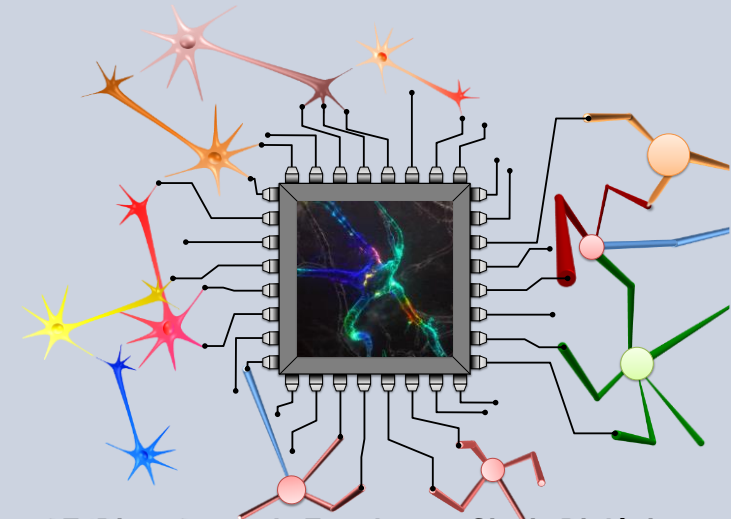


Universidade de São Paulo  
Instituto de Ciências Matemáticas e de Computação  
Departamento de Sistemas de Computação

