



**Universidade Tecnológica Federal do Paraná – UTFPR**  
**Bacharelado em Ciência da Computação**

## **BCC32B – Elementos de Lógica Digital**

**Prof. Rodrigo Hübner**

**Aula 12 – Aritmética computacional: adição e subtração em complemento de 2; Somador de alto desempenho**

# Adição e Subtração em Complemento de 2

## Exemplos de Adição:

a)

$$\begin{array}{r} -7 \\ + (+5) \\ \hline -2 \end{array}$$

$$7_{10} = 0111_2$$

$$5_{10} = 0101_2$$

$$\begin{array}{r} 1000 \\ + 1 \\ \hline 1001 \end{array}$$

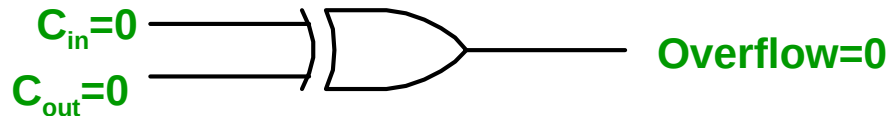
Complemento de 2 do valor  $-7_{10}$

$C_{in}=0$

$$\begin{array}{r} 0 \ 1 \\ 1001 \\ + 0101 \\ \hline 0 \ 1110 \end{array}$$

$C_{out}=0$

$= -2_{10}$



em Complemento de 2

Verificar → faz o mesmo  
para chegar no número +

$$\begin{array}{r} 0001 \\ + 1 \\ \hline 0010 \end{array}$$

Complemento de 2 de  $-2_{10}$

+ 2

# Adição e Subtração em Complemento de 2

## Exemplos de Adição:

b)

$$\begin{array}{r} -4 \\ + (+4) \\ \hline 0 \end{array}$$

$4_{10} = 0100_2$

Complemento de 2 do valor  $-4_{10}$

$$\begin{array}{r} 1011 \\ + 1 \\ \hline 1100 \end{array}$$

$C_{in}=1$

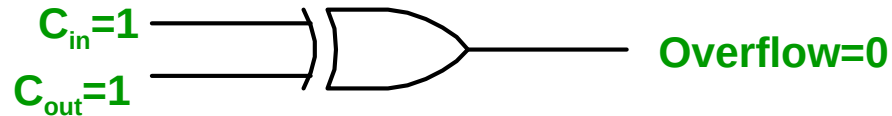
$$\begin{array}{r} 1 \\ 1100 \\ + 0100 \\ \hline 1 \\ 0000 \end{array}$$

$C_{out}=1$

$= 0_{10}$

em Complemento de 2

Despreza o carry



$$\begin{array}{r} 1111 \\ + 1 \\ \hline 0000 \end{array}$$

Complemento de 2 de  $0_{10}$

# Adição e Subtração em Complemento de 2

## Exemplos de Adição:

c)

$$\begin{array}{r} +3 \\ + (+4) \\ \hline +7 \end{array}$$

$3_{10} = 0011_2$   
 $4_{10} = 0100_2$

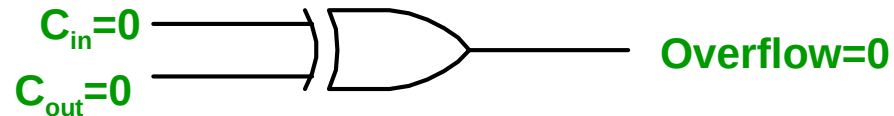
$C_{in} = 0$

$$\begin{array}{r} 0 \\ 0011 \\ + 0100 \\ \hline 0 \ 0111 \end{array}$$

$C_{out} = 0$

$= 7_{10}$

em Complemento de 2



# Adição e Subtração em Complemento de 2

## Exemplos de Adição:

d)

$$\begin{array}{r} -4 \\ + (-1) \\ \hline -5 \end{array}$$

$4_{10} = 0100_2$   
 $1_{10} = 0001_2$

$C_{in}=1$

$$\begin{array}{r} 1 \\ 1100 \\ + 1111 \\ \hline 11011 \end{array}$$

$C_{out}=1$

$= -5_{10}$

Despreza o carry

em Complemento de 2

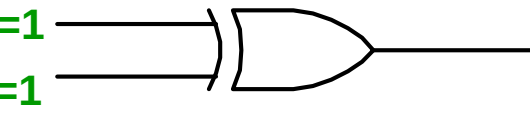
$$\begin{array}{r} 1011 \\ + 1 \\ \hline 1100 \end{array}$$

