FUNDAMENTOS COMPUTACIONAIS

Pedro Kislanskly

SAGAH
SOLUÇÕES EDUCACIONAIS INTEGRADAS

Portas lógicas e circuitos digitais

Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Conceituar as portas lógicas básicas.
- Listar as principais características das portas lógicas.
- Enumerar os tipos de portas lógicas e seus circuitos.

Introdução

Portas lógicas são a base construtiva de qualquer sistema digital. Elas são usadas para criar circuitos digitais e até mesmo circuitos integrados complexos. Por exemplo, circuitos integrados complexos podem ser circuitos digitais completos prontos para serem usados — processadores e microcontroladores são os melhores exemplos, mas internamente esses circuitos integrados foram projetados usando várias portas lógicas.

Neste capítulo, você vai estudar os conceitos básicos relacionados às portas lógicas, suas principais características, seus tipos e sua relação com circuitos digitais.

Portas lógicas

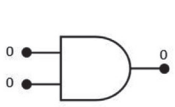
Portas lógicas são a base para compreender os circuitos digitais. Inicialmente, você vai entender as portas lógicas básicas; depois, por meio da álgebra de Boole, vai compreender as principais operações que podem ser realizadas por elas. O termo **porta** é usado para descrever um circuito que realiza uma operação lógica básica.

Os símbolos lógicos usados para representar as portas lógicas estão de acordo com o padrão 91-1984, da ANSI/IEEE. Esse padrão foi adotado pela indústria privada e militar para uso em documentações internas, bem como na literatura publicada (FLOYD, 2007).

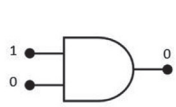
Tanto a lógica programável quanto a lógica de funções fixas são discutidas neste capítulo. Em função de os circuitos integrados (CIs) serem usados em todas as aplicações, as funções lógicas de um dispositivo geralmente são mais importantes para o técnico ou tecnólogo, do que os detalhes da operação do circuito em nível de componentes, dentro do encapsulamento do CI. Portanto, a abordagem detalhada de dispositivos em nível de componente pode ser tratada como um tópico opcional (FLOYD, 2007).

Porta AND

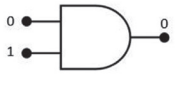
A porta AND é uma porta que implementa o "E" lógico, ou seja, ele só é verdadeiro (1) quando as duas entradas são verdadeiras (1 e 1). Todas as outras opções têm como saída **falso** (0), como você pode ver na Figura 1. Independentemente de quantas entradas a porta tem (duas, três, quatro, n), a saída só será 1 se todas as entradas forem 1.



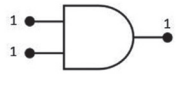
A	B	S = A.B
0	0	0
0	1	0
1	0	0
1	1	1



A	B	S = A.B
0	0	0
0	1	0
1	0	0
1	1	1



A	B	S = A.B
0	0	0
0	1	0
1	0	0
1	1	1

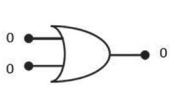


A	B	S = A.B
0	0	0
0	1	0
1	0	0
1	1	1

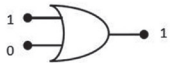
Figura 1. Porta lógica AND.

Porta OR

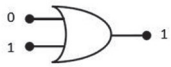
A porta OR implementa o "OU" lógico. Nesse caso, a saída será falsa (0) somente se todas as entradas forem falsas. Se pelo menos uma entrada for verdadeira (1), então a saída será verdadeira (Figura 2).



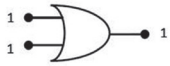
A	B	S = A+B
0	0	0
0	1	1
1	0	1
1	1	1



A	B	S = A+B
0	0	0
0	1	1
1	0	1
1	1	1



A	B	S = A+B
0	0	0
0	1	1
1	0	1
1	1	1

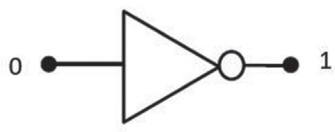


A	B	S = A+B
0	0	0
0	1	1
1	0	1
1	1	1

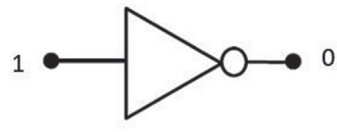
Figura 2. Porta lógica OR.

