Arquitetura e Organização de Computadores

Caros alunos, as videoaulas desta disciplina encontram-se no AVA (Ambiente Virtual de Aprendizagem).

UniFil

Unidade 3

Lógica Digital.

Introdução da Unidade

Olá, amigo(a) discente! Seja bem-vindo(a)!

Daremos mais um passo para o conhecimento dos sistemas computacionais. Na unidade passada, estudamos a forma que os sistemas computacionais representam a informação. Geralmente utiliza-se o sistema binário (0 e 1) e códigos.

Através do conceito da lógica binária podemos criar circuitos lógicos que tomam decisões coerentes, inteligentes e lógicas. Dessa forma, é importante aprender como descrever a funcionalidade dos circuitos.

Na aula 1 estudaremos os circuitos considerados mais básicos da lógica digital, as portas lógicas, que são componentes fundamentais a partir dos quais todos os circuitos lógicos são construídos. Estudaremos que a união de diferentes portas lógicas se constrói circuitos complexos que podem ser descritos e analisados pela álgebra booleana.

Na aula 2 aprenderemos a descrever os circuitos lógicos tendo em mãos a equação booleana do projeto, ou ao contrário, tendo um circuito em mãos e projetar a equação que represente o projeto lógico. Por fim, teremos um pouco de noção sobre as regras que integram a álgebra booleana. Essa álgebra é uma importante ferramenta, pois produz uma relação de entrada e saída para o projeto em si.

Existe muito conteúdo para você estudante, caso queira se aprofundar em todas as unidades que está lendo, tendo em vista que o nosso material cobre apenas os fundamentos. Porém, analisando as leituras obrigatórias da nossa bibliografia você entenderá muito bem o assunto. Bons estudos!

Objetivos

- Implementar circuitos lógicos usando portas lógicas.
- Criar através de ferramentas circuitos digitais e, fazer uma análise do comportamento de circuitos.

Conteúdo Programático

Aula 1: Portas Lógicas.

Aula 2: Descrevendo circuitos lógicos.

Você poderá também assistir às videoaulas em seu celular! Basta apontar a câmera para os **QRCodes** distribuídos neste conteúdo.

Pode ser necessário instalar um aplicativo de leitura QRcode no celular e efetuar login na sua conta Gmail.

Aula 1 Portas Lógicas

Seja bem-vindo(a), querido(a) aluno(a)! Estamos aqui juntos novamente para desvendarmos esse mundo digital, em especial, o computador. Nas unidades anteriores estudamos o funcionamento de alguns aspectos fundamentais da computação, como a arquitetura de um computador e suas representações de dados.

Vamos agora um pouco mais adiante ao nosso conhecimento, vamos aprofundar ao nível mais fundo na hierarquia de níveis do computador. Para que os computadores processem informações, eles se utilizam de elementos primitivos denominados portas (*Gates)* que são combinados de inúmeras maneiras para formar circuitos.

Portas lógicas são os circuitos mais básicos existentes, elas representam os blocos fundamentais a partir dos quais todos os outros circuitos lógicos e sistemas digitais são construídos.

Portas lógicas

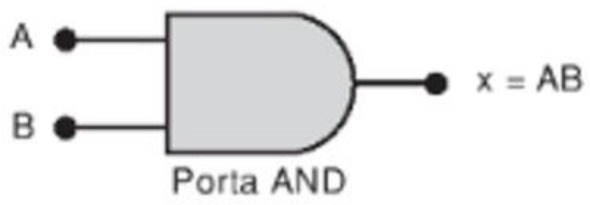
Vamos começar pela porta lógica AND. A porta AND é uma das portas básicas que pode ser combinada para formar qualquer função lógica. Uma porta AND pode ter duas ou mais entradas e realizar uma operação conhecida como multiplicação lógica (FLOYD, 2007).

Analisando a figura 1 abaixo podemos tirar algumas conclusões importantes para o nosso estudo.

AND

A	B	$x = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

(a)



(b)

Figura 1. a) tabela verdade para a operação AND; b) Símbolo da porta AND.

Fonte: TOCCI (2008).

Na figura 1.0(a) é mostrada uma tabela verdade. Tabela verdade é uma forma prática de mostrar as saídas de algum circuito lógico. Na tabela verdade da figura, repare o que acontece quando duas entradas lógicas, A e B, são combinadas usando a operação AND para gerar a saída x. Essa tabela mostra que x será nível lógico 1 apenas quando A e B forem 1. Para qualquer outro caso em que uma das entradas for 0, a saída será 0.

Na saída da tabela é gerada a seguinte expressão: **x = A.B**. Essa expressão matemática é conhecida como **expressão booleana.** Recebe esse nome, pois foi *George Boole* um matemático que publicou em 1854 os conceitos da lógica digital, onde objetos podem assumir apenas dois valores: verdadeiro ou falso.

Os circuitos digitais aplicam a ideia de *Boole*, onde máquinas usam voltagens altas e baixas para designar verdadeiro ou falso. É comum então para os circuitos interpretar o valor digital 0 como falso e o valor digital 1 como verdadeiro.

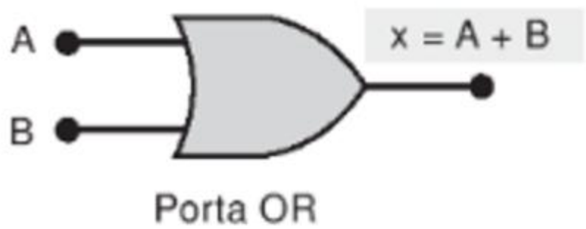
Porta OR

Uma porta OR produz um nível ALTO na saída quando qualquer das entradas for nível ALTO, observe a figura 2 abaixo. A saída será nível BAIXO apenas quando todas as entradas estiverem em nível BAIXO. Portanto, uma porta OR determina quando uma ou mais de suas entradas estiverem em nível ALTO e produz um nível ALTO na saída dela para indicar essa condição.

OR

A	B	$x = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

(a)



(b)

Figura 2. (a) Tabela verdade que define a operação OR; (b) Símbolo da porta OR.

Fonte: TOCCI (2008).

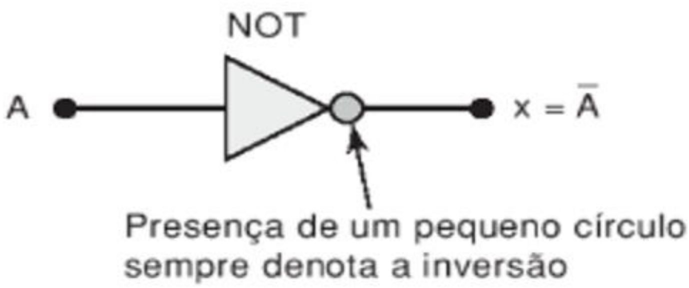
Porta NOT

O inversor (circuito NOT) realiza a operação denominada inversão ou complementação. O inversor troca um nível lógico para o nível lógico oposto, veja figura 3 abaixo. Em termos de bit, ele troca 1 por 0 e 0 por 1(FLOYD, 2007).

NOT

A	$x = \overline{A}$
0	1
1	0

(a)



(b)

Figura 3. (a) Tabela verdade que define a operação NOT; (b) Símbolo da porta NOT

Fonte: TOCCI (2008).

