

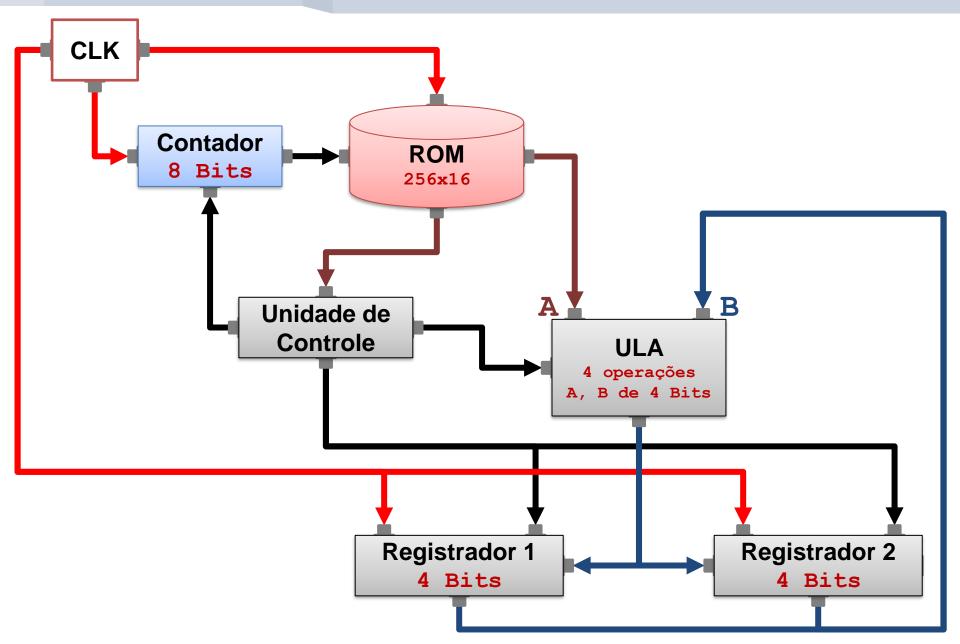
Universidade de São Paulo Instituto de Ciências Matemáticas e de Computação Departamento de Sistemas de Computação

SSC108 Prática em Sistemas Digitais GE4Bio – Grupo de Estudos em Sinais Biológicos

Projeto CPU - 02

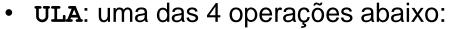
Prof.Dr. Danilo Spatti

São Carlos

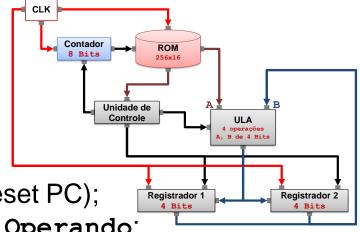


| 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
|------|----|------|----|-----|----|-----|----|------------|----|----|----|----------|----|----|----|
| RgTO | | RgIN | | JMP | | ULA | | Disponível | | | | Operando | | | |

- RgTO: Registrador de destino (4 bits)
- RgIN: Registrador de origem (4 bits)
- JMP: uma das 4 opções abaixo:
 - 00: Operação de ULA;
 - 01: Reinicia Registradores;
 - 10: Reinicia o contador de memória (reset PC);
 - 11: Jump para posição de memória do Operando;



- 00: RgTO ← RgIN + Operando;
- 01: **RgTO** ← **Operando** * 2;
- 10: RgTO ← RgIN Operando;
- 11: RgTO ← Operando / 2;
- Disponível: Disponível para melhorar a CPU caso queiram
- Operando: 4 bits diretamente da memória na ULA



SSC108

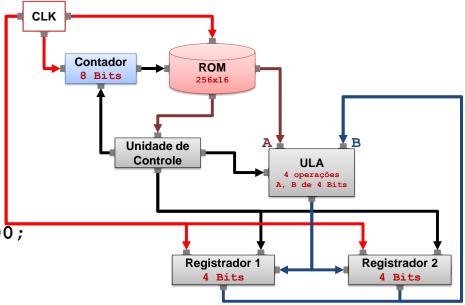
Projeto CPU – Memória 1

```
WIDTH=16;
DEPTH=256;
                                       RqT0
ADDRESS RADIX=UNS;
DATA RADIX=BIN;
CONTENT BEGIN
        0
                 011000000000101;
                 100100010000001;
                 0110001000000010;
                 0110001100000100;
                 0000110000001000;
        [5..7]
                    0000000000000000;
        8
                 0000010000000000;
                 011000000000001;
        10
                 100100000000001;
        11
                 100100000000010;
        12
                 0110001100000010;
        13
                 0110000100000010;
        14
                 1001001100000010;
        15
                 0110001000000000;
        16
                 011000000000001;
        17
                 00001000000000000;
        [18..255]
                       END;
```

```
10
                        09
                            08
                                     06 05 04
                                                  03 02 01 00
       13 | 12
               11
                                 07
       RqIN
                                 Disponível
                                                    Operando
                         ULA
                 JMP
 RgTO: Registrador de destino (4 bits)
 RgIN: Registrador de origem (4 bits)
 JMP: uma das 4 opções abaixo:
   • 00: Operação de ULA;
   • 01: Reinicia Registradores;
   • 10: Reinicia o contador de memória (reset PC):
   • 11: Jump para posição de memória do Operando:

    ULA: uma das 4 operações abaixo:

   • 00: RqTO ← RqIN + Operando:
   • 01: RgTO ← Operando * 2;
   • 10: RgTO ← RgIN - Operando;
   • 11: RqTO ← Operando / 2:
• Disponível: Disponível para melhorar a CPU caso queiram
  Operando: 4 bits diretamente da memória na ULA
```



