**ALLAN KEVIN SCAIN**

**FRANCISMO ASSIS MOREIRA DO NASCIMENTO**

**ARQUITETURA DE COMPUTADORES**

**TAREFA 7**

**1. Explique o que você entende por formato de instrução. Descreva os diferentes tipos de formatos de instrução de acordo com o número de operandos, fornecendo exemplos de instruções de cada tipo, vistas nos processadores K&S e MicMac.**

Uma instrução é uma operação que um dispositivo eletrônico programável suporta/disponibiliza para um programador. Um conjunto de instruções são várias instruções que se comunicam com alguma finalidade/objetivo, como fornecer a um usuário facilidade de interface, comunicação de dados, resgate de dados, busca de dados, seleção de memória, etc.

Um formato de Instrução se encontra perante o tanto de bits que um dispositivo eletrônico programável possui, incluindo números de endereço e etc. Para realizar a codificação em determinada linha de instrução de um processador.

Instruções do K&S;

R0 = R1 + R2; 1010000100 RR RR RR

Soma o R1 e o R2 e guarda em R0.

O processador K&S possui 2^5 bytes, tendo 32 espaços de memória na RAM e 4 endereços de memória. Já o MicMac possui 2^16 bytes, o que faz com que sua capacidade seja maior, por haver mais lugar para funções, mais espaços de memória, 65.536 mais especificamente.

Instruções do MicMac;

Mar:= pc; rd;

Pc:= pc + 1; rd;

Ir:= mbr; if n then got 28;

Tir:=lshift(ir + ir);if n then goto 19;

Instruções que fazem começar o programa do processador.

**2. Explique o que você entende por modos de endereçamento. Descreva cada um dos diferentes modos de endereçamento vistos em aula (livro texto do Tanenbaum 6ed), ilustrando com exemplos de instruções e como elas realizam o endereçamento (os exemplos devem ser diferentes dos vistos em aula).**

Modo de endereçamento é a maneira com que os bits de um campo de endereço são interpretados para achar o operando. Os tipos diferentes de endereçamento são:

**Endereçamento Imediato**, que se refere a busca imediata de um dado, onde o operando é automaticamente buscado na memória, ao mesmo tempo em que a própria instrução. Uma vantagem deste endereçamento é que não necessita de muito espaço na memória para a busca.



**Endereçamento direto**, um método de localizar algum operando com seu endereço de memória inteiro, porém o valor do operando possa mudar, a localização não pode, assim este tipo de endereçamento só pode ser utilizado para acessar variáveis globais.

**Endereçamento de registrador**, este método especifica um registrador invés de um endereço, geralmente é o modo mais presente nos computadores, devido a grande importância dos dados mutáveis, dados que ainda não estão salvos em memória. Também conhecido como modo registrador.



**Endereçamento indireto**, um método que pega os dados que vem da memória ou sai dela, o endereço não está ligado à instrução, mas sim está em um registrador, ou seja, um ponteiro. Backups de endereços. Sua maior utilidade é que não dispõe limite visível, consegue ser chamado várias vezes em diferentes instruções.

Exemplo: ADD(R4); AC:= AC + Mem [Reg[R4]]

**Endereçamento indexado** é nome que se dá ao endereçamento de memória que fornece um registrador mais um deslocamento constante.



**Endereçamento de base indexado** acontece quando um endereço de memória é calculado somando dois registradores. Um sempre será a base e o outro o índice.



**Endereçamento de Pilha**, instruções na qual adiciona um posicionamento como endereço e mais um acumulador. Muitas vezes interpretado como uma lista, em desenho, uma lista vertical. A pilha sempre terá um SP, que é um apontador para o topo da pilha.

**3. Agrupe as instruções do processador MicMac de acordo com o modo de endereçamento implementado por cada instrução e explicando como é realizado o cálculo do endereço em cada grupo de instruções.**

**Modo direto:** onde o operando será composto de um endereço de 12 bits. Dependendo da posição do dado na memória.

LODD, STOD, ADDD, SUBD, JPOS, JZER, JUMP, LOCO, JNEG, JNZE, CALL.

**Modo indireto:** o endereço é obtido por cálculo, o resultado do mesmo será armazenado em AC.

LODL, STOL, ADDL, SUBL, INSP, DESP.

**Modo local:** busca feita usando o SP, busca feita em pilhas.

PSHI, POPI, PUSH, POP, RETN, SWAP.

**4. É possível se ter vários modos de endereçamento em uma mesma instrução? Justifique sua resposta e explique de que maneira os modos de endereçamento afetam o tamanho e o desempenho dos programas.**

Sim, é possível usar vários modos de endereçamento, como visto no livro tenembaum, pagina 293, capt 5.4.5. Com este exemplo:

MOV r1 #0

MOV r2 #A

MOV r3 #A + 4046

LOOP ADD r1 (r2)

ADD r2 #4

COM r1 r2

BLT LOOP

As linhas 1, 2 e 3 usam o modo de endereçamento de registrador, onde indicam o destino do operando, então se usa o imediato para fazer uma operação no operando. O que acontece no exemplo é que R1, acumula a soma, sendo 0 inicialmente, R2 recebe o endereço de A, R3 recebe o endereço após a primeira palavra de A, então R1 busca o operando em R2, e R2 incrementa uma palavra e se R2 for menor que R3 um LOOP joga o endereço para linha 4.

De acordo com o exemplo, notamos que se não fosse utilizado um ponteiro, teríamos que usar variáveis próprias, que seriam alocadas em memória. Gastaria espaço e tempo de busca para o programa ser executado. Em um loop, as vezes é mais fácil utilizar busca em Registradores do que em memória alocada, por isso a velocidade de um array ponteiro é rápida, a busca acontece em um registrador devido ao seu endereço curto.