

Infraestrutura de Hardware

Aula 01 - Circuitos Digitais Combinatórios

2016.2

João Marcelo Teixeira
joao.marceloteixeira@ufrpe.br

DEINFO - UFRPE

1 Circuitos Digitais Combinatórios

Os primeiros dois capítulos contêm uma revisão de projeto de circuitos digitais:

- Combibatórios, ou sem memória, no Capítulo 1
- Circuitos sequenciais, com memória, no Capítulo 2

Tópicos neste capítulo:

1.1	Sinais, Operadores Lógicos e Portas
1.2	Funções e Expressões Booleanas
1.3	Projetando Redes de Portas
1.4	Elementos Combinatórios Úteis
1.5	Elementos Combinatórios Programáveis
1.6	Considerações sobre Temporização e Circuito

1.1 Sinais, Operadores Lógicos e Portas

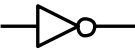

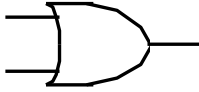

Name	NOT	AND	OR	XOR
Graphical symbol				
Operator sign and alternate(s)	x' $\neg x$ or \bar{x}	xy $x \wedge y$	$x \vee y$ $x + y$	$x \oplus y$ $x \neq y$
Output is 1 iff:	Input is 0	Both inputs are 1s	At least one input is 1	Inputs are not equal
Arithmetic expression	$1 - x$	$x \times y$ or xy	$x + y - xy$	$x + y - 2xy$

Figura 1.1 Alguns elementos básicos de circuitos lógicos digitais, com os operadores usados no livro em destaque.

Variações nos Símbolos das Portas

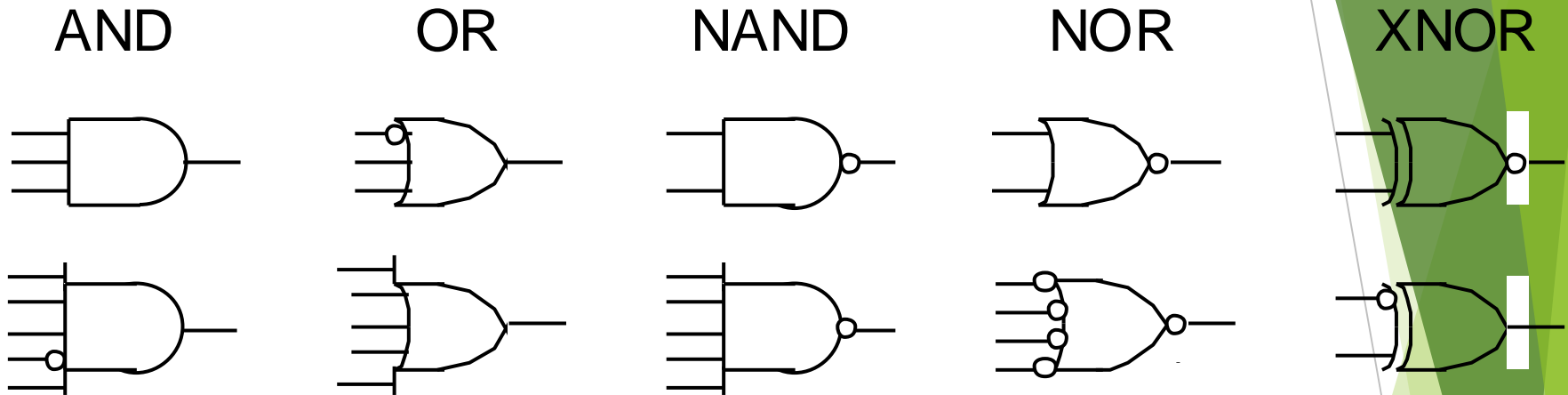
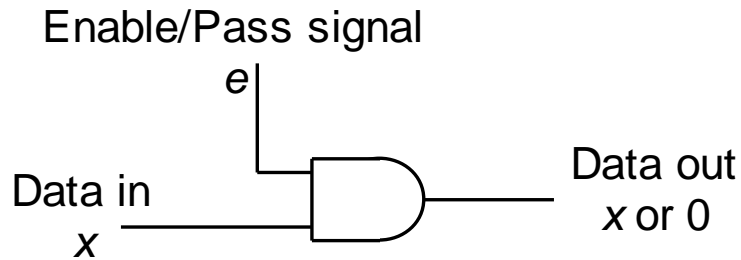
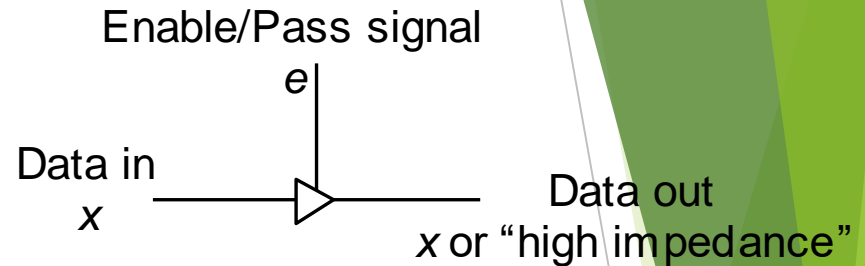


Figura 1.2 Portas com mais de duas entradas e/ou com sinais invertidos na entrada ou saída.