Complemento de 2 do valor  $-4_{10}$

$$\begin{array}{r} 1110 \\ + 1 \\ \hline 1111 \end{array}$$

Complemento de 2 do valor  $-1_{10}$

$C_{in}=1$   
 $C_{out}=1$



Overflow=0

$$\begin{array}{r} 0100 \\ + 1 \\ \hline 0101 \end{array}$$

Complemento de 2 de  $-5_{10}$

# Adição e Subtração em Complemento de 2

## Exemplos de Adição:

e)

$$\begin{array}{r} +5 \\ + (+4) \\ \hline +9 \end{array}$$

$5_{10} = 0101_2$   
 $4_{10} = 0100_2$

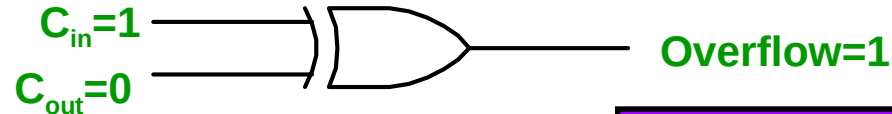
$C_{in}=1$

$$\begin{array}{r} 1 \\ 0101 \\ + 0100 \\ \hline 0 \ 1001 \end{array}$$

$C_{out}=0$

$= 9_{10}$

em Complemento de 2



2 números positivos  
somados não podem  
resultar num número  
negativo

Erro de Overflow

$1001_2 = 9_{10}$

Não dá para representar 9 com registrador de 4 bits,  
com 1 bit de sinal

# Adição e Subtração em Complemento de 2

## Exemplos de Adição:

f)

$$\begin{array}{r} -7 \\ + (-6) \\ \hline -13 \end{array}$$

$$7_{10} = 0111_2$$
$$6_{10} = 0110_2$$

$$\begin{array}{r} 1000 \\ + 1 \\ \hline 1001 \end{array}$$

Complemento de 2 do valor  $-7_{10}$

Complemento de 2 do valor  $-6_{10}$

$C_{in}=0$

$$\begin{array}{r} 0 \\ 1001 \\ + 1010 \\ \hline 1 \\ 0011 \end{array}$$

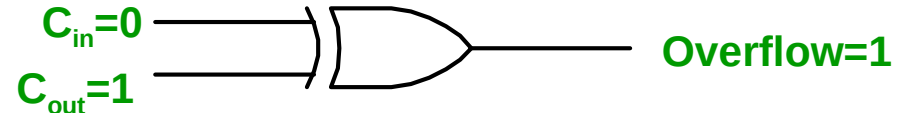
$C_{out}=1$

$= 3_{10}$

em Complemento de 2

Despreza o carry

$$\begin{array}{r} 1001 \\ + 1 \\ \hline 1010 \end{array}$$



2 números negativos somados não podem resultar num número positivo

# Adição e Subtração em Complemento de 2

## Subtração

A Subtração pode ser efetuada usando um circuito Somador

Minuendo  
- Subtraendo  

---

Subtração

Usa o Complemento de 2 do Subtraendo e soma-o ao Minuendo



# Adição e Subtração em Complemento de 2

## Exemplos de Subtração:

a)

$$\begin{array}{r} 2 \\ - (+7) \\ \hline -5 \end{array}$$

$2_{10} = 0010_2$   
 $7_{10} = 0111_2$

Complemento de 2 do valor  $7_{10}$

$$\begin{array}{r} 1000 \\ + 1 \\ \hline 1001 \end{array}$$

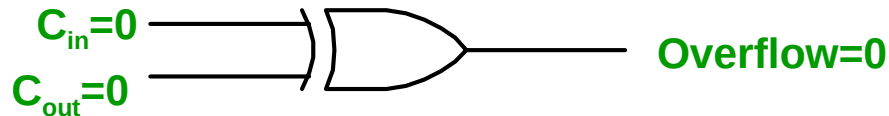
$C_{in}=0$

$$\begin{array}{r} 0 \\ 0010 \\ + 1001 \\ \hline 01011 \end{array}$$

$C_{out}=0$

$= -5_{10}$

em Complemento de 2



$$\begin{array}{r} 0100 \\ + 1 \\ \hline 0101 \end{array}$$

Complemento de 2 de  $-5_{10}$

# Adição e Subtração em Complemento de 2

## Exemplos de Subtração:

b)

$$\begin{array}{r} 5 \\ - (+2) \\ \hline +3 \end{array}$$

$5_{10} = 0101_2$   
 $2_{10} = 0010_2$

$$\begin{array}{r} 1101 \\ + 1 \\ \hline 1110 \end{array}$$

Complemento de 2 do valor  $2_{10}$

$C_{in}=1$

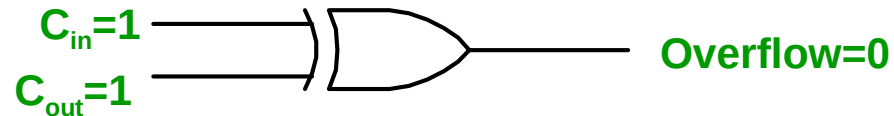
$$\begin{array}{r} 1 \\ 0101 \\ + 1110 \\ \hline 10011 \end{array}$$

