



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

BCC32B – Elementos de Lógica Digital
Prof. Rodrigo Hübner

Aula 11 – Aritmética computacional: adição e subtração

Aritmética Computacional

Circuitos Aritméticos: utilizados para construir a ULA

Adição

Exemplo de adição em decimal (dígitos de 0 a 9):

3 7 6	3 7 6	3 7 6	3 7 6
+ 4 6 1	+ 4 6 1	+ 4 6 1	+ 4 6 1
<hr/>	<hr/>	<hr/>	<hr/>
	7	3 7	8 3 7

Cada posição só pode representar um dígito, por isso, gera um *carry* (vai um)

Aritmética Computacional

Adição em Binário :

Exemplo

a) $\begin{array}{r} 0 \\ + 0 \\ \hline 0 \end{array}$ b) $\begin{array}{r} 0 \\ + 1 \\ \hline 1 \end{array}$ c) $\begin{array}{r} 1 \\ + 0 \\ \hline 1 \end{array}$ d) $\begin{array}{r} 1 \\ + 1 \\ \hline 10 \end{array}$

Cada posição só pode representar um dígito, por isso, gera um *carry*

$\begin{array}{r} 10101 \\ + 00111 \\ \hline \end{array}$ $\begin{array}{r} 10101 \\ + 00111 \\ \hline 0 \end{array}$ $\begin{array}{r} 10101 \\ + 00111 \\ \hline 00 \end{array}$ $\begin{array}{r} 10101 \\ + 00111 \\ \hline 100 \end{array}$ $\begin{array}{r} 10101 \\ + 00111 \\ \hline 1100 \end{array}$ $\begin{array}{r} 10101 \\ + 00111 \\ \hline 11100 \end{array}$

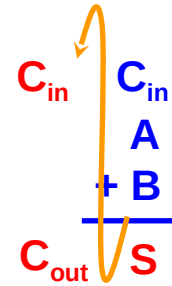
C_{in} (pink arrow pointing to the 1 in the 5th position of the second example)

C_{out} (pink arrow pointing to the 0 in the 5th position of the second example)

Soma (blue arrow pointing to the 11100 result of the sixth example)

Aritmética Computacional

Entradas			Saídas	
A	B	C_{in}	S	C_{out}
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



Aritmética Computacional

Entradas			Saídas	
A	B	C _{in}	S	C _{out}
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

$\overline{A} \overline{B} C_{in}$

$\overline{A} B \overline{C}_{in}$

$A \overline{B} \overline{C}_{in}$

$A B C_{in}$

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C}_{in} + A \overline{B} \overline{C}_{in} + A B C_{in}$$

Aritmética Computacional

Entradas			Saídas	
A	B	C _{in}	S	C _{out}
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

$\overline{A} B C_{in}$

$A \overline{B} C_{in}$

$A B \overline{C}_{in}$

$A B C_{in}$

$$C_{out} = \overline{A} B C_{in} + A \overline{B} C_{in} + A B \overline{C}_{in} + A B C_{in}$$

Aritmética Computacional

Simplificando as expressões

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C}_{in} + A \overline{B} \overline{C}_{in} + A B C_{in}$$

$$S = \overline{A} (\overline{B} C_{in} + B \overline{C}_{in}) + A (\overline{B} \overline{C}_{in} + B C_{in}) \quad \longleftarrow A \text{ e } \overline{A} \text{ em evidência}$$

$$\text{Como } B \oplus C_{in} = \overline{B} C_{in} + B \overline{C}_{in} \text{ e } B \odot C_{in} = \overline{B} \overline{C}_{in} + B C_{in}$$

$$S = \overline{A} (B \oplus C_{in}) + A (B \odot C_{in})$$

$$\text{Fazendo } X = B \oplus C_{in} \text{ e } \overline{X} = B \odot C_{in}$$

$$S = \overline{A} X + A \overline{X}$$

$$S = A \oplus X \quad \longleftarrow \text{Como } X = B \oplus C_{in}$$

$$S = A \oplus B \oplus C_{in}$$

Lembrete

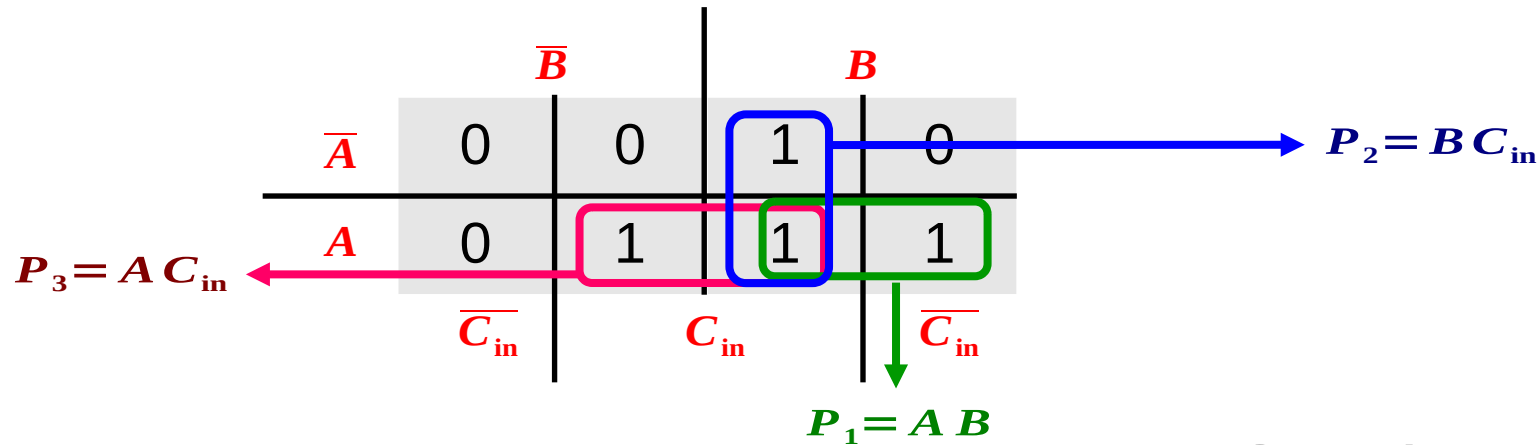
\oplus : XOR

\odot : XNOR

Aritmética Computacional

Simplificando as expressões

$$C_{out} = \bar{A} B C_{in} + A \bar{B} C_{in} + A B \bar{C}_{in} + A B C_{in}$$