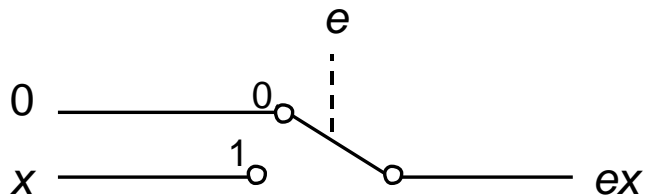
Portas como Elementos de Controle



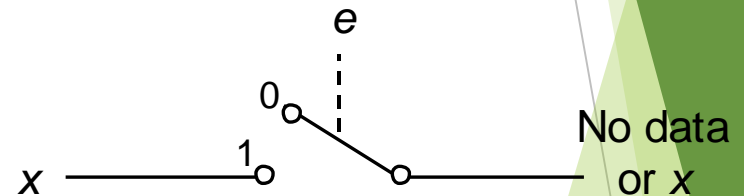
(a) AND gate for controlled transfer



(b) Tristate buffer



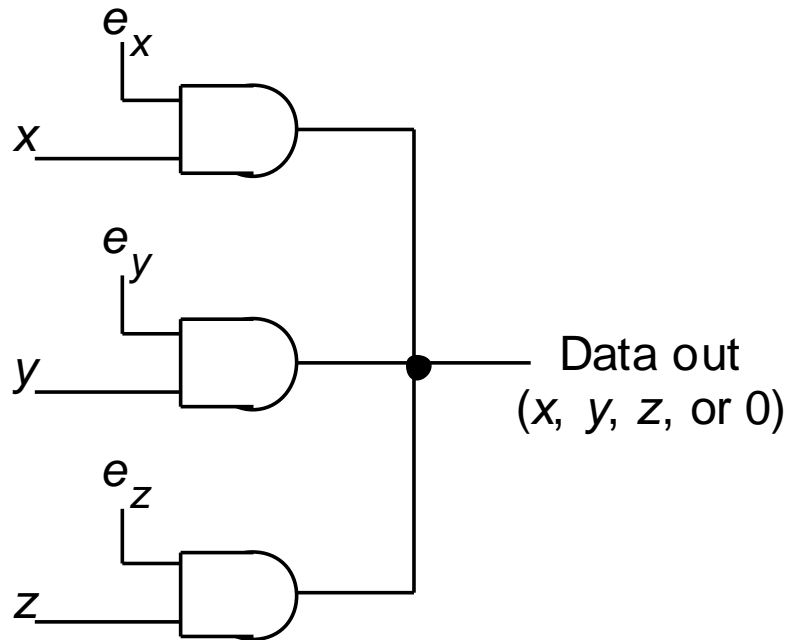
(c) Model for AND switch.



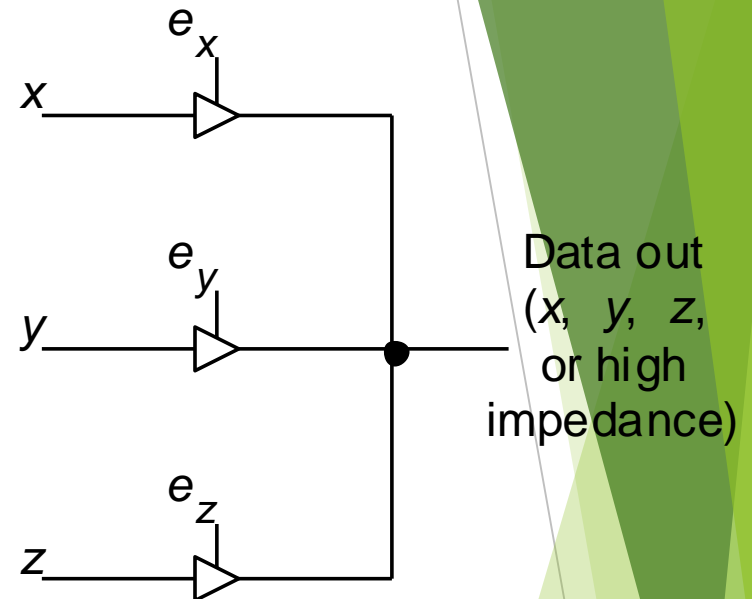
(d) Model for tristate buffer.

Figura 1.3 Uma porta AND e um buffer tri-estado podem atuar como chaves controladas ou válvulas. Um buffer inversor possui a mesma função lógica que uma porta NOT.

OR de Fiação e Conexões de Barramento



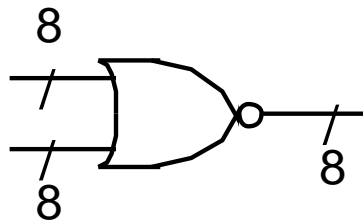
(a) Wired OR of product terms



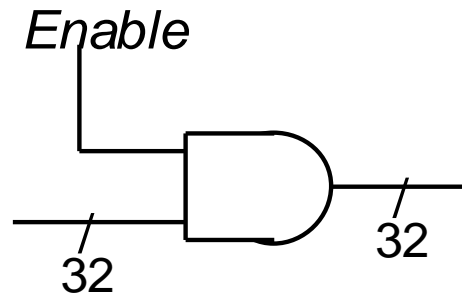
(b) Wired OR of tristate outputs

Figura 1.4 OR de fiação permite juntar vários sinais controlados.

