CIRCUITOS DIGITAIS
CIRCUITOS COMBINACIONAIS BÁSICOS (SOMADORES & SUBTRATORES)

Marco A. Zanata Alves

Slides baseados nos slides de Rodrigo Hausen - CMCC – UFABC (2013)
http://compscinet.org/hausen/courses/circuitos/

AULA PASSADA: FUNÇÕES LÓGICAS

Tabela verdade conjunção (e)

X	Y	$X \cdot Y$
V	V	**V**
V	F	**F**
F	V	**F**
F	F	**F**

Tabela verdade disjunção (ou)

X	Y	$X + Y$
V	V	**V**
V	F	**V**
F	V	**V**
F	F	**F**

Tabela verdade disjunção exclusiva (ou-ex)

X	Y	$X + Y$
V	V	**F**
V	F	**V**
F	V	**V**
F	F	**F**

Tabela verdade negação (não)

X	$\overline{X}$
V	**F**
F	**V**

Conjunção (e): resultado verdadeiro apenas se X e Y forem verdadeiros.

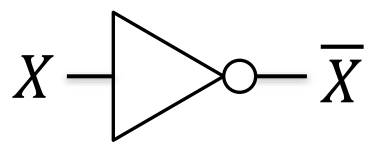
Disjunção (ou): resultado verdadeiro apenas se X ou Y forem verdadeiros.

Disjunção Exclusiva (ou-ex): resultado verdadeiro apenas se um for verdadeiro.

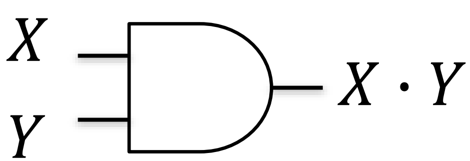
Negação (não): resultado só será verdadeiro se X não for verdadeiro.

AULA PASSADA: PORTAS LÓGICAS

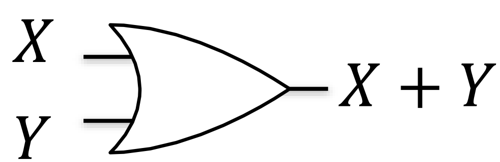
Trata-se de circuitos que efetuam operações básicas da álgebra booleana



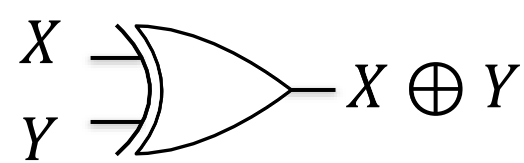
Porta **not**



Porta **and**



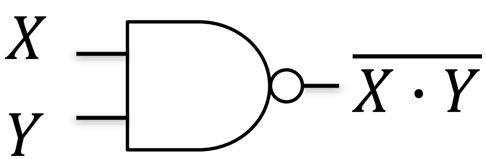
Porta **or**



Porta **xor**

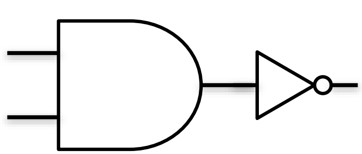
AULA PASSADA: PORTAS LÓGICAS COM SAÍDAS INVERTIDAS

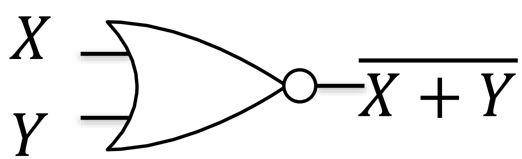
Também existem as seguintes portas com saída invertida (negada)



Porta **nand**

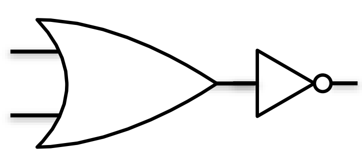
=

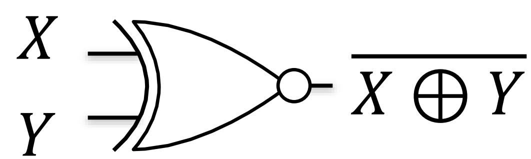




Porta **nor**

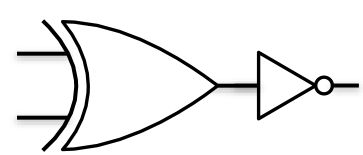
=





Porta **xnor**

=



BLOCOS DIGITAIS BÁSICOS

Os blocos mais elementares da eletrônica digital são as portas lógicas.