WIDTH=16;

END;

SSC108

Projeto CPU – Memória 2

03 02 01 00

Operando

Registrador 2

4 Bits

4 Bits

```
10
                                                                          09
                                                                             08
                                                                                    06 05 04
                                                            13 | 12
                                                                   11
                                                                                 07
DEPTH=256;
                                                      RqT0
                                                             RqIN
                                                                                 Disponível
                                                                           ULA
                                                                     JMP
                                                        RgTO: Registrador de destino (4 bits)
ADDRESS RADIX=UNS;
                                                        RgIN: Registrador de origem (4 bits)
DATA RADIX=BIN;
                                                        JMP: uma das 4 opções abaixo:
                                                          • 00: Operação de ULA;
                                                          • 01: Reinicia Registradores;
CONTENT BEGIN
                                                          • 10: Reinicia o contador de memória (reset PC):
                                                          • 11: Jump para posição de memória do Operando:
           0
                        011000000000100;
                        1001000100000010;

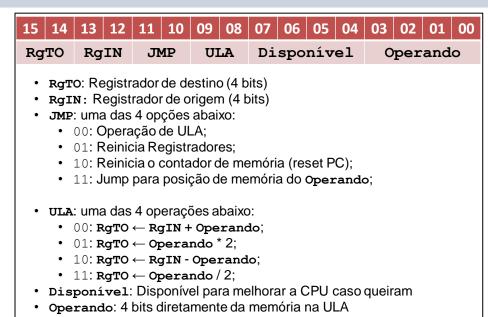
    ULA: uma das 4 operações abaixo:

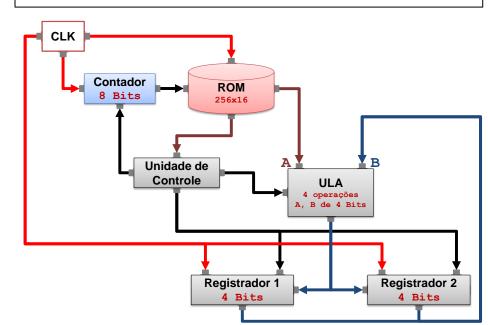
                                                          • 00: RqTO ← RqIN + Operando:
                        0110001000000001;
                                                          • 01: RgTO ← Operando * 2;
                        0110001100000101;
                                                          • 10: RgTO ← RgIN - Operando;
                                                          • 11: RqTO ← Operando / 2:
                        100100000000010;
                                                       • Disponível: Disponível para melhorar a CPU caso queiram
           5
                        0110000000000010;
                                                        Operando: 4 bits diretamente da memória na ULA
           6
                        100100000000011;
                        0000010000000000;
                                                         CLK
           8
                        0000110000001101;
           [9..12]
                             Contador
                                                                             ROM
                                                               8 Bits
                                                                            256x16
           13
                        100100000000010;
           14
                        0110001100000010;
           15
                        0110000100000001;
                                                                    Unidade de
           16
                        1001001100000010;
                                                                     Controle
                                                                                         ULA
                                                                                       4 operações
           17
                        0110001000000000;
                                                                                      A, B de 4 Bits
           18
                        011000000000011;
           19
                        00001000000000000;
           [20..255]
                                00000000000000000;
                                                                           Registrador 1
```

SSC108

Projeto CPU – Memória 1 (Gabarito)

| Count | Memória | | | | R1 | R2 | Instrução |
|-------|---------|---|---|---|----|----|----------------------------|
| 0 | 6 | 0 | 0 | 5 | 5 | 0 | $R1 \leftarrow R2 + 5$ |
| 1 | 9 | 1 | 0 | 1 | 5 | 2 | $R2 \leftarrow 1 \times 2$ |
| 2 | 6 | 2 | 0 | 2 | 0 | 2 | $R1 \leftarrow R2 - 2$ |
| 3 | 6 | 3 | 0 | 4 | 2 | 2 | $R1 \leftarrow 4 \div 2$ |
| 4 | 0 | С | 0 | 8 | 2 | 2 | <i>Jump</i> #8 |
| 8 | 0 | 4 | 0 | 0 | 0 | 0 | <i>CLR R1 e R2</i> |
| 9 | 6 | 0 | 0 | 1 | 1 | 0 | $R1 \leftarrow R2 + 1$ |
| 10 | 9 | 0 | 0 | 1 | 1 | 2 | $R2 \leftarrow R1 + 1$ |
| 11 | 9 | 0 | 0 | 2 | 1 | 3 | $R2 \leftarrow R1 + 2$ |
| 12 | 6 | 3 | 0 | 2 | 1 | 3 | $R1 \leftarrow 2 \div 2$ |
| 13 | 6 | 1 | 0 | 2 | 4 | 3 | $R1 \leftarrow 2 \times 2$ |
| 14 | 9 | 3 | 0 | 2 | 4 | 1 | $R2 \leftarrow 2 \div 2$ |
| 15 | 6 | 2 | 0 | 0 | 1 | 1 | $R1 \leftarrow R2 - 0$ |
| 16 | 6 | 0 | 0 | 1 | 2 | 1 | $R1 \leftarrow R2 + 1$ |
| 17 | 0 | 8 | 0 | 0 | 2 | 1 | CLR contador |

