CPU – Conjunto de instruções

Alunos:
Cláudio André Rocha Alvares de Oliveira
Luís Henrique Nunes da Silva
Gustavo José Pimentel Brasileiro

Organização e Arquitetura de Computadores

2021.1

Características de instrução de máquina

1. **Elementos de instrução de máquina;**
1. **Representação de instrução;**
1. **Tipos de instrução;**
1. **Número de endereços;**
1. **Projeto do conjunto de instrução.**

Características de instrução de máquina

UNIVERSIDADE DE PERNAMBUCO

1. Elementos de instrução de máquina

- Código de operação (opcode);
- Referência ao operando-fonte;
- Referência ao operando-destino;
- Referência da próxima instrução.

POLI
ESCOLA POLITÉCNICA DE PERNAMBUCO

Características de instrução de máquina

1. Elementos de instrução de máquina

- Memórias principal;
- Registradores da CPU;
- Dispositivos de entrada e saída (E/S).

Características de instrução de máquina

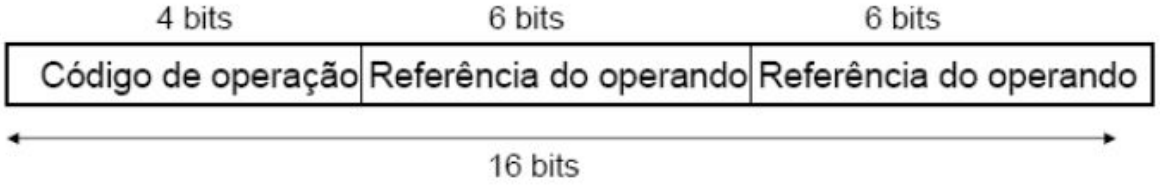
2. Representação de instrução

- Cada instrução é representada como uma sequência de bits

  - Cada código de máquina tem um padrão único de bit;
  - Para melhor compreensão dos programadores é utilizada um representação simbólica.

- Instruções são divididas em campos, a partir dos seus elementos, ou seja, formato da instrução.

- Exemplo do formato de instrução



Características de instrução de máquina

2. Representação de instrução

- Mnemônicos

- ADD – Adição;
- SUB – Subtração;
- MUL – Multiplicação;
- LOAD – Carregar dados da memória;
- STORE – Armazenar dados da memória

Características de instrução de máquina

UNIVERSIDADE DE PERNAMBUCO

3. Tipo de instrução

- Conjunto de instruções deve permitir formular qualquer tarefa de processamento

- Instruções de máquina podem ser agrupadas em:

- Processamento de dados;
- Armazenamento de dados;
- Movimentação de dados;
- Controle do fluxo de dados.

POLI
ESCOLA POLITÉCNICA DE PERNAMBUCO

Características de instrução de máquina

UNIVERSIDADE DE PERNAMBUCO

4. Número de endereços

- Considerando os elementos de uma instrução, pode-se haver até 4 endereços em uma operação

  - Máximo de 2 endereços dos operando de entrada;
  - 1 endereço do operando de saída;
  - 1 endereço da próxima instrução.

- Na prática, tem-se instruções com 1, 2 ou 3 endereços

  - Endereço da próxima instrução está implícito.

POLI ESCOLA POLITÉCNICA DE PERNAMBUCO

Características de instrução de máquina

UNIVERSIDADE DE PERNAMBUCO

4. Número de endereços

- **<u>Instruções com 3 endereços:</u>**

 Específica endereços para 2 operandos e para o resultados

 - Exemplo: ADD A, B, C A = B + C

 Não são comuns

Instrução		Comentário
SUB	Y, A, B	Y ← A − B
MPY	T, D, E	T ← D × E
ADD	T, T, C	T ← T + C
DIV	Y, Y, T	Y ← Y ÷ T

(a) Instruções com três endereços

POLI ESCOLA POLITÉCNICA DE PERNAMBUCO

Características de instrução de máquina

UNIVERSIDADE DE PERNAMBUCO

4. Número de endereços

- **<u>Instruções com 2 endereços:</u>**

 Específica endereços para os 2 operandos de entrada

Exemplo: ADD A,B A = A + B

 Vantagem e desvantagens

Instrução		Comentário
MOVE	Y, A	Y ← A
SUB	Y, B	Y ← Y − B
MOVE	T, D	T ← D
MPY	T, E	T ← T × E
ADD	T, C	T ← T + C
DIV	Y, T	Y ← Y ÷ T

(b) Instruções com dois endereços

POLI ESCOLA POLITÉCNICA DE PERNAMBUCO

Características de instrução de máquina

4. Número de endereços

- **<u>Instruções com 1 endereços:</u>**

 Comum nos primeiros computadores

 Especifica apenas o endereço de 1 dos operandos de entrada.

- Endereço implícito para o operando e o resultados

Instrução		Comentário
LOAD	D	AC ← D
MPY	E	AC ← AC × E
ADD	C	AC ← AC + C
STOR	Y	Y ← AC
LOAD	A	AC ← A
SUB	B	AC ← AC − B
DIV	Y	AC ← AC ÷ Y
STOR	Y	Y ← AC

(c) Instruções com um endereço

Características de instrução de máquina

4. Número de endereços

- **<u>Instruções com 0 endereços:</u>**

 Todos os endereços estão implícitos

Exemplo: Operações em registrados ou pilha

Características de instrução de máquina

4. Número de endereços

- Questão importante de projeto, a respeito da quantidade de endereços

 Mais endereços

✔ Instruções maiores e mais complexas;
✔ Mais registradores de propósito geral;
✔ Programas menores, ou seja, menos instruções.

 Menos endereços

✔ Instruções menores e primitivas;
✔ Mais instruções por programa;
✔ Ciclo de instrução mais rápido.

- Máquinas modernos usam 2 ou 3 endereços

Características de instrução de máquina

UNIVERSIDADE DE PERNAMBUCO

5. Projeto do conjunto de instrução

- Afeta diversos aspectos do sistema

✔ Efeito significativo sobre a implementação da CPU;
✔ Deve considerar as necessidades do programador.

- Decisões importantes do projeto:

- Repertório de operações;
- Tipos de dados;
- Formatos de instruções;
- Registradores;
- Endereçamento.