Daqui em diante, vamos aplicar o nosso conhecimento de **análise** e **síntese de circuitos** digitais para construir alguns blocos um pouco menos elementares:

- somadores e subtratores
- codificadores e decodificadores
- multiplexadores e demultiplexadores
- unidades lógico-aritméticas
- latches e flip-flops
- registradores e memórias

É extremamente útil saber a função de cada um desses blocos e as suas **interfaces** (ou seja, como conectar cada um deles em nossos circuitos).

HALF ADDER
MEIO SOMADOR

BLOCOS SOMADORES BINÁRIOS

Exemplo 1: Elabore um circuito digital com 2 entradas, a_0 e b_0, e 2 saídas, s_1 e s_0 de tal forma que $(s_1 s_0)_2$ represente a soma aritmética $a_0 + b_0$.

$$\begin{array}{r} a_0 \\ +\, b_0 \\ \hline s_1 s_0 \end{array}$$

Quais os passos para obter um circuito digital combinacional?

BLOCOS SOMADORES BINÁRIOS

Exemplo 1: Elabore um circuito digital com 2 entradas, a_0 e b_0, e 2 saídas, s_1 e s_0 de tal forma que $(s_1 s_0)_2$ represente a soma aritmética $a_0 + b_0$.

$$\begin{array}{r} a_0 \\ +\, b_0 \\ \hline s_1 s_0 \end{array}$$

Primeiro passo: obtenha e simplifique a expressão lógica para cada saída.

BLOCOS SOMADORES BINÁRIOS

Exemplo 1: Elabore um circuito digital com 2 entradas, a_0 e b_0, e 2 saídas, s_1 e s_0 de tal forma que $(s_1 s_0)_2$ represente a soma aritmética $a_0 + b_0$.

$$\begin{array}{r} a_0 \\ +\,b_0 \\ \hline s_1 s_0 \end{array}$$

Primeiro passo: obtenha e simplifique a expressão lógica para cada saída.

a_0	b_0	s_0
0	0	0
0	1	1
1	0	1
1	1	0

a_0	b_0	s_1
0	0	0
0	1	0
1	0	0
1	1	1

Neste caso, é elementar obter expressões simples para as saídas:

$s_0 = a_0 \oplus b_0$

$s_1 = a_0 b_0$

BLOCOS SOMADORES BINÁRIOS

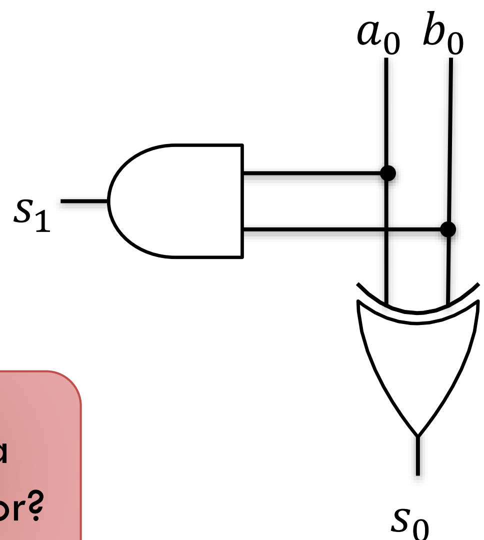
Exemplo 1: Elabore um circuito digital com 2 entradas, a_0 e b_0, e 2 saídas, s_1 e s_0 de tal forma que $(s_1 s_0)_2$ represente a soma aritmética $a_0 + b_0$.

$$\begin{array}{r} a_0 \\ +\, b_0 \\ \hline s_1 s_0 \end{array}$$

Segundo passo: desenhar o diagrama do circuito.

$$s_0 = a_0 \oplus b_0$$

$$s_1 = a_0 b_0$$



Para que serve a saída S1?

O que falta nesse somador?

BLOCOS SOMADORES BINÁRIOS

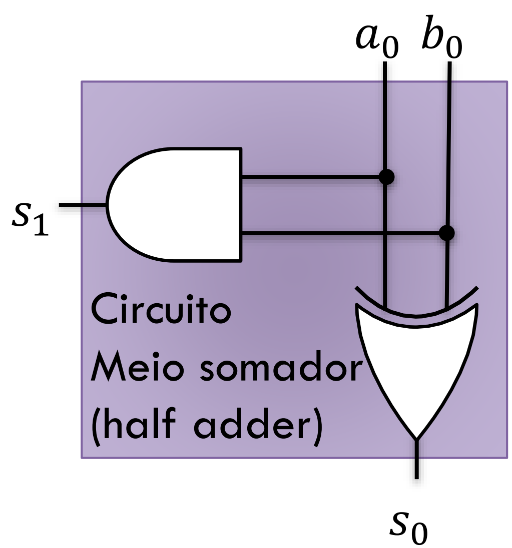
Exemplo 1: Elabore um circuito digital com 2 entradas, a_0 e b_0, e 2 saídas, s_1 e s_0 de tal forma que $(s_1 s_0)_2$ represente a soma aritmética $a_0 + b_0$.

$$\begin{array}{r} a_0 \\ +\, b_0 \\ \hline s_1 s_0 \end{array}$$

Segundo passo: desenhar o diagrama do circuito.