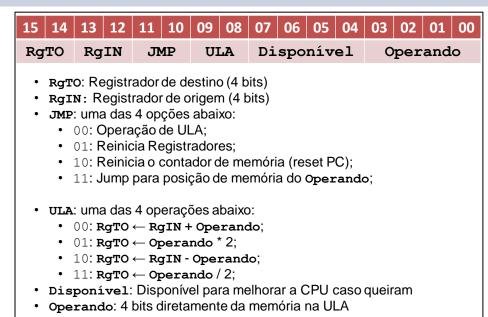


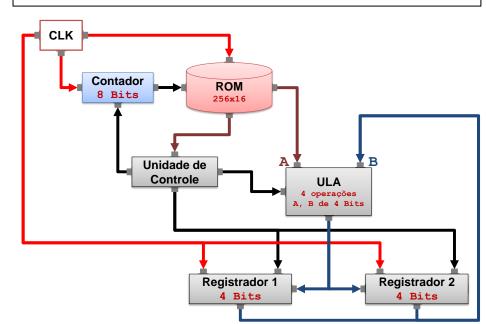


SSC108

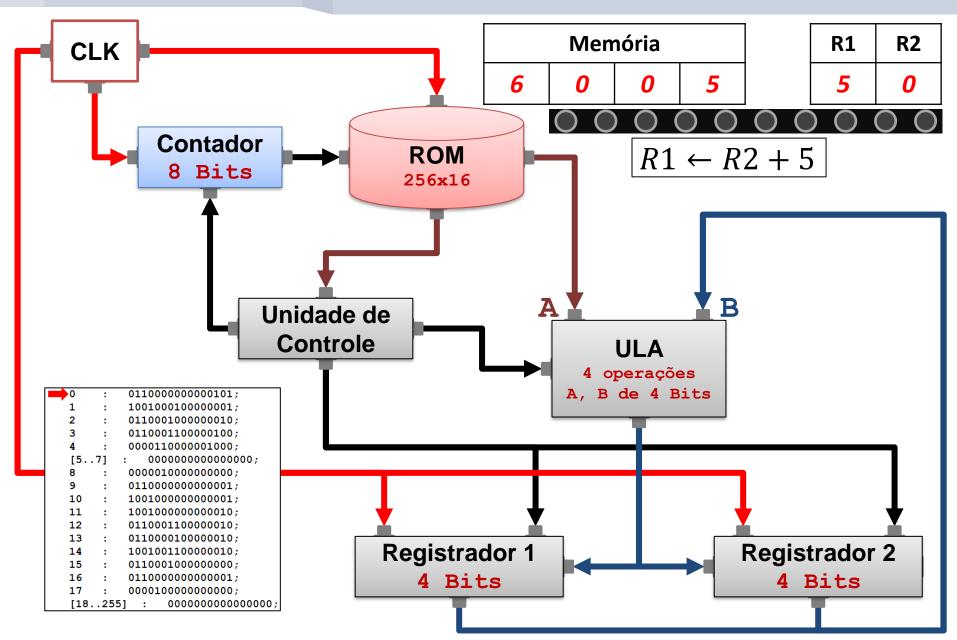
Projeto CPU – Memória 2 (Gabarito)

| Count | Memória | | | | R1 | R2 | Instrução | | | | |
|-------|---------|---|---|---|----|----|----------------------------|--|--|--|--|
| 0 | 6 | 0 | 0 | 4 | 4 | 0 | $R1 \leftarrow R2 + 4$ | | | | |
| 1 | 9 | 1 | 0 | 2 | 4 | 4 | $R2 \leftarrow 2 \times 2$ | | | | |
| 2 | 6 | 2 | 0 | 1 | 3 | 4 | $R1 \leftarrow R2 - 1$ | | | | |
| 3 | 6 | 3 | 0 | 5 | 2 | 4 | $R1 \leftarrow 5 \div 2$ | | | | |
| 4 | 9 | 0 | 0 | 2 | 2 | 4 | $R2 \leftarrow R1 + 2$ | | | | |
| 5 | 6 | 0 | 0 | 2 | 6 | 4 | $R1 \leftarrow R2 + 2$ | | | | |
| 6 | 9 | 0 | 0 | 3 | 6 | 9 | $R2 \leftarrow R1 + 3$ | | | | |
| 7 | 0 | 4 | 0 | 0 | 0 | 0 | <i>CLR R1 e R2</i> | | | | |
| 8 | 0 | С | 0 | d | 0 | 0 | Jump #d | | | | |
| 13 | 9 | 0 | 0 | 2 | 0 | 2 | $R2 \leftarrow R1 + 2$ | | | | |
| 14 | 6 | 3 | 0 | 2 | 1 | 2 | $R1 \leftarrow 2 \div 2$ | | | | |
| 15 | 6 | 1 | 0 | 1 | 2 | 2 | $R1 \leftarrow 1 \times 2$ | | | | |
| 16 | 9 | 3 | 0 | 2 | 2 | 1 | $R2 \leftarrow 2 \div 2$ | | | | |
| 17 | 6 | 2 | 0 | 0 | 1 | 1 | $R1 \leftarrow R2 - 0$ | | | | |
| 18 | 6 | 0 | 0 | 3 | 4 | 1 | $R1 \leftarrow R2 + 3$ | | | | |
| 19 | 0 | 8 | 0 | 0 | 4 | 1 | CLR contador | | | | |





