

Prova de LM

Jonathan Vinicius Nascimento

9,7

① PC = Endereço Inicial

ByteDeExecução = True

While (ByteDeExecução):

R = Memória[PC]

PC = PC + 1

Determina Tipo de Instrução (IR, Tipo da Instrução)

Acha Dados (Tipo da Instrução, IR, Localização Dado, Dado Requerido)

if (Dado Requerido):

dado = Memória[Localização Dado]

Execute (Tipo de Instrução, dado, memória, AC, PC, Bit de Instrução)

② 0705_{16}

③ a) al = EF

b) ax = CDEF

c) eax = 89ABCDEF

d) rax = 0123456789ABCDEF

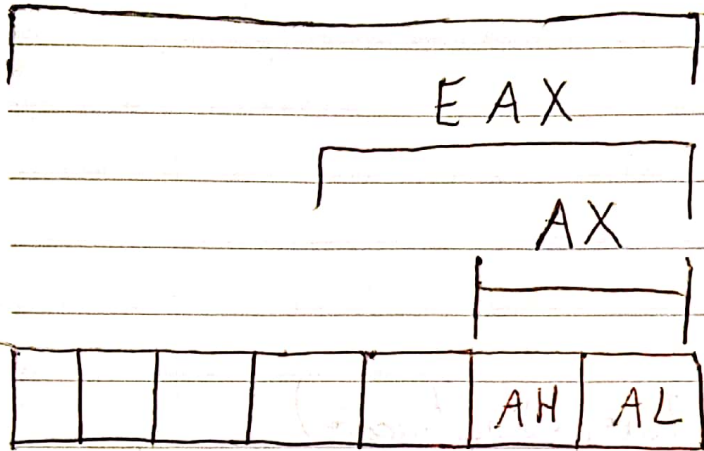
④ Cada registrador alcança no máximo de 1 byte ou 8 bits até 8 bytes ou 64 bits por cada endereço, dependendo do tipo da variável ou registro.

⑤ a) Bit Menos Significativo = 0

b) Bit Mais Significativo = 0

5) Desenho

RAX



6) a) $al = 8$

b) $ecx = 64$

c) $bx = 16$

d) $edx = 32$

8) a) Lea - Carrega o endereço efetivo + deslocamento, se o deslocamento for 0, não há necessidade de adicionar. Portanto, $LEA R, [Var]$ tem o mesmo efeito que $MOV R, Var$. No entanto, os comandos $MOV R, Var + deslocamento$ não são suportados. Para isso deve-se usar $LEA R, [Var + deslocamento]$

$MOV R - Var$ carrega o endereço de Var em R ou o conteúdo de Var no caso de $MOV R, [Var]$

b) $Section. text$: seção de código, onde o programa é feito
 $Section. data$: seção onde se declaram as variáveis inicializadas
 $Section. bss$: seção de variáveis não inicializadas

- c) RAX: Registrador conhecido como acumulador
 RBX: Registrador que pode ser usado índice de base
 RCX: Usado para contagem
 RDX: Auxiliar em algumas operações de multiplicação/divisão
 RIP: Guarda o endereço na linha de instrução
 RSI: Índice de origem
 RDI: Índice de destino

- OF: Flag overflow/flag
 DF: Flag de manipulação de dados
 IF: Flag de interrupção
 TF: Flag de busca
 ZF: Flag para resultado zero

d) RIP

- 9) a) $123_{10} = 0000\ 0000\ 0111\ 1011_2 = 000753_8 = 007B_{16}$
 b) $277_8 = 0000\ 0000\ 1011\ 1111_2 = 191_{10} = 00BF_{16}$
 c) $1011111_2 = 137_8 = 95_{10} = 005F_{16}$
 d) $AFA D_{16} = 1010\ 1111\ 1010\ 1101_2 = 127655_8 = -20563_{10}$

- 10) a) $6740_8 = 110\ 111\ 100\ 000$
 b) $10021_8 = 001\ 000\ 000\ 010\ 001$
 c) $4764_8 = 100\ 111\ 110\ 100$
 d) $477_8 = 100\ 111\ 111$