$s_0 = a_0 \oplus b_0$

$s_1 = a_0 b_0$



BLOCOS SOMADORES BINÁRIOS

Exemplo 1: Elabore um circuito digital com 2 entradas, a_0 e b_0, e 2 saídas, s_1 e s_0 de tal forma que $(s_1 s_0)_2$ represente a soma aritmética $a_0 + b_0$.

$$\begin{array}{r} a_0 \\ + \; b_0 \\ \hline s_1 s_0 \end{array}$$

Terceiro passo: analisar o circuito e verificar as saídas.

Quarto passo: monte o circuito e faça sua tabela verdade.

(para este circuito, os dois últimos passos não tem a menor graça)

FULL ADDER
SOMADOR COMPLETO

BLOCOS SOMADORES BINÁRIOS

Exemplo 2: Elabore um circuito digital com 3 entradas, a_i , b_i , c_{i-1} e 2 saídas, s_i e c_i de tal forma que si represente a soma aritmética $a_i + b_i + c_{i-1}$ e c_i represente o vai-um (carry) da operação.

Quais as entradas?

Quais as saídas?

$$
\begin{array}{lllll}
\ldots c_i & c_{i-1} & \ldots c_0 & \\
\ldots a_{i+1} & a_i & \ldots a_1 & a_0 \\
\ldots b_{i+1} & b_i & \ldots b_1 & b_0 \\
\hline
\ldots s_{i+1} & s_i & \ldots s_1 & s_0
\end{array}
$$

BLOCOS SOMADORES BINÁRIOS

Primeiro passo: obter as expressões para as saídas s_i e c_i

Segundo passo: diagrama do circuito digital.

$$
\begin{array}{cccccc}
\dots & c_i & c_{i-1} & \dots & c_0 & \\
\dots & a_{i+1} & a_i & \dots & a_1 & a_0 \\
\dots & b_{i+1} & b_i & \dots & b_1 & b_0 \\
\hline
\dots & s_{i+1} & s_i & \dots & s_1 & s_0
\end{array}
$$

BLOCOS SOMADORES BINÁRIOS

Primeiro passo: obter as expressões para as saídas s_i e c_i .

s_i

c_{i-1} \ $a_i b_i$	00	01	11	10
0	0	1	0	1
1	1	0	1	0

c_i

c_{i-1} \ $a_i b_i$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$
\begin{array}{rrrr}
\ldots\ c_i & c_{i-1} & \ldots\ c_0 & \\
\ldots\ a_{i+1} & a_i & \ldots\ a_1 & a_0 \\
\ldots\ b_{i+1} & b_i & \ldots\ b_1 & b_0 \\
\hline
\ldots\ s_{i+1} & s_i & \ldots\ s_1 & s_0
\end{array}
$$

BLOCOS SOMADORES BINÁRIOS

Primeiro passo: obter as expressões para as saídas s_i e c_i .

s_i

c_{i-1} \ $a_i b_i$	00	01	11	10
0	0	1	0	1
1	1	0	1	0

Note que s_i só é 1 se apenas um dos bits a_i , b_i , c_{i-1} é 1, ou se os três forem 1. Isto corresponde à expressão:

$$s_i = a_i \oplus b_i \oplus c_{i-1}$$

c_i

c_{i-1} \ $a_i b_i$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

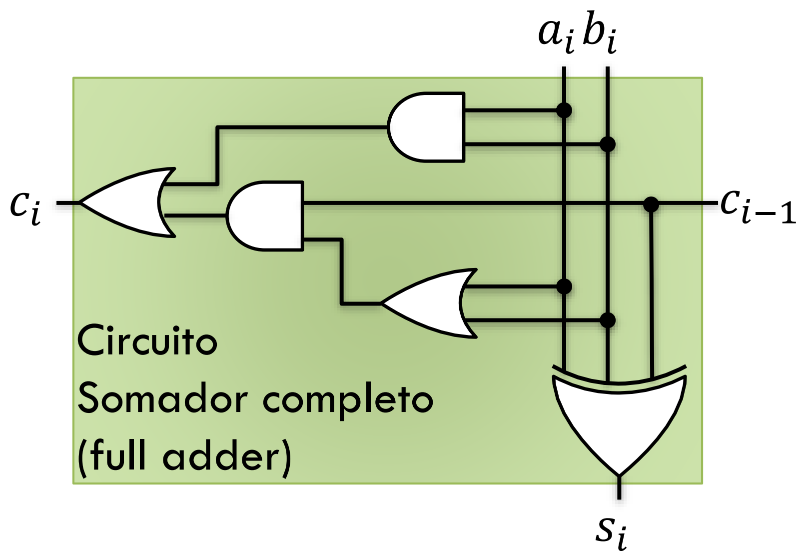
$$c_i = a_i b_i + a_i c_{i-1} + b_i c_{i-1}$$
$$= a_i b_i + (a_i + b_i) \cdot c_{i-1}$$

