Arquitetura de Computadores

Elementos Básicos de Hardware: Circuitos Lógicos Combinatórios

PARTE II

Referencias:

Floyd, Thomas L. ***Sistemas digitais: Fundamentos e Aplicações****, 9ª ed., Porto Alegre, ed. Bookman, 2007.*

Marcelo Marçula, Pio A. Benini Filho, ***Informática – Conceitos e Aplicações****, editora Érica*

Raul F. Weber, ***Fundamentos de Arquitetura de Computadores****, ed.Bookman*

Irv Englander, ***Arquitetura de Hardware Computacional, Software de Sistema e Comunicação em Rede. Uma abordagem da tecnologia da informação, (material suplementar) 4ª ed., LTC editora, 2011*** *.*

Introdução

Muitas funções em um computador são definidas em termos de suas equações booleanas.

Por exemplo, a soma de dois números binários de dígito único é representada por um par de tabelas verdade, uma para a soma de coluna efetiva e outra para o bit de transporte.

As tabelas verdade são mostradas na Figura S1.6.

Você deve reconhecer a tabela verdade para a soma como a operação exclusive-or, e a tabela para o transporte como a operação and.

De modo similar, a operação de complemento utilizada em subtração é simplesmente uma operação booleana not (ou nor).

Estas operações são **combinatórias**.

FIGURA S1.6

Tabelas verdade para a soma de dois números binários

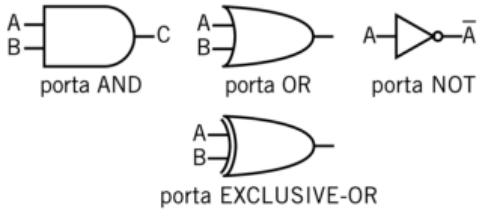
A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

soma

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

transporte

Representações de porta lógica padrão



Circuito Combinatório

Com diferentes combinações de portas lógicas, um sistema de computação realiza os cálculos que são a base para todas as suas operações.

O conjunto de portas lógicas agrupadas em um circuito é conhecido como **circuito combinatório**.

Circuitos combinacionais

São aqueles que não possuem memória ou quaisquer outros elementos de armazenamento.

Suas saídas são geradas exclusivamente a partir das entradas.

São construídos por portas lógicas sem realimentação.

Exemplo: Circuito Combinatório

A expressão **a + b . c'** pode ser expressa em termos de circuitos como:

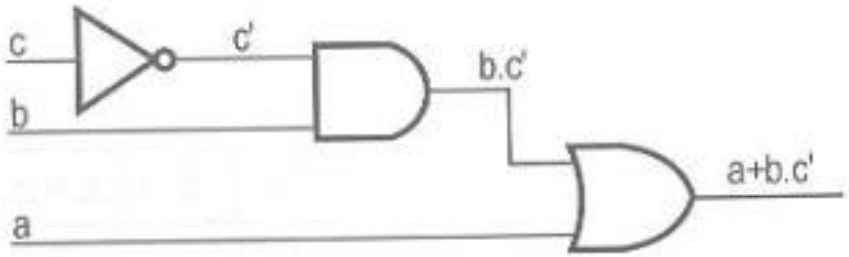
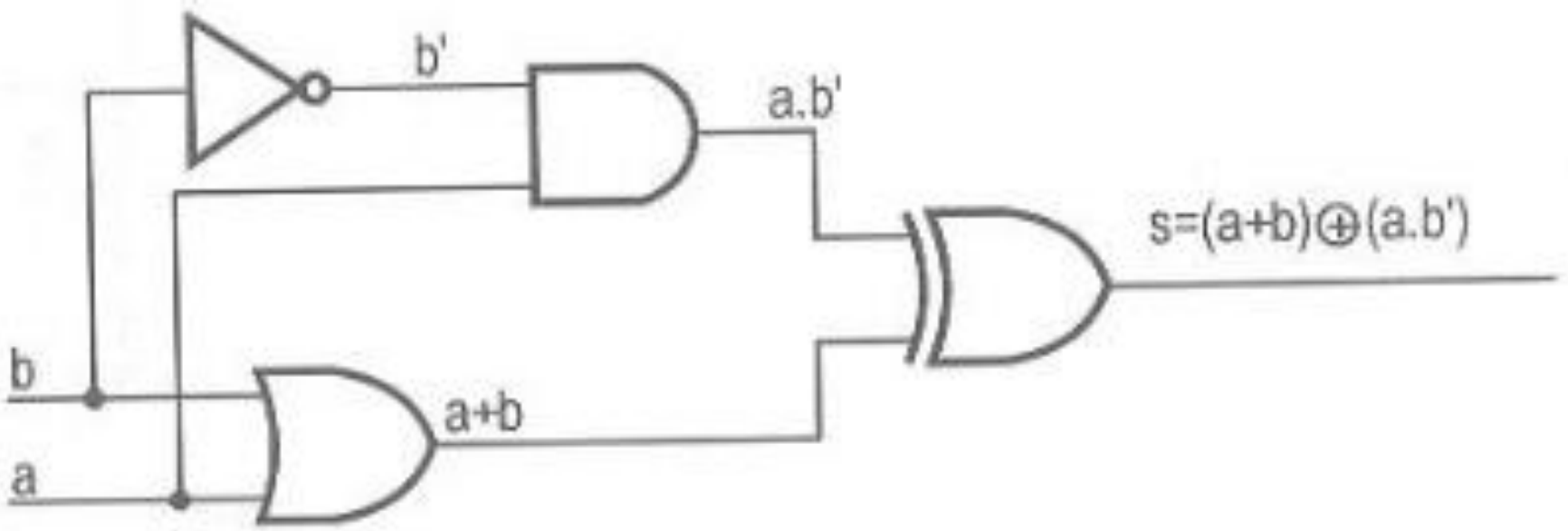


Figura 6.1 - Circuito combinatório.

a	b	c	S = a + b · c'
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Exemplo: Circuito Combinatório

Dada a equação booleana **(a + b) XOU (a . b')** temos:



Somador Parcial

Para realizar operações matemáticas, o computador utiliza combinações de portas lógicas chamadas somadores parciais e somadores completos.

O circuito abaixo é um somador parcial.

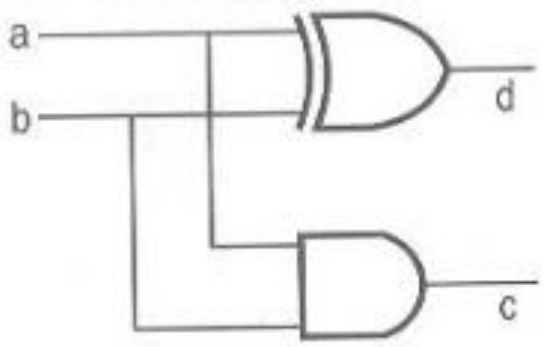


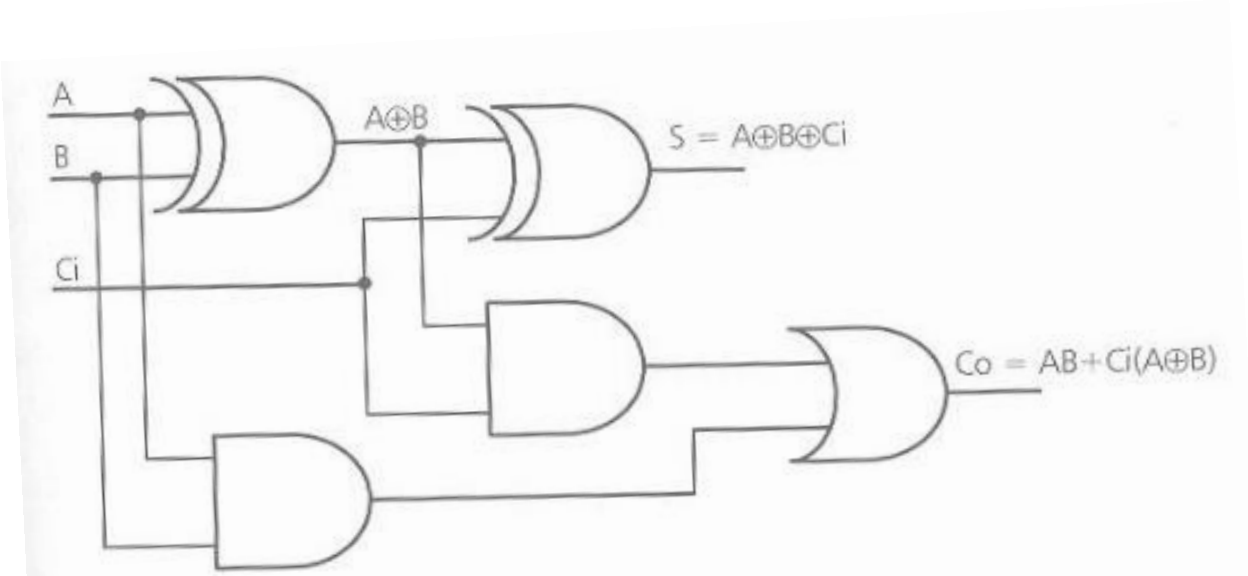
Figura 6.3 - Somador parcial.