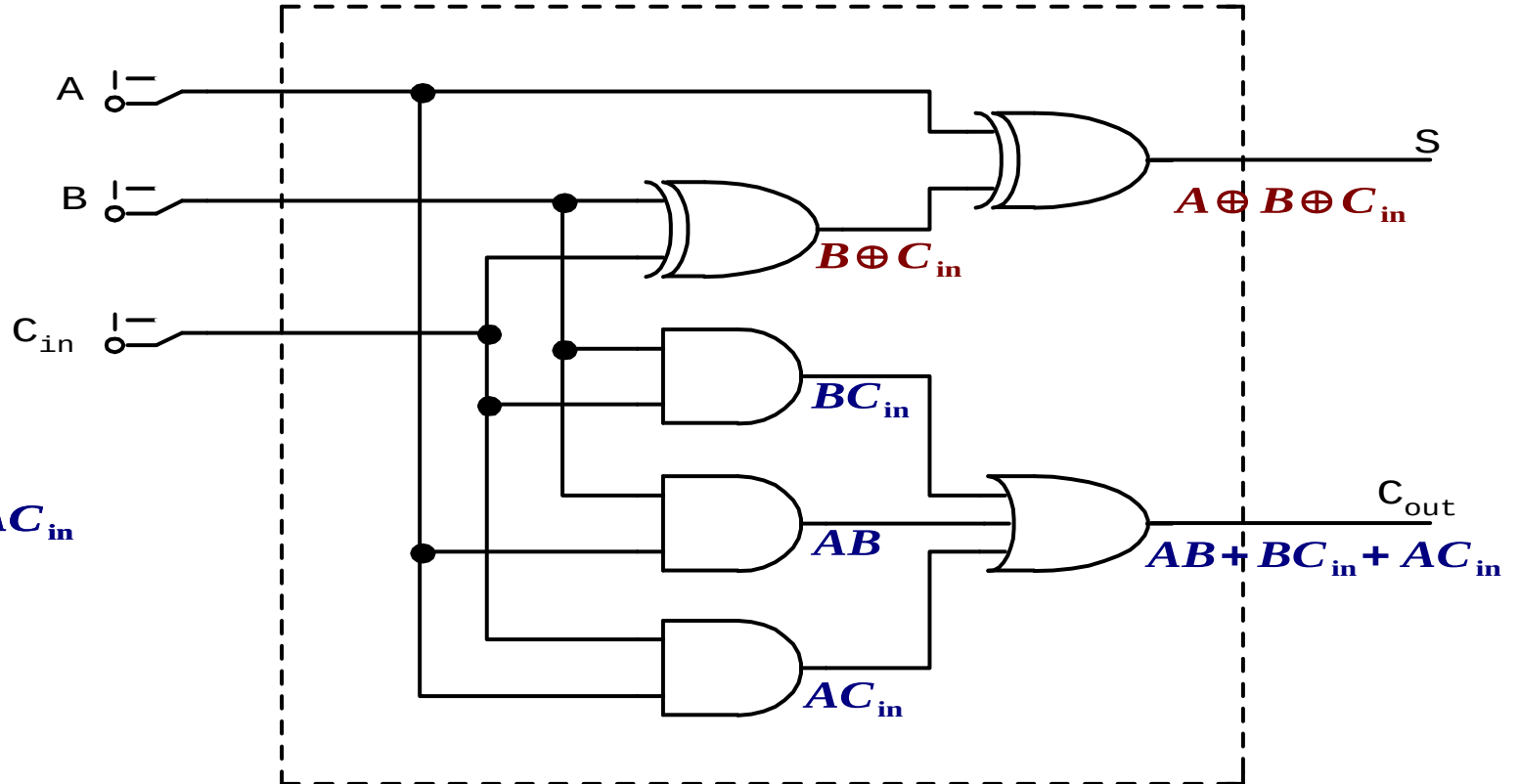
$$C_{out} = AB + BC_{in} + AC_{in}$$

Aritmética Computacional

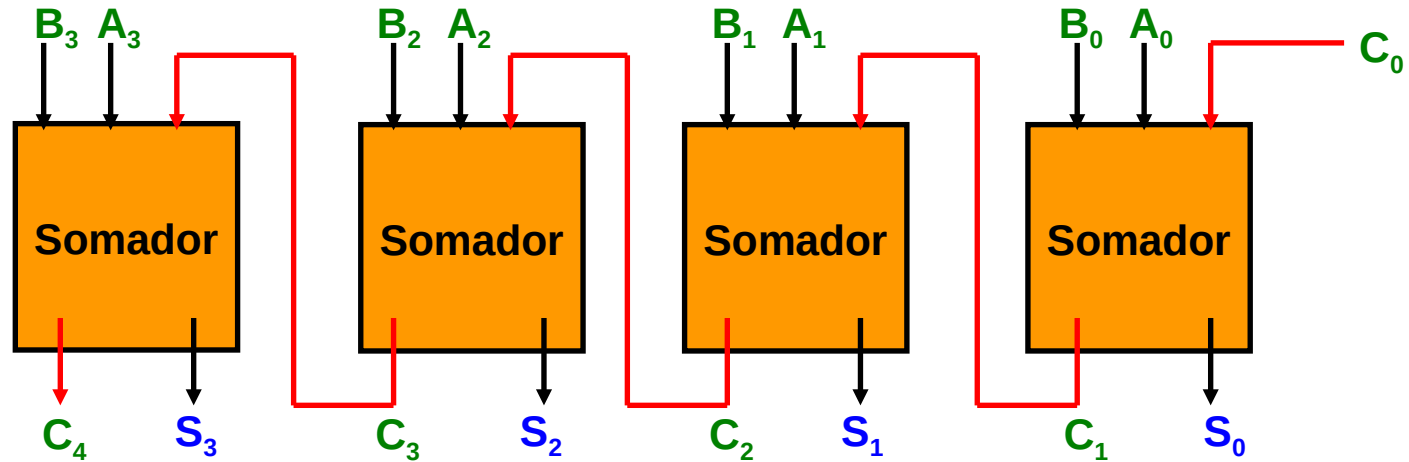
Circuito Somador

$$S = A \oplus B \oplus C_{in}$$

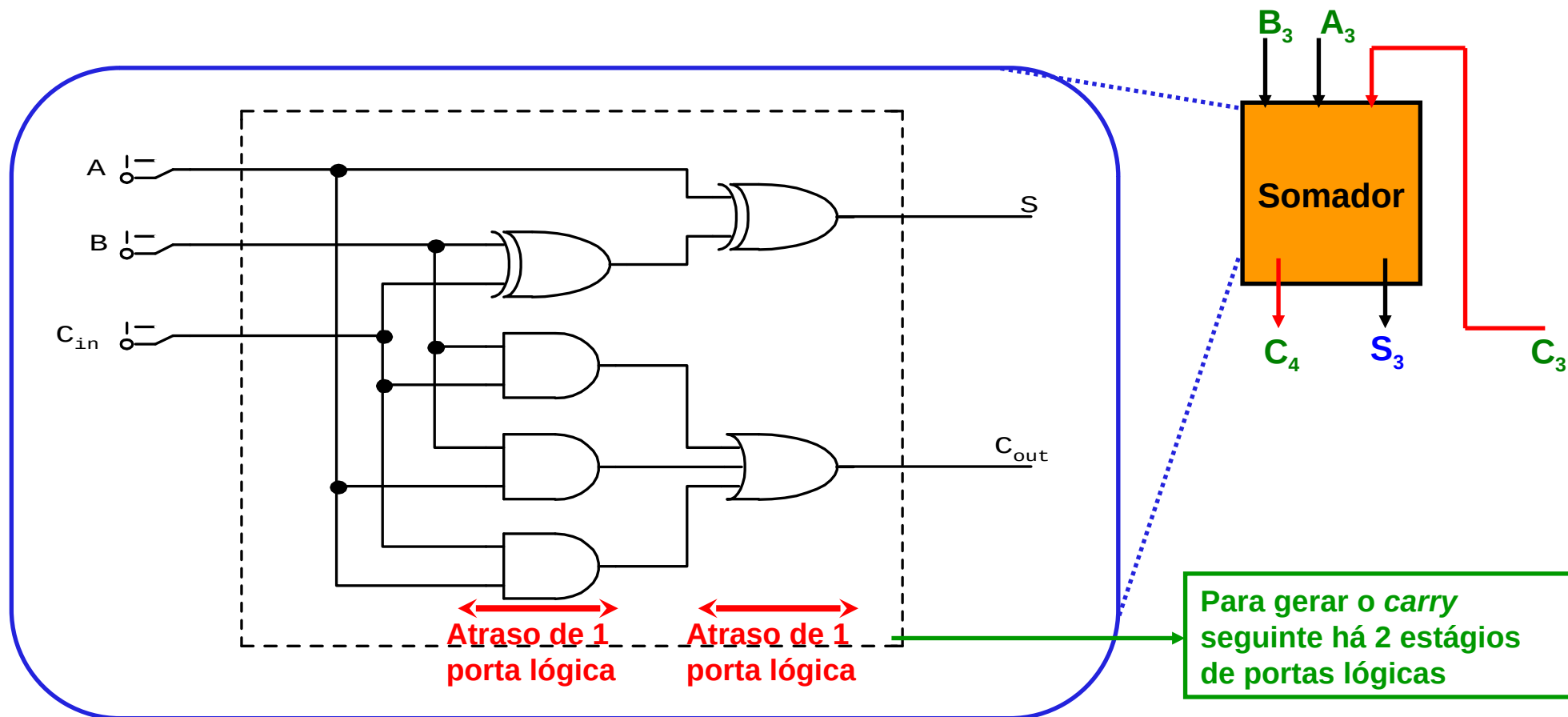
$$C_{out} = AB + BC_{in} + AC_{in}$$



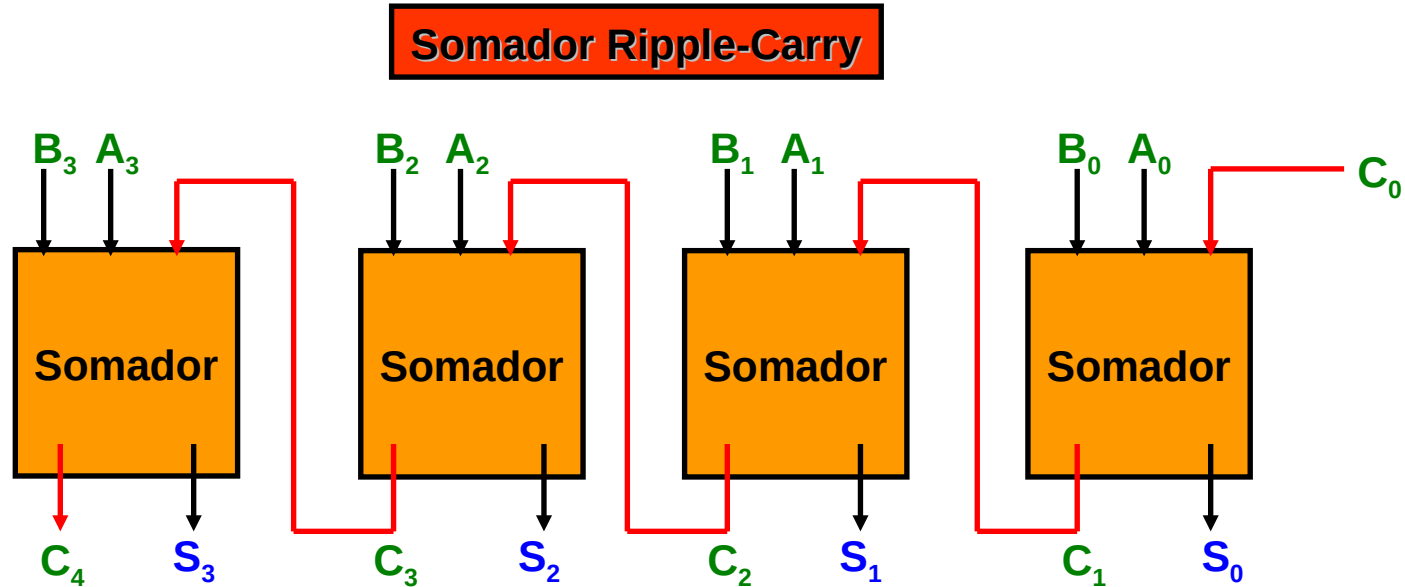
Somador de 4 bits



Somador Bit Slice



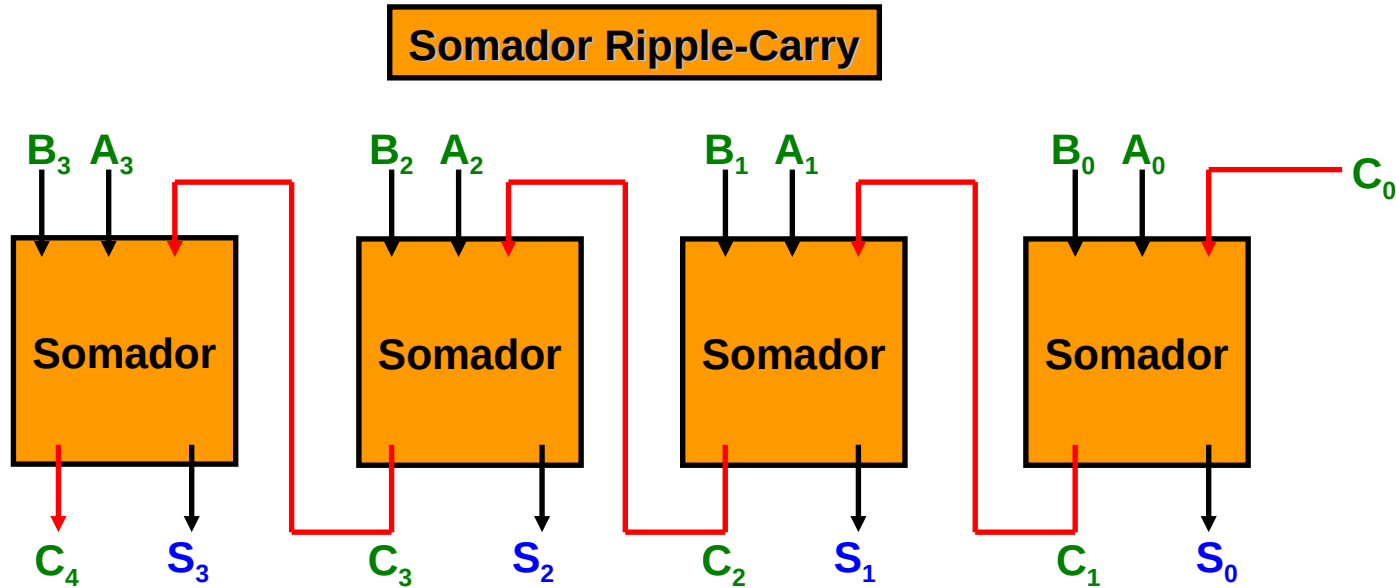
Somador de 4 bits



- Ripple-Carry: Ondulação ou Propagação do Carry. Carry-Out de um estágio se transforma no Carry-In do estágio seguinte.