8



4 Bits

4 Bits

14

15

16

17

[18..255] :

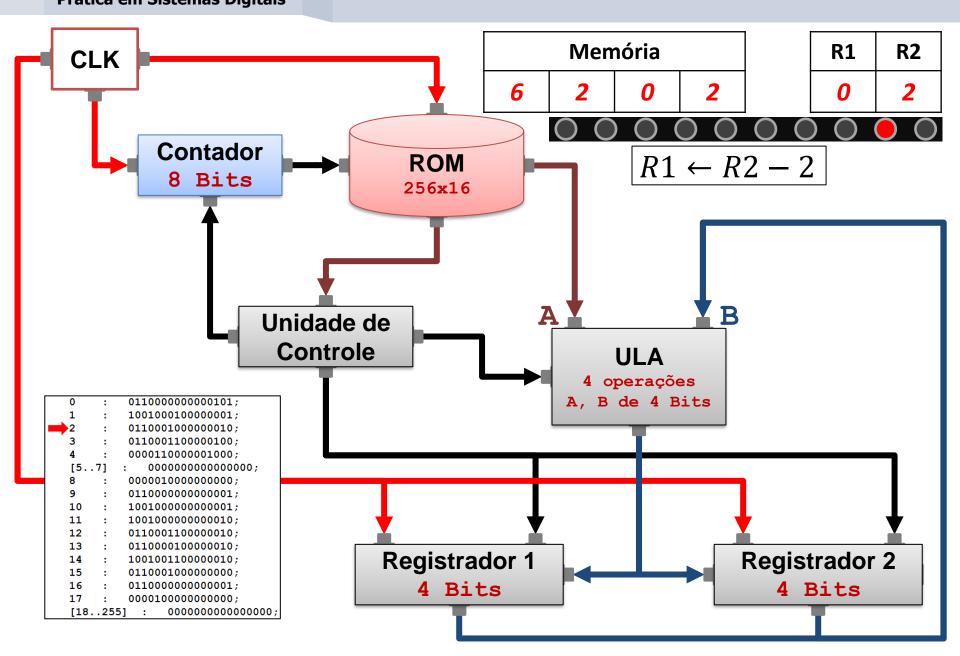
1001001100000010;

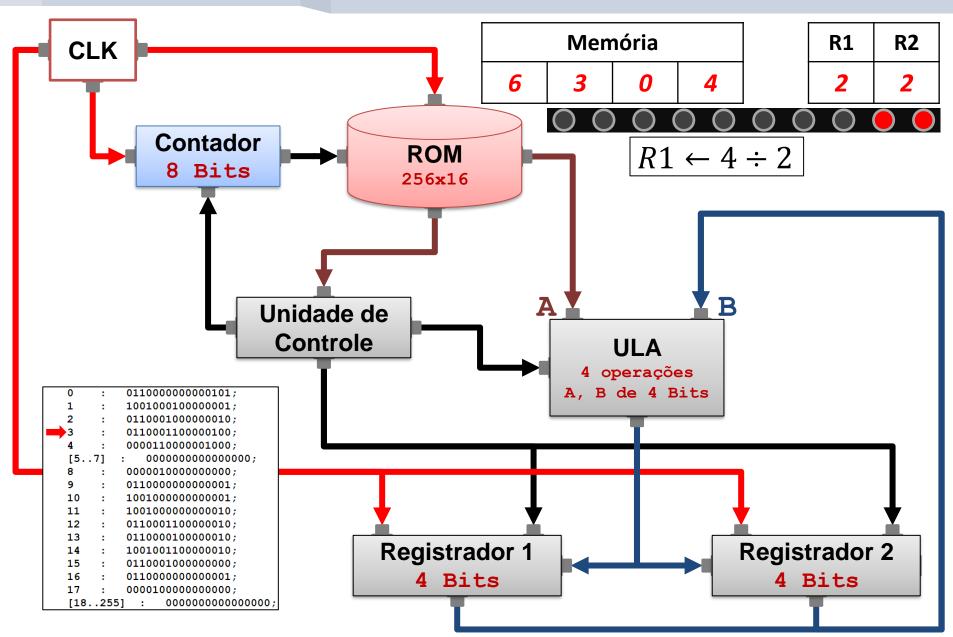
01100010000000000;

