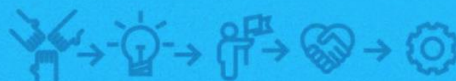




#SOMOS
MAIS
UNIFG



O Meio-Somador

Lembre-se das regras básicas para a adição binária

$$0 + 0 = 0$$

$$0 + 1 = 1$$

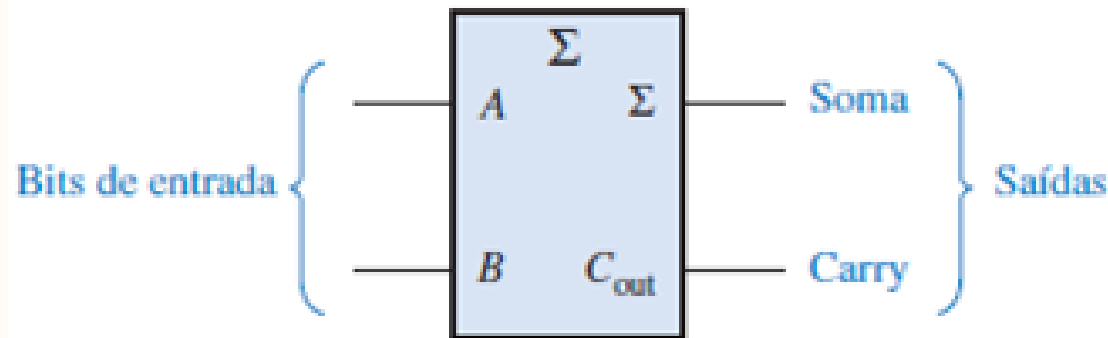
$$1 + 0 = 1$$

$$1 + 1 = 10$$

As operações são realizadas por um circuito lógico chamado de **meio-somador**.

O meio-somador aceita dois dígitos binários em suas entradas e produz dois dígitos binários em suas saídas, um bit de soma e um bit de carry.

Um meio-somador é representado pelo símbolo lógico



Lógica do Meio-Somador A partir da operação do meio-somador expressa na Tabela podemos deduzir expressões para a soma (resultado) e para o carry de saída como funções das entradas. Observe que o carry de saída (C_{out}) é 1 apenas quando A e B são 1s; portanto, C_{out} pode ser expresso como uma operação AND entre as variáveis de entrada.

$$C_{out} = AB$$

A	B	C_{out}	Σ
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Σ = soma

C_{out} = carry de saída

A e B = variáveis de entrada (operandos)

Agora observe que a saída soma (Σ) é 1 apenas se as variáveis de entrada, A e B , não forem iguais. A soma pode, portanto, ser expressa como a operação EX-OR entre as variáveis de entrada.

$$\Sigma = A \oplus B$$

pode-se deduzir a implementação lógica exigida para o meio-somador. O carry de saída é produzido com uma porta AND com A e B nas entradas e a saída da soma é gerada com uma porta EX-OR, conforme mostra a Figura. Lembre que a EX-OR é implementada com portas AND, uma porta OR e inversores.

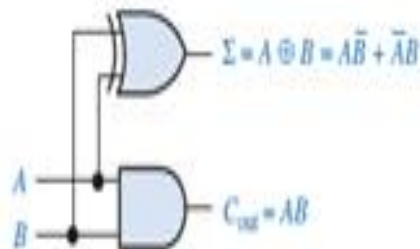




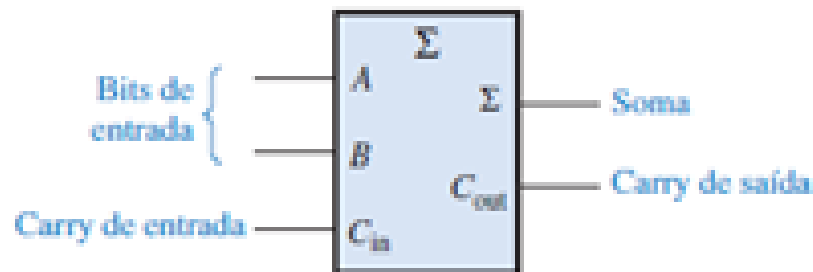
Diagrama lógico de um meio-somador.

