

Circuitos Lógicos

DIM 0128

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Aula 13

Circuitos Combinacionais:

Multiplicador, Deslocador, Comparador e BCD

Edgard de Faria Corrêa

2022.1

Objetivo

◇ Multiplicador Binário

- Conceito e Tipos

◇ Deslocador

- Conceito e Tipos
- Técnicas de deslocamento em operações aritméticas

◇ Comparador

- Conceito e Tipos

◇ BCD (*Binary Converter to Decimal*)

- Conceito e Projeto

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Conceito:

- A multiplicação binária é uma operação um pouco mais complexa de se implementar, se comparado ao somador ou ao subtrator.
- Um multiplicador é um bloco combinacional que efetua a multiplicação entre 2 números (N-bits e M-bits), resultando em uma saída de **(N+M)**-bits.

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números de 4 bits:

	1 1 0 0	(12 ₁₀)	multiplicando
x	0 1 1 0	(6 ₁₀)	multiplicador
	<hr/>		
	0 0 0 0	1º produto parcial = zero	
+	1 1 0 0	2º produto parcial = multiplicando	
	1 1 0 0	3º produto parcial = multiplicando	
	0 0 0 0	4º produto parcial = zero	
	<hr/>		
	1 0 0 1 0 0 0	(72 ₁₀)	resultado

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números de 4 bits:

	1 1 0 0	(12 ₁₀)	multiplicando
x	0 1 1 0	(6 ₁₀)	multiplicador
<hr/>			
	0 0 0 0	1º produto parcial = zero	
+	1 1 0 0	2º produto parcial = multiplicando	
	1 1 0 0	3º produto parcial = multiplicando	
	0 0 0 0	4º produto parcial = zero	
<hr/>			
	1 0 0 1 0 0 0	(72 ₁₀)	resultado

- São **4 produtos parciais** a serem somados

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números de 4 bits:

	1 1 0 0	(12 ₁₀)	multiplicando
x	0 1 1 0	(6 ₁₀)	multiplicador
<hr/>			
	0 0 0 0	1º produto parcial = zero	
+	1 1 0 0	2º produto parcial = multiplicando	
	1 1 0 0	3º produto parcial = multiplicando	
	0 0 0 0	4º produto parcial = zero	
<hr/>			
	1 0 0 1 0 0 0	(72 ₁₀)	resultado

- São 4 produtos parciais a serem somados
- Porém, os somadores vistos até agora só podem **somar dois números por vez!!!**

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números de 4 bits:

■ Reorganizando...

	1 1 0 0	(12 ₁₀)	multiplicando
x	0 1 1 0	(6 ₁₀)	multiplicador
<hr/>			
	0 0 0 0	1º produto parcial (= zero)	
+	1 1 0 0	2º produto parcial (= multiplicando)	
<hr/>			
	1 1 0 0 0		
+	1 1 0 0	3º produto parcial (= multiplicando)	
<hr/>			
	1 0 0 1 0 0 0		
+	0 0 0 0	4º produto parcial (= zero)	
<hr/>			
	1 0 0 1 0 0 0	(72 ₁₀)	resultado

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números de 4 bits:

- Outro exemplo...

	1 1 1 1	(15 ₁₀)	multiplicando
x	1 1 1 1	(15 ₁₀)	multiplicador
<hr/>			
	1 1 1 1	1º produto parcial (= multiplicando)	
+	1 1 1 1	2º produto parcial (= multiplicando)	
<hr/>			
	1 0 1 1 0 1		
+	1 1 1 1	3º produto parcial (= multiplicando)	
<hr/>			
	1 1 0 1 0 0 1		
+	1 1 1 1	4º produto parcial (= multiplicando)	
<hr/>			
	1 1 1 0 0 0 0 1	(225 ₁₀)	resultado

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números de 4 bits:

- Outro exemplo...

	1 1 1 1	(15 ₁₀)	multiplicando
x	1 1 1 1	(15 ₁₀)	multiplicador
<hr/>			
	1 1 1 1	1º produto parcial (= multiplicando)	
+	1 1 1 1	2º produto parcial (= multiplicando)	
<hr/>			
	1 0 1 1 0 1		
+	1 1 1 1	3º produto parcial (= multiplicando)	
<hr/>			
	1 1 0 1 0 0 1		
+	1 1 1 1	4º produto parcial (= multiplicando)	
<hr/>			
	1 1 1 0 0 0 0 1	(225 ₁₀)	resultado

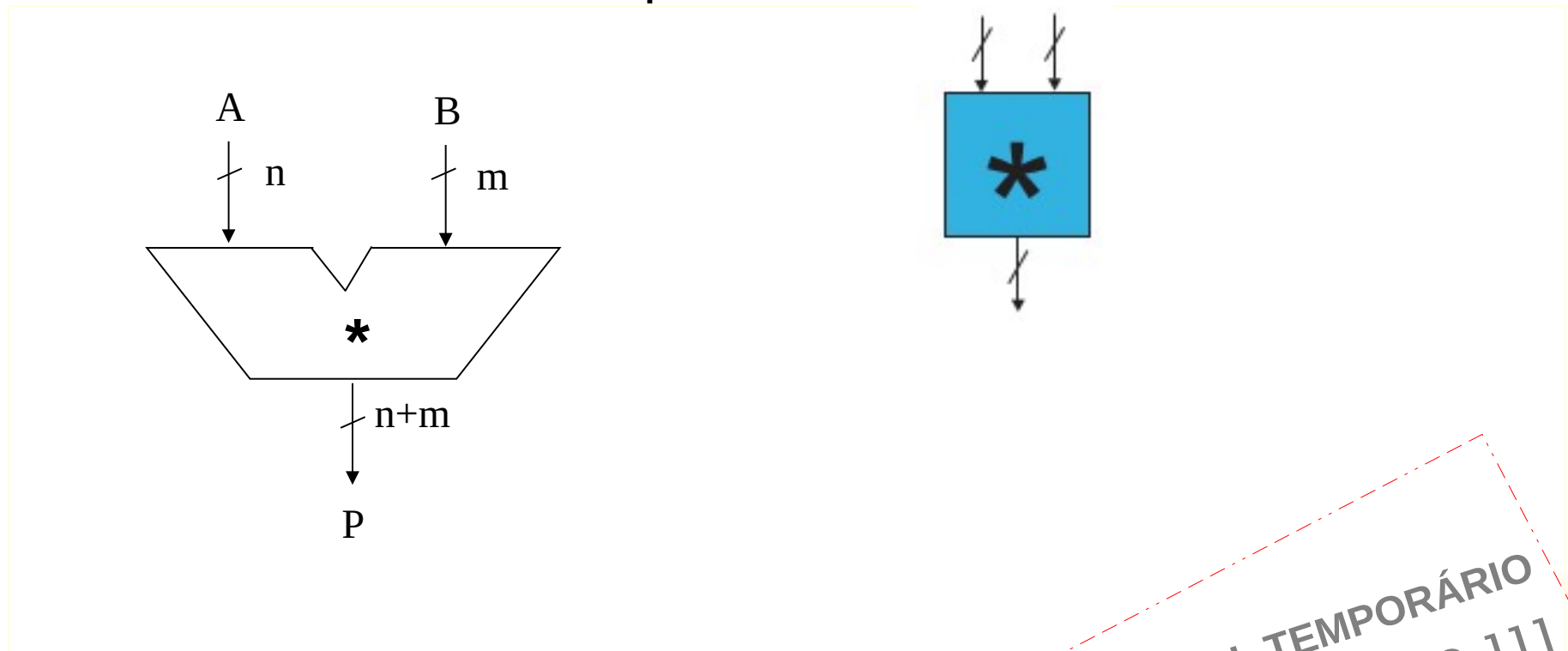
- Conclusão: **n** bits **x** **m** bits **até** **n+m** bits

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números:

- Símbolo de um multiplicador:



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

- ◇ Mas como multiplicar 2 números de 1 bit cada?

$$\begin{array}{r} 0 \\ \times 0 \\ \hline \end{array}$$

$$\begin{array}{r} 0 \\ \times 1 \\ \hline \end{array}$$

$$\begin{array}{r} 1 \\ \times 0 \\ \hline \end{array}$$

$$\begin{array}{r} 1 \\ \times 1 \\ \hline \end{array}$$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Mas como multiplicar 2 números de 1 bit cada?

$$\begin{array}{r} 0 \\ \times 0 \\ \hline 0 \end{array}$$

$$\begin{array}{r} 0 \\ \times 1 \\ \hline 0 \end{array}$$

$$\begin{array}{r} 1 \\ \times 0 \\ \hline 0 \end{array}$$

$$\begin{array}{r} 1 \\ \times 1 \\ \hline 1 \end{array}$$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Mas como multiplicar 2 números de 1 bit cada?

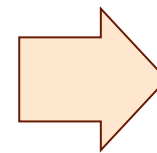
$$\begin{array}{r} 0 \\ \times 0 \\ \hline 0 \end{array}$$

$$\begin{array}{r} 0 \\ \times 1 \\ \hline 0 \end{array}$$

$$\begin{array}{r} 1 \\ \times 0 \\ \hline 0 \end{array}$$

$$\begin{array}{r} 1 \\ \times 1 \\ \hline 1 \end{array}$$

- Resulta 1 se os dois bits valem 1
- Nunca ocorre *overflow*!



operação E (**AND**)

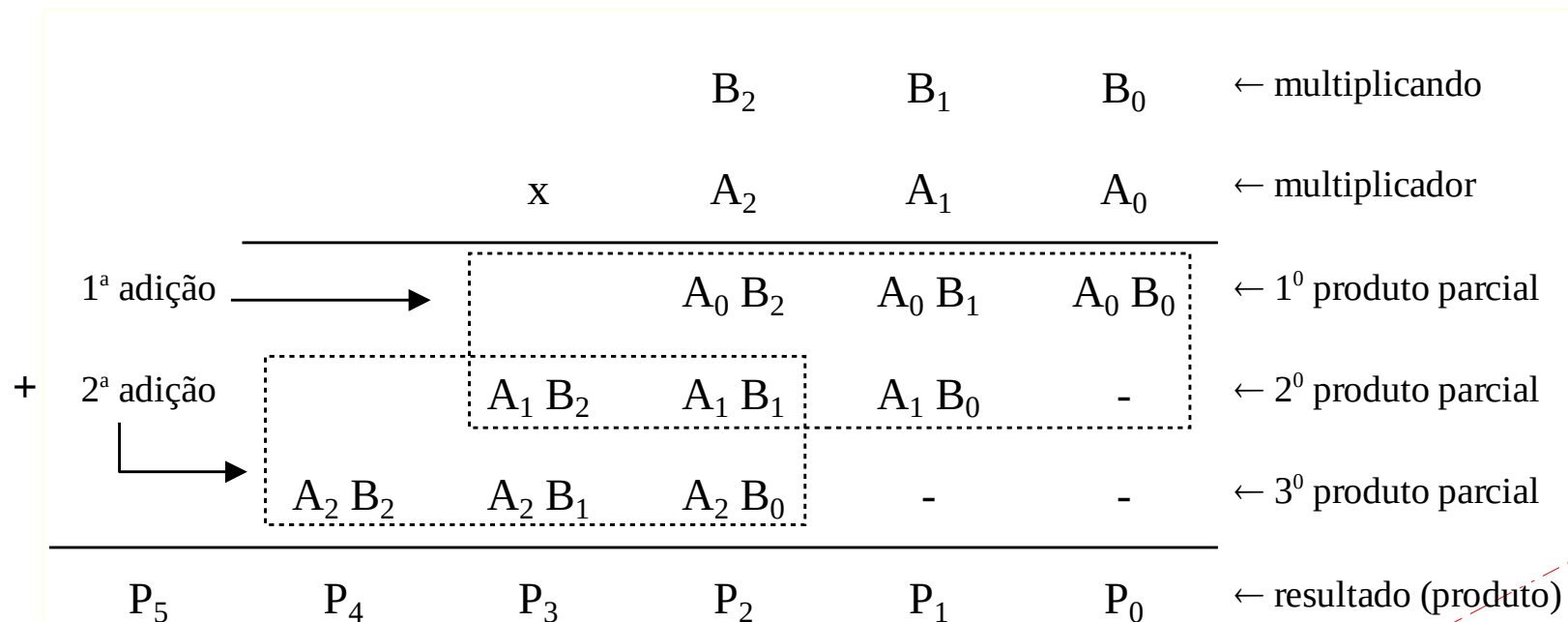
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicando 2 números:

■ Generalizando...



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Exemplo: **Conversor de temperatura**

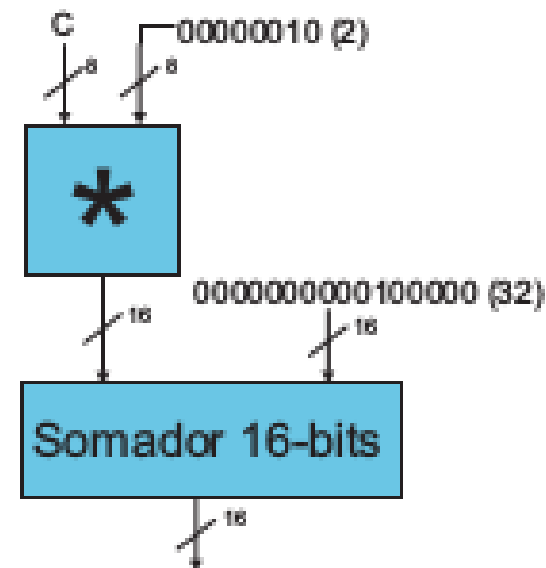
■ Equação de conversão:

- ◇ $F = (9/5)C + 32$

■ Equação de conversão aproximada:

- ◇ $F = 2C + 32$

■ Circuito combinacional:

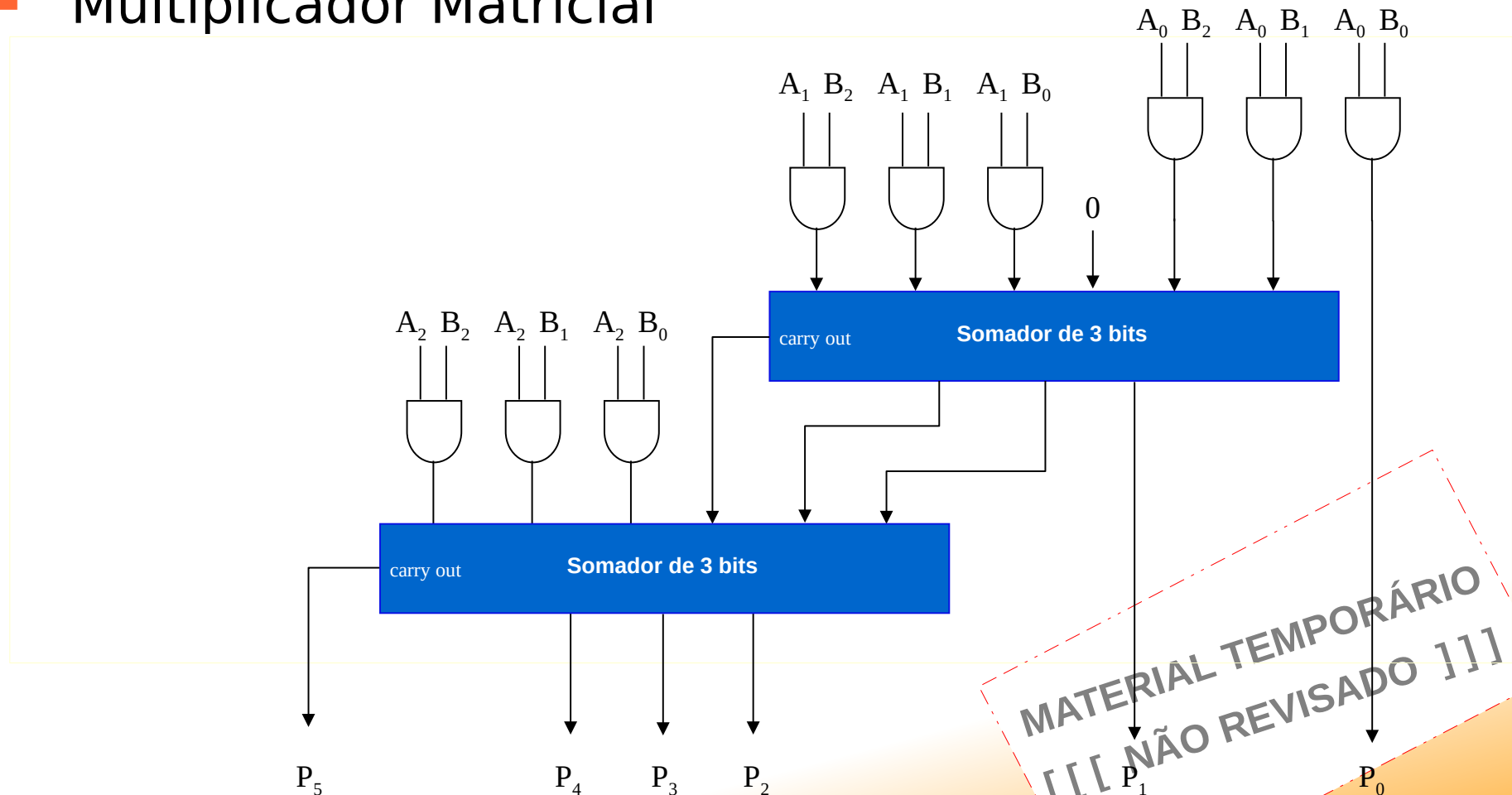


Multiplicação Binária

(números sem sinal)

◆ Implementação:

■ Multiplicador Matricial



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

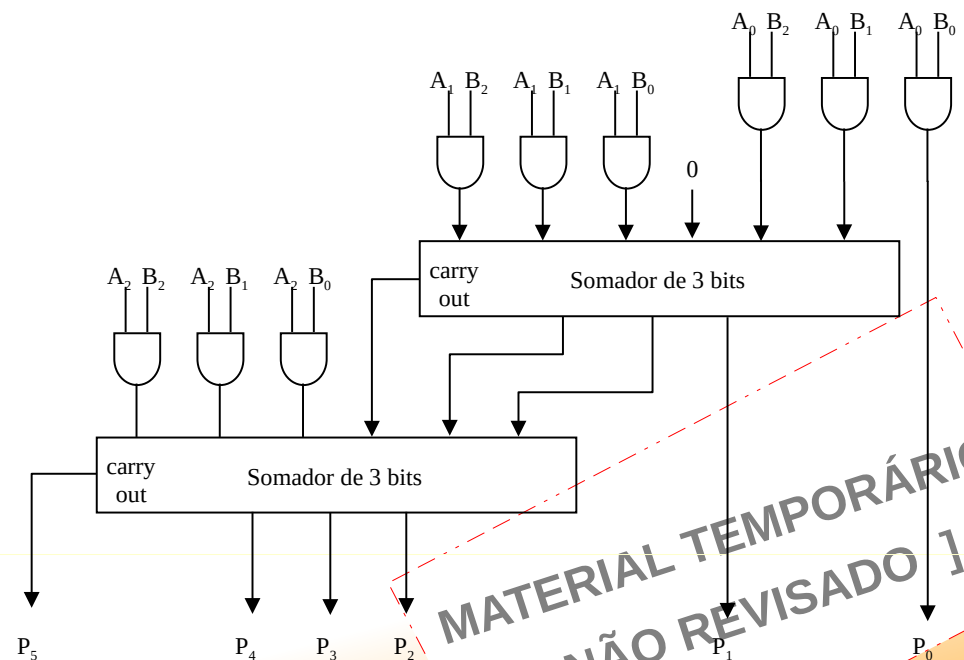
Multiplicação Binária

(números sem sinal)

◆ Implementação:

■ Multiplicador Matricial

— **Pergunta:** Qual é o custo de um multiplicador como este?



