Universidade Federal de Pelotas
Instituto de Física e Matemática
Departamento de Informática
Bacharelado em Ciência da Computação

Técnicas Digitais

Aula 14

4. Circuitos Combinacionais: Subtrator paralelo, somador/subtrator paralelo, implementação de somadores em tecnologia CMOS.

Prof. José Luís Güntzel

guntzel@ufpel.edu.br

www.ufpel.edu.br/~guntzel/TD/TD.html

4. Circuitos Combinacionais

Subtração Binária

Princípio Básico

$$A - B = A + (-B)$$

onde -B é o número B de sinal trocado

4. Circuitos Combinacionais

Subtração Binária

Princípio Básico

Trocar o sinal → equivale a → **Determinar o complemento de 2**

Então

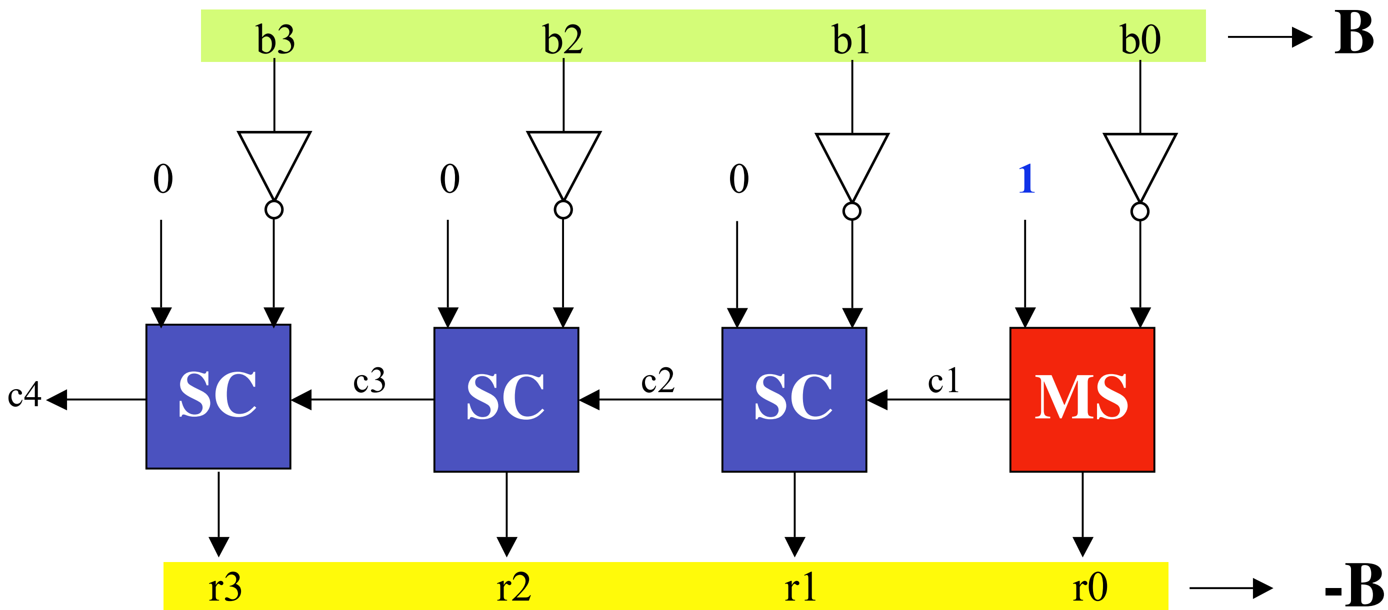
A - B = A + (- B) = A + (B em complemento de 2**)**

4. Circuitos Combinacionais

Subtração Binária

Como determinar o complemento de 2 de um número?

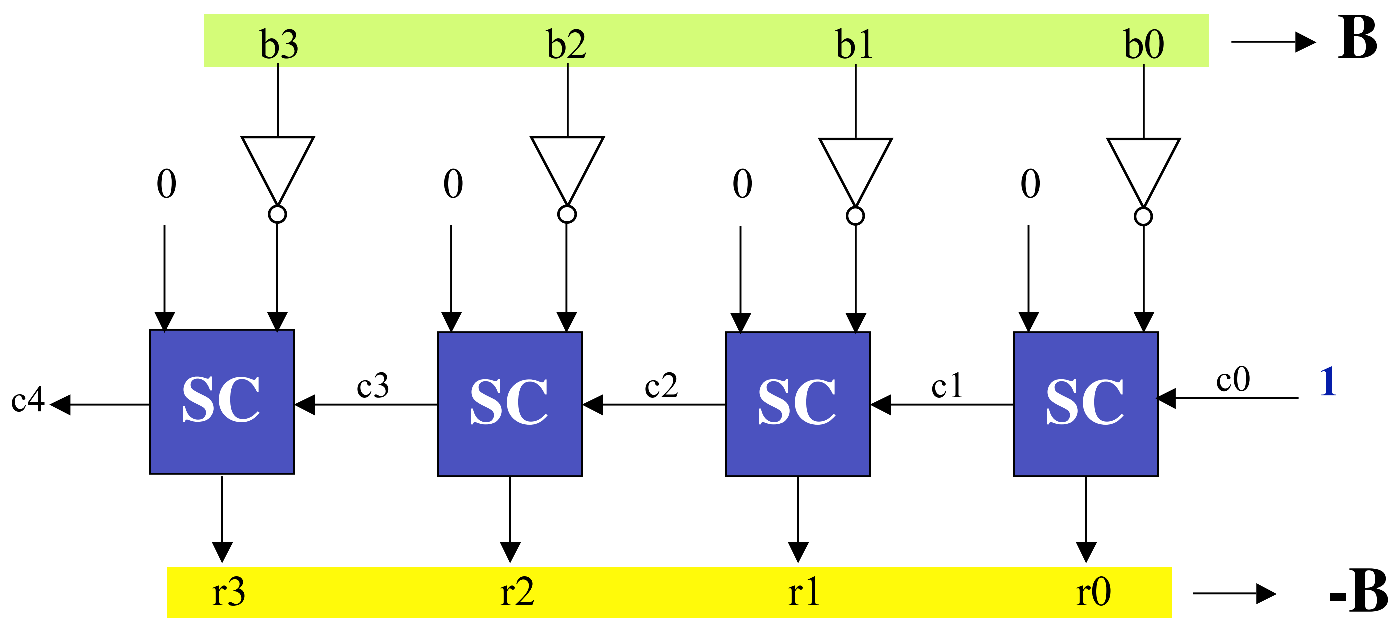
1. Toma-se a representação em sinal-magnitude
2. Inverte-se o número, bit a bit
3. Soma-se 1



4. Circuitos Combinacionais

Subtração Binária

Outra configuração de circuito...



