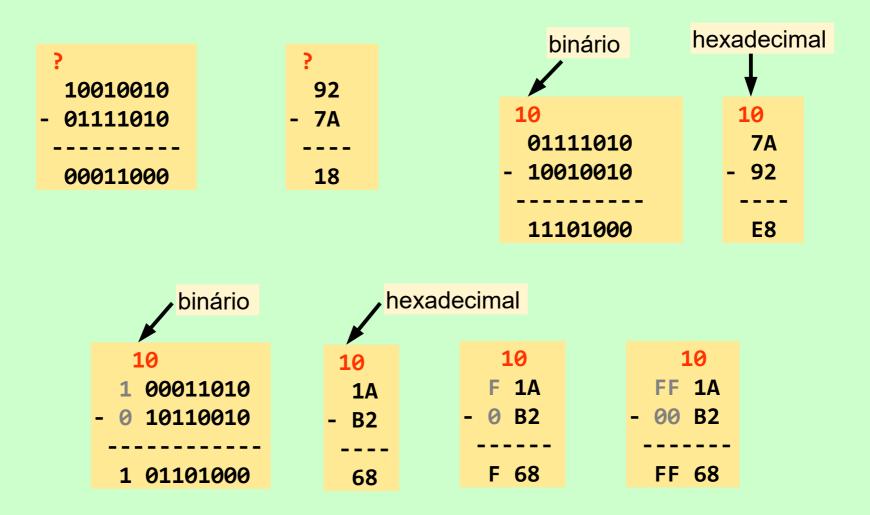


# ELTD03z Microcontroladores/Microprocessadores

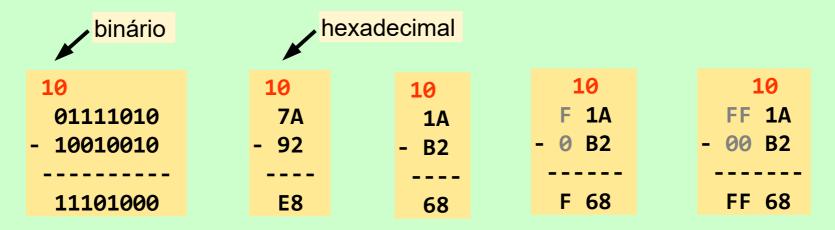
Teoria\_04a1\_3b

Prof. Enio R. Ribeiro

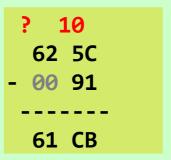
Microcontrolador => registros = 1 byte => subtração fatores de 1 byte => resultado: 2 bytes => subtrair com carry.



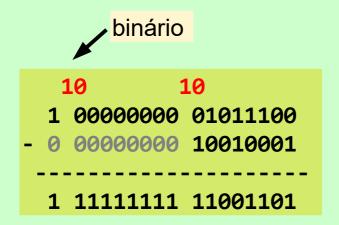
Microcontrolador => registros = 1 byte => subtração fatores de 1 e 2 (de 2) bytes => resultado: 3 bytes => subtrair com carry (subtração de múltipla precisão).

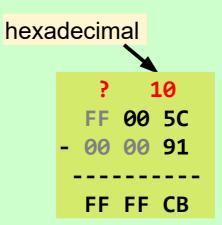


??	LØ
01100010	01011100
-	10010001
01100001	11001101



Microcontrolador => registros = 1 byte => subtração fatores de 1 e 2 (de 2) bytes => resultado: 3 bytes => subtrair com carry (subtração de múltipla precisão).



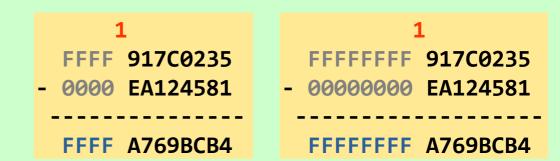


Microcontrolador (STM32F103C8) => registros = 4 bytes (word) => subtração fatores de 4 bytes => resultado 5 bytes => subtrair com carry (sub. múltipla precisão)

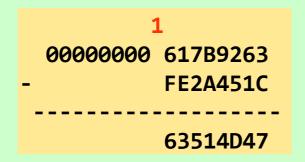
```
917C0235
- EA124581
 A769BCB4
```

```
FF 917C0235
 - 00 EA124581
FF A769BCB4
```





Microcontrolador (STM32F103C8) => registros = 4 bytes (word) => subtração fatores de 4 bytes => resultado 5 bytes => subtrair com carry (sub. múltipla precisão)



```
1
FF 00000000 617B9263
- 00000000 FE2A451C
-----FF FFFFFFF 63514D47
```

```
1
FFFF 00000000 617B9263
- 00000000 FE2A451C
------FFFF FFFFFFF 63514D47
```

```
1
FFFFFFFF 00000000 617B9263
- 00000000 FE2A451C
FFFFFFFF FFFFFFF 63514D47
```

#### Instruction: SUB

The SUB instruction subtracts values of registers and writes the result to the destination register.

### **Syntax**

```
SUB{S} Rd, Rn, Operand2; (Operand2) = Rm
```

- 'Rd' is the destination register
- 'Rn' is the source register
- 'Operand2' is a register for the second operand
- 'S' is an optional suffix. If S is specified, the condition code flags are updated on the result of the operation

#### Instruction: SBC

The SBC instruction subtracts with carry values of registers and writes the result to the destination register.

### **Syntax**

```
SBC{S} Rd, Rn, Operand2 ; (Operand2) = Rm
```

- 'Rd' is the destination register
- 'Rn' is the source register
- 'Operand2' is a register for the second operand
- 'S' is an optional suffix. If S is specified, the condition code flags are updated on the result of the operation

Meta: - Fazer programas para subtrair com carry (subtração múltipla precisão).

**Ex.: 4.7.1** - Faça um programa para subtrair duas variáveis: m e n. As variáveis m e n têm 1 byte. Guarde o resultado em local apropriado. O programa é cíclico. Faça as designações e alocações necessárias.

**Ex.: 4.7.2** - Faça um programa para subtrair duas variáveis: m e n. As variáveis m e n têm, 1 e 2 bytes, respectivamente. Guarde o resultado em local apropriado. O programa é cíclico. Faça as designações e alocações necessárias.

**Ex.: 4.7.3** - Faça um programa para subtrair duas variáveis: m e n. As variáveis m e n têm 4 bytes cada. Guarde o resultado em local apropriado. O programa é cíclico. Faça as designações e alocações necessárias.

**Ex.: 4.7.4** - Faça um programa para subtrair duas variáveis: m e n. As variáveis m e n podem representar valores entre:  $(0 \le m \le 2^{32}$  -1) e  $(0 \le n \le 2^{48}$  - 1) . Guarde o resultado em local apropriado. Faça as designações e alocações necessárias. Programa cíclico.

**Ex.: 4.7.5** - Faça um programa para subtrair duas variáveis: m e n. As variáveis m e n podem representar valores entre:  $(0 \le m \le 2^{32} - 1)$  e  $(0 \le n \le 2^{64} - 1)$ . Guarde o resultado em local apropriado. Faça as designações e alocações necessárias. Programa cíclico.