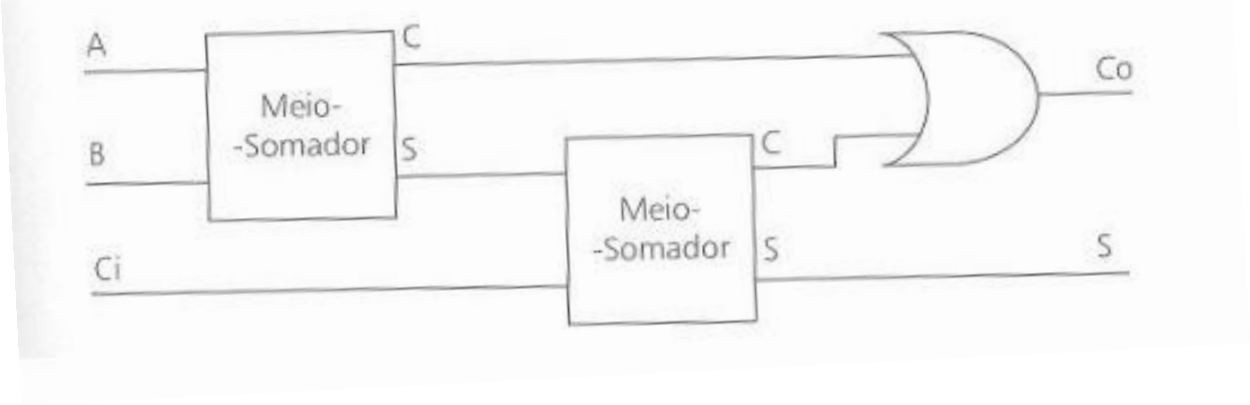
Entrada		Saída	
a	b	c	d
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Ele realiza a soma de dois bits (soma a e b).

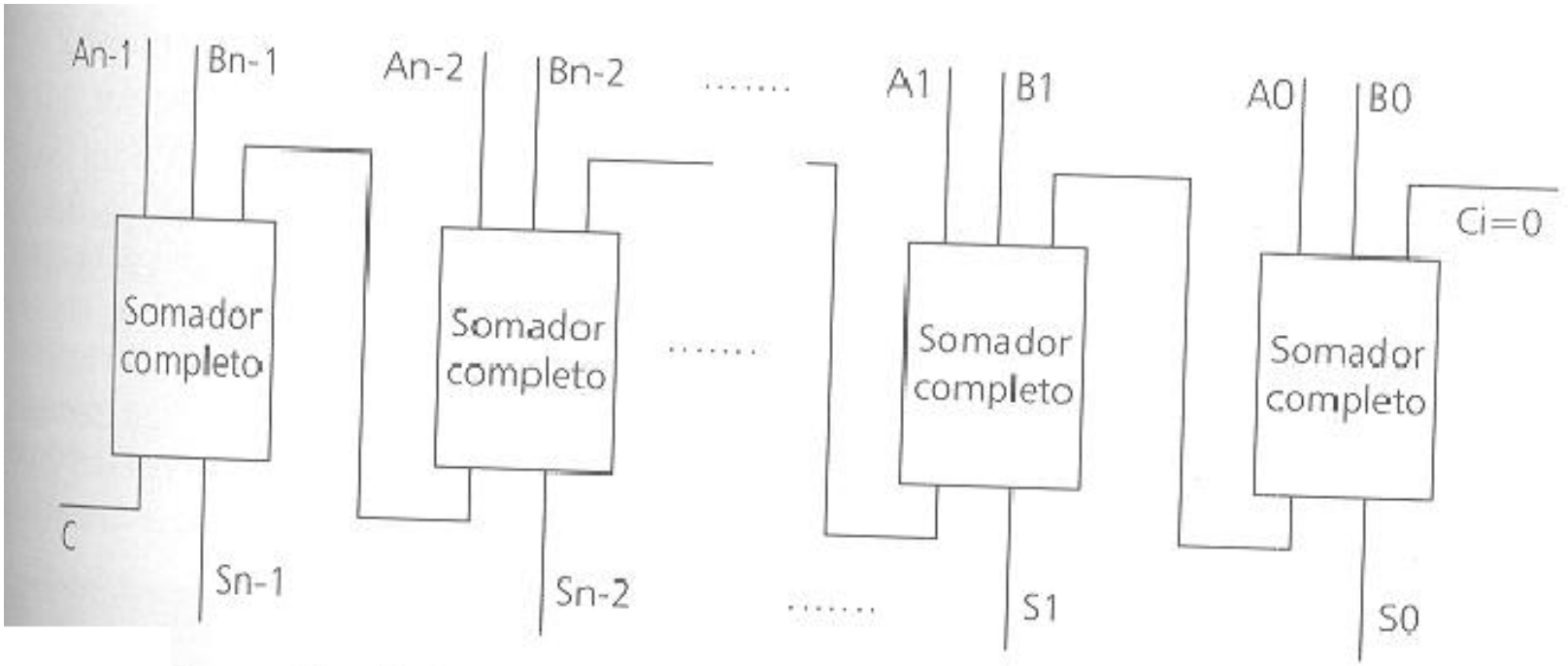
A saída ***d*** é a soma e a saída ***c*** é o bit “vai um”.

Somador Completo



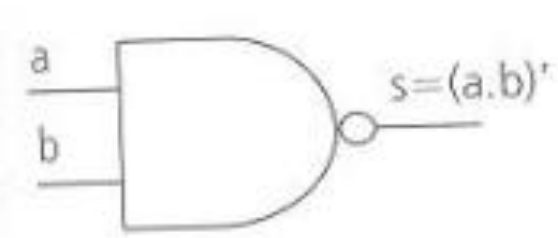


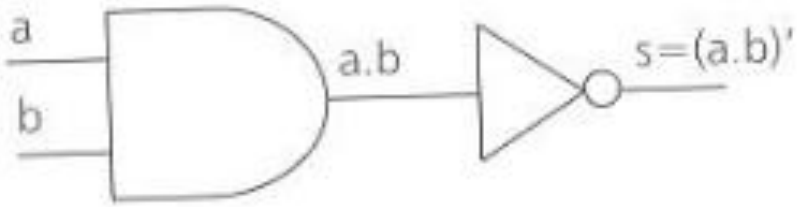
Somador Completo para n Bits

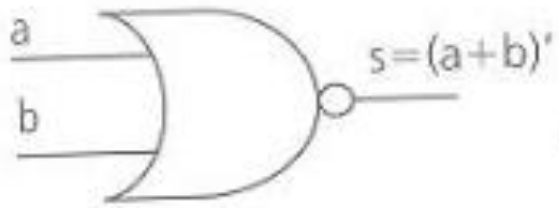


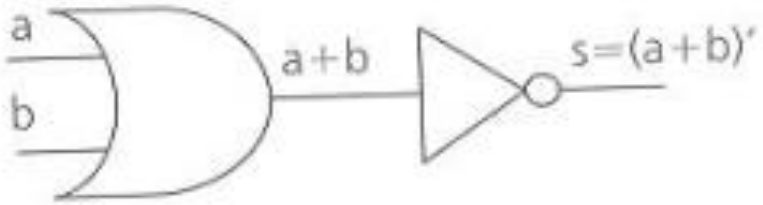
Somador binário de n bits.

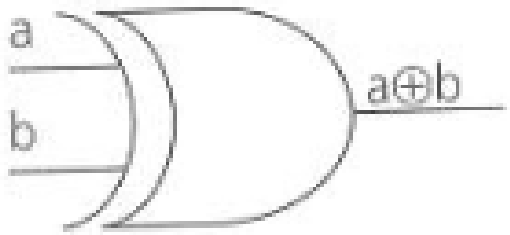
Há também as portas NE e NOU, que combinam as portas E e OU com uma porta NÃO. Existe também a XNOU.

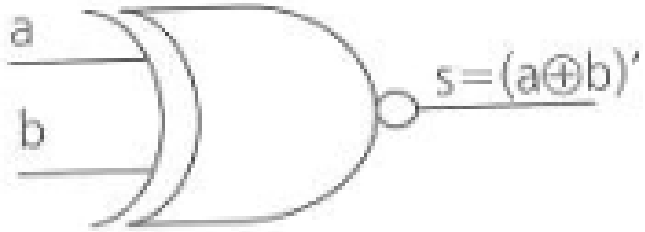












Equivalência entre portas x expressão

Equivalência de funções booleanas e portas lógicas

expressão ou porta	expressão equivalente
a + b	(a' . b')' a' nand b'
a . b	(a'+ b')' a' nor b'
a xor b	a.b' + a'.b
a'	(a.a)', a nand a (a+a)', a nor a a xor 1
a xnor b	a.b + a'.b'

Circuitos Multiplexadores

Dois circuitos combinacionais bem simples, bastante utilizados em sistemas digitais, são os ***multiplexadores*** e os ***decodificadores***.