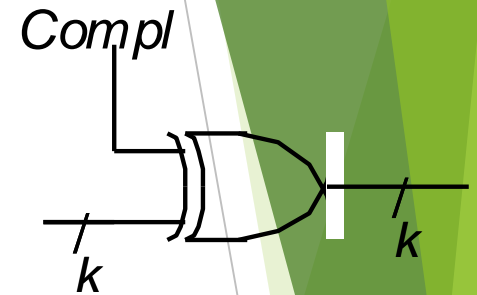
Sinais e Conjuntos de Sinais de Dado/Controle



(a) 8 NOR gates



(b) 32 AND gates



(c) k XOR gates

Figura 1.5 Vetores de portas lógicas representadas por um único símbolo de porta.

1.2 Funções Booleanas e Expressões

Formas de especificar uma função lógica

- Tabela verdade: 2^n linhas, “don’t-care” na entrada ou na saída
- Expressão lógica: $w' (x \vee y \vee z)$, produto de somas, soma de produtos, expressões equivalentes
- Declaração através de palavras: O alarme irá soar se a porta estiver aberta enquanto o Sistema de segurança estiver ativado, ou quando o detector de fumaça for ativado
- Diagrama de circuito lógico: Síntese vs Análise

Manipulando Expressões Lógicas

Tabela 1.2 Leis (identidades básicas) da Álgebra Booleana.

Nome da Lei	Versão OR	Versão AND
Identidade	$x \vee 0 = x$	$x 1 = x$
Um/Zero	$x \vee 1 = 1$	$x 0 = 0$
Idempotente	$x \vee x = x$	$x x = x$
Inversa	$x \vee x' = 1$	$x x' = 0$
Commutativa	$x \vee y = y \vee x$	$x y = y x$
Associativa	$(x \vee y) \vee z = x \vee (y \vee z)$	$(x y) z = x (y z)$
Distributiva	$x \vee (y z) = (x \vee y) (x \vee z)$	$x (y \vee z) = (x y) \vee (x z)$
de DeMorgan	$(x \vee y)' = x' y'$	$(x y)' = x' \vee y'$

Provando a Equivalência de Expressões Lógicas

Exemplo 1.1

- Método da tabela verdade: Verificação exaustiva

- Substituição aritmética

$$x \vee y = x + y - xy$$

$$x \oplus y = x + y - 2xy$$

Exemplo: $x \oplus y \equiv? x'y \vee xy'$

$$x + y - 2xy \equiv? (1-x)y + x(1-y) - (1-x)yx(1-y)$$

- Análise de caso: dois casos, $x = 0$ or $x = 1$

- Manipulação da expressão lógica

1.3 Projetando Redes de Portas

- AND-OR, NAND-NAND, OR-AND, NOR-NOR
- Otimização lógica: custo, velocidade, potência dissipada

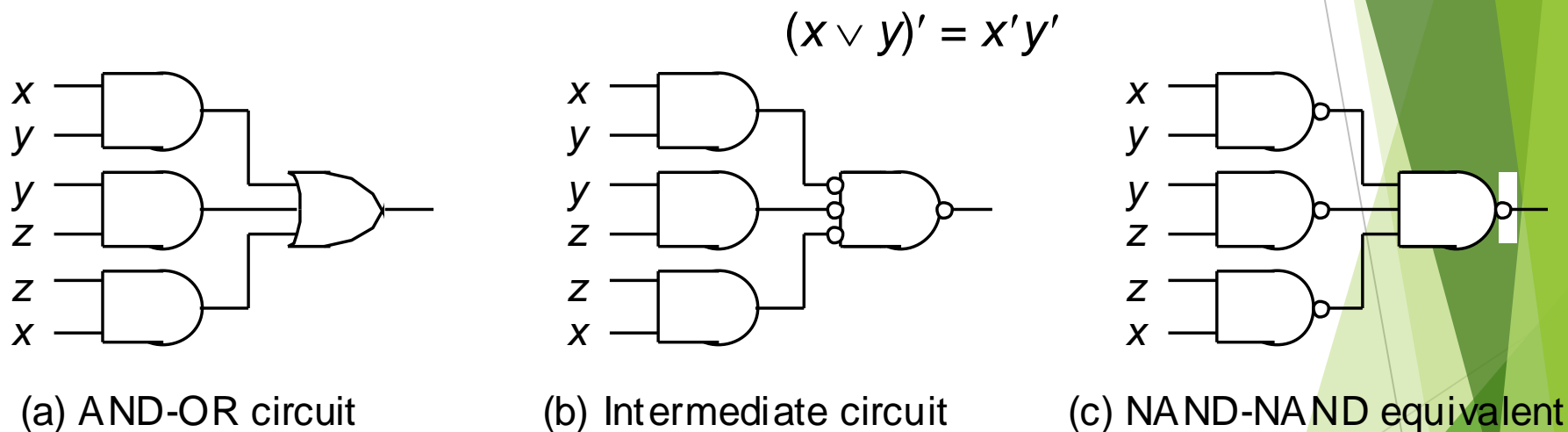


Figura 1.6 Um circuito em dois níveis AND-OR e dois circuitos equivalentes.