- A_i e B_i “alimentam” os somadores em paralelo, mas o circuito deve esperar a propagação dos Carries para concluir a operação.

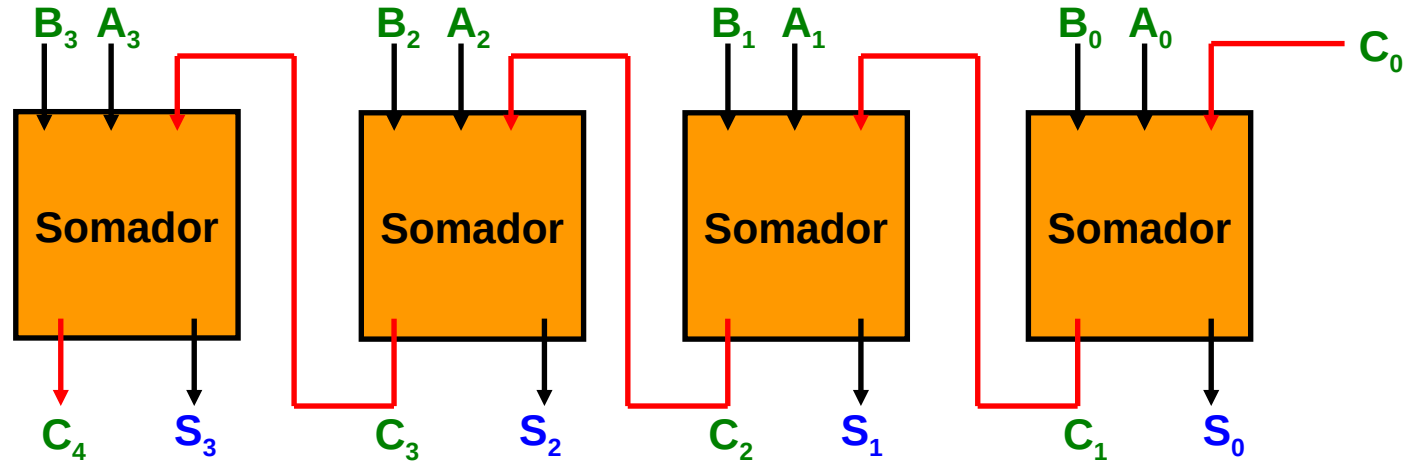
Exercícios

2. Considere um somador ripple-carry de 4 bits. Considere que as portas lógicas têm um atraso de 1ns. Qual é o atraso causado pelo somador ripple-carry para propagar o carry por todos os somadores?



Soluções

2)



Para gerar:

- C_1 consome-se 2ns
- C_2 consome-se 4ns
- C_3 consome-se 6ns
- C_4 consome-se 8ns

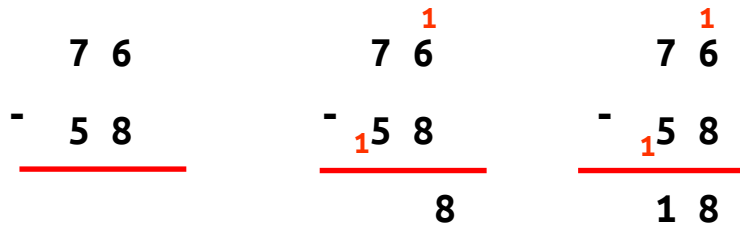
Quanto maior o número de bits do somador,
maior o atraso para gerar o carry final

Aritmética Computacional

Aritmética Computacional

Subtração

Exemplo de subtração em decimal (dígitos de 0 a 9):


$$\begin{array}{r} 76 \\ - 58 \\ \hline \end{array}$$
$$\begin{array}{r} 7\overset{1}{6} \\ - 158 \\ \hline 8 \end{array}$$
$$\begin{array}{r} 7\overset{1}{6} \\ - 158 \\ \hline 18 \end{array}$$

Aritmética Computacional

Aritmética Computacional

Subtração em Binário

Gera um “empresta 1” (carry out) da coluna seguinte: a 1ª coluna passa a valer $2_{10}=10_2$