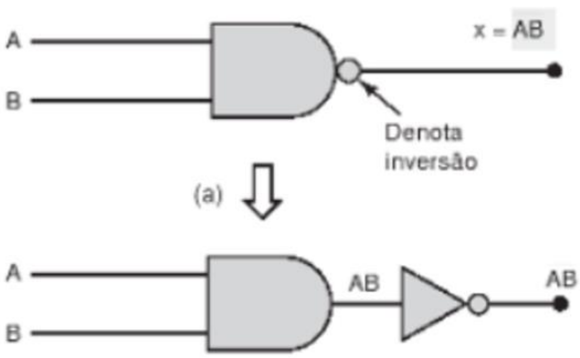
Videoaula 1

Utilize o QRcode para assistir!

Assista agora à videoaula para entender melhor o assunto sobre as portas AND, OR e NOT.

Porta NAND

O termo NAND é uma contração de NOT-AND e implica numa função AND com uma saída complementada (invertida). Uma porta NAND produz uma saída de nível BAIXO apenas quando todas as entradas estiverem em nível ALTO. Quando qualquer uma das entradas for nível BAIXO, a saída será nível ALTO (FLOYD, 2007).



		AND	NAND
A	B	AB	$\overline{AB}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

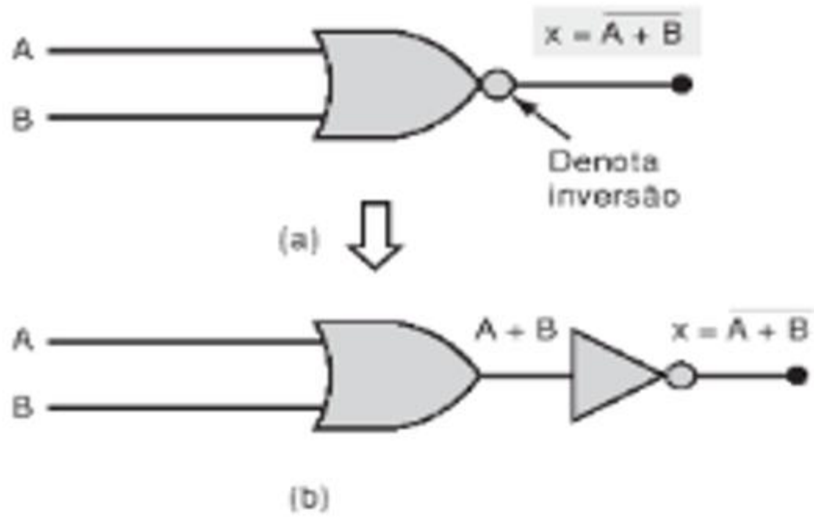
(c)

Figura 4: (a) Símbolo da porta NAND; (b) Circuito equivalente; (c) Tabela-verdade porta NAND.

Fonte: TOCCI (2008).

Porta NOR

Uma porta NOR produz uma saída de nível BAIXO quando qualquer uma de suas entradas for nível ALTO. Apenas quando todas as suas entradas estiverem em nível BAIXO é que a saída será nível ALTO. Para o caso específico de uma porta NOR de 2 entradas, conforme mostra a Figura 5 com as entradas identificadas por A e B e a saída identificada por X (FLOYD, 2007).



		OR	NOR
A	B	A + B	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

(c)

Figura 5. (a) Símbolo porta NOR; (b) Circuito equivalente; (c) Tabela verdade da porta NOR.

Fonte: TOCCI (2008).

Videoaula 2

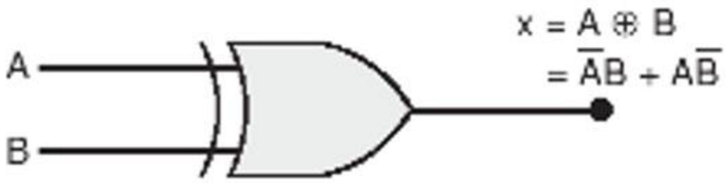
Utilize o QRcode para assistir!

Assista agora à videoaula para entender melhor o assunto sobre as portas NAND e NOR.

Porta EX-OR e EX-NOR

As portas OR exclusivo (EX-OR) e NOR exclusivo (EX-NOR) são formadas pela combinação de outras portas já estudadas acima. Entretanto, devido à importância fundamental dessas portas em muitas aplicações, elas são tratadas como elementos lógicos básicos tendo seus próprios símbolos lógicos.

Para uma porta ***EX-OR***, veja a figura 6 abaixo, a saída X é nível ALTO quando a entrada A for nível BAIXO e a entrada B for nível ALTO ou quando a entrada A for nível ALTO e a entrada B for nível BAIXO; a saída X é nível BAIXO quando A e B forem ambas nível ALTO ou ambas nível BAIXO (FLOYD, 2007).



A	B	x
0	0	0
0	1	1
1	0	1
1	1	0

Figura 6. Símbolo e tabela verdade da porta XOR.

Símbolos para porta XOR / Tabela verdade da porta XOR

Fonte: TOCCI (2008).

Agora nesse momento é importante mostrar uma aplicação direta da porta lógica EX-OR. Uma porta EX-OR pode ser usada como um somador de dois bits. Lembre-se, que as regras básicas para a adição binária são as seguintes: 0 + 0 = 0, 0 + 1 = 1, 1 + 0 = 1 e 1 + 1 = 10 (sem o *carry)*.

Um exame na tabela-verdade de uma porta EX-OR nos mostra que a sua saída é o resultado da soma binária dos dois bits das entradas, conforme mostra a figura 7.

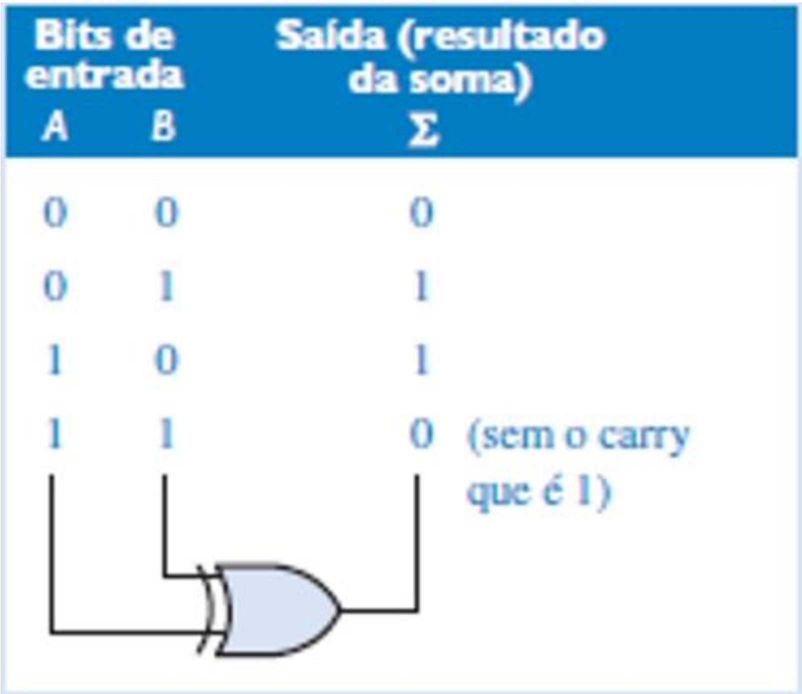
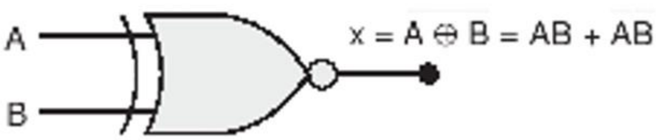


Figura 7. Uma porta EX-OR usada como um somador de dois bits.