Display de Sete Segmentos para Dígitos Decimais

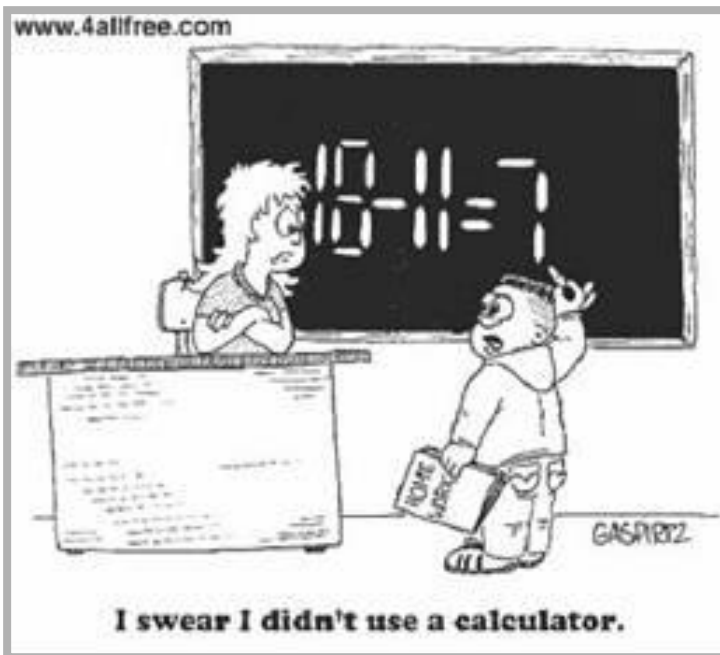
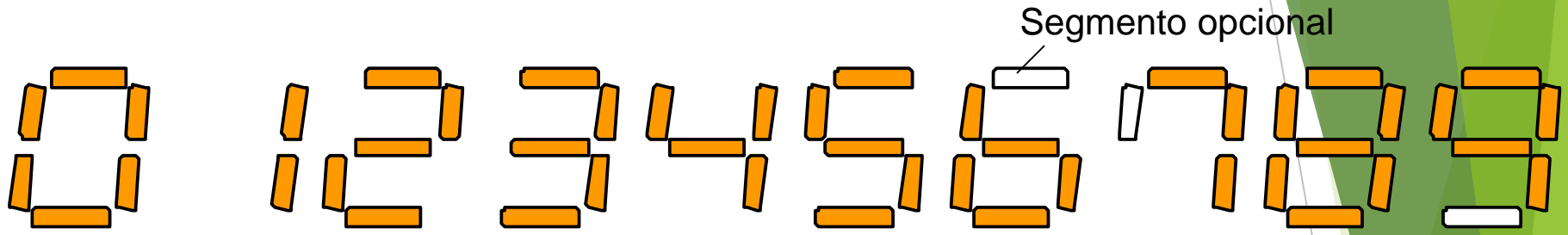


Figura 1.7 Display de sete segmentos de dígitos decimais. Os três segmentos não preenchidos podem ser opcionalmente utilizados. O dígito 1 pode ser mostrado de duas formas, sendo adotada a versão mais comum localizada no lado direito.

Decodificador BCD para Sete Segmentos

Exemplo 1.2

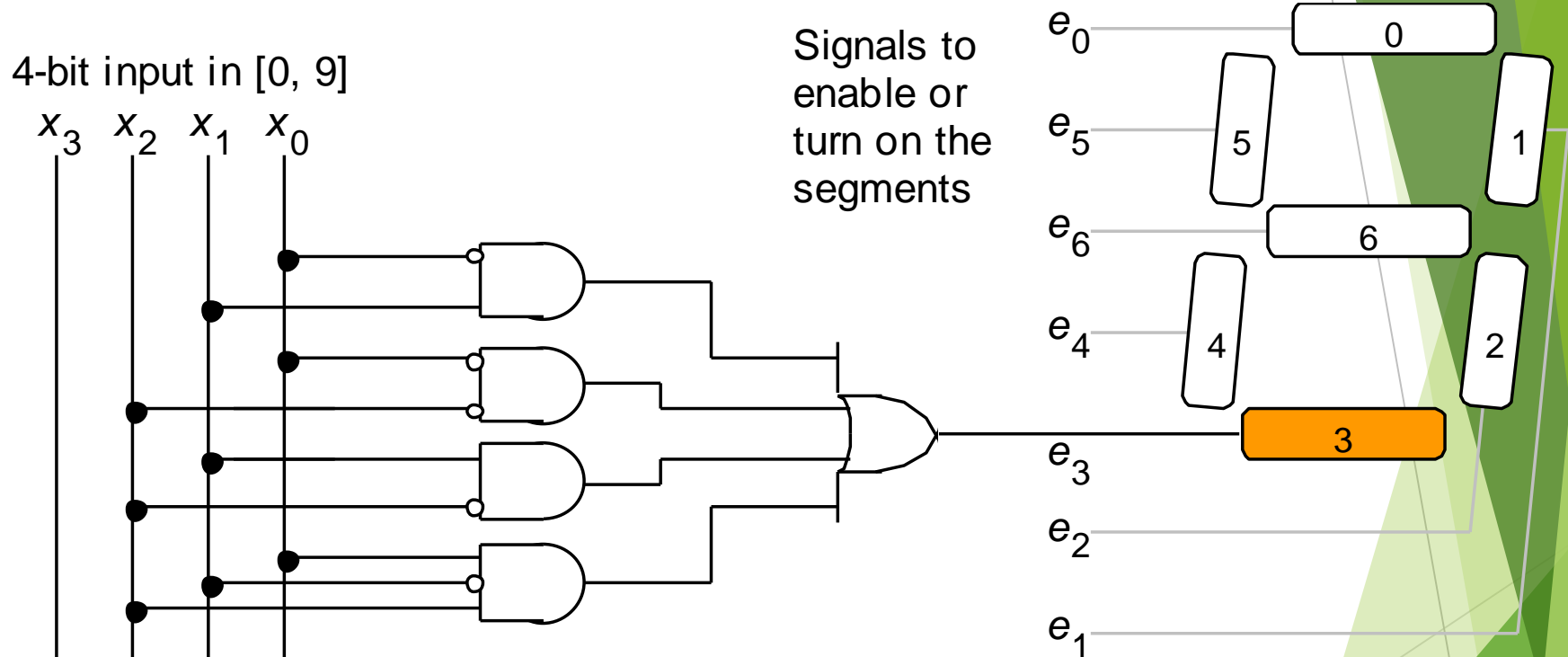
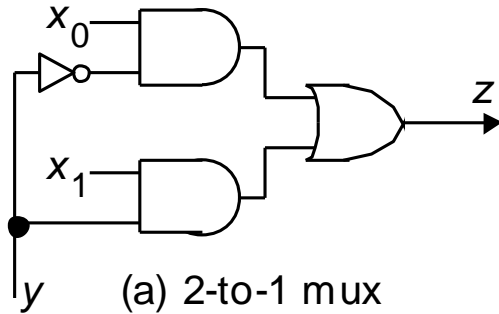


