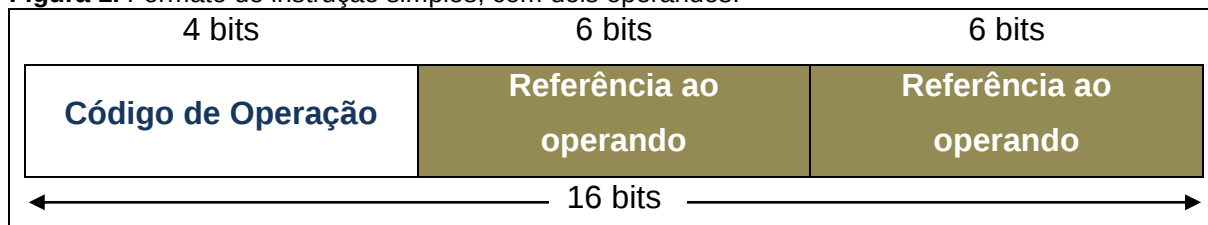
Essas ordens são transmitidas ao *hardware* para serem interpretadas e executadas por meio de sinais elétricos que podem representar o bit 0 ou bit 1, de um conjunto de bits, que por sua vez é denominado instrução de máquina (MONTEIRO, 2012).

A instrução de máquina possui, basicamente, dois campos, sendo cada um deles formados por uma determinada quantidade de bits, que varia de acordo com a

arquitetura e o processador, sendo eles: (i) Código de Operação (C. Op.): define o que fazer e como fazer ao processador; (ii) Operando: indica ao processador qual dado será manipulado.

Um exemplo simples de formato de instrução é mostrado na Figura 1.

Figura 1. Formato de instrução simples, com dois operandos.



Fonte: Adaptado de Monteiro, 2012.

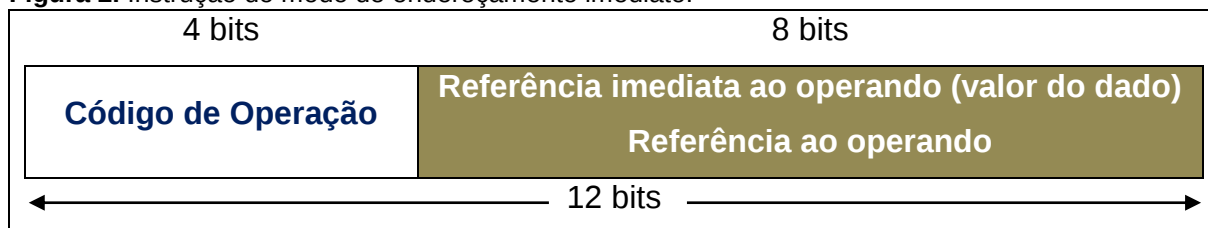
3 ENDEREÇAMENTO IMEDIATO

O método mais simples e rápido de obter um dado é indicar seu próprio valor no campo operando da instrução, em vez de buscá-lo na memória, assim é o modo de endereçamento imediato (MONTEIRO, 2012).

Segundo Stallings (2003), a principal vantagem deste método é economizar acesso a memória principal para obter o operando. Entretanto uma das desvantagens, de acordo com Monteiro (2012), consiste na limitação do tamanho do dado, pois o mesmo é limitado ao tamanho do campo do operando da instrução, assim apresentado restrições ao tamanho do dado referenciado. Utilizando como exemplo o formato de instrução da Figura 2, se o endereçamento fosse imediato, o dado estaria limitado a 8 bits de tamanho, podendo representar um inteiro entre -128 e 127 (com sinal), ou entre 0 e 255 (sem sinal).

Conforme Bertogna (2015), o exemplo a seguir de instrução em linguagem *Assembly*, utiliza o modo de endereçamento imediato para mover o valor 25_{16} para o registrador A: `MOV A, #25H`.

Figura 2. Instrução do modo de endereçamento imediato.



Fonte: Adaptado de Stallings, 2003.

4 ENDEREÇAMENTO DIRETO

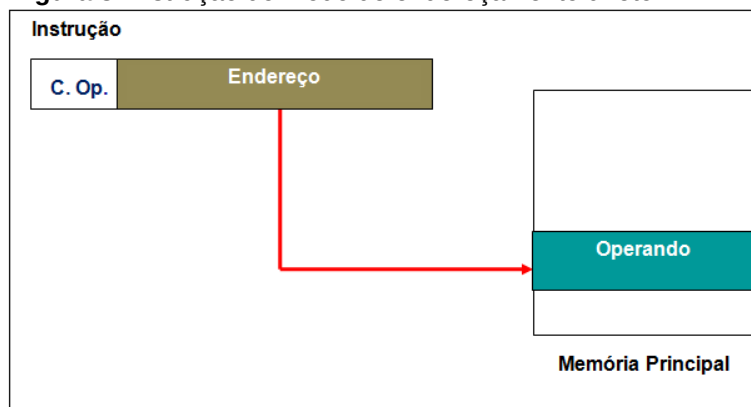
O modo de endereçamento direto, de acordo com Stallings (2003), é um método no qual o campo de endereço da instrução contém o endereço efetivo do operando, que por sua vez encontra-se na memória principal.

