

Aluno: Mateus Alves da Rocha **Matrícula:** 11/0132661

Data: 29/03/2017

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.

Mov.w #1, R11

Bis.w R5, R11

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

Mov.w #3, R12

Bis.w R6, R12

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

Mov.w #4, R13

Bic.w R7, R13

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

Mov.w #Ch, R14

Bic.w R8, R14

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

Mov.w #80h, R15

Xor.w R9, R15

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

Mov.w #F0h, R11

Xor.w R10, R11

Mov.w #0Fh, R12

Bis.w R10, R12

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;

Mov.w #10, R13

Mov.w R5, R14

cmp R7, R8

jge ELSE_LABEL

Add.w R6, R13

Add.w R5, R13

Mov.w R13, R4

Jmp EXIT

ELSE_LABEL:

Sub.w R13, R14

Sub.w R6, R14

Mov.w R14, R4

EXIT

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

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

Consideração: K está em R15

```
WHILE_LOOP: Mov.w R7, R12      ; R7 = i, R12 = temporário
              Rla R12           ; R12 = 2*i
              Add.w R10, R12    ; R10 = Save, R12 = Save+2*i = &save[i]
              Cmp 0(R12), R15   ; Compara save [i] com k ( R15)
              Jeq EXIT         ; save[i] == k?
              Inc.w R7          ; i++
              Jmp WHILE_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;
```

```
        bic.w #FFh, R7      ; R7 = i = 0;
```

```
FOR_LOOP: cmp R7, #100d
```

```
        Jge EXIT
```

```
        Mov.w R7, R12;
```

```
        Rla R12
```

```
        Mov.w 0(R12), R9
```

```
        Inc.w R7            ; R7 = i++
```

```
        Jmp FOR_LOOP
```

EXIT

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.w #99d, R7          ; R7 = 99 ;
```

```
FOR_LOOP: cmp R7, #0
```

```
        Jge EXIT
```

```
        Mov.w R7, R12;
```

```
        Rla R12
```

```
        dec.w R7          ; R7
```

```
        Jmp FOR_LOOP
```

EXIT