Figura 1.8 O circuito lógico que gera o sinal de habilitação para o segment inferior (número 3) em uma unidade de display de sete segmentos.

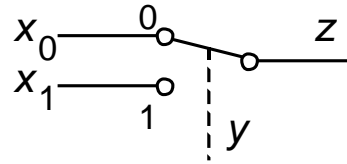
1.4 Elementos Combinatórios Úteis

- Blocos de construção de alto nível
- Assemelham-se a partes pré-fabricadas de uma casa
- Componentes aritméticos (somadores, multiplicadores, ULAs) serão abordados na parte III
- Aqui serão descritos três elementos úteis: multiplexadores, decodificadores/demultiplexadores, codificadores

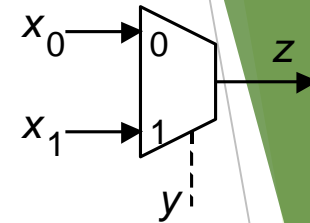
Multiplexadores



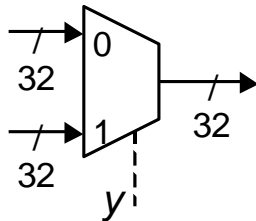
(a) 2-to-1 mux



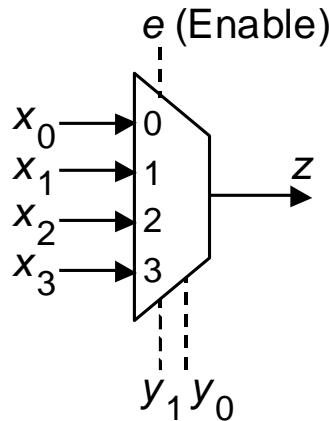
(b) Switch view



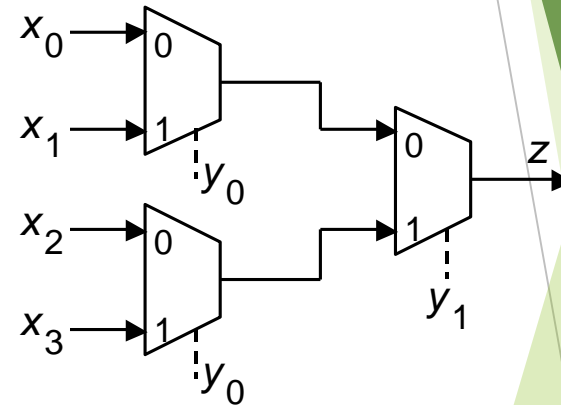
(c) Mux symbol



(d) Mux array



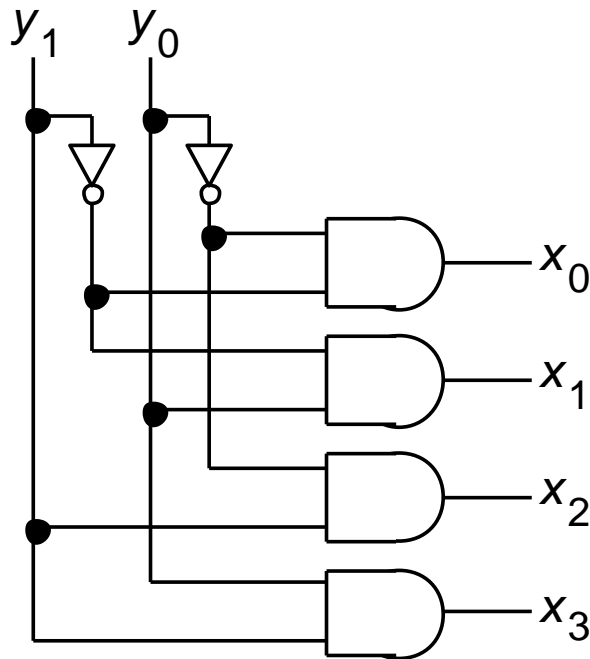
(e) 4-to-1 mux with enable



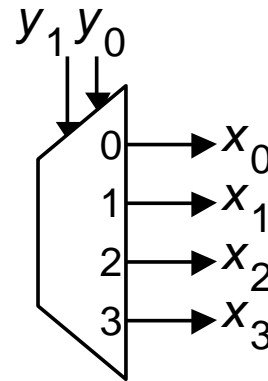
(e) 4-to-1 mux design

Figura 1.9 Um multiplexador (mux), ou selecionador, permite a uma de várias entradas ser selecionada e encaminhada para a saída, dependendo do valor binário de um conjunto de seleção ou do sinal de endereço fornecido.