**SSC108**  
**Prática em Sistemas Digitais**

**Projeto CPU - 03**

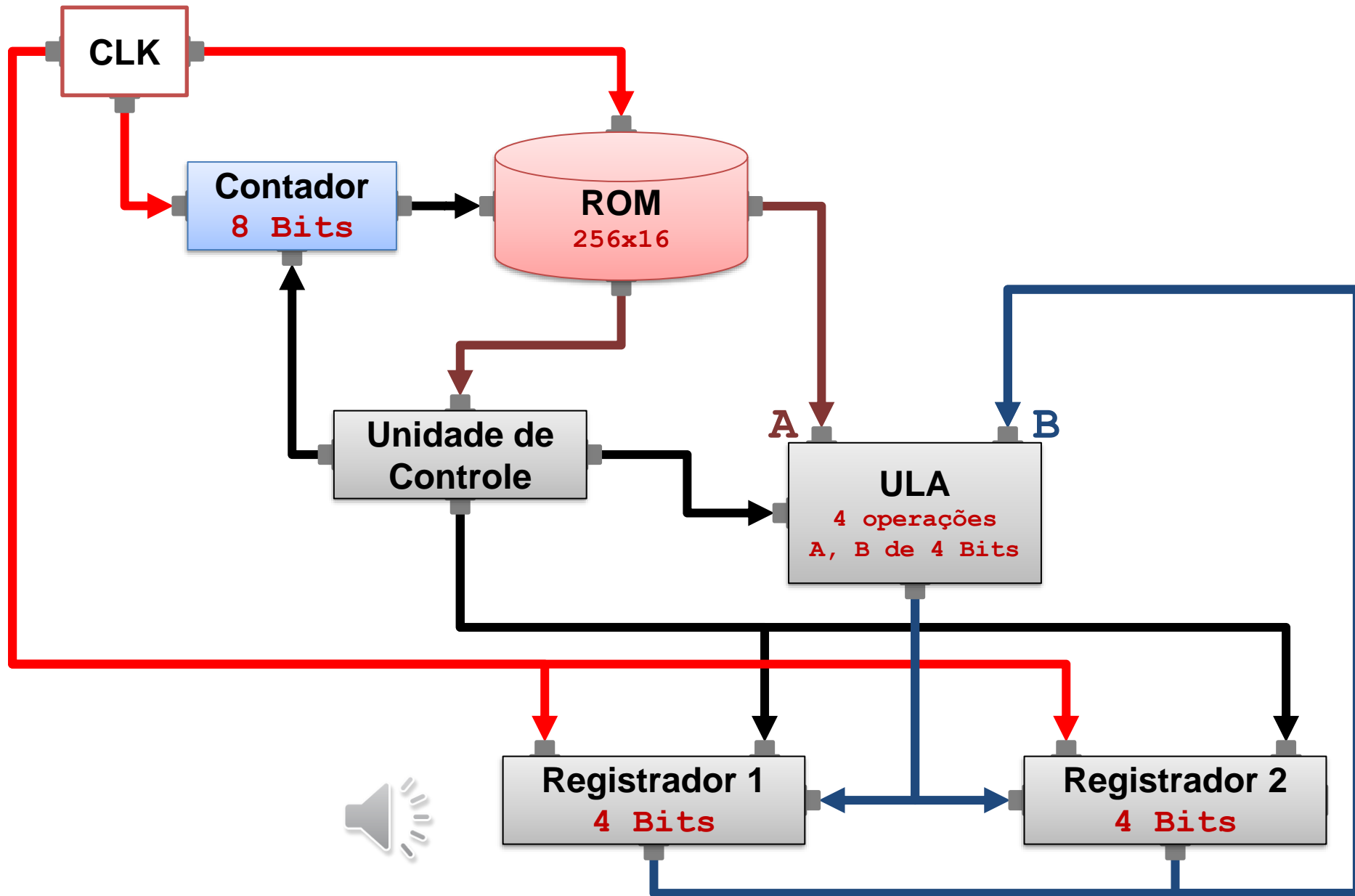


GE4Bio – Grupo de Estudos em Sinais Biológicos

**Prof.Dr. Danilo Spatti**

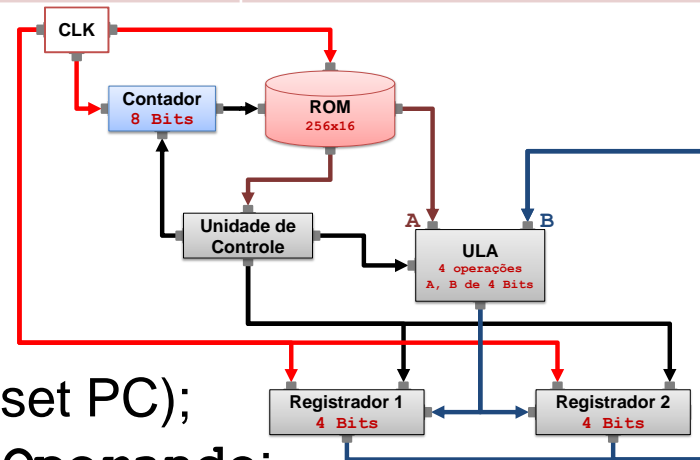
**São Carlos**





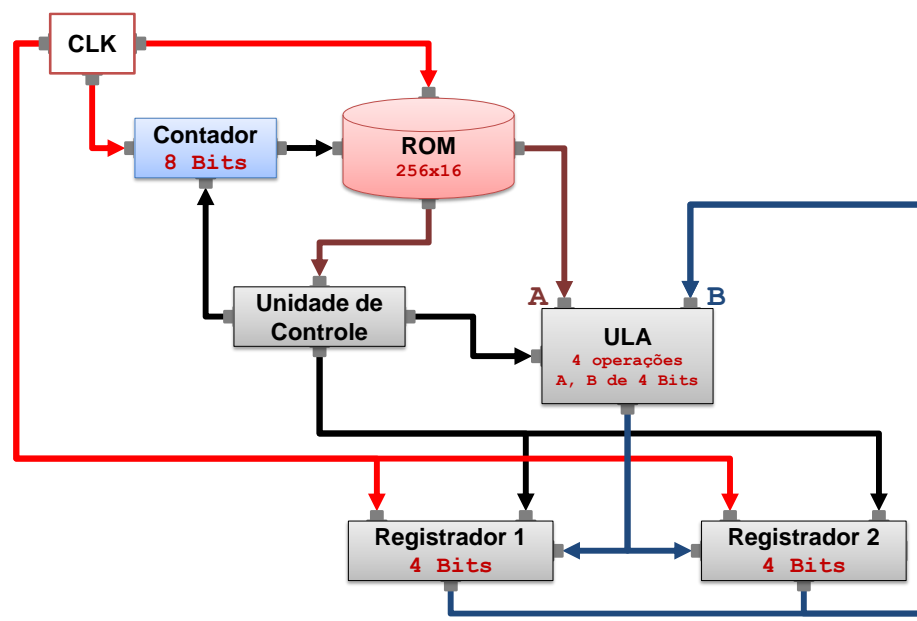
15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
RgTO				RgIN		JMP		ULA		Disponível			Operando		

- **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**;
- **ULA**: uma das 4 operações abaixo:
  - 00:  $\text{RgTO} \leftarrow \text{RgIN} + \text{Operando}$ ;
  - 01:  $\text{RgTO} \leftarrow \text{Operando} * 2$ ;
  - 10:  $\text{RgTO} \leftarrow \text{RgIN} - \text{Operando}$ ;
  - 11:  $\text{RgTO} \leftarrow \text{Operando} / 2$ ;
- **Disponível**: Disponível para melhorar a CPU caso queiram
- **Operando**: 4 bits diretamente da memória na ULA