4. Circuitos Combinacionais

Subtração Binária

Trocar o sinal → equivale a → **Determinar o complemento de 2**

Mas será que isso funciona se o número é negativo e queremos trocar seu sinal? Vejamos um exemplo…

1 0 0 1 = -7 (com 4 bits, complemento de 2)

Aplicando as regras do complemento de 2…

invertendo, bit a bit	0 1 1 0
somando 1	0 1 1 1 = +7 (com 4 bits)

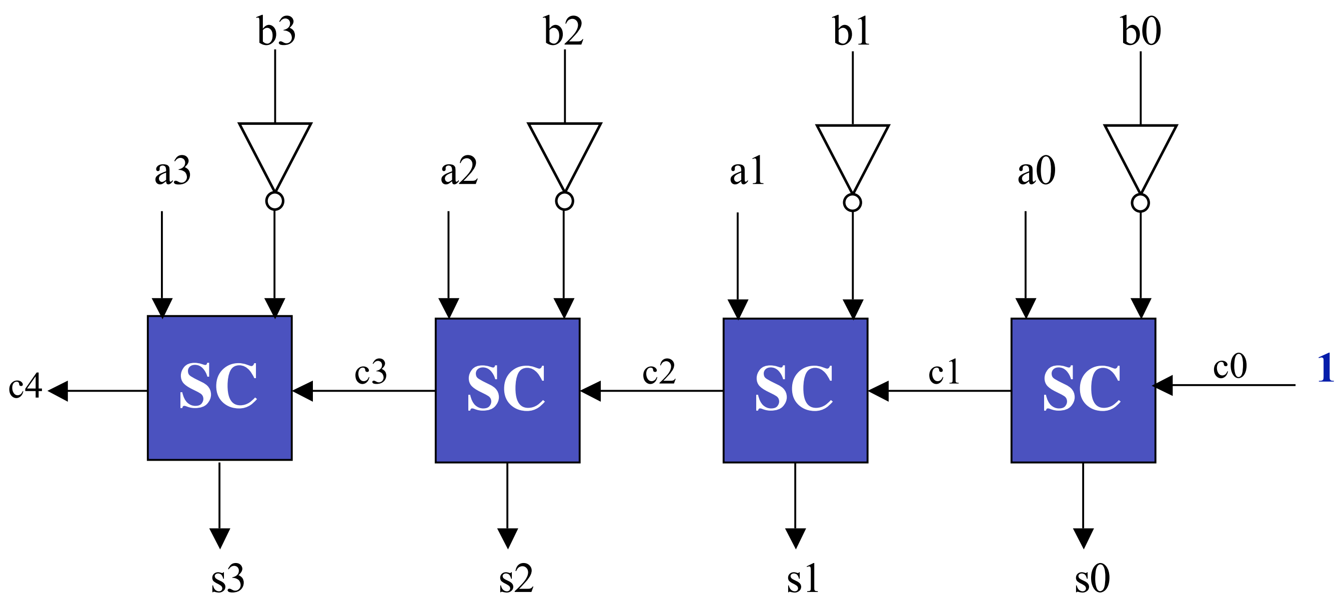
Funciona !!

4. Circuitos Combinacionais

Subtração Binária

Voltando à subtração:

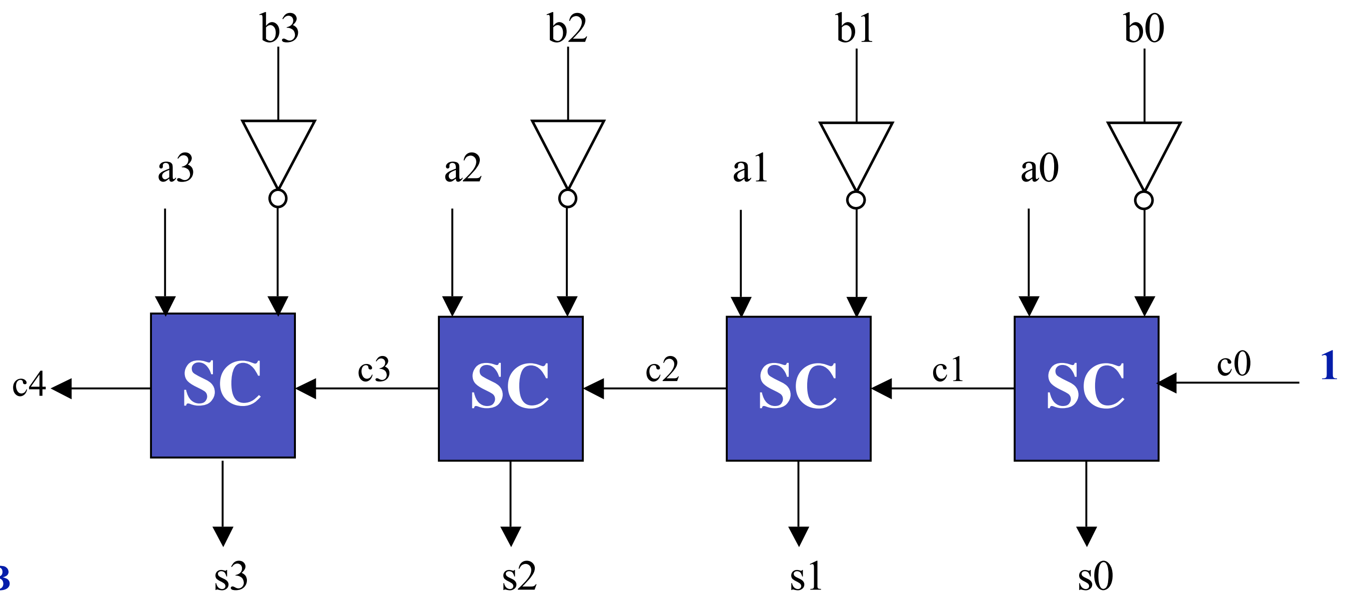
A - B = A + (B em complemento de 2**)**

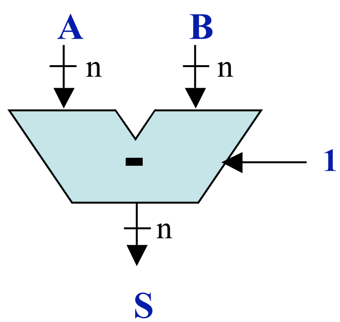


4. Circuitos Combinacionais

Subtrator Binário Paralelo (de 4 bits)