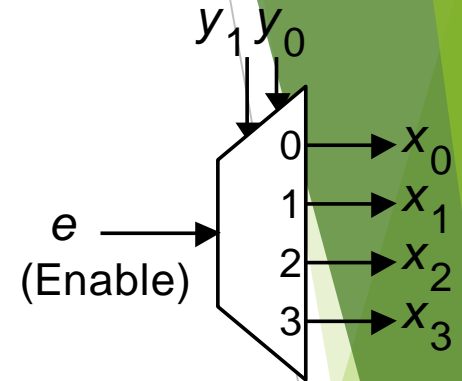
Decodificadores/Demultiplexadores



(a) 2-to-4 decoder



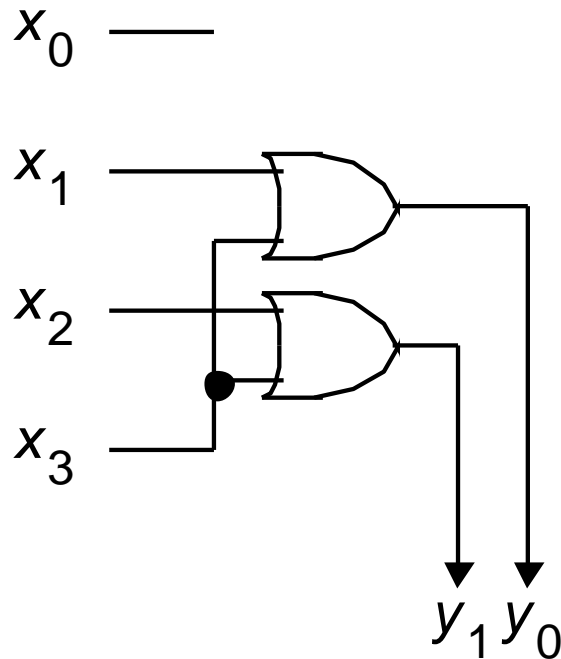
(b) Decoder symbol



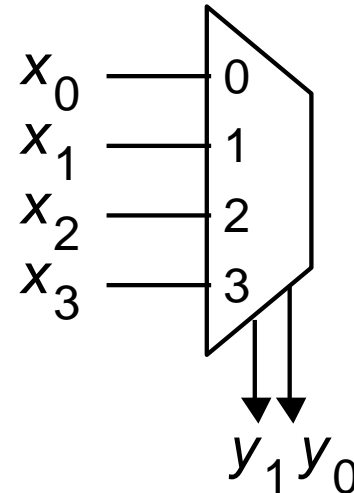
(c) Demultiplexer, or decoder with “enable”

Figura 1.10 Um decodificador permite a seleção de uma dentre 2^a opções usando um endereço com a bits de entrada. O demultiplexador (demux) é um decodificador que apenas seleciona uma saída se o sinal de habilitação estiver ativado.

Codificadores



(a) 4-to-2 encoder



(b) Encoder symbol

Figura 1.11 Um codificador 2^a -para- a , tendo como saída um número binário com a bits igual ao índice correspondente à entrada (dentre as suas 2^a) ativada.

1.5 Elementos Combinatórios Programáveis

Um elemento combinatório programável pode realizar a função de várias portas ou redes de portas

São programados através do corte de conexões existentes (*fusíveis*) ou estabelecendo novas conexões (*anti-fusíveis*)

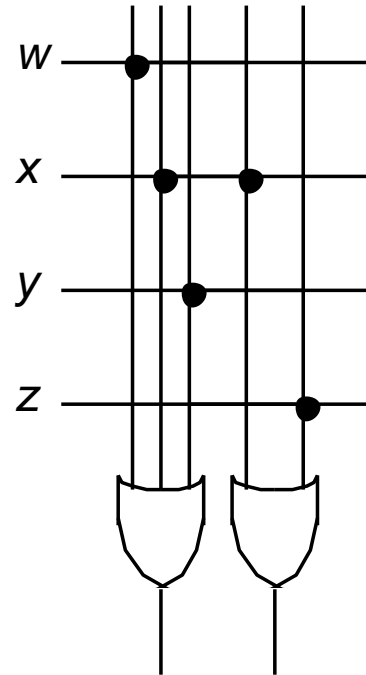
- ROM Programável (PROM)
- Programmable Array Logic (PAL)
- Programmable Logic Array (PLA)

ts

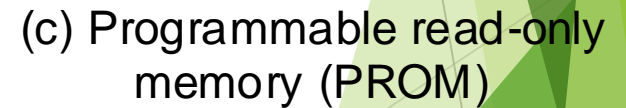
Outputs

Programmable read-only memory (PROM)

em uma PROM.