Somador-Completo

A segunda categoria de somadores é o **somador-completo**.

O somador-completo aceita dois bits de entrada e um carry de entrada, e gera uma saída de soma e um carry de saída.

A diferença básica entre um somador-completo e um meio-somador é que o somador-completo aceita um carry de entrada. A Figura  mostra o símbolo lógico para um somador-completo e a Tabela  mostra a operação da tabela-verdade para um somador-completo.



A	B	C_{in}	C_{out}	Σ
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

C_{in} = carry de entrada, algumas vezes indicado como CI

C_{out} = carry de saída, algumas vezes indicado como CO

Σ = soma

A e B = variáveis de entrada (operandos)

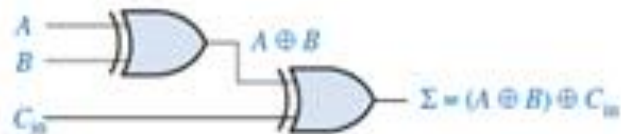
Tabela-verdade do somador-completo

Circuito Lógico do Somador-Completo O somador-completo soma os dois bits de entrada e o carry de entrada. A partir do meio-somador sabemos que a soma dos bits de entrada A e B é a EX-OR

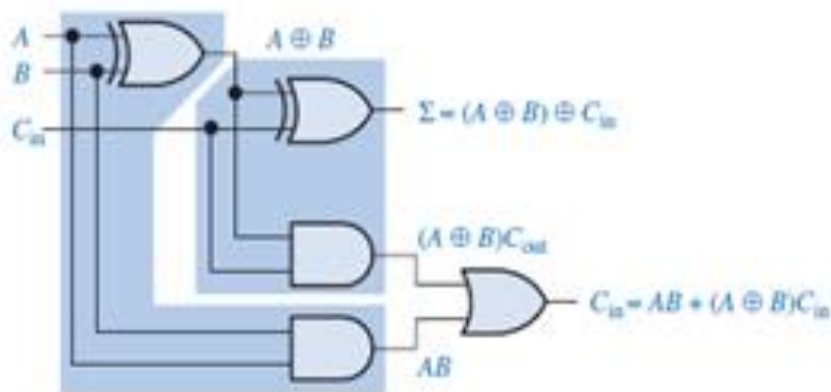
dessas duas variáveis $A \oplus B$. Para o carry de entrada (C_{in}) ser somado aos bits de entrada, deve-se fazer uma EX-OR com $A \oplus B$, resultando na equação para a saída da soma do somador-completo.

$$\Sigma = (A \oplus B) \oplus C_{in}$$

Isso significa que para implementar a função soma do somador-completo, usa-se duas portas EX-OR de 2 entradas. A primeira tem que gerar o termo $A \oplus B$ e a segunda tem como entradas a saída da primeira porta EX-OR e o carry de entrada, conforme ilustra a Figura



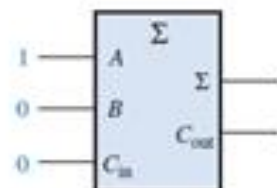
(a) Lógica necessária para construir um somador de três bits.



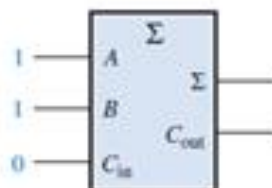
(b) Circuito lógico completo para um somador-completo (cada meio-somador se encontra numa área sombreada).

Para cada um dos três somadores-completos na Figura

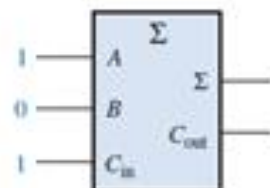
determine as saídas para as



(a)



(b)



(c)

Solução (a) Os bits de entrada são $A = 1$, $B = 0$ e $C_{in} = 0$.

$$1 + 0 + 0 = 1 \text{ sem carry}$$

Portanto, $\Sigma = 1$ e $C_{out} = 0$.

