

EXAME NACIONAL DO ENSINO SECUNDÁRIO

12.º Ano de Escolaridade (Decreto-Lei n.º 286/89, de 29 de Agosto)

Curso Tecnológico de Electrotecnia/Electrónica

Duração da prova: 90 min + 30 min de tolerância
1997

2.ª FASE

PROVA ESCRITA DE SISTEMAS DIGITAIS

- As justificações que apresentar devem ser completas e sucintas.
- A prova inclui uma folha com a tabela resumida do conjunto de instruções do microprocessador 8085.

I

1. Projecte um circuito sequencial com elementos de memória tipo J-K e a lógica combinacional adequada, que seja capaz de funcionar como divisor da frequência de 16 Hz do clock (CLK). Para o efeito, o circuito deverá dispor de uma linha de controlo X, que, ao ser colocada, respectivamente, a "0" ou a "1", permitirá obter numa saída S um sinal de 4 Hz ou 8 Hz.

2. A figura 1 representa a implementação física de um codificador com diodos de uma memória ROM, de 8 palavras de 4 bits.

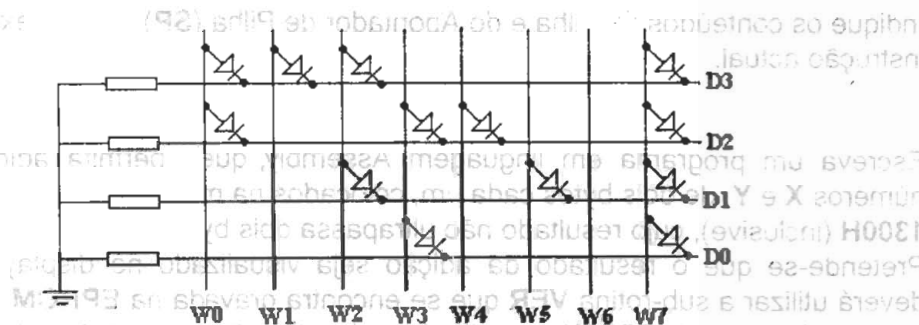


Fig. 1

2.1. Indique os conteúdos binários das 8 palavras que estão guardadas nessa memória.

2.2. Caracterize, justificando, uma memória ROM, quanto ao tipo de acesso e volatilidade.

3. A figura 2 representa o fluxograma de um circuito detector de sequências, com uma entrada X sincronizada com os impulsos de um clock e uma saída S.

3.1. Explique o funcionamento do circuito.

3.2. Utilizando elementos de memória tipo D, estabeleça, o mais simplificado possível, o circuito lógico do detector de sequências.

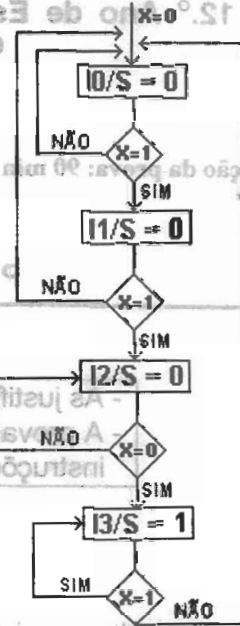


Fig 2

1. As instruções "CALL end" e "RET" têm aplicação quando há necessidade de utilizar uma sub-rotina.

Indique, justificando, se ambas as instruções fazem parte do programa principal.

2. Considere o seguinte extracto incompleto de um programa em Assembly. O Contador de Programa (PC) encontra-se carregado com o endereço 1205H

Endereço	Mnemónica da instrução	Operando
.....		
1200H	LXI	SP, 1FFFH
1203H	MVI	A, 55H
1205H	CALL	05C8H

Indique os conteúdos da Pilha e do Apontador de Pilha (SP), após a execução da instrução actual.

3. Escreva um programa em linguagem Assembly, que permita adicionar dois números X e Y, de dois bytes cada um, colocados na memória a partir da posição 1300H (inclusive), cujo resultado não ultrapassa dois bytes.

Pretende-se que o resultado da adição seja visualizado no display, pelo que deverá utilizar a sub-rotina VER que se encontra gravada na EPROM, a partir da posição de memória 05C8H, permitindo a visualização do conteúdo do par HL no campo de direcções do display.

FIM

INSTRUÇÕES DO MICROPROCESSADOR 8085

INSTRUÇÕES DE TRANSFERÊNCIA DE DADOS

<p>Mover</p> <p>MOV { A,A 7F A,B 78 A,C 79 A,D 7A A,E 7B A,H 7C A,L 7D A,M 7E</p> <p>MOV { B,A 47 B,B 40 B,C 41 B,D 42 B,E 43 B,H 44 B,L 45 B,M 46</p> <p>MOV { C,A 4F C,B 48 C,C 49 C,D 4A C,E 4B C,H 4C C,L 4D C,M 4E</p> <p>MOV { D,A 57 D,B 50 D,C 51 D,D 52 D,E 53 D,H 54 D,L 55 D,M 56</p>	<p>Mover (cont.)</p> <p>MOV { E,A 5F E,B 58 E,C 59 E,D 5A E,E 5B E,H 5C E,L 5D E,M 5E</p> <p>MOV { H,A 67 H,B 60 H,C 61 H,D 62 H,E 63 H,H 64 H,L 65 H,M 66</p> <p>MOV { L,A 6F L,B 68 L,C 69 L,D 6A L,E 6B L,H 6C L,L 6D L,M 6E</p> <p>MOV { M,A 77 M,B 70 M,C 71 M,D 72 M,E 73 M,H 74 M,L 75</p>	<p>Mover Imediato</p> <p>MVI { A, byte 3E B, byte 06 C, byte 0E D, byte 16 E, byte 1E H, byte 26 L, byte 2E M, byte 36</p> <p>Carregar Imediato</p> <p>LXI { B, dble 01 D, dble 11 H, dble 21 SP, dble 31</p> <p>Carregar/Armazenar</p> <p>LDAX B 0A LDAX D 1A LHLD end 2A LDA end 3A</p> <p>STAX B 02 STAX D 12 SHLD end 22 STA end 32</p>
---	--	--