O carry out será subtraído da coluna seguinte na continuação da operação

Exemplo

a)

$$\begin{array}{r} 0 \\ - 0 \\ \hline 0 \end{array}$$

b)

$$\begin{array}{r} 1 \\ 0 \\ - 1 \\ \hline 1 \end{array}$$

c)

$$\begin{array}{r} 1 \\ - 0 \\ \hline 1 \end{array}$$

d)

$$\begin{array}{r} 1 \\ - 1 \\ \hline 0 \end{array}$$

Aritmética Computacional

Aritmética Computacional

Subtração em Binário :

Exemplo

Gera um “empresta 1” (carry out) da coluna seguinte: a 1ª coluna passa a valer $10_2 = 2_{10}$

The diagram illustrates the binary subtraction process for $1010 - 0011$ through six stages:

- Stage 1:** Initial setup: $1010 - 0011$.
- Stage 2:** Borrow from the 4th column to the 3rd column. The 3rd column becomes 10 (2 in decimal). A blue arrow labeled c_{in} points to the 3rd column.
- Stage 3:** Borrow from the 3rd column to the 2nd column. The 2nd column becomes 10 (2 in decimal).
- Stage 4:** Borrow from the 2nd column to the 1st column. The 1st column becomes 10 (2 in decimal).
- Stage 5:** Borrow from the 1st column to the 0th column. The 0th column becomes 10 (2 in decimal).
- Stage 6:** Final result: 0111 .

The final result is shown as 0111 with a blue arrow pointing to it labeled "Subtração".

Exercício

1. Obter a Tabela Verdade para o circuito subtrator de 1 bit (considere como entradas: A , B e C_{in} ; e como saídas: S e C_{out}).
2. Obtenha as expressões para a subtração S e para o C_{out} a partir da Tabela Verdade.
3. Desenhe o diagrama de portas lógicas do circuito subtrator.

Soluções

1)

Entradas

Saídas

Tabela Verdade para o Subtrator

A	B	C _{in}	S	C _{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Soluções

2)

Entradas			Saídas	
A	B	C _{in}	S	C _{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C}_{in} + A \overline{B} \overline{C}_{in} + A B C_{in}$$

Soluções

2)

Entradas			Saídas	
A	B	C _{in}	S	C _{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C}_{in} + A \overline{B} \overline{C}_{in} + A B C_{in}$$

$$C_{out} = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C}_{in} + \overline{A} B C_{in} + A B C_{in}$$

Aritmética Computacional

2)

Simplificando as expressões

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C}_{in} + A \overline{B} \overline{C}_{in} + A B C_{in}$$

$$S = \overline{A} (\overline{B} C_{in} + B \overline{C}_{in}) + A (\overline{B} \overline{C}_{in} + B C_{in}) \quad \longleftarrow \text{A e } \overline{A} \text{ em evidência}$$

$$\text{Como } B \oplus C_{in} = \overline{B} C_{in} + B \overline{C}_{in} \quad \text{e} \quad B \odot C_{in} = \overline{B} \overline{C}_{in} + B C_{in}$$

$$S = \overline{A} (B \oplus C_{in}) + A (B \odot C_{in})$$

$$\text{Fazendo } X = B \oplus C_{in} \quad \text{e} \quad \overline{X} = B \odot C_{in}$$

$$S = \overline{A} X + A \overline{X}$$

$$S = A \oplus X \quad \longleftarrow \text{Como } X = B \oplus C_{in}$$

$$S = A \oplus B \oplus C_{in}$$

Lembrete

\oplus : XOR

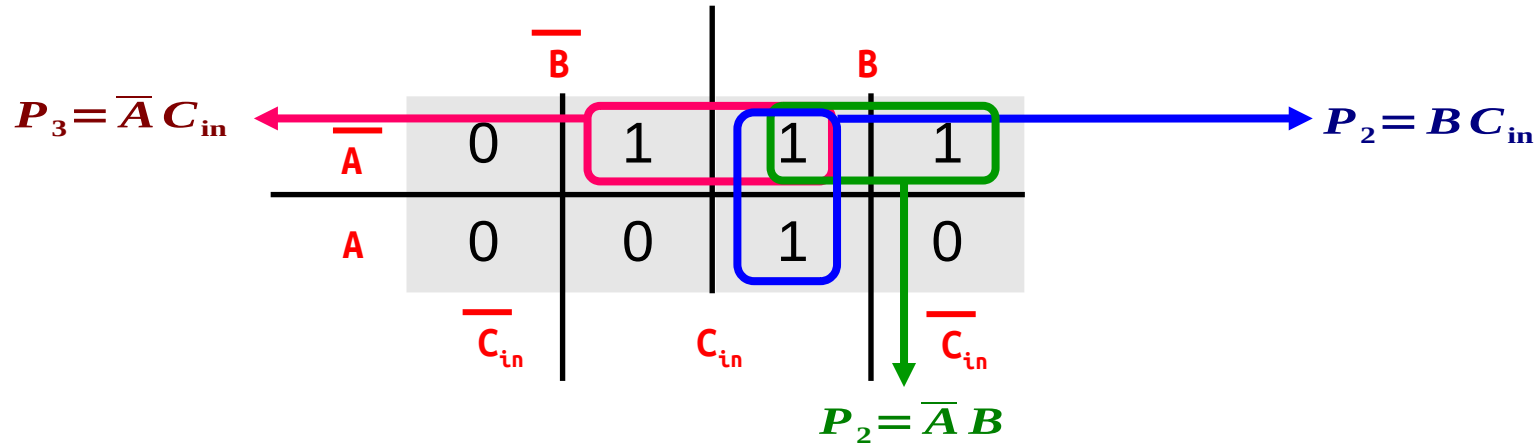
\odot : XNOR

Soluções

2)