(b) Os bits de entrada são $A = 1$, $B = 1$ e $C_{in} = 0$.

$$1 + 1 + 0 = 0 \text{ com carry de } 1$$

Portanto, $\Sigma = 0$ e $C_{out} = 1$.

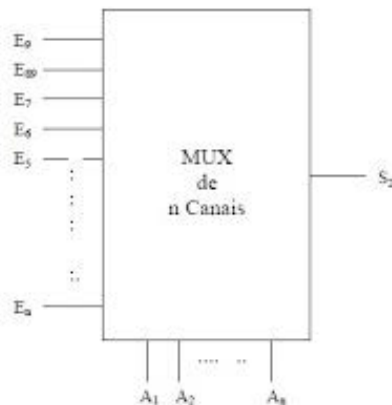
(c) Os bits de entrada são $A = 1$, $B = 0$ e $C_{in} = 1$.

$$1 + 0 + 1 = 0 \text{ com carry de } 1$$

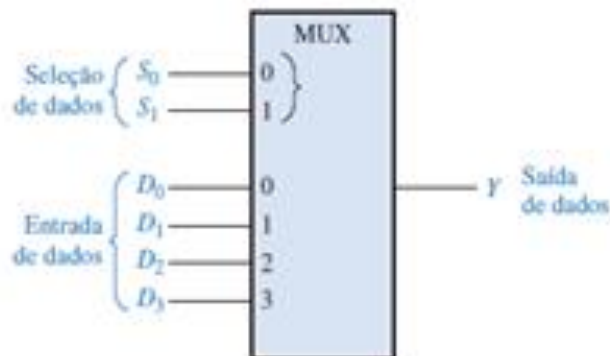
Portanto, $\Sigma = 0$ e $C_{out} = 1$.

MULTIPLEXADORES (SELETORES DE DADOS)

Um **multiplexador (MUX)** é um dispositivo que permite que informações digitais de diversas fontes sejam encaminhadas para uma única linha para serem transmitidas nessa linha para um destino comum. Um multiplexador básico tem várias linhas de entrada de dados e uma única linha de saída. Ele também possui entradas de seleção de dados, as quais permitem que os dados digitais de quaisquer entradas sejam comutados para a linha de saída. Os multiplexadores também são conhecidos como seletores de dados.



mostra o símbolo lógico para um multiplexador (MUX) de 4 bits. Observe que existem duas linhas de seleção de dados porque com dois bits de seleção, qualquer uma das quatro linhas de entrada de dados pode ser selecionada.



Símbolo lógico para um seletor/multiplexador de dados 1 de 4.

Na Figura um código de 2 bits nas entradas de seleção de dados (S) permitem que o dado na entrada selecionada passe para a saída de dados. Se um binário 0 ($S_1 = 0$ e $S_0 = 0$) for aplicado nas linhas de seleção de dados, o dado na entrada D_0 aparece na linha de saída de dados. Se um binário 1 ($S_1 = 0$ e $S_0 = 1$) for aplicado nas linhas de seleção de dados, o dado na entrada D_1 aparece na linha de saída de dados. Se um binário 2 ($S_1 = 1$ e $S_0 = 0$) for aplicado, o dado na entrada D_2 aparece na linha de saída de dados. Se um binário 3 ($S_1 = 1$ e $S_0 = 1$) for aplicado, o dado na entrada D_3 é comutado para a linha de saída de dados. A Tabela mostra um resumo dessa operação.

ENTRADAS DE SELEÇÃO DE DADOS		ENTRADA SELECIONADA
S_1	S_0	
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

Agora vamos pensar no circuito lógico necessário para realizar essa operação de multiplexação. A saída de dados é igual ao estado da entrada selecionada. Portanto, podemos deduzir uma expressão para a saída em termos da entrada de dados e das entradas de seleção.