Porta NOT

A porta NOT (NÃO) — ou inversor — implementa uma negação lógica: se a entrada é 1, a saída será 0; se a entrada for 0, a saída será 1. Essa função pode ser utilizada em conjunto com outras portas e, assim, serve para inverter o sinal de saída ou de uma entrada específica (daí o nome de inversor). A Figura 3 mostra a tabela e o desenho correspondentes à função NOT.



A	S = $\bar{A}$
0	1
1	0



A	S = $\bar{A}$
0	1
1	0

Figura 3. Porta lógica NOT ou inversor.

Porta NAND e porta NOR

As portas AND e OR podem ser combinadas com o inversor lógico NOT, produzindo as portas lógicas NAND e NOR. O sinal de saída é, como o nome sugere, inverso ao sinal da porta sem o NOT. Essas portas podem ser vistas na Figura 4.

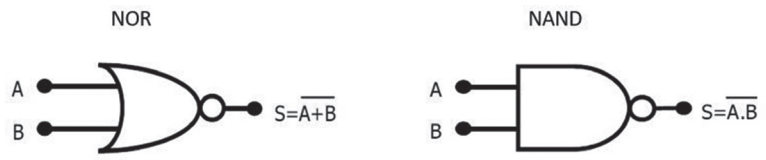


Figura 4. Portas lógicas NAND e NOR.

Porta XOR

A porta XOR (ou OU EXCLUSIVO) fornece saída 1 quando as entradas forem diferentes entre si, e 0 em caso contrário. Veja a figura correspondente na Figura 5.

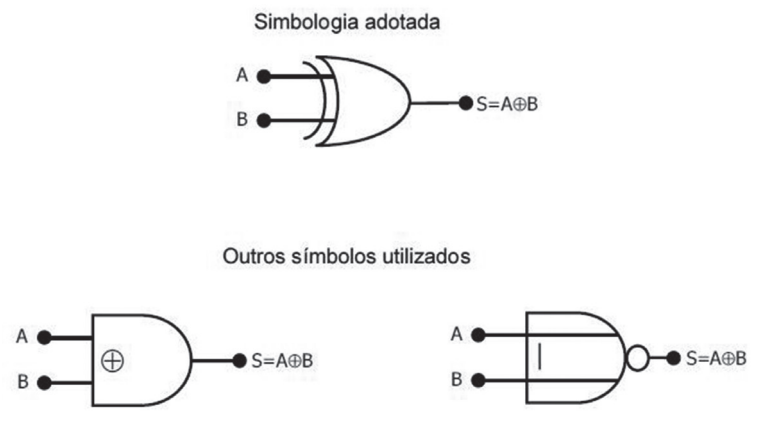


Figura 5. Porta lógica XOR (diferentes formas de representação).

A Figura 6 mostra um resumo das portas lógicas vistas até agora.

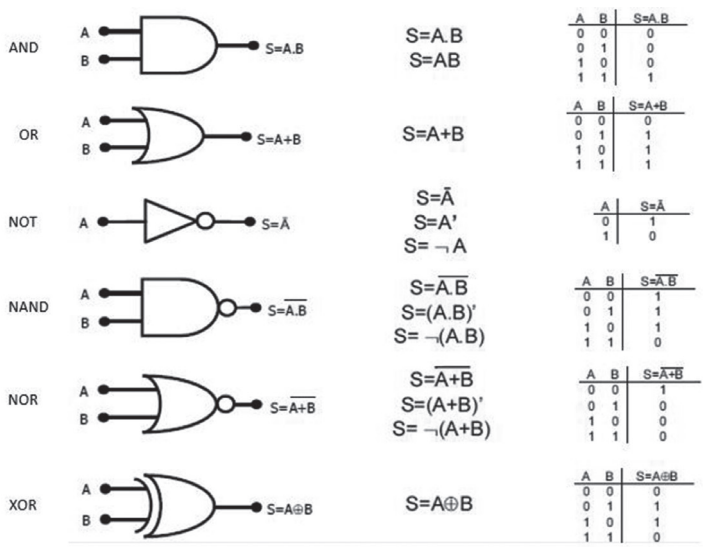


Figura 6. Resumo das portas lógicas básicas.

Circuitos integrados

Um circuito integrado, também chamados CI ou **chip**, é um pedaço quadrado de silício, de aproximadamente 5 x 5 mm, contendo um conjunto de portas lógicas e encapsulado em um invólucro retangular de plástico ou cerâmica, de 5 a 15 mm de largura e 20 a 50 mm de comprimento (FELGUEIRAS, [200-?]).

