## Lista de Exercícios nº 4

- 1. Escrever um programa em linguagem assembly do 8085 que multiplique um número por 8.
- 1) Pode ser qualquer número.
  - a) Inserir um número a partir da porta de entrada 01.
  - b) Armazenar o número no endereço de memória 30h.
- 2) Lembre-se que os números são expressos em hexadecimal.
- 3) Todas as operações que são realizadas na ALU implicam que os operandos devem estar nos registradores de uso geral (um deles no acumulador).
- 4) Para saber se existe transbordamento (overflow) na operação, mostrar ao final o conteúdo do flag de carry pela porta 02.
- 5) A saída sempre se dará pela porta 0h.

## Observações:

Para a realização deste exercício supomos os números a serem manuseados são de 8 bits.

No caso de que seja produzido um carry, a operação será interrompida e se mostrará 1 na porta 2.

- 2. Escrever um programa em linguagem assembly do 8085 para encontrar o maior entre dois números de 8 bits.
  - 1) Como no exercício anterior pode ser qualquer número.
    - a) Inserir os números a partir de entrada pelas portas 1 e 2 respectivamente.
    - b) Armazenar os números nos endereços de memória 30h e 31h.
  - 2) No caso de serem iguais, a saída será o número "1"
  - 3) A saída sempre se dará pela porta 00.
  - 3. Implementar um cronômetro de contagem regressiva.
  - 1) Usar a porta 0 para inserir uma série decrescente.
  - 2) Começar ligando o led do bit mais significativo (bit7) e ir até o menos significativo (bit0).
  - 3) Ao final ligar todos os leds para notificar o final da contagem regressiva.
  - 4. Soma de números decimais.
  - a) Somar dois números que se encontram em formato decimal nos endereços de memória 50h e 51h.
  - b) Armazenar o resultado nos endereços 52h e 53h (o primeiro endereço armazena o possível carrydecimal).
  - c) Devemos levar em conta que o processador considera todos os números como hexadecimais, pelo que as operações aritméticas serão realizadas nesta base.
  - d) Portanto, ao final da operação deve-se aplicar um processo de ajuste decimal. Serve como "dica" o estudo da representação BCD (Binary Coded Decimal).

- 5. Procurar o máximo e o mínimo de um conjunto de números
- a) Existe um segmento de memória no qual há um conjunto de dados onde queremos obter o maior e o menor elemento.
  - b) O menor elemento será mostrado na porta 0 e o maior na porta 1.
  - c) O segmento de memória está assim indentificado:

Endereço Conteúdo 00h programa

50h Endereço do primeiro elemento da lista

51h Restante dos elementos da lista

- d) Será utilizada uma lista fixa de 10 números.
- e) Podem existir valores repetidos.
- f) É recomendado inserir previamente os números nos endereços indicados.
- g) Criar um fluxograma para facilitar a codificação do programa.
- 6. Busca de elementos pares.
- a) Temos na memória uma tabela desordenada que começa no endereço 150h e cujo tamanho está especificado no endereço de memória 140h.
- b) Queremos trasladar os elementos da tabela que sejam pares para uma tabela que comece no endereço de memória 170h.
  - c) Finalmente é necessário mostrar pela porta 00 o número de elementos transferidos.
  - d) O mapa da memória para o programa será o seguinte:

Endereço	Conteúdo
50H,	Programa
140H	Número de elementos da lista
150H	Endereço do primeiro elemento da lista
151H,	Demais elementos da lista
170H,	Relação de elementos pares da lista

- e) Podem existir valores repetidos.
- f) É recomendado inserir previamente os números nos endereços indicados.
- g) Criar um fluxograma para facilitar a codificação do programa.
- 7. Busca de um elemento dentro de uma tabela
- a) Temos na memória uma tabela desordenada que contem N dados. El valor de "N" é especificado no endereço F7h.
  - b) O primeiro dado (inicio da tabela) se encontra no endereço de memória F8h.
- c) Buscamos um número (dado pela porta de entrada 00). Se durante a busca encontrar este número, se mostrará sua posição dentro da tabela pela porta 01.
- d) Caso contrario, será inserido ao final da tabela e se mostrará sua posição dentro da tabela pela porta 01 e deve-se atualizar o valor de "N".
  - e) O mapa de memória para o programa será o seguinte:

Endereço Conteúdo 00H Programa F7H Número de elementos da tabela

F8H Endereço do primeiro elemento da tabela

F9H, ... Demais elementos da lista

- f) Podem existir valores repetidos e neste caso somente será mostrado o endereço do primeiro elemento encontrado.
  - g) É recomendado inserir previamente os números nos endereços indicados.
  - h) Criar um fluxograma para facilitar a codificação do programa.
  - 8. Executar a divisão de dois números de 8bits
  - a. Usar dois números de 8 bits armazenados na memória (endereços 100h e 101h).
  - b. O quociente deve ficar armazenado no endereço 102h e o resto da divisão no endereço 103h
  - b) Para simplificar não trabalharemos com números negativos
  - c) No caso de divisão decimal, o quociente será 0.
  - d) Cuidado quando o dividendo ou o divisor seja 0.
  - 9. Calcular a média de um conjunto de números
  - a) Dispomos de uma série de números decimais armazenados na memória a partir do endereço 200h.
  - b) O tamanho desta lista será especificado a partir de entrada pela porta 00.
  - c) Obter a média destes números e armazená-la no endereço 250h.
  - 10. Executar a soma de dois números de 32 bits
  - a) Os números serão armazenados nos endereços de memória 100h-103h e 104h-107h.
  - b) A soma deverá ser armazenada nos endereços 108h-111h.
  - c) Levar em conta os possíveis "carrys" produzidos nas operações. O último carry deverá ser mostrado através da porta 00.