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	c	0	8	2	2	<i>Jump #8</i>
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	$R2 \leftarrow 1 \times 2$
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	<i>CLR contador</i>

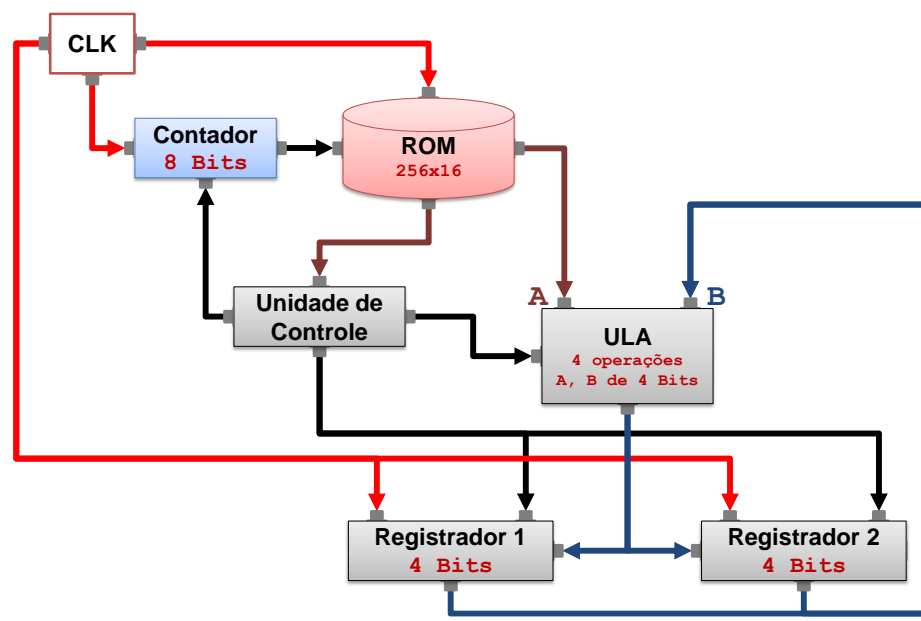
15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
RgTO				RgIN		JMP		ULA		Disponível			Operando		
<ul style="list-style-type: none"><li>• <b>RgTO</b>: Registrador de destino (4 bits)</li><li>• <b>RgIN</b>: Registrador de origem (4 bits)</li><li>• <b>JMP</b>: uma das 4 opções abaixo:<ul style="list-style-type: none"><li>• 00: Operação de ULA;</li><li>• 01: Reinicia Registradores;</li><li>• 10: Reinicia o contador de memória (reset PC);</li><li>• 11: Jump para posição de memória do <b>Operando</b>;</li></ul></li><li>• <b>ULA</b>: uma das 4 operações abaixo:<ul style="list-style-type: none"><li>• 00: <math>RgTO \leftarrow RgIN + Operando</math>;</li><li>• 01: <math>RgTO \leftarrow Operando * 2</math>;</li><li>• 10: <math>RgTO \leftarrow RgIN - Operando</math>;</li><li>• 11: <math>RgTO \leftarrow Operando / 2</math>;</li></ul></li><li>• <b>Disponível</b>: Disponível para melhorar a CPU caso queiram</li><li>• <b>Operando</b>: 4 bits diretamente da memória na ULA</li></ul>															

