# **Exercício Prático 1**

***Fernando Campos Silva Dal Maria (765053)***

***Professor: Romanelli Lodron Zuim***

## **Questões Propostas:**

1. A saída das portas lógicas é indeterminada, ou seja, o resultado é flutuante.
2. Esse tipo de somador apresenta um longo tempo de resposta por ser dependente do *carry*. Sendo assim, para cada novo somador adicional teremos um atraso no tempo de resposta. Veja, portanto, que se considerarmos um atraso médio de 10ns (nanosegundos) por porta lógica, teremos um tempo de resposta de 30ns para o primeiro somador e mais 20ns adicionais para cada somador subsequente no circuito.
3. O tempo total de resposta para um somador de 4 bits pode ser calculado da seguinte forma:
   1. O tempo de resposta para o primeiro somador será de 30ns.
   2. Para cada um dos somadores subsequentes podemos considerar um atraso de 20ns adicionais.
   3. Portanto teremos um total de: 30ns + 20ns x (4 - 1) = 90ns
4. Para construir um somador completo de 32 bits são necessários 8 somadores de 4 bits (8 x 4 = 32). O tempo para a execução de uma soma será de 30ns + 20ns x (32 - 1) = 650ns.
5. A frequência máxima para um somador de 32 bits será de ou seja, 1,5MHz
6. É possível tornar o processo de soma mais veloz se utilizarmos uma técnica de implementação do circuito que não seja dependente do *carry*. Para isso podemos utilizar a técnica chamada de *carry lookahead* que permite com que as somas de cada bit sejam realizadas em paralelo.

## **Projeto de Somador**

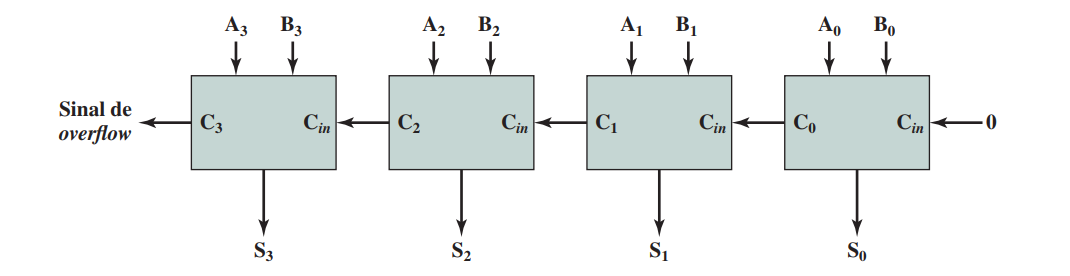
Para entender o funcionamento do circuito somador observe a tabela verdade abaixo:

| **A** | **B** | ***Carry in (Cin)*** |  | **S** | ***Carry out (Cout)*** |
| --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Pela tabela verdade obtemos as seguintes soluções:

Por meio dessas equações podemos implementar um circuito que seja capaz de realizar a soma de A e B, gerar o resultado desejado e um *carry out* que pode ser utilizado por outro somador completo que esteja ligado de forma subsequente. Veja abaixo uma imagem ilustrativa para o circuito de um somador de 4 bits:

**Figura 1:** Somador completo de 4 bits.

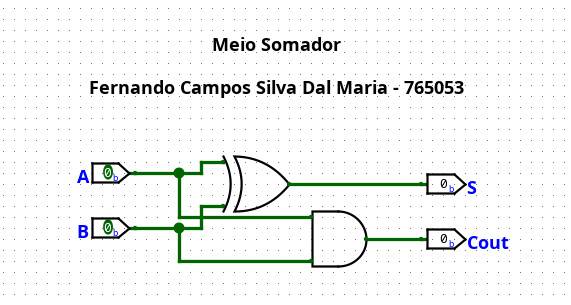


**Fonte:** WILLIAM, Stallings. Pearson, 2017.

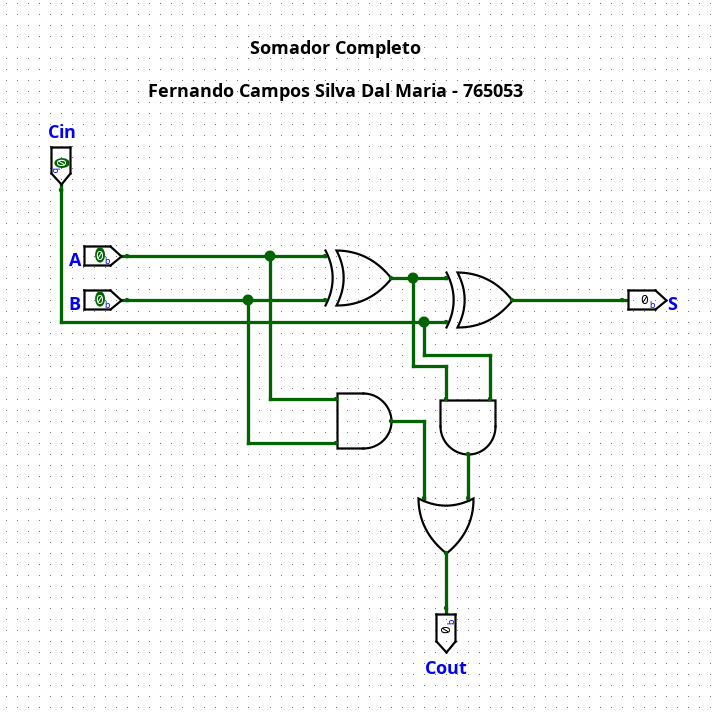
Note também que em qualquer adição o resultado pode ser maior do que pode ser mantido no tamanho da palavra. Por essa razão, a figura 1 apresenta uma *flag* *overflow* que sinaliza quando esse tipo de situação ocorre.

**Exemplo de implementação de um somador no Logisim:**

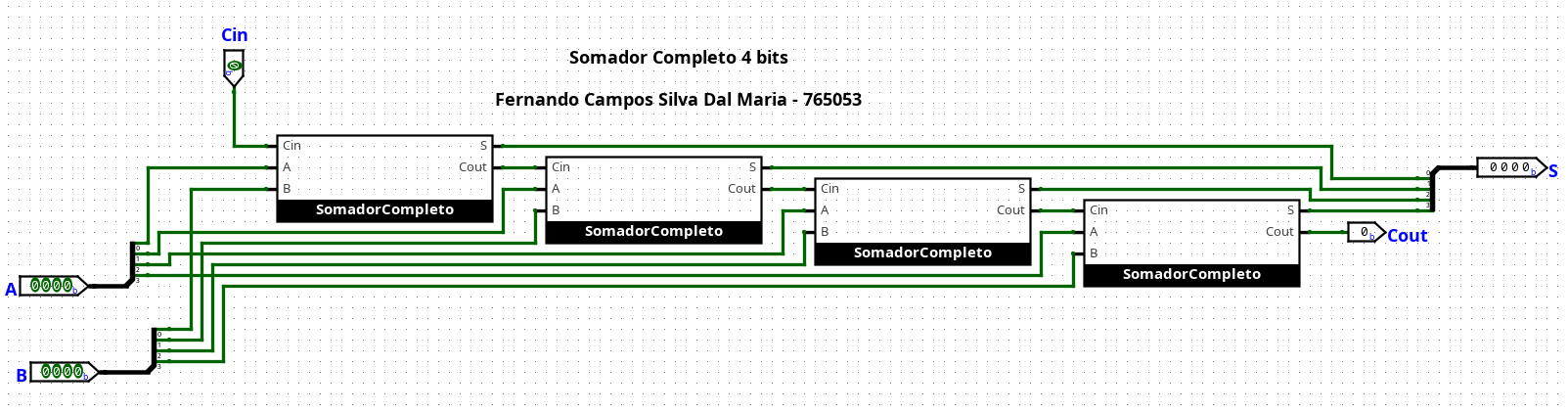
1. Implementação de um meio somador:



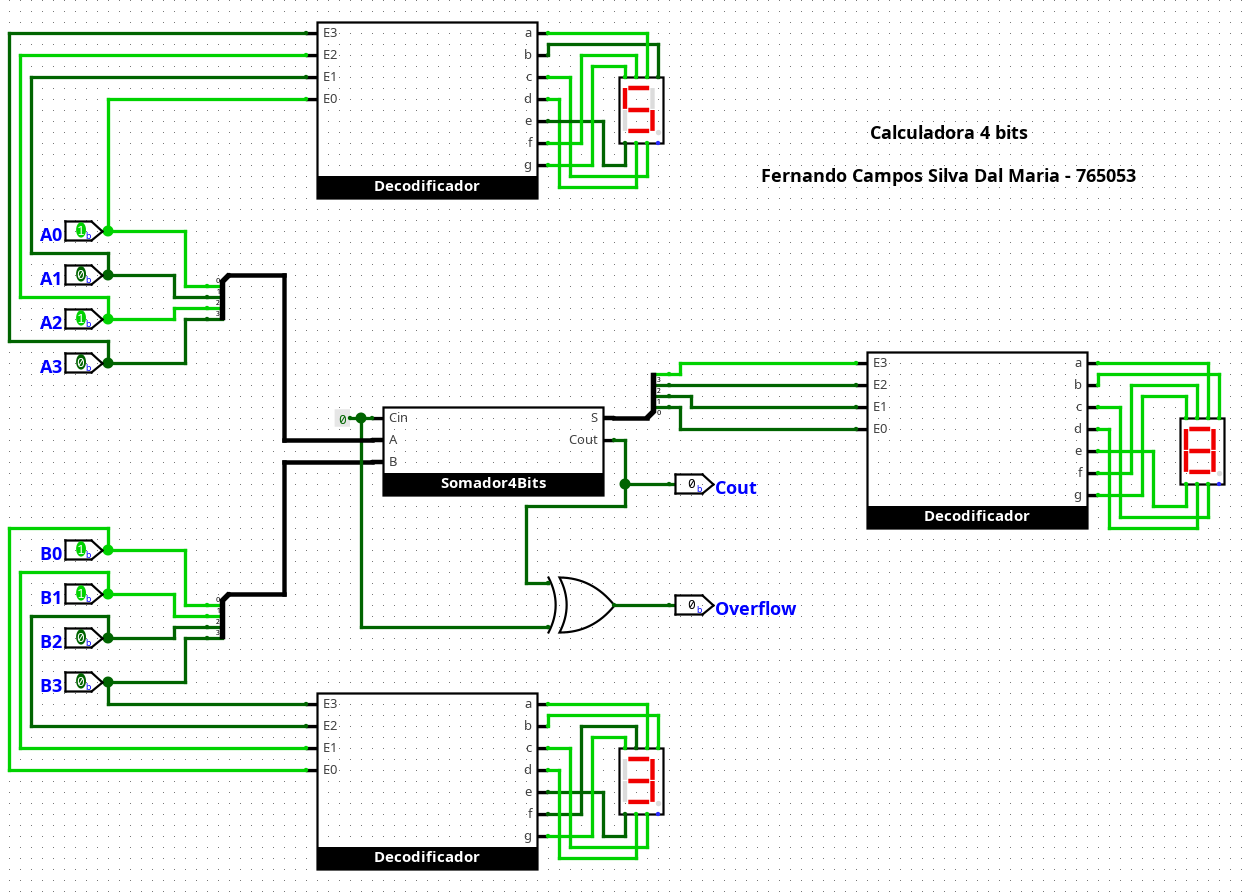
1. Implementação de um somador completo:



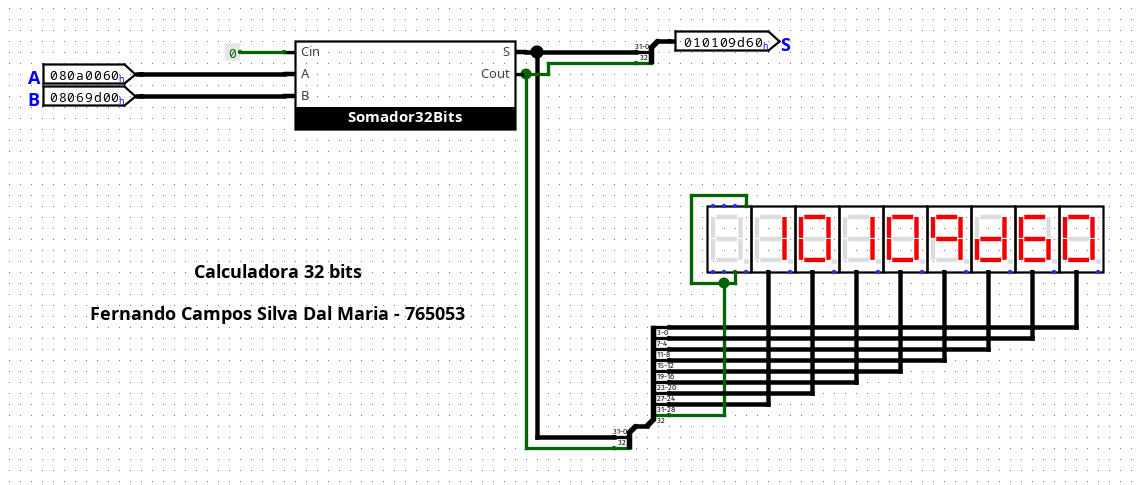
1. Implementação de um somador completa de 4 bytes



