

Circuitos Aritméticos

Paulo Valim

Soma de números binários

- A soma de números binários é similar a soma em decimal:

Decimal

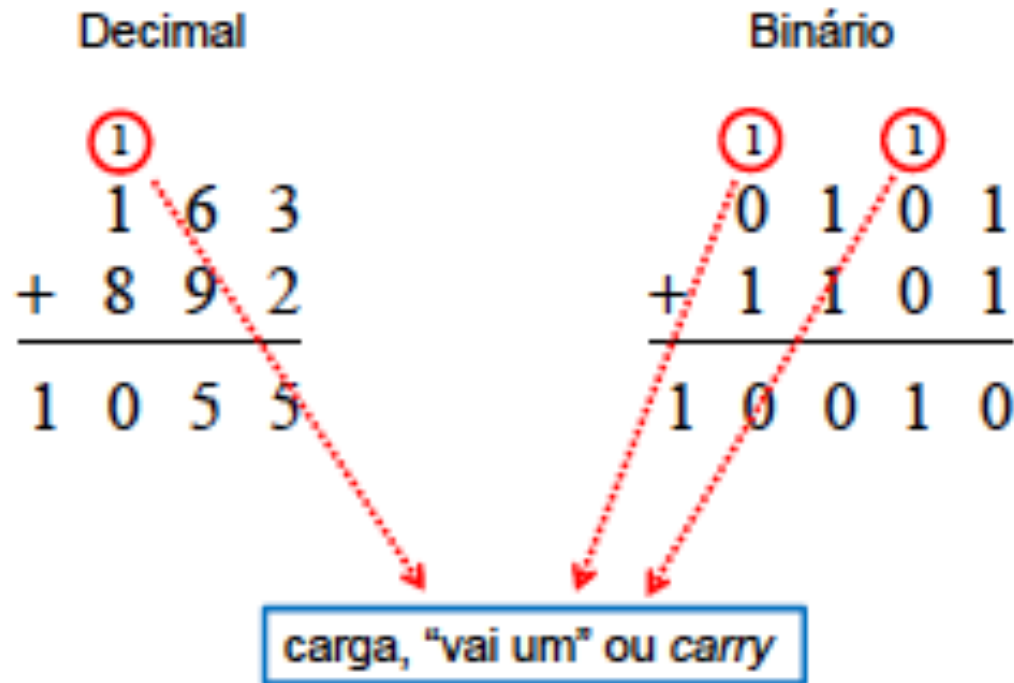
$$\begin{array}{r} 1 \\ 163 \\ + 892 \\ \hline 1055 \end{array}$$

Binário

$$\begin{array}{r} 1 \quad 1 \\ 0101 \\ + 1101 \\ \hline 10010 \end{array}$$

Soma de números binários

- A soma de números binários é similar a soma em decimal:



Soma de números binários

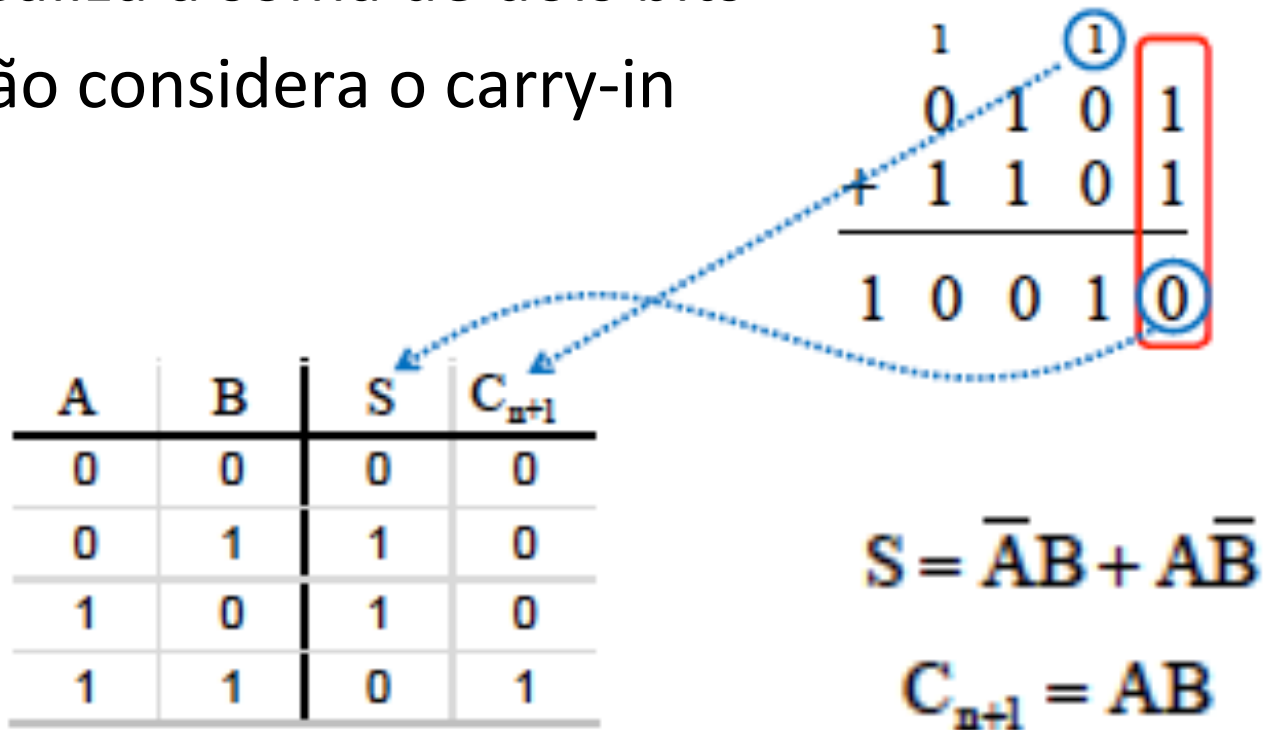
- Meio somador
 - Realiza a soma de dois bits
 - Não considera o carry-in

