

Universidade de Brasília  
Ingrid Miranda de Sousa  
Matrícula: 14/0143980

### **Microprocessadores e microcontroladores - Aula 5**

Para as questões 2 a 5, considere que as variáveis 'f', 'g', 'h', 'i' e 'j' são do tipo inteiro (16 bits na arquitetura do MSP430), e que o vetor 'A[]' é do tipo inteiro. Estas variáveis estão armazenadas nos seguintes registradores:

**f: R4**

**g: R5**

**h: R6**

**i: R7**

**j: R8**

**A: R9**

Utilize os registradores R11, R12, R13, R14 e R15 para armazenar valores temporários.

**1. Escreva os trechos de código assembly do MSP430 para:**

**(a) Somente setar o bit menos significativo de R5.**

or.w R5, 0001h

**(b) Somente setar dois bits de R6: o menos significativo e o segundo menos significativo.**

or.w R6, 0003h

**(c) Somente zerar o terceiro bit menos significativo de R7.**

and.w R7, 0004h

**(d) Somente zerar o terceiro e o quarto bits menos significativo de R8.**

and.w R8, 000Ch

**(e) Somente inverter o bit mais significativo de R9.**

xor.w R9, 8000h

**(f) Inverter o nibble mais significativo de R10, e setar o nibble menos significativo de R10.**

xor.w R10, F000h

or.w R10, 000Fh

**2. "Traduza" o seguinte trecho de código em C para o assembly do MSP430:**

**if(i>j) f = g+h+10;**

**else f = g-h-10;**

cmp R8, R7; R7 = i e R8 = j

jge SOMA

mov R5, R4; R5 = g e R4 = f

mov R6, R11; R6 = h

add #10, R11

add R11, R4

SOMA:

mov R5, R4

mov R6, R11

add #10, R11

sub R11, R4

jmp EXIT

EXIT:

**3. "Traduza" o seguinte trecho de código em C para o assembly do MSP430:**

**while(save[i]!=k) i++;**

LOOP:

mov R7, R11; R7 = i e R11 = temporário

rla R11

add.w R12, R11; R12 = save

```

cmp 0(R11), R13; R13 = k
jmp EXIT
inc.w R7
jne LOOP
EXIT:

```

**4. "Traduza" o seguinte trecho de código em C para o assembly do MSP430:**

```

for(i=0; i<100; i++) A[i] = i*2;

```

```

mov #0, R7
FOR_MULT2: cmp #100, R7
           jge END_FOR_MULT2
           mov R7, R11; R11 = temporário
           rla R11; i*2
           mov R11, R7(R9)
           inc R7
           jmp FOR_MULT2
END_FOR_MULT2:

```

**5. "Traduza" o seguinte trecho de código em C para o assembly do MSP430:**

```

for(i=99; i>=0; i--) A[i] = i*2;

```

```

mov #99, R7
FOR_MULTIP2: cmp #0, R7
            jnl END_FOR_MULTIP2
            mov R7, R11
            rla R11
            mov R11, R7(R9)
            dec R7
            jmp FOR_MULTIP2
END_FOR_MULTIP2:

```