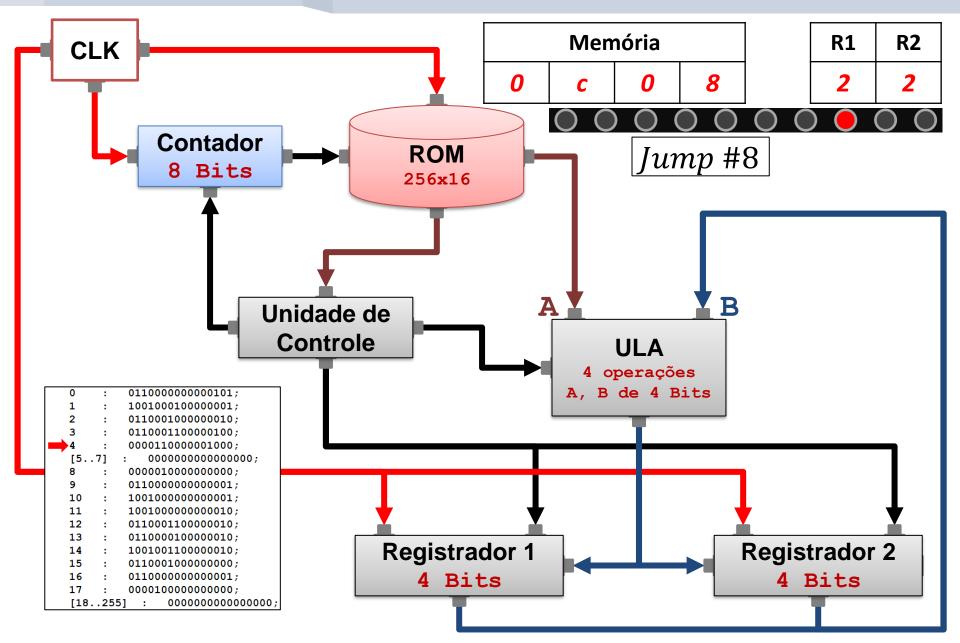
0110000000000001;

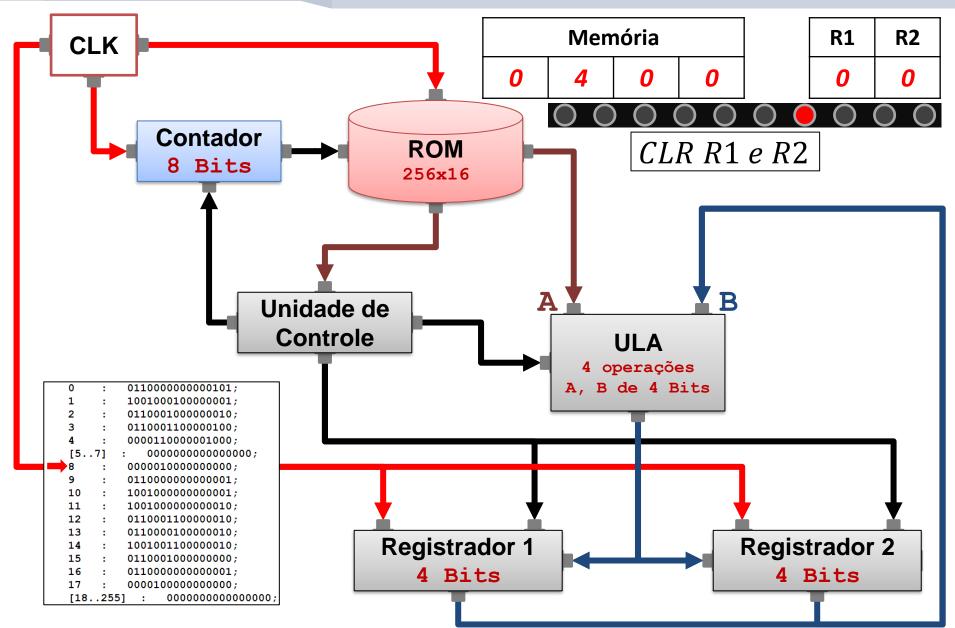
00001000000000000;