POLI
ESCOLA POLITÉCNICA DE PERNAMBUCO

Tipos de operandos

1. **Números;**

1. **Caracteres;**

1. **Dados lógicos.**

Tipos de operandos

- Instruções de máquina operam sobre dados

- Tipos mais importante:

Endereços

Dados numéricos

✔ Inteiro e ponto flutuante

✔ Decimal (BCD)

Caracteres

✔ Representação por sequências de bits

✔ Exemplo: códigos ASCII e EBCDIC

Dados lógicos

✔ Unidade de N bits correspondente a N itens de dados com valores 0 ou 1

Tipos de dados do Pentium II e dos Power PC

UNIVERSIDADE DE PERNAMBUCO

❖ **Tipos de dados do Pentium**

- Agrupamento de dados

✔ Byte: 8 bits;
✔ Palavra: 16 bits;
✔ Palavra dupla: 32 bits;
✔ Palavra quádrupla: 64 bits.

- Endereçamento é feito por unidades de 8 bits (byte)

✔ Disposição de múltiplos bytes: little-endian.

- Dados não precisam ser alinhados na memórias em endereços divisíveis por 4.

POLI
ESCOLA POLITÉCNICA DE PERNAMBUCO

Tipos de dados do Pentium II e dos Power PC

UNIVERSIDADE DE PERNAMBUCO

- **Tipos de dados do Power PC**

  - Byte sem sinal;
  - Meia palavra sem sinal.

POLI ESCOLA POLITÉCNICA DE PERNAMBUCO

Tipos de operações

- **Os tipos de operações**

1. Operações de transferência de dados;
2. Operações aritméticas;
3. Operações lógicas;
4. Operações de conversão;
5. Operações de entrada/saída (E/S);
6. Operações de controle de sistema;
7. Operações de transferência de controle.

Tipos de operações

1. **Operações de transferência de dados**

- Tipo de operação mais fundamental

- Esse tipo de operação especifica

  - Origem e destino dos dados
  - Tamanho dos dados a serem transferidos;
  - Modo de endereçamento dos operandos.

- Quando envolve endereço de memória

  - Determina o endereço;
  - Converte endereço de memória virtual e real;
  - Verifica memória cache;
  - Inicia escrita na memória.

Tipos de operações

2. Operações aritméticas

- **As operações aritméticas envolve**

- Transferência de dados antes e/ou depois da operação;
- Execução da operação na ULA;
- Atualização dos códigos de condição.

- **Operações básicas**

- Soma
- Subtração
- Multiplicação
- Divisão

- **Outras possíveis operações**

- Tomar o valor absoluto do operando.
- Negar o operando.
- Incrementar o operando de 1.
- Decrementar o operando de 1.

Tipos de operações

3. Operações lógicas

- Manipulam bits individuais de qualquer unidade endereçável
- Operações básicas:

- NOT (NÃO)
- AND (E)
- OR (OU)
- XOR (OU-EXCLUSIVO)
- EQUAL (TESTE DE IGUALDADE BINÁRIA)

Tabela 9.6 Operações lógicas básicas

P	Q	NOT P	P AND Q	P OR Q	P XOR Q	P=Q
0	0	1	0	0	0	1
0	1	1	0	1	1	0
1	0	0	0	1	1	0
1	1	0	1	1	0	1

Tipos de operações

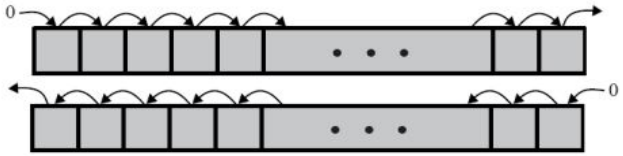
3. Operações lógicas

❖ **Outras operações**

- Deslocamento de bits

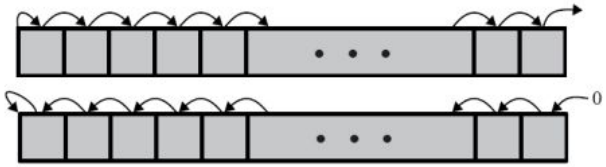
✔ Lógico

Deslocamento Lógico



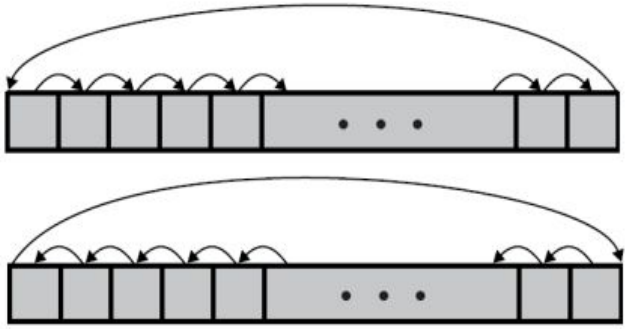
✔ Aritmético

Deslocamento Aritmético



- Rotação

Rotação



Tipos de operações

4. Operações de conversão

- Esse tipo de conversão mudam ou operam sobre o formato de dados
  Exemplo: conversão de um número decimal para binário

5. Operações de entrada/saída (E/S)

- Realizam a transferência de dados com os dispositivos de E/S

- E/S mapeado por memória
- E/S mapeado independentemente

Tipos de operações

6. Operações de controle de sistema

- Instruções privilegiadas

 CPU em estado privilegiado
 CPU executando um programa na área especial

- Para uso pelo Sistema Operacional

7. Operações de transferência de controle

- Altera a sequência de execução das instruções
- Motivo para o desvio

Tipos de operações

7. Operações de transferência de controle

- As operações de transferência de controle encontradas mais frequentemente num Conjunto de instruções são:

- Operações de desvio;
- Operação de salto;
- Operações de chamada de procedimento.