A	B	S	C_{n+1}
0	0		
0	1		
1	0		
1	1		

$$\begin{array}{rcccc} & 1 & & 1 & \\ & 0 & 1 & 0 & 1 \\ + & 1 & 1 & 0 & 1 \\ \hline 1 & 0 & 0 & 1 & 0 \end{array}$$

Soma de números binários

- Meio somador
 - Realiza a soma de dois bits
 - Não considera o carry-in

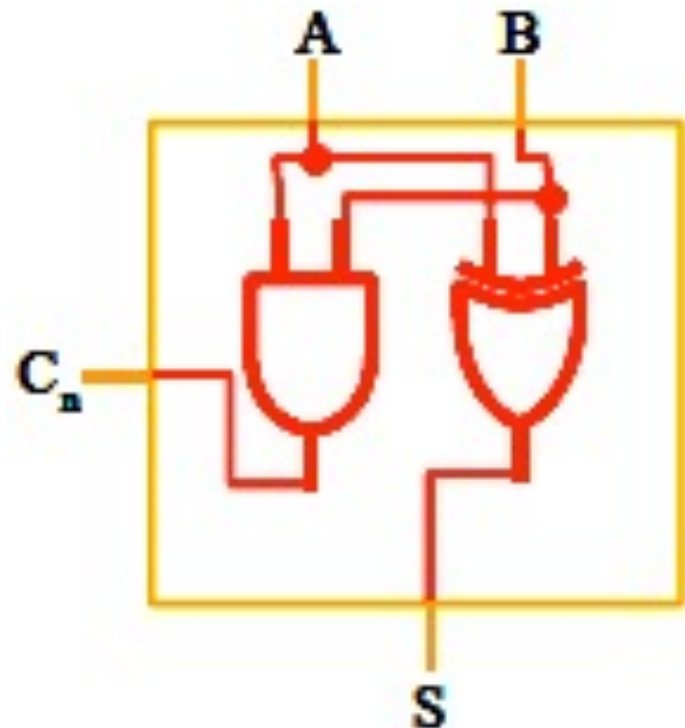


Soma de números binários

- Meio somador
 - Realiza a soma de dois bits
 - Não considera o carry-in

$$S = \bar{A}B + A\bar{B}$$

$$C_{n+1} = AB$$



Soma de números binários

- Somador completo
 - Realiza a soma de dois bits,
 - Considera o carry-in.

A	B	C_n	S	C_{n+1}
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

$$\begin{array}{rcccccc} & & 1 & & 1 & & \\ & & 0 & 1 & 0 & 1 & \\ + & 1 & 1 & 0 & 1 & & \\ \hline 1 & 0 & 0 & 1 & 0 & & \end{array}$$

Soma de números binários

- Somador completo
 - Realiza a soma de dois bits,
 - Considera o carry-in.

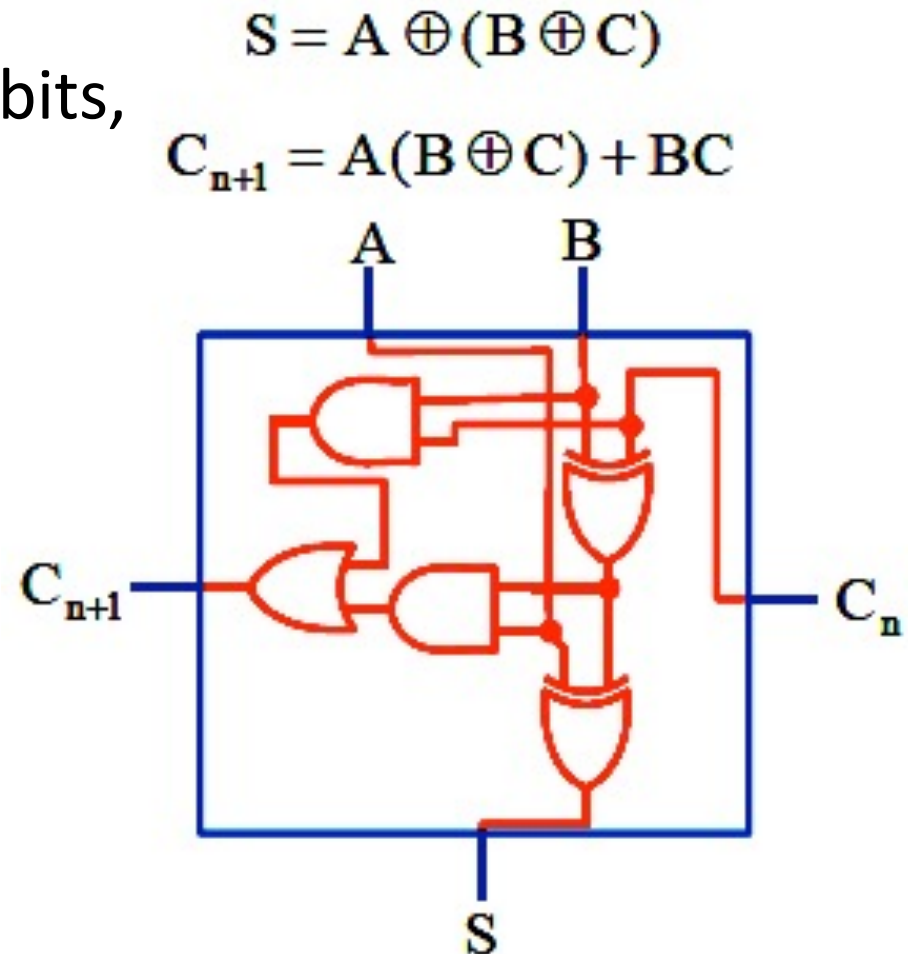
A	B	C_n	S	C_{n+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\begin{array}{rcccc} & 1 & & 1 & \\ & 0 & 1 & 0 & 1 \\ + & 1 & 1 & 0 & 1 \\ \hline 1 & 0 & 0 & 1 & 0 \end{array}$$