BLOCOS SOMADORES BINÁRIOS

Segundo passo: diagrama do circuito digital.

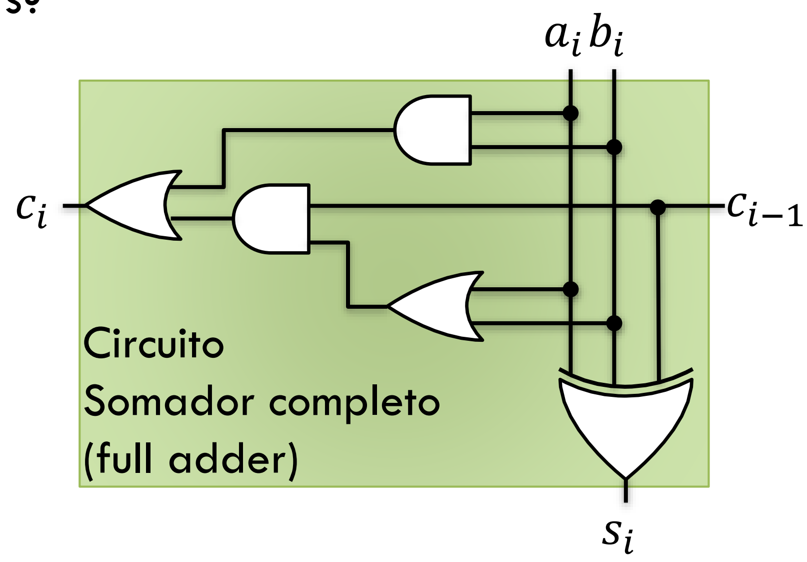
$$s_i = a_i \oplus b_i \oplus c_{i-1}$$

$$c_i = a_i b_i + (a_i + b_i) \cdot c_{i-1}$$



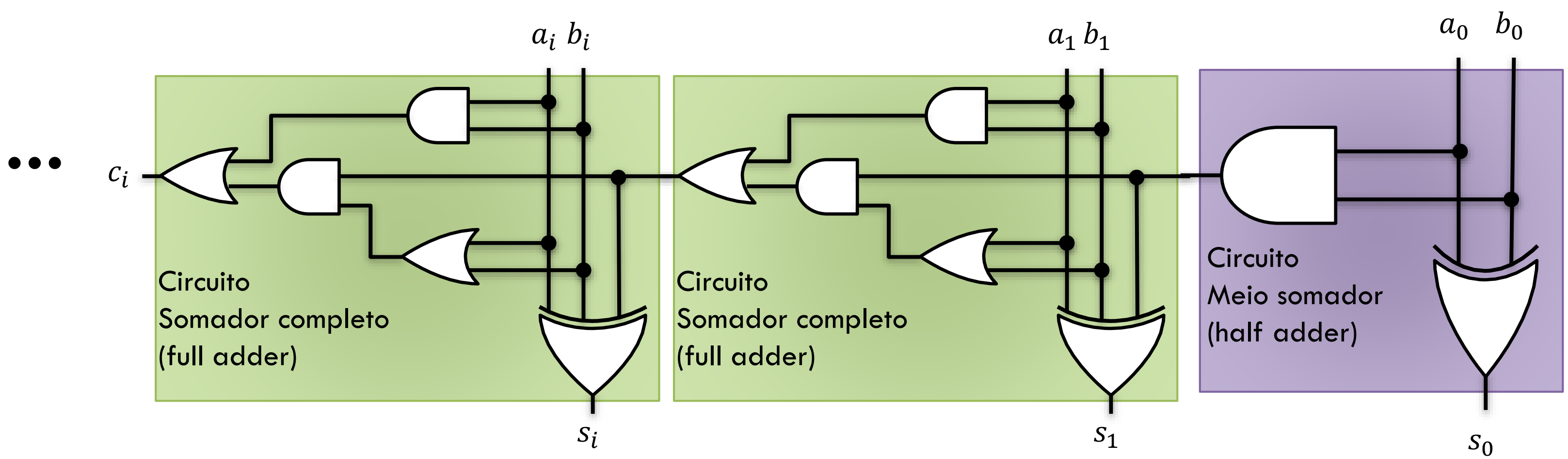
BLOCOS SOMADORES BINÁRIOS

Como podemos fazer um somador de 4 bits?