Tipos de operações

7. Operações de transferência de controle

- Instrução de desvio

- Um dos operandos é o endereço da próxima instrução
- Desvio pode ser para frente ou para trás
- Desvio condicional
- Desvio incondicional

Tipos de operações

7. Operações de transferência de controle

- Instrução de desvio

- BRP X – desviará para a instrução de endereço X se o resultado for positivo
- BRN X – desviará para a instrução de endereço X se o resultado for negativo
- BRZ X – desviará para a instrução de endereço X se o resultado for zero
- BRO X – desviará para a instrução de endereço X se ocorrer overflow

Tipos de operações

7. Operações de transferência de controle

- Instrução de salto

- Incluem endereço de desvio implícito

- Área destinada ao endereço pode ser utilizada para outra finalidade
  Exemplo: a respeito de uma variável de controle

Tipos de operações

7. Operações de transferência de controle

- Instrução de chamada de procedimento

- Subrotina ou procedimento: subprograma incorporado em um programa maior

- Uma subrotina pode ser chamada de qualquer ponto do programa
  Chamada: executar subrotina e retornar à instrução seguinte a chamada

- Vantagens da subrotina/procediemento

Tipos de operações do x86

1. Instruções CALL/RETURN
   PUSH EBP
   MOV EBP, ESP
   SUB ESP, space_for_locals
1. Gerenciamento de Memória
2. Flags de estado e códigos de condição

Tabela 12.8

Flags de estado do x86.

Bit de estado	Nome	Descrição
C	*Carry*	Indica a existência do bit de transporte ou empréstimo (*carry bit — vai um*) na posição do bit mais à esquerda após uma operação aritmética. Também modificado por algumas das operações de deslocamento e rotação.
P	Paridade	Paridade do byte menos significativo do resultado de uma operação aritmética ou lógica. 1 indica paridade par; 0 indica paridade ímpar.
A	*Carry* auxiliar	Representa a existência do bit de transporte ou empréstimo (*carry bit — vai um*) na posição entre dois bytes após uma operação aritmética ou lógica de 8 bits. Usado na aritmética BCD.
Z	Zero	Indica que o resultado de uma operação aritmética ou lógica é 0.
S	Sinal	Indica o sinal do resultado de uma operação aritmética ou lógica.
O	*Overflow*	Indica um *overflow* aritmético após uma adição ou subtração em aritmética de complemento de dois.

Tipos de operações do x86

Tabela 12.9

Códigos de condição do x86 para instruções de salto condicional e SETcc.

Símbolo	Condição testada	Comentário
A, NBE	C = 0 AND Z = 0	Acima; Não abaixo ou igual (maior que, sem sinal)
AE, NB, NC	C = 0	Acima ou igual; Não abaixo (maior que ou igual, sem sinal); Sem *carry*
B, NAE, C	C = 1	Abaixo; Não acima ou igual (menor que, sem sinal); *Carry* definido
BE, NA	C = 1 OR Z = 1	Abaixo ou igual; Não acima (menor que ou igual, sem sinal)
E, Z	Z = 1	Igual; Zero (com ou sem sinal)
G, NLE	[(S = 1 AND O = 1) OR (S = 0 AND O = 0)]AND[Z = 0]	Maior que; Não menor que ou igual (com sinal)
GE, NL	(S = 1 AND O = 1) OR (S = 0 AND O = 0)	Maior que ou igual; Não menor que (com sinal)
L, NGE	(S = 1 AND O = 0) OR (S = 0 AND O = 0)	Menor que; Não maior que ou igual (com sinal)
LE, NG	(S = 1 AND O = 0) OR (S = 0 AND O = 1) OR (Z = 1)	Menor que ou igual; Não maior que (com sinal)
NE, NZ	Z = 0	Não igual; Não zero (com ou sem sinal)
NO	O = 0	Sem *overflow*
NS	S = 0	Sem sinal (não negativo)
NP, PO	P = 0	Sem paridade; Paridade ímpar
O	O = 1	*Overflow*
P	P = 1	Paridade; Paridade par
S	S = 1	Sinal (negativo)

Tipos de operações do x86

1. Instruções MMX

Tabela 12.10

Conjunto de instruções MMX.

Categoria	Instrução	Descrição
Aritmética	PADD [B, W, D]	Adição paralela de oito pacotes de bits, quatro palavras de 16 bits ou duas palavras duplas de 32 bits, com *wraparound*.
	PADDS [B, W]	Adição com saturação.
	PADDUS [B, W]	Adição sem sinal com saturação.
	PSUB [B, W, D]	Subtração com *wraparound*.
	PSUBS [B, W]	Subtração com saturação.
	PSUBUS [B, W]	Subtração sem sinal com saturação.
	PMULHW	Multiplicação paralela de quatro palavras de 16 bits com sinal, com 16 bits de alta ordem do resultado de 32 bits escolhidos.
	PMULLW	Multiplicação paralela de quatro palavras de 16 bits com sinal, com 16 bits de baixa ordem do resultado de 32 bits escolhidos.
	PMADDWD	Multiplicação paralela de quatro palavras de 16 bits com sinal; soma pares adjacentes de resultados de 32 bits.
Comparação	PCMPEQ [B, W, D]	Comparação paralela de igualdade; resultado é máscara de 1s se verdadeiro ou 0s se falso.
	PCMPGT [B, W, D]	Comparação paralela de maior que; resultado é máscara de 1s se verdadeiro ou 0s se falso.
Conversão	PACKUSWB	Agrupa palavras em bytes com saturação sem sinal.
	PACKSS [WB, DW]	Agrupa palavras em bytes, ou palavras duplas em palavras, com saturação com sinal.
	PUNPCKH [BW, WD, DQ]	Desagrupa em paralelo (mesclagem intervalada) bytes, palavras ou palavras duplas de alta ordem do registrador MMX.
	PUNPCKL [BW, WD, DQ]	Desagrupa em paralelo (mesclagem intervalada) bytes, palavras ou palavras duplas de baixa ordem do registrador MMX.
Lógica	PAND	AND lógico bit a bit com 64 bits.
	PNDN	AND NOT lógico bit a bit com 64 bits.
	POR	OR lógico bit a bit com 64 bits.
	PXOR	XOR lógico bit a bit com 64 bits.
Deslocamento	PSLL [W, D, Q]	Deslocamento lógico paralelo à esquerda de pacotes de palavra, palavras duplas ou quatro palavras pela quantidade especificada no registrador MMX ou valor imediato.
	PSRL [W, D, Q]	Deslocamento lógico paralelo à direita de pacotes de palavra, palavras duplas ou quatro palavras agrupadas.
	PSRA [W, D]	Deslocamento aritmético paralelo à direita de pacotes de palavra, palavras duplas ou quatro palavras.
Transferência de dados	MOV [D, Q]	Move palavras duplas ou quatro palavras de/para registrador MMX.
Statemgt	EMMS	Esvazia estado MMX (esvazia bits de tag dos registradores FP).