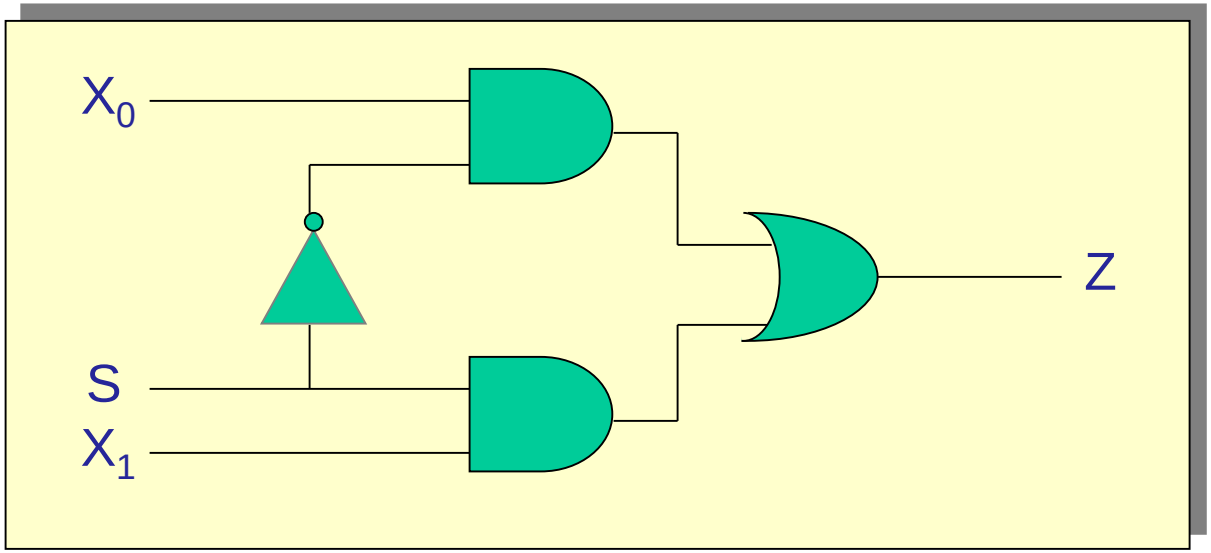
Um multiplexador (ou seletor) é um circuito combinacional que possui várias entradas e uma saída.

A cada instante, o valor da saída é igual ao valor de uma das entradas, conforme determinado por um conjunto de linhas de controle (ou linhas de seleção).

Tipos de Multiplexadores

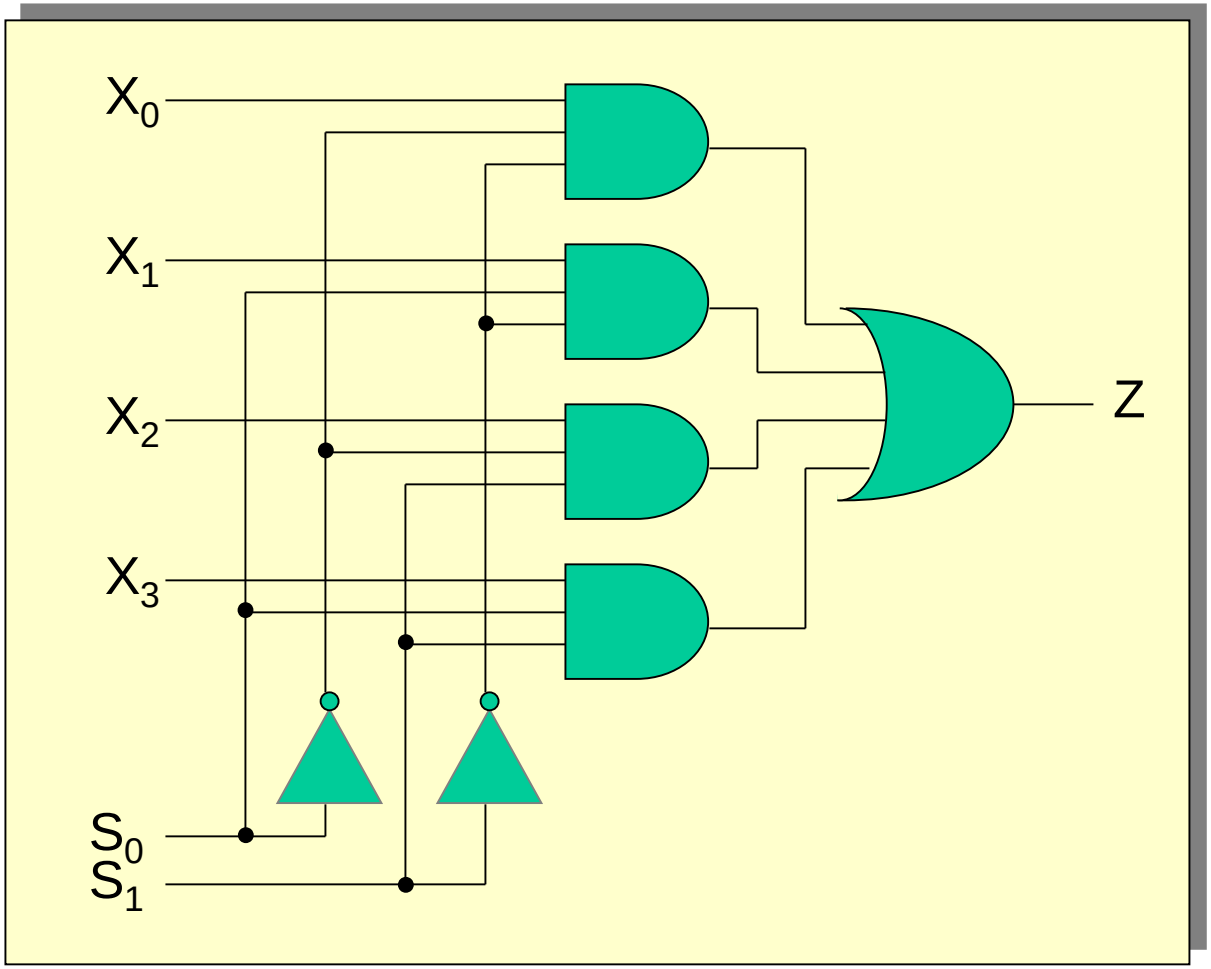
Multiplexador	Número de entradas	Número de linhas de seleção
2-para-1	2	1
4-para-1	4	2
8-para-1	8	3
16-para-1	16	4

