
 <b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

#### Observações

- Responda ao teste apenas com caneta azul ou preta.
- Preencha todo o cabeçalho da(s) folha(s) de teste: nome completo do estudante, número do estudante, data da realização da prova de avaliação, nome da unidade curricular e nome do curso de licenciatura.
- Coloque em cima da mesa onde irão realizar a prova de avaliação o seu cartão de estudante ou outro cartão que os identifique.
- Coloque em cima da mesa para a realização da prova de avaliação, salvo indicação em contrário pelo docente, apenas os seguintes materiais, caneta, lápis, borracha. Todo o restante material deverá ser colocado debaixo da mesa.
- Não deverão sair da sala de exame sem terem assinado a folha de presenças no caso de um exame final ou de passar o cartão de estudante na máquina de registo de presenças no caso de um momento de avaliação que não exame final.
- Só podem sair da sala ao fim de 30 minutos depois do início da prova.
- Caso queira desistir deverá escrever na folha de exame "Desisto" e colocar por baixo a sua assinatura.
- Apresente a resolução desta prova apenas na(s) folha(s) fornecida(s) para esse fim.
- Justifique convenientemente todas as respostas.
- Se necessitar de mais folhas de teste ou mais folhas de rascunho deverá solicitar as mesmas ao docente.
- Quando solicitar uma nova folha de teste não deverá esquecer-se de no cabeçalho atualizar o número de folhas de teste. Cada folha de teste é constituída por quatro páginas, assim o número de folhas é 1/1. Caso o estudante solicite nova folha de teste o número de folhas a indicar na primeira folha será 1/2 e na segunda folha 2/2. Para não haver engano na contagem este parâmetro do cabeçalho deve apenas ser preenchido aquando da conclusão da prova de avaliação.
- Não é permitido o uso de qualquer dispositivo eletrónico, tais como por exemplo, máquina de calcular, salvo indicação em contrário, dada pelo docente responsável da unidade curricular.
- Não é permitido o uso de qualquer documentação além da indicada/fornecida pelo docente.
- Na altura da entrega da prova pelo estudante, este deve entregar tudo o que lhe foi entregue pelo docente, folha de teste, folha de rascunho, enunciado, folhas de apoio, etc.

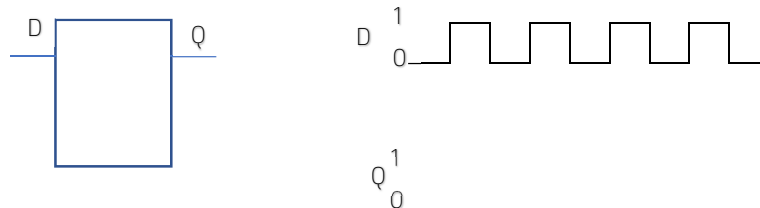
- A representação do número decimal  $5_{(10)}$  em binário é:
  - $00101_{(10)}$
  - $00101_{(2)}$
  - $00111_{(2)}$
  - Nenhuma das anteriores
- O código BCD excesso 3 é um código que:
  - Utiliza as 3 primeiras e as 3 últimas combinações das 16 primeiras combinações do código binário
  - Não utiliza as 3 primeiras nem as 3 últimas combinações das 16 primeiras combinações do código binário
  - Utiliza as 5 primeiras combinações e as 5 últimas combinações das 16 primeiras combinações do código binário
  - Nenhuma das anteriores
- No código de Hamming não existe erro na informação quando o valor das funções  $C1$ ,  $C2$  e  $C3$  são:
  - $C1 = 1, C2 = 1$  e  $C3 = 1$
  - $C1 = 0, C2 = 0$  e  $C3 = 0$
  - $C1 = 1, C2 = 0$  e  $C3 = 1$
  - Nenhuma das anteriores
- Num multiplexer a relação entre entradas de informação e entradas de controlo é dada por:
  - Número de entradas de informação é igual a  $2^{n^{\circ} \text{ de entradas de controlo}}$
  - Número de entradas de informação é maior que  $2^{n^{\circ} \text{ de entradas de controlo}}$
  - Número de entradas de informação é menor que  $2^{n^{\circ} \text{ de entradas de controlo}}$
  - Nenhuma das anteriores
- A arquitetura de um computador corresponde:
  - À forma como a arquitetura é implementada
  - À forma como o sistema é visto pelo programador de assembly
  - À forma como os componentes são dispostos no sistema
  - Nenhuma das anteriores