A saída de dados é igual a D_0 apenas se $S_1 = 0$ e $S_0 = 0$: $D_0 \bar{S}_1 \bar{S}_0$.

A saída de dados é igual a D_1 apenas se $S_1 = 0$ e $S_0 = 1$: $D_1 \bar{S}_1 S_0$.

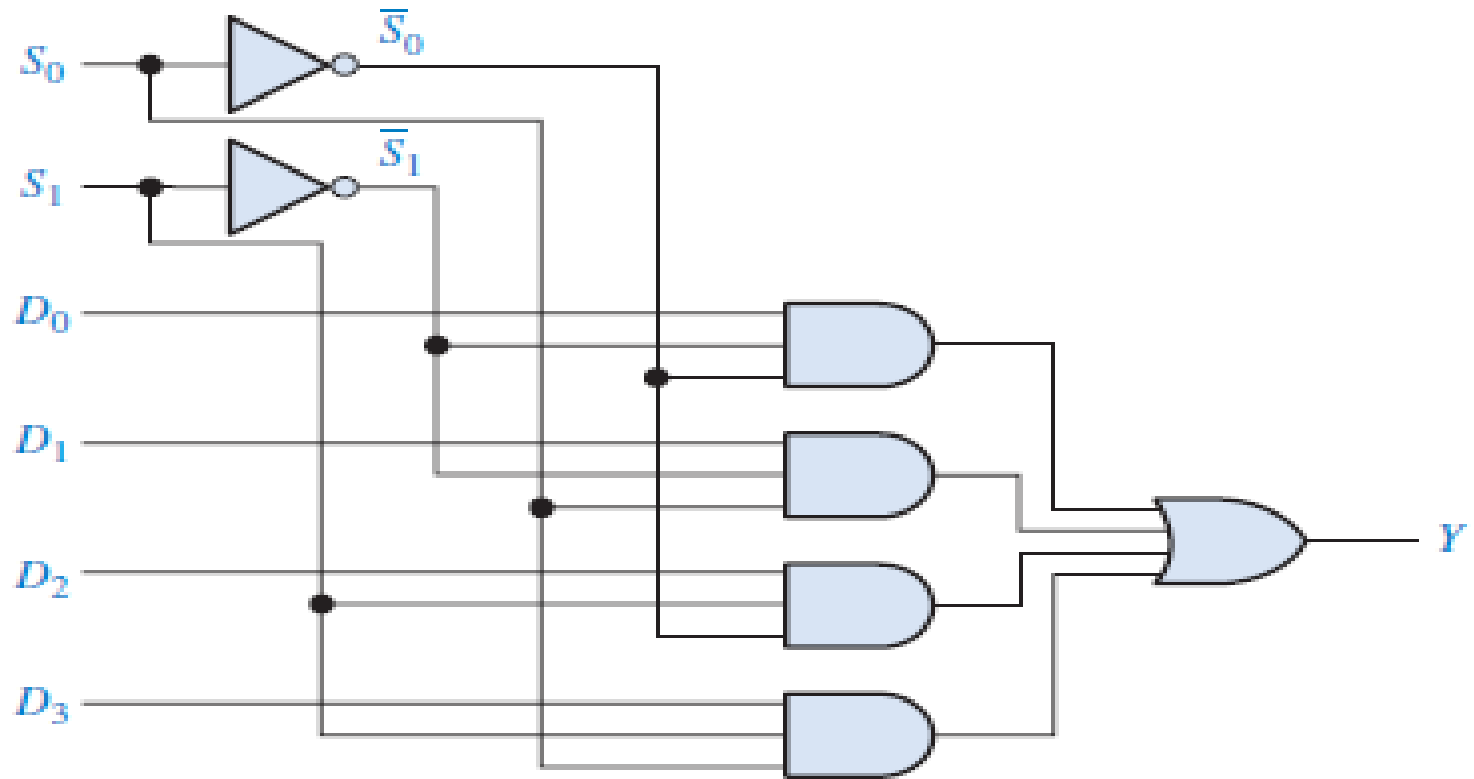
A saída de dados é igual a D_2 apenas se $S_1 = 1$ e $S_0 = 0$: $D_2 S_1 \bar{S}_0$.