Multiplexador 2 para 1



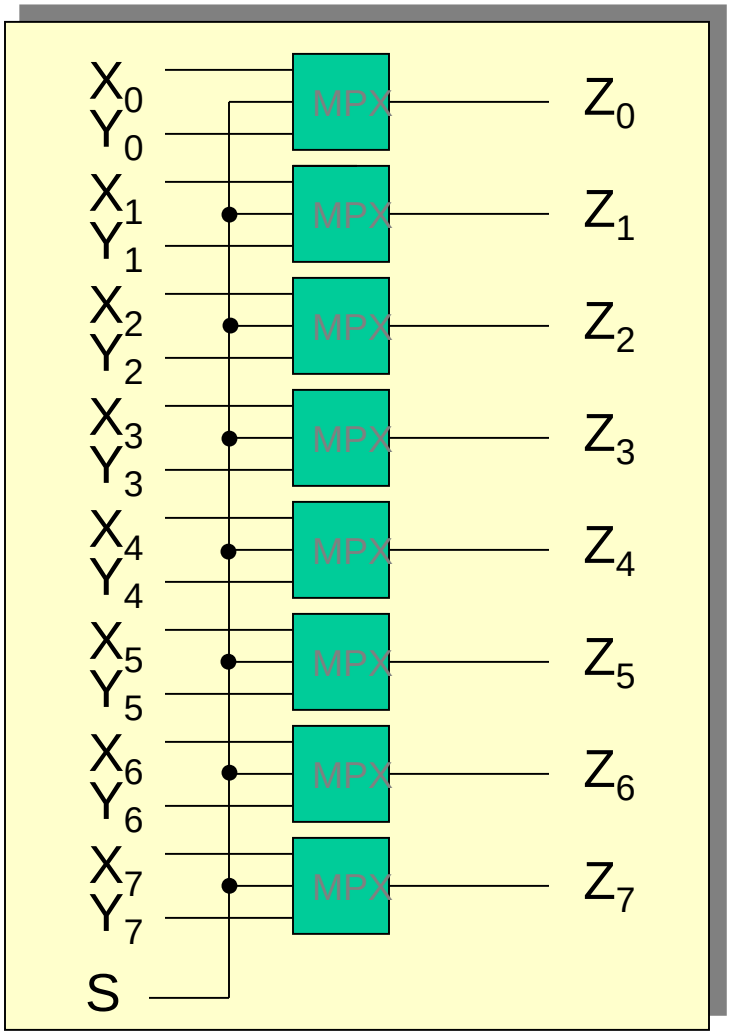
S	Z
0	X_0
1	X_1

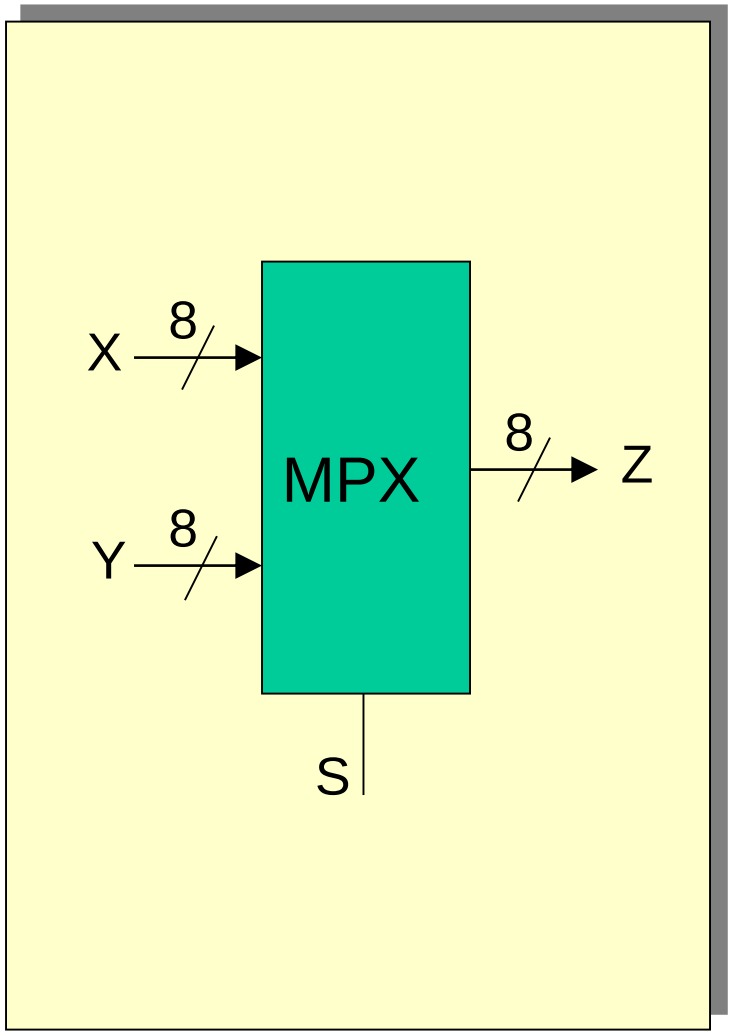
Multiplexador 4 para 1



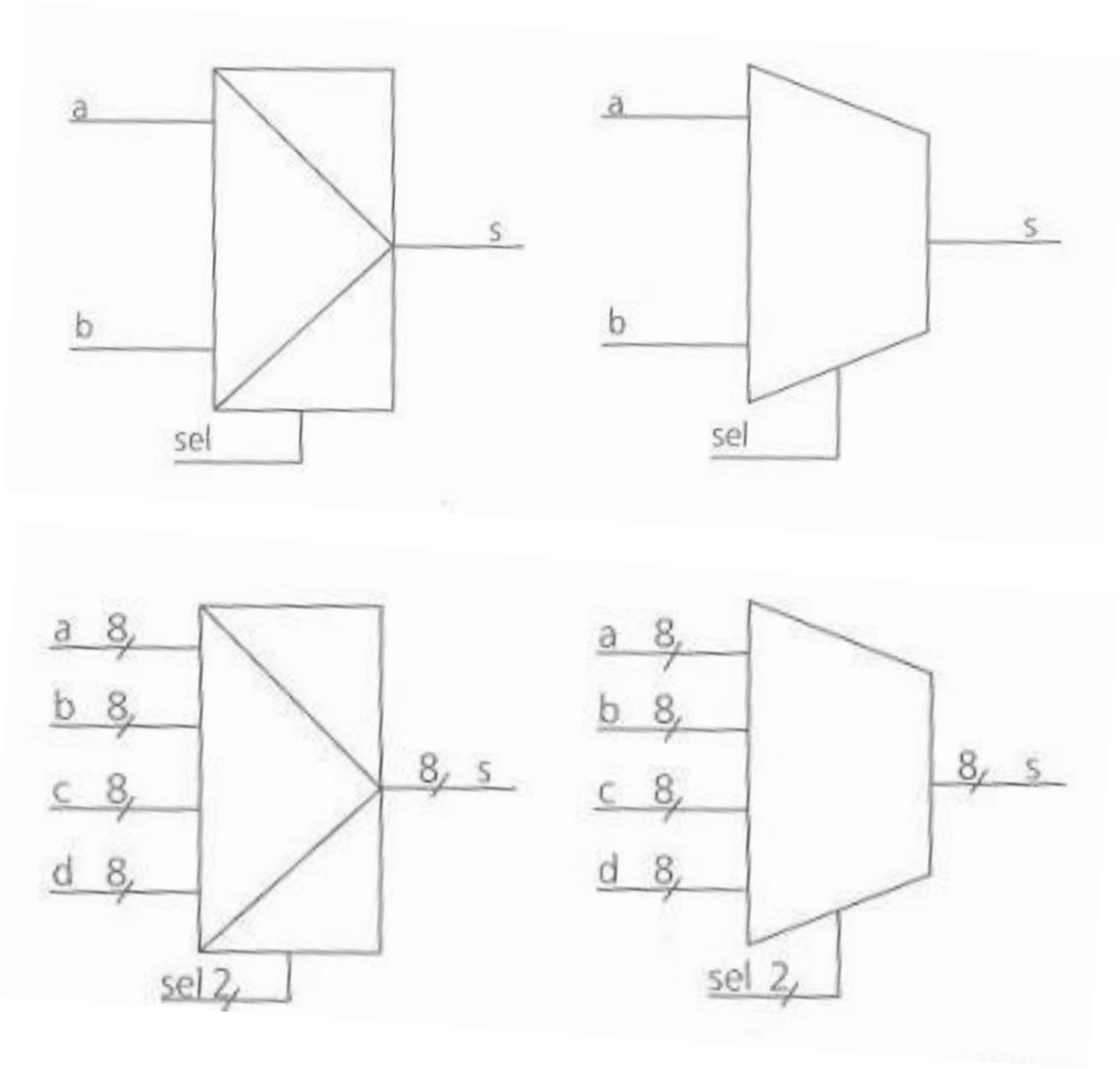
S_1	S_0	Z
0	0	X_0
0	1	X_1
1	0	X_2
1	1	X_3

Multiplexdoar 2 para 1 de 8 bits





Formas de Representação de Multiplexadores



Ccto.Combinacional: Decodificador

Outro circuito combinacional muito usado é um decodificador.

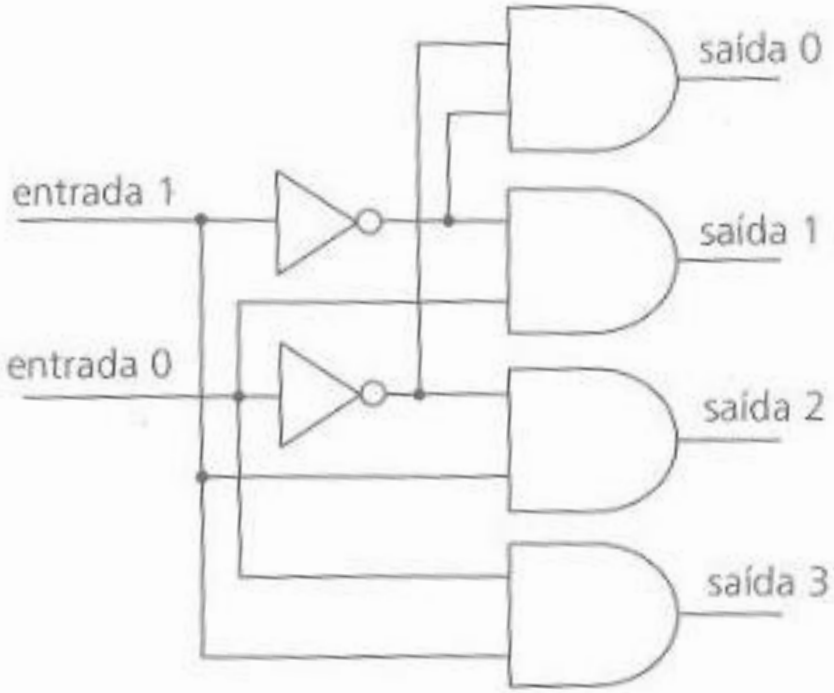
Ele possui **n** entradas e **2n** saídas.