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◆ Implementação:

■ Multiplicador Matricial

— **Pergunta:** Qual é o custo de um multiplicador como este?

— **Resposta:**

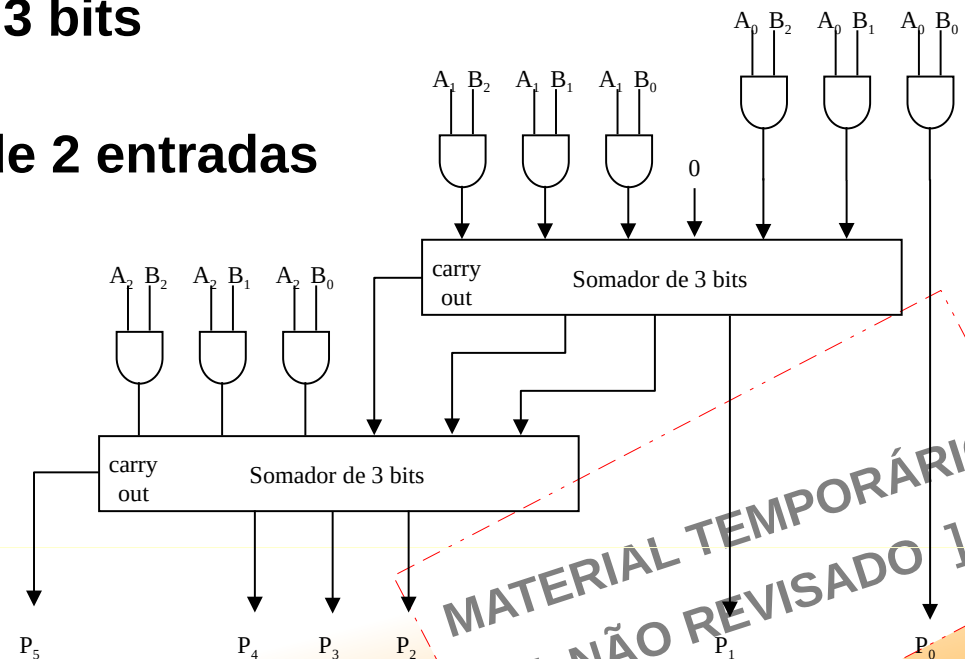
2 x custo do somador de 3 bits

+

9 x custo de uma NAND de 2 entradas

+

9 x custo de um inversor



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◆ Implementação:

■ Multiplicador Matricial

— **Pergunta:** Qual é o custo de um multiplicador como este?

— **Resposta:**

2 x custo do somador de 3 bits

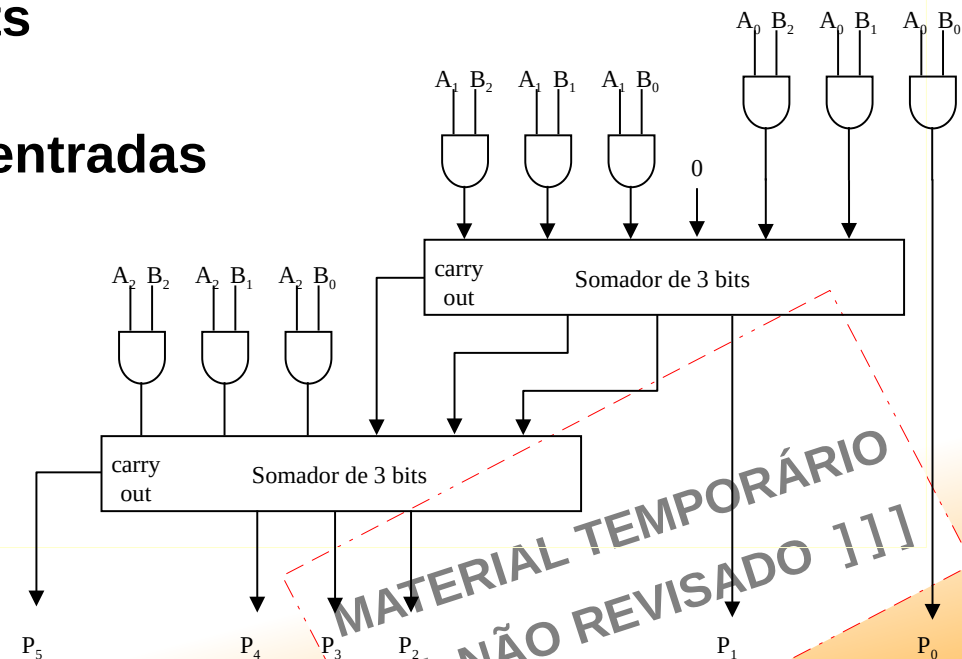
+

9 x custo de uma NAND de 2 entradas

+

9 x custo de um inversor

Inversor???



Multiplicação Binária

(números sem sinal)

◆ Implementação:

■ Multiplicador Matricial

— **Pergunta:** Qual é o custo de um multiplicador como este?

— **Resposta:**

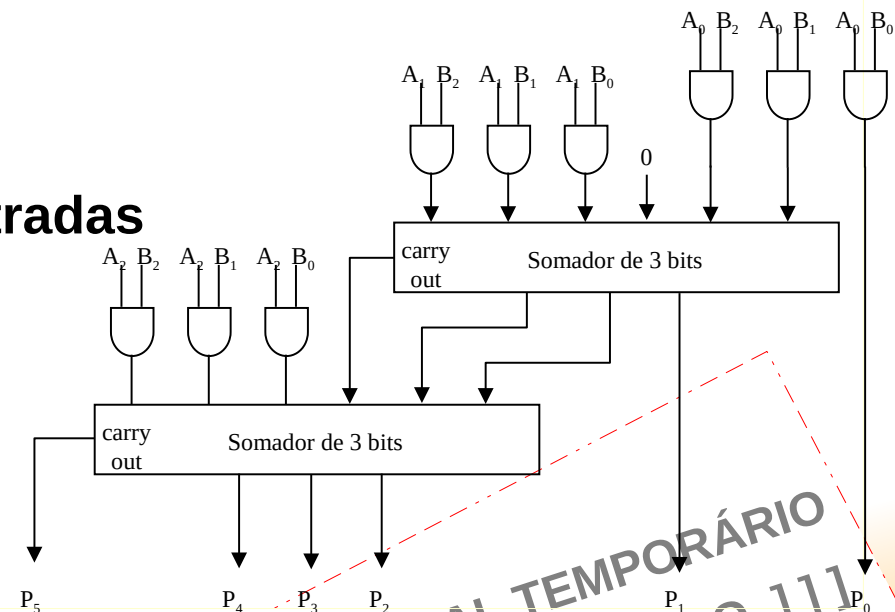
2 x custo do somador de 3 bits

+

9 x custo de uma NAND de 2 entradas

+

9 x custo de um inversor



Alternativas mais simples/econômicas???

Multiplicação Binária

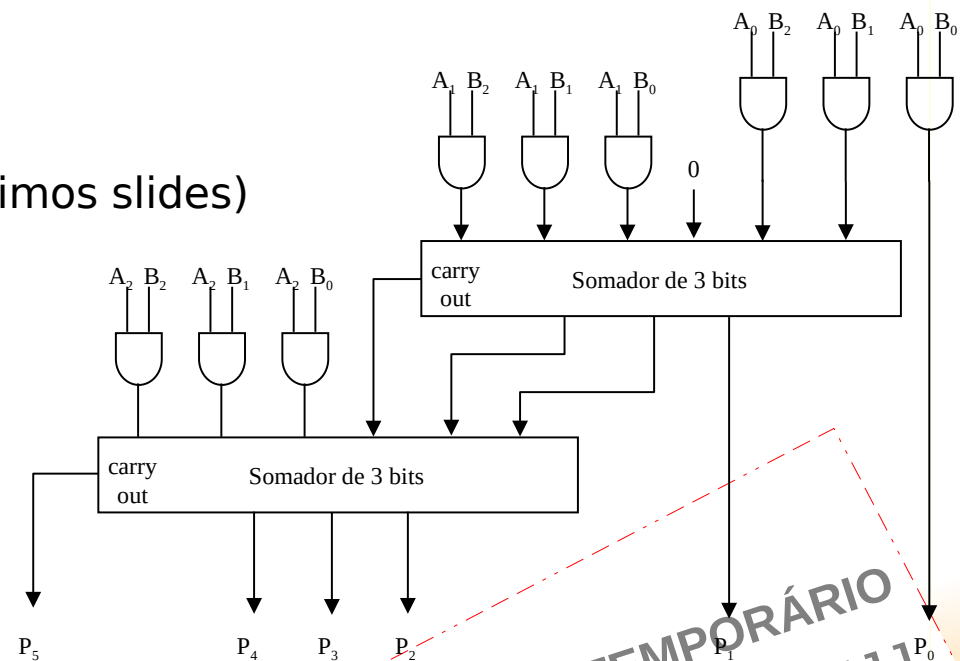
(números sem sinal)

◆ Implementação:

■ Multiplicador Matricial

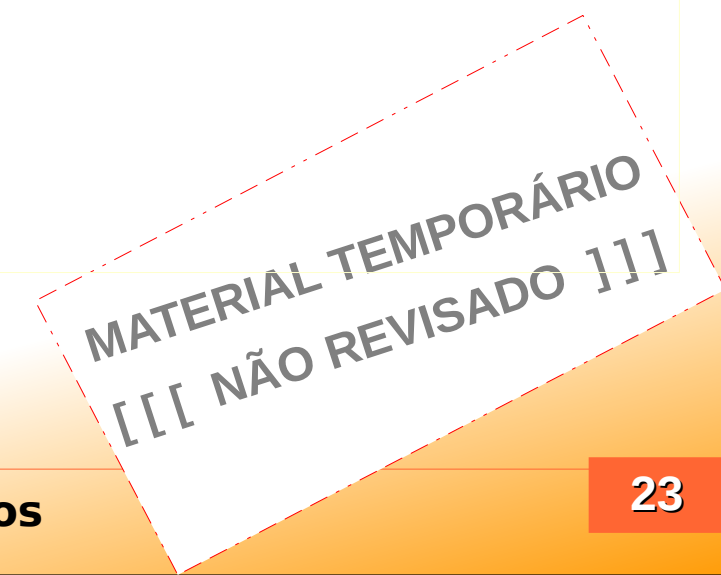
— Alternativas:

- **Mútiplas somas**
- **Deslocamento** (próximos slides)



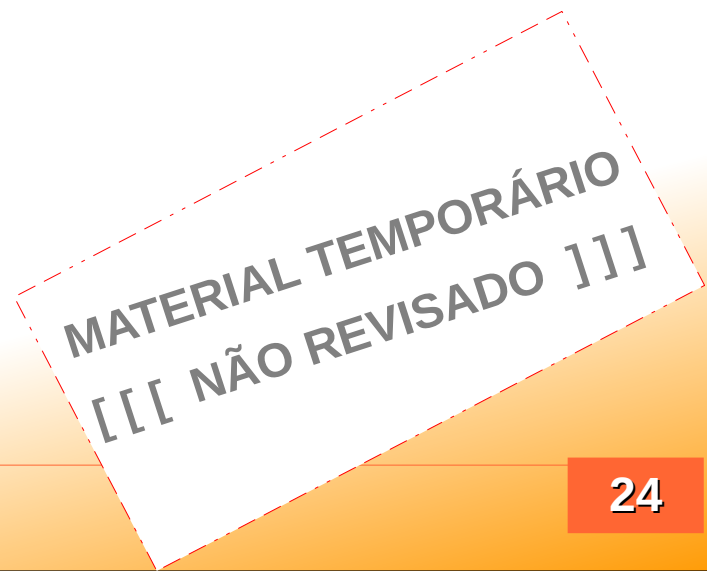
Alternativas mais simples/econômicas???

Deslocador



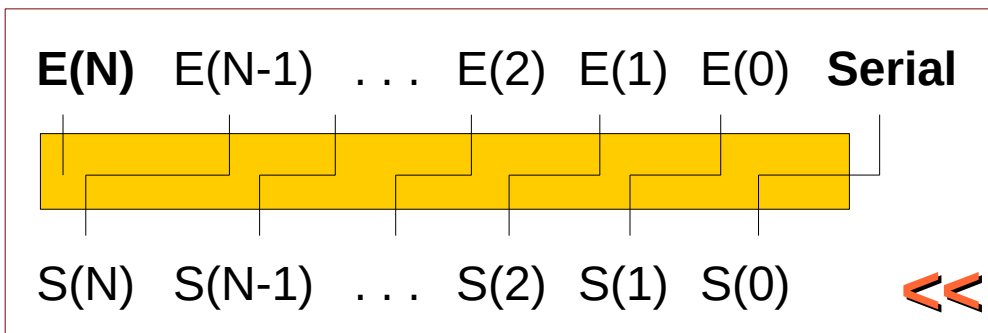
Deslocador

- ◇ Circuito combinacional, com **n** entradas e **n** saídas, que permite deslocar **n** bits de uma palavra binária para direita (**>>**) ou para esquerda (**<<**).
 - Não utiliza nenhuma porta lógica, apenas fios.
 - Descarta bit:



Deslocador

- ◇ Circuito combinacional, com **n** entradas e **n** saídas, que permite deslocar **n** bits de uma palavra binária para direita (**>>**) ou para esquerda (**<<**).
 - Não utiliza nenhuma porta lógica, apenas fios.
 - Descarta bit:
 - mais significativo (bit N), se deslocamento à esquerda. (**<<**)

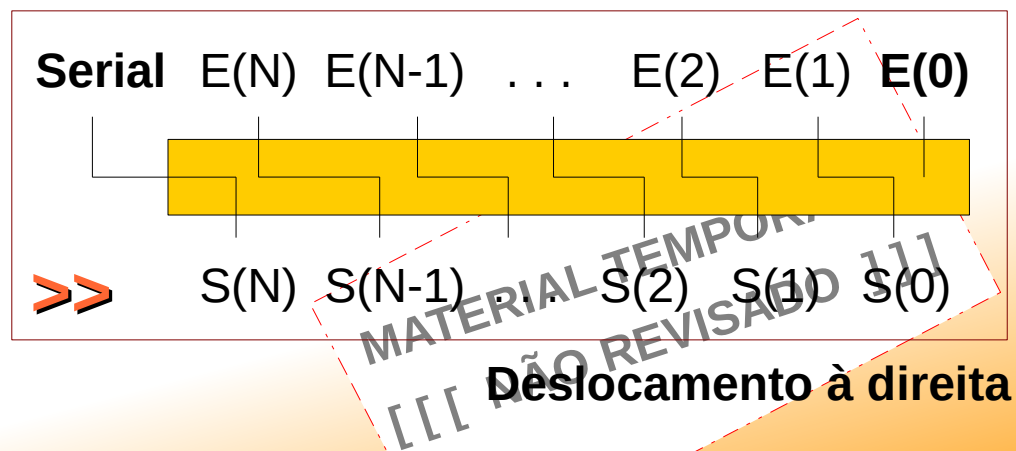


Deslocamento à esquerda

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

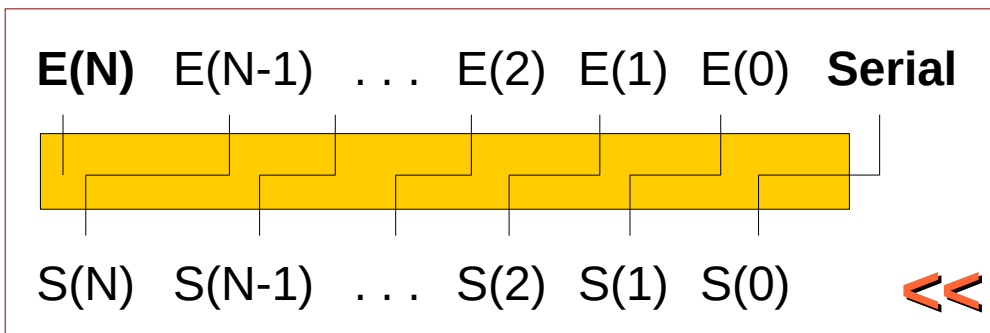
Deslocador

- ◇ Circuito combinacional, com **n** entradas e **n** saídas, que permite deslocar **n** bits de uma palavra binária para direita (**>>**) ou para esquerda (**<<**).
 - Não utiliza nenhuma porta lógica, apenas fios.
 - Descarta bit:
 - menos significativo (bit 0), se deslocamento à direita. (**>>**)

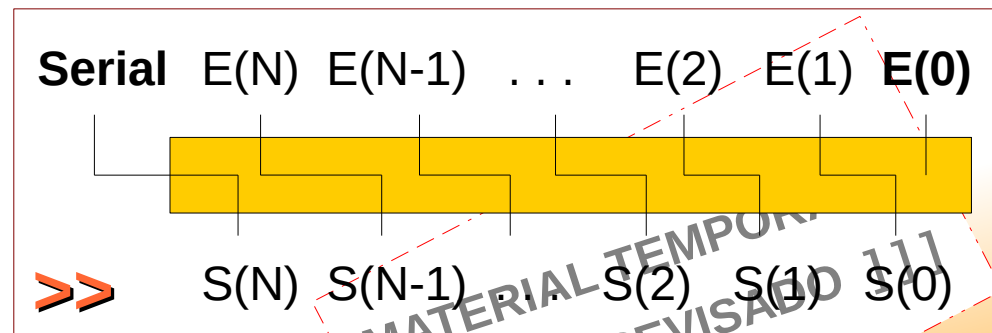


Deslocador

- ◇ Circuito combinacional, com **n** entradas e **n** saídas, que permite deslocar **n** bits de uma palavra binária para direita (**>>**) ou para esquerda (**<<**).
 - Não utiliza nenhuma porta lógica, apenas fios.
 - Descarta bit:
 - mais significativo (bit N), se deslocamento à esquerda. (**<<**)
 - menos significativo (bit 0), se deslocamento à direita. (**>>**)