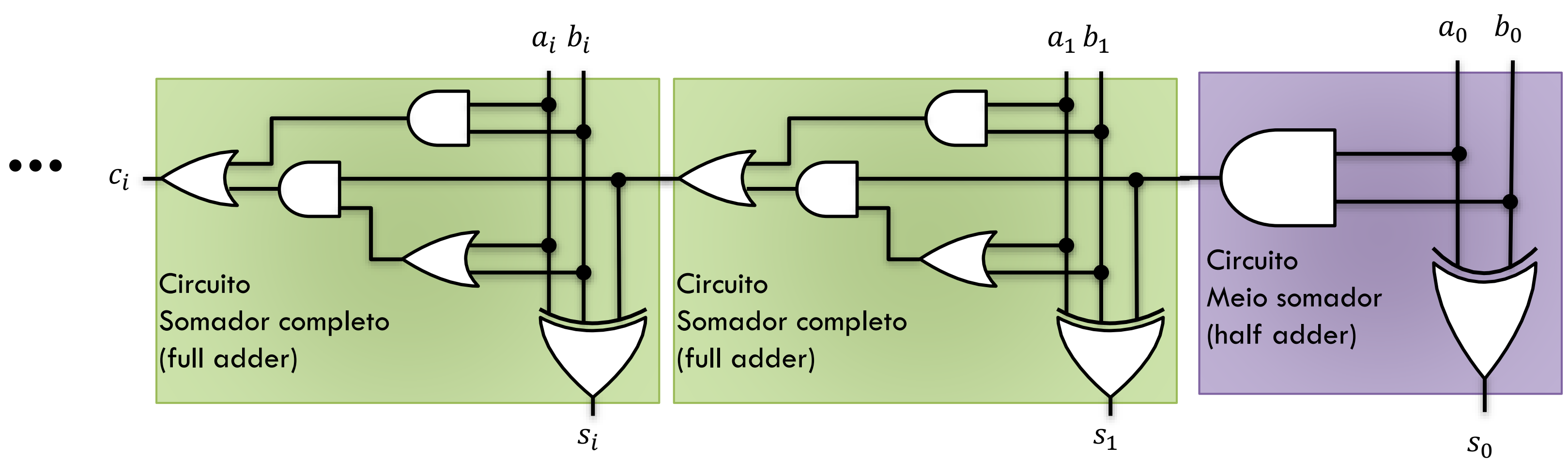
BLOCOS SOMADORES BINÁRIOS

Note que, juntando blocos Half Adder e Full Adder, podemos montar um somador para números de n bits.



BLOCOS SOMADORES BINÁRIOS

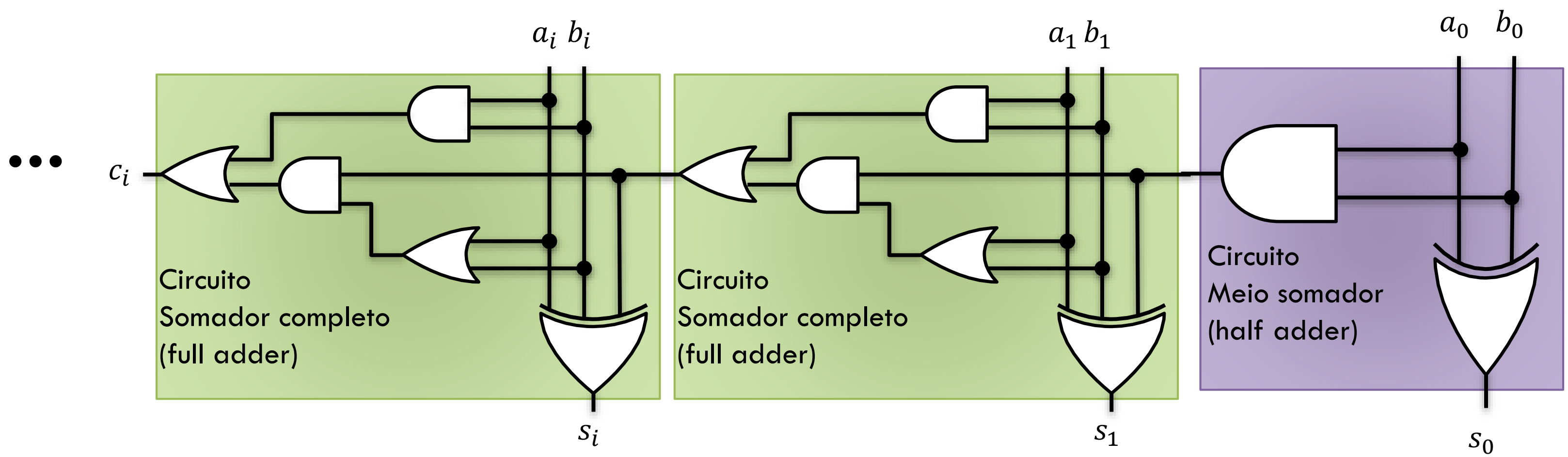
Somador ripple carry de n bits: leva este nome pois os vai-uns (carry) são propagados como uma ondulação (ripple) da direita para a esquerda.