 <b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	


6. A unidade de controlo:
  - a) Armazena dados temporariamente
  - b) Busca instruções na memória e controla o fluxo de dados entre a ALU e a memória
  - c) Realiza as operações matemáticas e lógicas
  - d) Nenhuma das anteriores
7. A comunicação entre a unidade de controlo e a unidade de processamento é executada através:
  - a) Das palavras de estado
  - b) Das palavras de controlo
  - c) Dos bits de estado
  - d) Nenhuma das anteriores
8. Uma interrupção externa é:
  - a) Gerada por um dispositivo I/O
  - b) Gerada por um programa
  - c) Gerada aleatoriamente por um dispositivo de Input
  - d) Nenhuma das anteriores
9. Uma das técnicas usadas no tratamento de múltiplas interrupções é denominado por disable do mecanismo de interrupções e consiste:
  - a) No tratamento sequencial de interrupções
  - b) Na atribuição de prioridades às interrupções
  - c) No não tratamento de qualquer interrupção
  - d) Nenhuma das anteriores
10. Num barramento multiplexado:
  - a) Existem fios condutores para dados e endereços em separado
  - b) Os mesmos fios condutores são usados para endereços ou dados
  - c) Existem fios condutores para registos que são usados para comunicarem endereços
  - d) Nenhuma das anteriores
11. Nos barramentos que utilizam temporização síncrona:
  - a) Todos os eventos são controlados pelo sinal de relógio
  - b) Todos os eventos ocorrem e dependem da ocorrência de outros eventos
  - c) Todos os eventos são controlados pelo sinal de relógio e dependem da ocorrência de outros eventos
  - d) Nenhuma das anteriores
12. A memória externa é:
  - a) Diretamente acessível pelo processador
  - b) Acessível ao processador através de controladores I/O
  - c) Inacessível pelo processador
  - d) Nenhuma das anteriores
13. Numa memória, o tempo de ciclo representa:
  - a) A velocidade a que os dados são transmitidos de ou para a memória
  - b) O tempo para aceder a uma posição de memória
  - c) O tempo de acesso, mais o tempo necessário até que um segundo endereço possa ser acedido
  - d) Nenhuma das anteriores

<b>P.PORTO</b> <b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

14. Nas memórias cache que usam mapeamento associativo por conjuntos:
- Cada bloco da memória principal é mapeado numa única linha da cache
  - Um bloco da memória principal pode ser mapeado em qualquer linha da cache
  - Um bloco da memória principal é mapeado em qualquer linha de um determinado conjunto
  - Nenhuma das anteriores
15. Converta o seguinte número  $100_{(10)}$  para a base 2.
16. Projete a seguinte função lógica,  $F = X.Z + \bar{X}.Z$ , utilizando um decodificador. Utilize a seguinte ordem para as variáveis, X e Z.
17. Implemente a seguinte função lógica,  $F = X.Y.Z + \bar{X}.Z$ , utilizando um multiplexer de 4 canais de entrada de informação. Caso necessite utilize a seguinte ordem para as variáveis, X, Y e Z. Se necessitar de descartar alguma variável, descarte a variável X.
18. Construa, o diagrama temporal do seguinte circuito sequencial assíncrono, de acordo com os sinais de entrada da figura.



19. Adicione o conteúdo de duas posições de memória consecutivas (1040H e 1041H) e guarde o resultado na posição de memória seguinte (1042H).
20. Elabore um programa que carregue os números E2H e 58H nos registos B e C, respetivamente. Guarde o número A2H na posição de memória 1065H, usando o par de registos HL como apontador da memória. Realize a subtração entre E2H – 58H. Ache o complemento do resultado da subtração obtida anteriormente, e some esse valor com o número A2H contido na posição de memória 1065H. Finalmente guarde o resultado na posição de memória 1066H.
21. Elabore um programa que adicione dois números de 16 bits colocados cada um em duas posições sucessivas de memória (1040H e 1042H), sabendo que o byte mais significativo está na posição de memória com endereço superior. Guarde o resultado nas duas posições de memória seguintes.