Curiosidades!

As portas EX-OR conectadas para formar um circuito somador permite a um computador realizar adição, subtração, multiplicação e divisão em uma unidade lógica e aritmética (ALU – *arithmetic logic unit*). Uma porta EX-OR combina as lógicas básicas AND, OR e NOT.

Vamos para outra porta exclusiva agora. Na porta ***EX-NOR***, a saída X é nível BAIXO quando a entrada A for nível BAIXO e a entrada B for nível ALTO, ou quando A for nível ALTO e B for nível BAIXO. A saída X é nível ALTO quando A e B estiverem ambas em nível ALTO ou ambas em nível BAIXO.



A	B	x
0	0	1
0	1	0
1	0	0
1	1	1

Figura 8. Uma porta EX-NOR e sua Tabela verdade.

Símbolos para porta XNOR / Tabela verdade da porta XNOR.

Fonte: TOCCI (2008).

Videoaula 3

Utilize o QRcode para assistir!

Assista agora à videoaula para entender melhor o assunto sobre as portas EX NOR e EX OR.

Que tal um exemplo com a porta EX-NOR? Vamos lá então? Leia com atenção o enunciado do seguinte projeto:

> *Projete um circuito lógico que permita a passagem de um sinal para a saída apenas quando uma entrada, mas não ambas, for nível ALTO; caso contrário, a saída permanecerá em nível ALTO."*
>
> *TOCCI, 2008*

O resultado é mostrado na figura 9. Uma porta OR é utilizada porque estando a porta A em nível baixo, a saída estará em nível ALTO quando as demais entradas estiverem em níveis IGUAIS. E se B e C tiveram níveis diferentes a saída na porta OR será baixa. As portas B e C estão combinadas em uma porta XNOR.

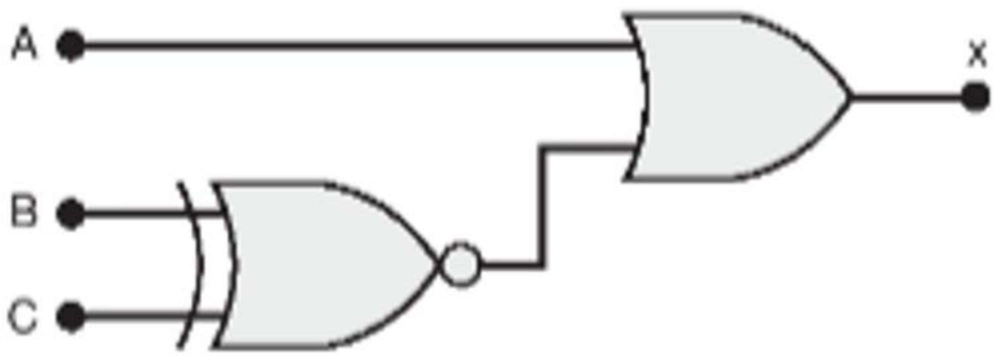


Figura 9. Resultado do exemplo.

Fonte: TOCCI (2008).

Então, queridos(as) alunos(as), conforme vimos nos exemplos acima, de uma simples combinação de portas lógicas conseguimos criar alguns circuitos que tenham alguma função computacional. É interessante isso, não é?

Seria interessante também agora dar um passo a mais no conhecimento dos circuitos. Atente-se para o seguinte projeto, que é descrito em Tocci (2008):

Projete um circuito lógico usando entradas x1, x0, y1 e y0 cuja saída será nível ALTO apenas quando dois números binários, x1 x0, y1 y0 forem iguais."

TOCCI, 2008

Geralmente em projetos de circuitos digitais usam-se algumas técnicas para se chegar a valores de saída de um circuito lógico. Umas das técnicas de construção de circuitos é o uso da tabela verdade, cuja função é mostrar todas as possiblidades de combinações de valores de entrada para as variáveis envolvidas.

Para nosso projeto as variáveis de entrada são: x_1x_0, y_1y_0 *e* sua saída podemos denominar de Z. Nota-se que temos então 4 variáveis de entrada que pode gerar 16 combinações pois 2^4=16. Desenvolvendo a tabela verdade teremos:

x_1	x_0	y_1	y_0	z (Saída)
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Tabela 1. Tabela verdade do projeto proposto.

Como dito, a saída Z tem que ser alto sempre que os valores de x_1x_0, y_1y_0 forem iguais, ou seja, $x_1=y_1$ e $x_0=y_0$. A tabela mostra que existem quatro casos desse tipo. Após o desenvolvimento da tabela verdade, teríamos outros passos para obter o circuito:

a. Obtenção da expressão para a saída Z.

b. Simplificação da expressão.

c. Construção do circuito.

Nosso intuito até aqui é indicar as fases da obtenção de um circuito lógico, os detalhes irão ser mostrados em aula posterior. O importante nesse momento, é você se atentar apenas para as etapas de desenvolvimento. Então, o circuito para o nosso projeto é mostrado na figura 10 abaixo:

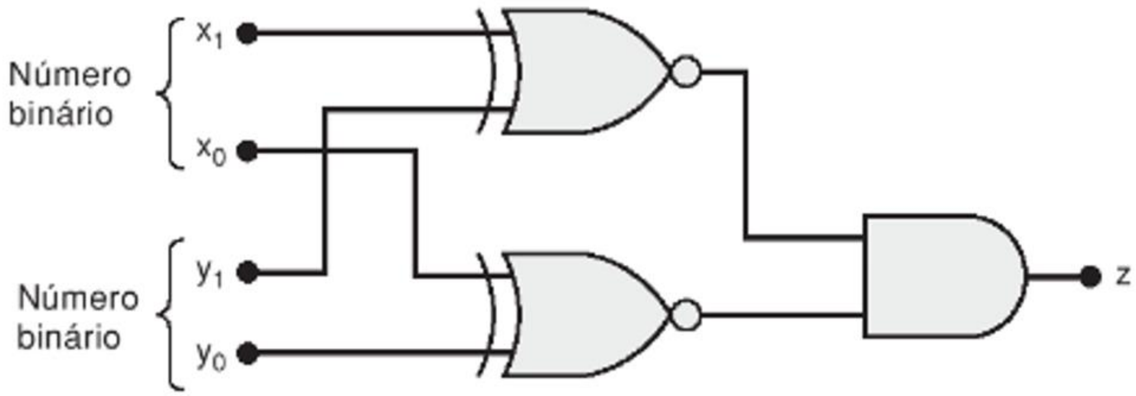


Figura 10. Circuito do projeto.

Fonte: TOCCI (2008).

Indicação de Vídeo

Assista a esse vídeo onde explica sobre as portas lógicas, Circuitos lógicos e tabela verdade.

Disponível em: <https://www.youtube.com/watch?v=aYVz0l3ZMWc>. Acesso em: 20.06.2020.