Simplificando as expressões

$$C_{out} = \overline{A}\overline{B}C_{in} + \overline{A}B\overline{C}_{in} + \overline{A}BC_{in} + ABC_{in}$$



$$C_{out} = \overline{A}B + BC_{in} + \overline{A}C_{in}$$

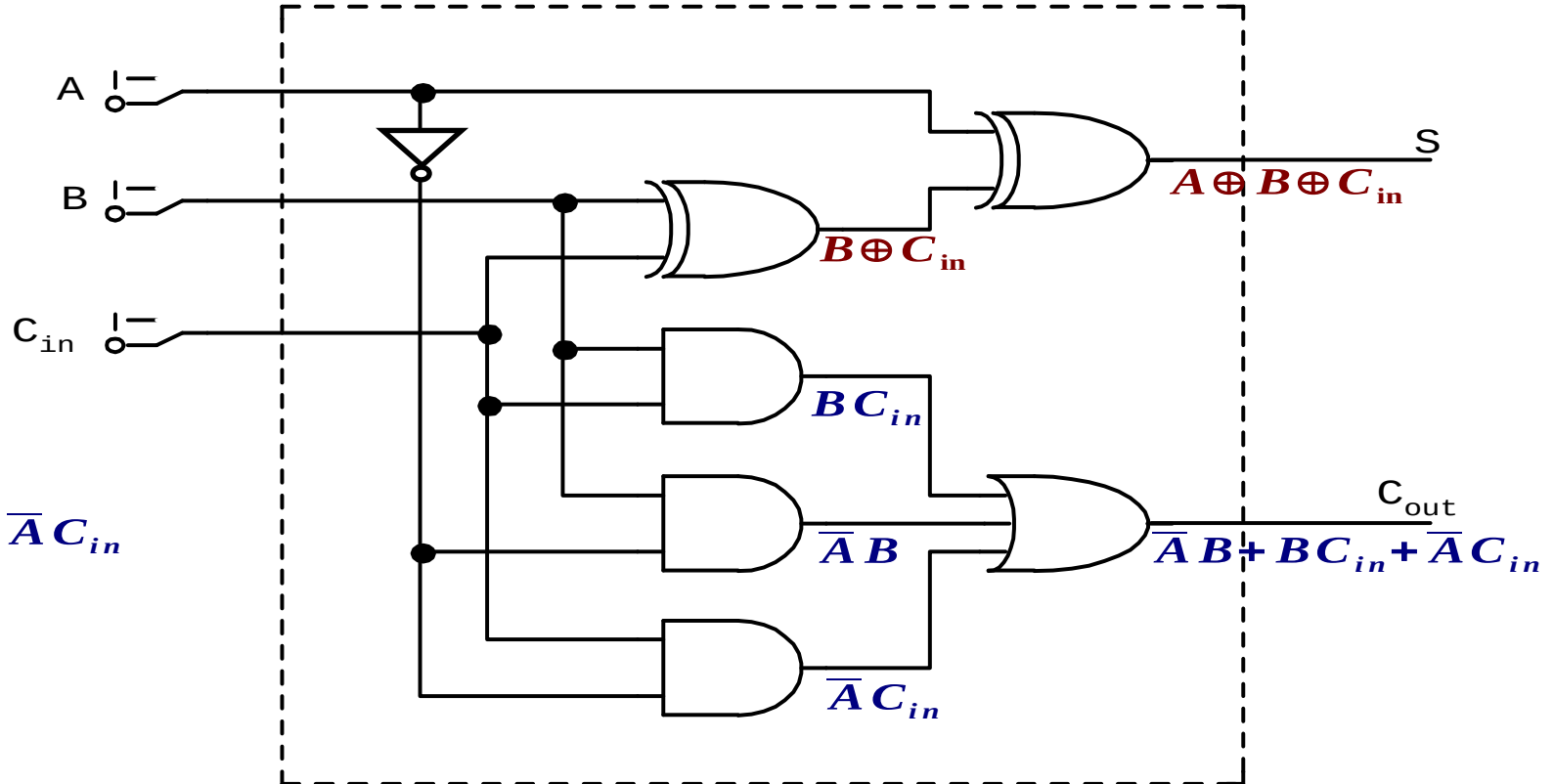
Soluções

3)