BLOCOS SOMADORES BINÁRIOS

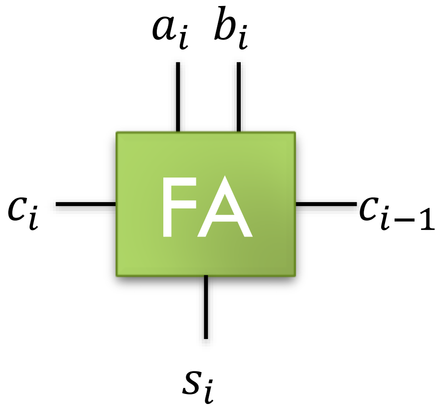
Somador ripple carry de n bits: leva este nome pois os vai-uns (carry) são propagados como uma ondulação (ripple) da direita para a esquerda.

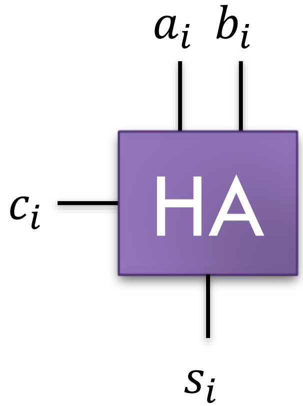
Quantas e quais são as portas lógicas usadas em um somador do tipo **ripple carry (propagador de vai-um)** de n bits?



BLOCOS SOMADORES BINÁRIOS

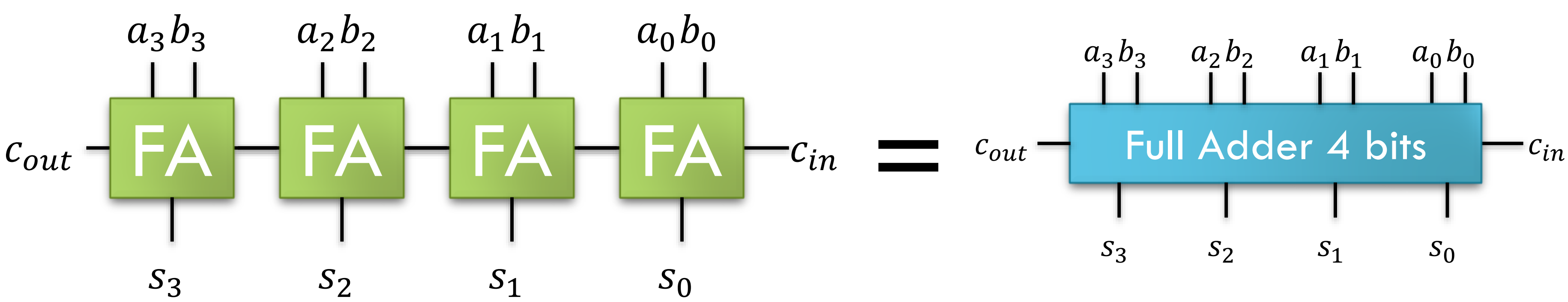
Podemos enxergar os blocos somadores (half adder e full adder) como componentes fechados.





BLOCOS SOMADORES BINÁRIOS

Também podemos combinar esses componente para criar novos componentes.



