

Computador Hipotético Ahmes

Disciplina: Introdução à Arquitetura de Computadores

Luciano Moraes Da Luz Brum

Universidade Federal do Pampa – Unipampa – Campus Bagé

Email: lucianobrum18@gmail.com

- Largura de dados e endereços de 8 bits;
- Dados representados em complemento de dois;
- 1 acumulador de 8 bits (AC);
- 1 contador de programa de 8 bits (PC);
- 1 registrador de estado com 5 códigos de condição: **negativo** (N), **zero** (Z), **carry out** (vai-um) (C), **borrow out** (empresta-um) (B) e **overflow** (V);

- Modos de endereçamento: Direto.

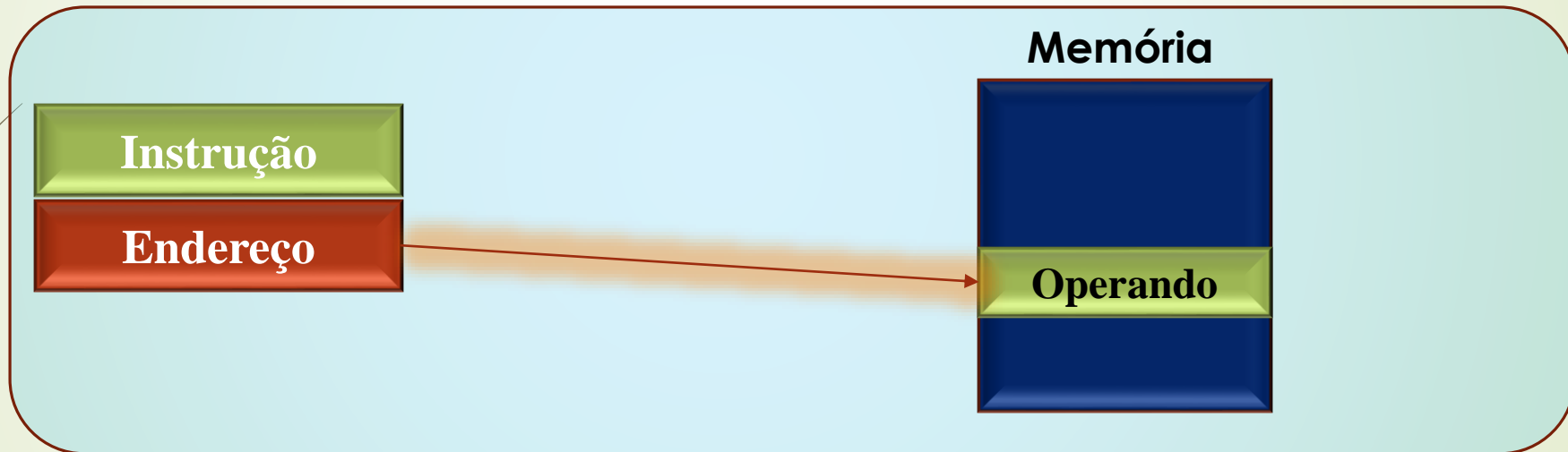


Figura 1: Endereçamento Direto. Fonte: Adaptado de Weber, 2001.

- Obs: em instruções de desvio, o endereço contido na instrução é a posição de memória onde está a instrução a ser executada.

Código Binário (relevante)	Código Binário	Código Decimal	Instrução (Mnemônico)
0000 xxxx	0000 0000	0	NOP
0001 xxxx	0001 0000	16	STA end
0010 xxxx	0010 0000	31	LDA end
0011 xxxx	0011 0000	48	ADD end
0100 xxxx	0100 0000	64	OR end
0101 xxxx	0101 0000	80	AND end
0110 xxxx	0110 0000	96	NOT
0111xxxx	0111 0000	112	SUB end
1000 xxxx	1000 0000	128	JMP end
1001 00xx	1001 0000	144	JN end
1001 01xx	1001 0100	148	JP end
1001 10xx	1001 1000	152	JV end
1001 11xx	1001 1100	156	JNV end


Tabela 1. Conjunto de instruções do Píndes. Fonte: Weber, 2001.