XCHG EB

end = endereço de 16 bits (segundo e terceiro bytes de instruções de 3 bytes)

INSTRUÇÕES ARITMÉTICAS E LÓGICAS

<p>Somar *</p> <p>ADD { A 87 B 80 C 81 D 82 E 83 H 84 L 85 M 86</p> <p>ADC { A 8F B 88 C 89 D 8A E 8B H 8C L 8D M 8E</p> <p>Subtrair *</p> <p>SUB { A 97 B 90 C 91 D 92 E 93 H 94 L 95 M 96</p> <p>SBB { A 9F B 98 C 99 D 9A E 9B H 9C L 9D M 9E</p> <p>Soma dupla +</p> <p>DAD { B 09 D 19 H 29 SP 39</p> <p>ANA { A A7 B A0 C A1 D A2 E A3 H A4 L A5 M A6</p> <p>XRA { A AF B A8 C A9 D AA E AB H AC L AD M AE</p> <p>ORA { A B7 B B0 C B1 D B2 E B3 H B4 L B5 M B6</p> <p>CMP { A BF B B8 C B9 D BA E BB H BC L BD M BE</p>	<p>Incrementar **</p> <p>INR { A 3C B 04 C 0C D 14 E 1C H 24 L 2C M 34</p> <p>INX { B 03 D 13 H 23 SP 33</p> <p>Decrementar **</p> <p>DCR { A 3D B 05 C 0D D 15 E 1D H 25 L 2D M 35</p> <p>DCX { B 0B D 1B H 2B SP 3B</p> <p>Especiais</p> <p>DAA* 27</p> <p>Imediatas</p> <p>ADI byte C6 ACI byte CE SUI byte D6 SBI byte DE</p> <p>Imediato</p> <p>ANI byte E6 XRI byte EE ORI byte F6 CPI byte FE</p> <p>Rotação</p> <p>RLC 07 RRC 0F RAL 17 RAR 1F</p> <p>Especiais</p> <p>CMA 2F STC 37 CMC 3F</p>
---	--

INSTRUÇÕES DE DESVIO

<p>Saltar</p> <p>JMP end C3 JNZ end C2 JZ end CA JNC end D2 JC end DA JPO end E2 JPE end EA JP end F2 JM end FA PCHL end E9</p> <p>Chamar</p> <p>CALL end CD CNZ end C4 CZ end CC CNC end D4 CC end DC CPO end E4 CPE end EC CP end F4 CM end FC</p>	<p>Retornar</p> <p>RET C9 RNZ C0 RZ C8 RNC D0 RC D8 RPO E0 RPE E8 RP F0 RM F8</p> <p>Recomeçar</p> <p>RST { 0 C7 1 CF 2 D7 3 DF 4 E7 5 EF 6 F7 7 FF</p>
--	---

end = endereço de 16 bits (segundo e terceiro bytes de instruções de 3 bytes)

INSTRUÇÕES DE CONTROLO DA MÁQUINA

<p>Operação com pilha</p> <p>PUSH { B C5 D D5 H E5 PSW F5</p> <p>POP { B C1 D D1 H E1 PSW F1</p> <p>XTHL E3 SPHL F9</p>	<p>Entrada/Saída</p> <p>OUT byte D3 IN byte DB</p> <p>Controlo</p> <p>DI F3 EI FB NOP 00 HLT 76</p> <p>Novas instruções (somente 8085)</p> <p>RIM 20 SIM 30</p>
--	--

byte = constante ou expressão de registo que, avaliada, fornece um dado de 8 bits (segundo byte de instruções de 2 bytes).

- * = todos os bits de flag (C, Z, P, CY, AC) são afectados.
- ** = todos os bits de flag com excepção de CARRY (transporte) são afectados. (excepção: INX e DCX não afectam nenhum flag).
- + = apenas CARRY (transporte) é afectado.

Todos os direitos reservados pela Intel Corporation 1975.
P = 1 se paridade par

INSTRUÇÕES DO MICROPROCESSADOR 8085

INSTRUÇÕES ARITMÉTICAS E LÓGICAS

COTAÇÕES INSTRUÇÕES DE TRANSFERÊNCIA DE DADOS

Operações	Opções	Operações	Opções
Incrementar **	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Decrementar **	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Adição	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Subtração	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Multiplicação	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Divisão	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
AND	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
OR	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
XOR	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
NOT	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Shift Left	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Shift Right	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Rotate	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI
Compare	A, KI, B, RI, C, KI, H, KI, M, KI	Subtrair *	A, KI, B, RI, C, KI, H, KI, M, KI

Operações	Opções	Operações	Opções
Mover	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M

Operações	Opções	Operações	Opções
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M
Transferir	A, B, C, D, E, H, M	Mover	A, B, C, D, E, H, M

As operações de transferência de dados são executadas no modo de 8 bits. O modo de 16 bits é suportado apenas para as instruções de transferência de dados. As operações de transferência de dados são executadas no modo de 8 bits. O modo de 16 bits é suportado apenas para as instruções de transferência de dados. As operações de transferência de dados são executadas no modo de 8 bits. O modo de 16 bits é suportado apenas para as instruções de transferência de dados.