 <b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

Cotação:


1. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
2. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
3. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
4. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
5. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
6. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
7. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
8. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
9. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
10. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
11. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
12. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
13. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
14. Pergunta de escolha múltipla:
  - Resposta correta 1 valor, resposta errada -0.25 valor, sem resposta 0 valores
15. 0.5 valores
16. 1 valor
17. 1 valor
18. 1 valor
19. 0.5 valores
20. 1 valor
21. 1 valor

### Instruções do microprocessador da INTEL 8085

Nomenclatura:

#### LEGENDA

pr	Par de registos: HL, BC, DE, SP, PC
reg	Registo: A, B, C, D, E, H, L
M	Posição de memória

 <b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

addr Endereço de 16 bits de uma posição de memória

x O bit do registo de flags é afetado

byte Constante, ou expressão lógica/aritmética que representa um dado de 8 bits

double Constante, ou expressão lógica/aritmética que representa um dado de 16 bits

[] Conteúdo do que se encontra dentro de parênteses retos

[[ ]] Conteúdo do conteúdo do que se encontra dentro de parênteses retos

CS Flag de carry

label Endereço de uma posição de memória

port Endereço de um dispositivo I/O

### Grupo de transferência de dados

INSTRUÇÃO	OPERANDOS	STATUS DO REGISTO DE FLAGS					OPERAÇÃO REALIZADA
		CS	AC	Z	S	P	
LDAX	pr						$[A] \leftarrow [[pr]]$ Load A using implied addressing by BC (pr=B) or DE (pr=D)
STAX	pr						$[[pr]] \leftarrow [A]$ Store A using implied addressing by BC (pr=B) or DE (pr=D)
MOV	r,M						$[r] \leftarrow [[HL]]$ Load any register using implied addressing by HL
MOV	M,r						$[[HL]] \leftarrow [r]$ Store any register using implied addressing by HL
LDA	addr						$[A] \leftarrow [addr]$ Load A using direct addressing
STA	addr						$[addr] \leftarrow [A]$ Store A using direct addressing
LHLD	addr						$[L] \leftarrow [addr]$ and $[H] \leftarrow [addr+1]$ Load H and L registers using direct addressing
SHLD	addr						$[addr] \leftarrow [L]$ and $[addr+1] \leftarrow [H]$ Store H and L registers using direct addressing
MOV	r,r						$[r] \leftarrow [r]$ Move any register to any register
XCHG							$[D] \leftrightarrow [H]$ and $[E] \leftrightarrow [L]$ Exchange DE with HL
SPHL							$[HL] \leftarrow [SP]$ Move HL to SP
LXI	pr,double						$[pr] \leftarrow \text{double}$ Load 16 bits immediate data into BC (pr=B), DE (pr=D), HL (pr=H), SP (pr=SP)

<b>P.PORTO</b>	<b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
		Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
		Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

MVI                      M,byte     $[[HL]] \leftarrow \text{byte}$   
Load 8 bit immediate data into memory location with implied addressing by HL

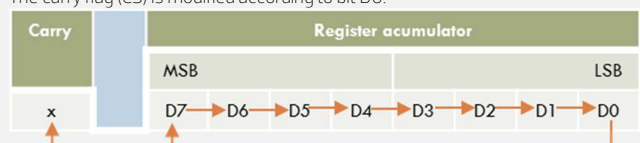
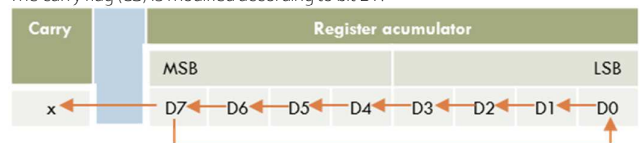
MVI                      r,byte     $[r] \leftarrow \text{byte}$   
Load 8 bit immediate data into any register