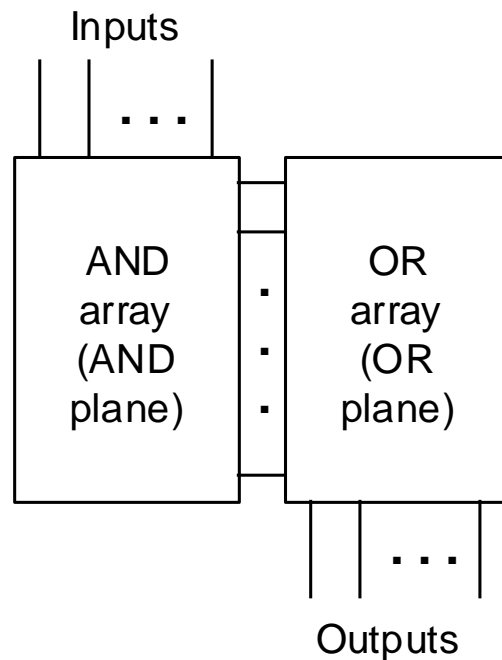
(b) Logic equivalent
of part a



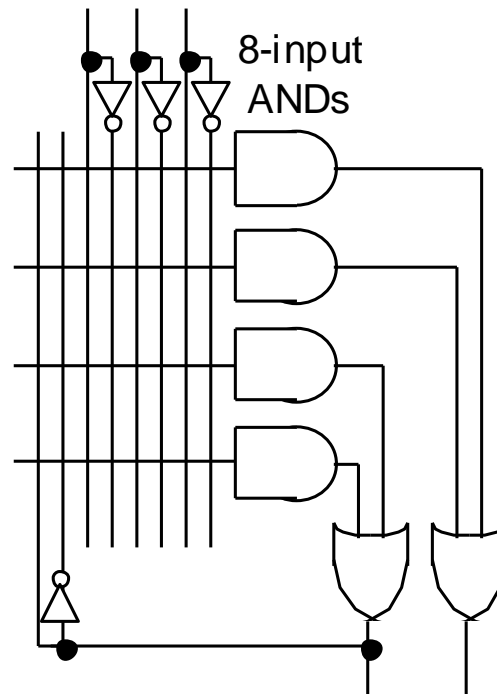
Slide 20

B. Parham

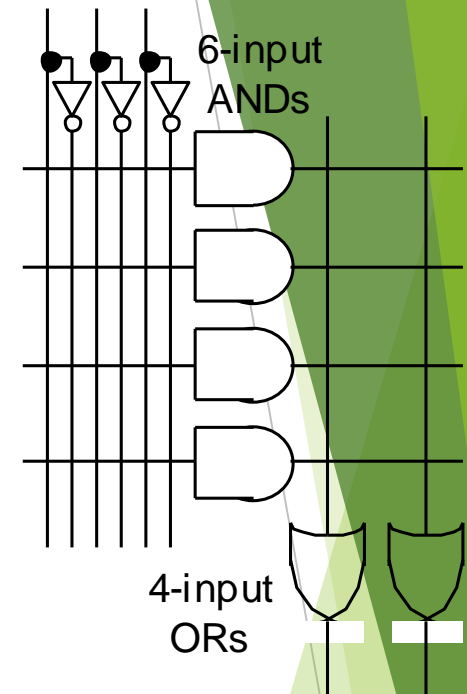
PALs e PLAs



(a) General programmable combinational logic



(b) PAL: programmable AND array, fixed OR array



(c) PLA: programmable AND and OR arrays

Figura 1.13 Lógica combinatória programável: estrutura geral e duas classes conhecidas como dispositivos PAL e PLA. Não são apresentados a PROM com vetor AND fixo (um decodificador) e o vetor OR programado.

1.6 Considerações sobre Temporização e Circuito

Mudanças na saída da porta/circuito, acionadas por mudanças nas respectivas entradas, não são instantâneas

- Atraso da porta (δ): uma fração de, até poucos nanossegundos
- Atraso da conexão, previamente ignorável, agora é importante (sinais eletrônicos viajam cerca de 15cm por ns)
- Simulação do circuito para verificar funcionalidade e temporização