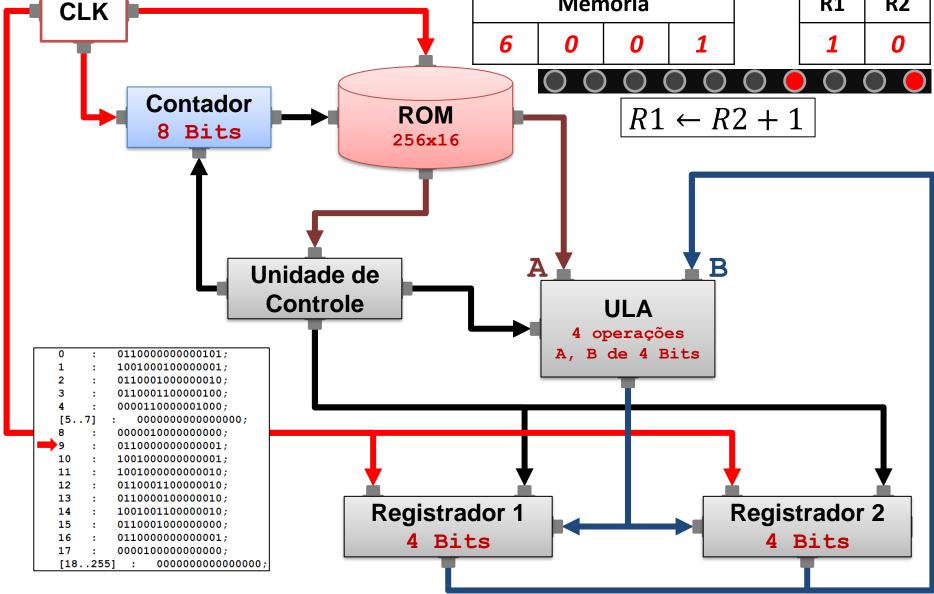
0000000000000000



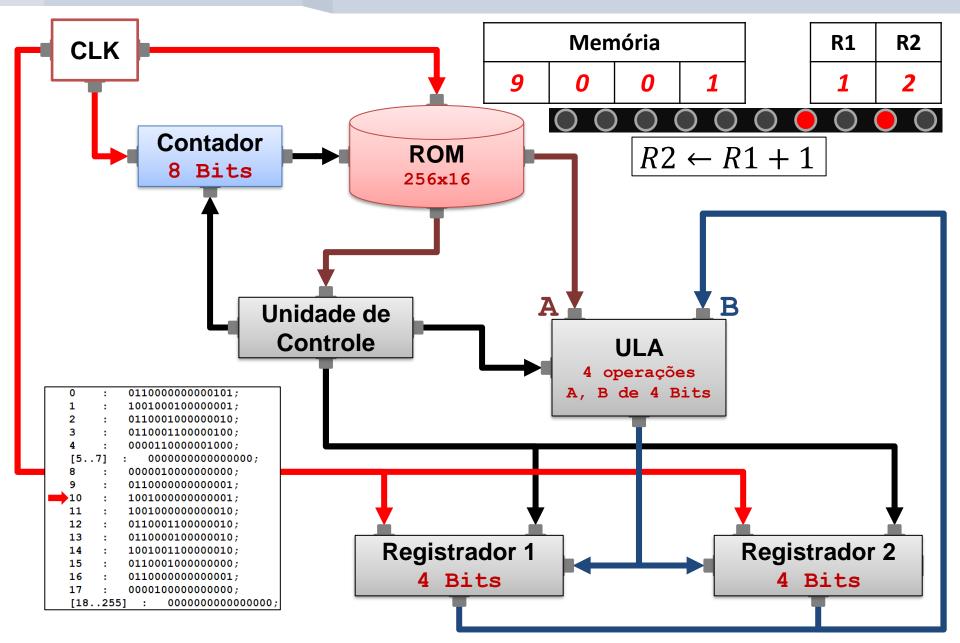








15



Registrador 1

4 Bits

Registrador 2

4 Bits

1001000000000010;

0110001100000010;

0110000100000010;

1001001100000010;

01100010000000000;

0110000000000001;

00001000000000000;