25 min.

Indicação de Leitura

Para complementar seus estudos até aqui, leia o livro de TOCCI os itens 3.1 a 3.5. Nessas páginas estão um bom conteúdo para o aprendizado sobre portas lógicas. E também o item 4.9.

Disponível em:
<https://plataforma.bvirtual.com.br/Leitor/Publicacao/740/pdf/142?code=cXI5AdU0iFMMzkwYOMqk49T6np/R3q5Zh28HdlA11vjVBa3CW8hK93iZ2FUYcHxk7scZeiZpNEsOI+QtuLDlFw==>. Acesso em: 20.06.2020.

Simulação de circuitos Digitais

Geralmente quando se estuda eletrônica, a vontade de montar um circuito e vê-lo funcionando é inerente ao projetista. No nosso caso, também não fugimos a esta regra, por isso convém aqui, querido(a) aluno(a), estudarmos também algumas ferramentas de simulação de circuitos.

Essas ferramentas possibilitam aos estudantes, respostas rápidas a resultados que apenas seriam obtidos após as montagens com componentes e análises utilizando complexas ferramentas. Então, estudando tais ferramentas você poderia testar os nossos exemplos que estão nesse material.

Vamos começar mostrando um software gratuito e fácil de usar. O *Logisim* (figura: 11) é uma ferramenta educacional para a concepção e a simulação digital de circuitos lógicos.

Tem uma interface simples e contêm ferramentas para simular circuitos à medida em que são construídos, é simples o bastante para facilitar a aprendizagem dos conceitos mais básicos relacionados aos circuitos lógicos.

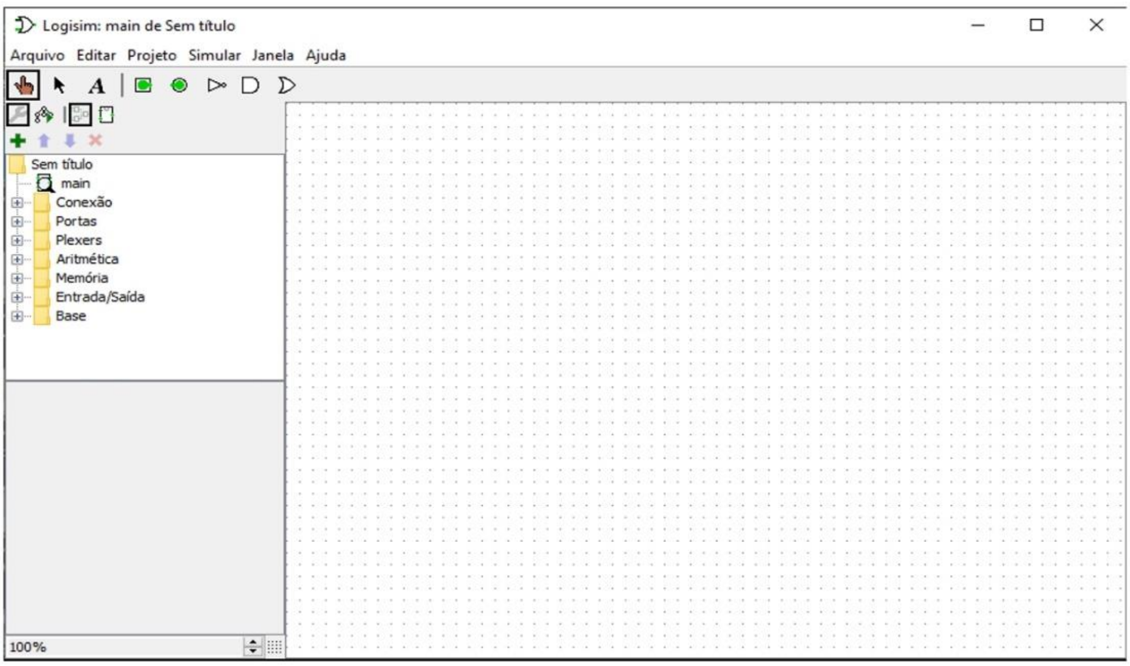


Figura 11. Tela do ***Logisim.***

Fonte: elaborado pelo próprio autor.

Curiosidades!

Para saber mais sobre o software *Logisim* acesse o seguinte link:

Disponível em: <http://www.cburch.com/logisim/pt/index.html>. Acesso em: 20.06.2020.

Analisando-se a figura 11, no seu lado esquerdo, percebemos que a ferramenta possui um conjunto de ferramentas que permite ao usuário interagir com um circuito clicando-o e arrastando-o com o *mouse* na área de desenho. A ferramenta pode ser baixada gratuitamente no endereço acima desse material. Teste o software refazendo os circuitos apresentados nesse material e veja os seus resultados.

Indicação de Vídeo

Assista a esse vídeo onde é mostrado um tutorial básico que te ensina como montar um esquema simples de circuito eletrônico a partir de uma expressão booleana com o auxílio do programa LOGISIM.

Disponível em: <https://www.youtube.com/watch?v=TdDCWG2inoY>. Acesso em: 20.06.2020. - 13 mim.

Outro importante software para simulação de circuitos digitais é o ***DEEDS** (Digital Electronics Education and Design Suite).* Trata-se também de um conjunto de ferramentas educacionais para a eletrônica digital, onde possibilita o estudo desde portas lógicas a sistemas microprocessados completos, em diferentes níveis de abstração.

O usuário interage com um circuito clicando e arrastando os componentes com o mouse na área de desenho. Contém ferramentas básicas e avançadas

A ferramenta DEEDS é mostrada na figura 12 abaixo.

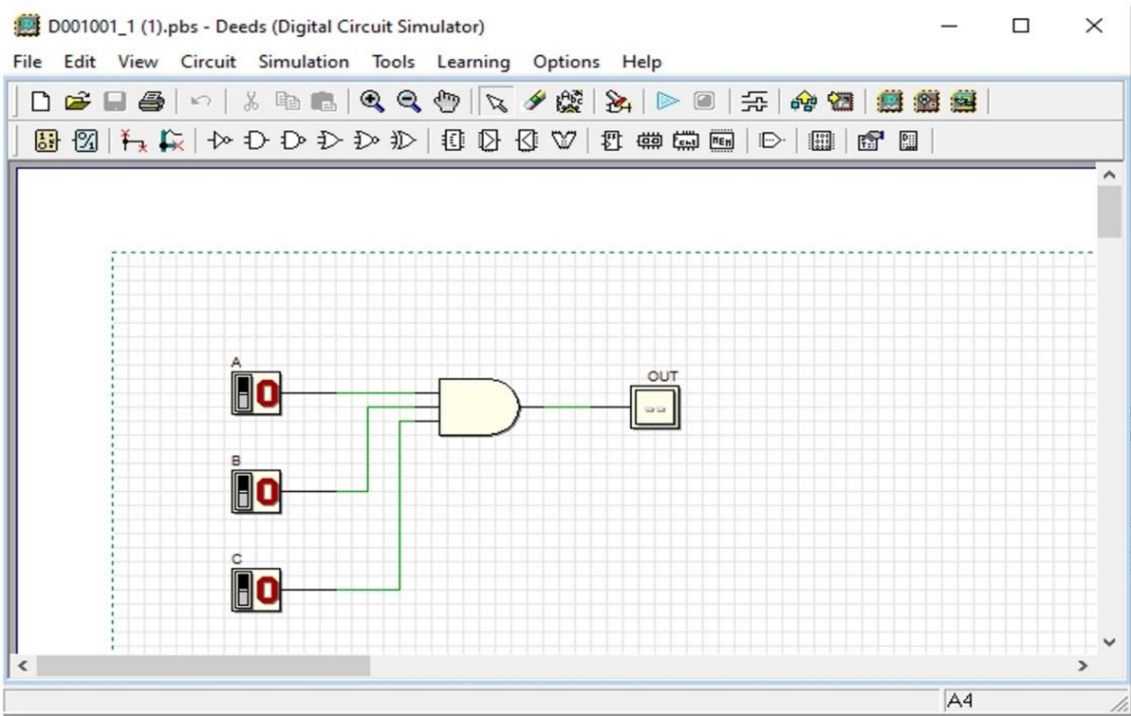


Figura 12. Ferramenta DEEDS.