### Grupo aritmético, lógico e de rotação

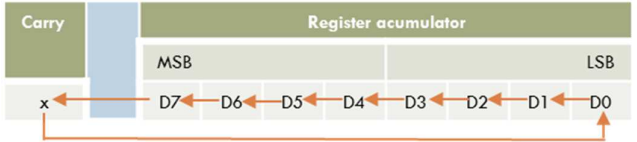
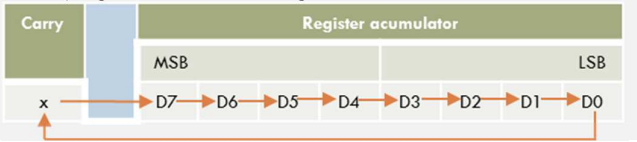
INSTRUÇÃO	OPERANDOS	STATUS DO REGISTO DE FLAGS					OPERAÇÃO REALIZADA
		CS	AC	Z	S	P	
ADD	M	x	x	x	x	x	$[A] \leftarrow [A] + [[HL]]$ Add register A with implied addressing by HL and store the result in register A
ADC	M	x	x	x	x	x	$[A] \leftarrow [A] + [[HL]] + [CS]$ Add register A with carry with implied addressing by HL and store the result in register A
SUB	M	x	x	x	x	x	$[A] \leftarrow [A] - [[HL]]$ Subtract register A with implied addressing by HL and store the result in register A
SBB	M	x	x	x	x	x	$[A] \leftarrow [A] - [[HL]] - [CS]$ Subtract register A with carry with implied addressing by HL and store the result in register A
ANA	M	0	1	x	x	x	$[A] \leftarrow [A] \text{ AND } [[HL]]$ AND between register A with implied addressing by HL and store the result in register A
XRA	M	0	0	x	x	x	$[A] \leftarrow [A] \text{ XOR } [[HL]]$ Exclusive-OR between register A with implied addressing by HL and store the result in register A
ORA	M	0	0	x	x	x	$[A] \leftarrow [A] \text{ OR } [[HL]]$ OR between register A with implied addressing by HL and store the result in register A
CMP	M	x	x	x	x	x	$[A] - [[HL]]$ Compare register A with implied addressing by HL If register A < $[[HL]]$ then the carry flag is set (1) If register A = $[[HL]]$ then the zero flag is set (1) If register A > $[[HL]]$ then the carry and zero flags are reset (0)
INR	M	x	x	x	x	x	$[[HL]] \leftarrow [[HL]] + 1$ Increment memory
DCR	M	x	x	x	x	x	$[[HL]] \leftarrow [[HL]] - 1$ Decrement memory
ADI	byte	x	x	x	x	x	$[A] \leftarrow [A] + \text{byte}$ Add register A with 8 bit immediate data and store the result in register A
ACI	byte	x	x	x	x	x	$[A] \leftarrow [A] + \text{byte} + [CS]$ Add register A with 8 bit immediate data with carry and store the result in register A
SUI	byte	x	x	x	x	x	$[A] \leftarrow [A] - \text{byte}$ Subtract register A with 8 bit immediate data and store the result in register A
SBI	byte	x	x	x	x	x	$[A] \leftarrow [A] - \text{byte} - [CS]$ Subtract register A with 8 bit immediate data with carry and store the result in register A
ANI	byte	0	1	x	x	x	$[A] \leftarrow [A] \text{ AND byte}$ AND between register A with 8 bit immediate data and store the result in register A
XRI	byte	0	0	x	x	x	$[A] \leftarrow [A] \text{ XOR byte}$ Exclusive-OR between register A with 8 bit immediate data and store the result in register A
ORI	byte	0	0	x	x	x	$[A] \leftarrow [A] \text{ OR byte}$ OR between register A with 8 bit immediate data and store the result in register A

<b>P.PORTO</b>	<b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
		Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
		Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