Código Binário (Relevante)	Código Binário	Código Decimal	Instrução (Mnemônico)
1010 00xx	1010 0000	160	JZ end
1010 01xx	1010 0100	164	JNZ end
1011 00xx	1011 0000	176	JC end
1011 01xx	1011 0100	180	JNC end
1011 10xx	1011 1000	184	JB end
1011 11xx	1011 1100	188	JNB end
1110 xx00	1110 0000	224	SHR
1110 xx01	1110 0001	225	SHL
1110 xx10	1110 0010	226	ROR
1110 xx11	1110 0011	227	ROL
1111 xxxx	1111 0000	240	HLT

Tabela 2: Conjunto de instruções de 7 bits. Fonte: Weber, 2001.

Instrução	Execução	Comentário
NOP	Nenhuma operação	Nenhuma operação
STA end	MEM(end) <- AC	Armazena AC na memória
LDA end	AC <- MEM(end)	Carrega AC da memória
ADD end	AC <- AC + MEM(end)	Soma
OR end	AC <- AC or MEM(end)	“ou” lógico
AND end	AC <- AC and MEM(end)	“e” lógico
NOT	AC <- NOT AC	Inverte (complementa) o acumulador
SUB end	AC <- AC - MEM(end)	Subtração
JMP end	PC <- end	Desvio incondicional
JN end	IF N=1 THEN PC <- end	Desvio condicional se negativo
JP end	IF N=0 THEN PC <- end	Desvio condicional se positivo
JV end	IF V=1 THEN PC <- end	Desvio condicional se houve estouro
JNV end	IF V=0 THEN PC <- end	Desvio condicional se não houve estouro


Tabela 3: Ações das instruções. Fonte: Weber, 2001.



unipampa
Universidade Federal do Pampa

Características
Modo de endereçamento
Conjunto de Instruções
Códigos de Condição

Formato das instruções
Exercícios



ENGENHARIA DE
COMPUTAÇÃO

Instrução	Execução	Comentário
JZ end	IF Z=1 THEN PC <- end	Desvio condicional se resultado é zero
JNZ end	IF Z=0 THEN PC <- end	Desvio condicional se resultado não é zero
JC end	IF C=1 THEN PC <- end	Desvio condicional se houve carry
JNC end	IF C=0 THEN PC <- end	Desvio condicional se não houve carry
JB end	IF B=1 THEN PC <- end	Desvio condicional se houve borrow
JNB end	IF B=0 THEN PC <- end	Desvio condicional se não houve borrow
SHR	IF V=1 THEN PC <- end	Deslocamento para a direita (shift right)
SHL	IF V=0 THEN PC <- end	Deslocamento para a esquerda (shift left)
ROR	IF N=1 THEN PC <- end	Rotação para a direita (rotate right)
ROL	IF N=1 THEN PC <- end	Rotação para a esquerda (rotate left)
HLT	IF N=0 THEN PC <- end	Término de execução do programa

Tabela 4: Ações das instruções. Fonte: Weber, 2001.

➤ Códigos de condição:

- Utilizando pelas instruções JN, JZ, JC, JV e JB;
- N - negativo: sinal do resultado (1 se é negativo, 0 se é positivo ou nulo);
- Z – zero: indica se o resultado é zero (1 se é igual a 0, 0 se é diferente de 0);
- C - carry: indica se ocorreu carry (1 se ocorreu, 0 se não ocorreu);
- V – overflow: indica se ocorreu overflow (1 se ocorreu, 0 se não ocorreu);
- B – borrow: indica se ocorreu borrow (1 se ocorreu, 0 se não ocorreu);
- Apenas as instruções lógicas, aritméticas, de deslocamento e de transferência (LDA) afetam os códigos de condição. As instruções de desvio testam os códigos mas não os alteram;

Formato das instruções



Figura 2: Formato das Instruções no Ahmes. Fonte: Adaptado de Weber, 2001.

- As instruções têm 2 bytes: o 1º byte contém o código da instrução e o 2º byte contém um endereço;
- Instruções de 2 bytes são as que fazem referência a memória;
- **NOP, HLT, NOT, SHR, SHL, ROL, ROR?**

Formato das instruções



Figura 2: Formato das Instruções no Ahmes. Fonte: Adaptado de Weber, 2001.