Soma de números binários

- Somador completo
 - Realiza a soma de dois bits,
 - Considera o carry-in.

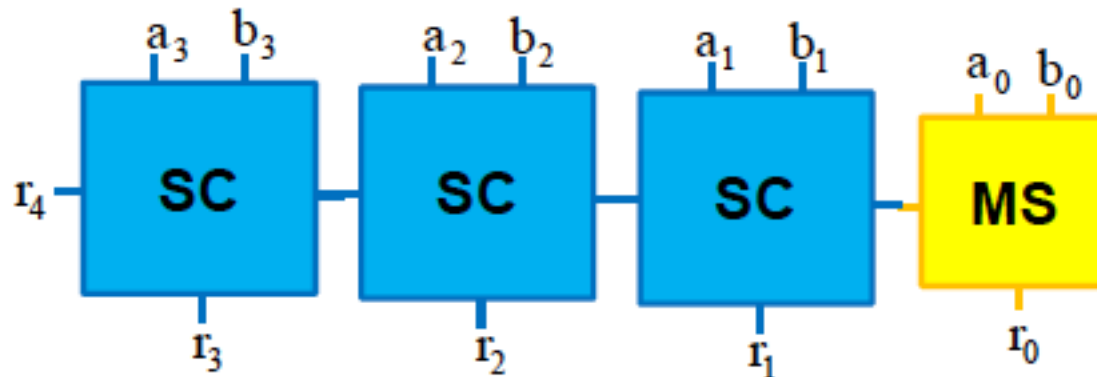
A	B	C_n	S	C_{n+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Soma de números binários

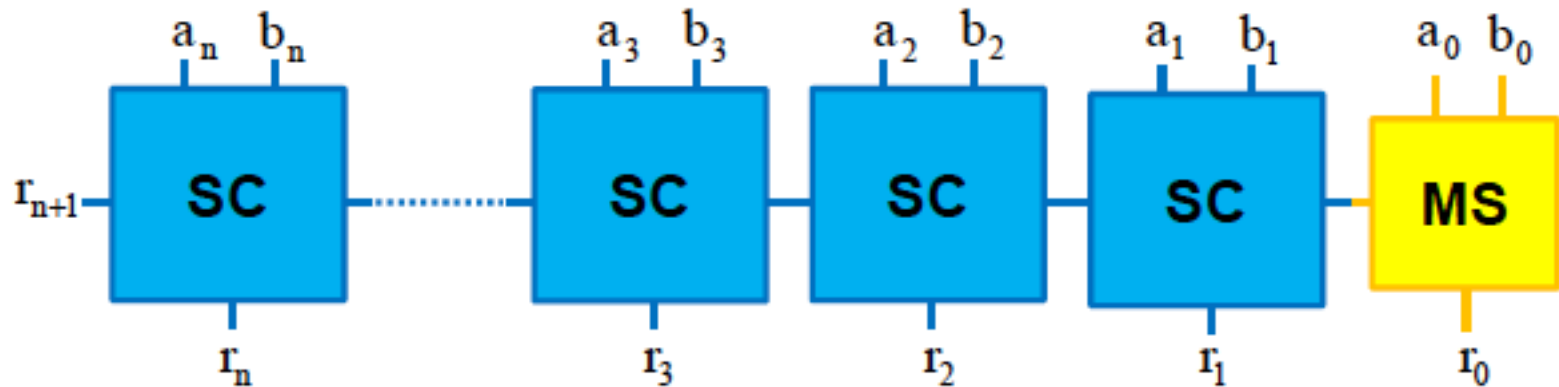
- Somador Binário Paralelo
 - Soma de números de vários bits
 - Exemplo de 4 bits:

$$\begin{array}{rcccc} & a_3 & a_2 & a_1 & a_0 \\ + & b_3 & b_2 & b_1 & b_0 \\ \hline r_4 & r_3 & r_2 & r_1 & r_0 \end{array}$$