$C_{out}=1$

$= 3_{10}$

em Complemento de 2

Despreza o carry



# Adição e Subtração em Complemento de 2

## Exemplos de Subtração:

c)

$$\begin{array}{r} -5 \\ - (+2) \\ \hline -7 \end{array}$$

$$5_{10} = 0101_2$$

$$2_{10} = 0010_2$$

$$\begin{array}{r} 1010 \\ + 1 \\ \hline 1011 \end{array}$$

Complemento de 2 do valor  $-5_{10}$

Complemento de 2 do valor  $2_{10}$

$$\begin{array}{r} 1101 \\ + 1 \\ \hline 1110 \end{array}$$

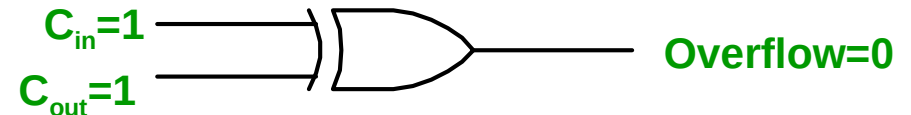
$C_{in}=1$

$$\begin{array}{r} 11 \\ 1011 \\ + 1110 \\ \hline 11001 \end{array}$$

$C_{out}=1$

$= -7_{10}$

em Complemento de 2



Despreza o carry

$$\begin{array}{r} 0110 \\ + 1 \\ \hline 0111 \end{array}$$

Complemento de 2 de  $-7_{10}$

# Adição e Subtração em Complemento de 2

## Exemplos de Subtração:

d)

$$\begin{array}{r} 5 \\ - (-2) \\ \hline 7 \end{array}$$

$$5_{10} = 0101_2$$

$$2_{10} = 0010_2$$

$C_{in}=0$

$$\begin{array}{r} 0 \\ 0101 \\ + 0010 \\ \hline 0 \\ 0111 \end{array} = 7_{10}$$

$C_{out}=0$

em Complemento de 2

$$\begin{array}{r} 1101 \\ + 1 \\ \hline 1110 \end{array}$$


$$\begin{array}{r} 0001 \\ + 1 \\ \hline 0010 \end{array}$$

Complemento de 2 do valor  $2_{10}$

Complemento de 2 do valor  $-2_{10}$

Complemento de 2 de  $-2=2$

$C_{in}=0$   
 $C_{out}=0$



Overflow=0

# Adição e Subtração em Complemento de 2

## Exemplos de Subtração:

e)

$$\begin{array}{r} 7 \\ - (-7) \\ \hline 14 \end{array}$$

$$7_{10} = 0111_2$$

$C_{in}=1$

$$\begin{array}{r} 111 \\ 0111 \\ + 0111 \\ \hline 01110 \end{array}$$

$C_{out}=0$

$01110 = -2_{10}$

em Complemento de 2

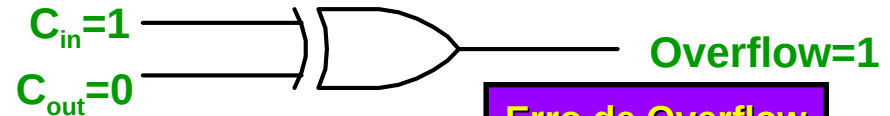
$$\begin{array}{r} 1000 \\ + 1 \\ \hline 1001 \end{array}$$

Complemento de 2 do valor  $7_{10}$

$$\begin{array}{r} 0110 \\ + 1 \\ \hline 0111 \end{array}$$

Complemento de 2 do valor  $-7_{10}$

Complemento de 2 de  $-7=7$



Erro de Overflow

$$01110_2 = 14_{10}$$

Não dá para representar 14 com registrador de 4 bits, com 1 bit de sinal

# Adição e Subtração em Complemento de 2

## Exemplos de Subtração:

f)