Tipos de operações do x86

- Aritmética de saturação:

F000h = 1111 0000 0000 0000
+ 3000h = 0011 0000 0000 0000
10010 0000 0000 0000 = 2000h

F000h = 1111 0000 0000 0000
+ 3000h = 0011 0000 0000 0000
10010 0000 0000 0000
1111 1111 1111 1111 = FFFFh

Pixel_resultante = Pixel_A × fade + Pixel_B × (1 – fade)

Tipos de operações do ARM

- Instruções Load e store;
- Instruções de desvio;
- Instruções de processamento de dados;
- Instruções de multiplicação;
- Instruções paralelas de adição e subtração;
- Instruções de extensão;
- Instruções de acesso do registrador de estado.

Tipos de operações do ARM

- Códigos de condição
  - 4 flags (N, Z, C e V)

Tabela 12.11

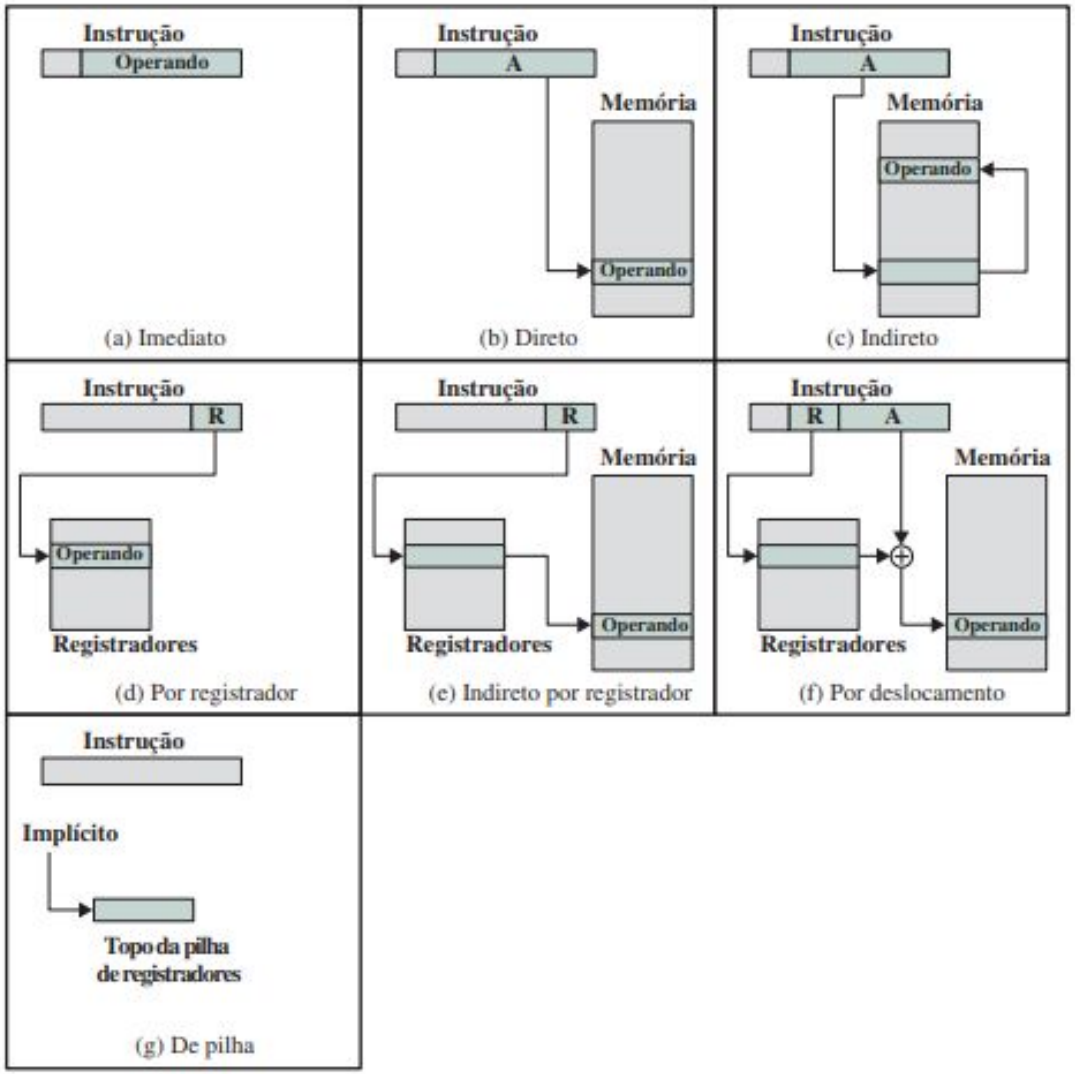
Condições do ARM para execução de instrução condicional.

Código	Símbolo	Condição testada	Comentário
0000	EQ	Z = 1	Igual
0001	NE	Z = 0	Não igual
0010	CS/HS	C = 1	*Carry* em um/acima ou igual sem sinal
0011	CC/LO	C = 0	*Carry* zerado/abaixo sem sinal
0100	MI	N = 1	Menos/negativo
0101	PL	N = 0	Mais/positivo ou zero
0110	VS	V = 1	*Overflow*
0111	VC	V = 0	Sem *overflow*
1000	HI	C = 1 AND Z = 0	Acima sem sinal
1001	LS	C = 0 OR Z = 1	Abaixo ou igual sem sinal
1010	GE	N = V [(N = 1 AND V = 1) OR (N = 0 AND V = 0)]	Sinalizado maior que ou igual
1011	LT	N ≠ V [(N = 1 AND V = 0) OR (N = 0 AND V = 1)]	Sinalizado menor que
1100	GT	(Z = 0) AND (N = V)	Sinalizado maior que
1101	LE	(Z = 1) OR (N ≠ V)	Sinalizado menor que ou igual
1110	AL	—	Sempre (incondicional)
1111	—	—	Esta instrução só pode ser executada incondicionalmente

Modos de Endereçamento

Figura 13.1

Modos de endereçamento.