Glitching

Usando o PAL da Fig. 1.13b para implementar $f = x \vee y \vee z$

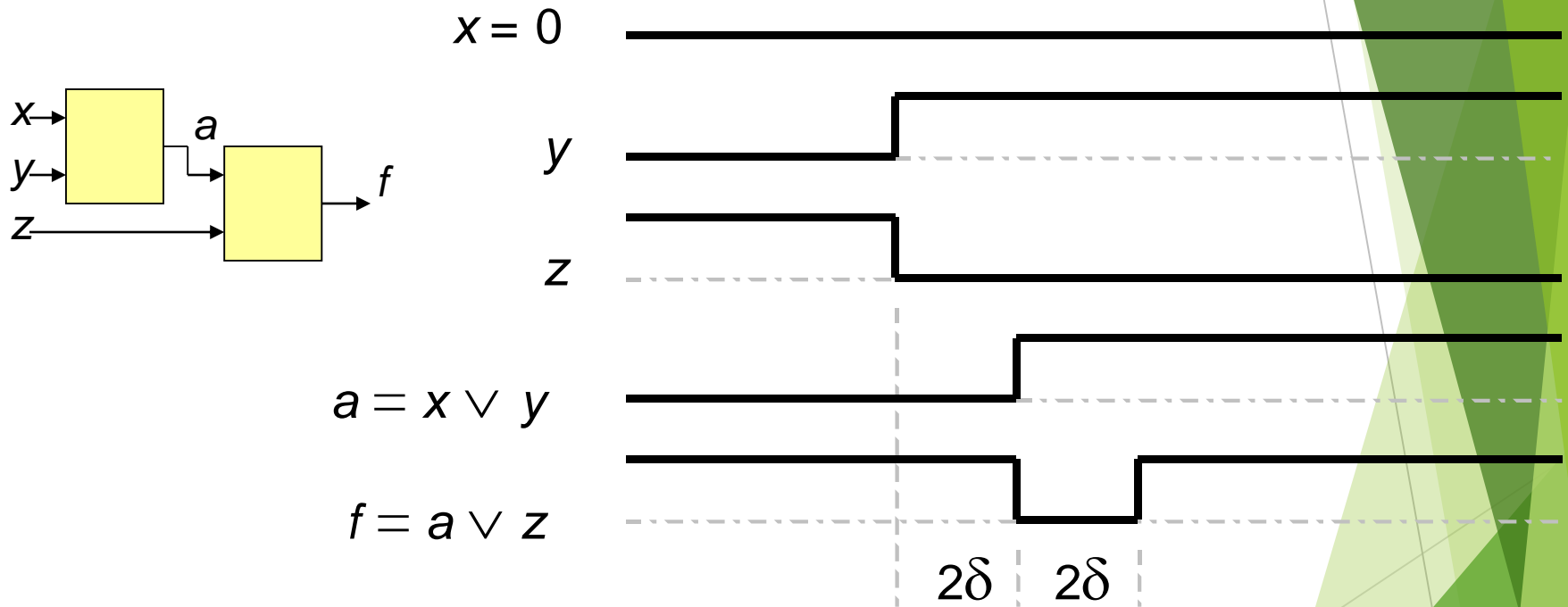
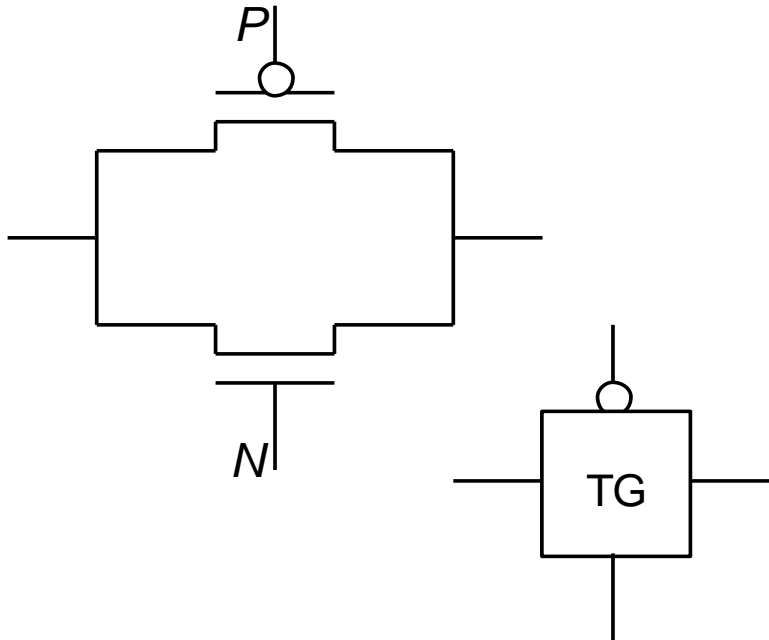
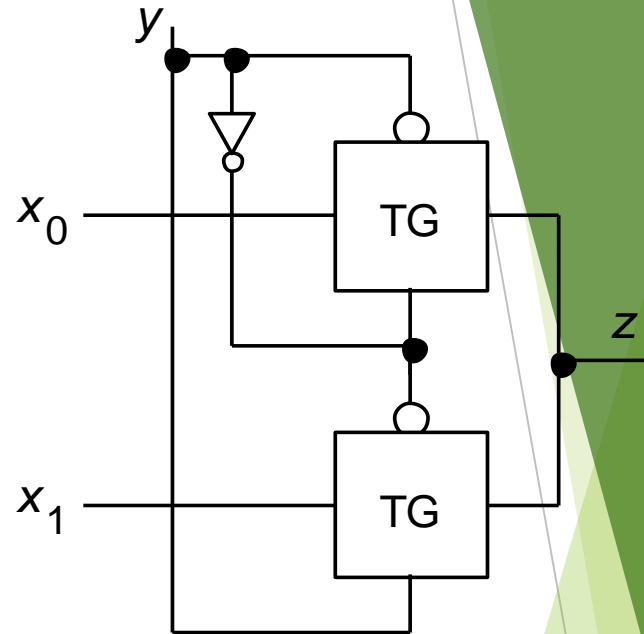


Figura 1.14 Diagrama de tempo para um circuito que apresenta glitching.

Portas de Transmissão CMOS



(a) CMOS transmission gate: circuit and symbol



(b) Two-input mux built of two transmission gates

Figura 1.15 Uma porta de transmissão CMOS e seu uso na construção de um MUX 2:1.