0000000000000000

12

13

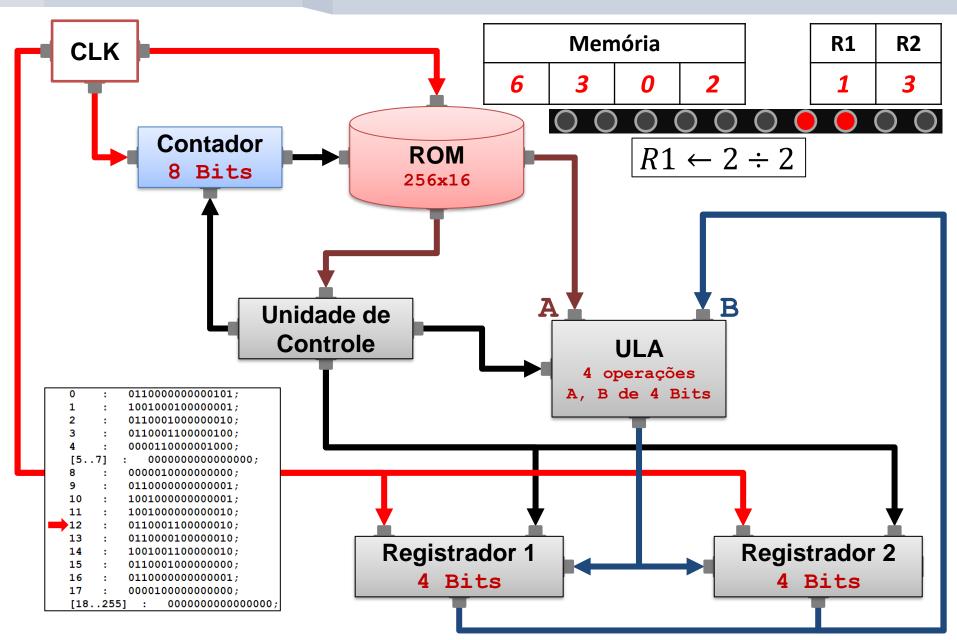
14

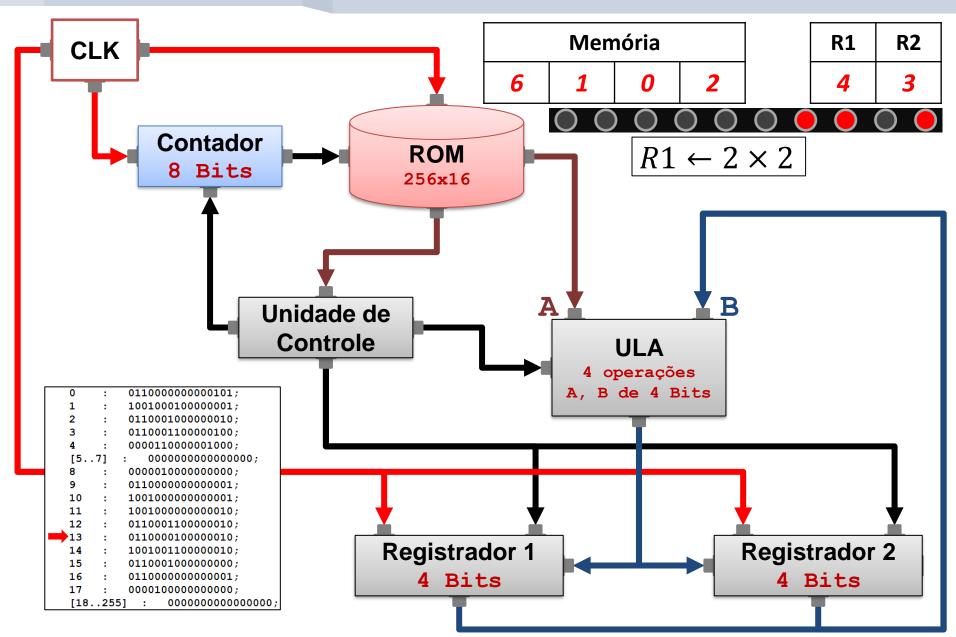
15

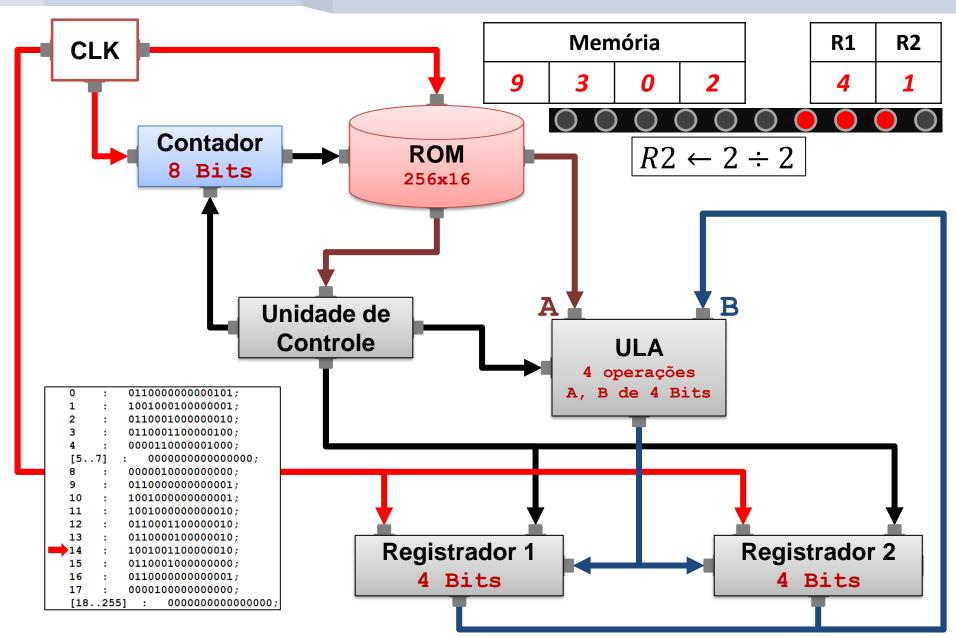
16

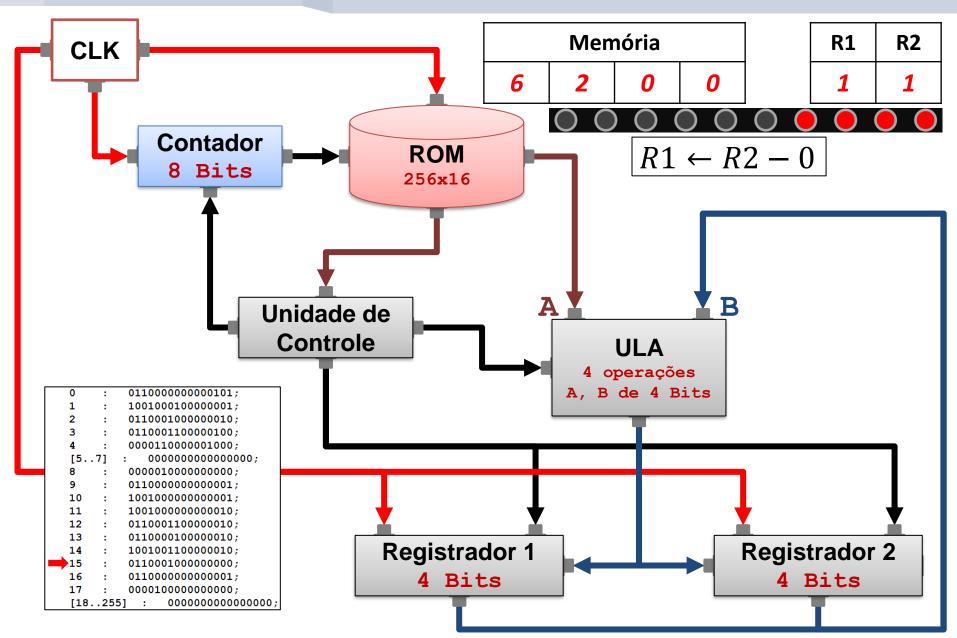
17

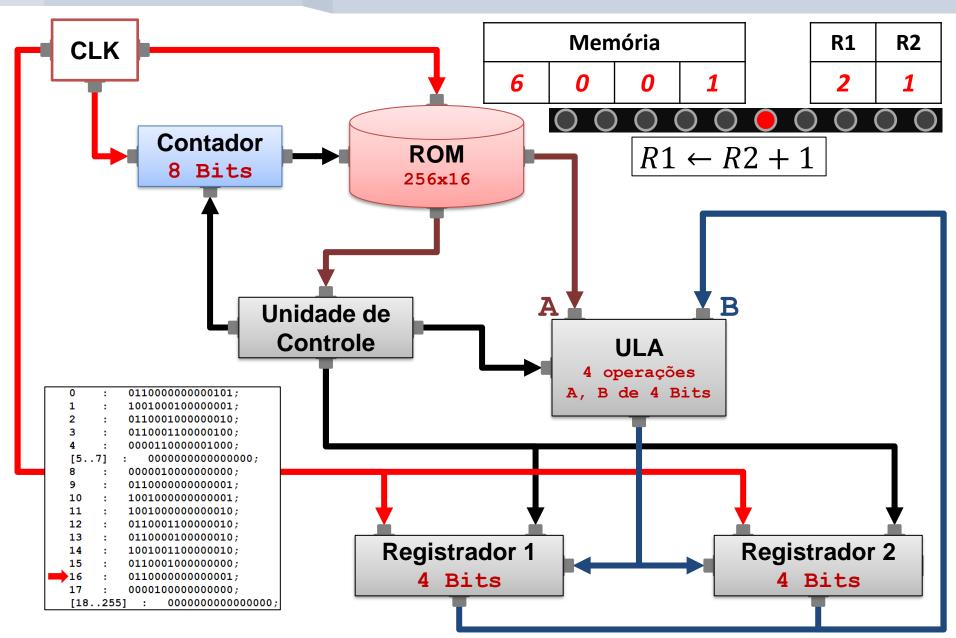
[18..255] :

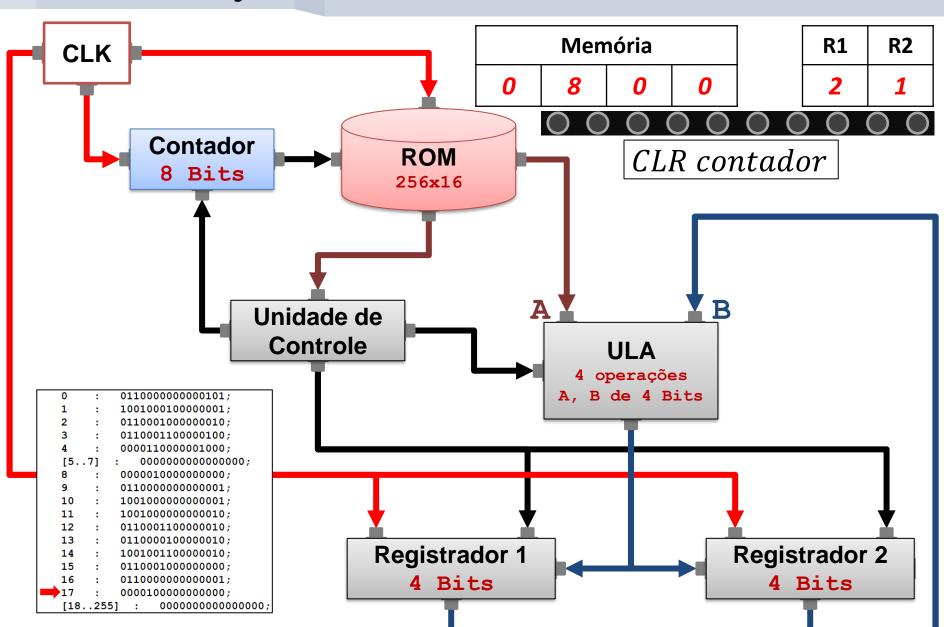


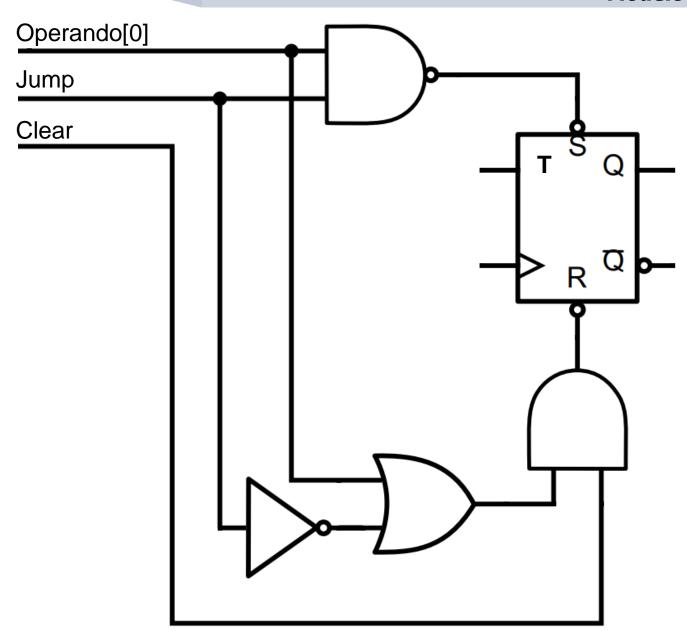












- Memória: Displays 7 seg mais significativos.
- Registradores: Displays 7 seg menos significativos.
- O CLK do circuito deverá ser de agora por diante associado à chave mais significativa disponível (SW9 por exemplo).
- O Master CLR do circuito deverá ser associado de agora por diante à chave menos significativa (SW0).

25

- Realizar as adaptações necessárias ao circuito para poder utilizar o JUMP do contador.
- Atentar para o fato de que o operando do JUMP é de 4 bits.
- Realizar as adaptações necessárias ao circuito para poder integrar os registradores na ULA.
- A lógica da Unidade de Controle deve ser CENTRALIZADA.

spatti@icmc.usp.br