esquemático de blocos



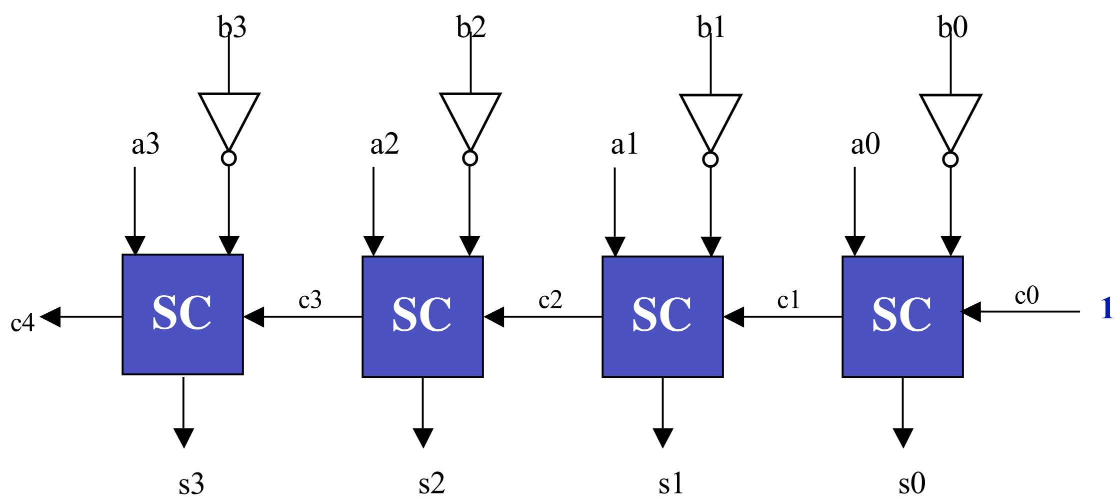


símbolo
(no nível RT)

4. Circuitos Combinacionais

Somador/Subtrator Paralelo

- **Seria possível modificar este circuito, de modo que ele possa ser "programado" para ser somador ou subtrator?**



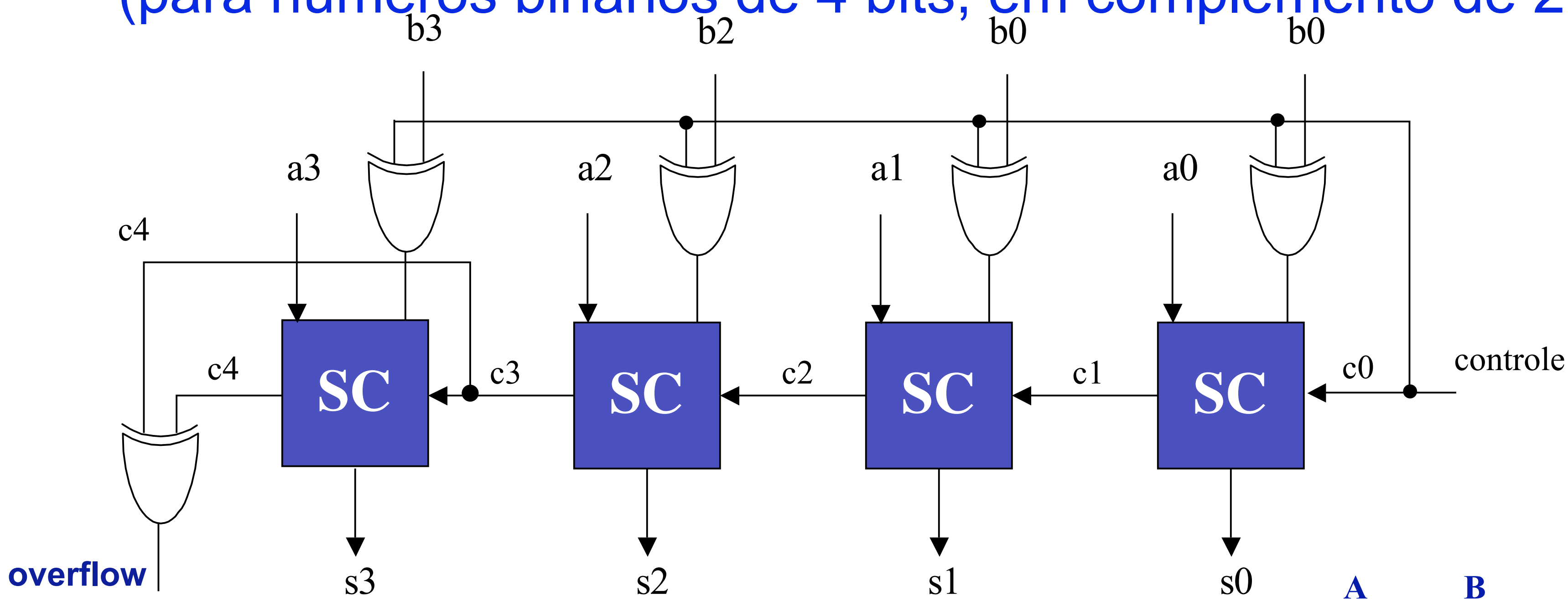
Resposta: Positivo! Modificações necessárias:

- **Substituir os inversores por "negadores controlados" (xors)**
- **Controlar o valor de c0 (0 para adição/1 para subtração)**

