

Processador RAMSES

Prof. Sérgio Luis Cechin

RAMSES

Compatibilidade

- Tudo o que se faz com o RAMSES pode ser feito no NEANDER...
- ...entretanto, o RAMSES é mais eficiente do que o NEANDER
- Todo código escrito para o NEANDER pode rodar no RAMSES...
- ...uma vez que, o RAMSES incorpora todos os recursos do NEANDER

Características Gerais

- Incorpora os recursos do NEANDER
- Acrescenta
 - Modos de endereçamento
 - Registradores
 - Registros de estado
 - Instruções
- É capaz de executar qualquer programa construído para o NEANDER
 - É “código-compatível”

Características específicas (1)

- Largura dos dados e endereços: 8 bits
 - Tamanho da memória: 256 bytes
- Representação dos dados em complemento de 2
 - Afeta a forma com a ULA efetua seus cálculos
- Registradores de uso geral (8 bits)
 - A, equivalente ao encontrado no NEANDER
 - B

Características específicas (2)

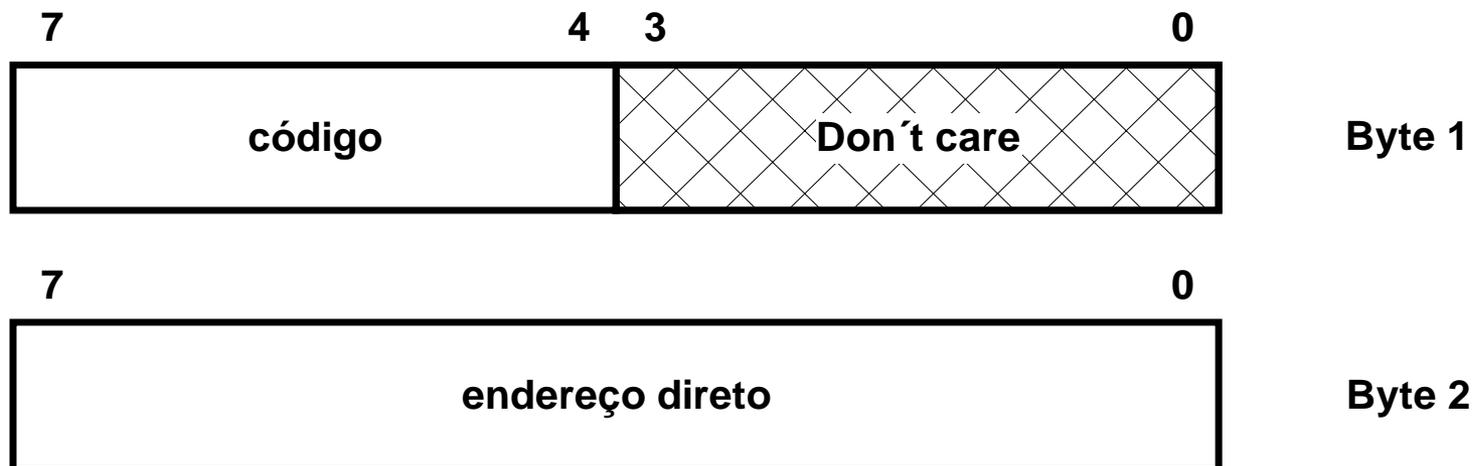
- Registradores específicos (8 bits)
 - Registrador X
 - 8 bits
 - Usado como índice no acesso à memória
 - Pode ser usado como registrador de uso geral
 - Registrador PC
 - 8 bits
 - Apontador de programa
 - Equivalente ao do NEANDER

Características específicas (3)

- Registrador de estado
 - Código de condição N (1 bit)
 - Se 1, indica valores negativos (complemento de 2)
 - Equivalente ao do NEANDER
 - Código de condição Z (1 bit)
 - Se 1, indica valor zero
 - Equivalente ao do NEANDER
 - Código de condição C (1 bit)
 - Se 1, indica a ocorrência de vai-um (números sem sinal)

Formato das Instruções (1)

- Formato das instruções NEANDER
- Observar a parte *don't care*



Formato das Instruções (2)

- *A parte don't care será usada para*
 - Indicar o registro (A, B ou X), nos bits 2 e 3
 - 00, 01 ou 10, respectivamente
 - Indicar o modo de endereçamento, nos bits 0 e 1

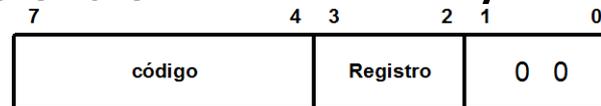


Modos de Endereçamento

- São quatro

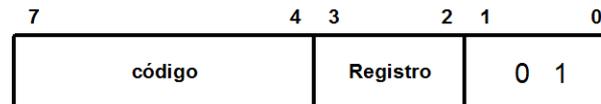
- Modo direto (igual ao do NEANDER)

