

SEL0415

Introdução à Organização de Computadores

Lista 09 – Set de Instruções & Modelos de Arquitetura

RESOLUÇÃO

1. Instrução é um padrão de código binário que é armazenado em um dispositivo de memória e que comanda o processador na execução de determinada tarefa. Já os mnemônicos (ADD, MOV, entre outros) são palavras curtas, geralmente em inglês, que representam a instrução binária na máquina.

2.

a) Opcode é um código binário referente a uma instrução/operação a ser executada pelo microprocessador. Operando são dados ou endereços a serem manipulados, atrelados à um opcode.

b) Com um Opcode de 8 bits, podem existir até $2^8 = 256$ instruções.

3. Assinale F para as afirmativas falsas, justificando-as, e V para as verdadeiras.

(V) No processo de programação de um μC , o programador edita/cria um código em uma linguagem específica (Assembly, por exemplo), que é traduzido para binário por meio de um compilador e então gravado na memória de programa (uma ROM) por meio de um circuito gravador.

(V) O código de uma instrução é composto por um opcode, que define o tipo de operação a ser executada, e por um ou mais operandos, que contém dados a serem manipulados nessa operação.

(F) A criação da linguagem Assembly permitiu a possibilidade de se criar e executar um mesmo programa em diferentes microprocessadores.

Um programa em Assembly é escrito para um processador em específico, uma vez que cada processador possui um conjunto de instruções diferentes.

(F) Em geral, um μC com arquitetura Von Neumann possui conjunto de instruções RISC, ao passo que um μC com arquitetura Harvard possui conjunto de instruções do tipo CISC.

A associação entre arquiteturas e conjuntos de instruções está invertida.

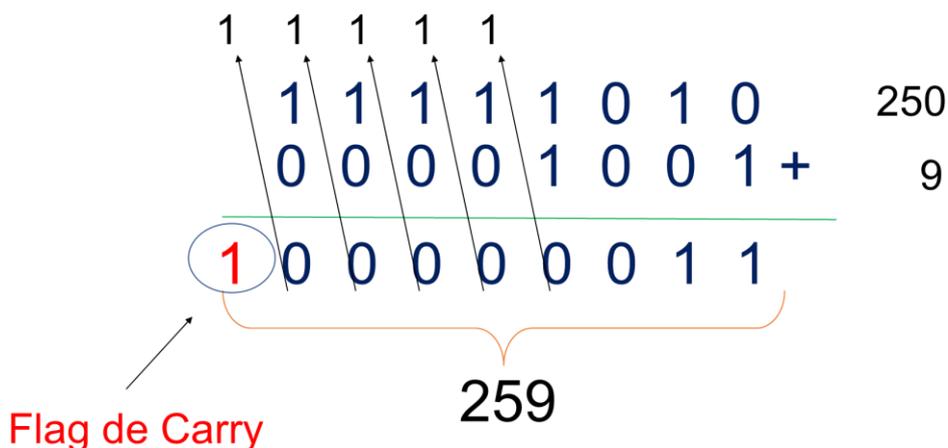
4. **Ciclo de Máquina:** É definido como uma quantidade fixa de períodos do clock, que varia conforme o microprocessador. Essa quantidade é a necessária de períodos (ou pulsos) para que o microprocessador busque uma palavra (opcode ou operando) na memória ROM, armazene ela no IR (Instruction Register), e execute a operação correspondente.

Ciclo de Instrução: É o tempo que leva para uma instrução ser completamente executada, e pode ser variável para cada instrução. Como, por exemplo, algumas instruções podem incluir mais de um operando, podem ser necessários mais de um ciclo de máquina para compor um único ciclo de instrução.

5. O ciclo de máquina (CM) do 8051 compreende 12T, ou seja, 12 períodos de clock. Para um clock de 4MHz, temos que $T = \frac{1}{4 \cdot 10^6} = 0,25 \mu s$. Logo, $CM = 12T = 12 \frac{1}{4 \cdot 10^6} = 3 \mu s$.

6. São bits indicadores de estado da ULA (Unidade Lógica Aritmética) e servem para indicar que “algo aconteceu” após determinada instrução/tarefa. Como exemplo, pode-se citar uma flag bem típica, flag de CARRY.

Supondo um microcontrolador com 8 bits de dados, imaginemos a soma de dois números, por exemplo, 250 (1111 1010 b) e 9 (0000 1001 b).



Note que o resultado correto tem 9 bits (1 a mais que o duto de dados). Se olhássemos somente para os 8 bits (tamanho do duto de dados), teríamos o número 3 (0000 0011b). O que é incorreto. Logo, a Flag de Carry é “setada” “dizendo” que, na verdade, deve-se considerar um bit a mais no resultado obtido naquela soma.

7. Das características abaixo, indique/associe quais se aplicam a um conjunto de instruções CISC e quais se aplicam a um conjunto RISC.
- I. Os programas são mais complexos.
 - II. A maioria das instruções tem a mesma duração.
 - III. Mais instruções disponíveis.
 - IV. Utiliza menos espaço na memória de programa.
 - V. Processamento de cada instrução é mais lento.
 - VI. Tempo de execução das instruções depende da frequência do clock.

CISC: III, V, VI

RISC: I, II, IV, VI

8. A arquitetura Harvard, diferente da Von Neumann, não possui barramento compartilhado pelas memórias RAM e ROM. Possui um duto separado para dados (RAM) e instruções (ROM). A execução das instruções tende a ser mais rápida, pois é possível acessar as duas memórias ao mesmo tempo, permitindo o pipelining. Em geral, a arquitetura Harvard possui conjunto de instruções RISC, ao passo que um μC com arquitetura Von Neumann possui conjunto de instruções do tipo CISC.
9. Pipelining (linha de montagem) é uma técnica que permite mais de uma instrução ser executada ao mesmo tempo. O microprocessador lê uma instrução ao mesmo tempo que executa a instrução lida anteriormente. Requer RISC, pois as instruções devem ter o mesmo tamanho (com exceção das de salto).

10.

OPCODE	MNEM.	UTILIZAÇÃO	EQUAÇÃO
1010	ADD	ADD (operando)	ACC = ACC + (operando)
0101	SUB	SUB (operando)	ACC = ACC - (operando)
0010	MPY	MPY (operando)	ACC = ACC * (operando)
0001	DIV	DIV (operando)	ACC = ACC / (operando)
1100	LDA	LDA (operando)	ACC = (operando)
1001	STA	STA (operando)	(operando) = ACC

a)

LDA A: ACC = A
 ADD C: ACC = ACC + C = A + C
 STA X: X = ACC = A + C
 LDA B: ACC = B
 MPY D: ACC = ACC * D = B * D
 SUB E: ACC = ACC - E = (B * D) - E
 STA Y: Y = ACC = (B * D) - E
 LDA X: ACC = X = A + C
 ADD Y: ACC = ACC + Y = A + C + Y = A + C + (B * D) - E
 DIV F: ACC = ACC / F = (A + C + (B * D) - E) / F
 STA X: X = ACC = (A + C + (B * D) - E) / F

Ou seja, $X = \frac{(A+C+(B \cdot D)-E)}{F}$

b) O opcode é de 4 bits. Logo, pode haver $2^4 = 16$ instruções.

c)

$$T = \frac{1}{6 \cdot 10^6} = 0,167 \mu s$$

$$CM = 4T = \frac{4}{6 \cdot 10^6} = 0,667 \mu s$$

Como temos 11 instruções, o tempo total é de:

$$\text{Tempo programa} = 11CM = \frac{44}{6 \cdot 10^6} = 7,33 \mu s$$