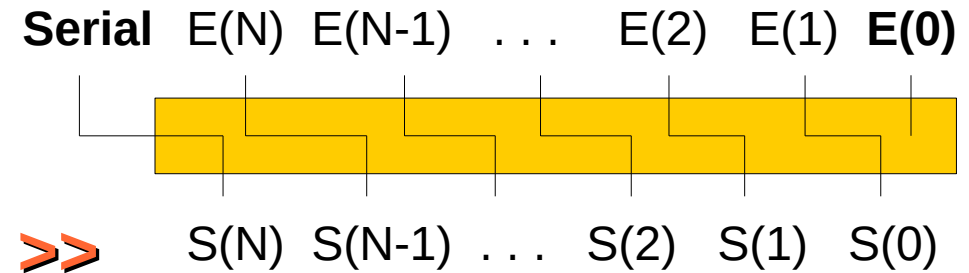
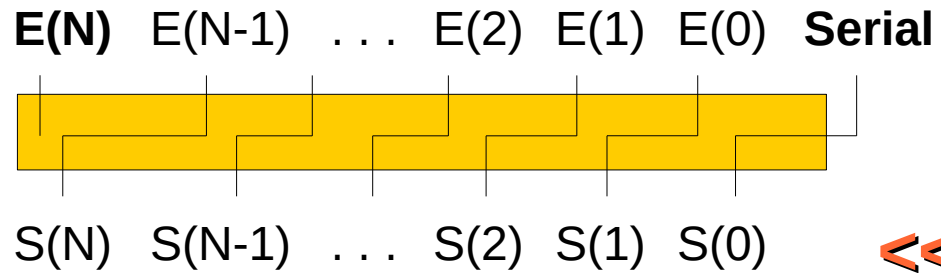


Deslocamento à esquerda



Deslocamento à direita

Deslocador: Operações aritméticas



◇ Exemplos de deslocamentos à esquerda

- $0110 \rightarrow 1100$:
- $0101 \rightarrow 1010$:
- $0010 \rightarrow 0100$:

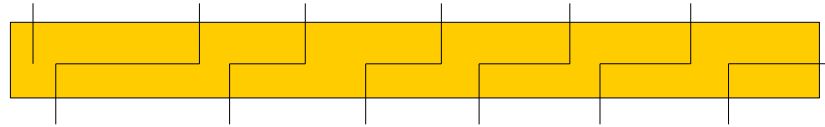
◇ Exemplos de deslocamentos à direita

- $0110 \rightarrow 0011$:
- $0100 \rightarrow 0010$:
- $1110 \rightarrow 0111$:

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

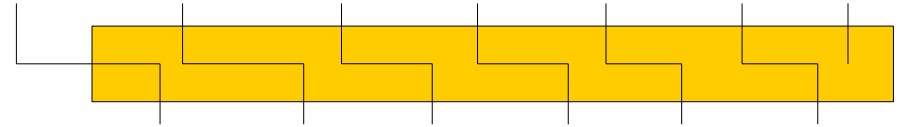
Deslocador: Operações aritméticas

E(N) E(N-1) ... E(2) E(1) E(0) Serial



S(N) S(N-1) ... S(2) S(1) S(0) <<

Serial E(N) E(N-1) ... E(2) E(1) E(0)



>> S(N) S(N-1) ... S(2) S(1) S(0)

◇ Exemplos de deslocamentos à esquerda

- 0110 → 1100: 6 → 12
- 0101 → 1010: 5 → 10
- 0010 → 0100: 2 → 4

◇ Exemplos de deslocamentos à direita

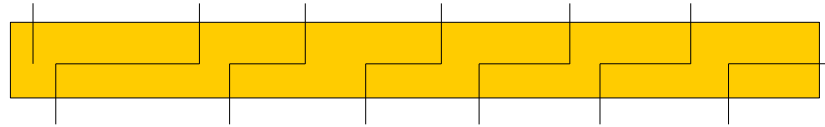
- 0110 → 0011: 6 → 3
- 0100 → 0010: 4 → 2
- 1110 → 0111: 14 → 7

◇ Resumindo...

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

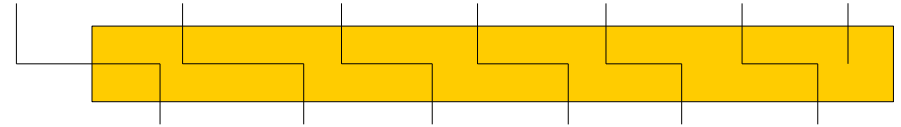
Deslocador: Operações aritméticas

E(N) E(N-1) ... E(2) E(1) E(0) Serial



S(N) S(N-1) ... S(2) S(1) S(0) <<

Serial E(N) E(N-1) ... E(2) E(1) E(0)



>> S(N) S(N-1) ... S(2) S(1) S(0)

◇ Exemplos de deslocamentos à esquerda

- 0110 → 1100: 6 → 12
- 0101 → 1010: 5 → 10
- 0010 → 0100: 2 → 4

◇ Exemplos de deslocamentos à direita

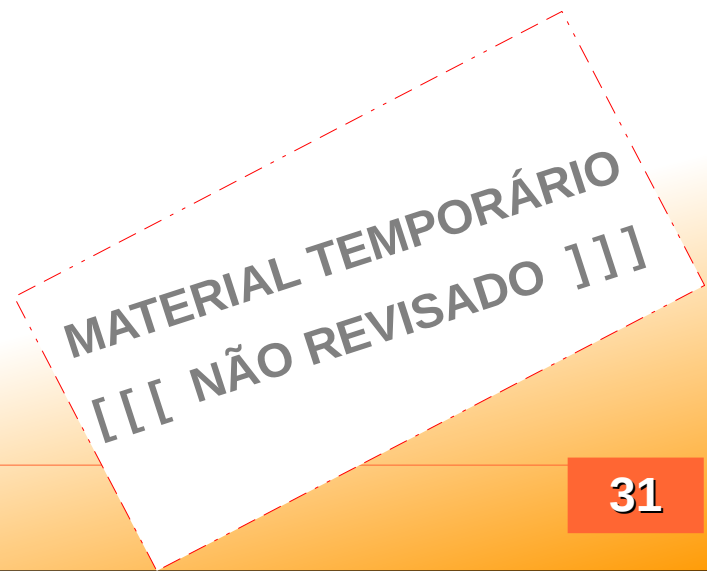
- 0110 → 0011: 6 → 3
- 0100 → 0010: 4 → 2
- 1110 → 0111: 14 → 7

◇ Resumindo...

- Deslocamento de 1 bit à **esquerda**: multiplica por 2
- Deslocamento de 1 bit à **direita**: divide por 2

Deslocador: Operações aritméticas

- ◇ O que acontece quando o deslocamento de 1 bit ocorre duas vezes?
 - à esquerda:
 - $001100 \rightarrow 110000$: $12 \rightarrow 48$
 - à direita:
 - $001100 \rightarrow 000011$: $12 \rightarrow 3$
- ◇ Resumindo...



Deslocador: Operações aritméticas

- ◇ O que acontece quando o deslocamento de 1 bit ocorre duas vezes?
 - à esquerda:
 - $001100 \rightarrow 110000$: $12 \rightarrow 48$
 - à direita:
 - $001100 \rightarrow 000011$: $12 \rightarrow 3$
- ◇ Resumindo...
 - Deslocamento à **esquerda de 2 bits**: multiplica por 4
 - Deslocamento à **direita de 2 bits**: divide por 4

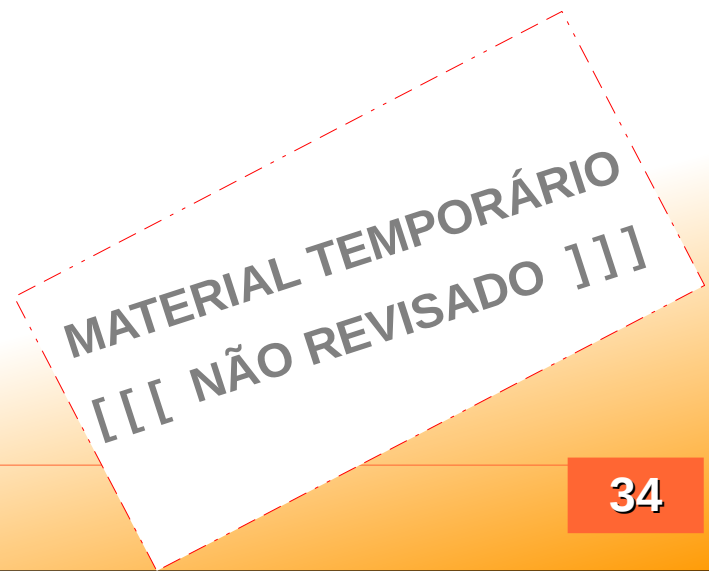
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Deslocador: Operações aritméticas

- ◇ O que acontece quando o deslocamento de 1 bit ocorre duas vezes?
 - à esquerda:
 - $001100 \rightarrow 110000$: $12 \rightarrow 48$
 - à direita:
 - $001100 \rightarrow 000011$: $12 \rightarrow 3$
- ◇ Resumindo...
 - Deslocamento à esquerda de 2 bits: multiplica por 4
 - Deslocamento à direita de 2 bits: divide por 4
- ◇ Conclusão:
 - Realizar **n** deslocamentos à **esquerda** de 1 bit, equivale a **multiplicar** o número por **2^n**
 - Realizar **n** deslocamentos à **direita** de 1 bit, equivale a **dividir** o número por **2^n**

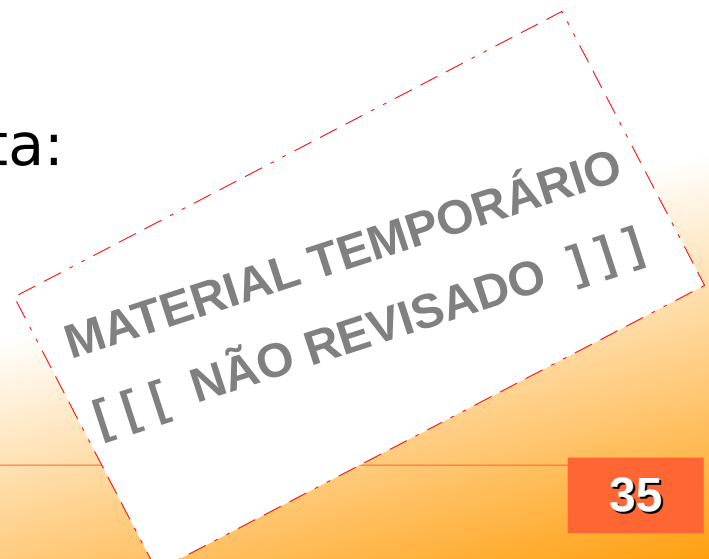
Deslocador: Operações aritméticas

- ◇ Nem sempre é válido realizar operações aritméticas com deslocadores
 - Entrada $E(n)$ é desconsiderada no deslocamento à esquerda.
 - Entrada $E(0)$ é desconsiderada no deslocamento à direita.
 - Se uma delas for **1**, operação aritmética não será válida.



Deslocador: Operações aritméticas

- ◇ Nem sempre é válido realizar operações aritméticas com deslocadores
 - Entrada $E(n)$ é desconsiderada no deslocamento à esquerda.
 - Entrada $E(0)$ é desconsiderada no deslocamento à direita.
 - Se uma delas for **1**, operação aritmética não será válida.
 - Exemplo:
 - Deslocamento de 1 bit à esquerda:
 - ◇ $\textcircled{1}001 \rightarrow 0010$:
 - Deslocamento de 1 bit à direita:
 - ◇ $100\textcircled{1} \rightarrow 0100$:



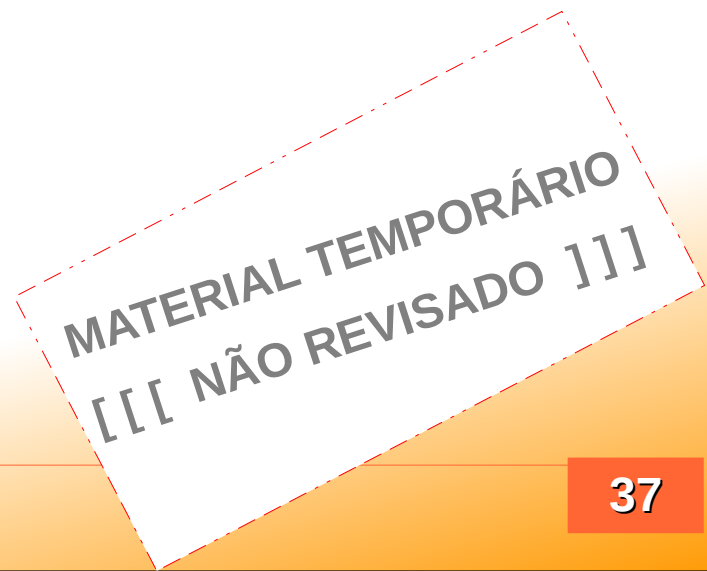
Deslocador: Operações aritméticas

- ◇ Nem sempre é válido realizar operações aritméticas com deslocadores
 - Entrada E(n) é desconsiderada no deslocamento à esquerda.
 - Entrada E(0) é desconsiderada no deslocamento à direita.
 - Se uma delas for 1, operação aritmética não será válida.
 - Exemplo:
 - Deslocamento de 1 bit à esquerda:
 - ◇ $\textcircled{1}001 \rightarrow 0010$: **9** → **2**
 - Deslocamento de 1 bit à direita:
 - ◇ $100\textcircled{1} \rightarrow 0100$: **9** → **4**

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Deslocador: Exemplo de uso

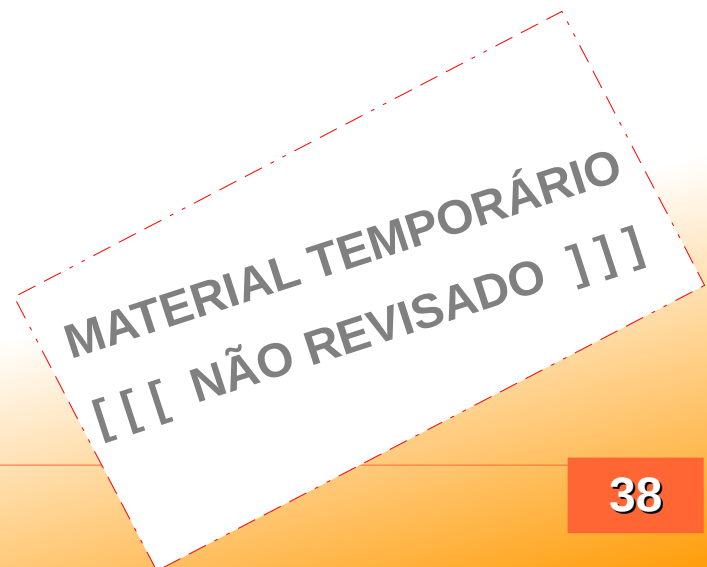
- ◇ **Termômetro mede a temperatura em Celsius:**
 - A temperatura é representada por 8 bits.
 - **Como representar a temperatura em Farenheit?**



Deslocador: Exemplo de uso

❖ Termômetro mede a temperatura em Celsius:

- A temperatura é representada por 8 bits.
- **Como representar a temperatura em Farenheit?**
 - Converter de Celsius para Farenheit:
 - ◆ $F = 1,8 * C + 32$
 - Arredondando o valor, obtém-se a seguinte expressão:
 - ◆ $F = 2 * C + 32$
 - São necessárias as seguintes operações:
 - ◆ Multiplica o valor por 2
 - ◆ Soma o resultado a 32



Deslocador: Exemplo de uso

❖ Termômetro mede a temperatura em Celsius:

- A temperatura é representada por 8 bits.
- Como representar a temperatura em Farenheit?
 - Converter de Celsius para Farenheit:
 - ◆ $F = 1,8 * C + 32$
 - Arredondando o valor, obtém-se a seguinte expressão:
 - ◆ $F = 2 * C + 32$
 - São necessárias as seguintes operações:
 - ◆ Multiplica o valor por 2
 - ◆ Soma o resultado a 32

■ Conclusão:

- ◆ Pegar valor de 8 bits que representa temperatura em Celsius.
- ◆ Usar deslocador à esquerda de 1 bit para multiplicar por 2.
- ◆ Usar somador para acrescentar o valor $32_{10} = 100000_2$ ao obtido do deslocador.
- ◆ **O resultado é a temperatura em Farenheit.**

Tipos de Deslocadores

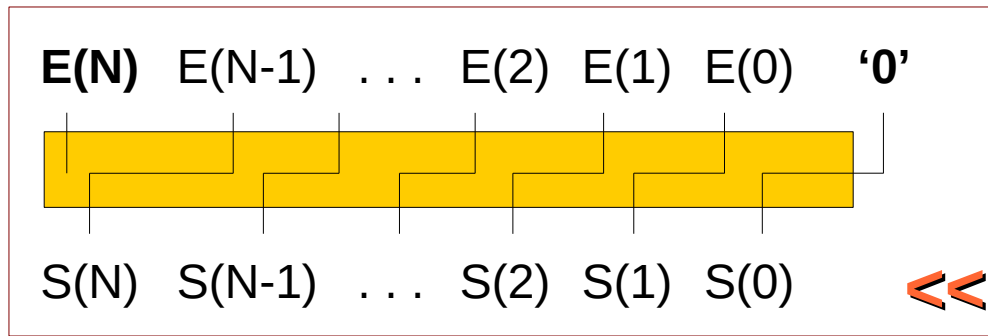
- ◇ Deslocador de bit à esquerda
- ◇ Deslocador de bit à direita
- ◇ Deslocador com *shift*
- ◇ Deslocador com escolha de lado
- ◇ Deslocador Barrel