$$\begin{array}{r} (-6) \\ - (+4) \\ \hline -10 \end{array}$$

$$6_{10} = 0110_2$$

$$4_{10} = 0100_2$$

$C_{in}=0$

$$\begin{array}{r} 0 \\ 1010 \\ + 1100 \\ \hline 1 \\ 0110 \end{array}$$

$C_{out}=0$

$= 6_{10}$

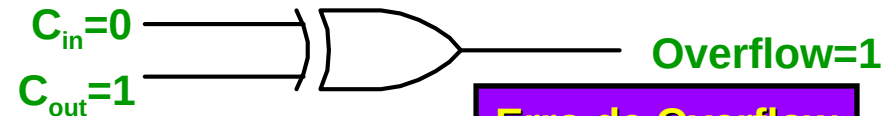
em Complemento de 2

$$\begin{array}{r} 1001 \\ + 1 \\ \hline 1010 \end{array}$$

$$\begin{array}{r} 1011 \\ + 1 \\ \hline 1100 \end{array}$$

Complemento de 2 do valor  $6_{10}$

Complemento de 2 do valor  $4_{10}$



**Erro de Overflow**

2 números negativos somados não podem resultar num número positivo

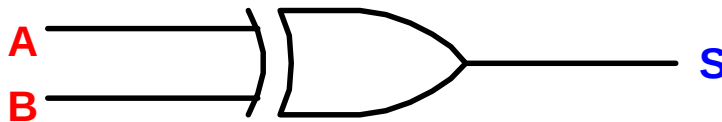
# Circuito Somador/Subtrator em Complemento de 2

Obs: Para obter o circuito somador/subtrator em complemento de 2 vamos usar a porta XOR

TV da Porta XOR

| Entradas |   | Saída |
|----------|---|-------|
| A        | B | S     |
| 0        | 0 | 0     |
| 0        | 1 | 1     |
| 1        | 0 | 1     |
| 1        | 1 | 0     |

Símbolo da Porta XOR

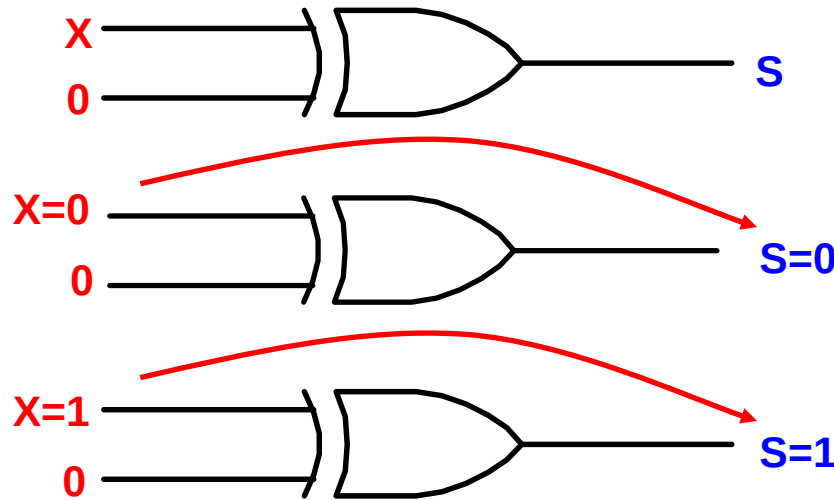


# Circuito Somador/Subtrator em Complemento de 2

TV da Porta XOR

Se fixar uma entrada em 0

| Entradas |   | Saída |
|----------|---|-------|
| A        | B | S     |
| 0        | 0 | 0     |
| 0        | 1 | 1     |
| 1        | 0 | 1     |
| 1        | 1 | 0     |



Deixando uma entrada de controle em 0, o dado "X" é copiado para a saída

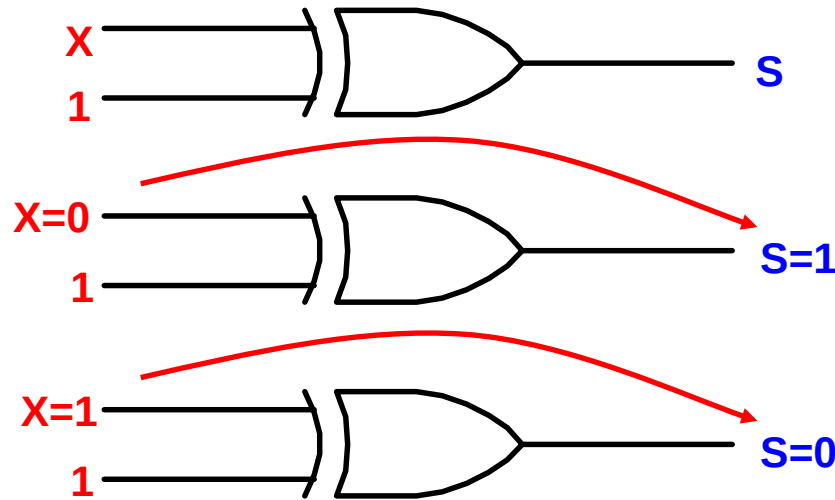


# Circuito Somador/Subtrator em Complemento de 2

TV da Porta XOR

Se fixar uma entrada em 1

| Entradas |   | Saída |
|----------|---|-------|
| A        | B | S     |
| 0        | 0 | 0     |
| 0        | 1 | 1     |
| 1        | 0 | 1     |
| 1        | 1 | 0     |