- Código = 00



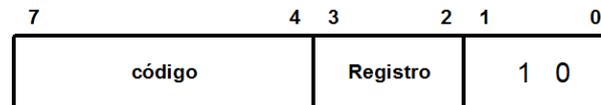
- Modo indireto

- Código = 01



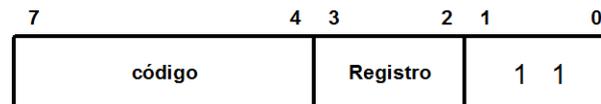
- Modo imediato

- Código = 10



- Modo indexado

- Código = 11



Modos de Endereçamento

Modo Direto

- Único modo compatível com o NEANDER
- O endereço do operando está no segundo byte da instrução
- **Operando** = MEM (segundo byte da instrução)
- Operação:

- Busca de instrução (Fetch)
 - $RI \leftarrow MEM(PC)$
 - $PC \leftarrow PC + 1$
- Acesso no modo direto
 - Operando $\leftarrow MEM (MEM (PC))$

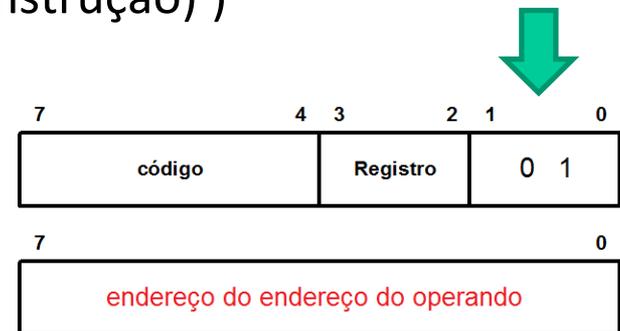


- Exemplo

- Carregar “A”, usando modo **direto**
 - Considere que a instrução ocupa os endereços H03 e H04
 - Considere que o segundo byte da instrução é H93
 - Considere que o endereço H93 contém HA5

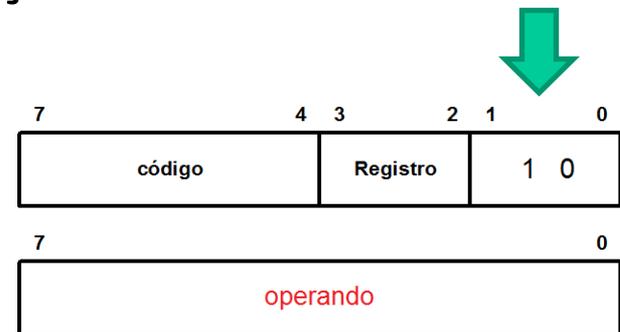
Modo Indireto

- Usado na carga de vetores (substitui alteração de código)
- O endereço do endereço do operando está no segundo byte da instrução
- **Operando** = MEM (MEM (segundo byte da instrução))
- Operação:
 - Busca de instrução (Fetch)
 - $RI \leftarrow MEM(PC)$
 - $PC \leftarrow PC + 1$
 - Acesso no modo indireto
 - Operando $\leftarrow MEM (MEM (MEM (PC)))$
- Exemplo
 - Carregar “A”, usando modo **indireto**
 - Considere que a instrução ocupa os endereços H03 e H04
 - Considere que o segundo byte da instrução é H81
 - Considere que o endereço H81 contém H93
 - Considere que o endereço H93 contém HA5



Modo Imediato

- Usado na carga de constantes
- O operando está no segundo byte da instrução
- **Operando** = segundo byte da instrução
- Operação:
 - Busca de instrução (Fetch)
 - $RI \leftarrow MEM(PC)$
 - $PC \leftarrow PC + 1$
 - Acesso no modo imediato
 - $Operando \leftarrow MEM(PC)$
- Exemplo
 - Carregar “A”, usando o modo **imediato**
 - Considere que a instrução ocupa os endereços H03 e H04
 - Considera que o segundo byte da instrução contém HA5



Modo Indexado

- Requer um registro de índice, chamado de RX (ou X)
- O segundo byte da instrução é chamado de **deslocamento**
 - Número com sinal, representado em complemento de 2
- O **endereço do operando** é calculado
 - Soma: $RX + \text{deslocamento}$
- **Operando** = $MEM (X + \text{Segundo byte da instrução})$
- Operação:
 - Busca de instrução (Fetch)
 - $RI \leftarrow MEM(PC)$
 - $PC \leftarrow PC + 1$
 - Acesso no modo indexado
 - Operando $\leftarrow MEM (X + MEM (PC))$
- Exemplo
 - Carregar “A”, usando o modo **indexado**
 - Considere que a instrução ocupa os endereços H03 e H04
 - Considere que $RX = H8D$
 - Considere que o segundo byte da instrução vale H06
 - Considere que o endereço H93 contém HA5