Soma de números binários

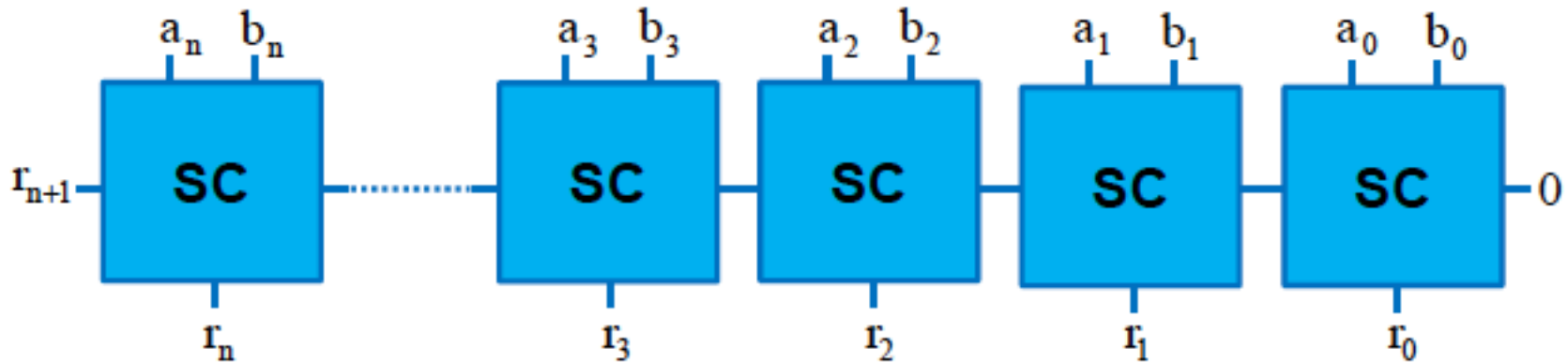
- Somador Binário Paralelo
 - Generalizando:



- Considerando um número fixo de bits para a saída igual ao número de bits de entrada. O **overflow** é indicado por r_{n+1}

Soma de números binários

- Somador Binário Paralelo
 - Outra possibilidade



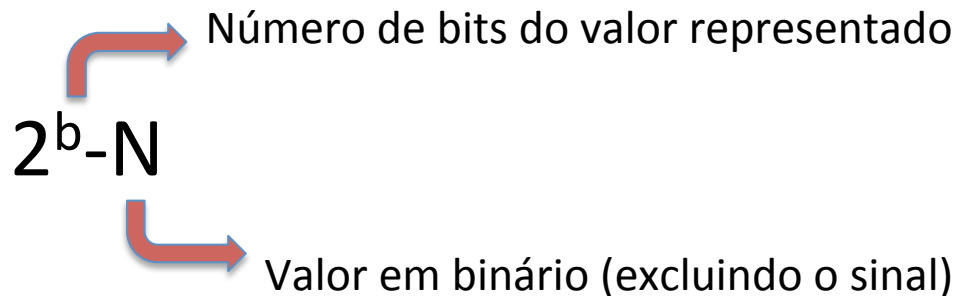
Desvantagem em relação ao circuito anterior: custo ligeiramente mais alto

Subtração de números binários

- Para a operação de subtração é preciso ter uma forma de representar números negativos em binário.
- Vamos lembrar que, se simplesmente adotar-mos um bit como sendo o de sinal, o problema que surge é a necessidade de circuitos diferentes para realizar as operações de adição sobre dados com sinal e sem sinal.
- A representação em complemento-2 tem a vantagem de permitir que um mesmo circuito realize as operações sobre dados com sinal ou sem sinal.

Subtração em binário

- Números com sinal em complemento-2
 - Bit mais significativo é o bit de sinal
 - Números positivos: normais, como foi visto
 - Números negativos: o valor representado é dado por:


$$2^b - N$$

- Exemplo: com 5 bits é possível representar -16 a 15

$0_{10} = 00000_2$	$-1 = 11111_2$
$5_{10} = 00101_2$	$-9 = 10111_2$
$15_{10} = 01111_2$	$-16 = 10000_2$

Subtração em binário

- Alguns exemplos de operações com dados de 5 bits:

$$\begin{array}{r} 3 \\ + 9 \\ \hline 12 \end{array}$$

$$\begin{array}{r} 00^11^11 \\ + 01001 \\ \hline 01100 \end{array}$$

ADIÇÃO

$$\begin{array}{r} 9 \\ - 3 \\ \hline 6 \end{array} \quad \begin{array}{r} 9 \\ + (-3) \\ \hline 6 \end{array}$$

$$\begin{array}{r} 0^110^101 \\ + 11101 \\ \hline \cancel{1}00110 \end{array}$$

SUBTRAÇÃO

Subtração em binário

- Alguns exemplos de operações com dados de 5 bits:

$$\begin{array}{r}
 4 \\
 - 7 \\
 \hline
 - 3
 \end{array}
 \quad
 \begin{array}{r}
 4 \\
 + (-7) \\
 \hline
 -3
 \end{array}
 \quad
 \begin{array}{r}
 00100 \\
 + 11001 \\
 \hline
 11101
 \end{array}$$

$$\begin{array}{r}
 - 2 \\
 - 8 \\
 \hline
 - 10
 \end{array}
 \quad
 \begin{array}{r}
 (-2) \\
 + (-8) \\
 \hline
 -10
 \end{array}
 \quad
 \begin{array}{r}
 1 \\
 11110 \\
 + 11000 \\
 \hline
 110110
 \end{array}$$

Subtração em binário

- Princípio da subtração em binário:
 - Usar a representação do valor correspondente ao minuendo em complemento-2 e fazer