Tipos de Deslocadores

◇ Deslocador de bit à esquerda

- Adiciona **0** à ao bit menos significativo **S(0)**

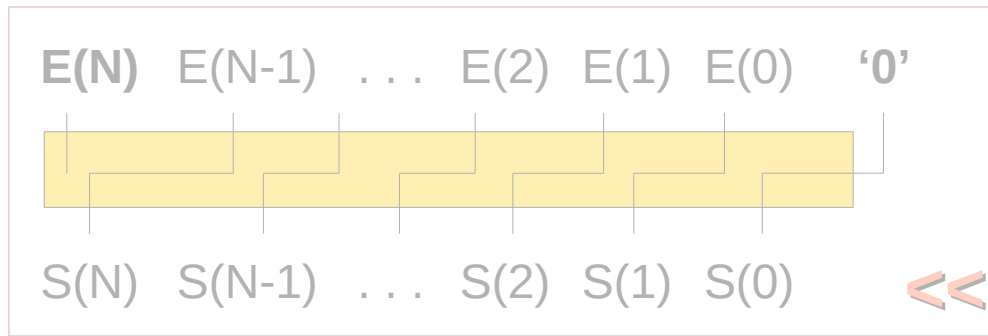


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Tipos de Deslocadores

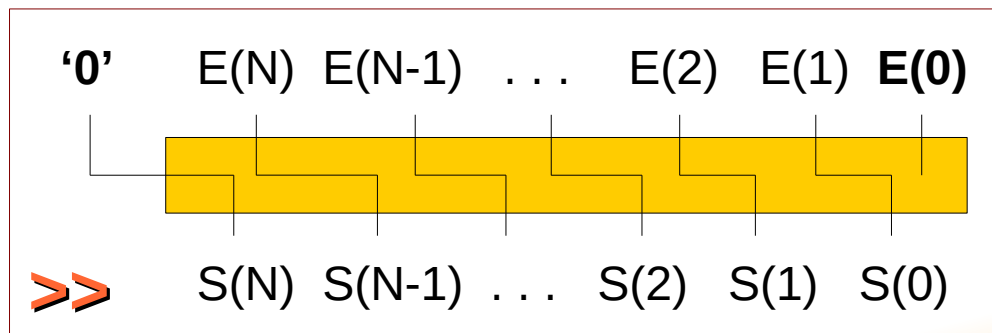
◇ Deslocador de bit à esquerda

- Adiciona **0** à ao bit menos significativo **S(0)**



◇ Deslocador de bit à direita

- Adiciona **0** à ao bit mais significativo **S(N)**

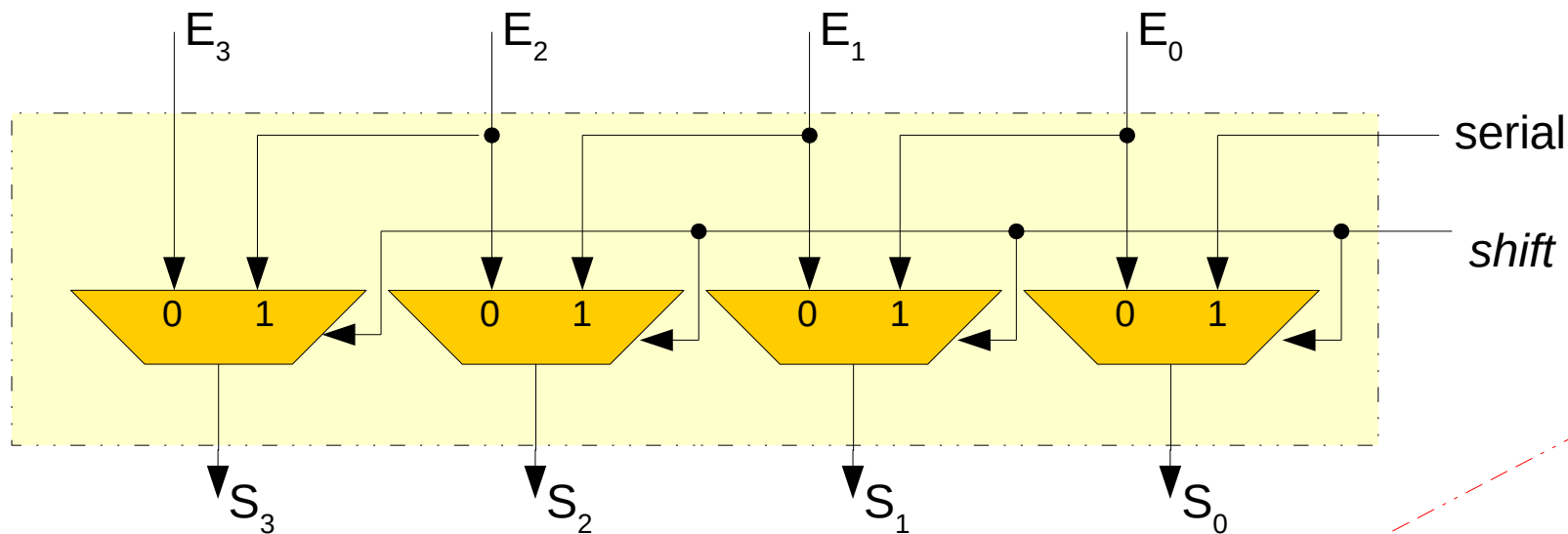


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Tipos de Deslocadores

◇ Deslocador com *shift* (para esquerda)

- Variável *shift* permite definir se deslocamento será ou não realizado.
- Se *shift* for igual a **1**: **deslocamento à esquerda**.
- Se *shift* for igual a **0**: a **saída** será **igual à entrada**.

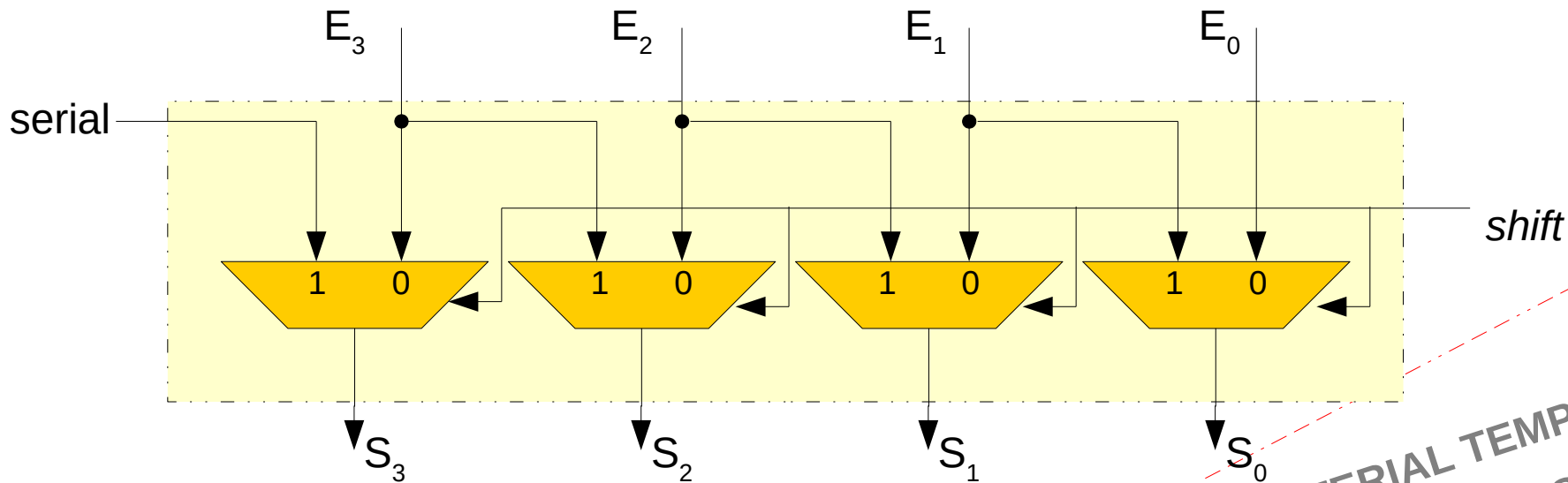


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Tipos de Deslocadores

◇ Deslocador com *shift* (para direita)

- Variável *shift* permite definir se deslocamento será ou não realizado.
- O mesmo deslocador pode ser feito para **direita**.
- Se *shift* for igual a **1**: **deslocamento à direita**.
- Se *shift* for igual a **0**: a **saída** será **igual à entrada**.

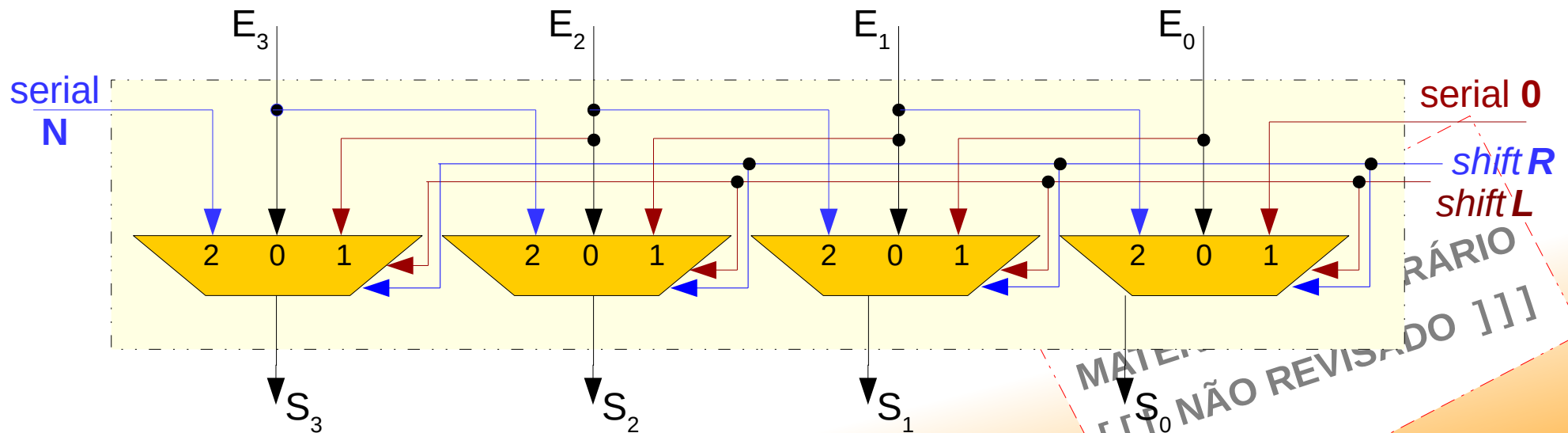


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Tipos de Deslocadores

◇ Deslocador com escolha de lado

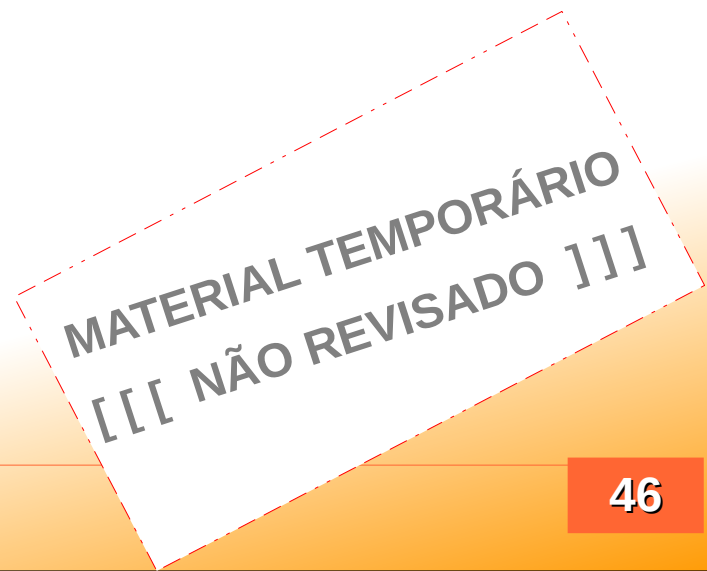
- Permite escolher o lado do deslocamento (direita ou esquerda)
- Possui variáveis de controle: *shiftL* e *shiftR*
- Se **shiftL** for igual a **1**: deslocamento à **esquerda**
- Se **shiftR** for igual a **1**: deslocamento à **direita**
- Se **shiftR** e **shiftL** forem igual a **0**: a **saída** será **igual à entrada**



Tipos de Deslocadores

◇ Deslocador Barrel

- Permite escolher nº de bits que serão deslocados.
- Formado através de deslocadores em série.
- Tamanho dos deslocadores depende do tamanho da palavra binária de entrada.
 - Palavras binárias de 2 bits
 - Ex.: 11
 - Só é possível realizar **um** deslocamento
 - 1 deslocador de 1 bit



Tipos de Deslocadores

◇ Deslocador Barrel

- Permite escolher nº de bits que serão deslocados.
- Formado através de deslocadores em série.
- Tamanho dos deslocadores depende do tamanho da palavra binária de entrada.
 - Palavras binárias de 2 bits
 - Ex.: 11
 - Só é possível realizar **um** deslocamento
 - 1 deslocador de 1 bit
 - Palavras binárias de 4 bits
 - Ex.: 1111
 - É possível realizar **três** deslocamentos
 - 1 deslocador de 1 bit + 1 deslocador de 2 bits

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Tipos de Deslocadores

◇ Deslocador Barrel

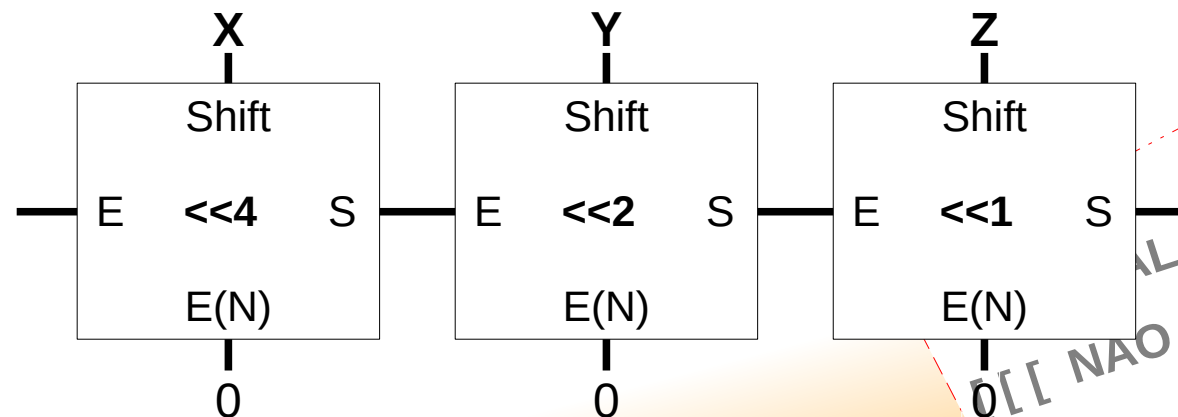
- Permite escolher nº de bits que serão deslocados.
- Formado através de deslocadores em série.
- Tamanho dos deslocadores depende do tamanho da palavra binária de entrada.
 - Palavras binárias de 2 bits
 - Ex.: 11
 - Só é possível realizar **um** deslocamento
 - 1 deslocador de 1 bit
 - Palavras binárias de 4 bits
 - Ex.: 1111
 - É possível realizar **três** deslocamentos
 - 1 deslocador de 1 bit + 1 deslocador de 2 bits
 - Palavras binárias de 8 bits
 - Ex.: 11111111
 - Pode realizar **sete** deslocamentos
 - 1 deslocador de 1 bit +
1 deslocador de 2 bits + 1 deslocador de 4 bits

MATERIAL TEMPORÁRIO
NÃO REVISADO !!!

Tipos de Deslocadores

◇ Deslocador Barrel

- Cada um dos deslocadores tem uma entrada *shift*:
 - X, Y, Z, W, ...
- As entradas *shift* indicam se haverá deslocamento para aquela entrada ou não
- Exemplo: **XYZ = 010 = 2₁₀**
 - Apenas o deslocador de 2 bits deslocará a sua entrada
 - Serão realizados então 2 deslocamentos



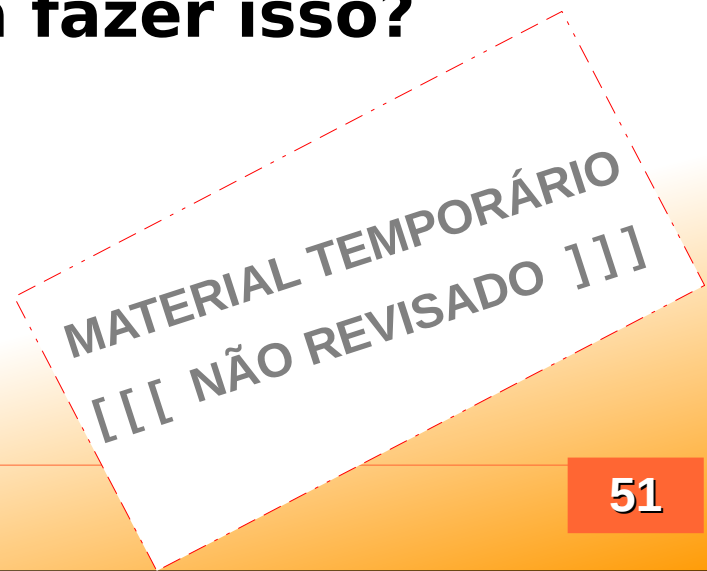
Exemplo

- ◇ Como criar um circuito que fornece a média de quatro números binários de 4 bits cada?



Exemplo

- ◇ **Como criar um circuito que fornece a média de quatro números binários de 4 bits cada?**
 - Como é obtida a média de 4 números?
 - $M = (A + B + C + D)/4$
 - É preciso:
 - Somar as quatro entradas
 - Dividir o valor resultante por 4
 - **O que se pode utilizar para fazer isso?**

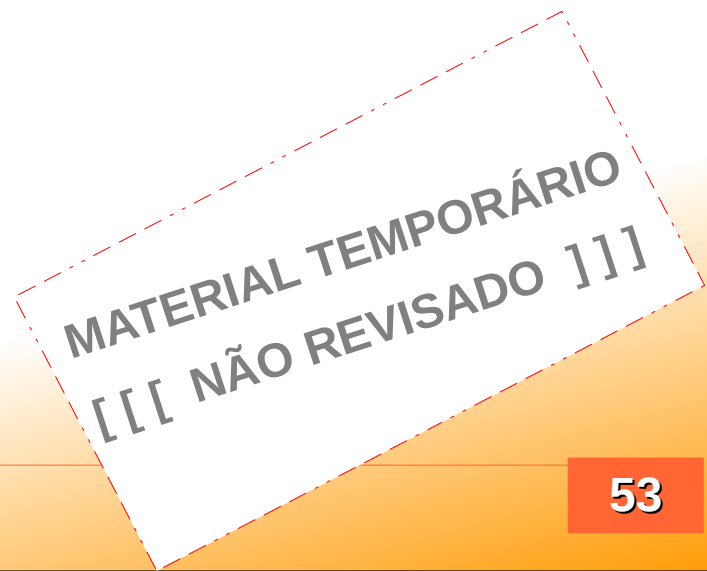


Exemplo

- ◇ **Como criar um circuito que fornece a média de quatro números binários de 4 bits cada?**
 - Como é obtida a média de 4 números?
 - $M = (A + B + C + D)/4$
 - É preciso:
 - Somar as quatro entradas
 - Dividir o valor resultante por 4
 - O que se pode utilizar para fazer isso?
 - **Somador de 4 bits**
 - **Deslocador de 2 bits à direita**

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador



Comparador

◇ Conceito:

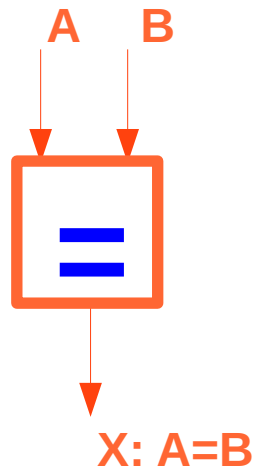
- São circuitos que comparam dois números binários



Comparador

◇ Conceito:

- São circuitos que comparam dois números binários
- Considerando palavras binárias de 1 bit:
 - **Comparador de Igualdade** (de 1 bit):