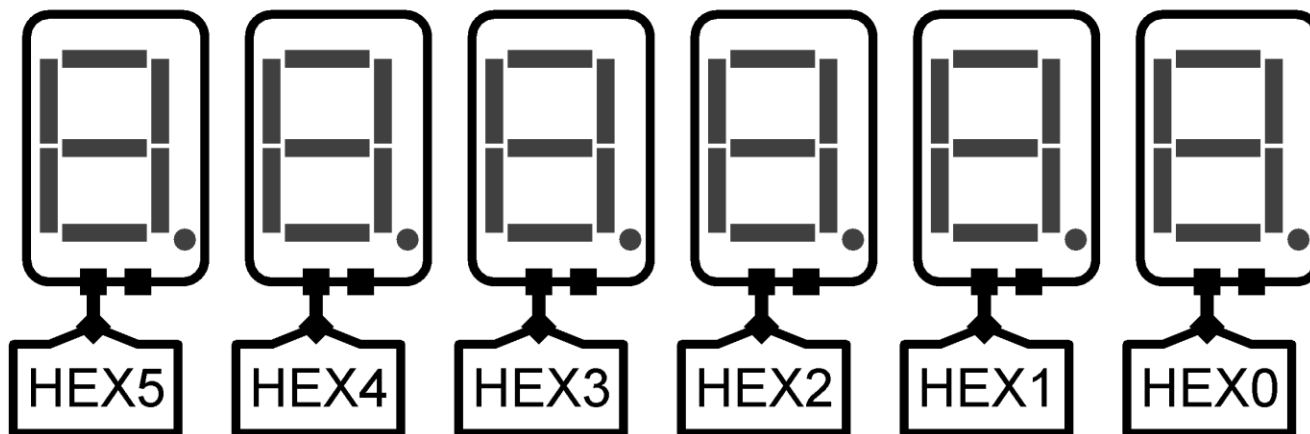


Count	Memória				R1	R2	Instrução
0	6	0	0	4	4	0	$R1 \leftarrow R2 + 4$
1	9	1	0	2	4	4	$R2 \leftarrow 2 \times 2$
2	6	2	0	1	3	4	$R1 \leftarrow R2 - 1$
3	6	3	0	5	2	4	$R1 \leftarrow 5 \div 2$
4	9	0	0	2	2	4	$R2 \leftarrow R1 + 2$
5	6	0	0	2	6	4	$R1 \leftarrow R2 + 2$
6	9	0	0	3	6	9	$R2 \leftarrow R1 + 3$
7	0	4	0	0	0	0	CLR R1 e R2
8	0	c	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

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**;
- **ULA**: uma das 4 operações abaixo:
  - 00:  $RgTO \leftarrow RgIN + \text{Operando}$ ;
  - 01:  $RgTO \leftarrow \text{Operando} * 2$ ;
  - 10:  $RgTO \leftarrow RgIN - \text{Operando}$ ;
  - 11:  $RgTO \leftarrow \text{Operando} / 2$ ;
- **Disponível**: Disponível para melhorar a CPU caso queiram
- **Operando**: 4 bits diretamente da memória na ULA



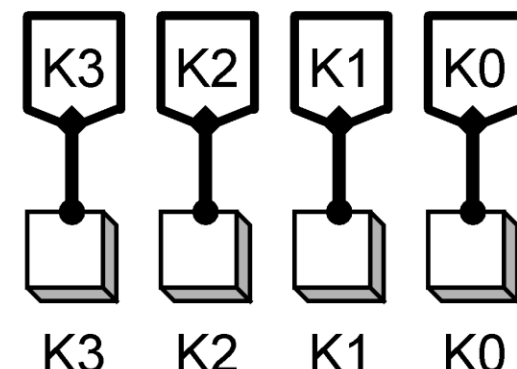
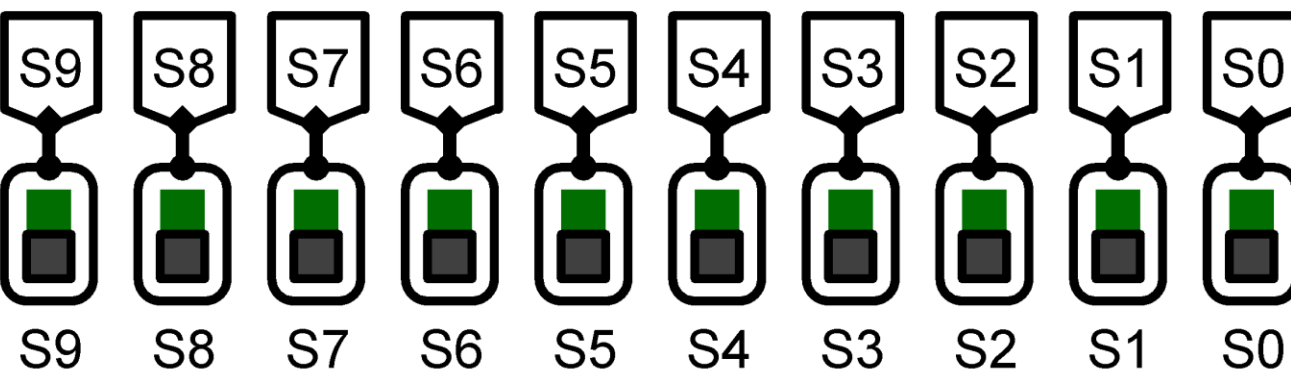
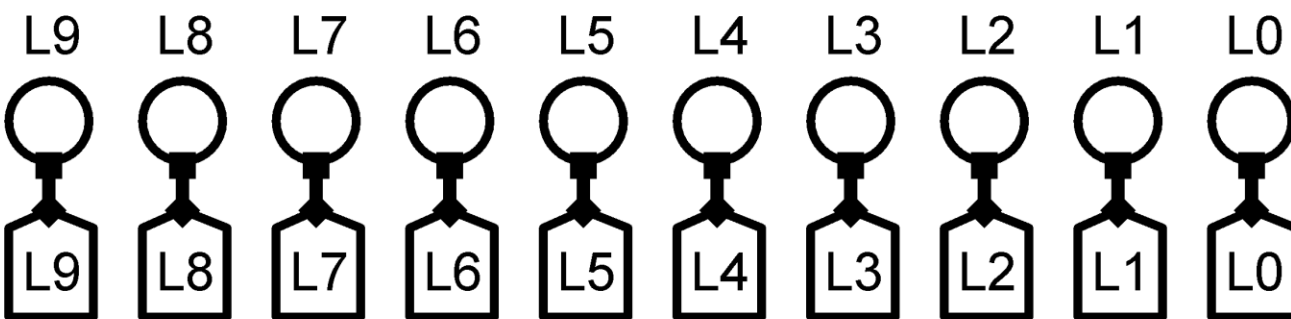