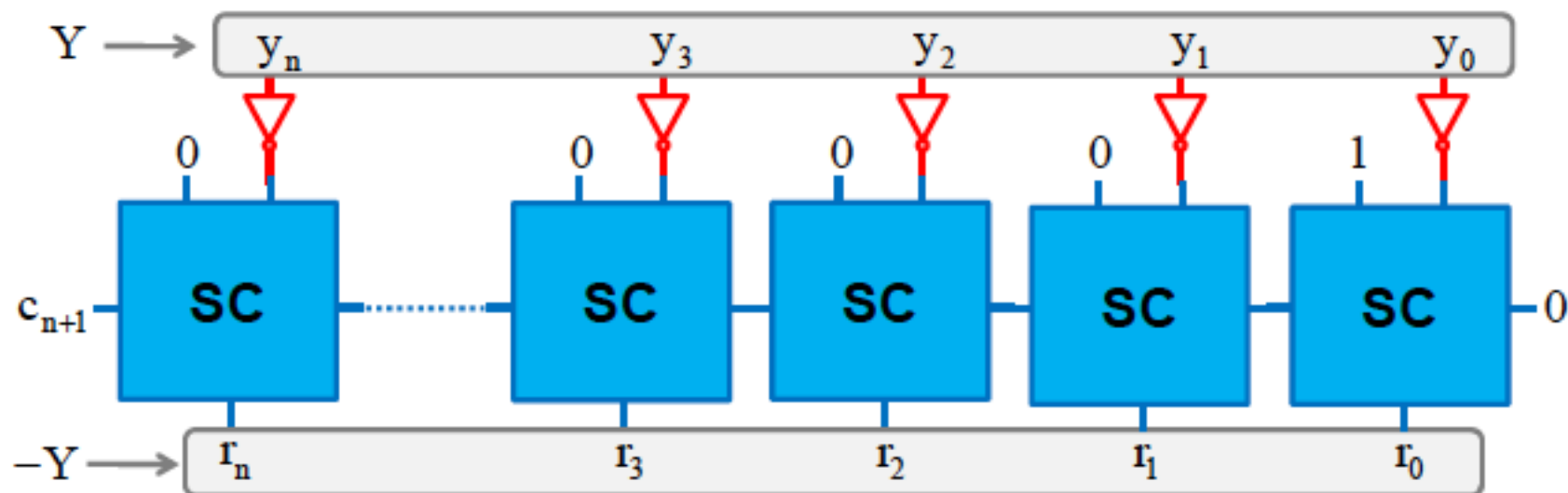
$$X - Y = X + (-Y)$$

adição

número negativo

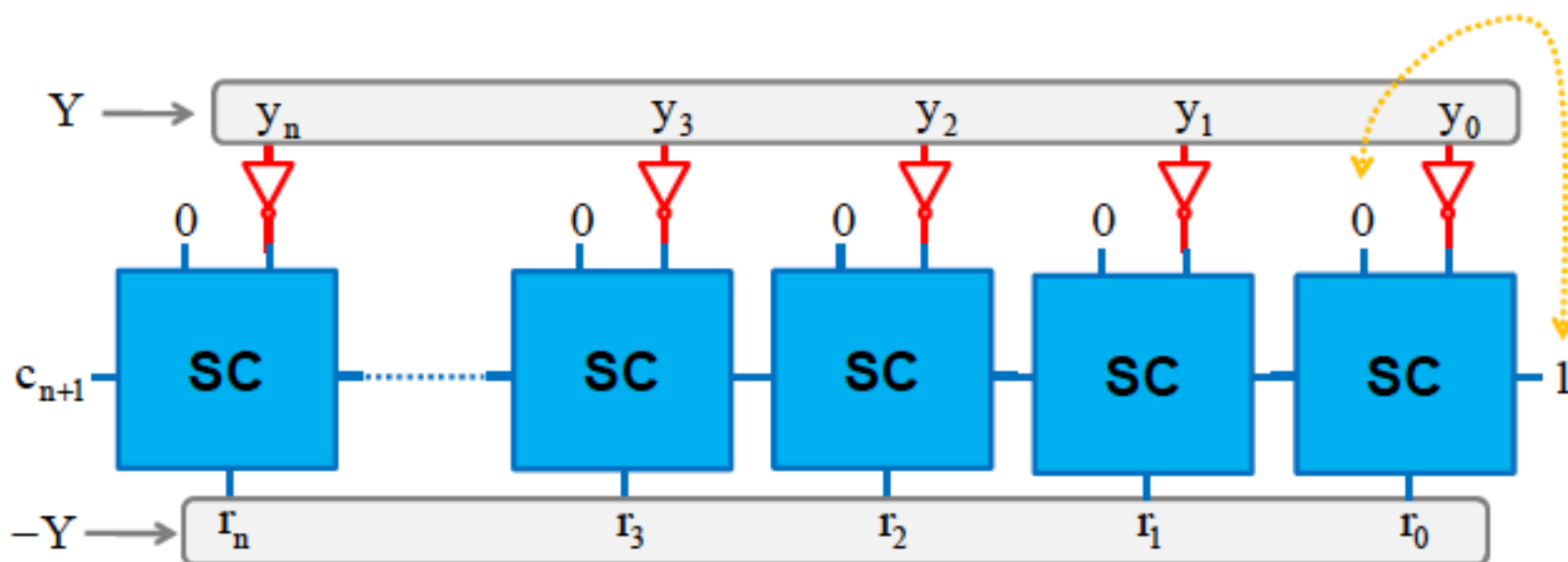
Subtração de números binários

- Forma de obter o correspondente negativo de um valor em complemento-2:
 - Obter o complemento-1 do valor (inverter todos os bits)
 - adicionar 1 ao resultado obtido no passo anterior.