Para cada combinação de entradas, uma saída possui sinal 1, o resto sinal 0.

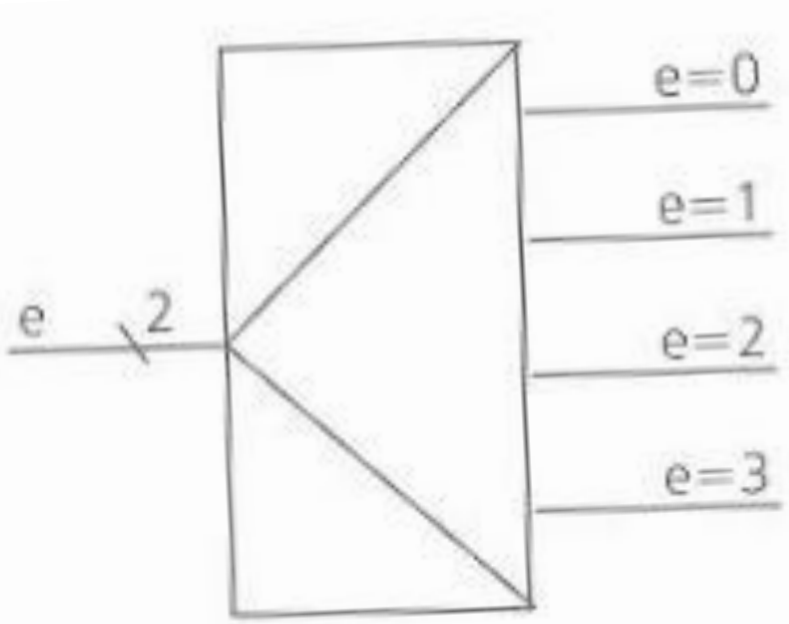
Exemplo: Ccto. Decodificador

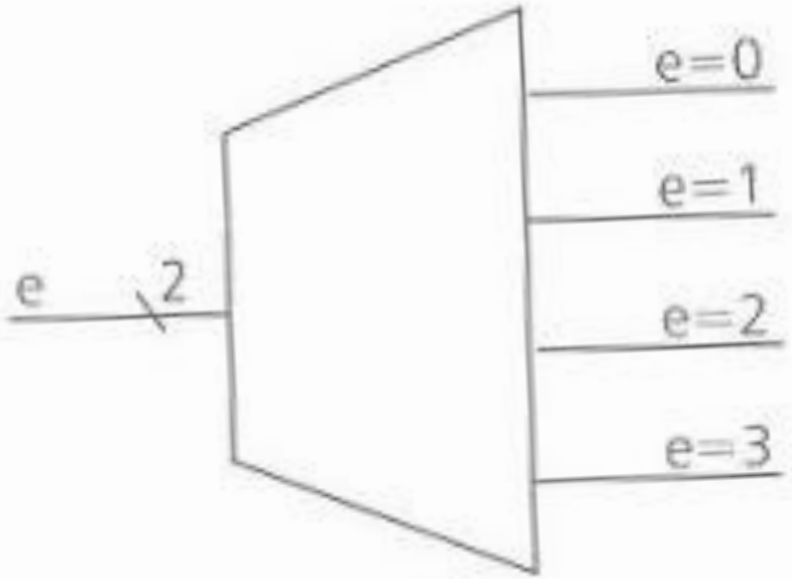
Tabela-verdade de um decodificador de 2-para-4

entrada 1	entrada 0	saída 0	saída 1	saída 2	saída 3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



Forma de Representação: Decodificador





Circuitos sequenciais

São aqueles que possuem memória.

Suas saídas são função tanto das entradas como dos valores da saída.

Ou seja, o novo valor de saída tem relação com o valor da saída anterior.

Circuitos Sequênciais

Dois circuitos sequenciais bastante utilizados são os registradores e os contadores.

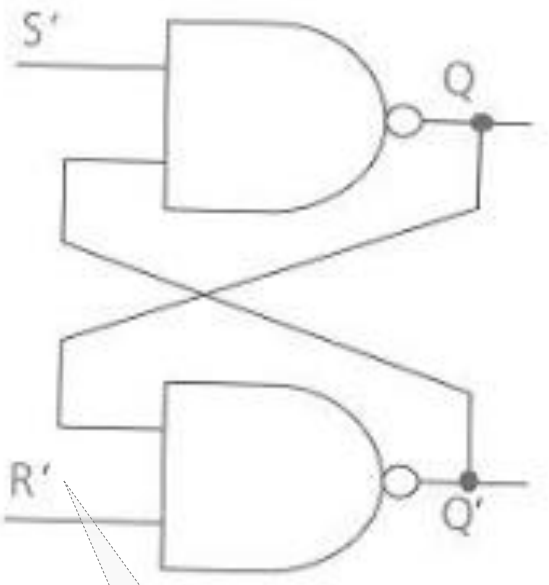
Ambos são construídos com ***flip-flops***, ou seja, registradores capazes de armazenar um único bit.

O ***flip-flop*** mais simples é o tipo ***RS***, que possui duas entradas: **R (*reset* ou desligar)** e **S (*set* ou ligar)**.

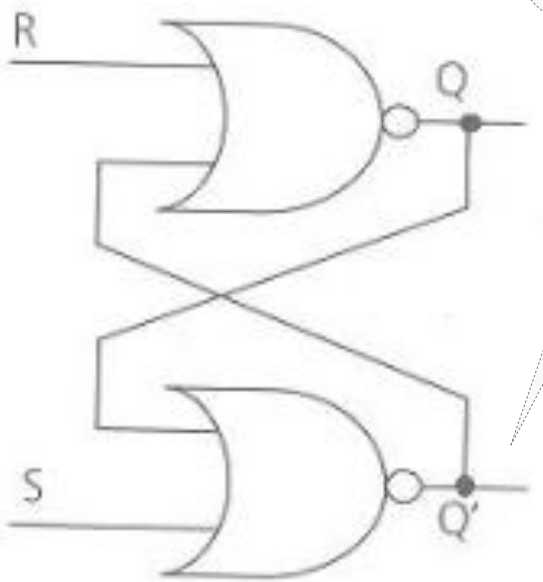
Se S recebe 1, a saída é 1, se R recebe 1, a saída é 0.

Exemplo de Ccto. Sequencial

Abaixo, duas formas de implementar um ***flip-flop RS***.



Obs.: R e S invertidos



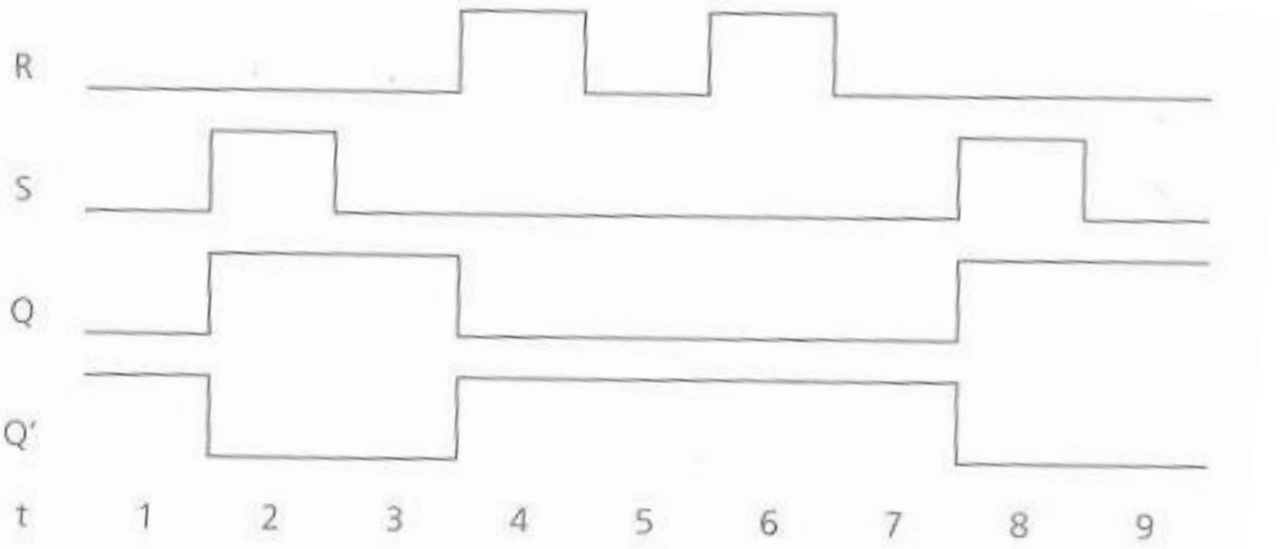
Obs.:Q' sempre o inverso de Q

R	S	Q_{t+1}	Resultado
0	0	Q_t	Estado inalterado
0	1	1	Passa para 1
1	0	0	Passa para 0
1	1	indeterminado	Condição de erro