4. Circuitos Combinacionais

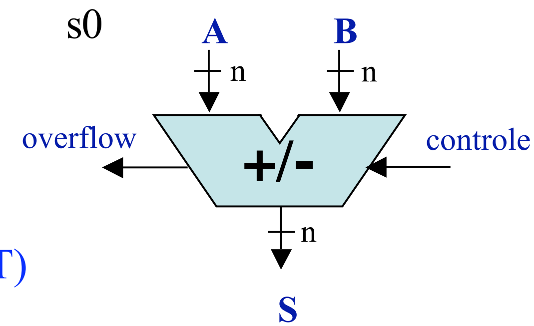
Somador/Subtrator Paralelo

(para números binários de 4 bits, em complemento de 2)



controle	operação
0	S=A+B
1	S=A-B

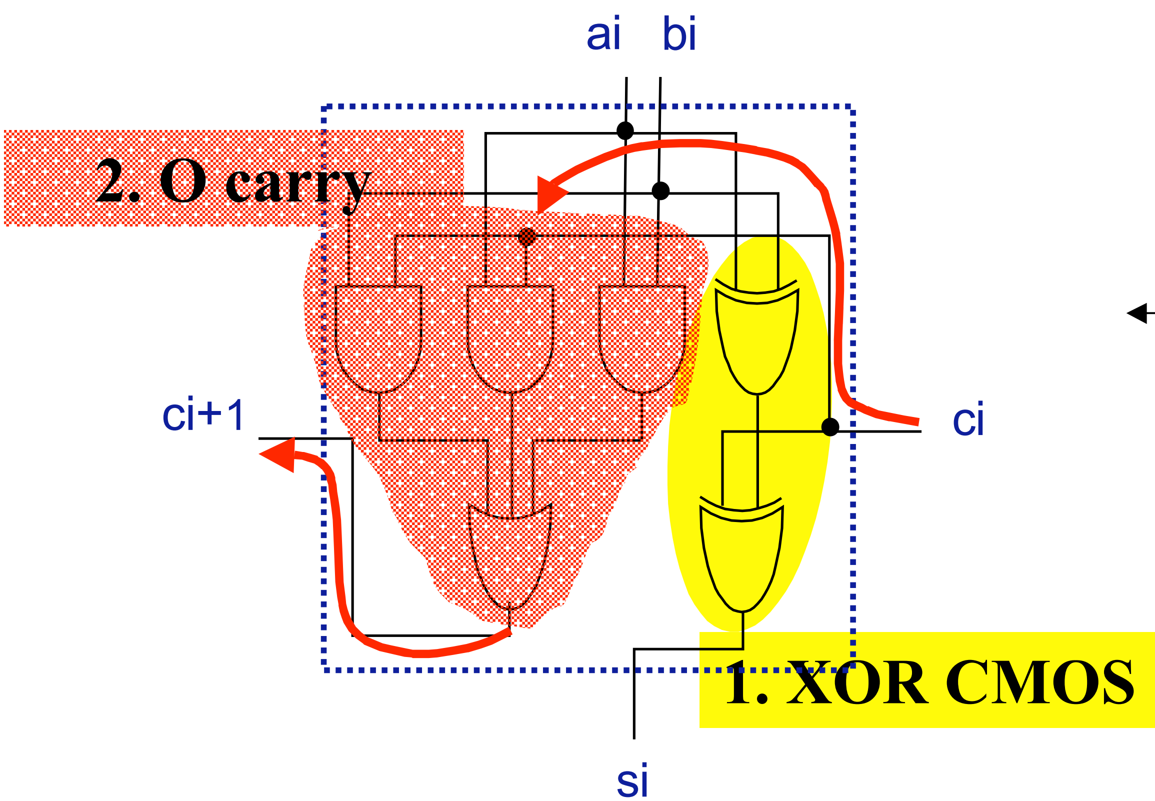
símbolo
(no nível RT)

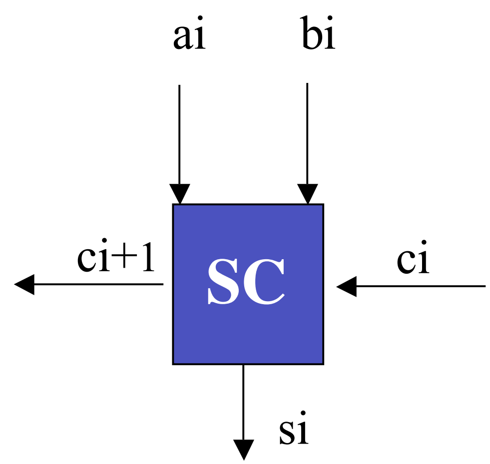


4. Circuitos Combinacionais

Implementação de Somadores

O Somador Completo (independente de tecnologia)





si = ci ⊕ ai ⊕ bi

ci+1 = ai·bi+ ai·ci+ bi·ci

4. Circuitos Combinacionais

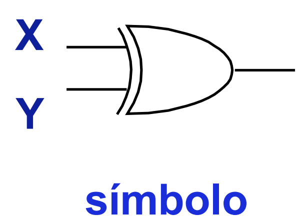
Implementação de Somadores

A Função OU Exlcusivo (XOR)

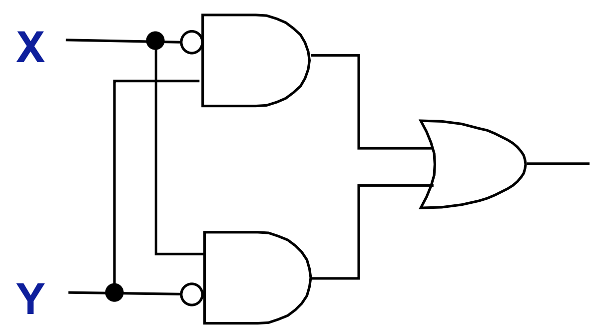
- A função XOR resulta 1 se um número ímpar de entradas valer 1
- Tem um papel importantíssimo na aritmética: implementa a soma (sem o transporte)

X	Y	X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

$$X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$$



≡



4. Circuitos Combinacionais

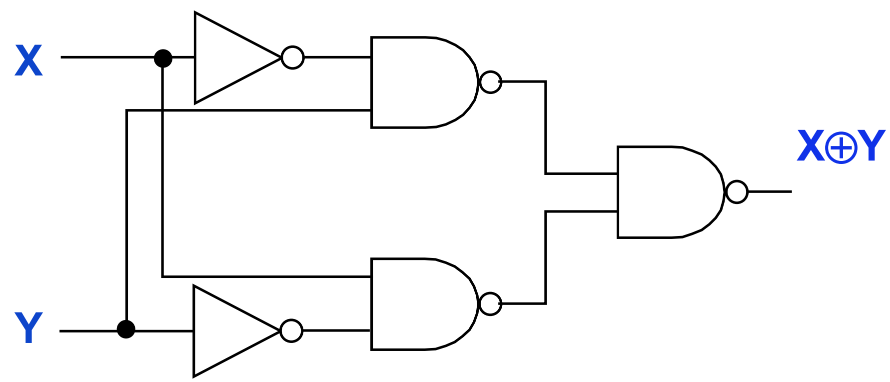
Implementação de Somadores

Topologias para Portas XOR - 1

- **Implementação como soma de produtos em tecnologia CMOS**

X	Y	X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

$$X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$$



- É uma implementação robusta
- Porém, gasta muitos transistores (16)