Esse endereçamento é também um modo simples de acesso, pois requer apenas uma referência a memória principal para buscar o dado, sendo, porém, mais lento que o modo imediato por fazer referência à memória (MONTEIRO, 2012).

A limitação da memória a ser usada, conforme o campo operando, oferece uma desvantagem a este modo de endereçamento. Apresentando um formato de instrução diferente, agora com campos operandos de seis bits, pode endereçar somente 64 Bytes (endereços de 0_{16} a $3F_{16}$).

Neste tipo de endereçamento, Bertogna (2015) demonstra que os dados são acessados na posição de memória diretamente especificada na instrução, movendo para A o conteúdo da posição de memória 25_{16} . Segue um exemplo de instrução em linguagem *Assembly*: `MOV A, 25H`. Este modo de endereçamento é exemplificado pela ilustração da Figura 3.

Figura 3. Instrução do modo de endereçamento direto.

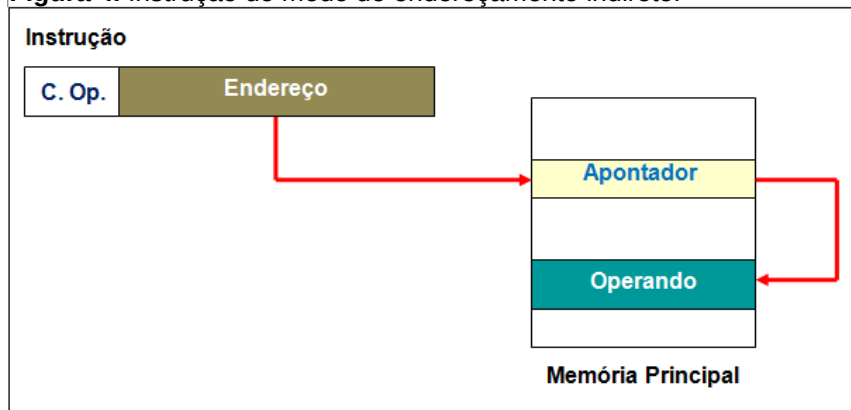


Fonte: Adaptado de Stallings, 2003.

5 ENDEREÇAMENTO INDIRETO

No modo de endereçamento indireto, a instrução apresenta um endereço de posição de memória (similar ao endereçamento direto), entretanto esta posição contém endereço de outra que contém o operando (NULL; LOBUR, 2010). Esse endereço intermediário é chamado de apontador ou ponteiro, conforme está ilustrado na Figura 4.

Figura 4. Instrução do modo de endereçamento indireto.



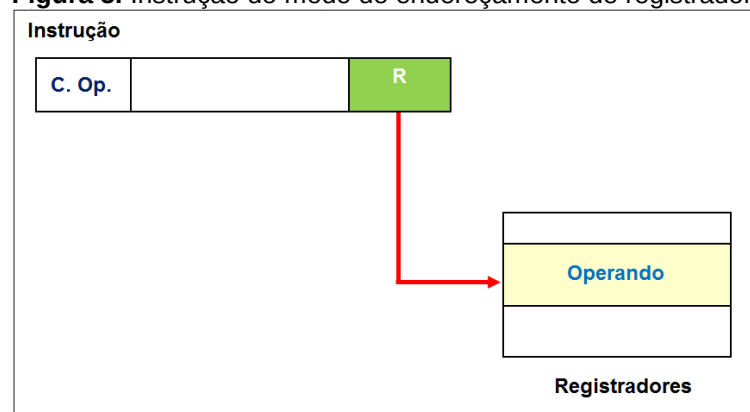
Fonte: Adaptado de Stallings, 2003.

Uma vantagem deste método é que o endereço e a quantidade memória que pode ser gerenciada, já que uma célula da memória principal contém mais bits que o campo operando da instrução, e ainda o endereço pode se estender para outras células de memória principal. A desvantagem que mais se destaca nesse modo de endereçamento é ser mais lenta, por requerer dois ciclos de memória principal, uma para ler o apontador e outra para ler o operando.