Os CIs podem ser classificados, quanto à quantidade de portas lógicas, da seguinte forma:

- **Circuito SSI (*Small Integration Scale*):** de 1 a 10 portas lógicas (Figura 7).
- **Circuito MSI (*Medium Integration Scale*):** de 10 a 100 portas lógicas.
- **Circuito LSI (*Large Integration Scale*):** de 100 a 100.000 portas lógicas.
- **Circuito VLSI (*Very Large Integration Scale*):** > 100.000 portas lógicas.

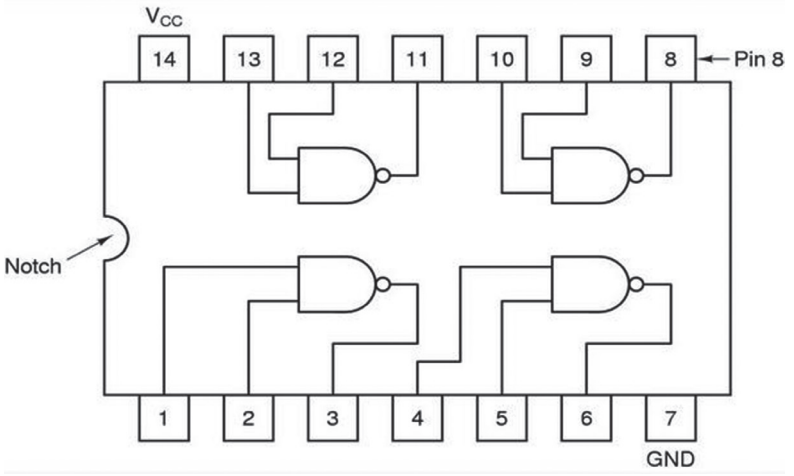


Figura 7. Chip SSI com 4 portas NAND.

Os circuitos lógicos dos sistemas digitais podem ser de dois tipos: circuitos combinacionais ou circuitos sequenciais. Um **circuito combinacional** é constituído por um conjunto de portas lógicas, as quais determinam os valores das saídas diretamente a partir dos valores atuais das entradas. Pode-se dizer que um circuito combinacional realiza uma operação de processamento de informação, a qual pode ser especificada por meio de um conjunto de equações booleanas. Cada combinação de valores de entrada pode ser vista como uma informação diferente, e cada conjunto de valores de saída representa o resultado da operação (GUNTZEL, [200-?]).

Um **circuito sequencial**, por sua vez, emprega elementos de armazenamento denominados ***latches*** e ***flip flops***, além de portas lógicas. Os valores das saídas do circuito dependem dos valores das entradas e dos estados dos *latches* ou *flip flops* utilizados. Como os estados dos *latches* e *flip flops* é função dos valores anteriores das entradas, diz-se que as saídas de um circuito sequencial dependem dos valores das entradas e do histórico do próprio circuito. Logo, o comportamento de um circuito sequencial é especificado pela sequência temporal das entradas e de seus estados internos (GUNTZEL, [200-?]). A seguir, você verá como obter as expressões booleanas geradas por um circuito lógico.

Dado o circuito apresentado na Figura 8, começamos por dividi-lo em portas lógicas básicas — nesse caso, temos uma porta AND (E1) e uma porta

OR (E2). A porta AND pode ser definida como E1 = A.B. Já na porta OR, uma de suas entradas é a saída de E1; logo, ela pode ser definida como E2 = E1 + C. Trocando as variáveis, obtemos então a expressão final: **F = (A.B) + C**.

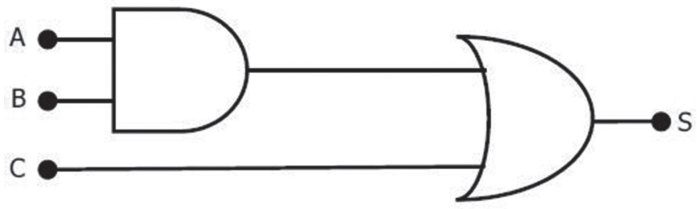


Figura 8. Exemplo de circuito (A.B) + C.

No próximo exemplo, descrito na Figura 9, temos duas portas AND e duas portas OR. A primeira porta AND tem três entradas (A, B e C); logo, temos A.B.C (E1). A porta OR tem duas entradas (A e B) e pode ser representada por A + B (E2). A segunda porta AND tem E2 como entrada e a entrada C; logo, a representação é (A + B).C (E3). A última porta (OR) tem duas entradas (E1 e E3); portanto, a sua representação é E1 + E3, ou seja, S = (A.B.C) + (A + B).C.

