



UNIVERSIDADE FEDERAL DO CEARÁ - CAMPUS SOBRAL

CURSO DE ENGENHARIA DA COMPUTAÇÃO

DISCIPLINA: MICROPROCESSADORES

PROFESSOR: MARCELO SOUZA

EXERCÍCIO 03 - SOMA DE BITS MAIS E MENOS SIGNIFICATIVOS

YARA MARIA SANTOS MORAIS - 475867

Sobral - CE

2023

O exercício 03 requer que seja feita O cálculo da soma dos 4 bits menos significativos com os 4 bits mais significativos de uma variável de memória de 8 bits e que resultado seja armazenado em outra variável.

O número escolhido foi 5A que em decimal hexadecimal corresponde ao valor 90. De início foi definida uma constante LSB_MASK para isolar os bits menos significativos, após foram declaradas as seguintes variáveis: variável que armazenou o número de 8 bits (BYTE_VAR), uma para armazenar os bits mais e menos significativos, MSB_VAR e LSB_VAR respectivamente e uma para guardar o valor da soma (RESULT_VAR).

Na figura 01 podemos observar a definição de todo o PORTB como saída e a definição do valor da variável de 8 bits.

Figura 01 – Código no MPLABX

```
#include "p18F4550.inc"

; Definição de constantes
LSB_MASK EQU 0x0F ; Máscara para isolar os 4 bits menos significativos

; Definição de variáveis
BYTE_VAR EQU 0x20 ; Variável de 8 bits
MSB_VAR EQU 0x21 ; Variável para armazenar os 4 bits mais significativos
LSB_VAR EQU 0x22 ; Variável para armazenar os 4 bits menos significativos
RESULT_VAR EQU 0x23 ; Variável para armazenar o resultado da soma

ORG 0x0000
GOTO main

ORG 0x0008
RETFIE

ORG 0x0018
main:
    ; Configuração de portas
    BANKSEL TRISB
    CLRF TRISB ; Define todo o PORTB como saída

    ; Inicialização da variável de 8 bits
    BANKSEL BYTE_VAR
    MOVLW 0x5A ; Valor da variável de 8 bits
    MOVWF BYTE_VAR
```

Fonte: Autor

Após todas as inicializações, foi feito o isolamento dos bits menos e mais significativos por meio das funções já descritas anteriormente e foram armazenados nas variáveis MOVWF MSB_VAR e MOVWF LSB_VAR respectivamente. Com os bits separados, foram carregados para o registrador W (MOVF MSB_VAR, W), feita a soma (ADDWF LSB_VAR, W) e guardado o valor em MOVWF RESULT_VAR como pode ser visto na figura 02.

Figura 02 – Código no MPLABX

```

; Isolar os 4 bits mais significativos
BANKSEL BYTE_VAR
;RRF    BYTE_VAR, W      ; Rotação para a direita para trazer os 4 MSBs para o byte menos significativo
ANDLW   LSB_MASK        ; Máscara para manter apenas os 4 bits menos significativos

; Armazenar os 4 bits mais significativos em uma variável
BANKSEL MSB_VAR
MOVWF   MSB_VAR

; Isolar os 4 bits menos significativos
BANKSEL BYTE_VAR
ANDLW   LSB_MASK        ; Máscara para manter apenas os 4 bits menos significativos

; Armazenar os 4 bits menos significativos em uma variável
BANKSEL LSB_VAR
MOVWF   LSB_VAR

; Soma dos 4 bits mais significativos com os 4 bits menos significativos
BANKSEL RESULT_VAR
MOVWF   MSB_VAR, W      ; Carrega o valor dos 4 bits mais significativos para o registrador W
ADDWF   LSB_VAR, W      ; Soma o valor dos 4 bits menos significativos ao registrador W
MOVWF   RESULT_VAR      ; Armazena o resultado na variável de resultado

; Loop infinito
loop:
GOTO    loop

END

```

Fonte: Autor

Executando o programa no MPLABX obtemos o valor escolhido, os bits isolados e o resultado da soma respectivamente. Como pode ser analisado na figura 03.

Figura 03 – Resultado da soma

Address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
020	5A	0A	0A	14	00	00	00	00	00	00	00	00	00	00	00	00
030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Fonte: Autor