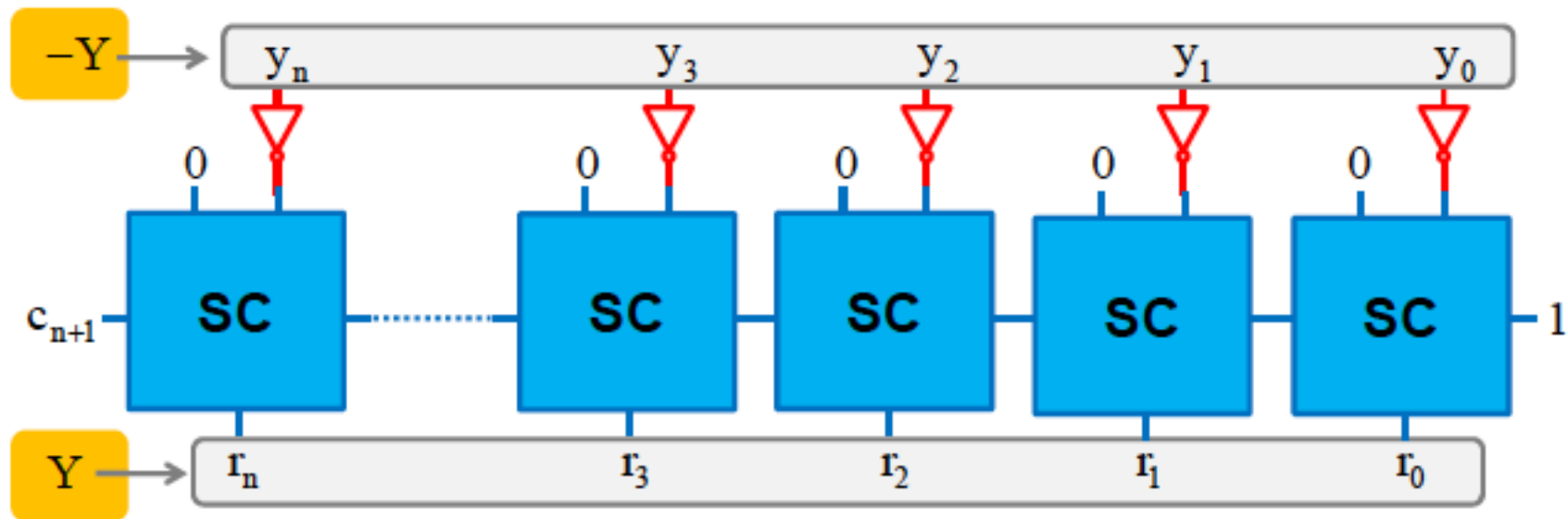
Subtração de números binários

- Forma de obter o correspondente negativo de um valor em complemento-2:
 - Obter o complemento-1 do valor (inverter todos os bits)
 - adicionar 1 ao resultado obtido no passo anterior.



Subtração de números binários

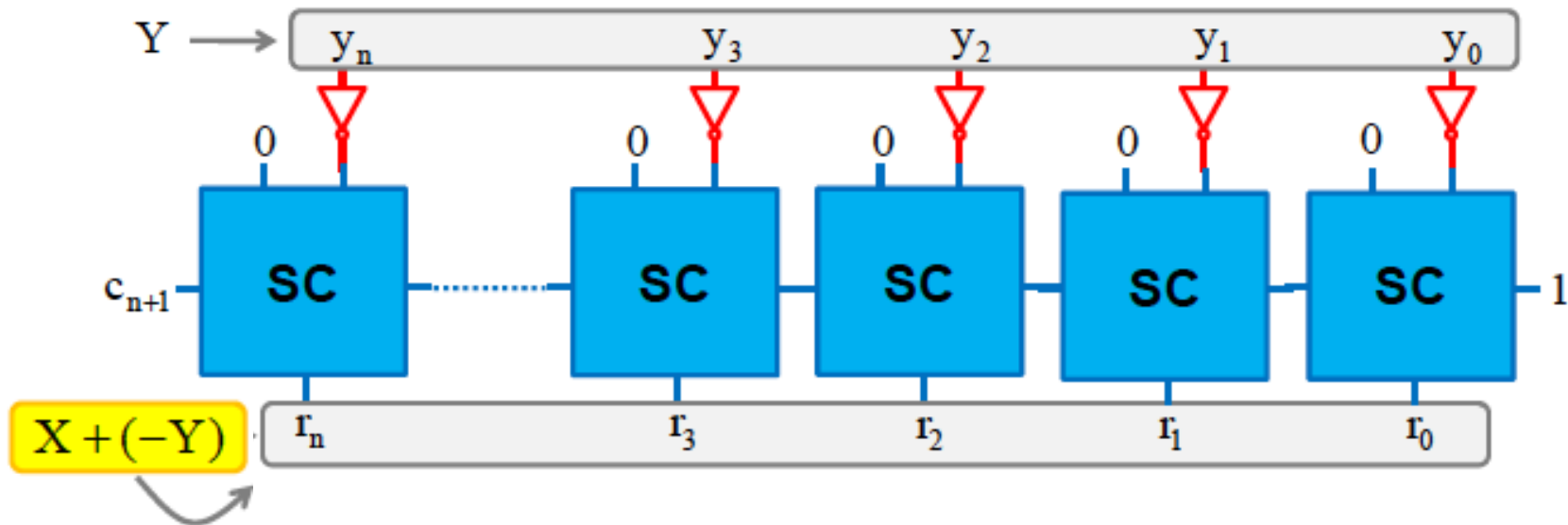
- Funciona também para transformar números negativos em positivos:



Subtração de números binários

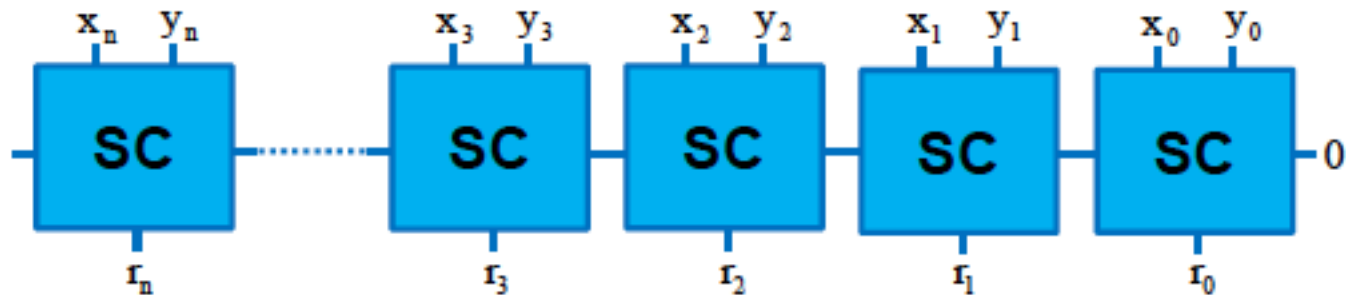
- Como fazer para realizar a operação indicada abaixo usando o circuito anterior?

$$X - Y = X + (-Y)$$

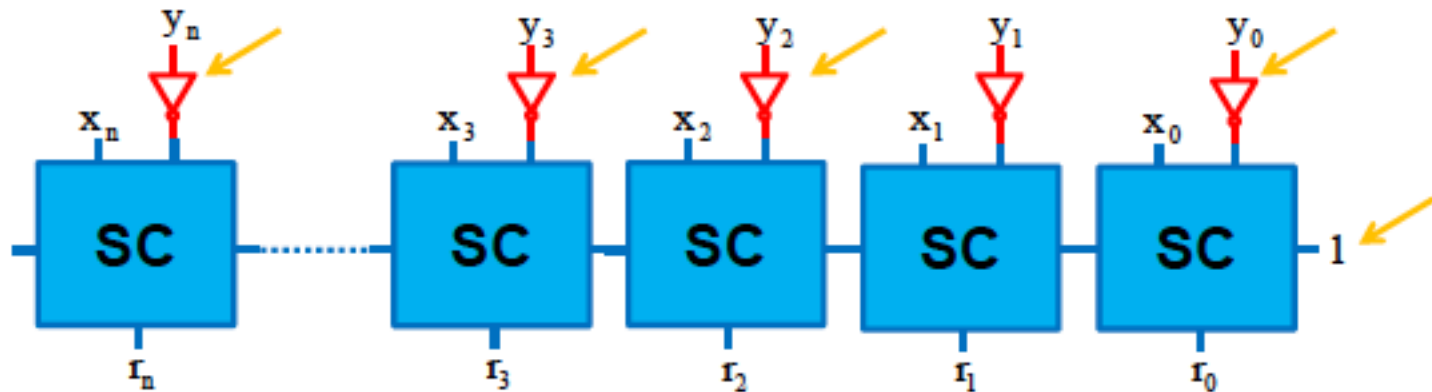


Somador/subtrator paralelo

- Somador:



- Subtrator:



Somador/Subtrator paralelo

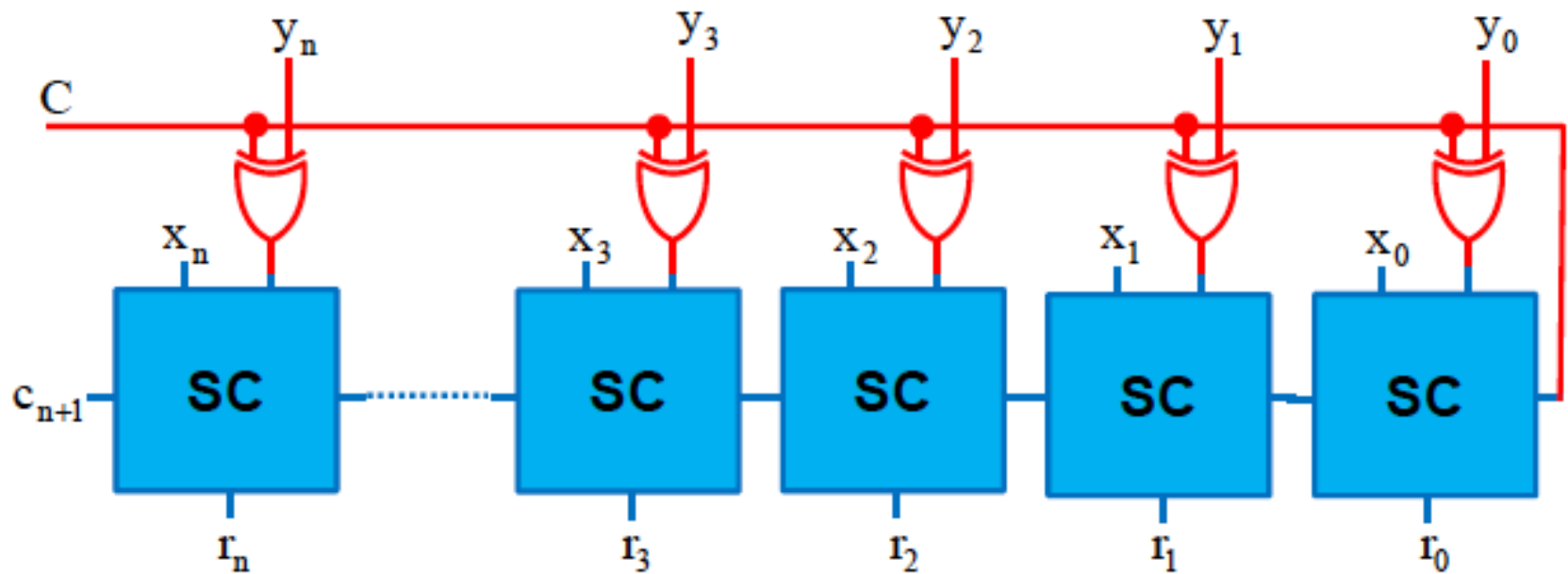
- Para obter o somador/subtrator, precisamos de um inversor que possa ser controlado (ativado ou desativado)

Controle	Entrada	Saída
0	0	0
0	1	1
1	0	1
1	1	0

$$S = \bar{A}B + A\bar{B} = A \oplus B$$

XOR funciona como um inversor controlado.