Modos de Endereçamento

Tabela 13.1

Modos básicos de endereçamento.

Modo	Algoritmo	Principal vantagem	Principal desvantagem
Imediato	Operando = A	Nenhuma referência à memória	Magnitude de operando limitada
Direto	EA = A	Simples	Espaço de endereçamento limitado
Indireto	EA = (A)	Espaço de endereçamento grande	Múltiplas referências à memória
Por registrador	EA = R	Nenhuma referência à memória	Espaço de endereçamento limitado
Indireto por registrador	EA = (R)	Espaço de endereçamento grande	Referência extra de memória
Por deslocamento	EA = A + (R)	Flexibilidade	Complexidade
De pilha	EA = topo da pilha	Nenhuma referência à memória	Aplicabilidade limitada

Modos de Endereçamento

- **Endereçamento Imediato**

  - Mais simples;

Operando = A

  - Pode ser usado para definir e utilizar constantes ou definir valores iniciais das variáveis.
  - Complemento a 2

Modos de Endereçamento

- **Endereçamento Direto**

  - Muito simples;

    EA(operando efetivo) = A (Campo de endereço)

  - Comum nas primeiras gerações dos computadores;
  - Requer apenas uma referência à memória e nenhum cálculo especial;
  - Oferece um espaço de endereços limitado.

Modos de Endereçamento

- **Endereçamento Indireto**

EA = (A)

- Vantagem desta abordagem é que, para o tamanho N de uma palavra, um espaço de endereçamento de 2N estará disponível;
- A desvantagem é que a execução da instrução requer duas referências à memória para obter o operando;
- Níveis em cascata: EA = (. . . (A) . . .)

Modos de Endereçamento

- **Endereçamento por registradores**
  - Semelhante ao endereçamento direto;

$$EA = R$$

  - Vantagens: Só precisa de um pequeno campo de endereço de instrução; Não consome tempo de acesso a referência de memória;
  - A desvantagem do endereçamento por registradores é o espaço de endereçamento muito limitado.

Modos de Endereçamento

- **Endereçamento indireto por registradores**
  - É análogo ao endereçamento indireto;

$$EA = (R)$$

  - A limitação do espaço de endereçamento do campo de endereço é superada;
  - Uma referência à memória a menos do que o endereçamento indireto.

Modos de Endereçamento

- **Endereçamento por deslocamento**
  - Combina as capacidades do endereçamento direto e do endereçamento indireto por registradores;

$$EA = A + (R)$$

  - Requer que a instrução tenha dois campos de endereço;
  - Valor A é usado diretamente;
  - Valor de referência R;
  - Usos comuns:
  - Endereçamento relativo, Endereçamento por registrador base, Indexação.

Modos de Endereçamento

- **Endereçamento por deslocamento (Endereçamento relativo)**
  - Registrador implicitamente referenciado é o contador do programa (PC);

$$EA = A + (PC)$$

  - Conceito de Localidade;
  - Salva bits de endereço de instrução se a maioria das referências de memória estiverem relativamente próximas da instrução sendo executada.

Modos de Endereçamento

- **Endereçamento por deslocamento (Endereçamento por registrador base)**
  - o registrador base contém um endereço da memória principal e o campo de endereço contém um deslocamento desse endereço;
  - A referência ao registrador pode ser explícita ou implícita;

$$EA = A + R$$

  - A = deslocamento/ campo de endereço;
  - R = endereço base.

Modos de Endereçamento

- **Endereçamento por deslocamento (Indexação)**
  - A = endereço base/ campo de endereço;
  - R = deslocamento;