Fonte: elaborado pelo próprio autor.

Com esses simuladores podem-se vivenciar experiências bem próximas da realidade, ajudando você, estudante da área computacional, a entender como funcionam as máquinas.

Curiosidades!

Para saber mais sobre o software *DEEDS* acesse o seguinte link:

Disponível em: <https://www.digitalelectronicsdeeds.com/>. Acesso em: 20.06.2020.

Aula 2 Descrevendo circuitos lógicos

Opa! Estamos aqui de novo, queridos(as) alunos(as). Na aula anterior estudamos os componentes mais elementares da lógica digital que são as portas lógicas. Obviamente que as portas lógicas isoladas não podem fazer muita coisa, mas pense em milhões delas juntas. Combinadas de uma forma lógica formam circuitos importantes para os nossos computadores.

Escrevendo expressões a partir de circuitos

Que tal combinarmos algumas portas nesse momento? Na aula anterior tivemos alguns exemplos dessa combinação, onde formamos pequenos circuitos que tinha alguma função prática. Também notamos que os circuitos geram uma expressão lógica que representam de forma algébrica o sistema.

Então nesse tópico vamos descrever algebricamente um circuito lógico. Repare no primeiro exemplo mostrado na figura 1 abaixo.

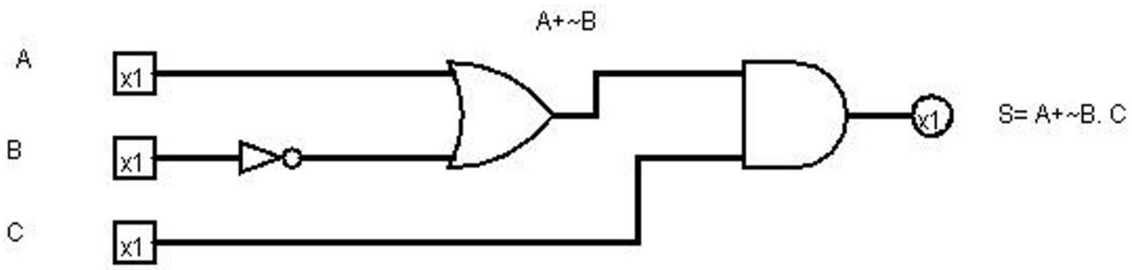


Figura 1. Circuito lógico com sua expressão booleana.

Fonte: elaborado pelo próprio autor.

Analisando a figura 1, observamos um circuito que contém três entradas: A, B, C e uma saída S. Nota-se ainda que a entrada B é invertida através da porta NOT (~). Podemos então retirar desse circuito uma expressão lógica que o represente.

É comum começar a análise da esquerda para direita do circuito. A expressão para a saída da porta OR é (A+~B). Esta saída está conectada a uma porta AND juntamente com C, que é a outra entrada do circuito. A porta AND opera sobre as entradas de modo que a saída seja o resultado da operação AND sobre as entradas. Dessa forma, podemos expressar a saída da porta AND como sendo **S=A+~B.C**.

Com essa equação desenvolvida, torna-se mais prático montar a tabela verdade, a qual podemos acompanhar o comportamento do circuito. Veja a tabela 01.

a	b	c	**S=A+~B.C**
0	0	0	**0**
0	0	1	**1**
0	1	0	**0**
0	1	1	**0**
1	0	0	**0**
1	0	1	**1**
1	1	0	**0**
1	1	1	**1**

Tabela 1. Tabela verdade para a expressão **S=A+~B.C.**

Fonte: elaborado pelo próprio autor.

De repente um exemplo apenas é pouco para uma aprendizagem efetiva. Então vamos para outro exemplo, considere o circuito da figura 2.

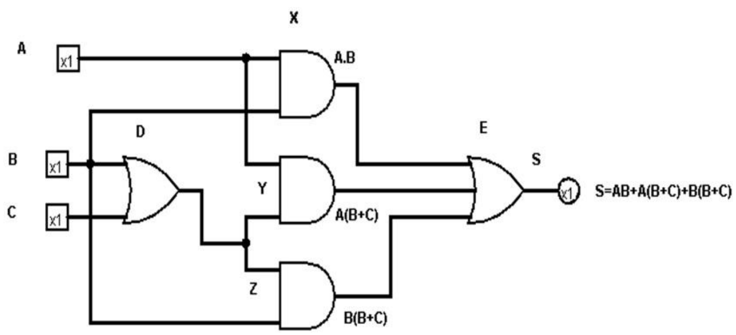


Figura 2. Circuito lógico com sua expressão booleana.

Fonte: elaborado pelo próprio autor.

Notamos a existência de três entradas para o circuito da figura 2: A, B, C. Existindo também uma saída S. Analisando da esquerda para direita encontramos a porta OR (D) que opera com as variáveis B e C, que depois se liga às portas AND assinaladas como Y e Z. Temos também a outra entrada do circuito que é A, cuja entrada está conectada a outra porta AND assinalada como X. As saídas das portas AND´s X, Y, Z irão se ligar a entrada da porta OR (E). Dessa forma lógica, será obtida a seguinte expressão booleana para o circuito **S=AB+A (B+C) +B (B+C).**

Então, como anteriormente, podemos montar uma tabela verdade para o circuito 2. Veja a tabela 2 abaixo.

a	**b**	**c**	**S=AB+A (B+C) +B (B+C)**
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Tabela 2. Tabela verdade para a expressão **S=AB+A (B+C) +B (B+C).**

Fonte: elaborado pelo próprio autor.

Aproveite esse trecho do nosso estudo e teste o circuito da figura 2 nos softwares apresentados na aula anterior, pois dessa maneira você poderá testar e tirar suas próprias conclusões a respeito dos circuitos apresentados.

Até aqui extraímos equações algébricas de circuitos lógicos, agora como seria se tivéssemos uma equação e quiséssemos montar um circuito? Em projetos de engenharia muitas vezes temos uma equação algébrica desenvolvida para um determinado projeto. Esse é o nosso próximo assunto.

Videoaula 1

Utilize o QRcode para assistir!

Assista agora à videoaula para entender melhor o assunto: Descrevendo circuitos lógicos.