Somador/Subtrator paralelo



Se $C = 0 \rightarrow R = X + Y$

Se $C = 1 \rightarrow R = X - Y$

Overflow

- Ocorre quando o resultado de uma operação é maior (ou menor) do que o valor máximo (ou mínimo) que pode ser representado com um determinado número (fixo) de bits.

– Exemplo:

5 bits $\rightarrow 2^5$ valores diferentes

Em complemento de 2: de **-16 até 15**

No entanto,

$$10 + 7 = 17$$

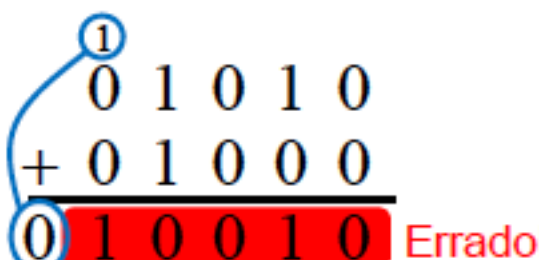
$$-11 - 9 = -20$$




– Em geral, o overflow precisa ser detectado e tratado

Overflow

- O overflow só ocorre se os dois operandos forem positivos ou negativos
- Exemplo (5 bits):

$$\begin{array}{r} 10 \\ + 8 \\ \hline 18 \end{array}$$


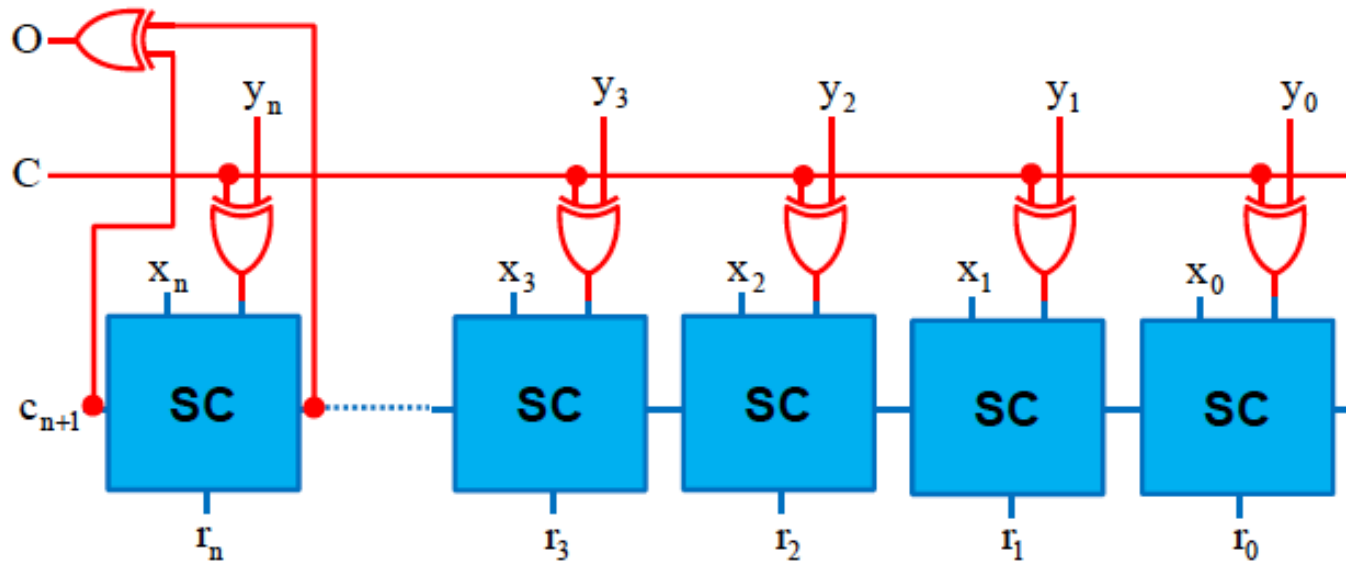
The diagram shows the 5-bit binary representation of 10 (01010) and 8 (01000). A blue circle with a '1' is above the first bit of 10, and a blue circle with a '0' is below the first bit of the sum. The resulting 5-bit sum is 10010, which is highlighted in red and labeled 'Errado!'.

$$\begin{array}{r} -9 \\ + -11 \\ \hline -20 \end{array}$$


The diagram shows the 5-bit binary representation of -9 (10111) and -11 (10101). A blue circle with a '0' is above the first bit of -9, and a blue circle with a '1' is below the first bit of the sum. The resulting 5-bit sum is 01100, which is highlighted in red and labeled 'Errado!'.

Overflow

- Circuito somador/subtrator com detecção de overflow



Se $C = 0 \rightarrow R = X + Y$

Se $C = 1 \rightarrow R = X - Y$

Se $O = 1 \rightarrow$ ocorreu overflow