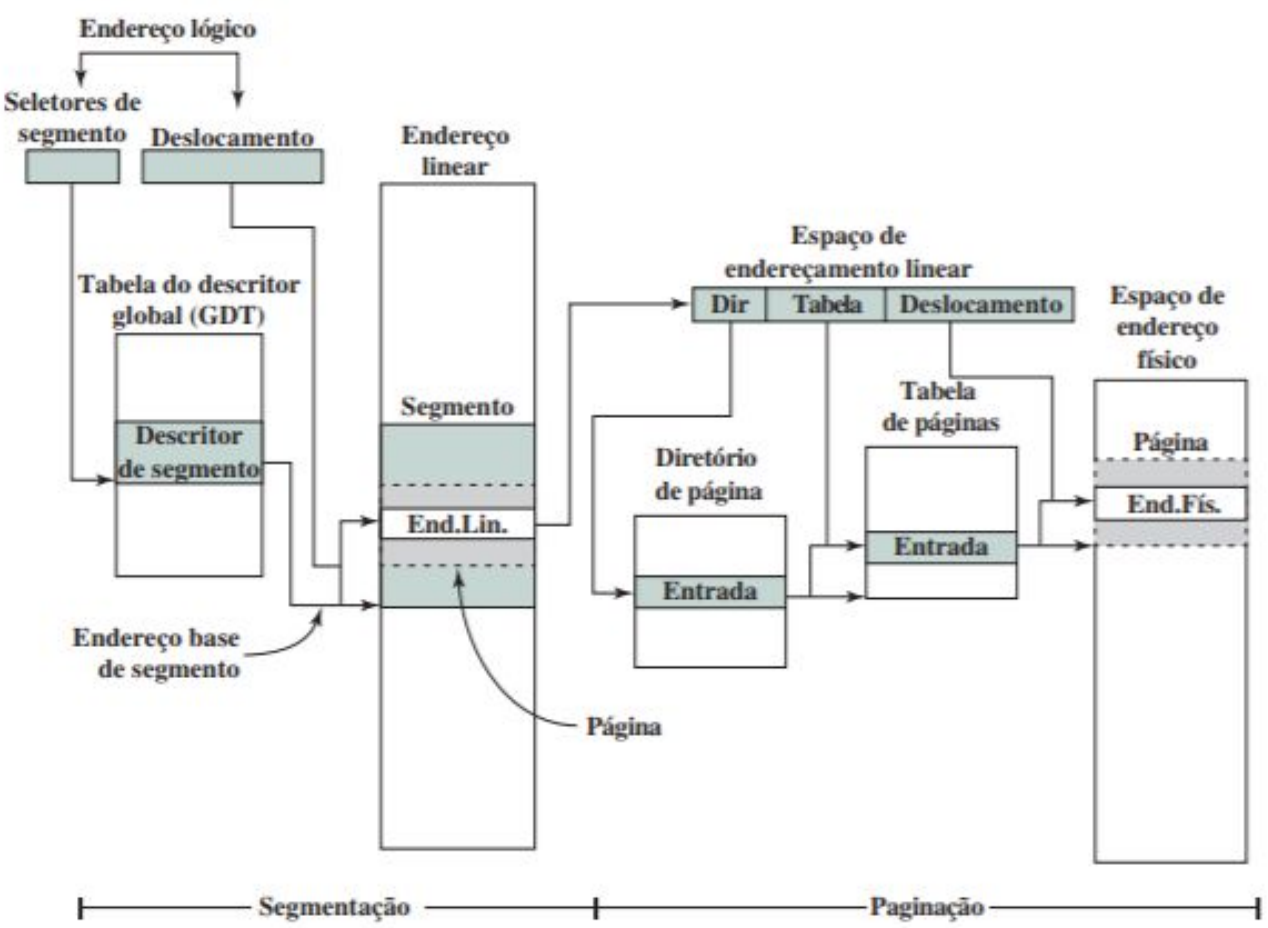
$$EA = A + R$$

  - Autoindexação
    EA = A + (R)
    (R) ← (R) + 1
  - Pós-indexação
    EA = (A) + (R)
  - Pré-indexação
    EA = (A + (R))

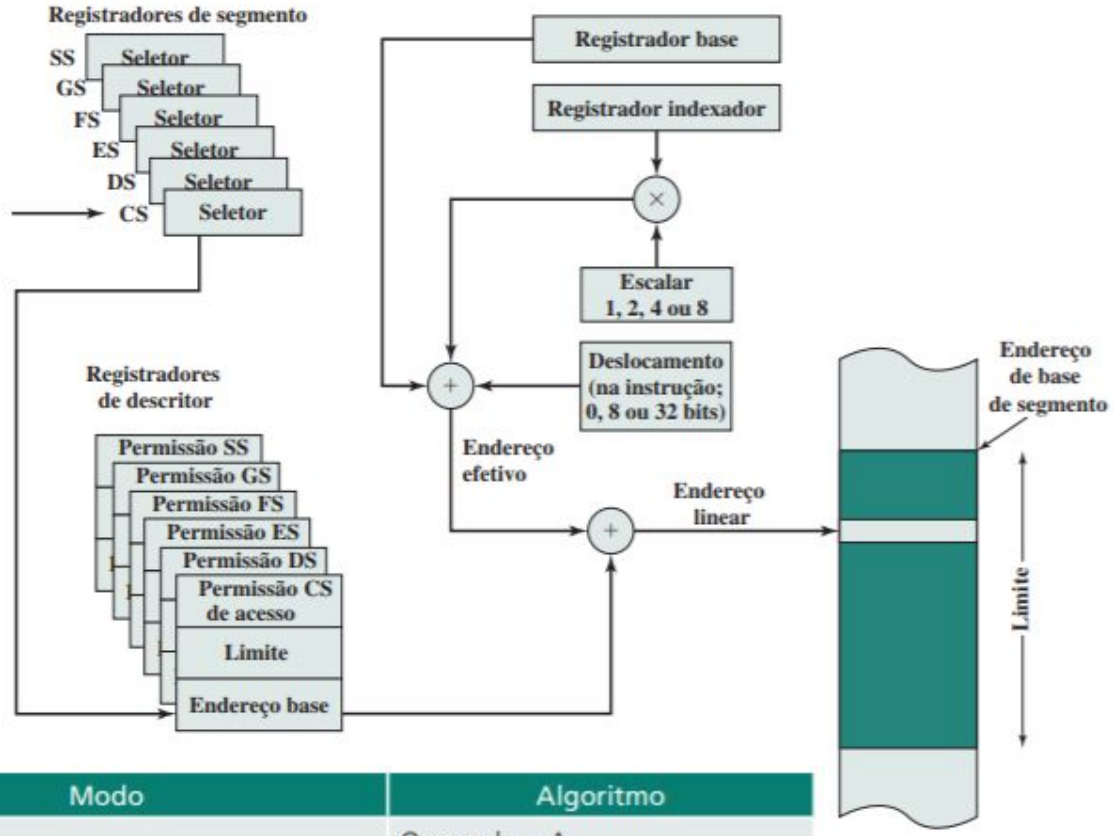
Modos de Endereçamento

- **Endereçamento de pilha**
  - Itens são adicionados ao topo da pilha;
  - Temos um ponteiro cujo valor é o endereço do topo da pilha;
  - O modo de endereçamento de pilha é uma forma de endereçamento implícito. As instruções da máquina não precisam incluir uma referência de memória, e sim operar no topo da pilha.

Modos de Endereçamento x86



Modos de Endereçamento x86



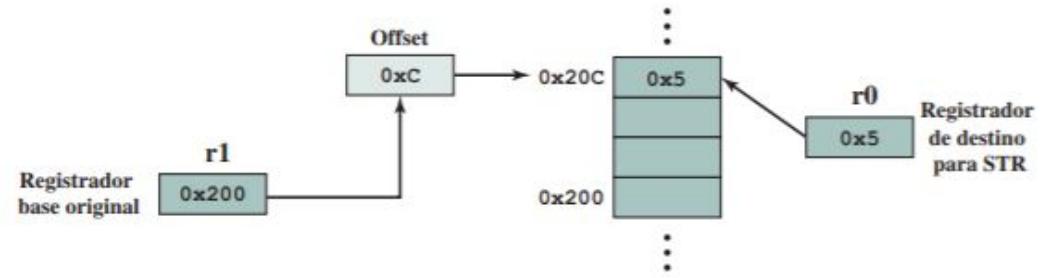
Modo	Algoritmo
Imediato	Operando = A
Operando em registrador	LA = R
Deslocamento	LA = (SR) + A
Base	LA = (SR) + (B)
Base com deslocamento	LA = (SR) + (B) + A
Índice escalado com deslocamento	LA = (SR) + (I) × S + A
Base com índice e deslocamento	LA = (SR) + (B) + (I) + A
Base com índice escalado e deslocamento	LA = (SR) + (I) × S + (B) + A
Relativo	LA = (PC) + A