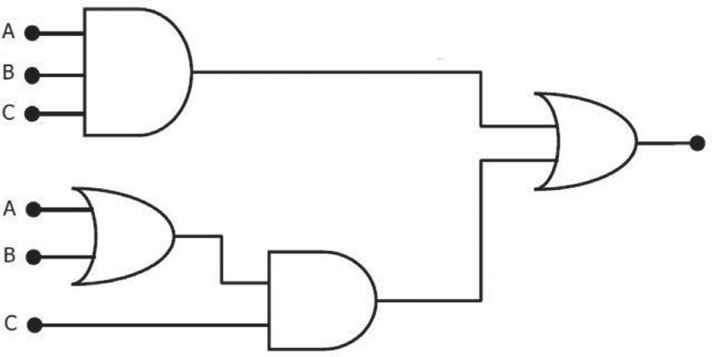


Figura 9. Exemplo de circuito (A.B.C) + (A + B).C.

O inverso também é bastante útil, isto é, fazer o circuito a partir de dada representação. Por exemplo, digamos que temos a representação S = (() + ())'. A forma mais fácil é dividir a expressão em blocos: () é o primeiro bloco; () o segundo bloco; S seria o bloco final. O primeiro e o segundo blocos são portas AND com o inversor, ou seja, portas NAND. O bloco final S é a porta NOR com duas entradas: primeiro e segundo blocos. A Figura 10 mostra o resultado.

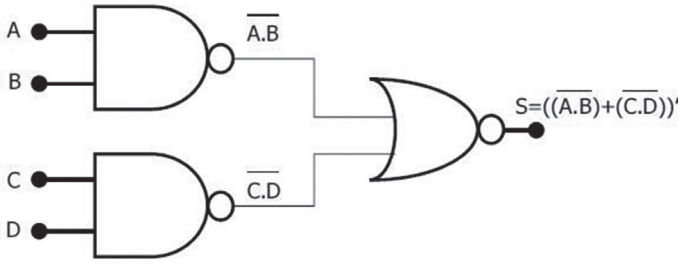


Figura 10. Resultado da expressão S = (() + ())'

Uma forma de estudar uma função booleana consiste em utilizar a sua tabela verdade. Como visto anteriormente, há uma equivalência entre o circuito lógico e a sua expressão característica:

- podemos obter um circuito a partir de sua expressão;
- podemos obter expressões a partir dos circuitos.

Uma tabela verdade representa o comportamento tanto do circuito, como de sua expressão característica. Considere a expressão: S = A.B.C + A.D + A.B.D; como são quatro entradas, temos $2^4 = 16$ possibilidades. Colocamos primeiro as possibilidades para cada variável, depois para cada expressão; no final, o resultado fica como mostra a Tabela 1.

Tabela 1. Verdade da expressão S = A.B.C + A.D + A.B.D.

A	B	C	D	A.B.C	A.D	A.B.D	S
0	0	0	0	0	0	0	**0**
0	0	0	1	0	0	0	**0**
0	0	1	0	0	0	0	**0**
0	0	1	1	0	0	0	**0**
0	1	0	0	0	0	0	**0**
0	1	0	1	0	0	0	**0**
0	1	1	0	0	0	0	**0**
0	1	1	1	0	0	0	**0**
1	0	0	0	0	0	0	**0**
1	0	0	1	0	1	0	**1**
1	0	1	0	0	0	0	**0**
1	0	1	1	0	1	0	**1**
1	1	0	0	0	0	0	**0**
1	1	0	1	0	1	1	**1**
1	1	1	0	1	0	0	**1**
1	1	1	1	1	1	1	**1**

Os blocos mais elementares da eletrônica são as portas lógicas, como vimos até agora. Vamos agora aplicar esse conhecimento para construir alguns blocos menos elementares — por exemplo, vamos fazer um circuito que implemente a soma de dois binários A e B. O problema dessa soma é quando A é 1 e B é 1: sua soma será 0 e vai 1. Como fazemos isso? Na Figura 11, temos a tabela verdade e o circuito, que chamamos **meio somador**.

Note que a saída do XOR corresponde à soma dos dois bits, enquanto a saída da porta AND corresponde ao transporte de bits, ou seja, “vai 1”.