6 ENDEREÇAMENTO DE REGISTRADOR

Nesse modo de endereçamento o campo operando contém uma referência a um registrador que contém o operando. Conforme Stallings (2003), o campo de registrador tem de 3 a 4 bits, assim pode referenciar um total de 8 a 16 registradores. As suas vantagens são: possuir um campo pequeno de referência ao registrador e não usa referência de memória principal, conforme a Figura 5.

Figura 5. Instrução do modo de endereçamento de registrador.



Fonte: Adaptado de Stallings, 2006.

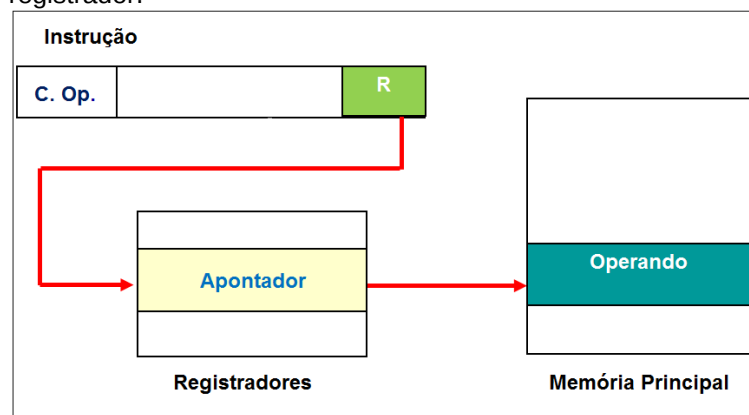
Segundo Tanenbaum (2006), este modo de endereçamento é importante devido aos endereços curtos e muito comum em computadores. Os compiladores, em sua grande maioria, fazem todos os esforços para determinar quais variáveis serão acessadas e as colocam nos registradores.

Uma estratégia para o bom desempenho computacional é manter os valores armazenados nos registradores o máximo de tempo o possível para diferentes operações, evitando chamadas de operandos na memória principal. Uma desvantagem citada por Stallings (2003) é que no endereçamento por registrador o espaço do campo operando é limitado.

7 ENDEREÇAMENTO INDIRETO VIA REGISTRADOR

No endereçamento indireto via registrador, o campo operando se refere a um registrador que contém o endereço do dado na memória principal. Portanto esse modo é análogo ao modo de endereçamento indireto, pois como afirmam Null e Lobur (2010), a única diferença é que o campo de endereço se refere a um registrador e não a posição de memória, com isso as vantagens e desvantagens destes modos são semelhantes. Entretanto, o endereçamento indireto via registrador requer um acesso a menos à memória principal que o endereçamento indireto (STALLINGS, 2003), conforme exemplo da Figura 6.

Figura 6. Instrução do modo de endereçamento indireto via registrador.



Fonte: Adaptado de Stallings, 2003.

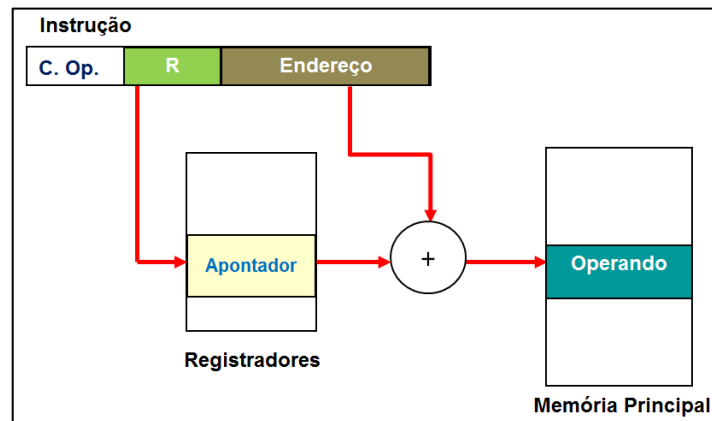
Segue o exemplo de Bertogna (2015), neste tipo de endereçamento o dado é acessado na posição de memória apontada por um dos registros R_0 ou R_1

especificado na instrução, como no exemplo a seguir move para A o conteúdo da posição de memória apontada por R₁: MOV A, @R₁.

8 ENDEREÇAMENTO POR DESLOCAMENTO

Segundo Stallings (2003), o endereçamento por deslocamento é um método muito poderoso porque combina a capacidade do endereçamento direto com o endereçamento indireto via registrador. No modo de endereçamento por deslocamento, o endereçamento do dado é determinado pela soma de dois operandos, conforme é ilustrado no exemplo abaixo pela Figura 7.