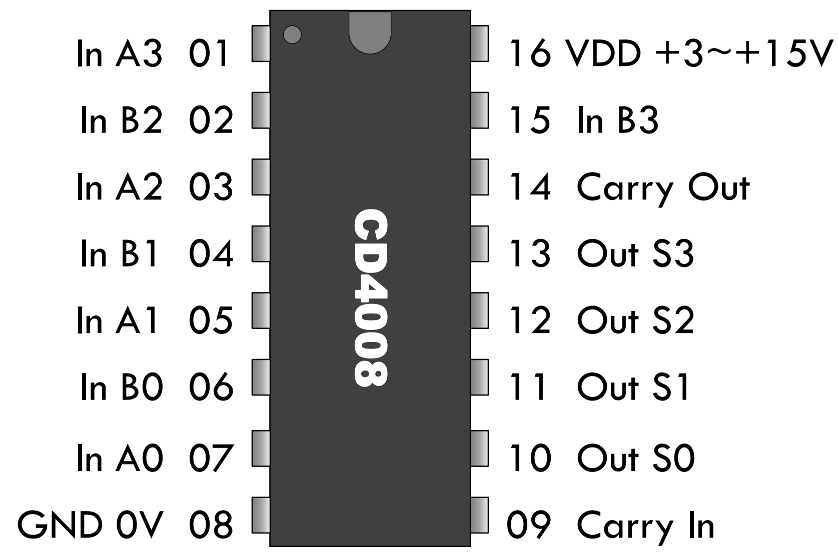
BLOCOS SOMADORES BINÁRIOS

Podemos comprar blocos somadores integrados:

- 7483 (TTL);
- CD4008 (CMOS);
- E outros.

Podemos unir blocos somadores completos para obter somadores com quantidade maior de bits

Por ex. juntar 8 integrados CD4008 para fazer um somador de 32 bits



SUBTRATOR

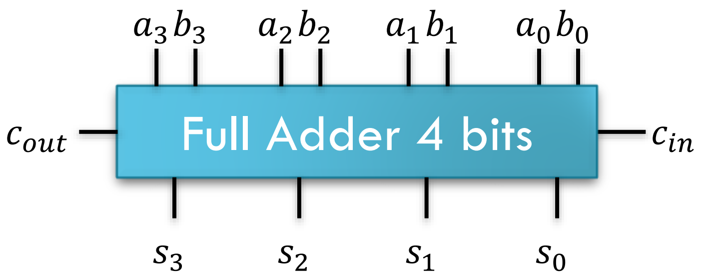
BLOCO SUBTRATOR BINÁRIO

Para subtração iremos usar o complemento de 2 e um somador.

$A - B = A + (\overline{B} + 1)$, desprezando o último "vai-um"

SUBTRAÇÃO DE NÚMEROS COM SINAL

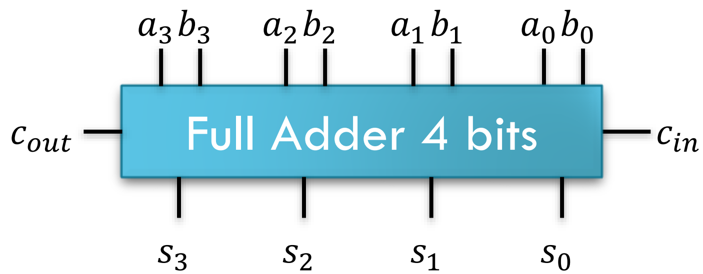
Este é um somador para palavras de n bits que representam números inteiros sem sinal.



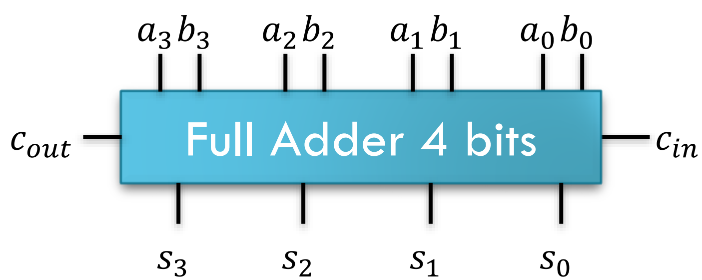
Como fazer um somador para palavras de n bits que representam números inteiros com sinal no formato complemento de 2?

SUBTRAÇÃO DE NÚMEROS COM SINAL

Este é um somador para palavras de n bits que representam números inteiros sem sinal.



Como fazer um somador para palavras de n bits que representam números inteiros com sinal no formato complemento de 2?



Não há nenhuma diferença no circuito!

SUBTRAÇÃO DE NÚMEROS COM SINAL

Sabendo que: $a - b = a + (-b)$, então podemos utilizar um somador completo de n bits.

Precisamos garantir que o número a ser subtraído (b) esteja em complemento de 2.

Faça o diagrama de um circuito digital para um subtrator de n bits.
Você só precisará de:

- Somador completo de n bits
- Portas NOT

SUBTRAÇÃO DE NÚMEROS COM SINAL

Sabendo que: $a - b = a + (-b)$, então podemos utilizar um somador completo de n bits.