4. Circuitos Combinacionais

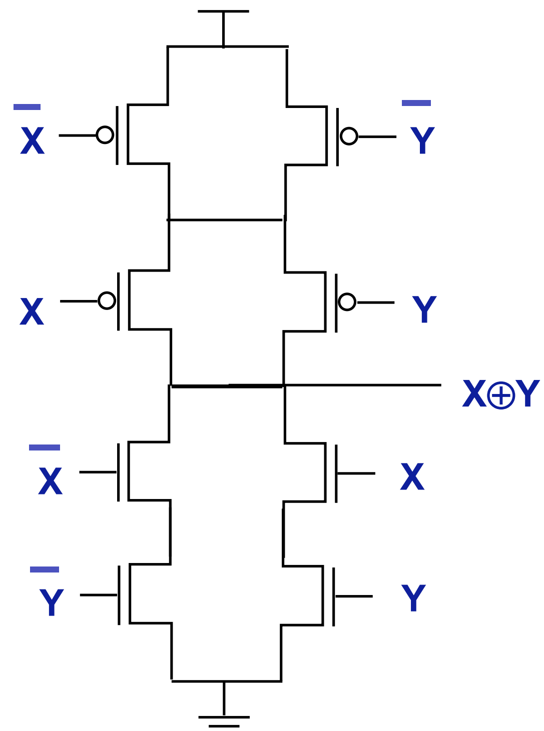
Implementação de Somadores

Topologias para Portas XOR - 2

- **Implementação com porta CMOS complexa**

X	Y	X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

$$X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$$



- Implementação robusta (menos que a anterior)
- Gasta menos transistores que a anterior (12, considerando os 2 inversores necessários para $\overline{X}$ e $\overline{Y}$)

4. Circuitos Combinacionais

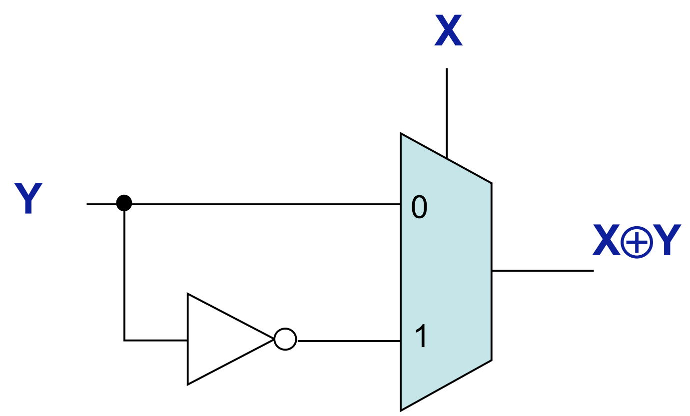
Implementação de Somadores

Topologias para Portas XOR - 3

- Implementação como um negador controlado (usando mux)

X	Y	X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

$X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$



4. Circuitos Combinacionais

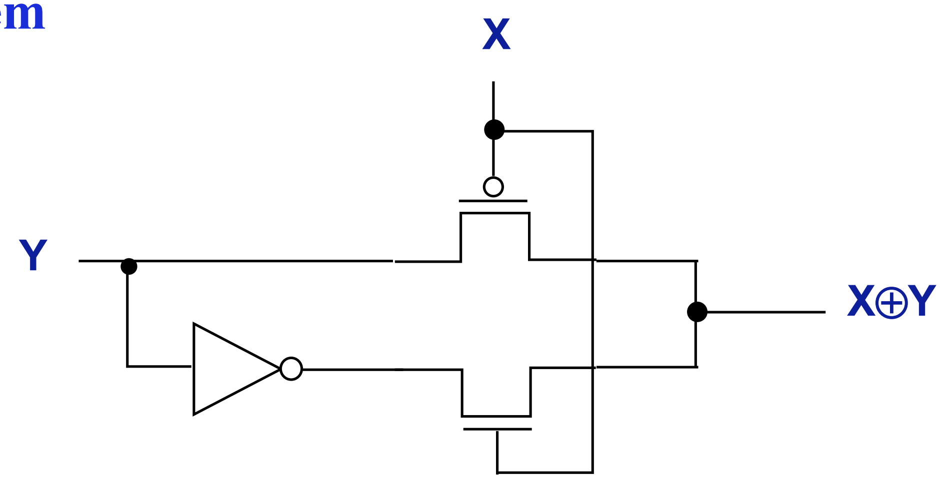
Implementação de Somadores

Topologias para Portas XOR - 3

- **Implementação como um negador controlado, versão com transistores de passagem**

X	Y	X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

$X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$



- Usa poucos transistores (apenas 4)
- Porém, tem problema de degradação de sinal

4. Circuitos Combinacionais

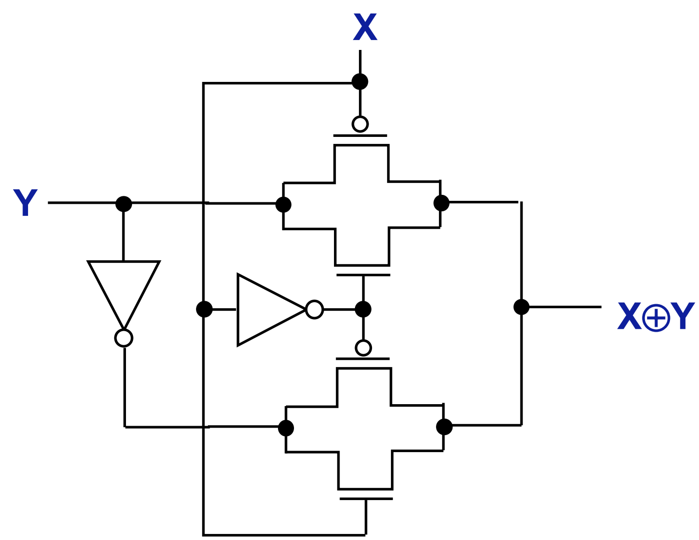
Implementação de Somadores

Topologias para Portas XOR - 3

- **Implementação como um negador controlado, versão com TGs**

X	Y	X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

$X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$



- Usa o dobro de transistores que a versão anterior, mas ainda assim poucos (8)
- Degrada menos o sinal: não usar em série…