Implementando circuitos a partir de expressões booleanas

Quando a operação de um circuito lógico está definida por meio de equações booleanas, podemos desenvolver o circuito com base nessas equações. Se precisarmos de um circuito definido por **S= A.B.C**, por exemplo, saberemos que precisamos de uma porta AND de três entrada.

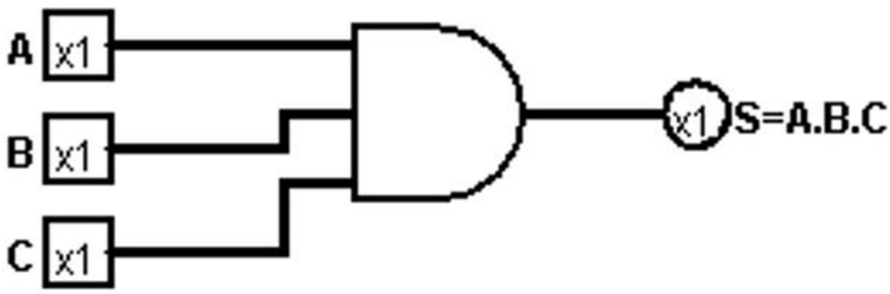


Figura 3. Porta AND com 3 entradas.

Fonte: elaborado pelo próprio autor.

Esse caso simples apresentado na figura 2.2 pode ser estendido a circuitos mais complexos.

Vejamos o circuito que é definido pela equação $S = A\bar{B} + \bar{A}B$. Note nessa equação o uso do sinal em cima das variáveis A e B. Esses sinais também indicam a negação das mesmas.

A expressão contêm dois termos($A\bar{B}$, $\bar{A}B$) sobre os quais se aplicam a operação OR(+). Essa expressão nos indica que é necessário o uso de uma porta OR de duas entradas que equivalem a $A\bar{B}$ e $\bar{A}B$, isto é mostrado na figura 4 abaixo.

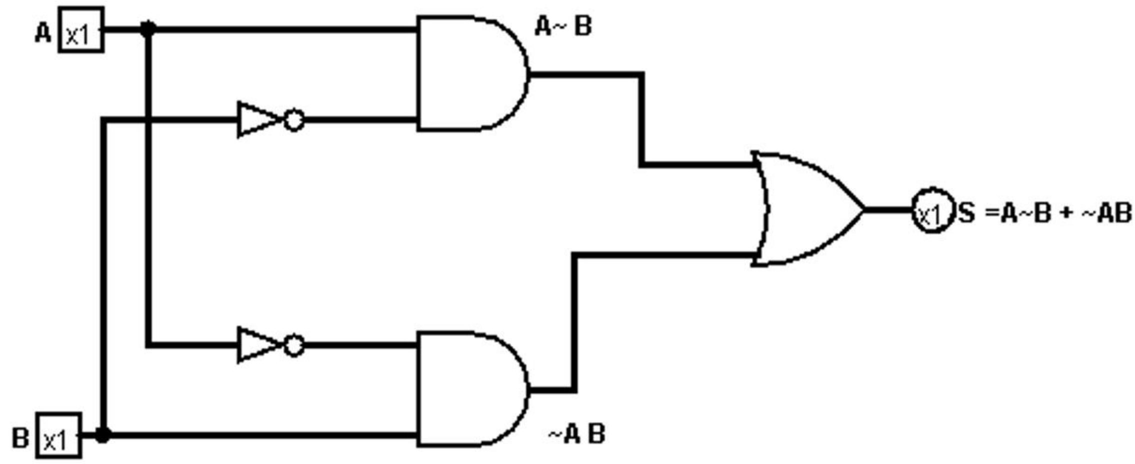


Figura 4. Circuito lógico a partir da expressão lógica.

Fonte: elaborado pelo próprio autor.

Cada entrada da porta OR tem um termo que é um produto lógico AND. Observe que as portas NOT geram os termos $\sim A$ ou $\bar{A}$ e $\sim B$ ou $\bar{B}$.

Videoaula 2

Utilize o QRcode para assistir!

Assista agora à videoaula para entender melhor o assunto: Implementando circuitos a partir de expressões booleanas.

Noções da Álgebra Booleana

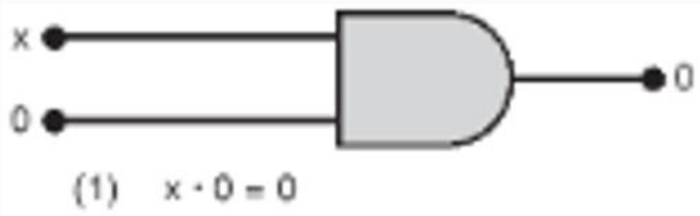
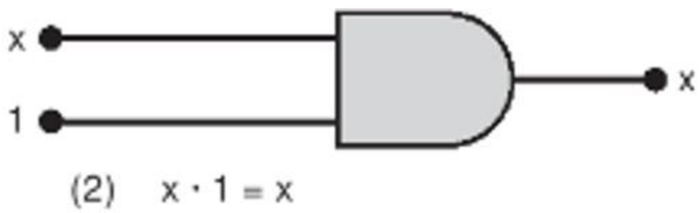
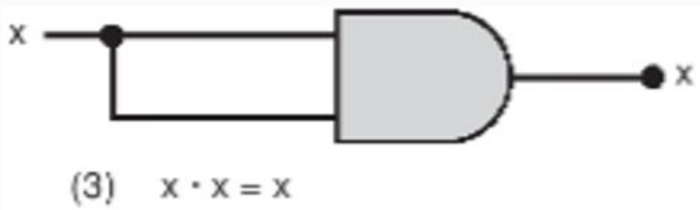
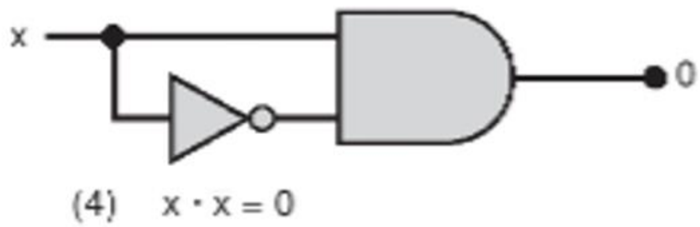
Para Floyd (2007), a álgebra Booleana é a matemática dos sistemas digitais. Essa álgebra além de lidar com variáveis, também faz operações lógicas sobre essas mesmas variáveis, ou seja, da mesma forma que podemos manipular e simplificar expressões aritméticas, podemos também manipular e simplificar as expressões lógicas.