Para obter o complemento de 2 precisa somar 1 ao bit menos significativo do dado

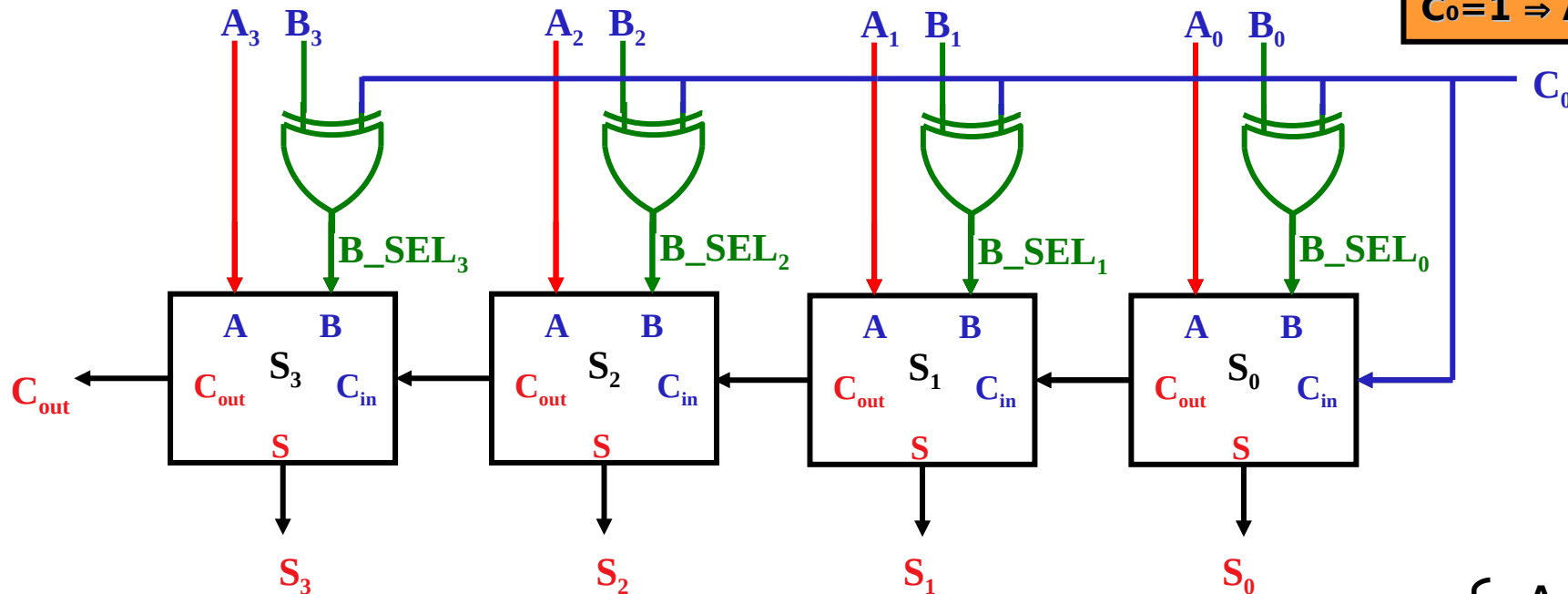
Deixando uma entrada de controle em 1, o dado "X" é complementado na saída:  $S = \bar{X}$