4. Circuitos Combinacionais

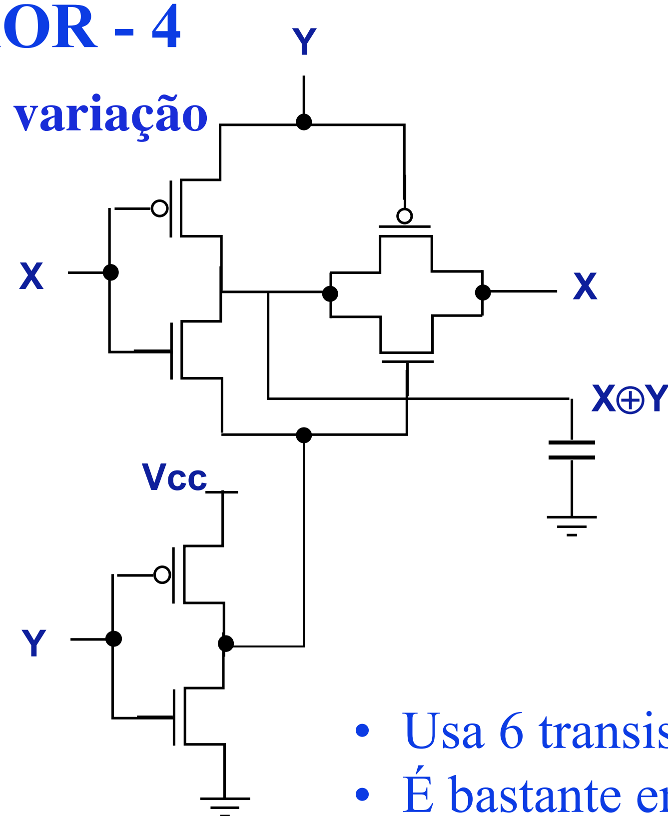
Implementação de Somadores

Topologias para Portas XOR - 4

- **Implementação não estática: variação das anterior**

X	Y	X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

$$X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$$

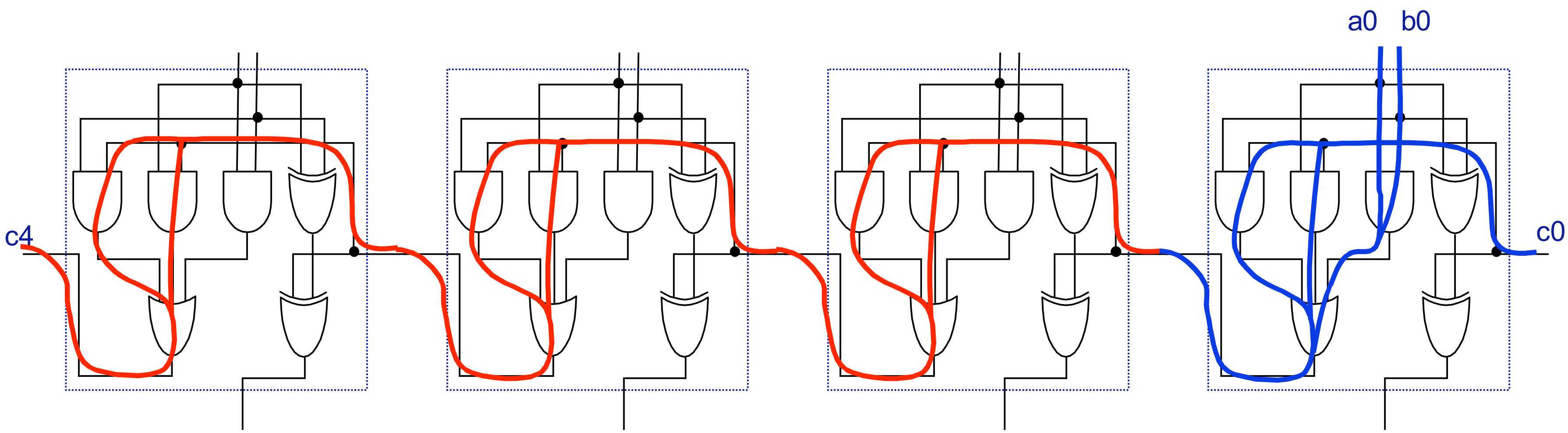


- Usa 6 transistores
- É bastante empregada…

4. Circuitos Combinacionais

Implementação de Somadores

O Problema da Propagação do Transporte (*Carry Propagation*)



Estimativa do atraso crítico do somador paralelo a partir de um dos SCs:

- Encontrar o caminho de maior atraso, que inicie por a0, b0, ou ci e termine em ci+1
- Encontrar o caminho de maior atraso, que inicie por ci e termine em ci+1

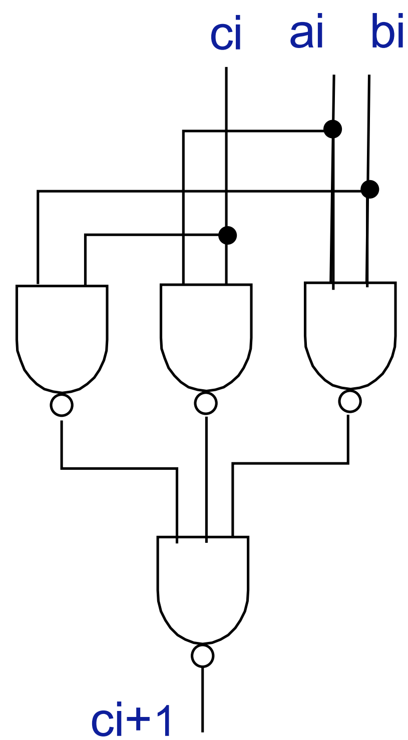
4. Circuitos Combinacionais

Implementação de Somadores

Topologias para o circuito de Carry - 1

- **Implementação como soma de produtos em tecnologia CMOS**

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



- É uma implementação robusta
- Porém, gasta muitos transistores (18)