CPI	byte	x	x	x	x	x	[A] ← byte Compare register A with 8 bit immediate data If register A < byte then the carry flag is set (1) If register A = byte then the zero flag is set (1) If register A > byte then the carry and zero flags are reset (0)
ADD	r	x	x	x	x	x	[A] ← [A] + [r] Add register A with any register and store the result in register A
ADC	r	x	x	x	x	x	[A] ← [A] + [r] + [CS] Add register A with any register with carry and store the result in register A
SUB	r	x	x	x	x	x	[A] ← [A] - [r] Subtract register A with any register and store the result in register A
SBB	r	x	x	x	x	x	[A] ← [A] - [r] - [CS] Subtract register A with any register with carry and store the result in register A
ANA	r	0	1	x	x	x	[A] ← [A] AND [r] AND between register A with any register and store the result in register A
XRA	r	0	0	x	x	x	[A] ← [A] XOR [r] Exclusive-OR between register A with any register and store the result in register A
ORA	r	0	0	x	x	x	[A] ← [A] OR [r] OR between register A with any register and store the result in register A
CMP	r	x	x	x	x	x	[A] - [r] Compare register A with any register If register A < r then the carry flag is set (1) If register A = r then the zero flag is set (1) If register A > r then the carry and zero flags are reset (0)
INR	r		x	x	x	x	[r] ← [r] + 1 Increment any register
DCR	r		x	x	x	x	[r] ← [r] - 1 Decrement any register
CMA							[A] ← [A] Complement register A
DAA		x	x	x	x	x	The contents of the accumulator are changed from a binary value to two 4-bit binary coded decimal (BCD) digits. This is the only instruction that uses the auxiliary flag (AC) to perform the binary to BCD conversion, and the conversion procedure is described below. If the value of the low-order 4-bits in the accumulator is greater than 9 or if AC flag is set (1), the instruction adds 6 to the low-order four bits. If the value of the high-order 4-bits in the accumulator is greater than 9 or if the Carry flag (CS) is set (1), the instruction adds 6 to the high-order four bits.
RLC		x					Each binary bit of the register accumulator is rotated left by one position. Bit D7 is placed in the position of D0 as well as in the Carry flag. The carry flag (CS) is modified according to bit D7.
RRC		x					Each binary bit of the register accumulator is rotated right by one position. Bit D0 is placed in the position of D7 as well as in the Carry flag. The carry flag (CS) is modified according to bit D0.
RAL		x					Each binary bit of the register accumulator is rotated left by one position through the carry flag.




<b>P.PORTO</b> <b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

			<p>Bit D<sub>7</sub> is placed in the carry flag, and the carry flag is placed in the least significant position D<sub>0</sub>. The carry flag (CS) is modified according to bit D<sub>7</sub>.</p> 
RAR		x	<p>Each binary bit of the register acumulador is rotated right by one position through the carry flag. Bit D<sub>0</sub> is placed in the carry flag, and the carry flag is placed in the least significant position D<sub>7</sub>. The carry flag (CS) is modified according to bit D<sub>0</sub>.</p> 
DAD	pr	x	<p><math>[HL] \leftarrow [HL] + [pr]</math> Add HL to a register pair BC (pr=B), DE (pr=D), HL (pr=H), SP (pr=SP) and store the result in HL</p>
INX	pr		<p><math>[pr] \leftarrow [pr] + 1</math> Increment register pair BC (pr=B), DE (pr=D), HL (pr=H), SP (pr=SP) and store the result in a register pair BC (pr=B), DE (pr=D), HL (pr=H), SP (pr=SP)</p>
DCX	pr		<p><math>[pr] \leftarrow [pr] - 1</math> Decrement register pair BC (pr=B), DE (pr=D), HL (pr=H), SP (pr=SP) and store the result in a register pair BC (pr=B), DE (pr=D), HL (pr=H), SP (pr=SP)</p>

### Grupo de controlo e de salto

INSTRUÇÃO	OPERANDOS	STATUS DO REGISTO DE FLAGS					OPERAÇÃO REALIZADA
		CS	AC	Z	S	P	
JMP	label						$[PC] \leftarrow \text{label}$ Jump to instruction at address label
PCHL							$[PC] \leftarrow [HL]$ Jump to instruction at address contained in HL
CALL	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label
CC	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the carry flag (CS) equal to 1
CNC	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the carry flag (CS) equal to 0
CZ	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the zero flag (Z) equal to 1
CNZ	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the zero flag (Z) equal to 0
CP	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the sign flag (S) equal to 0
CM	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the sign flag (S) equal to 1
CPE	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the parity flag (P) equal to 1
CPO	label						$[SP] \leftarrow [PC], [PC] \leftarrow \text{label}, [SP] \leftarrow [SP] - 2$ Jump to subroutine starting at address label if the parity flag (P) equal to 0