Representação Simbólica

(para uso no Daedalus)

- Modo direto (modo=00)
 - <endereço>
 - Ex: H93 e NSYMB
- Modo indireto (modo=01)
 - <endereço>,I
 - Ex: H93,I e NSYMB,I
- Modo imediato (modo=10)
 - #<valor>
 - Ex: #H93 e #NSYMB
- Modo indexado (modo=11)
 - <endereço>,X
 - Ex: H93,X e NSYMB,X

RAMSES

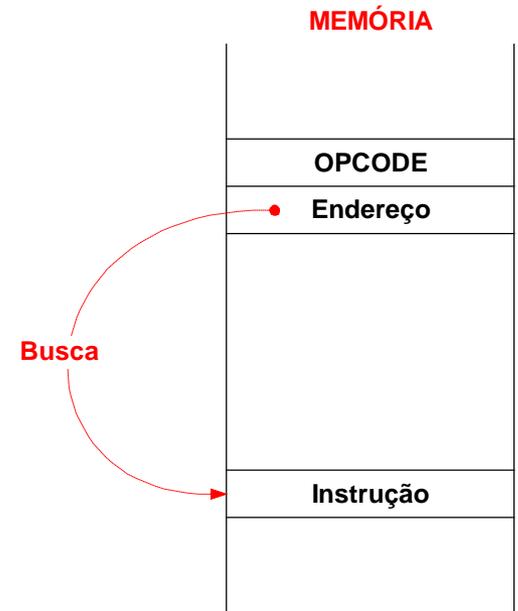
Modos de Endereçamento
... e os desvios (jumps)

Modos de Endereçamento e os Desvios

- Instruções de transferência (*storage e load*)
 - O **operando** é um **dado**
 - O operando é usado durante a execução
- Instruções de desvio
 - O **operando** é uma **instrução**
 - O desvio indica qual a próxima instrução a ser executada
 - Isso é feito alterando-se o valor do PC
 - O operando é usado no próximo ciclo de busca
 - O operando é lido no próximo ciclo de busca
- Conclusão
 - O endereçamento nas instruções de desvio devem ter **uma leitura a menos**, durante a execução
 - Esta leitura será realizada no próximo ciclo de busca

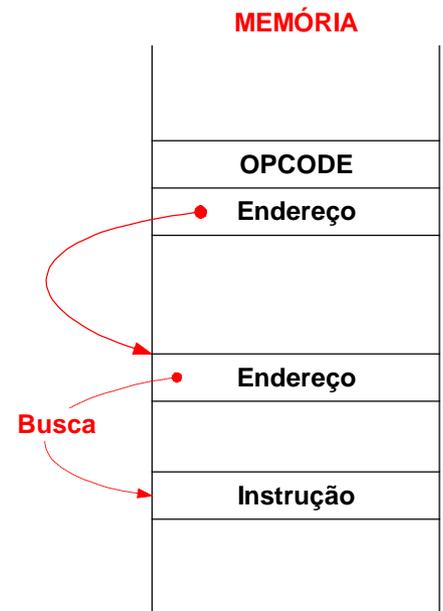
Modo Direto e os Desvios

- Instruções de transferência
 - Operando = **MEM (MEM (PC))**
- Instruções de desvio
 - O destino, no desvio, é sempre o PC
 - $PC \leftarrow MEM (PC)$
 - No próximo ciclo de busca
 - O destino do ciclo de busca é o RI
 - $RI \leftarrow MEM(PC)$
 - **$RI \leftarrow MEM (MEM (PC))$**



Modo Indireto e os Desvios

- Instruções de transferência
 - Operando = $MEM (MEM (MEM (PC)))$
- Instruções de desvio
 - O destino é sempre o PC
 - $PC \leftarrow MEM (MEM (PC))$
 - No próximo ciclo de busca
 - O destino é o RI
 - $RI \leftarrow MEM(PC)$
 - $RI \leftarrow MEM (MEM (MEM (PC)))$