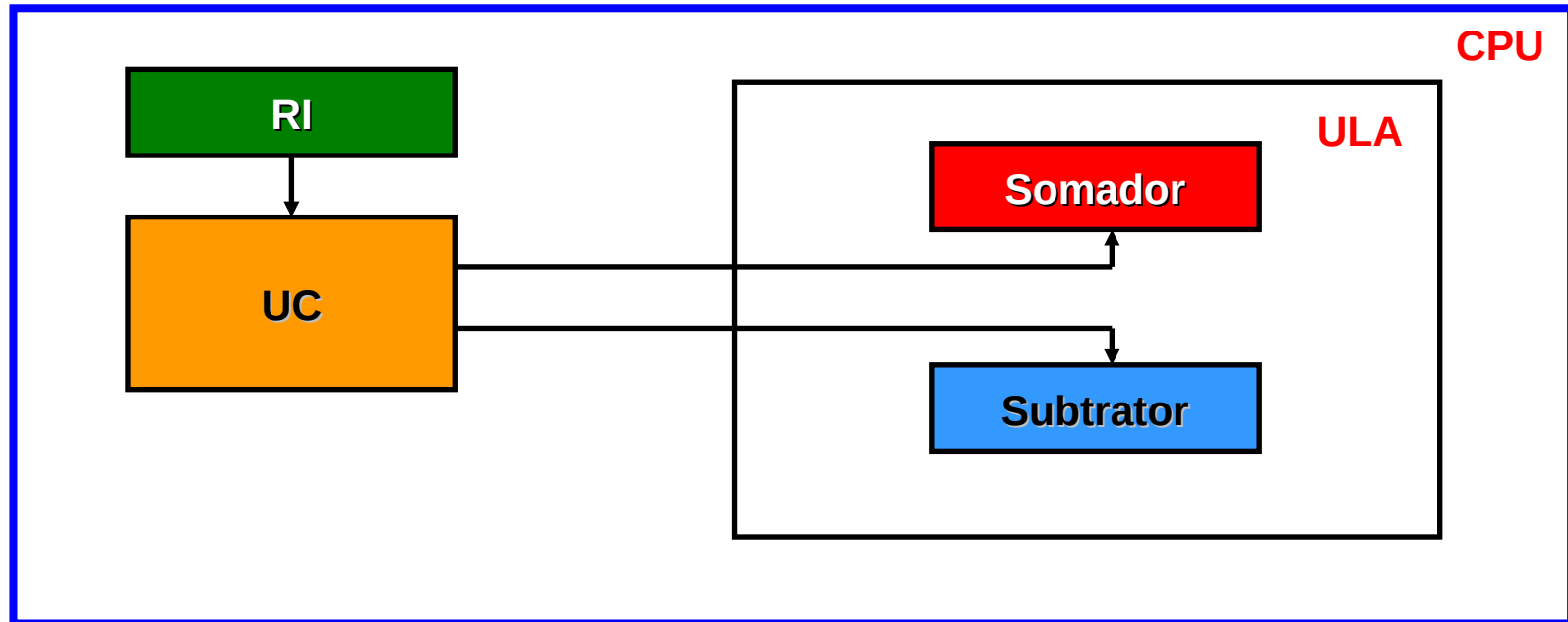
Circuito Subtrator

$$S = A \oplus B \oplus C_{in}$$

$$C_{out} = \overline{A}B + BC_{in} + \overline{A}C_{in}$$



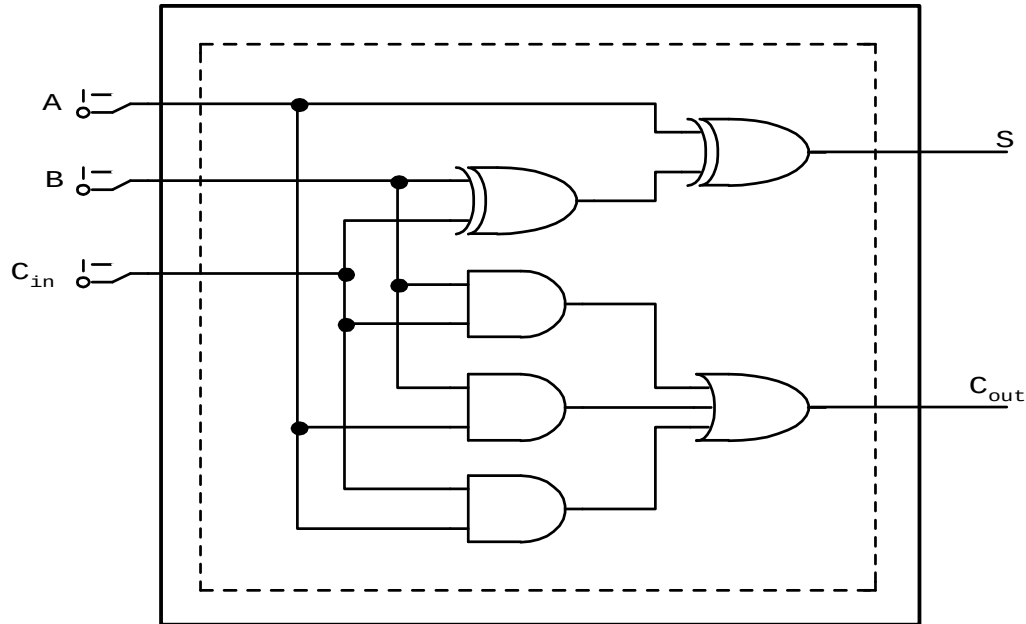
ULA: Somador e Subtrator



ULA com 2 circuitos para efetuar a adição e a subtração

ULA: Somador e Subtrator

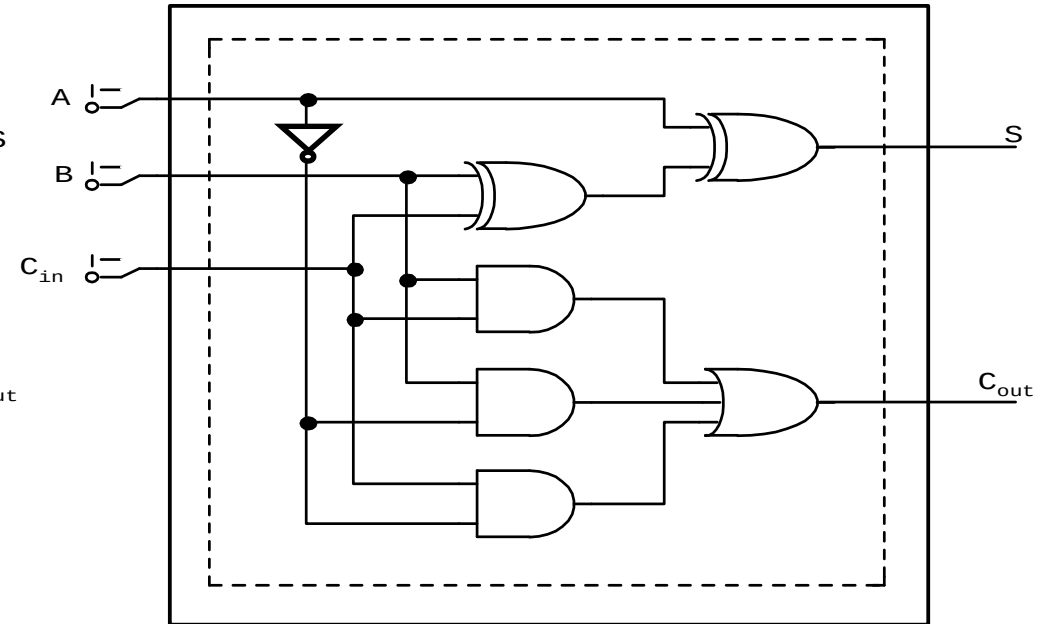
Circuito Somador



$$S = A \oplus B \oplus C_{in}$$

$$C_{out} = AB + BC_{in} + AC_{in}$$

Circuito Subtrator



$$S = A \oplus \bar{B} \oplus C_{in}$$

$$C_{out} = \bar{A}B + BC_{in} + \bar{A}C_{in}$$

ULA: Somador e Subtrator