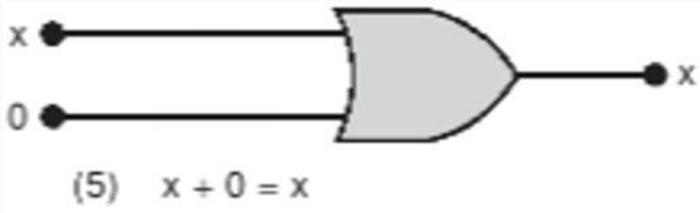
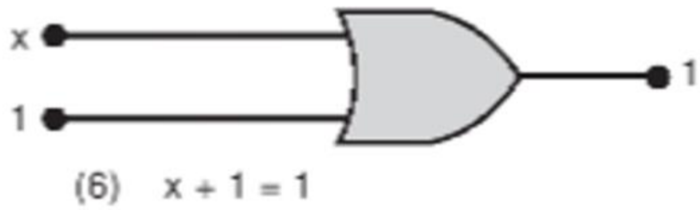
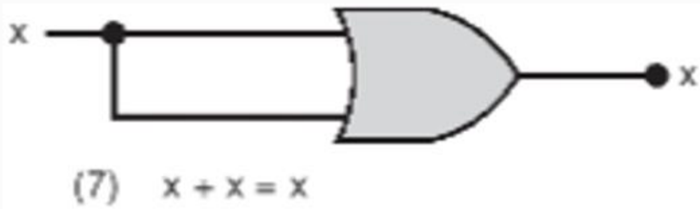
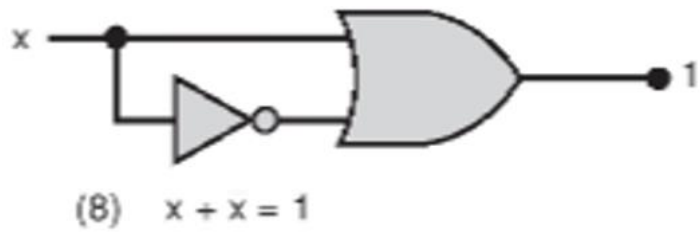
Continuaremos o nosso estudo agora, estudando os vários teoremas que são conhecidos como teoremas booleanos. Estes teoremas booleanos podem ajudar a simplificar expressões lógicas e circuitos.

Em Tocci (2008, p.63), é apresentado dois grupos de teorema: os que utilizam uma variável e o outro que utiliza mais de uma variável. Vamos primeiro estudar os teoremas com apenas uma variável que está evidenciado na tabela 3 abaixo.

Essa tabela contém cada teorema enumerados de 1 a 8 com seu respectivo circuito que demonstra a sua validade. Nessa tabela, a variável x é uma variável lógica que pode ser 0 e 1. Observe abaixo:

Tabela 03. Teoremas para uma única variável.

<table>
<tr><td>
</td><td>Se for realizada uma operação AND de qualquer variável com 0, o resultado tem que ser 0.</td></tr>
<tr><td>
</td><td>Se for realizada uma operação AND de qualquer variável com 1, o resultado tem que ser a variável em questão.</td></tr>
<tr><td>
</td><td>A operação AND de uma variável com ela mesma é sempre igual a variável.</td></tr>
<tr><td>
</td><td>A operação AND de uma variável e o seu complemento é sempre igual a 0.</td></tr>
</table>

	A operação OR de uma variável com 0 é sempre igual a variável.
	A operação OR da variável com 1 é igual a 1.
	A operação OR da variável com ela mesma é sempre igual a variável.
	A operação OR da variável com o seu complemento é sempre igual a 1.

Fonte: TOCCI (2008, p.63).

Pode-se notar pela tabela 3, que os 8 teoremas trataram apenas de uma variável. Vamos aprender os outros teoremas que usam mais de uma variável (TOCCI, 2008, p.63). Observe os teoremas abaixo.

$$
\begin{array}{ll}
(9) & x + y = y + x \\
(10) & x \cdot y = y \cdot x \\
(11) & x + (y + z) = (x + y) + z = x + y + z \\
(12) & x(yz) = (xy)z = xyz \\
(13a) & x(y + z) = xy + xz \\
(13b) & (w + x)(y + z) = wy + xy + wz + xz \\
(14) & x + xy = x \\
(15a) & x + \overline{x}y = x + y \\
(15b) & \overline{x} + xy = \overline{x} + y
\end{array}
$$

O teorema 9 e 10 são chamados de leis comutativas, pois a ordem das variáveis não importa; o resultado é o mesmo. Os teoremas 11 e 12 são as leis associativas que nos falam que podemos agrupar a variáveis em expressões AND e OR de modo que desejarmos.

O teorema 13 é a lei distributiva, onde permite que uma expressão pode ser expandida multiplicando termo a termo. Ainda nesse teorema podemos fatorar (colocar em evidencia termos semelhantes).

Os teoremas 14 e 15 não são triviais, porém podem ser demonstrados testando as possiblidades para x e y. Vamos então, a título de exemplo, provar o teorema 14. Acompanhe o raciocínio abaixo.

$$
\begin{aligned}
x + xy &= x(1 + y) \\
&= x \cdot 1 && \text{[usando o teorema (6)]} \\
&= x && \text{[usando o teorema (2)]}
\end{aligned}
$$

Então, como pode-se notar, primeiramente é fatorado a expressão. Após isso, é apenas aplicado os teoremas 6 e 2. Interessante não é mesmo?

Agora você deve estar pensando, mas por que aprender tanto teorema? Essa dúvida é normal. Mas explicando de uma maneira simples, esses teoremas servem para simplificar as expressões que por consequência simplifica circuitos tornando-os mais baratos.

Nada como um exemplo para tirar dúvidas. Repare na equação **Y=AB+A (B+C) +B (B+C)**. Essa equação não está simplificada, tornando assim o circuito grande. Usando os teoremas acima apresentados veremos que podemos simplificá-la deixando-a mais reduzida.

Y=AB+A (B+C) +B (B+C)

AB+AB+AC+BB+BC ➔ distributiva

AB+AC+B+BC ➔ Teorema 7 (AB+AB) e 3(B.B)

B (1+A+C) +AC ➔ Teorema 6 (B (1+A+C))

B1 + AC ➔ Teorema 2 (B.1)

B +AC ➔ Equação reduzida

Indicação de Leitura

Para complementar seus estudos até aqui, leia o livro de TOCCI os itens 3.6 a 3.10. Nessas páginas estão um bom conteúdo sobre implementação de circuitos lógicos.

Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/740/pdf/142?code=cXI5AdU0iFMMzkwYOMqk49T6np/R3q5Zh28HdlA11vjVBa3CW8hK93iZ2FUYcHxk7scZeiZpNEsOI+QtuLDlFw==>. Acesso em: 20.06.2020.

Desenvolvendo circuitos a partir de problemas da vida real

É normal pensarmos quando estamos estudamos alguma disciplina, onde é a aplicação real dos conhecimentos. Nesse tópico vamos apresentar um exemplo prático de como desenvolver um sistema lógico utilizando os conceitos da álgebra booleana.

Para Tocci (2008), qualquer projeto lógico pode ser desenvolvido seguindo algumas etapas:

1. Interprete o problema e construa uma tabela-verdade para descrever o seu funcionamento.
2. Escreva o **termo AND** (produto) para cada caso em que a saída seja 1.
3. Escreva a expressão da **soma de produtos** para a saída.
4. Simplifique a expressão de saída, se possível.
5. Implemente o circuito para a expressão final, simplificada.

Não se preocupe com alguns termos novos que surgiram no nosso texto aqui, fazendo-se uma leitura das nossas bibliografias você estará apto a entender perfeitamente.

Leia atentamente o seguinte projeto:

Suponha–se que desejamos instalar um ar condicionado em uma sala. O ar condicionado funciona nas seguintes condições:

- O aparelho pode ser ligado e desligado manualmente através de um interruptor;
- O aparelho ligará automaticamente, quando o interruptor estiver desligado e o termostato detectar que a temperatura exterior ultrapassou o limite de 30° graus;
- Existirá um sensor que desligará o aparelho quando o interruptor estiver ligado e alguma porta do ambiente estiver aberta.