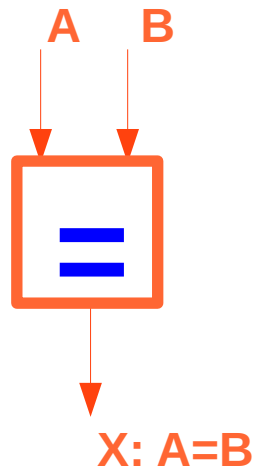


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador

◇ Comparador de Igualdade:

- de 1 bit:



A	B	X: A=B
0	0	1
0	1	0
1	0	0
1	1	1

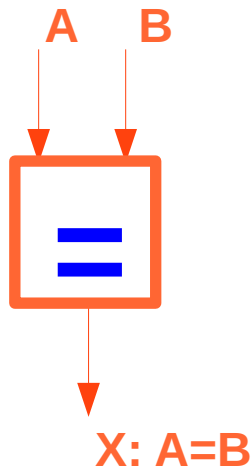
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador

◇ Comparador de Igualdade:

- de 1 bit:

$$X = \overline{A} \cdot \overline{B} + A \cdot B = \overline{A \oplus B}$$



A	B	X: A=B
0	0	1
0	1	0
1	0	0
1	1	1

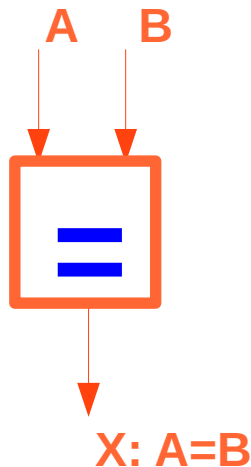
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador

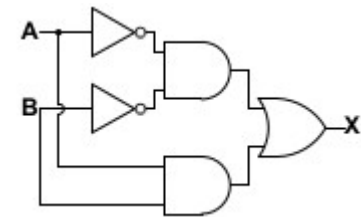
◇ Comparador de Igualdade:

- de 1 bit:

$$X = \overline{A} \cdot \overline{B} + A \cdot B = \overline{A \oplus B}$$



A	B	X: A=B
0	0	1
0	1	0
1	0	0
1	1	1



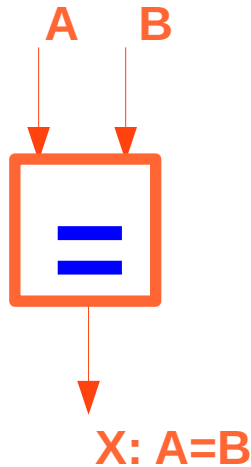
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador

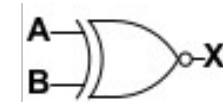
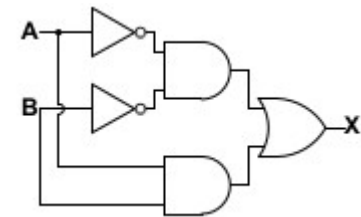
◇ Comparador de Igualdade:

- de 1 bit:

$$X = \overline{A} \cdot \overline{B} + A \cdot B = \overline{A \oplus B}$$



A	B	X: A=B
0	0	1
0	1	0
1	0	0
1	1	1

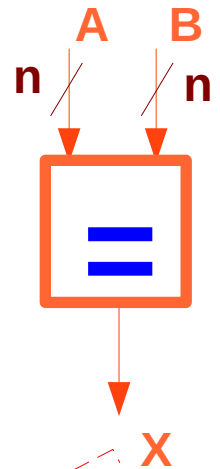


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador

◇ Comparador de Igualdade de N bits:

- Comparador de duas palavras binárias de N bits:
 - Comparação bit-a-bit das palavras binárias.
 - A_0 com B_0 , A_1 com B_1 , e assim por diante.

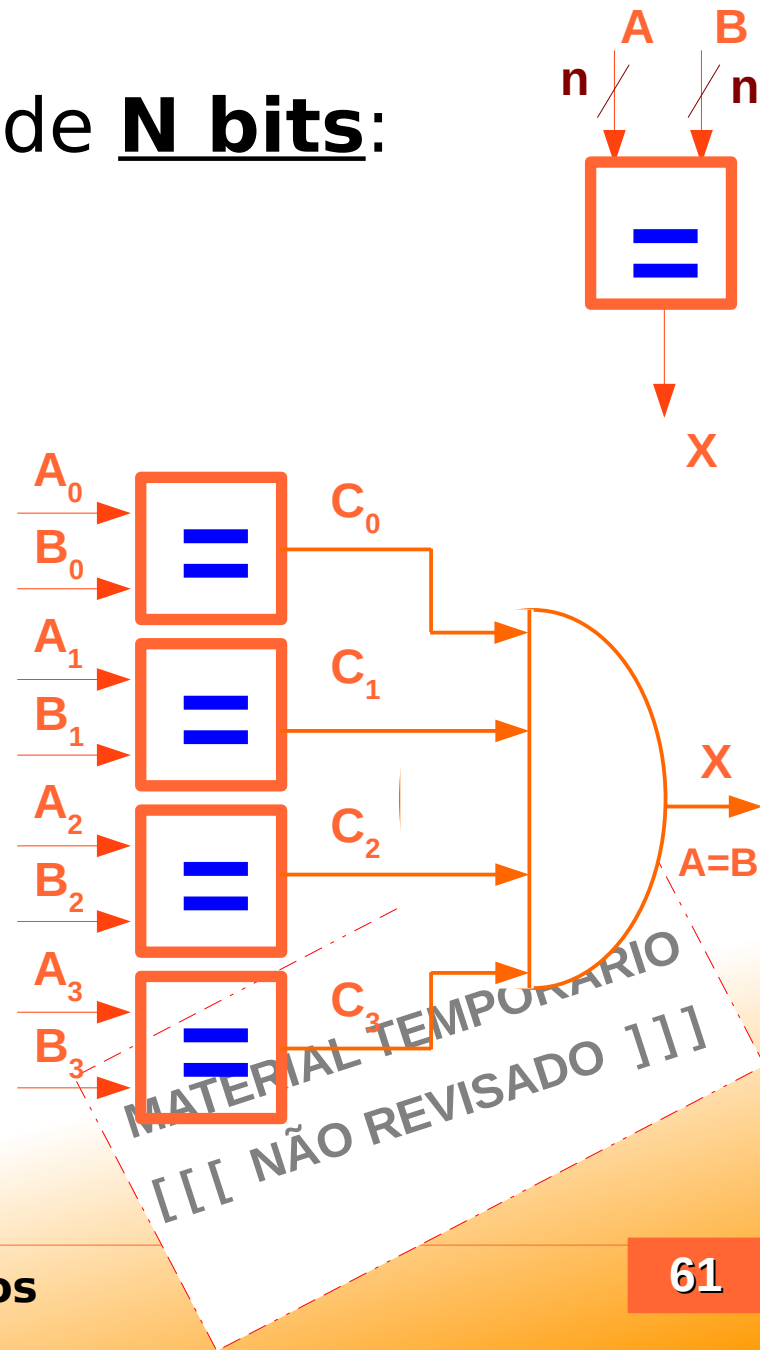


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador

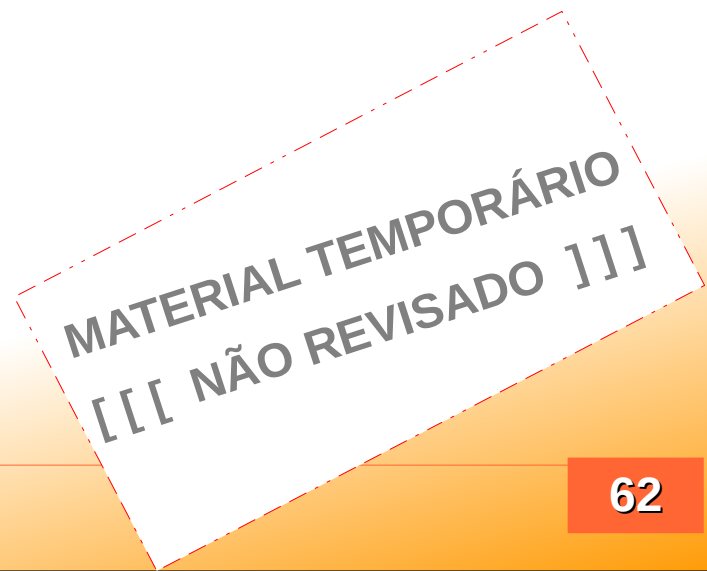
◇ Comparador de Igualdade de N bits:

- N comparadores de 1 bit:
 - Depois, os resultados passam por uma porta AND para saber se TODAS as comparações tiveram o resultado '1', ou seja, se todos os bits eram iguais.



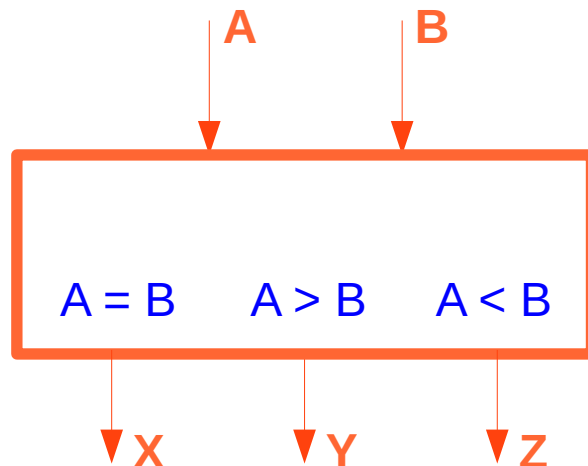
Comparador de 1 bit com 3 saídas

- ◇ Compara se duas palavras binárias são iguais.
- ◇ Se forem diferentes, indica qual é a maior.
- ◇ 3 Saídas:
 - $X : A = B$
 - $Y : A > B$
 - $Z : A < B$



Comparador de 1 bit com 3 saídas

- ◇ Compara se duas palavras binárias são iguais.
- ◇ Se forem diferentes, indica qual é a maior.
- ◇ 3 Saídas:
 - $X : A = B$
 - $Y : A > B$
 - $Z : A < B$

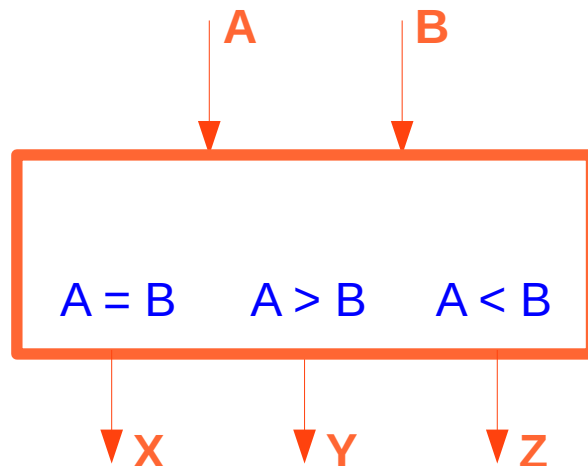


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de 1 bit com 3 saídas

- ◇ Compara se duas palavras binárias são iguais.
- ◇ Se forem diferentes, indica qual é a maior.
- ◇ 3 Saídas:

- $X : A = B$
- $Y : A > B$
- $Z : A < B$



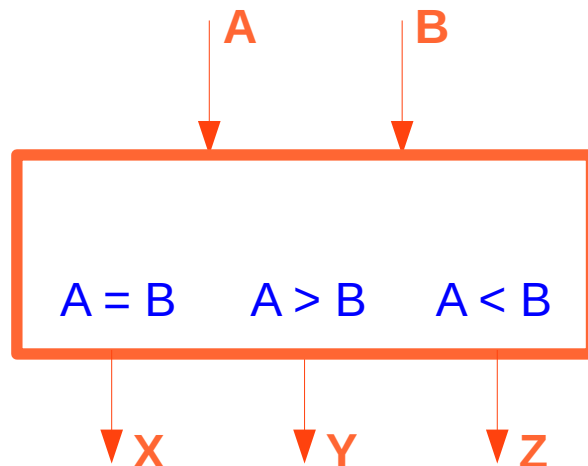
A	B	X : $A = B$	Y : $A > B$	Z : $A < B$
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de 1 bit com 3 saídas

- ◇ Compara se duas palavras binárias são iguais.
- ◇ Se forem diferentes, indica qual é a maior.
- ◇ 3 Saídas:

- $X : A = B$
- $Y : A > B$
- $Z : A < B$



A	B	X : A = B	Y : A > B	Z : A < B
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

$$X = \bar{A} \cdot \bar{B} + A \cdot B = A \oplus B$$

$$Y = A \cdot \bar{B}$$

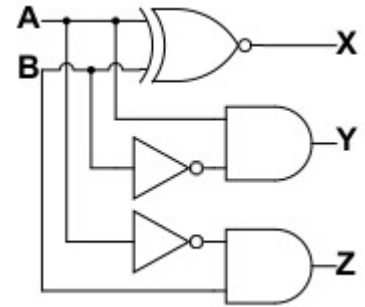
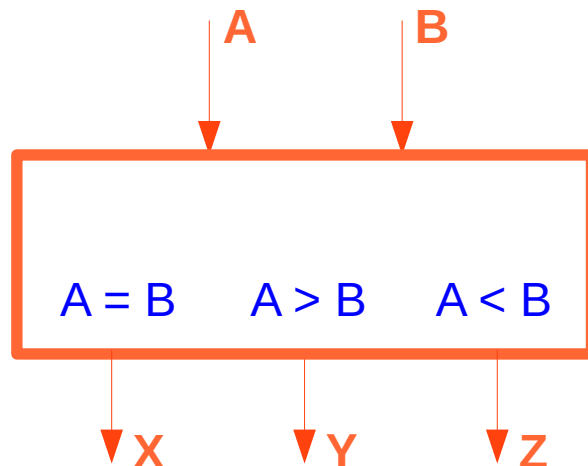
$$Z = \bar{A} \cdot B$$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de 1 bit com 3 saídas

- ◇ Compara se duas palavras binárias são iguais.
- ◇ Se forem diferentes, indica qual é a maior.
- ◇ 3 Saídas:

- $X : A = B$
- $Y : A > B$
- $Z : A < B$



A	B	X : $A = B$	Y : $A > B$	Z : $A < B$
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

$$X = \bar{A} \cdot \bar{B} + A \cdot B = A \oplus B$$

$$Y = A \cdot \bar{B}$$

$$Z = \bar{A} \cdot B$$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Como saber qual das seguintes palavras binárias é maior?

- $A = 1001$
- $B = 0110$

A_i B_i	
A	B
I_(A=B)	O_(A=B)
I_(A>B)	O_(A>B)
I_(A<B)	O_(A<B)

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Como saber qual das seguintes palavras binárias é maior?

■ $A = 1001$

■ $B = 0110$

A_i B_i	
A	B
I_(A=B)	O_(A=B)
I_(A>B)	O_(A>B)
I_(A<B)	O_(A<B)

◇ Passos:

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Como saber qual das seguintes palavras binárias é maior?

■ $A = 1001$

■ $B = 0110$

A_i B_i	
A	B
I_(A=B)	O_(A=B)
I_(A>B)	O_(A>B)
I_(A<B)	O_(A<B)

◇ Passos:

1) Identificar o bit mais significativo das duas palavras binárias

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Como saber qual das seguintes palavras binárias é maior?

■ $A = 1001$

■ $B = 0110$

A_i B_i	
A	B
I_(A=B)	O_(A=B)
I_(A>B)	O_(A>B)
I_(A<B)	O_(A<B)

◇ Passos:

1) Identificar o bit mais significativo das duas palavras binárias

2) Se $A_N > B_N$, então $A > B$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Como saber qual das seguintes palavras binárias é maior?

■ $A = 1001$

■ $B = 0110$

A_i	B_i
A	B
I_(A=B)	O_(A=B)
I_(A>B)	O_(A>B)
I_(A<B)	O_(A<B)

◇ Passos:

1) Identificar o bit mais significativo das duas palavras binárias

2) Se $A_N > B_N$, então $A > B$

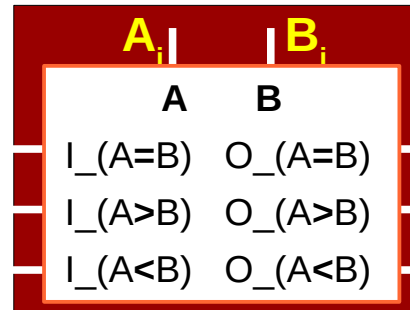
3) Se $A_N < B_N$, então $A < B$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

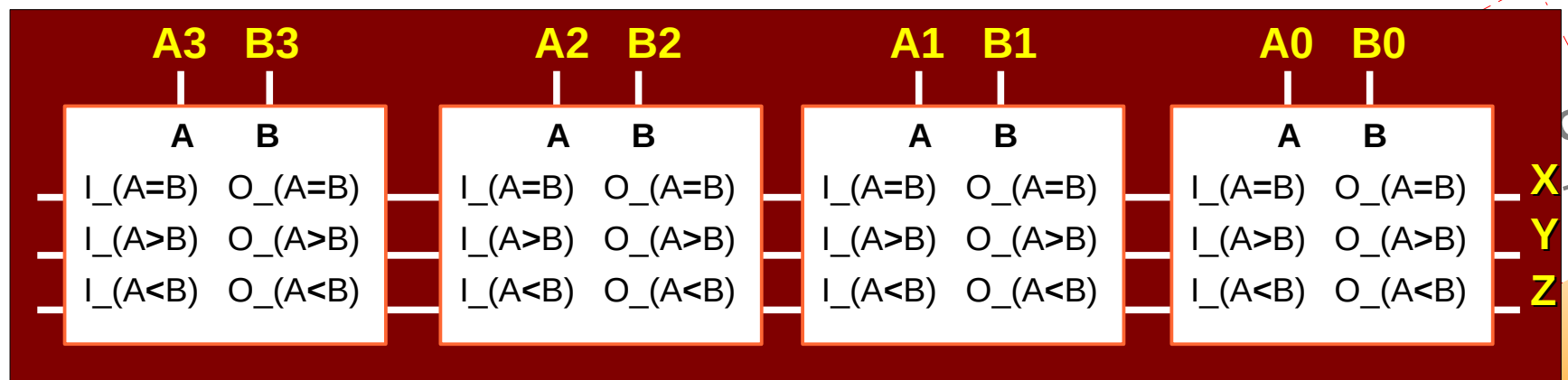
◇ Como saber qual das seguintes palavras binárias é maior?