1. Calculadora de 4 bits

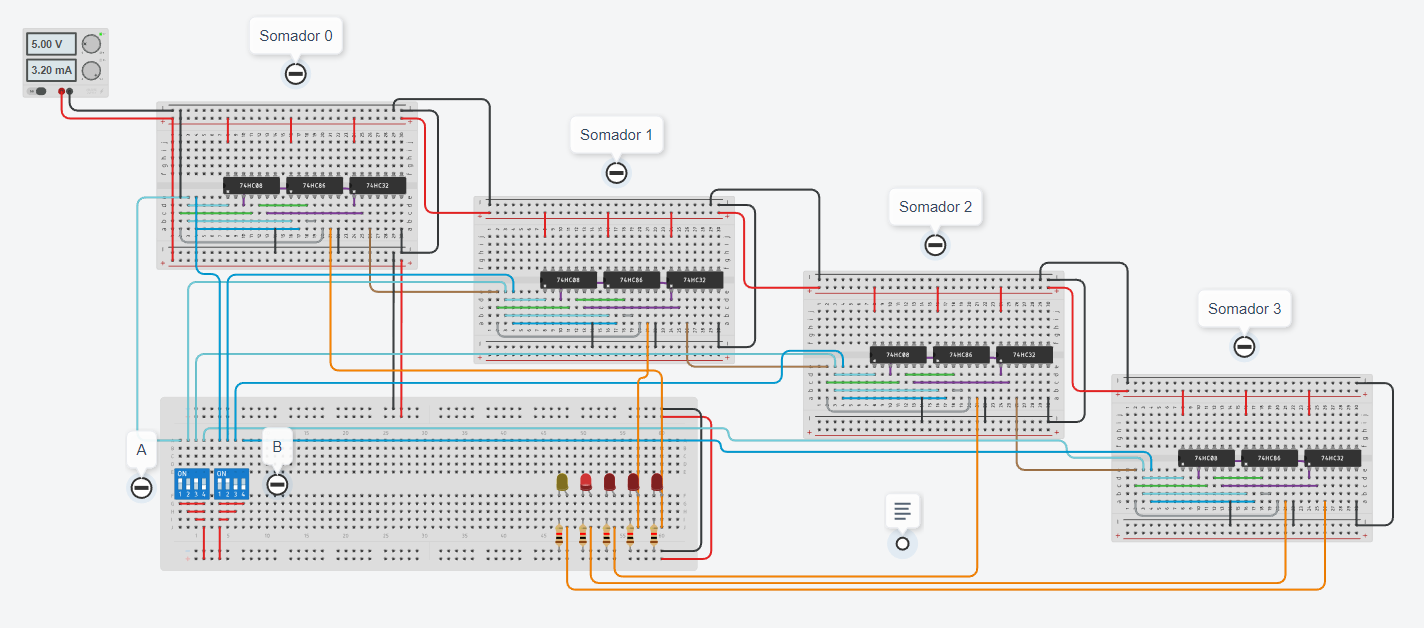


O processo de conectar circuitos somadores de forma sequencial pode ser repetido quantas vezes se queira, entretanto, devido ao atraso gerado pelo *carry,* o desempenho do somador diminui sucessivamente. Veja abaixo outro exemplo de calculadora no logisim, mas dessa vez com 32 bits:



Para ver mais detalhadamente os circuitos acesse:

## **Exemplo de Somador de 4 bits (Tinkercad)**



O simulador acima foi feito utilizando os circuitos comerciais 74HC08, 74HC86 e 74HC32. Os *datasheets* para esses circuitos podem ser encontrados nas referências deste texto.

# **Referências**

1. PATTERSON, David A. & HENNESSY, John L. **Organização e Projeto de Computadores.** 5. ed. Rio de Janeiro : Elsevier, 2017.
2. WILLIAM, Stallings. **Arquitetura e organização de computadores.** 10. ed. São Paulo: Pearson Education do Brasil, 2017.
3. PHILIPS SEMICONDUCTORS. **Quad 2-input AND gate**. 74HC08; 74HCT08. Jul, 2003.
4. PHILIPS SEMICONDUCTORS. **Quad 2-input EXCLUSIVE-OR gate**. 74HC/HCT86. Dec, 1990.
5. PHILIPS SEMICONDUCTORS. **Quad 2-input OR gate**. 74HC/HCT32. Dec, 1990.
6. PHILIPS SEMICONDUCTORS. **4-bit binary full adder with fast carry**. 74HC283. Nov, 2004.