# Circuito Somador/Subtrator em Complemento de 2

**C=Controle da Operação**

$C_0=0 \Rightarrow A_i+B_i$

$C_0=1 \Rightarrow A_i-B_i$



Lembrete:  $C_{in} = 1$ , Soma "1" em B

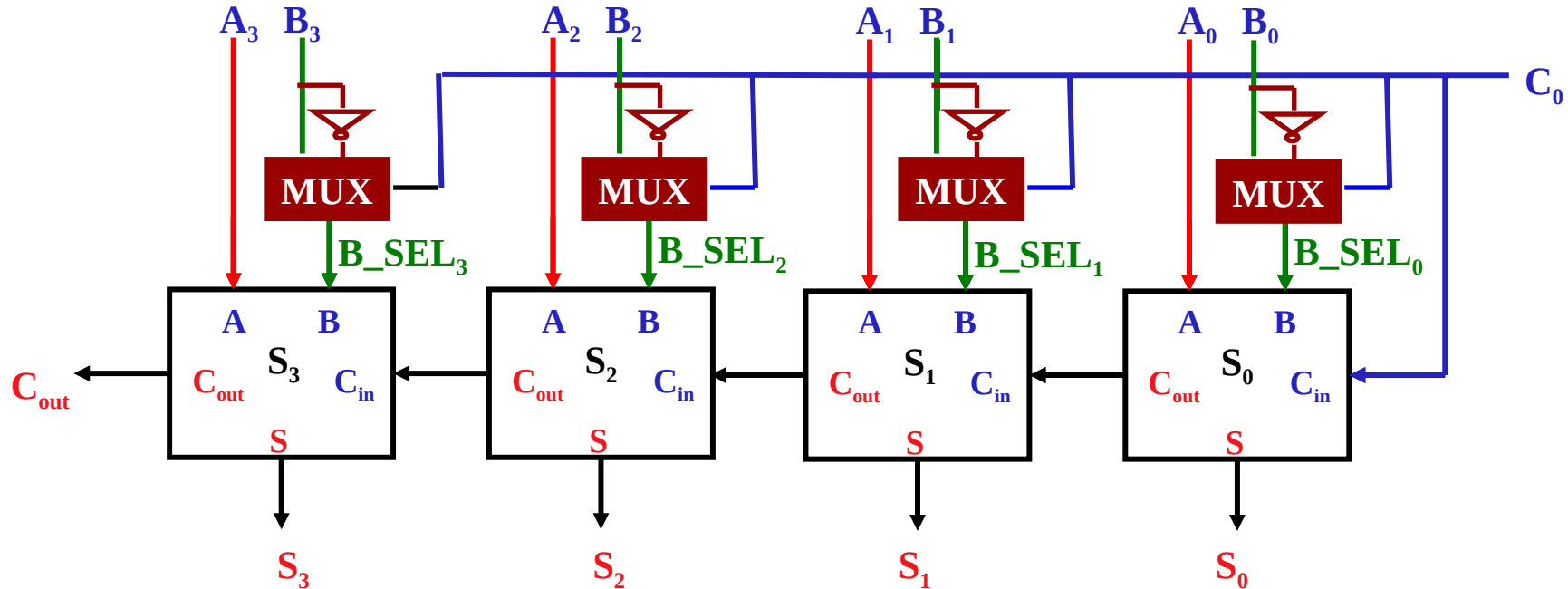
$$C_0=0 \left\{ \begin{array}{r} A_0 \\ B_0 \\ \hline +0 \end{array} \right. \quad C_0=1 \left\{ \begin{array}{r} A_0 \\ \overline{B_0} \\ \hline +1 \end{array} \right.$$

# Exercícios

- 1. Projete um circuito somador/subtrator em complemento de 2 usando um MUX para fazer o controle da geração do complemento de 2 do dado (subtraendo).**
- 2. O custo do “novo” circuito somador/subtrator é maior ou menor que o do circuito anterior?**

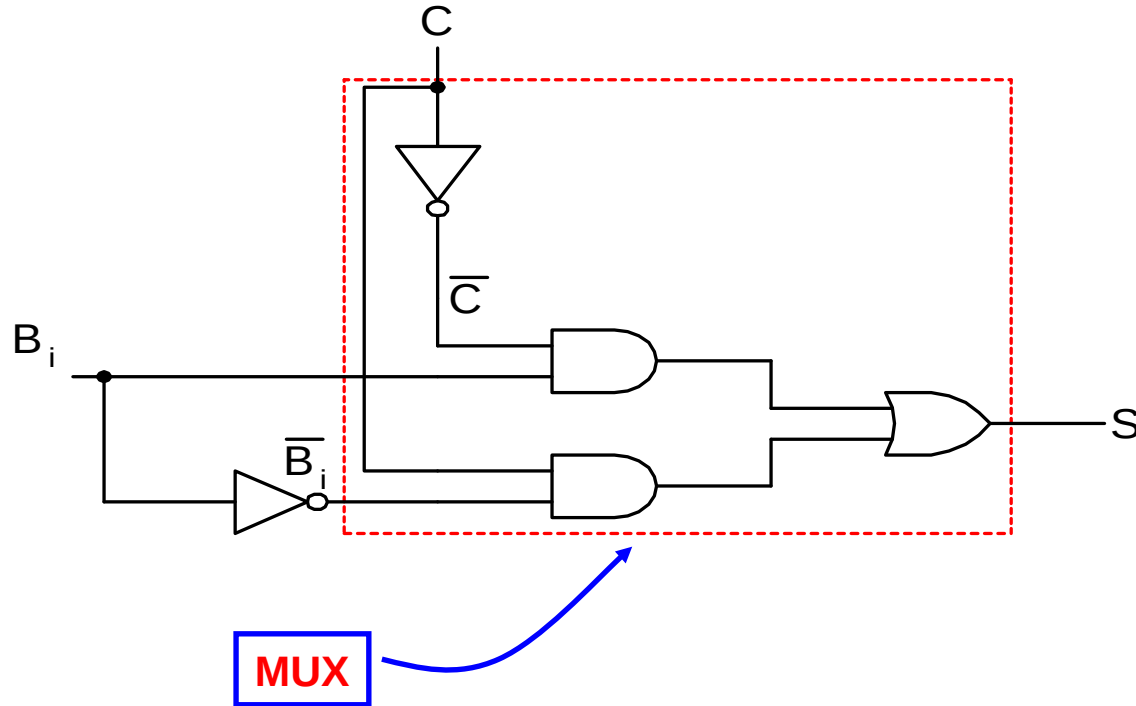
# Soluções

1. Projete um circuito somador/subtrator em complemento de 2 usando um MUX para fazer o controle da geração do complemento de 2 do dado (subtraendo).