Adotaremos as seguintes especificações:

- **Interruptor**: 0 apagado e 1 ligado.
- **Termostato**: indica 0 para menor do que 30°C e 1 para maior do que 30°C.
- **Porta**: 0 para fechada e 1 para aberta.
- Também vamos considerar**: A=Interruptor, B=Termostato, C=Porta, X= Saída.**
- Nível 0 na saída= **desligado**. Nível 1 na saída= **ligado**.

Após interpretação do problema vamos à construção da tabela verdade. Com base no enunciado a saída X será valor alto apenas em três ocasiões; quando o interruptor estiver desligado, temperatura acima de 30°C e porta fechada; quando interruptor ligado e temperatura abaixo de 30°C e porta fechada; quando interruptor ligado e temperatura abaixo acima 30°C e porta fechada.

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

1	0	1	0
1	1	0	1
1	1	1	0

Tabela 4. Tabela verdade do projeto.

Com o desenvolvimento da tabela verdade, colocamos agora o termo AND que é conhecido como *soma dos produtos para* cada caso onde a saída é 1.

A	**B**	**C**	**X**	
0	0	0	0	
0	0	1	0	
0	1	0	1	$\bar{A}B\bar{C}$
0	1	1	0	
1	0	0	1	$A\bar{B}\bar{C}$
1	0	1	0	
1	1	0	1	$AB\bar{C}$
1	1	1	0	

Tabela 5. Tabela verdade do projeto.

Com os termos AND definidos, podemos escrever a expressão da **soma de produtos** para a saída. No caso será X=

$$\bar{A}B\bar{C} + A\bar{B}\bar{C} + AB\bar{C}$$

Estamos quase lá! Com a expressão do projeto lógico desenvolvido podemos tentar simplificá-la para deixá-la otimizada. Para isso, vamos fazer simples manipulações.

$$\bar{A}B\bar{C} + AB\bar{C} + A\bar{B}\bar{C}$$

$$B\bar{C}(\bar{A} + A) + A\bar{B}\bar{C} \quad => Fatorando$$

$$B\bar{C} + A\bar{B}\bar{C}$$

$$\bar{C}(B + A\bar{B}) => Fatorando\ e\ aplicando\ teorema\ 15a$$

$$\bar{C}(A + B)$$

$$A\bar{C} + B\bar{C} => Expressão\ simplificada$$

Conseguimos, queridos(as) alunos(as)! O projeto lógico do ar condicionado está pronto. Utilizamos técnicas desenvolvimento de circuitos lógicos aliada com a álgebra de Boole. Observe o resulto do circuito resultante na figura 5 abaixo.

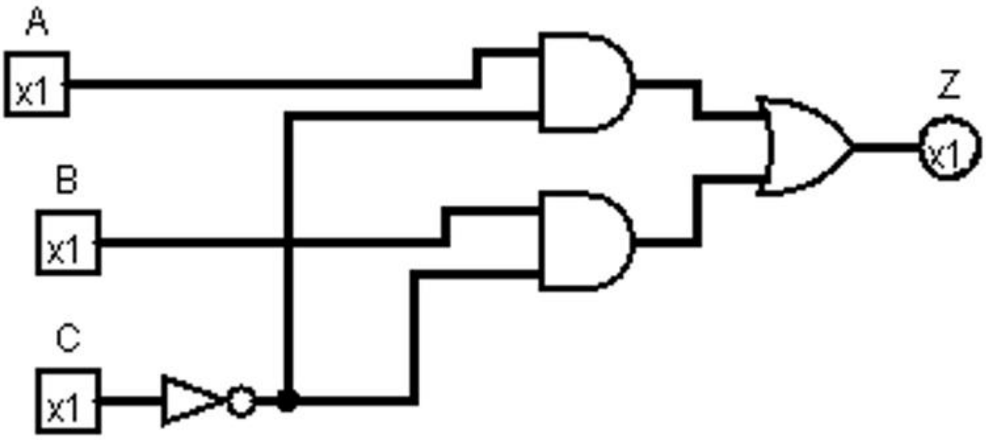


Figura 5. Circuito lógico do ar-condicionado.

Então chegamos ao fim dessa aula. Espero que tenha gostado dessa viagem pelos circuitos fundamentais da computação. Vejo vocês na próxima unidade.

Videoaula 3

Utilize o QRcode para assistir!

Assista agora à videoaula para entender melhor o assunto sobre: Desenvolvendo circuitos a partir de problemas da vida real.

Indicação de Leitura

Para complementar seus estudos até aqui, leia o livro de TOCCI os itens 4.1 a 4.4. Nessas páginas estão um bom conteúdo sobre projetos de circuitos cobinacionais.

Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/740/pdf/142?code=cXI5AdU0iFMMzkwYOMqk49T6np/R3q5Zh28HdlA11vjVBa3CW8hK93iZ2FUYcHxk7scZeiZpNEsOI+QtuLDlFw==>. Acesso em: 20.06.2020.

Encerramento

Nesta unidade estudamos como funcionam os circuitos digitais e as portas lógicas. Tenho certeza que agora você já consegue montar e analisar pequenos circuitos utilizando-se de ferramentas que foram mostradas: logisim e Deeds.

Na aula 1 começamos fazendo um estudo dos componentes mais básicos existentes na lógica digital, as portas lógicas. Com elas você notou que se constrói circuitos que podem solucionar problemas da vida real. Então você entrou em contato com as seguintes portas básicas: AND, OR, NOT.

Na aula 2 aprendemos a criar circuitos digitais por meio da combinação das portas lógicas. Com esses circuitos em mãos, vimos que podemos equacioná-lo utilizando-se da álgebra booleana. Com esse equacionamento em mãos poderíamos fazer uma análise ou uma simplificação do projeto.

Que pena! Terminamos mais um estudo. Parabéns pela sua dedicação até aqui. Lembrando que o sucesso dos estudos depende unicamente de você, aluno(a). Aproveite o seu tempo e as oportunidades. Faça as leituras da bibliografia proposta, pesquise na internet e tire todas suas dúvidas. Sucesso!

Referências

FLOYD, Thomas L. **Sistemas digitais:** fundamentos e aplicações. 9. ed. Porto Alegre, RS: Bookman, 2007. 888 p. ISBN 978-85-60031-93-1*. Observação: Entre no catálogo online da Biblioteca da UNIFIL e pesquise com os números de acervo: 251153 e 251154.*

TOCCI, Ronald J.; WIDNER, Neal S.; MOSS, Gregory L. **Sistemas digitais**: princípios e aplicações. 10. ed. São Paulo: Pearson, 2008. 804 p. ISBN 978-85-7605-095-7. Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/740/pdf/142?code=cXI5AdU0iFMMzkwYOMqk49T6np/R3q5Zh28HdlA11vjVBa3CW8hK93iZ2FUYcHxk7scZeiZpNEsOI+QtuLDlFw== >. Acesso em: 20.06.2020.

Esperamos que este guia o tenha ajudado compreender a organização e o funcionamento de seu curso. Outras questões importantes relacionadas ao curso serão disponibilizadas pela coordenação.

Grande abraço e sucesso!

UniFil