Figura 7. Instrução no modo de endereçamento por deslocamento.



Fonte: Adaptado Stallings, 2003.

Recapitulando o que foi dito na primeira seção, os operandos podem ser implícitos baseados no C. Op. Segundo Stallings (2003), há três usos comuns do endereçamento por deslocamento: (i) endereçamento relativo: também conhecido como endereçamento por descolamento ao contador de instrução. Com esse conceito a instrução faz uma referência implícita ao registrador PC (*Program Counter*, em português significa contador de programa). Esse registrador se refere a próxima instrução a ser executada. O valor do campo operando é somado ao registrador PC produzindo o endereço do dado na memória principal. (ii) endereçamento baseado em registrador: nesse conceito de endereçamento o registrador referenciado pela instrução indica um segmento de memória a ser usado e o outro operando indica qual a célula deste segmento a instrução está se referindo. Dessa forma, uma parte da memória principal é vista como um arranjo

ordenado de dados, uma estrutura ou registro, e alterando o valor do registrador de base altera-se o arranjo referido, mas a mesma instrução efetua a operação no novo arranjo. (iii) indexação: o campo de operando da instrução faz referência ao bloco de memória e o registrador indica o descolamento a partir do início do bloco. Em outras palavras, embora o funcionamento do modo de endereçamento seja o mesmo, a maneira que o programador o conceitua é diferente. Essa ideia é ótima para efetuar uma mesma operação sobre um grande conjunto de dados dentro de um laço, usando um registrador para controlar o número de iterações, a mesma instrução vai se referir a dados que estão na sequência na memória principal.

Segundo Null e Lobur (2010), a vantagem do endereçamento baseado em registrador e indexação é sua utilidade em acessar elementos de *arrays* e caracteres em *strings*. Os registradores usados nesses modos de endereçamento podem ser de propósito geral ou específico de indexação com referência implícita.

9 CONSIDERAÇÕES FINAIS

Quadro 1. Demonstrativo das características dos modos de endereçamento.

Modo de Endereçamento	Definição	Vantagens	Desvantagens
Imediato	O campo operando contém o dado	Nenhuma referência à memória principal	Limitada magnitude do operando
Direto	O campo operando contém o endereço do dado	Baixa complexidade	Perda de tempo, se o dado é uma constante
Indireto	O campo operando contém o endereço da posição de memória que contém o endereço definitivo do dado	Espaço de endereçamento grande	Muitos acessos à memória principal para execução
De Registrador	O campo operando contém endereço de um registrador	Nenhuma referência à memória principal	Espaço de endereçamento limitado
Indireto Via Registrador	O campo operando contém um endereço de um registrador e esse registrador contém o endereço do dado na memória	Requer um acesso a menos na memória	Referência extra à memória principal
Por Deslocamento	O endereço do operando é determinado por soma de campo operando e um registrador	Uso ideal para <i>array</i> , <i>string</i> e registros	Alta complexidade

Fonte: Baseado em Monteiro (2012) e Stallings (2003).

Este artigo científico apresentou as premissas para o uso dos modos de endereçamento de memória por intermédio de um levantamento bibliográfico, com o

foco em apenas seis principais modos de endereçamento, sendo que não foram discutidas abordagens mais específicas como o modo por pilha. Foram destacados também conjuntos de instruções associados aos processadores. Para trabalhos futuros, podem-se aprofundar os conceitos de endereçamento de memória, visando aplicação dessas abordagens em arquiteturas específicas de microcontroladores.

O Quadro 1 apresenta uma comparação entre os seis principais modos de endereçamento, servindo como um breve resumo da pesquisa. Esse esquema evidencia uma das principais contribuições do levantamento bibliográfico que foi apontar definição, vantagens e desvantagens para cada um dos modos de endereçamento.

REFERÊNCIAS

BERTOIGNA, Eduardo. Microcontroladores 8051: Teoria e Prática. Curitiba: Editora do Autor, 2015.

LEITE, Mário. Técnicas de Programação: Uma Abordagem Moderna. Rio de Janeiro: Brasport, 2006.

MONTEIRO, Mario. Introdução à Organização de Computadores. 5. ed., Rio de Janeiro: LTC, 2012.

NULL, Linda; LOBUR, Julia. Princípios Básicos de Arquitetura e Organização de Computadores. 2. ed., Porto Alegre: Artmed, 2010.

STALLINGS, William. Arquitetura e Organização de Computadores. 5. ed., São Paulo: Prentice-Hall, 2003.

TANENBAUM, Andrew. Organização Estruturada de Computadores. 5. ed., Rio de Janeiro: Prentice-Hall, 2006.