Modos de Endereçamento ARM

- **Endereçamento de LOAD/STORE**
  - Offset
  - Pré-Indexação
  - Pós Indexação

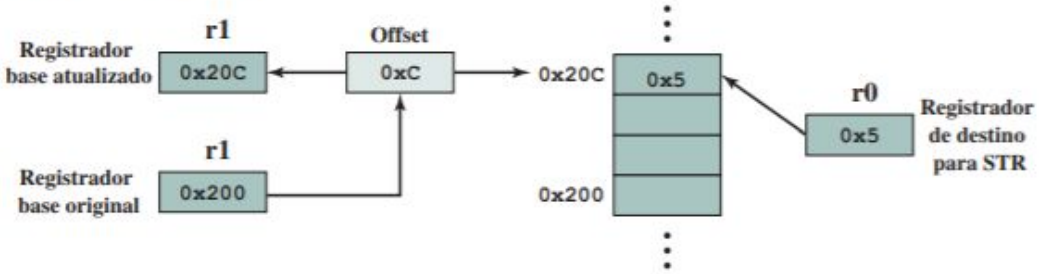
Modos de Endereçamento ARM

```
STRB r0, [r1, #12]
```



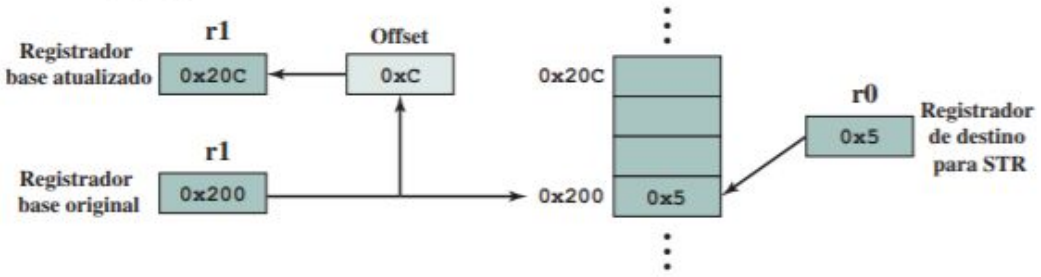
(a) Offset

```
STRB r0, [r1, #12]!
```



(b) Pré-indexação

```
STRB r0, [r1], #12
```



(c) Pós-indexação

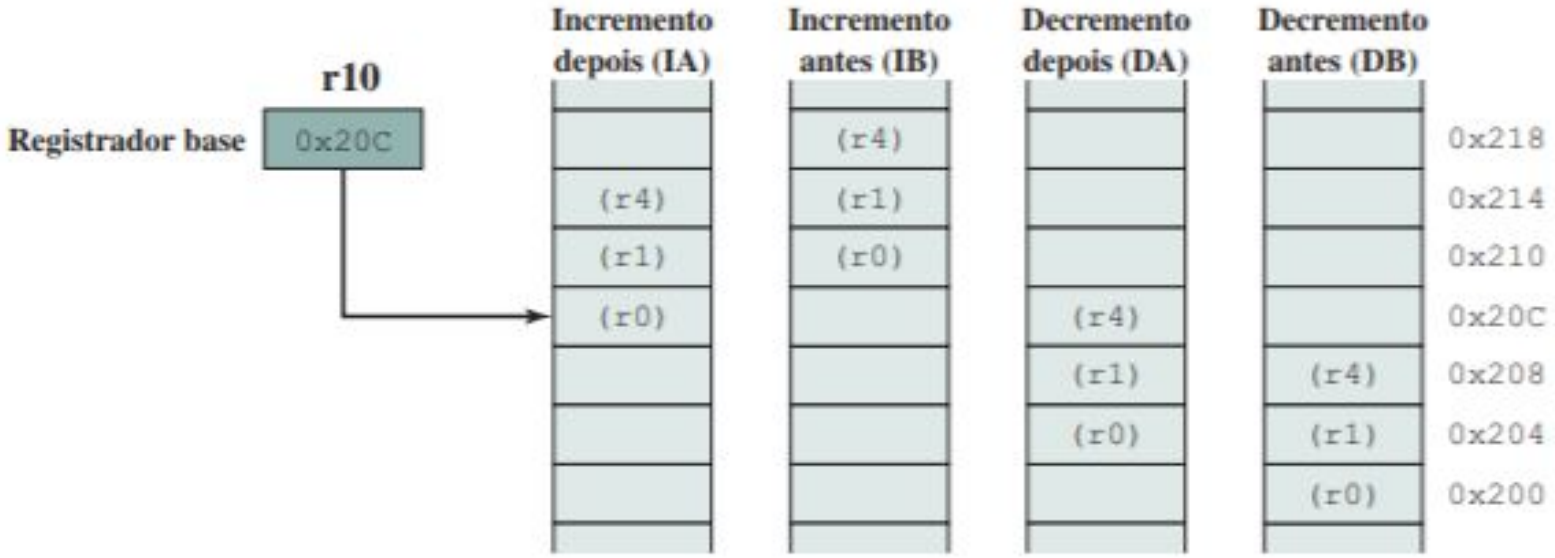
Modos de Endereçamento ARM

- **Endereçamento de Instruções de Processamento de Dados**
- **Instruções de Desvios**
- **Endereçamento Múltiplo de LOAD/STORE**

Modos de Endereçamento ARM

- **Endereçamento Múltiplo de LOAD/STORE**

```
LDMxx r10, {r0, r1, r4}
STMxx r10, {r0, r1, r4}
```



Formato de Instruções

- **Tamanho das Instruções**
  - Influenciado por diversos fatores
  - A busca pelo tamanho ideal
  - Vantagem dos modos de endereçamento
  - Taxa de Transferência de memória