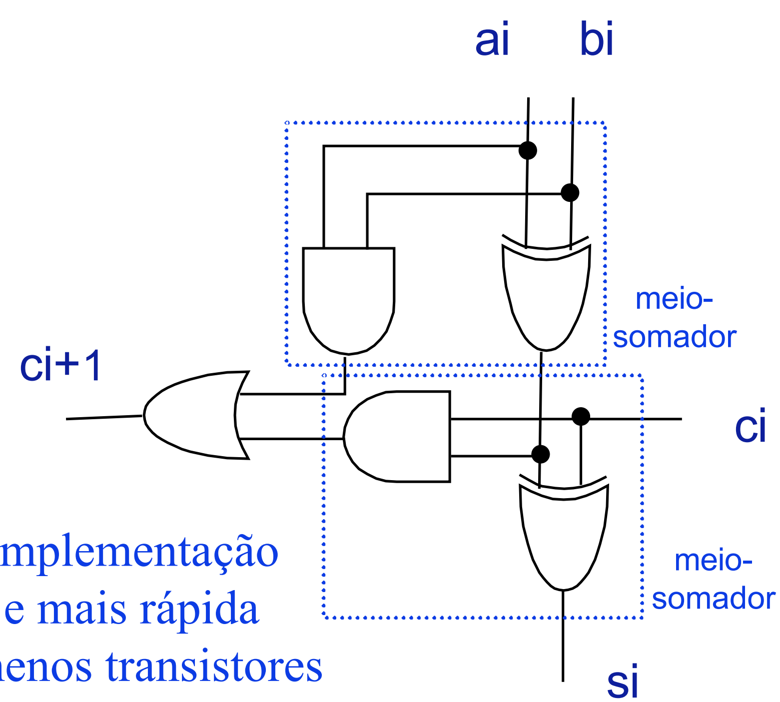
4. Circuitos Combinacionais

Implementação de Somadores

Topologias para o circuito de Carry - 2

- **Implementação derivada da composição de 2 meio-somadores**

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



- É uma implementação robusta e mais rápida
- E usa menos transistores que a anterior (12)

4. Circuitos Combinacionais

Implementação de Somadores

Observação: as funções para ci+1 e si são simétricas

- Conseqüência: qualquer que seja a ordem das variáveis de entrada, a ordem das saídas é sempre a mesma! Exemplos:

ci	ai	bi	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

$\equiv$

bi	ci	ai	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

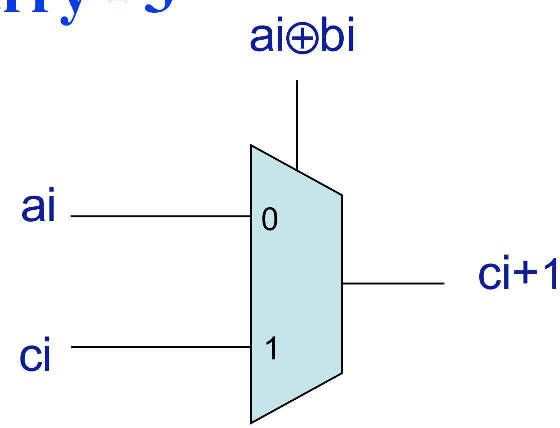
$\equiv$

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Implementação de Somadores

Topologias para o circuito de Carry - 3

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



- Tem como princípio uma interpretação da tabela-verdade, baseada em colocar a saída ci+1 em função de ai e do xor entre ai e bi.

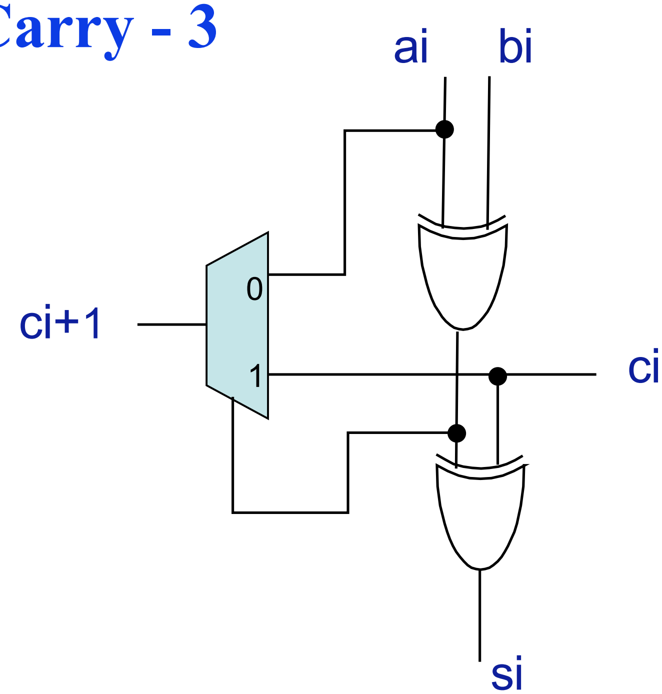
$$c_{i+1} = \overline{a_i \oplus b_i} \cdot a_i + a_i \oplus b_i \cdot c_i$$

