Prof. Roberto de Matos roberto.matos@ifsc.edu.br

## **BIP II**

- 1. Descreva as modificações feitas na organização do BIP I para o BIP II?
- 2. O que significa um rótulo no código assembly?
- 3. Considerando H, J e i variáveis de 8 *bits*, escreva o trecho de código abaixo utilizando o *Assembly* do BIP2.

```
IF (i==3){
    H = H + 2;
} ELSE {
    H = H + 1;
}

J = H + 1;
// Fim do programa, pare o processador.
```

```
.text
init:
     LD I
     SUBI 3
     BNE else
then:
     LD H
     ADDI 2
     STO H
     JMP fim
else:
     LD H
     ADDI 1
     STO H
fim:
     ADDI 1
     STO J
     HLT
```

- 4. Responda as questões considerando o programa abaixo para o BIP II e as seguintes características:
  - Os endereços das variáveis X, Y e W são 0x3C, 0x3D e 0x3E, respectivamente.
  - Os valores iniciais das variáveis X, Y e W são 5, 5 e 0, respectivamente.

```
1 LD X
2 SUB Y
3 BEQ EQUAL
4 LDI 1
5 JMP FIM
6 EQUAL:
7 LDI 2
8 FIM:
9 ADD X
10 STO W
11 HLT
```

IFSC – CAMPUS SÃO JOSÉ Página 1 de 5

a) Demonstre em linguagem de alto nível o que o programa Assembly está fazendo.

```
if (x == y){
W = x + 2;
W = x + 1;
```

b) Monte o programa.

	Endereço	Conteúdo (Hex)			
	0	103C	LD X (3C) -	Mnemonic	Op. Code
			1 ` ´	HLT	00000
	1	303D	SUB Y (3D)	STO	00001
	2	4005	BEQ EQUAL (5)	LD	00010
	3	1801	LDI 1	LDI	00011
	4	7006	JMP FIM (6)	ADD	00100
E01141			LDI2	ADDI	00101
EQUAL:	5	1802		SUB	00110
FIM:	6	203C	ADD X (3C)	SUBI	00111
	7	083E	STO W (3E)	BEQ	01000
	8	0000	HLT	BNE	01001
	<u> </u>			BGT	01010
				BGE	01011
				BLT	01100
				BLE	01101
				JMP	01110

c) Demonstre os valores solicitados abaixo a cada ciclo de *clock*:

clk				_						
PC	0	1	2	2 X	5	6	7	8	8	8
ACC	0	5	<u> </u>			2	7	7	7	7
Var. X	5	X	<u></u>							
Var. Y	5	X	<u></u>			X				
Var. W	0	X	<u> </u>			X	0	7		

- 5. Responda as questões considerando o programa abaixo para o BIP II e as seguintes características:
  - Os endereços das variáveis X, Y e W são 0x3C, 0x3D e 0x3E, respectivamente.
  - Os valores iniciais das variáveis X, Y e W são 5, 5 e 0, respectivamente.

```
1 LD X
2 SUB Y
3 BEQ EQUAL
4 LDI 1
5 JMP FIM
6 EQUAL:
7 LDI 2
8 FIM:
9 ADD X
10 STO W
11 HLT
```

a. Demonstre em linguagem C o que o programa Assembly está fazendo.

1		
1		
1		
l .		
1		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
I .		
1		
1		

b. Monte o programa.

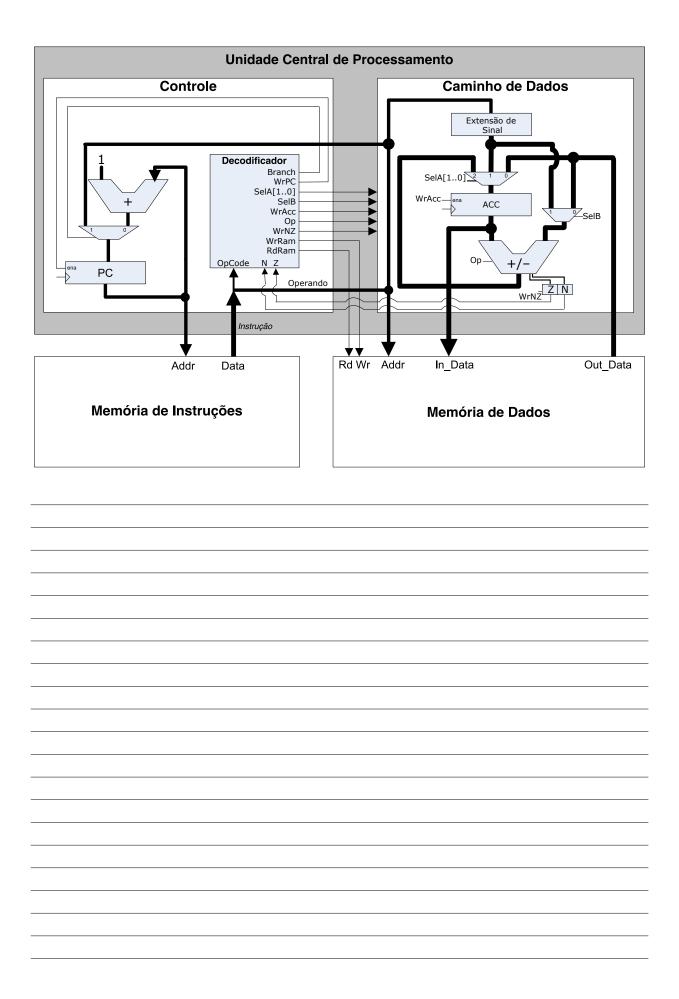
Endereço	Conteúdo (Hex)

Mnemonic	Op. Code
HLT	00000
STO	00001
LD	00010
LDI	00011
ADD	00100
ADDI	00101
SUB	00110
SUBI	00111
BEQ	01000
BNE	01001
BGT	01010
BGE	01011
BLT	01100
BLE	01101
JMP	01110
	•

c. Demonstre os valores dos sinais de controle e dos registradores durante a execução do programa na tabela abaixo.

IFSC – CAMPUS SÃO JOSÉ Página 3 de 5

Registradores (valor esperado após a execução) Sinais de controle (valores para executar a instrução)	Instrução Opcode PC ACC ACC N N Z WrPC SeIA1 SeIA0 SeIB WrACC OpSeI RdRAM					
	WrNZ					
	Branch					



IFSC – CAMPUS SÃO JOSÉ Página 5 de 5