Tabela Verdade : Flip-flop RS

Variação de sinais em um flip-flop RS

t	R	S	Q	Q'
1	0	0	0	1
2	0	1	1	0
3	0	0	1	0
4	1	0	0	1
5	0	0	0	1
6	1	0	0	1
7	0	0	0	1
8	0	1	1	0
9	0	0	1	0



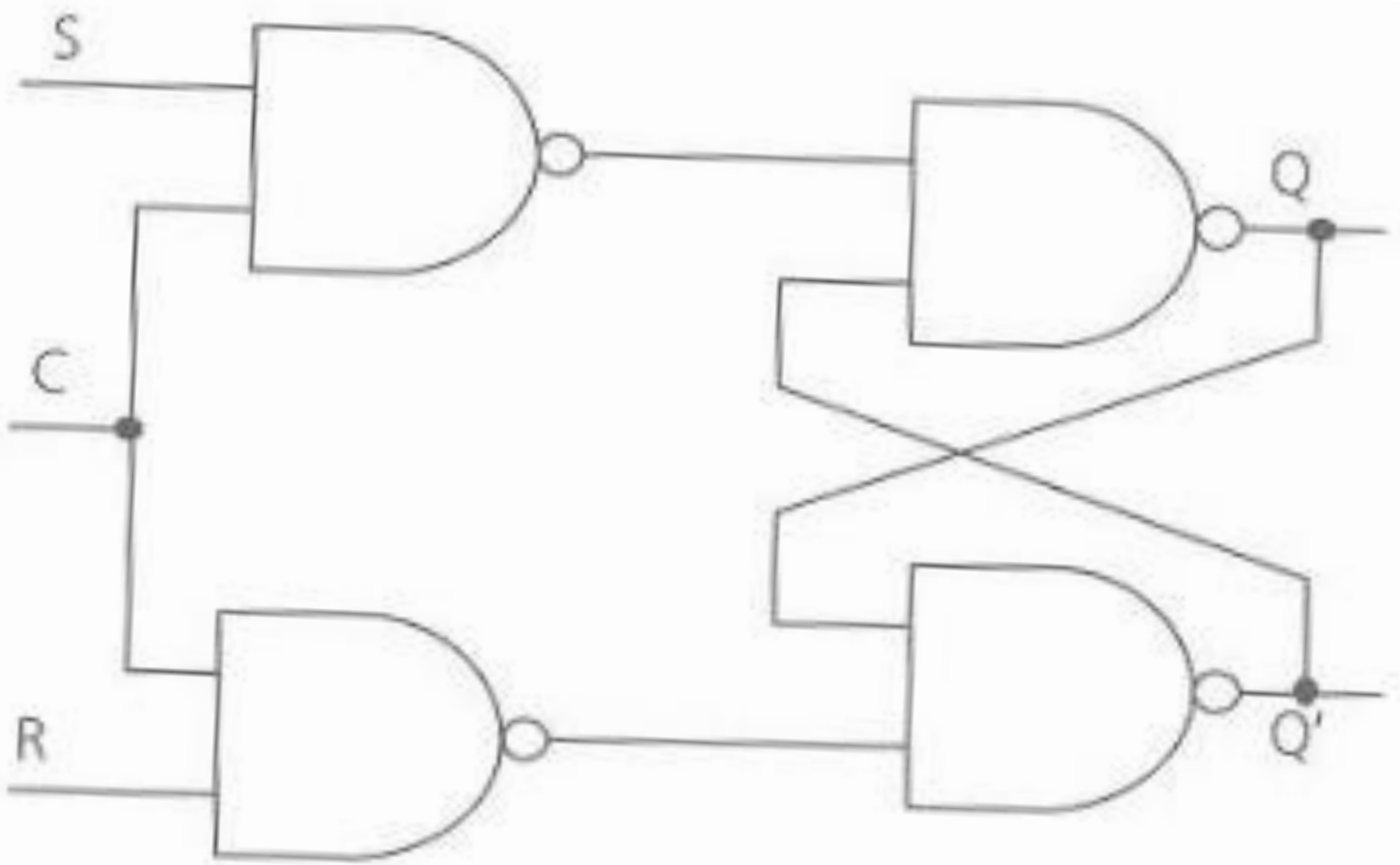
Flip-flop: Característica

O problema com esse *flip-flop (RS)* é que ele altera a saída sempre que houver uma variação nas entradas.

Houve a necessidade de criar um *flip-flop* que levasse em conta a alteração da entrada somente em determinados momentos.

Foi então criada uma entrada de controle chamada *clock*.

Flip-flop RS com controle



Flip-flop RS com controle.

Tem-se ainda o problema que, quando R=S=C =1 ; Q e Q' serão iguais (erro de lógica).

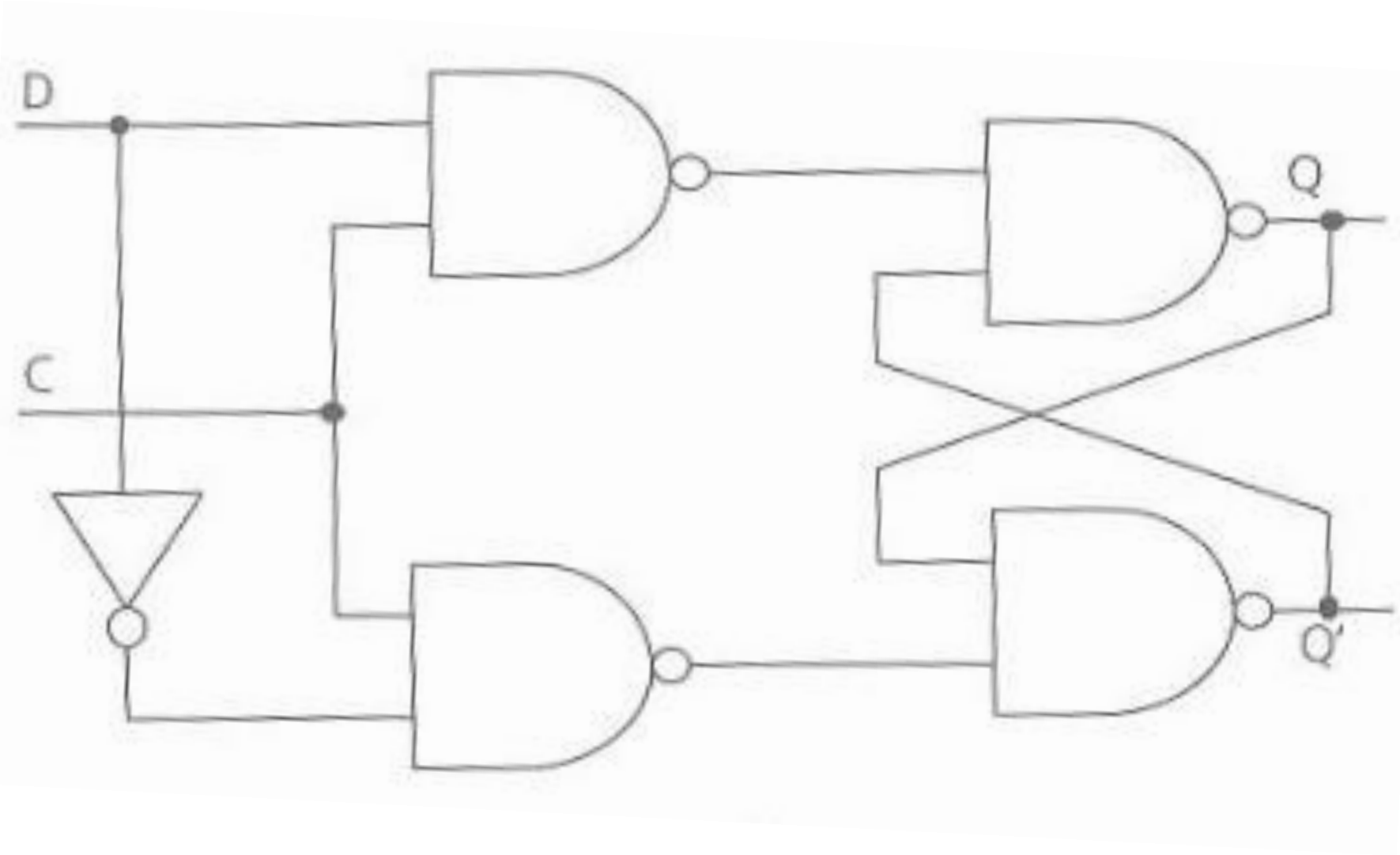
Flip-flop tipo D

Para evitar a indeterminação quando R e S estão com entrada 1, pode ser colocado um inversor para interligá-los.