4. Circuitos Combinacionais

Implementação de Somadores

Topologias para o circuito de Carry - 3

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



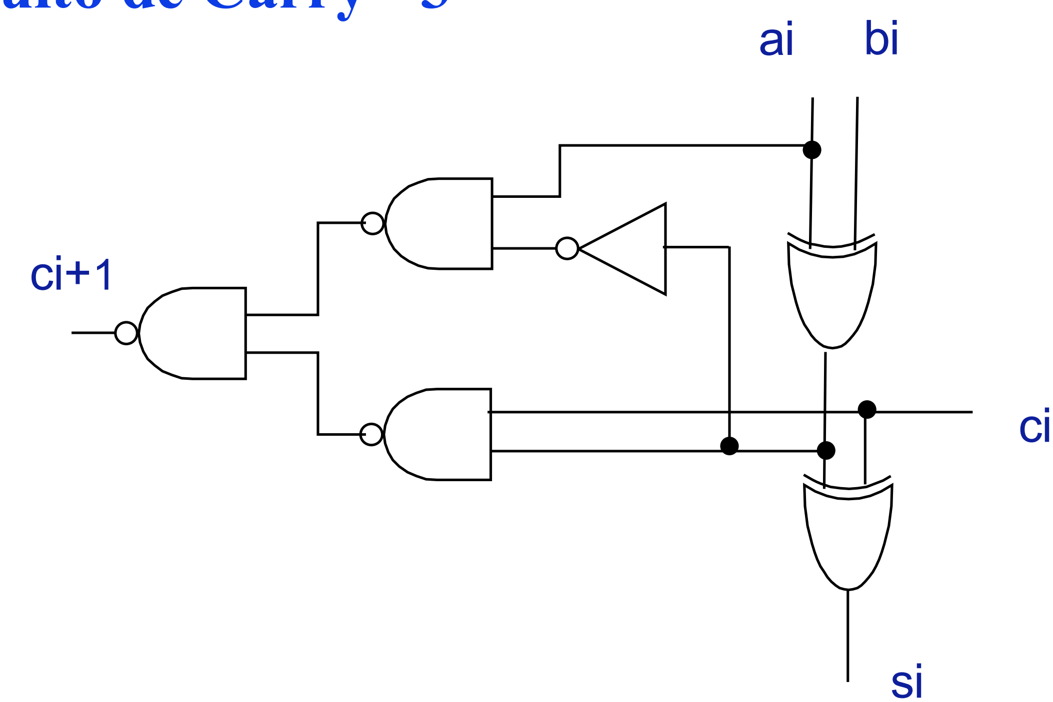
- Esta topologia dá origem a 3 outros circuitos, cada um com uma das versões de multiplexador estudadas. É o que veremos a seguir

4. Circuitos Combinacionais

Implementação de Somadores

Topologias para o circuito de Carry - 3

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



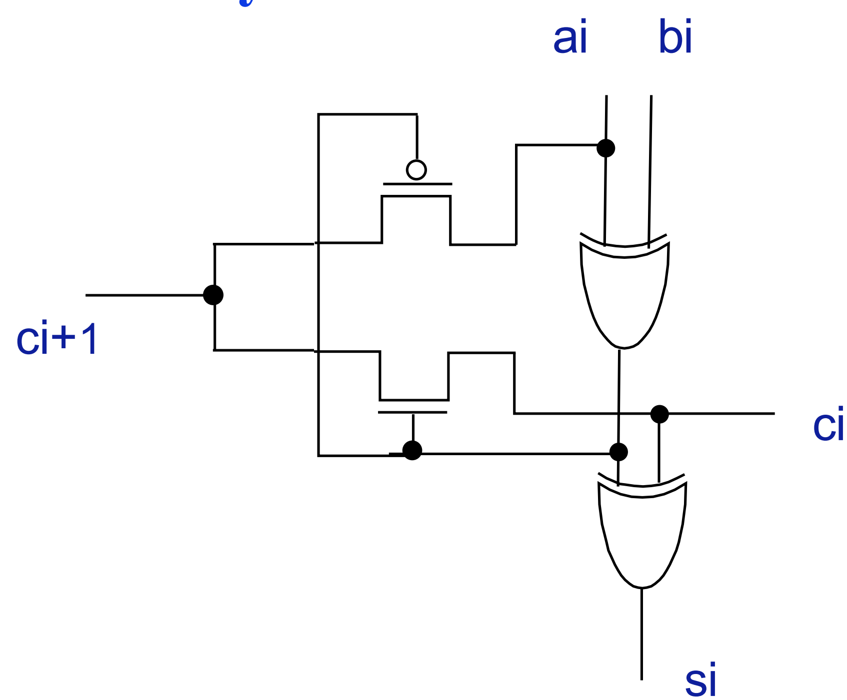
- Multiplexador com portas CMOS estáticas

4. Circuitos Combinacionais

Implementação de Somadores

Topologias para o circuito de Carry - 3

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



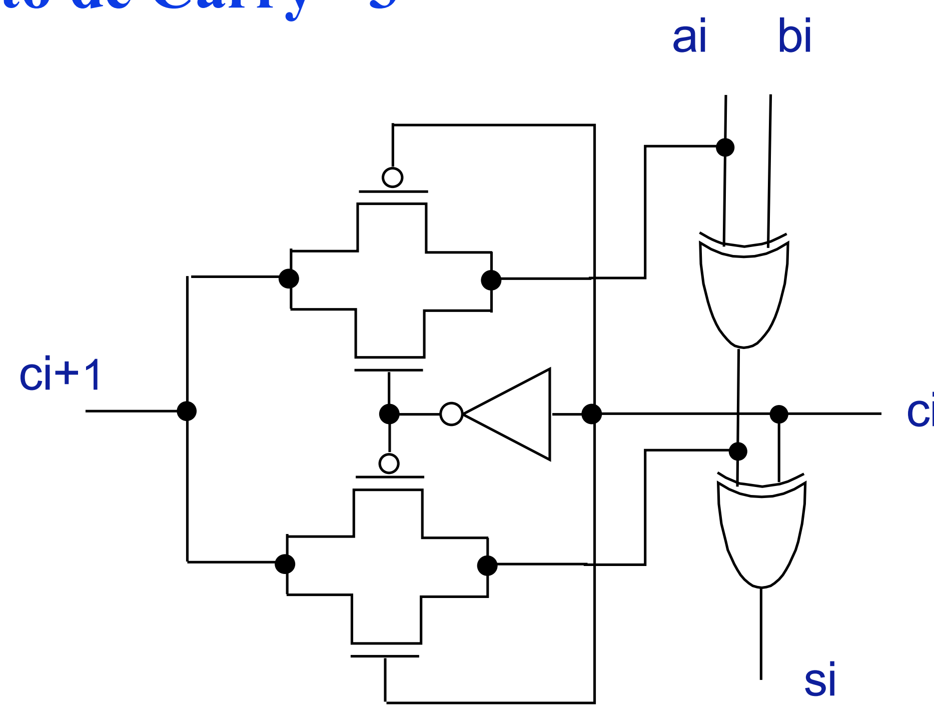
- Multiplexador com transistores de passagem

4. Circuitos Combinacionais

Implementação de Somadores

Topologias para o circuito de Carry - 3

ai	bi	ci	ci+1	si
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



- Multiplexador com *transmission gates*