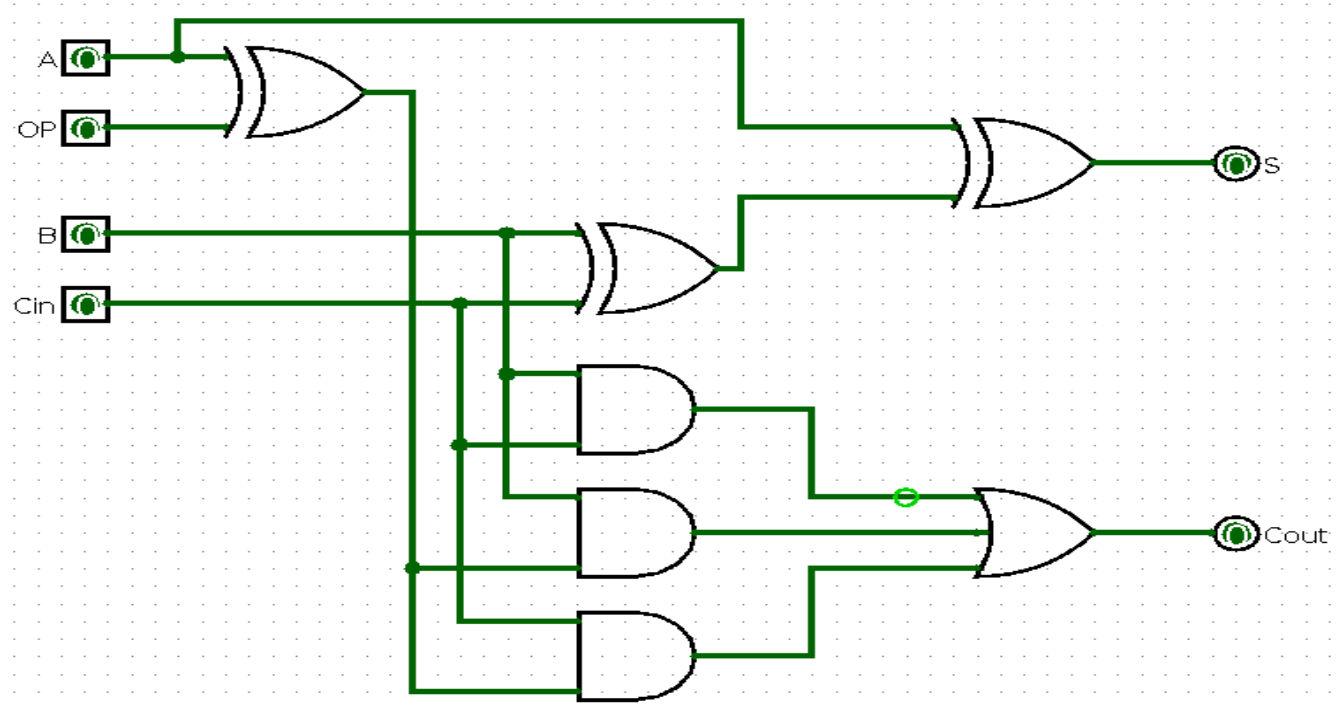
TV da Porta XOR

Entradas

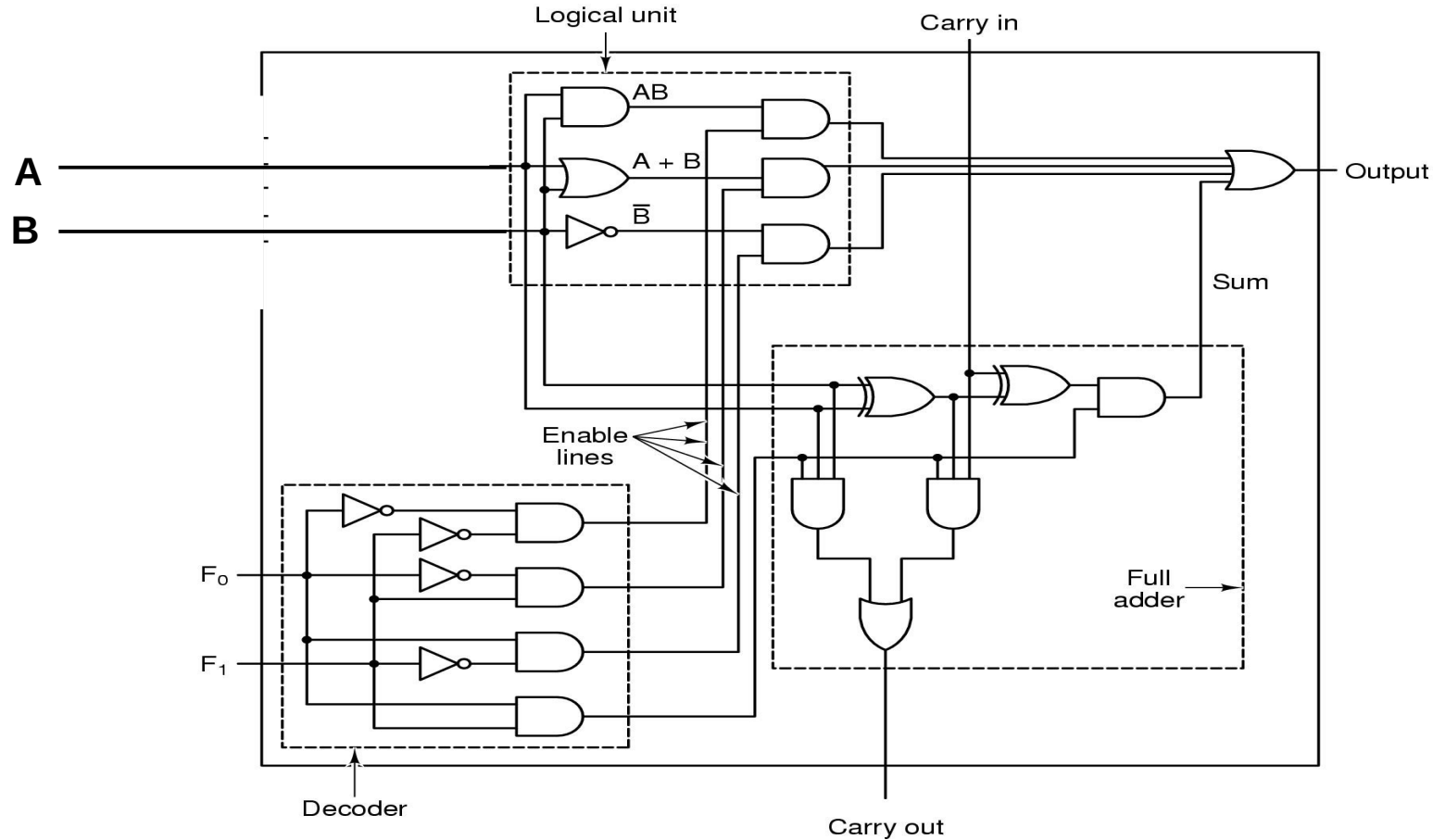
Saída

A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

Circuito Somador/Subtrator



Exemplo: ULA Simplificada



Exercício

Projeto de ULA de 8 bits no Logisim

Projeto de ULA de 8 bits com as seguintes funções:

- 1. AND**
- 2. OR**
- 3. NOT**
- 4. NAND**
- 5. NOR**
- 6. XOR**
- 7. XNOR**
- 8. SOMADOR**

Obs.: Utilize subcircuitos (Lógica, Somador, Decodificador)

Próxima aula

- **Aritmética Computacional**
 - Adição e subtração em complemento de 2
 - Somador de alto desempenho