- As instruções têm 2 bytes: o 1º byte contém o código da instrução e o 2º byte contém um endereço;
- Instruções de 2 bytes são as que fazem referência a memória;
- **NOP, HLT, NOT, SHR, SHL, ROL, ROR? Instruções de 1 byte !**

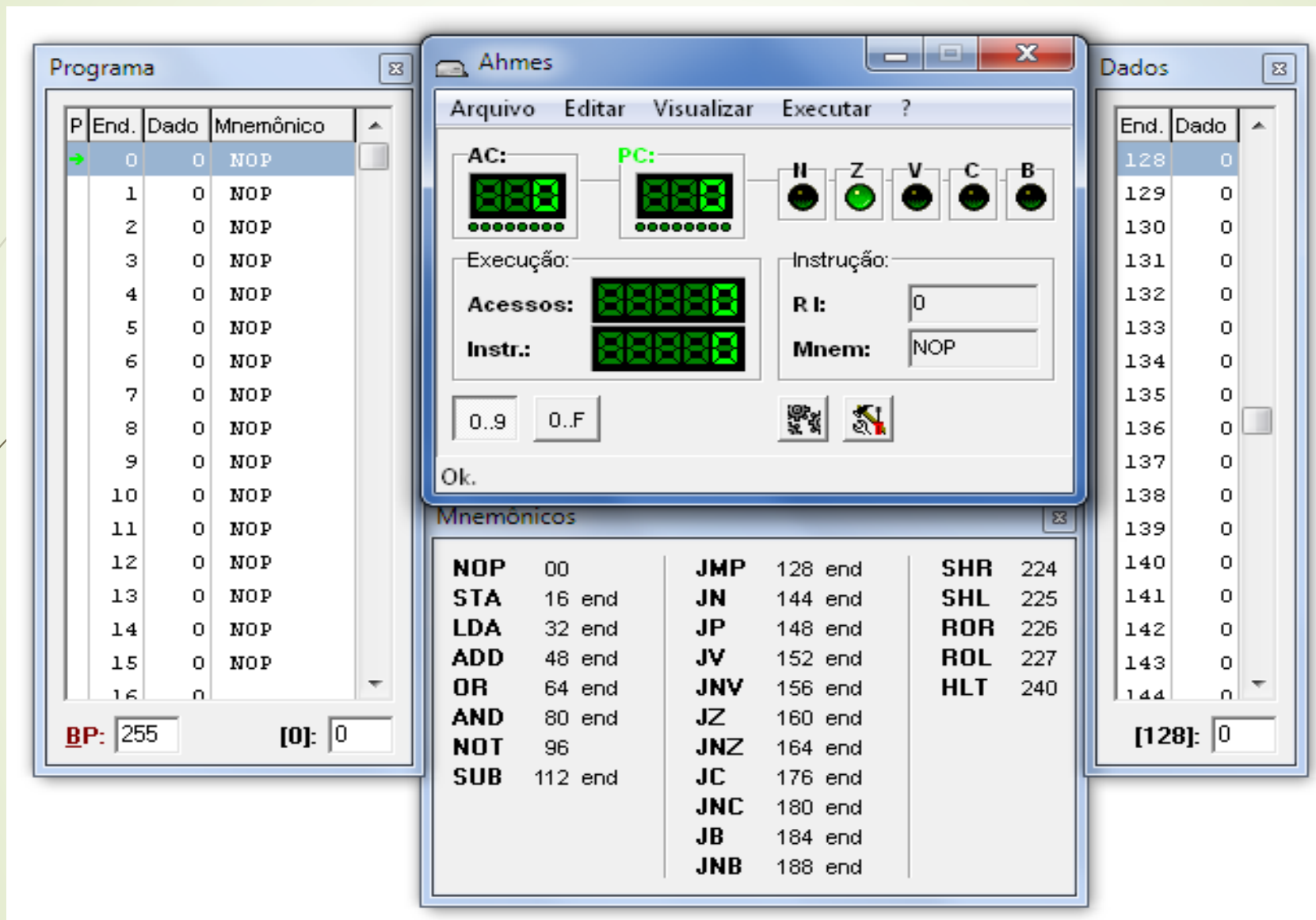


Figura 3: Simulador Ahmes. Fonte: Elaborada pelo autor.

Exercícios

1. **Construa um programa que multiplique por 4 o conteúdo da posição 128 de memória e armazene o resultado na posição 129.**
2. **Construa um programa que Divida por 2 o conteúdo da posição 128 de memória e armazene o resultado na posição 129.**
3. **Construa um programa que faça a multiplicação de dois números positivos em complemento de B.**

Referências Bibliográficas:

- Canal de Ensino do prof. Dr. Sandro Camargo:
<https://www.youtube.com/user/scamargo10/videos>
- Leitura dos capítulos 4 e 6 do livro Fundamentos de Arquitetura de Computadores (Raul Fernando Weber).



Dúvidas ?