A saída de dados é igual a D_3 apenas se $S_1 = 1$ e $S_0 = 1$: $D_3 S_1 S_0$.

Quando esses termos são relacionados por uma operação OR, a expressão total para a saída de dados é

$$Y = D_0 \bar{S}_1 \bar{S}_0 + D_1 \bar{S}_1 S_0 + D_2 S_1 \bar{S}_0 + D_3 S_1 S_0$$

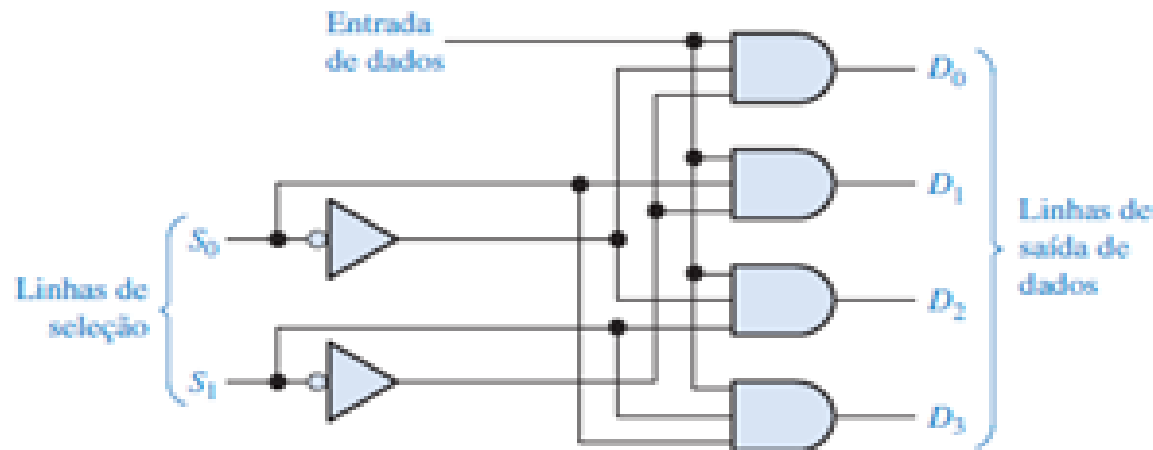
A implementação dessa equação requer quatro portas AND de 3 entradas, uma porta OR de 4 entradas e dois inversores para gerar o complemento de S_1 e S_0 , conforme mostra a Figura . Como os dados podem ser selecionados a partir de qualquer uma das linhas de entrada, esse circuito também é denominado de **seletor de dados**.



DEMULTIPLEXADORES

Um **demultiplexador (DEMUX)** basicamente inverte a função da multiplexação. Ele recebe informações digitais a partir de uma linha e as distribui para um determinado número de linhas de saída. Por essa razão, o demultiplexador também é conhecido como distribuidor de dados. Conforme estudaremos, os decodificadores também podem ser usados como demultiplexadores.

circuito de um demultiplexador (DEMUX) de 1 linha para 4 linhas. A linha de entrada de dados está conectada em todas as portas AND. As duas linhas de seleção de dados habilitam uma porta de cada vez, e os dados que aparecem na linha de entrada de dados passam, através da porta selecionada, para a linha de saída de dados associada.



A forma de onda de entrada de dados em série e as entradas de seleção de dados (S_0 e S_1) são mostradas na Figura . Determine as formas de onda da saída de dados D_0 a D_3 para o demultiplexador



#SOMOS
MAIS
UNIFG