 <b>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</b>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

RET		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine																								
RC		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the carry flag (CS) equal to 1																								
RNC		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the carry flag (CS) equal to 0																								
RZ		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the zero flag (Z) equal to 1																								
RNZ		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the zero flag (Z) equal to 0																								
RM		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the sign flag (S) equal to 0																								
RP		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the sign flag (S) equal to 1																								
RPE		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the parity flag (P) equal to 1																								
RPO		$[PC] \leftarrow [SP], [SP] \leftarrow [SP] + 2$ Return from subroutine if the parity flag (P) equal to 0																								
JC	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the carry flag (CS) equal to 1																								
JNC	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the carry flag (CS) equal to 0																								
JZ	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the zero flag (Z) equal to 1																								
JNZ	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the zero flag (Z) equal to 0																								
JP	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the sign flag (S) equal to 0																								
JM	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the sign flag (S) equal to 1																								
JPE	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the parity flag (P) equal to 1																								
JPO	label	$[PC] \leftarrow \text{label}$ Jump to instruction at address label if the parity flag (P) equal to 0																								
RST	n	<p>The RST instruction is equivalent to a 1-byte call instruction to one of eight memory locations depending upon the number. The instructions are generally used in conjunction with interrupts and inserted using external hardware. However these can be used as software instructions in a program to transfer program execution to one of the eight locations. The addresses are:</p> <p>Instruction Restart Address</p> <table> <tr><td>RST</td><td>0</td><td>0000H</td></tr> <tr><td>RST</td><td>1</td><td>0008H</td></tr> <tr><td>RST</td><td>2</td><td>0010H</td></tr> <tr><td>RST</td><td>3</td><td>0018H</td></tr> <tr><td>RST</td><td>4</td><td>0020H</td></tr> <tr><td>RST</td><td>5</td><td>0028H</td></tr> <tr><td>RST</td><td>6</td><td>0030H</td></tr> <tr><td>RST</td><td>7</td><td>0038H</td></tr> </table>	RST	0	0000H	RST	1	0008H	RST	2	0010H	RST	3	0018H	RST	4	0020H	RST	5	0028H	RST	6	0030H	RST	7	0038H
RST	0	0000H																								
RST	1	0008H																								
RST	2	0010H																								
RST	3	0018H																								
RST	4	0020H																								
RST	5	0028H																								
RST	6	0030H																								
RST	7	0038H																								

<b>P.PORTO</b> <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame da época normal	Ano letivo 2020/2021	Data 24-06-2021
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Sistemas digitais e arquitetura de computadores	Duração 02:00 horas	

INSTRUÇÃO	OPERANDOS	STATUS DO REGISTO DE FLAGS					OPERAÇÃO REALIZADA
		CS	AC	Z	S	P	
IN	port						$[A] \leftarrow [port]$ Input to register accumulator (A) from I/O port
OUT	port						$[port] \leftarrow [A]$ Output from register accumulator (A) to I/O port
PUSH	pr						$[[SP]] \leftarrow [pr], [SP] \leftarrow [SP] - 2$ Push register pair BC (pr=B), DE (pr=D), H (pr=HL), PSW (pr=PSW) contents onto stack
POP	pr						$[pr] \leftarrow [[SP]], [SP] \leftarrow [SP] + 2$ Pop stack into register pair BC (pr=B), DE (pr=D), H (pr=HL), PSW (pr=PSW)
XTHL							$[HL] \leftrightarrow [[SP]]$ Exchange HL with top of stack
EI							Enable interrupts following execution of next instruction
DI							Disable interrupts
SIM							Set interrupt mask
RIM							Read interrupt mask
NOP							$[PC] \leftarrow [PC] + 1$ No operation but program counter (PC) is incremented
HLT							HALT Stop CPU operation