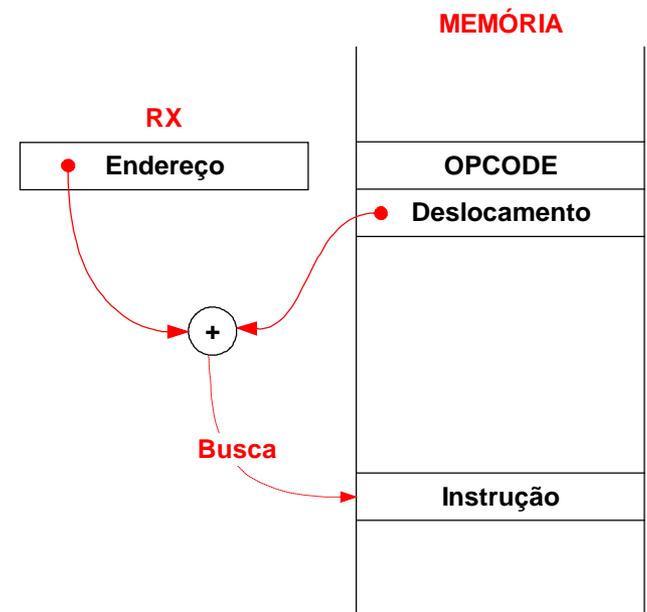


Modo Imediato e os Desvios

- Modo ilegal para os desvios (= NOP)
- Instruções de transferência
 - Operando = MEM (PC)
- Instruções de desvio
 - O destino é sempre o PC
 - $PC \leftarrow PC$?????

Modo Indexado e os Desvios

- Instruções de transferência
 - Operando = $MEM(X + MEM(PC))$
- Instruções de desvio
 - O **destino** é sempre o PC
 - $PC \leftarrow X + MEM(PC)$
 - No próximo ciclo de busca
 - O destino é o RI
 - $RI \leftarrow MEM(PC)$
 - $RI \leftarrow MEM(X + MEM(PC))$



Conclusão

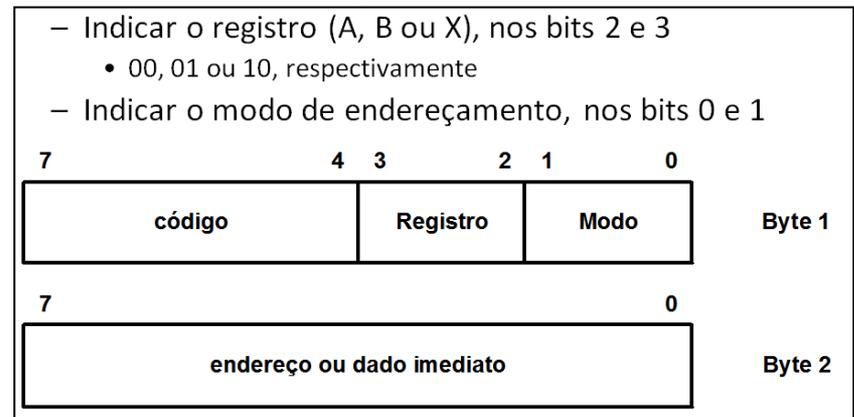
- O endereçamento nas instruções de desvio devem ter **uma leitura a menos**, durante a execução
- Esta leitura será realizada no próximo ciclo de busca

Exercícios

1

- Codificar as instruções abaixo. Cada uma delas está localizada a partir do endereço H00.

– 00 LDR A,H80
– 00 LDR B,H80,X
– 00 LDR X,H80,I
– 00 LDR X,#H80

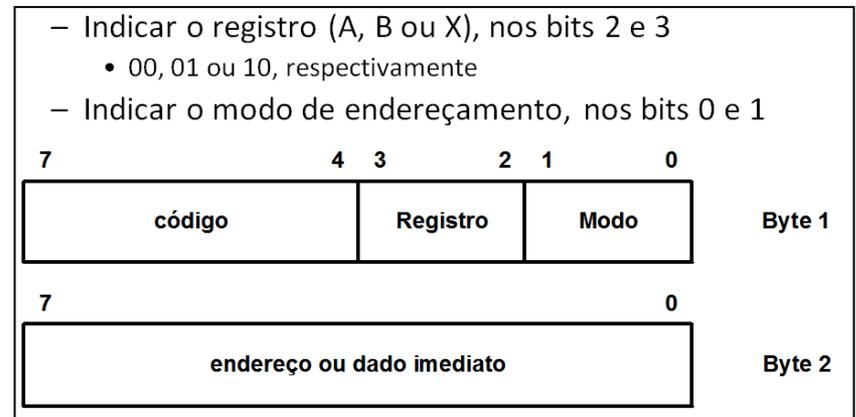


- Conferir com o uso do Daedalus ou com o Simulador

2

- Codificar as instruções abaixo. Cada uma delas está localizada a partir do endereço H00.

– 00 JMP H80
– 00 JMP H80,X
– 00 JMP H80,I



Processador RAMSES

Prof. Sérgio Luis Cechin