Precisamos garantir que o número a ser subtraído (b) esteja em complemento de 2.

$$(-b) = \overline{b} + 1$$

$$-(b_n b_{n-1} \dots b_2 b_1 b_0) = \overline{b_n b_{n-1} \dots b_2 b_1 b_0} + 1$$

Como fazer?

SUBTRAÇÃO DE NÚMEROS COM SINAL

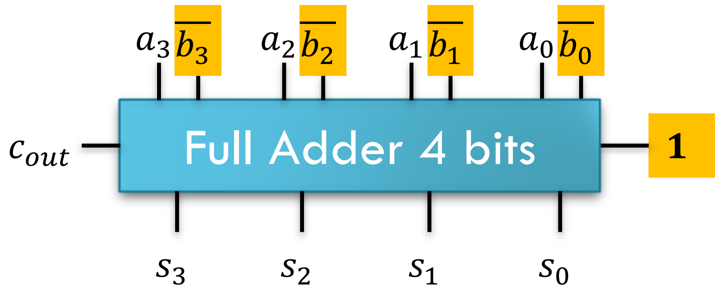
Sabendo que: $a - b = a + (-b)$, então podemos utilizar um somador completo de n bits.

Precisamos garantir que o número a ser subtraído (b) esteja em complemento de 2.

$$(-b) = \overline{b} + 1$$

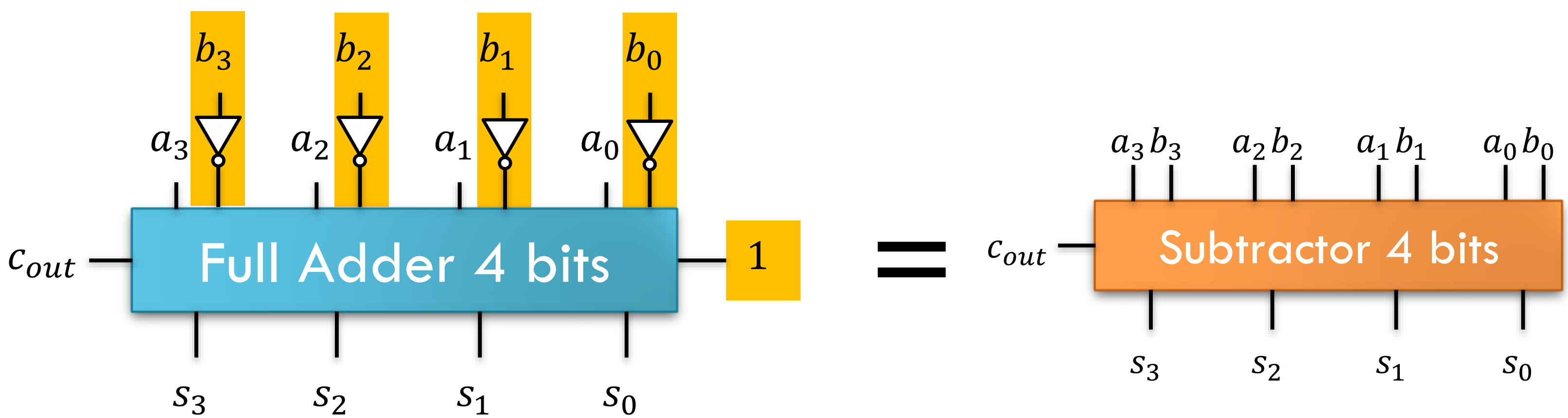
$$-(b_n b_{n-1} \dots b_2 b_1 b_0) = \overline{b_n b_{n-1} \dots b_2 b_1 b_0} + 1$$

Utilizamos o b invertido, e ativando a entrada c_{in} para somar 1.



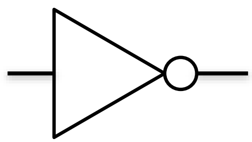
SUBTRAÇÃO DE NÚMEROS COM SINAL

Invertendo o b e ativando a entrada c_{in} para somar 1 temos nosso subtrator.

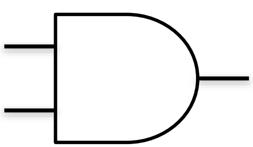


EXERCÍCIO

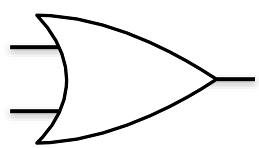
Elabore um Somador completo usando apenas meio somadores e portas lógicas básicas.



Porta **not**



Porta **and**



Porta **or**