Dessa forma temos um *flip-flop* que copia o valor lógico da entrada D (entrada de dado) quando o controle estiver ativo.

Esse é o ***flip-flop* do tipo D** que armazena o valor de D quando o controle é 1.

Flip-flop tipo D



Tipos de Flip-flop

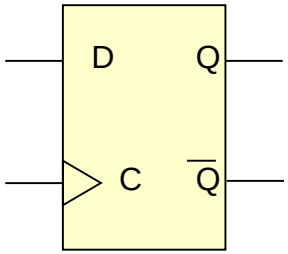
Há *flip-flops* sensíveis ao nível 1(do clock), outros sensíveis ao nível 0;

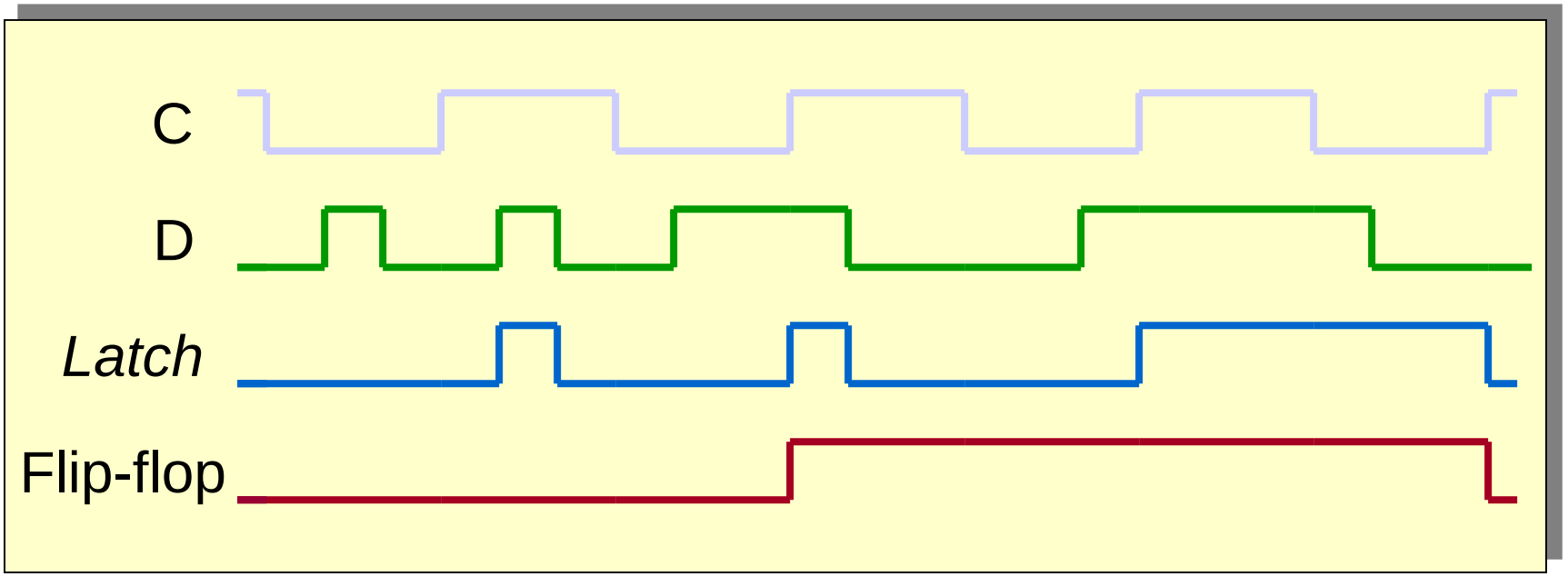
Há *flip-flops* sensíveis à subida e sensíveis à decida (da borda do clock).

Há *flip-flops* do tipo **T** (*toggle*) que muda o valor a cada mudança do *clock*.

Há flip-flops do tipo **JK**, que é similar ao D mas tem duas entradas.

Latches e Flip-flops D





- *Latch: memoriza o valor de D quando C está habilitado (C=1)*
- *Flip-flop: memoriza o valor de D quando C transita de 0 para 1 (sensível a borda positiva)*

Registrador

Um registrador, além de armazenar, pode executar algumas funções nos bits que armazena.

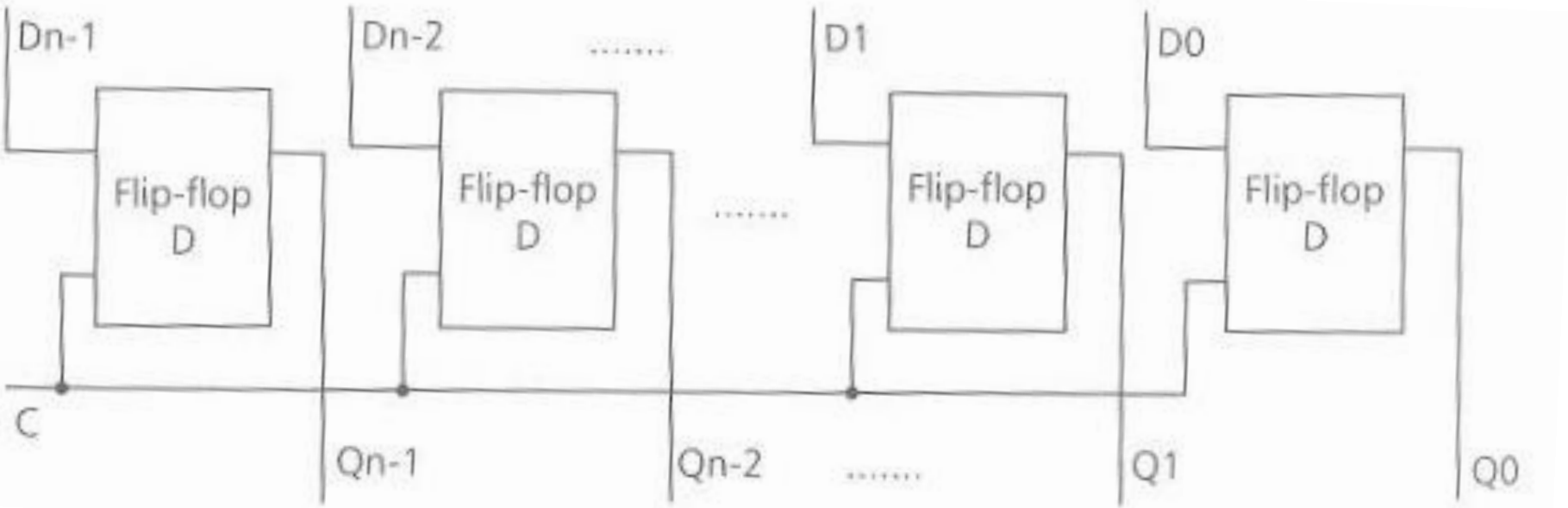
Por exemplo registradores de deslocamento (*shift*) e registrador contador (*counter*).

Esses registradores especiais são facilmente implementados com *flip-flops*.