- $A = 1001$
- $B = 0110$

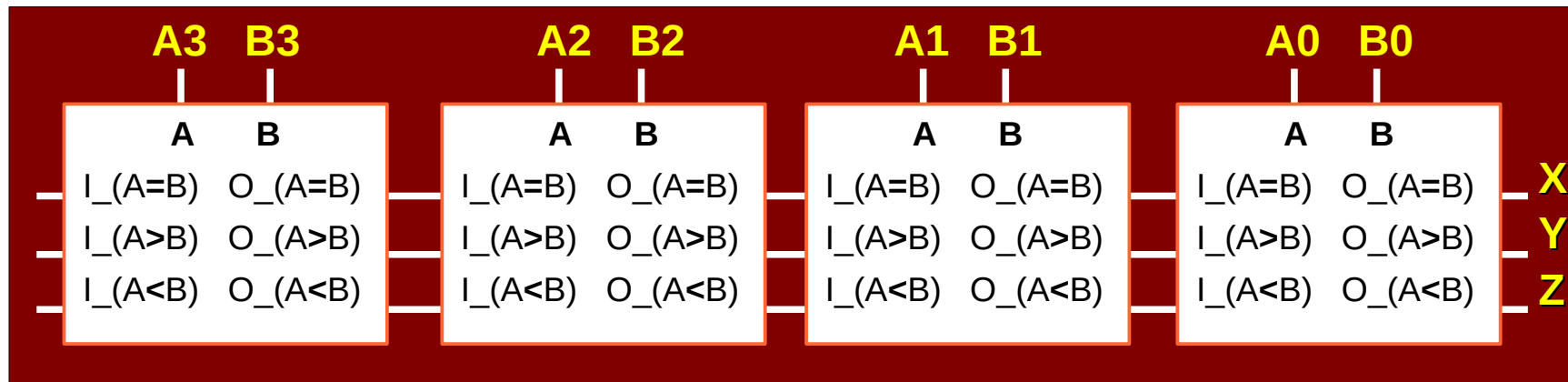


◇ Passos:

- 1) Identificar o bit mais significativo das duas palavras binárias
- 2) Se $A_N > B_N$, então $A > B$
- 3) Se $A_N < B_N$, então $A < B$
- 4) Se $A_N = B_N$, então devemos testar o bit seguinte



Comparador de N bits com 3 saídas

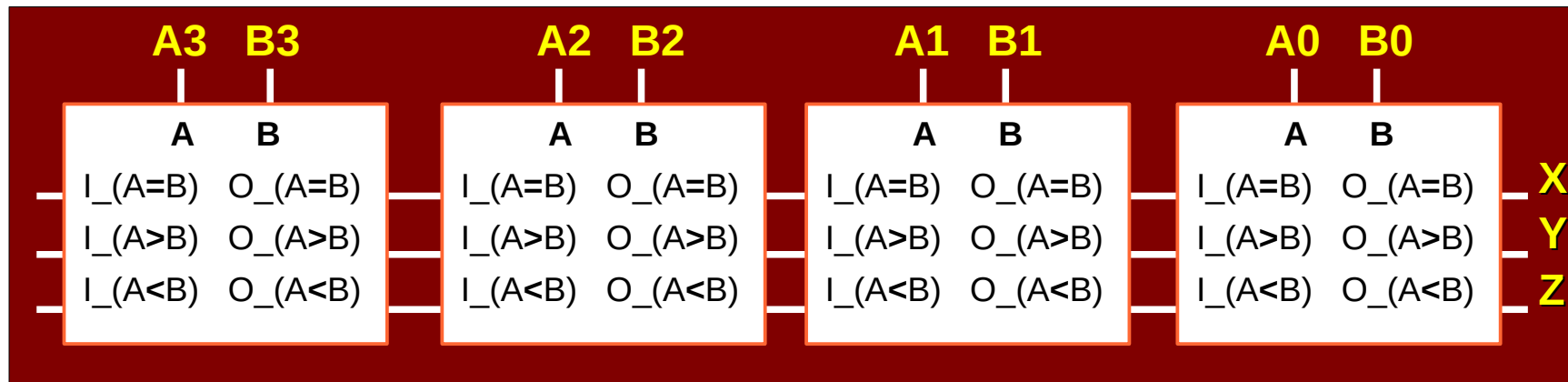


◇ Se $I_{A>B} = 1 \rightarrow$ no estágio anterior: **A > B**

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas



◇ Se $I_{A>B} = 1 \rightarrow$ no estágio anterior: **A > B**

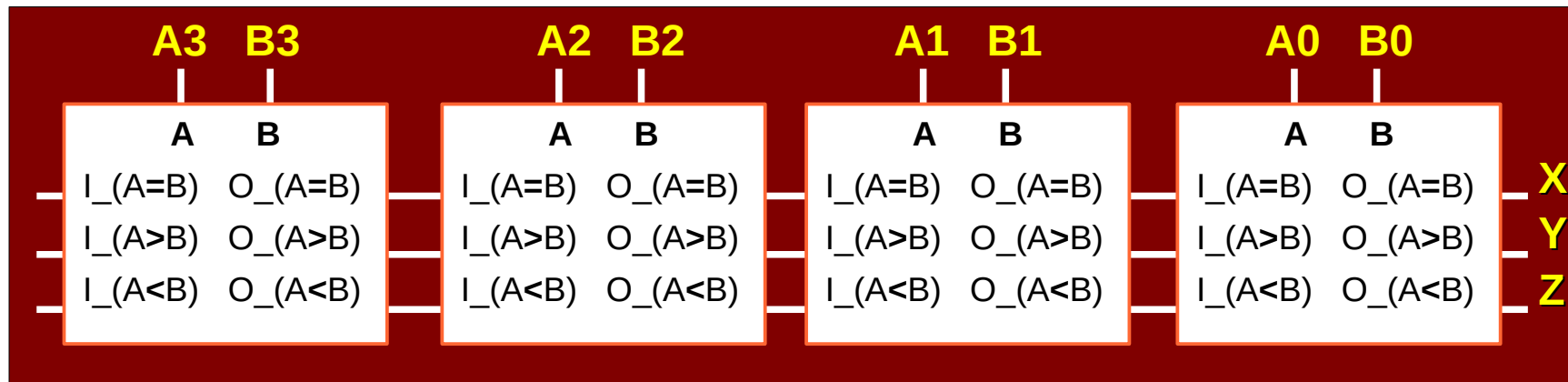
- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B} = 1 \rightarrow$ no estágio anterior: **A < B**

- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas



◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

- **É preciso analisar o bit deste estágio**

MATERIAL TEMPORÁRIO
NÃO REVISADO !!!

Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

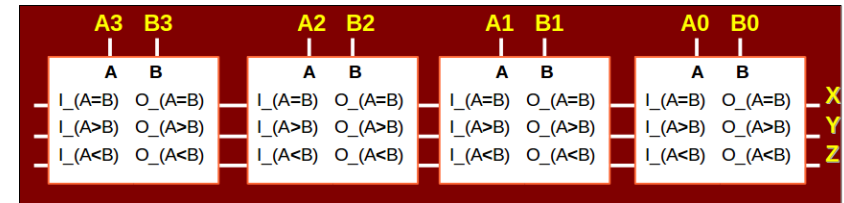
- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

- **É preciso analisar o bit deste estágio**



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

■ $O_{A>B} = I_{A>B} = 1$

■ $O_{A=B} = I_{A=B} = 0$

■ $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

■ $O_{A>B} = I_{A>B} = 0$

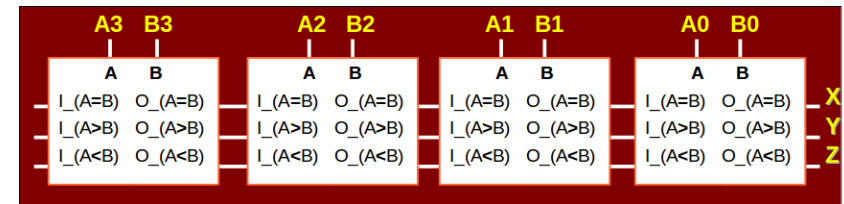
■ $O_{A=B} = I_{A=B} = 0$

■ $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

■ **É preciso analisar o bit deste estágio**

◇ **CONCLUSÃO:**



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

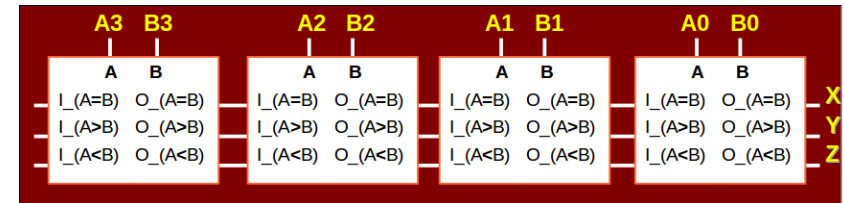
- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

- **É preciso analisar o bit deste estágio**

◇ **CONCLUSÃO:**

- **$O_{A>B} = 1$:**



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

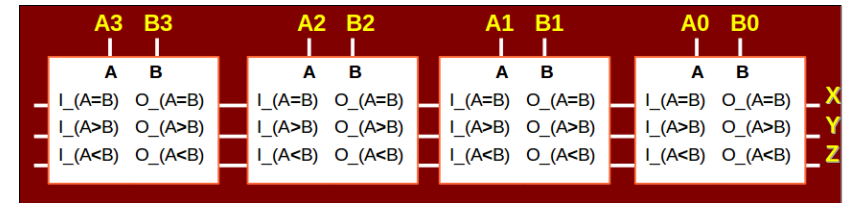
◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

- **É preciso analisar o bit deste estágio**

◇ **CONCLUSÃO:**

- **$O_{A>B} = 1$:**

— [Se ($I_{A>B}=1$)] OU [se ($I_{A=B}=1$) E, neste nível, o bit de A for **maior** que o bit de B]



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

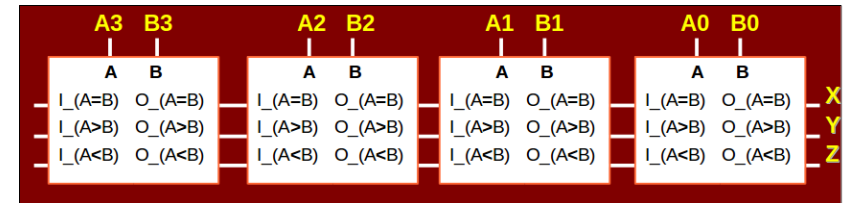
- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

- **É preciso analisar o bit deste estágio**

◇ **CONCLUSÃO:**

- **$O_{A>B} = 1$:**
 - [Se ($I_{A>B}=1$)] OU [se ($I_{A=B}=1$) E, neste nível, o bit de A for **maior** que o bit de B]
- **$O_{A<B} = 1$:**



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

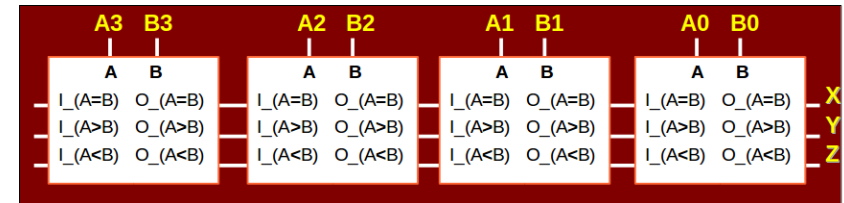
- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

- **É preciso analisar o bit deste estágio**

◇ **CONCLUSÃO:**

- **$O_{A>B} = 1$:**
 - Se ($I_{A>B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for **maior** que o bit de B.
- **$O_{A<B} = 1$:**
 - Se ($I_{A<B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for **menor** que o bit de B.



Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: $A > B$

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: $A < B$

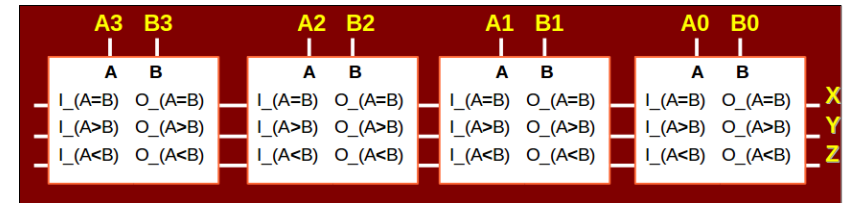
- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: $A = B$

- **É preciso analisar o bit deste estágio**

◇ **CONCLUSÃO:**

- $O_{A>B} = 1$:
 - Se ($I_{A>B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for **maior** que o bit de B.
- $O_{A<B} = 1$:
 - Se ($I_{A<B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for **menor** que o bit de B.
- $O_{A=B} = 1$:



Comparador de N bits com 3 saídas

◇ Se $I_{A>B}=1 \rightarrow$ no estágio anterior: **A > B**

- $O_{A>B} = I_{A>B} = 1$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 0$

◇ Se $I_{A<B}=1 \rightarrow$ no estágio anterior: **A < B**

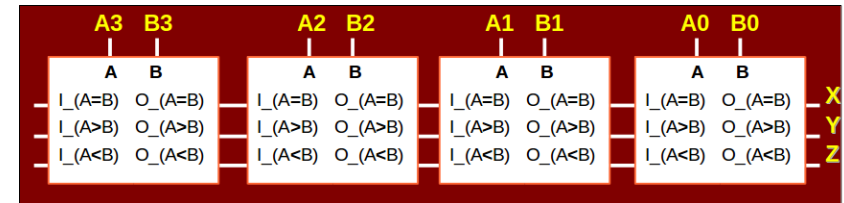
- $O_{A>B} = I_{A>B} = 0$
- $O_{A=B} = I_{A=B} = 0$
- $O_{A<B} = I_{A<B} = 1$

◇ Se $I_{A=B}=1 \rightarrow$ no estágio anterior: **A = B**

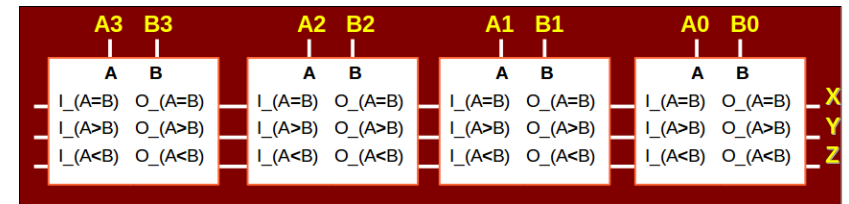
- **É preciso analisar o bit deste estágio**

◇ **CONCLUSÃO:**

- **$O_{A>B} = 1$:**
 - Se ($I_{A>B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for **maior** que o bit de B.
- **$O_{A<B} = 1$:**
 - Se ($I_{A<B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for **menor** que o bit de B.
- **$O_{A=B} = 1$:**
 - Se ($I_{A=B}=1$) E, neste nível, o bit de A for **igual** que o bit de B.



Comparador de N bits com 3 saídas



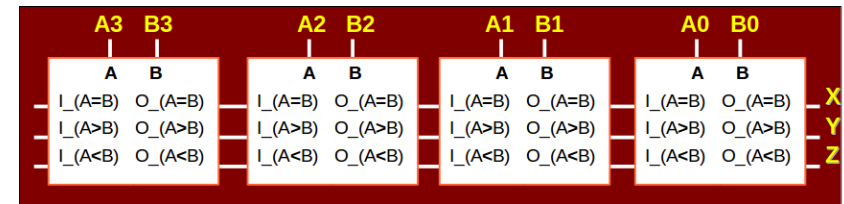
- ◇ $O_{A>B} = 1$: Se ($I_{A>B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for maior que o bit de B.
- ◇ $O_{A<B} = 1$: Se ($I_{A<B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for menor que o bit de B.
- ◇ $O_{A=B} = 1$: Se ($I_{A=B} = 1$) E, neste nível, o bit de A for igual que o bit de B.

◇ Resultado:

- $O_{A>B} =$
- $O_{A<B} =$
- $O_{A=B} =$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas



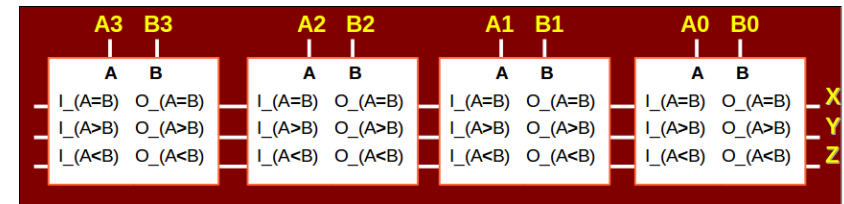
- ◇ $O_{A>B} = 1$: Se ($I_{A>B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for maior que o bit de B.
- ◇ $O_{A<B} = 1$: Se ($I_{A<B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for menor que o bit de B.
- ◇ $O_{A=B} = 1$: Se ($I_{A=B} = 1$) E, neste nível, o bit de A for igual que o bit de B.

◇ Resultado:

- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \bar{B})$
- $O_{A<B} =$
- $O_{A=B} =$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas



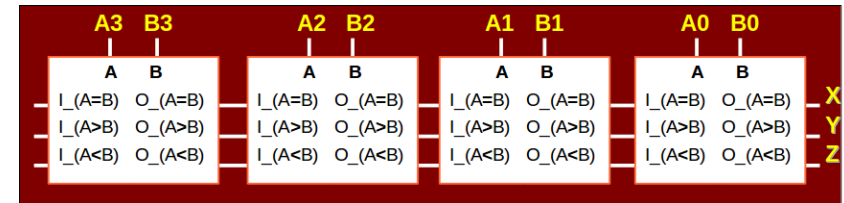
- ◇ $O_{A>B} = 1$: Se ($I_{A>B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for maior que o bit de B.
- ◇ $O_{A<B} = 1$: Se ($I_{A<B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for menor que o bit de B.
- ◇ $O_{A=B} = 1$: Se ($I_{A=B} = 1$) E, neste nível, o bit de A for igual que o bit de B.

◇ Resultado:

- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot \overline{A} \cdot B)$
- $O_{A<B} = I_{A<B} + (I_{A=B} \cdot A \cdot \overline{B})$
- $O_{A=B} =$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas



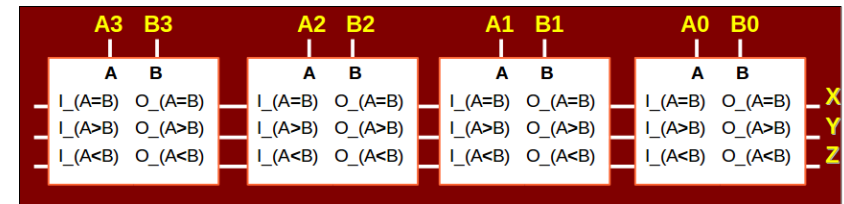
- ◇ $O_{A>B} = 1$: Se ($I_{A>B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for maior que o bit de B.
- ◇ $O_{A<B} = 1$: Se ($I_{A<B}=1$) OU se ($I_{A=B}=1$) E, neste nível, o bit de A for menor que o bit de B.
- ◇ $O_{A=B} = 1$: Se ($I_{A=B}=1$) E, neste nível, o bit de A for igual que o bit de B.