# Soluções

1. O custo do “novo” circuito somador/subtrator é maior ou menor que o do circuito anterior?



Para circuitos de 4 bits:

**Circuito 1: 4 Portas XOR**

**Circuito 2:  $4 \times 5 = 20$  Portas**

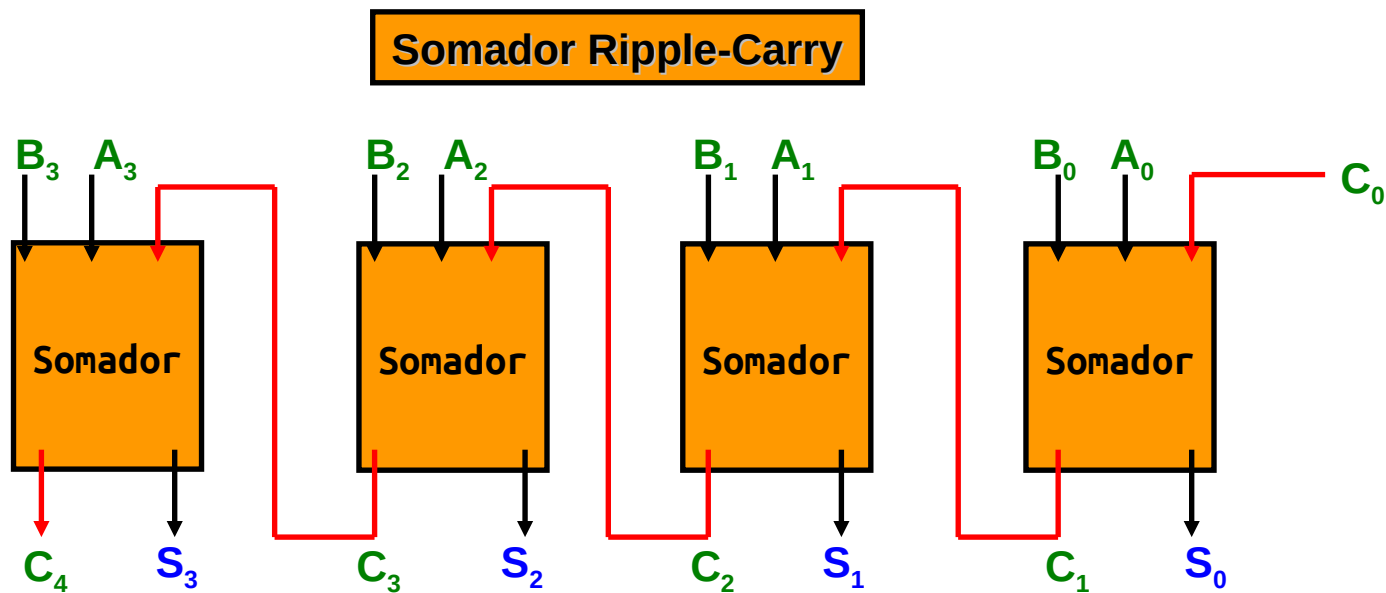
Decisão de Projeto:

-Arquitetura: decide se vai oferecer adição em Complemento de 2

-Organização: decide como implementar (escolhe entre o circuito 1 e o circuito 2)

