

# Projeto de circuito

MAC0329 – Álgebra booleana e aplicações (DCC / IME-USP — 2021)

– Todas as etapas do projeto deverão ser feitas usando o Logisim-evolution

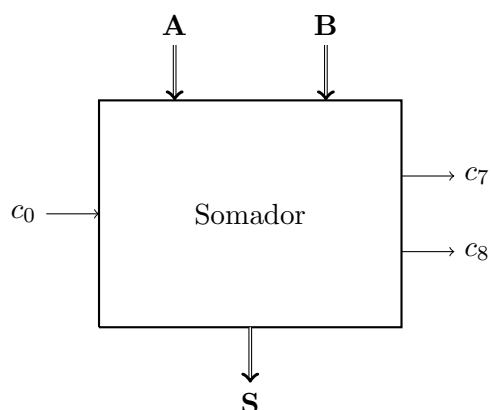
<https://github.com/reds-heig/logisim-evolution> –

## Parte 1: Circuito Somador – entrega no e-disciplinas, até 29/05

Nesta primeira etapa do projeto (Parte 1), o objetivo é a implementação de um circuito somador de 8 *bits*, e sua utilização para realizar as operações de adição e subtração com números de 8 *bits*. (Veja também os capítulos 2 e 3 das notas de aula)

### 1 Especificação de um somador de 8 *bits*

Em nosso projeto, o somador de 8 bits terá a seguinte configuração



#### Entradas:

$\mathbf{A} = (a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0)$  (operando 1)

$\mathbf{B} = (b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0)$  (operando 2)

$c_0$  (*carry* na coluna 0)

#### Saídas:

$\mathbf{S} = \mathbf{A} + \mathbf{B}$  (8 *bits* da soma)

$c_7$  (*carry* na coluna 7)

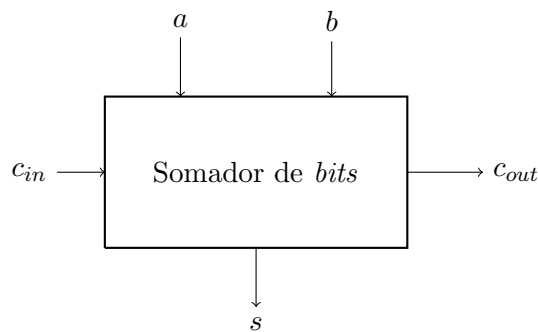
$c_8$  (*carry* na coluna 8)

### 2 Organização do circuito

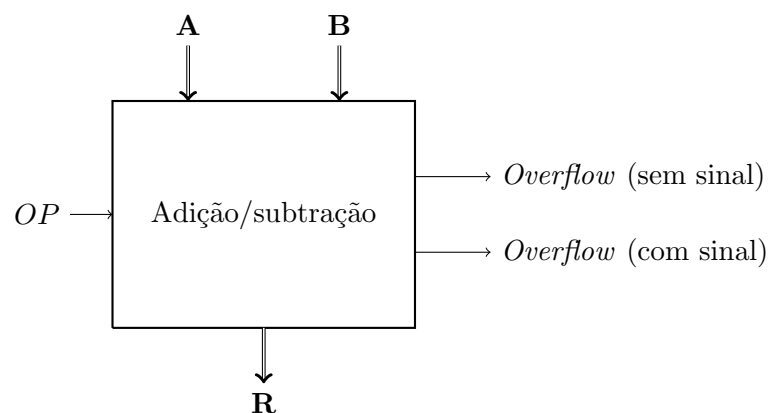
Um circuito complexo pode ser decomposto muitas vezes em subcircuitos. Supondo que os subcircuitos estejam prontos e disponíveis, podemos implementar um circuito complexo a partir dos subcircuitos prontos. Mais do que isso, um mesmo componente pode ser utilizado em diferentes partes de um circuito maior. É importante que os componentes tenham uma interface e funcionalidade bem definidas.

Neste EP seguiremos o princípio descrito acima. Começando do componente mais básico até o mais complexo. Primeiro faremos um somador de bits, depois usaremos ele para fazer o circuito somador propriamente dito. Por fim, faremos um circuito principal (**main**) para testar o circuito somador, tanto para realizar as operações de adição como de subtração.

1. **somabits**: este deve ser o circuito somador de *bits*. Ele recebe três *bits*,  $a$ ,  $b$  e  $c_{in}$ , e devolve dois *bits*,  $s$  e  $c_{out}$ , conforme mostrado no diagrama a seguir.



2. **somador**: este deve ser o circuito somador descrito na seção 1 acima e ele deve ser construído usando o subcircuito **somabits** acima.
3. **main**: o circuito principal servirá para testar o somador. Aqui espera-se um circuito com a seguinte interface:



Neste diagrama,

- **A** e **B** são os operandos, ambos de 8 *bits*;
- **OP** é um *bit* para indicar o tipo de operação: se 0, deve ser calculada a adição  $A + B$  e se 1 deve ser calculada a subtração  $A - B$ ;
- **R** é o resultado da operação (8 *bits*);
- “*overflow* sem sinal” é um *bit*: se 1, indica que a operação calculada, interpretando-se os números como sem sinal, resultou em *overflow*;
- “*overflow* com sinal” é um *bit*: se 1, indica que a operação calculada, interpretando-se os números como com sinal (na notação complemento de dois), resultou em *overflow*.

O circuito principal deve utilizar o circuito **somador**.

**Observação:** Embora aqui estejamos falando de modularização de circuitos lógicos, esse conceito estará bastante presente na especificação e desenvolvimento de softwares em geral. Em MAC0110, em breve vocês aprenderão “funções”, que correspondem a um dos elementos modulares na organização do código de um programa.

### 3 O que fazer e entregar

Os circuitos devem ser implementados usando o software **logisim-evolution** (um programa em Java).

A ordem natural para o EP1 é fazer o circuito seguindo a estratégia *bottom-up*, isto é, começa-se com o subcircuito mais básico (**somabits**), em seguida faz-se o **somador**, e finalmente o **main**. Cada um deles deve ser testado individualmente, antes de ser empregado em um circuito de nível superior. Adicione nomes (labels) aos pinos de entrada e saída dos subcircuitos. Use nomes coerentes com os descritos nos esquemas acima, de forma que qualquer um possa usar o seu circuito como um componente “caixa-preta”.

Deve ser entregue o arquivo com a extensão `.circ`, gerado pelo `logisim-evolution`, contendo o circuito desenvolvido.

## 4 Critérios de avaliação

Serão avaliados os seguintes aspectos:

- corretude dos circuitos
- aderência à organização proposta, com uma clara identificação (nome consistente) dos pinos de entrada e saída em cada circuito

**Dúvidas ?** Poste suas perguntas no fórum de Dúvidas/Comentários no moodle do e-disciplinas