A	B	Soma	Vai 1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

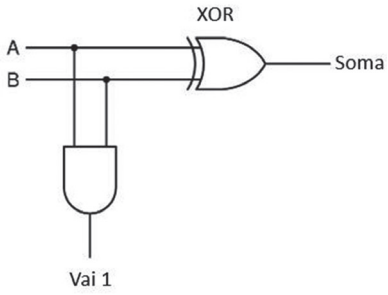


Figura 11. Circuito meio somador.

Link

Acesse o link a seguir para saber mais sobre somadores, codificadores e decodificadores.

https://goo.gl/usk8VE

Em conjunto com o conceito de *flip flops*, precisamos ter em mente também o conceito de lógica sequencial. De maneira simples, porém clara, circuitos sequências são aqueles que têm as saídas dependentes das variáveis de entrada e/ou de seus estados anteriores, que permanecem armazenados e que operam sob o comando de uma sequência de pulsos (*clocks*). Voltando aos *flip flops*, temos em seu circuito suas variáveis de entrada, uma entrada para o *clock* e duas saídas, normalmente denominadas Q e Q' (MINIPA, 2009).

Quando falamos de *clocks* e circuitos sequências, precisamos entender apenas um conceito muito simples: as saídas se alteram de acordo com a entrada apenas quando damos um pulso no *clock*. Como você já deve ter notado, os *flip flops* são circuitos sequências lógicos desenvolvidos para inúmeras aplicações, como o controle de alguma produção industrial: temos várias entradas, que devem funcionar de acordo com determinada lógica, para que a produção possa ser otimizada e nunca parar (MINIPA, 2009).

O *flip flop* mais básico é o RS. Nele temos duas saídas Q e Q', e as suas variáveis de entrada são um *Set* e um *Reset* — o *Set* seleciona o nível lógico 1 na saída do circuito, e o *Reset* seleciona o nível lógico 0 (Figura 12).

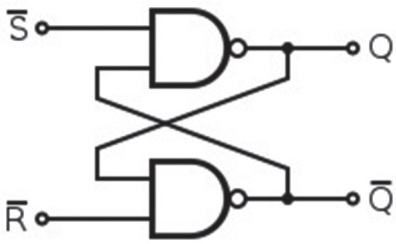


Figura 12. Circuito *flip flop* RS.

Link

Acesse o link a seguir para saber mais sobre *flip flops*.

https://goo.gl/SdaRll

O assunto é extenso, e aqui trouxemos o conhecimento básico com relação a portas lógicas e circuitos, mas você pode aprender muito mais com as referências trazidas neste texto, como o livro de Floyd (2007), o qual traz todo o material em detalhes e de forma bem didática.

Referências

FELGUEIRAS, C. A. *Portas lógicas e álgebra de boole*. [200-?]. Disponível em: <http://www.dpi.inpe.br/~carlos/Academicos/Cursos/ArqComp/aula_5bn1.html>. Acesso em: 9 abr. 2018.

FLOYD, T. L. *Sistemas digitais*: fundamentos e aplicações. 9. ed. Porto Alegre: Bookman, 2007.

GUNTZEL, L. J. *Circuitos combinacionais*. [200-?]. Disponível em: <https://www.inf.ufsc.br/~j.guntzel/isd/isd3.pdf>. Acesso em: 9 abr. 2018.

MINIPA. *Flip flops*. 2009. Disponível em: <http://escolaindustrial.com.br/escolaindustrial.com.br/Apostilas/M-1113a-1100-Aluno-Por.pdf>. Acesso em: 9 abr. 2018.

Leituras recomendadas

BARANAUSKAS, J. A. *Funções lógicas e portas lógicas*. 2012. Disponível em: <http://dcm.ffclrp.usp.br/~augusto/teaching/aba/AB-Funcoes-Logicas-Portas-Logicas.pdf>. Acesso em: 9 abr. 2018.

DEAECTO, G. S. *Circuitos lógicos*. 2013. Disponível em: <http://www.fem.unicamp.br/~grace/circuitos_combinacionais.pdf>. Acesso em: 9 abr. 2018.

ELETRÔNICA FÁCIL. Eletrônica digital 2 - funções e portas lógicas - porta lógica e - eletrônica fácil. *Youtube*, 4 ago. 2013. Disponível em: <https://www.youtube.com/watch?v=dnW293BodTo>. Acesso em: 9 abr. 2018.

Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.

Conteúdo:

SAGAH
SOLUÇÕES EDUCACIONAIS INTEGRADAS