◇ Resultado:

- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \bar{B})$
- $O_{A<B} = I_{A<B} + (I_{A=B} \cdot \bar{A} \cdot B)$
- $O_{A=B} = I_{A=B} \cdot A \oplus B$

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

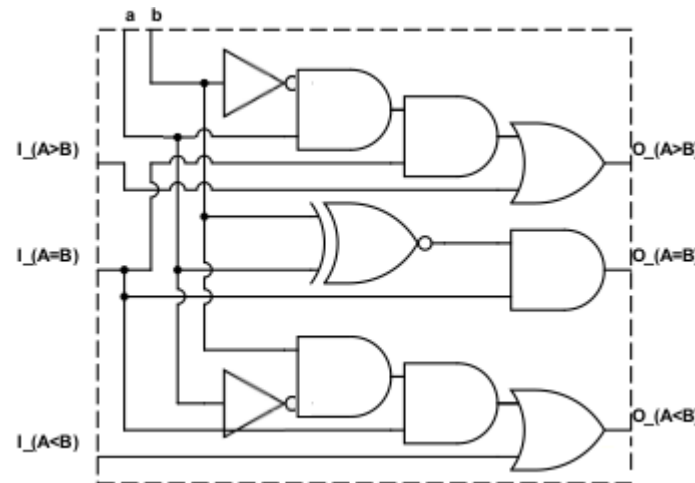
Comparador de N bits com 3 saídas



- ◇ $O_{A>B} = 1$: Se ($I_{A>B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for maior que o bit de B.
- ◇ $O_{A<B} = 1$: Se ($I_{A<B} = 1$) OU se ($I_{A=B} = 1$) E, neste nível, o bit de A for menor que o bit de B.
- ◇ $O_{A=B} = 1$: Se ($I_{A=B} = 1$) E, neste nível, o bit de A for igual que o bit de B.

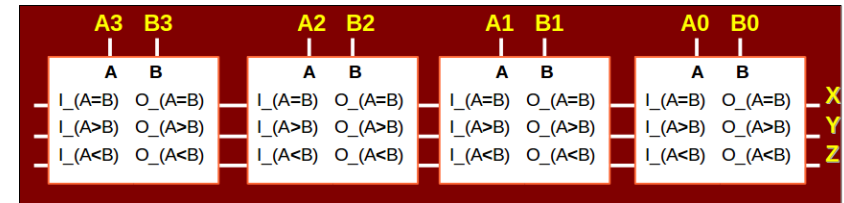
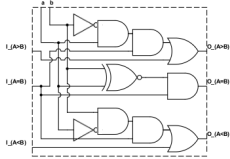
◇ Resultado:

- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \bar{B})$
- $O_{A<B} = I_{A<B} + (I_{A=B} \cdot \bar{A} \cdot B)$
- $O_{A=B} = I_{A=B} \cdot A \oplus B$



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

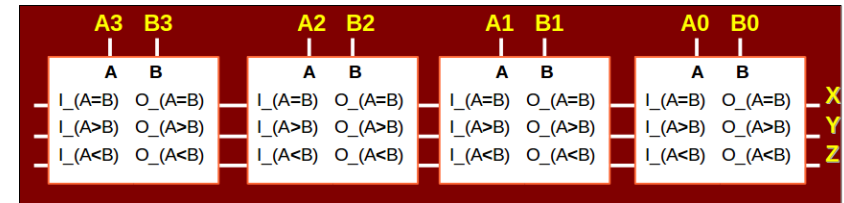
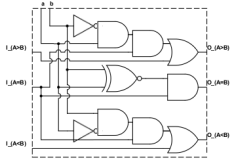
Comparador de N bits com 3 saídas



- ◊ Que valores $I_{A>B}$, $I_{A=B}$ e $I_{A<B}$ devem ter estágio no mais significativo para que o processo de comparação seja correto?

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

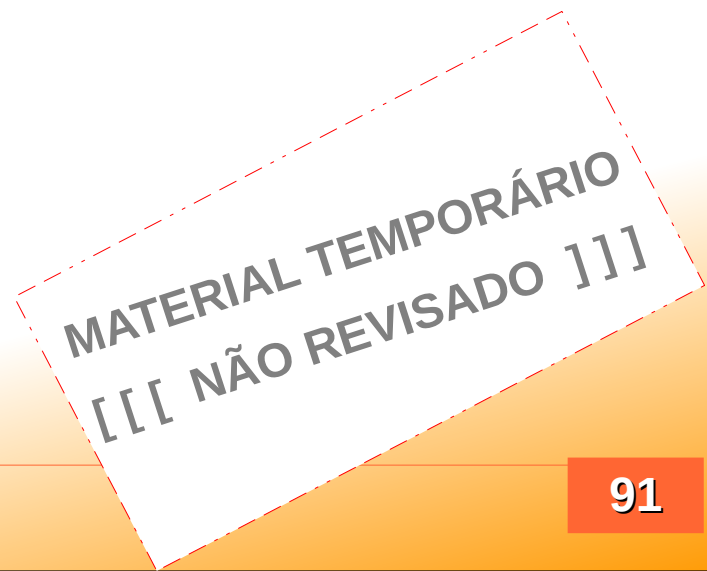


- ◇ Que valores $I_{A>B}$, $I_{A=B}$ e $I_{A<B}$ devem ter estágio no mais significativo para que o processo de comparação seja correto?
 - No nível mais significativo, a análise deve ser a seguinte:
 - Houve um teste anteriormente ($A_{N+1} = B_{N+1} = 0$)
 - Os bits eram iguais
 - Então, as entradas devem ser as seguintes:
 - $I_{A>B} = 0$
 - $I_{A=B} = 1$
 - $I_{A<B} = 0$
 - Assim, o teste nos níveis menos significativos continua!

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

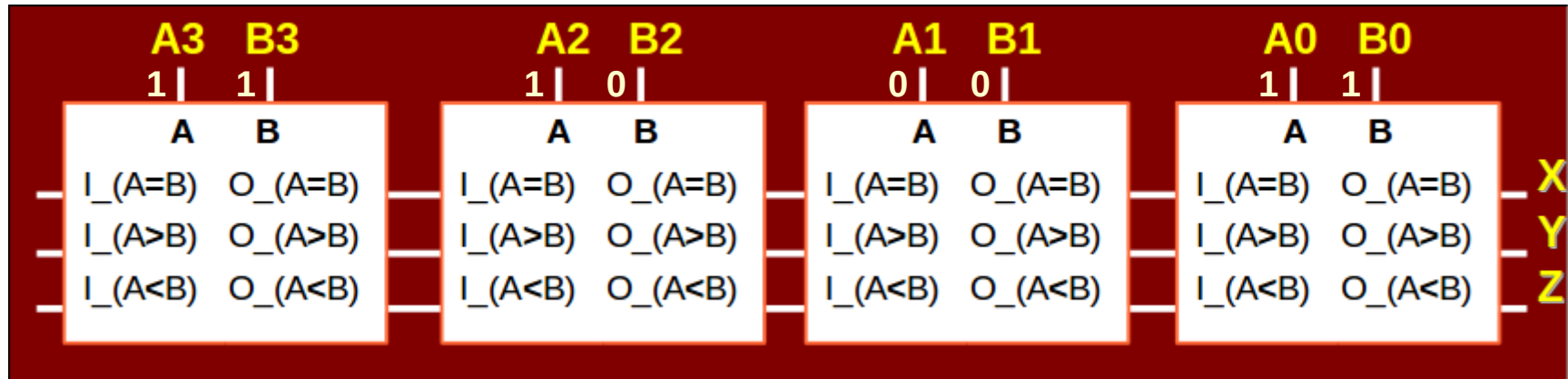
Comparador de N bits com 3 saídas

- ◇ Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?



Comparador de N bits com 3 saídas

- Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

- Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?

	A3	B3
	1	1
	A	B
1	I_(A=B)	O_(A=B)
0	I_(A>B)	O_(A>B)
0	I_(A<B)	O_(A<B)

1 1

1 0

0 0

1 1

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

- ♦ Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?

	A3	B3
	1	1
	A	B
1	I_(A=B)	O_(A=B)
0	I_(A>B)	O_(A>B)
0	I_(A<B)	O_(A<B)

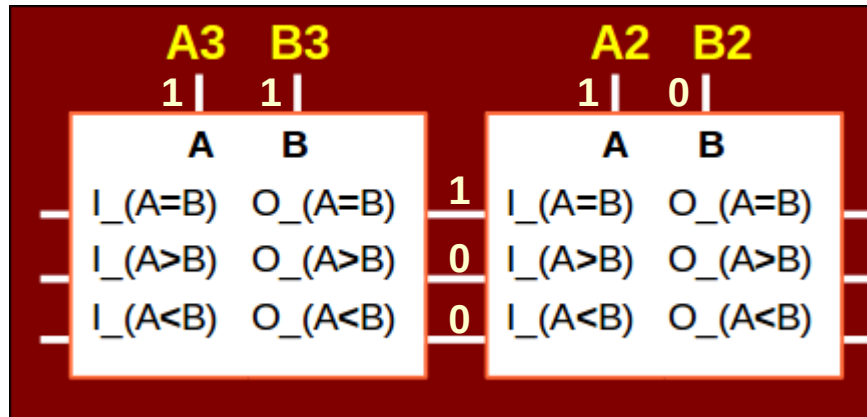
- $O_{A=B} = I_{A=B} \cdot \overline{A \oplus B} = 1 \cdot 1 \oplus 1 = 1 \cdot 0 = 1 \cdot 1 = 1$
- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \overline{B}) = 0 + (1 \cdot 1 \cdot 0) = 0 + 0 = 0$
- $O_{A<B} = I_{A<B} + (I_{A=B} \cdot \overline{A} \cdot B) = 0 + (1 \cdot 0 \cdot 1) = 0 + 0 = 0$

A3		B3	
1		1	
A		B	
I_(A=B)	O_(A=B)	1	
I_(A>B)	O_(A>B)	0	
I_(A<B)	O_(A<B)	0	

10
111

Comparador de N bits com 3 saídas

- Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?



1 1

1 0

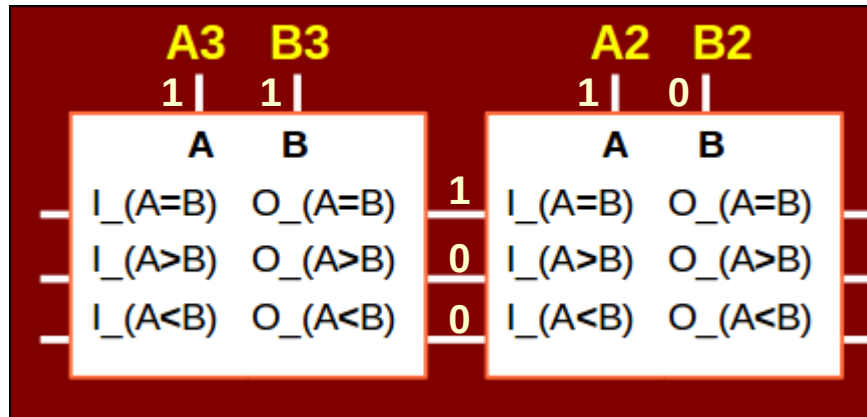
0 0

1 1

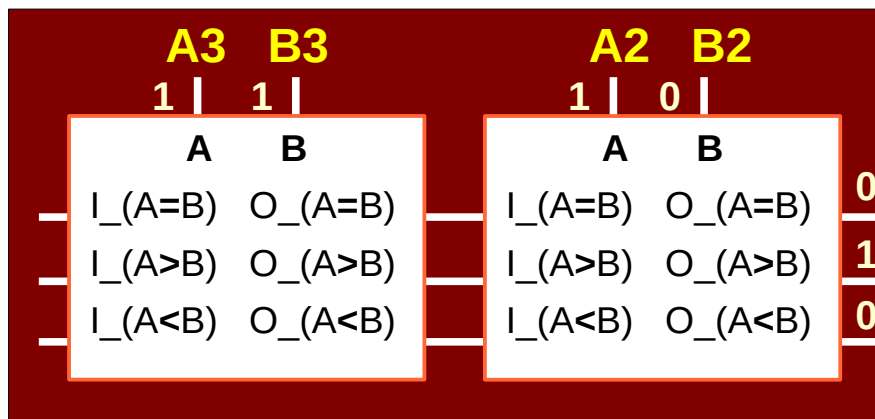
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

- ◇ Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?

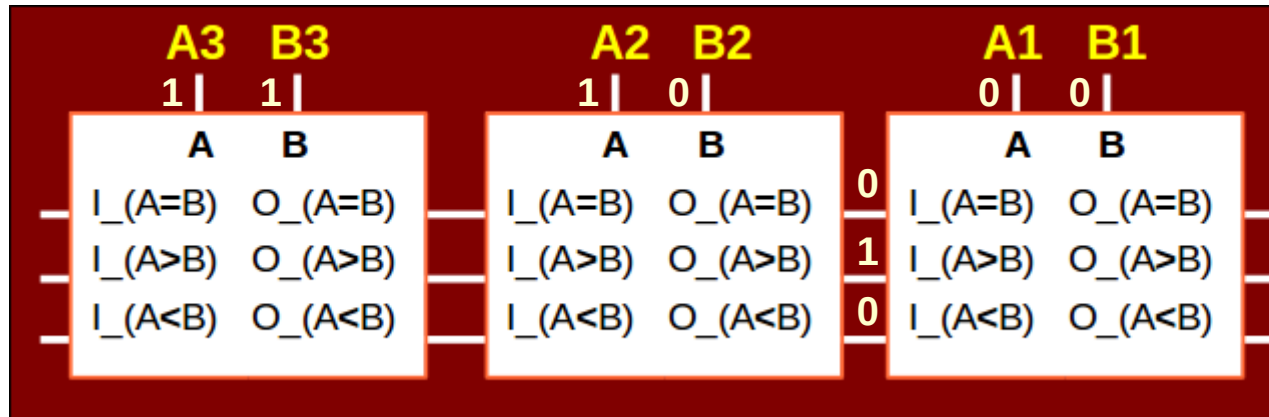


- $O_{A=B} = I_{A=B} \cdot \overline{A \oplus B} = 1 \cdot \overline{1 \oplus 0} = 1 \cdot \overline{1} = 1 \cdot 0 = 0$
- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \overline{B}) = 0 + (1 \cdot 1 \cdot 1) = 0 + 1 = 1$
- $O_{A<B} = I_{A<B} + (I_{A=B} \cdot \overline{A} \cdot B) = 0 + (1 \cdot 0 \cdot 0) = 0 + 0 = 0$



Comparador de N bits com 3 saídas

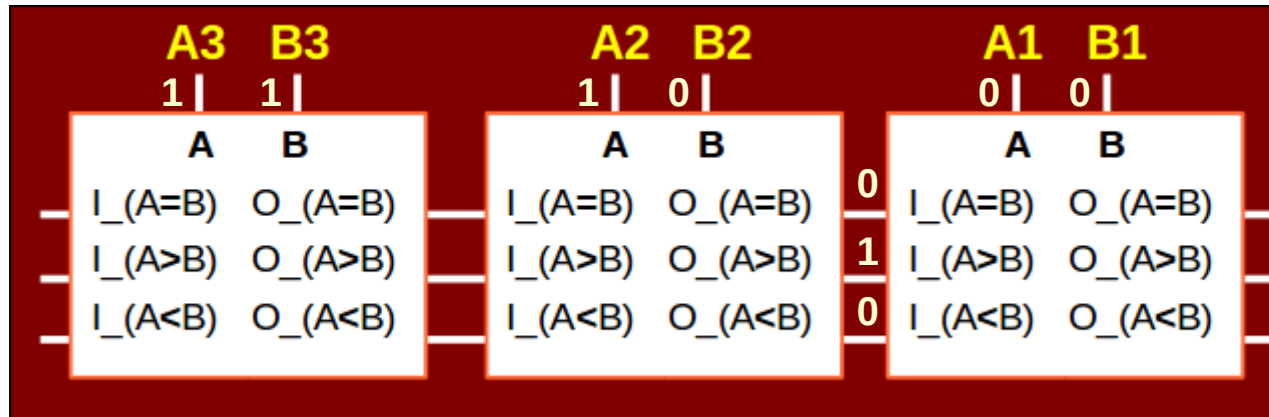
- Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?



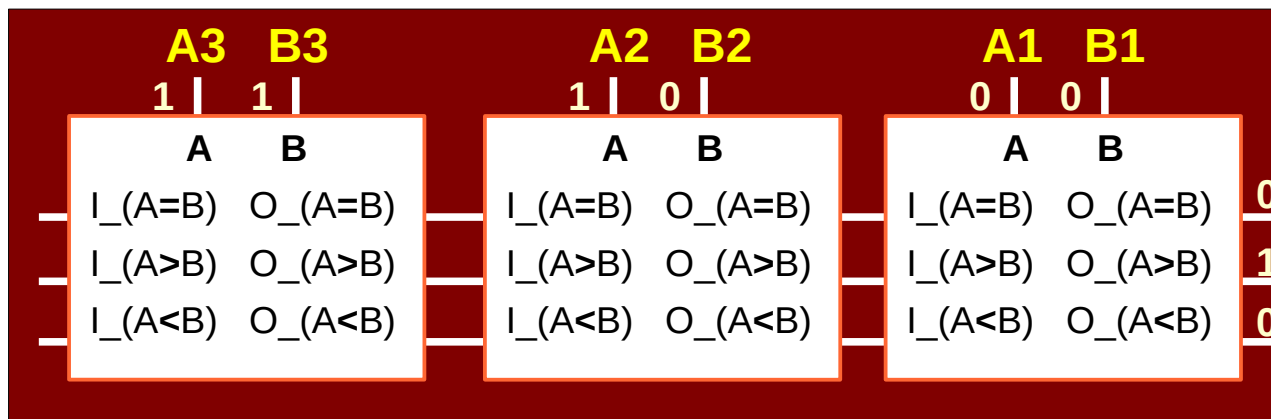
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

- ◇ Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?



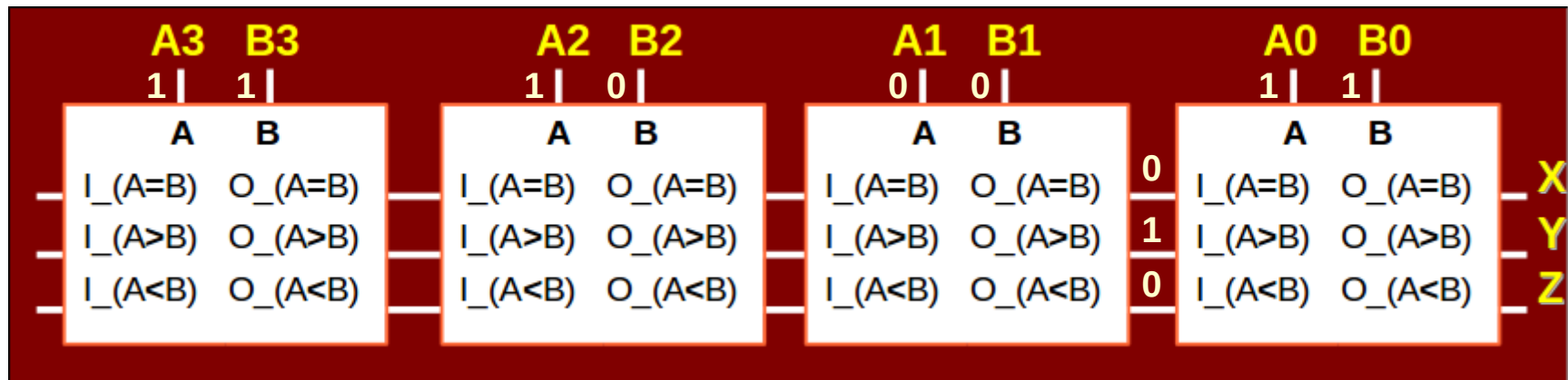
- $O_{A=B} = I_{A=B} \cdot \overline{A \oplus B} = 0 \cdot \overline{0 \oplus 0} = 0 \cdot \overline{0} = 0 \cdot 1 = 0$
- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \overline{B}) = 1 + (0 \cdot 0 \cdot 1) = 1 + 0 = 1$
- $O_{A<B} = I_{A<B} + (I_{A=B} \cdot \overline{A} \cdot B) = 0 + (0 \cdot 1 \cdot 0) = 0 + 0 = 0$



RIO
111

Comparador de N bits com 3 saídas

- ◇ Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?