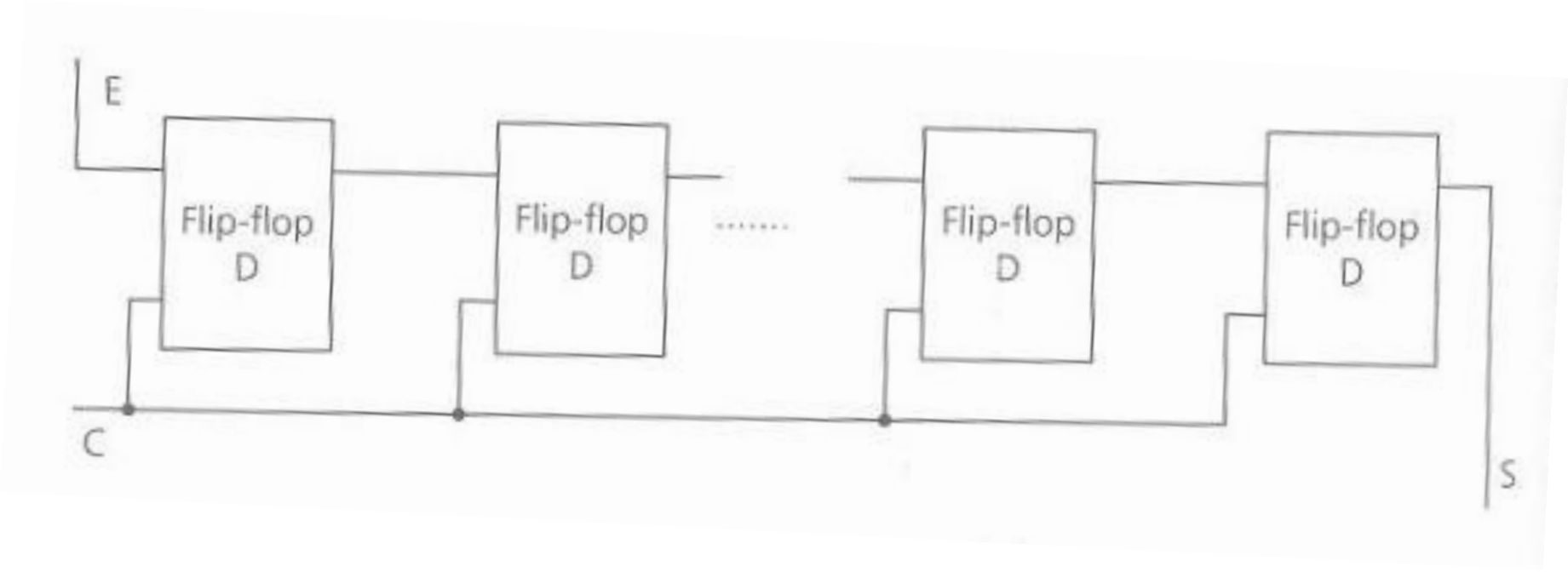
Registrador com flip-flop tipo D

Um conjunto de **n** ***flip-flops*** pode ser interconectado para formar um **registrador** de n bits, ou seja, um registrador capaz de armazenar n bits.

Um **registrador** desse tipo possui uma entrada D para cada bit e um controle em conjunto para todos os bits.



Registrador de Deslocamento



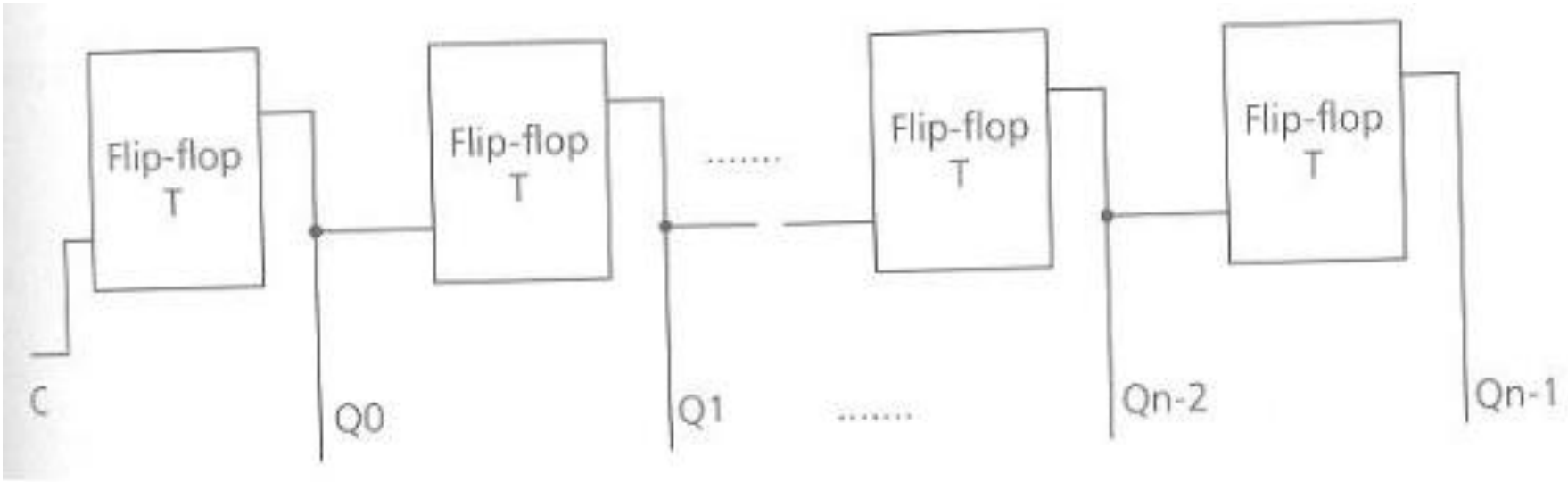
Registrador Contador

Um registrador contador, ou simplesmente um contador, é um registrador que, com a ativação do sinal de controle, incrementa (ou decrementa) o seu valor em uma unidade.

Dependendo da contagem desejada (binária, BCD, etc), o contador apresenta uma estrutura interna adequada.

Contadores sensíveis à borda de subida são contadores decrescentes e sensíveis à borda de descida são crescentes.

Registrador Contador



Contador binário de n bits.

ULA: Unidade Lógica Aritmética

Uma das partes mais importantes de um computador é sua ULA.

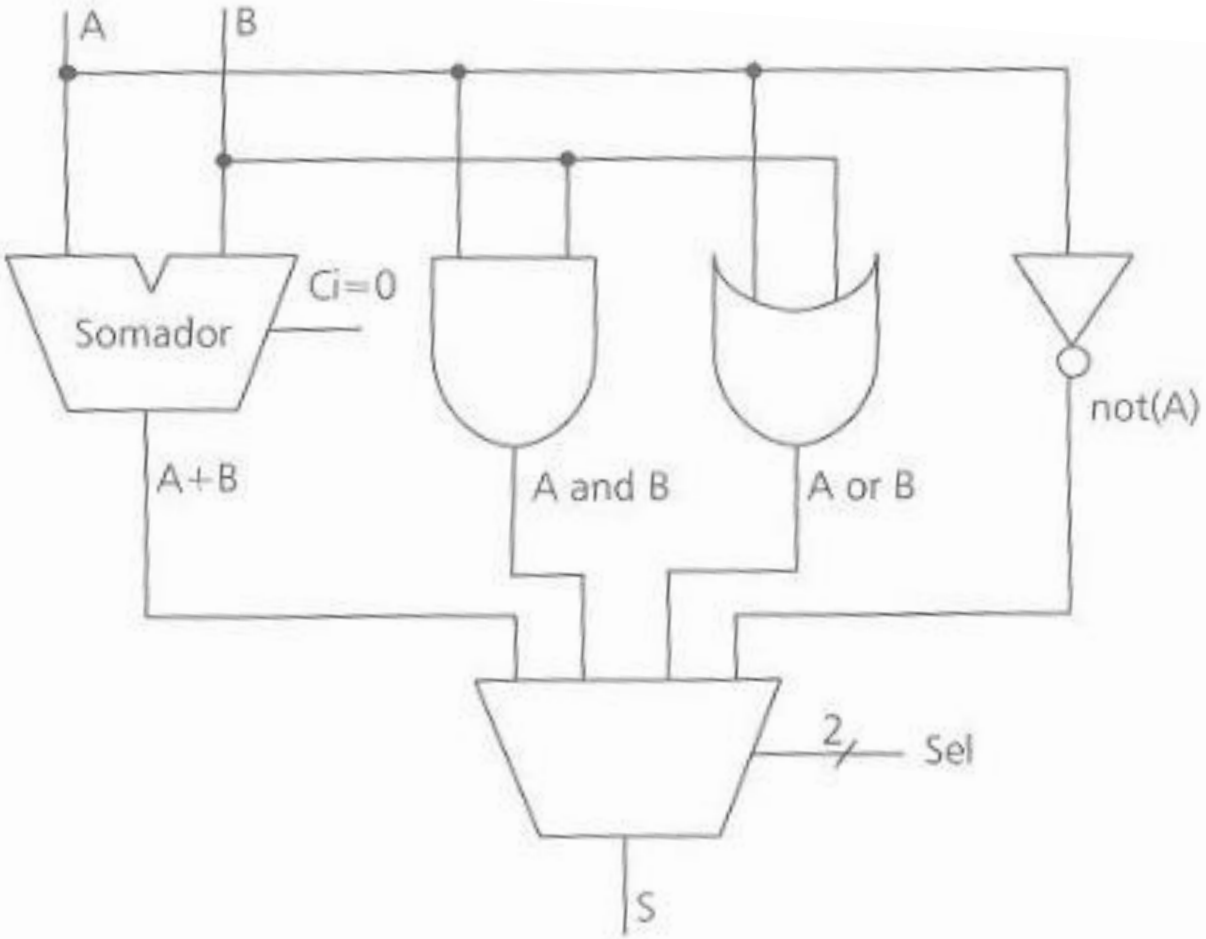
Essa unidade é responsável por cálculos: soma, subtração, funções booleanas, etc.

Sua complexidade é proporcional à complexidade do conjunto de instruções do computador.

Se uma ULA realiza várias funções, uma forma simples de implementá-la é implementar cada função e juntá-las por meio de um multiplexador.

Exemplo: suponha uma ULA com as operações SOMA, E, OU e NÃO para entradas de n bits.

ULA com 4 Operações



ULA com 4 operações.