# Somador de Alto Desempenho

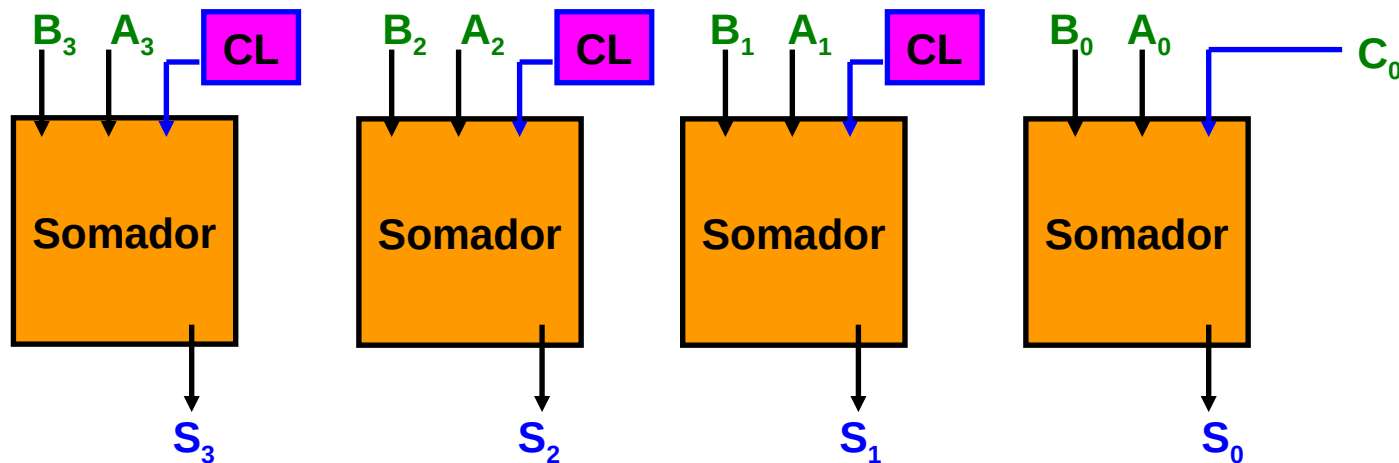
## Somador Convencional



**Somador Convencional: Atrasos para propagar o carry**

# Somador de Alto Desempenho

## Somador Carry Lookahead



CL: Lógica para antecipar o *carry* sem passar pelo somador

# Aritmética Computacional

| Entradas |   |          | Saídas |           |                    |
|----------|---|----------|--------|-----------|--------------------|
| A        | B | $C_{in}$ | S      | $C_{out}$ | $C_{out}$          |
| 0        | 0 | 0        | 0      | 0         | "nada" 0           |
| 0        | 0 | 1        | 1      | 0         | "nada" 0           |
| 0        | 1 | 0        | 1      | 0         | "propaga" $C_{in}$ |
| 0        | 1 | 1        | 0      | 1         | "propaga" $C_{in}$ |
| 1        | 0 | 0        | 1      | 0         | "propaga" $C_{in}$ |
| 1        | 0 | 1        | 0      | 1         | "propaga" $C_{in}$ |
| 1        | 1 | 0        | 0      | 1         | "gera" 1           |
| 1        | 1 | 1        | 1      | 1         | "gera" 1           |

| A | B | $C_{out}$          |
|---|---|--------------------|
| 0 | 0 | 0 "nada"           |
| 0 | 1 | $C_{in}$ "propaga" |
| 1 | 0 | $C_{in}$ "propaga" |
| 1 | 1 | 1 "gera"           |



# Somador de Alto Desempenho

## Expressão do Carry do Somador

$$C_{i+1} = A_i B_i + A_i C_i + B_i C_i$$

### 1. Fatorando a expressão

$$C_{i+1} = A_i B_i + C_i (A_i + B_i)$$

### 2. Chamando $A_i B_i$ de $G_i$ e $A_i + B_i$ de $P_i$

$$C_{i+1} = G_i + P_i C_i$$

### 3. Substituindo os índices para obter os carries para um somador de 4 bits

$$C_1 = G_0 + P_0 C_0$$

### 4. Para simplificar a análise, vamos considerar $C_0=0$ para soma

$$C_1 = G_0$$

| A | B | C-out |           |
|---|---|-------|-----------|
| 0 | 0 | 0     | “nada”    |
| 0 | 1 | C-in  | “propaga” |
| 1 | 0 | C-in  | “propaga” |
| 1 | 1 | 1     | “gera”    |

# Somador de Alto Desempenho

## Expressão do Carry do Somador

4. Para simplificar a análise, vamos considerar  $C_0=0$  para soma

$$C_1 = G_0$$

$$C_2 = G_1 + P_1 C_1$$

5. Substituindo  $C_1=G_0$

$$C_2 = G_1 + P_1 G_0$$

6. Obtendo  $C_3$

$$C_3 = G_2 + P_2 C_2$$

7. Substituindo  $C_2 = G_1 + P_1 G_0$

$$C_3 = G_2 + P_2(G_1 + P_1 G_0) \Rightarrow C_3 = G_2 + P_2 G_1 + P_2 P_1 G_0$$

# Somador de Alto Desempenho

## Expressão do Carry do Somador

### 8. Obtendo $C_4$

$$C_4 = G_3 + P_3 C_3 \Rightarrow C_4 = G_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0) \Rightarrow$$

$$C_4 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0$$

# Exercício prático

1. Faça o diagrama de blocos do circuito somador de 4 bits com a Lógica *Carry Lookahead* – somador de alto desempenho no Logisim.
2. Calcule o atraso para gerar os carries.

# Próxima aula

→ Circuitos sequenciais