Formato das Instruções

- **Alocação de Bits**
  - Opcodes X capacidade de Endereçamento
  - Número de modos de endereçamento
  - Número de operandos
  - Registrador X memória
  - Número de conjuntos de registradores
  - Intervalo de endereços
  - Granularidade do endereço

Formato das Instruções PDP-8

Instruções de referência à memória

Opcode			D/I	Z/C	Deslocamento						
0		2	3	4	5						11

Instruções de Entrada/Saída

1	1	0	Dispositivo						Opcode		
0		2	3					8	9		11

Instruções de referência aos registradores

Microinstruções de Grupo 1

1	1	1	0	CLA	CLL	CMA	CML	RAR	RAL	BSW	IAC
0	1	2	3	4	5	6	7	8	9	10	11

Microinstruções de Grupo 2

1	1	1	0	CLA	SMA	SZA	SNL	RSS	OSR	HLT	0
0	1	2	3	4	5	6	7	8	9	10	11

Microinstruções de Grupo 3

1	1	1	0	CLA	MQA	0	MQL	0	0	0	1
0	1	2	3	4	5	6	7	8	9	10	11

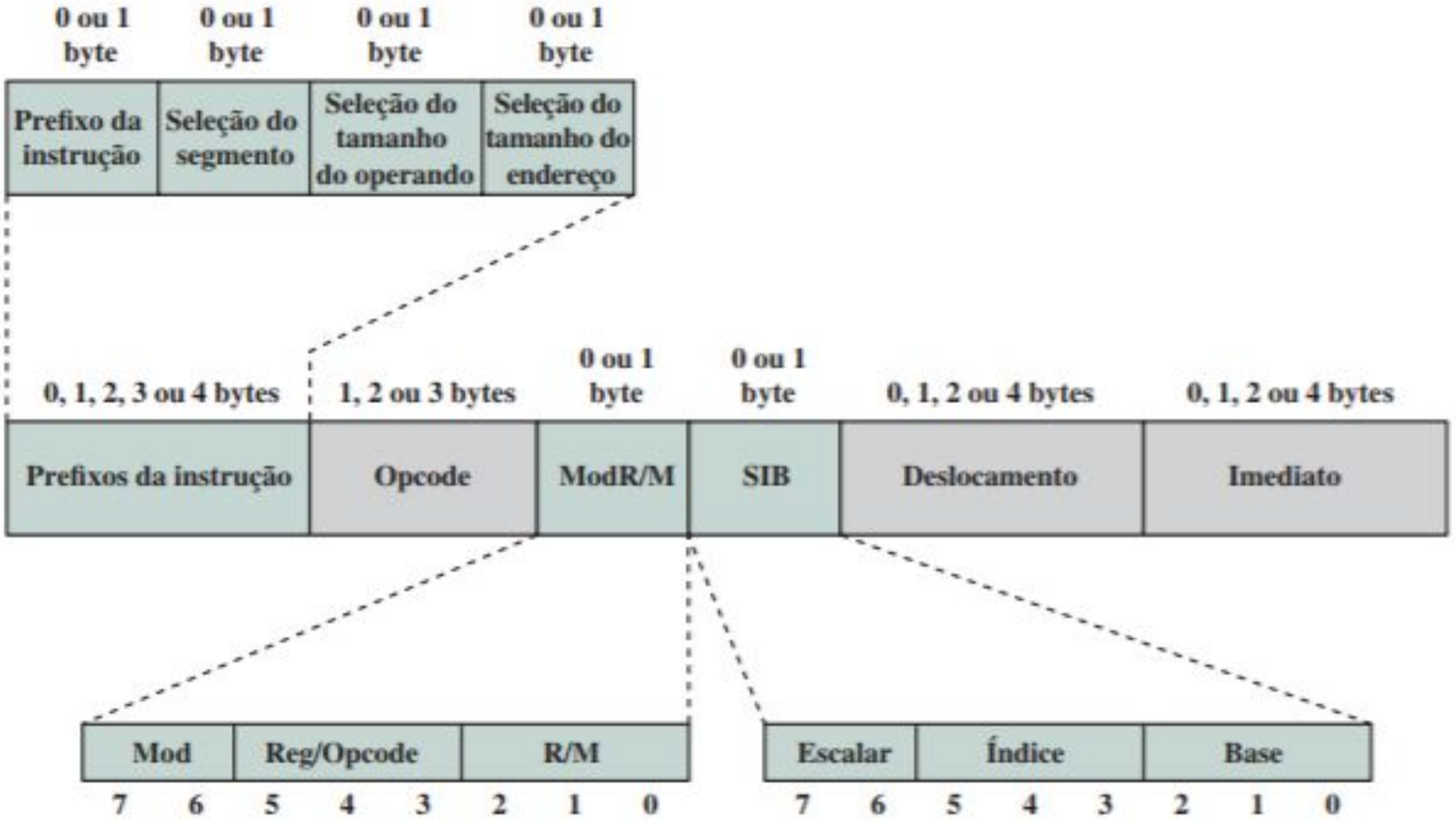
D/I = endereço direto/indireto
Z/C = página 0 ou atual
CLA = limpar acumulador
CLL = limpar link
CMA = complementar acumulador
CML = complementar link
RAR = rotacionar acumulador para direita
RAL = rotacionar acumulador para esquerda
BSW = trocar byte

IAC = incrementar acumulador
SMA = pular quando acumulador é negativo
SZA = pular quando acumulador é zero
SNL = pular quando link não é zero
RSS = reverter sentido quando pular
OSR = OR com troca de registrador
HLT = parar
MQA = quociente do multiplicador no registrador
MQL = carregar quociente do multiplicador

Formato das Instruções

- **Instruções com Tamanho Variável**
  - Mais possibilidades de opcodes
  - Endereçamento mais flexível
  - Complexidade do processador
  - Reconhecimento do processador

Formato das Instruções x86



Questões

1)Liste as três operações a respeito das transferências de controle encontradas frequentemente num conjunto de instruções e descreva suas características.

2) Liste os modos de endereçamento e descreva brevemente suas características.

3) Qual é a diferença entre pós-indexação e pré-indexação?