Contador: L7-L0

Memória: HEX5-HEX2.

Registradores:  
R1: HEX1  
R2: HEX0.

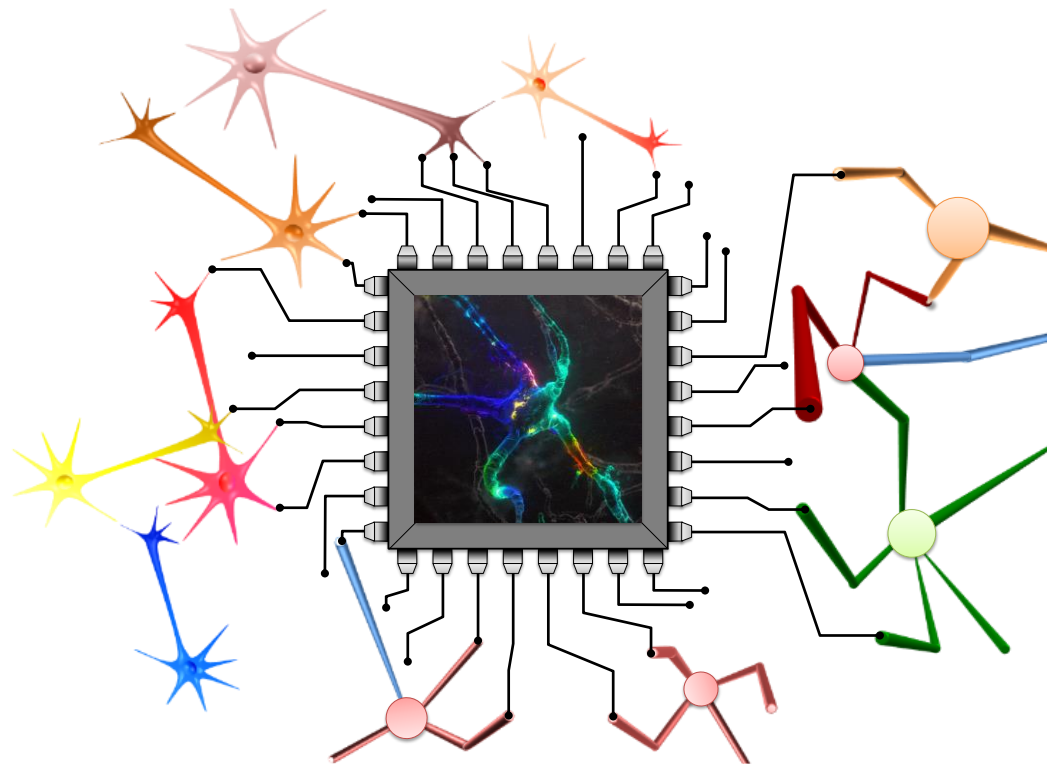
CLK: chave S9  
Master CLR: chave S0



- Faça as **adaptações** necessárias para que os **resultados** da ULA sejam armazenados no **registrador**.
- Atentar para o fato de que a saída da ULA, entrada e saída dos registradores são de **4 bits**.
- Realize **simulações** utilizando o **osciloscópio** na **saída** da ULA, **entrada** do registrador e **saída** do registrador.
- Integre o circuito da ULA com os registradores à Máscara da **DE0-CV**.



spatti@icmc.usp.br



GE4Bio – Grupo de Estudos em Sinais Biológicos