1 1

1 0

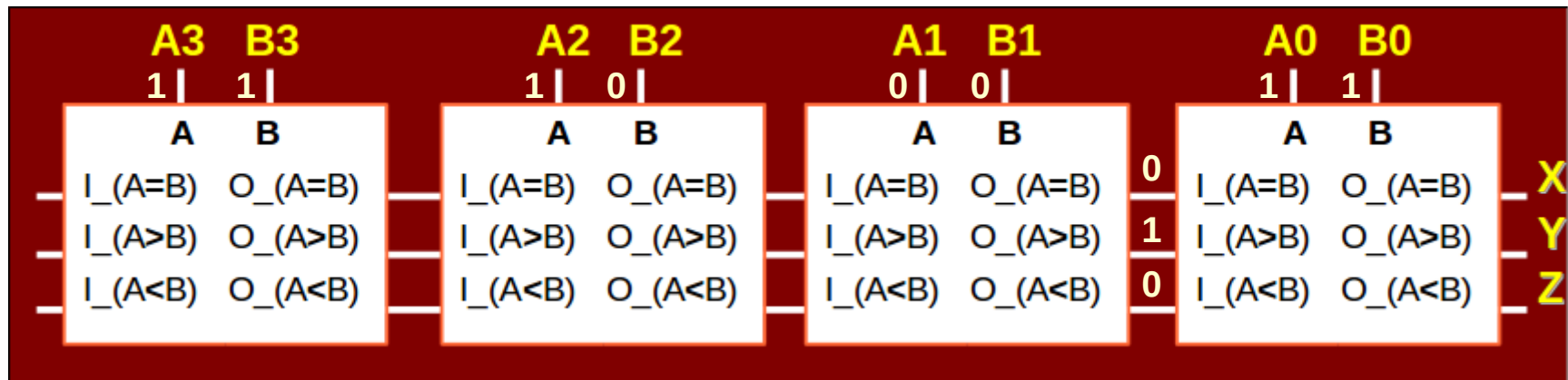
0 0

1 1

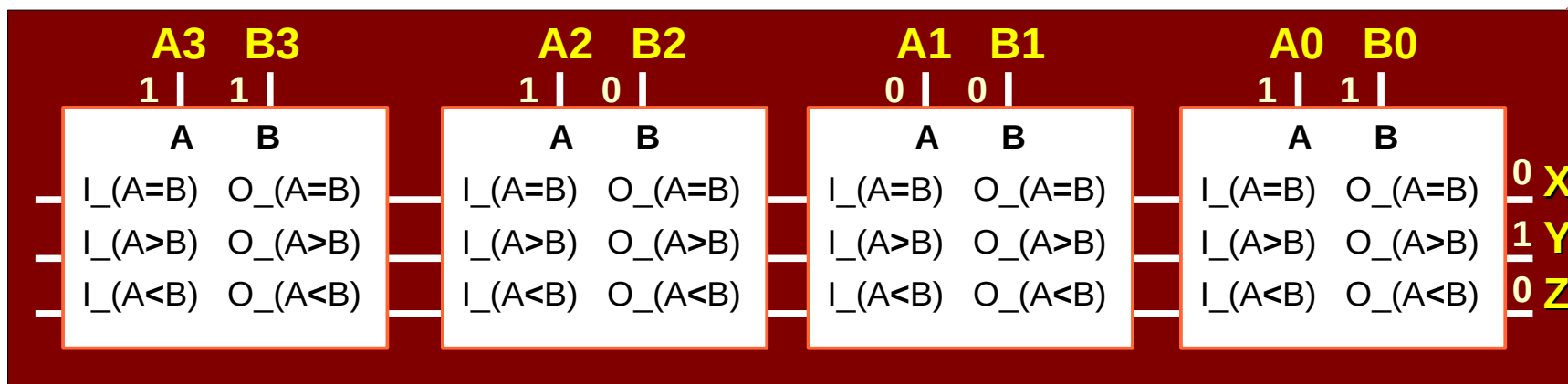
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Comparador de N bits com 3 saídas

- ◇ Qual o passo a passo da comparação dos seguintes números binários: **A = 1101** e **B = 1001**?



- $O_{A=B} = I_{A=B} \cdot \overline{A \oplus B} = 0 \cdot \overline{0 \oplus 0} = 0 \cdot \overline{0} = 0 \cdot 1 = 0$
- $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \overline{B}) = 1 + (0 \cdot 0 \cdot 1) = 1 + 0 = 1$
- $O_{A<B} = I_{A<B} + (I_{A=B} \cdot \overline{A} \cdot B) = 0 + (0 \cdot 1 \cdot 0) = 0 + 0 = 0$



Comparador: Exemplo de uso

- ◇ **Controle automático de ar condicionado:**



Comparador: Exemplo de uso

◇ Controle automático de ar condicionado:

- No controle do aquecedor são consideradas 2 variáveis:
 - Temperatura do ambiente
 - Temperatura definida no ar condicionado
- Usar comparador para saber se foi atingida a temperatura desejada
- É necessária a conversão de analógico para digital
- Entradas do comparador:
 - A: temperatura do ambiente
 - B: temperatura desejada



Comparador: Exemplo de uso

◇ Controle automático de ar condicionado:

- No controle do aquecedor são consideradas 2 variáveis:
 - Temperatura do ambiente
 - Temperatura definida no ar condicionado
- Usar comparador para saber se foi atingida a temperatura desejada
- É necessária a conversão de analógico para digital
- Entradas do comparador:
 - A: temperatura do ambiente
 - B: temperatura desejada
- Análise da saída:
 - $A = B \rightarrow$ temperatura desejada atingida
 - $A > B \rightarrow$ necessário resfriamento
 - $A < B \rightarrow$ necessário aquecimento

BCD

(Binary Converter to Decimal)



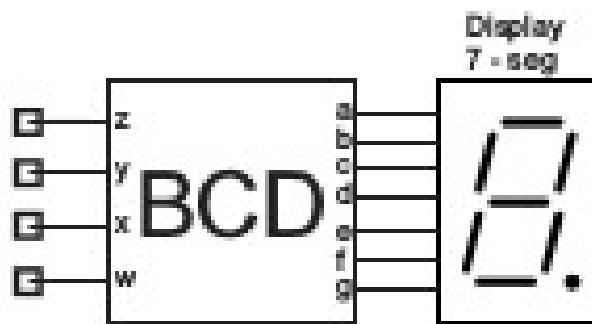
**MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]**

BCD

(Binary Converter to Decimal)

◆ Conceito:

- BCD (Conversor Binário para decimal)
- Utilizado para converter números binários em decimais, utilizando um *display* de 7 segmentos para apresentar o resultado,
- E a lógica combinacional é modelada com base na representação dos bits no display.

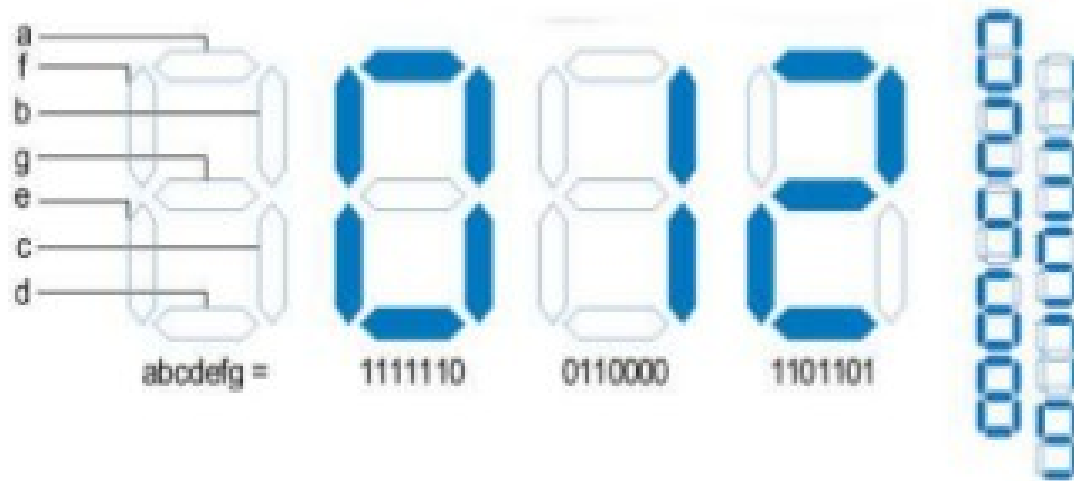
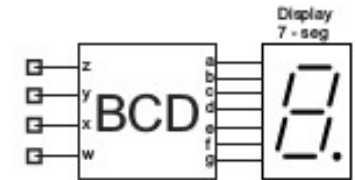


MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

BCD

(*Binary Converter to Decimal*)

◆ Projeto:



MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

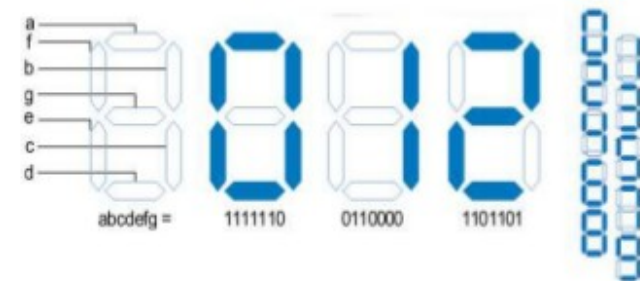
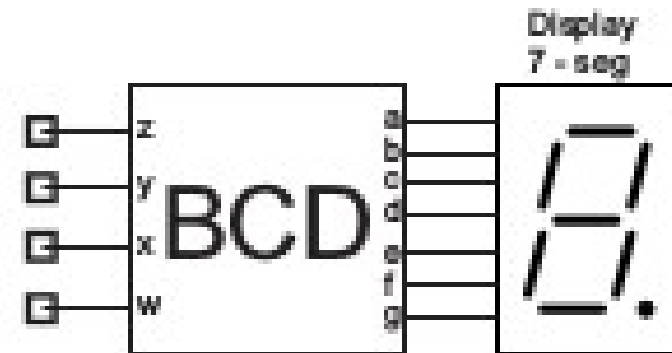
BCD

(Binary Converter to Decimal)

◆ Projeto:

■ Tabela verdade:

w	x	y	z	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0



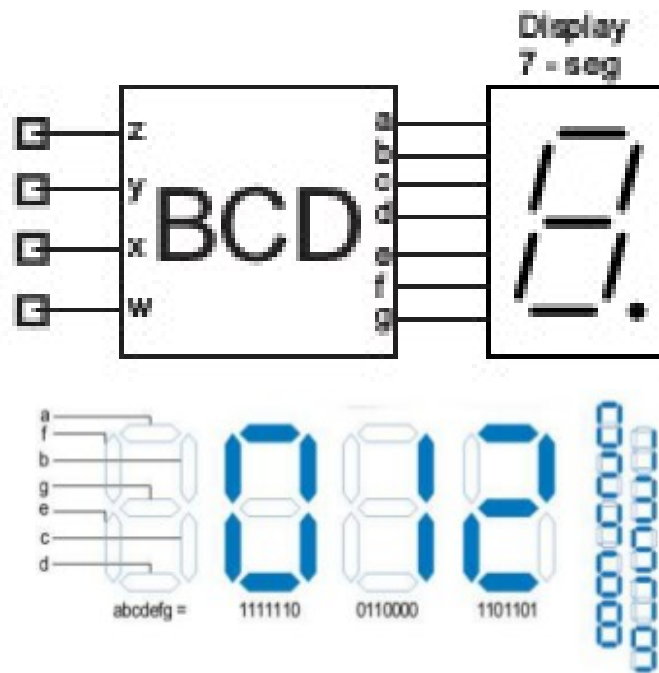
MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

BCD

(Binary Converter to Decimal)

Projeto:

Equações Booleanas:



w	x	y	z	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

$$a = \overline{w}xyz + \overline{w}xy\overline{z} + \overline{w}x\overline{y}z + \overline{w}x\overline{y}\overline{z} + \overline{w}xy\overline{z} + \overline{w}x\overline{y}z + wxyz + wxy\overline{z}$$

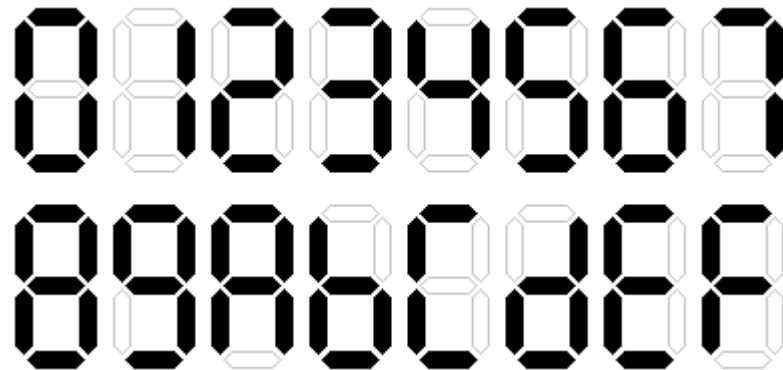
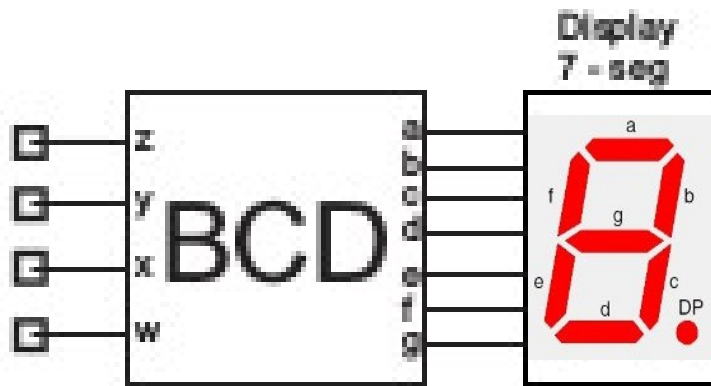
$$b = \overline{w}xyz + \overline{w}xy\overline{z} + \overline{w}x\overline{y}z + \overline{w}x\overline{y}\overline{z} + \overline{w}xy\overline{z} + \overline{w}x\overline{y}z + wxyz + wxy\overline{z}$$

$$c = \dots$$

BCD

(Binary Converter to Decimal)

◇ E se fosse Binário para Hexadecimal?



w	x	y	z	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	1	1	1	0	1	1	1
1	0	1	1	0	0	1	1	1	1	1
1	1	0	0	1	0	0	1	1	1	0
1	1	0	1	0	1	1	1	1	0	1
1	1	1	0	1	0	0	1	1	1	1
1	1	1	1	1	0	0	0	1	1	1

Resumo

◇ Multiplicador

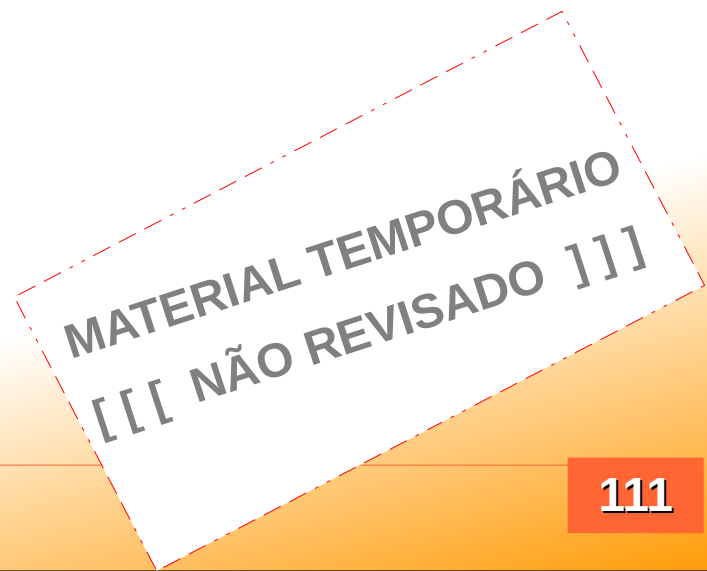
- Matricial
- Somas sucessivas
- Outras alternativas



Resumo

◇ Deslocador

- Circuitos onde a saída representa a entrada deslocada de uma certa quantidade de bits, e para um determinado lado (direita ou esquerda).
- Em um operador no qual são inseridos 0s, é possível realizar operações aritméticas com os números binários:
 - n deslocamentos à esquerda de 1 bit, equivale multiplicar n^o por 2^n
 - n deslocamentos à direita de 1 bit, equivale dividir n^o por 2^n
- Tipos de Deslocadores:
 - Deslocador de bit à esquerda;
 - Deslocador de bit à direita;
 - Deslocador com *shift*;
 - Deslocador com escolha de lado;
 - Deslocador Barrel.



Resumo

◇ Comparador

- Circuito que compara dois números binários (A e B), possuindo três possíveis saídas: $A=B$, $A>B$ e $A<B$.
- Comparador de n bits:
 - n comparadores de 1 bit.
- Comparador de n bits com 3 saídas:
 - lógica que interliga os níveis das palavras binárias, comparando bit a bit, propagando os resultados aos bits menos significativos das palavras binárias.
- Equações de um comparador de n bits com 3 saídas:
 - $O_{A>B} = I_{A>B} + (I_{A=B} \cdot A \cdot \bar{B})$
 - $O_{A<B} = I_{A<B} + (I_{A=B} \cdot \bar{A} \cdot B)$
 - $O_{A=B} = I_{A=B} \cdot \overline{A \oplus B}$

◇ BCD

- Binário (computador) → Decimal (humano)

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicação Chinesa

■ ???

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Multiplicação Binária

(números sem sinal)

◇ Multiplicação Chinesa

■ Vídeo

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Próxima Aula

◇ Implementação

- Multiplicador
- Comparador

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]

Bibliografia

- ◇ TOCCI, Ronald J; WIDMER, Neal S; MOSS, Gregory L. **Sistemas digitais: princípios e aplicações**. 11. ed. São Paulo: Pearson, 2011. 817 p. ISBN: 9788576050957.
- ◇ PEDRONI, Volnei A. **Eletrônica digital moderna e VHDL**. Rio de Janeiro: Elsevier, c2010. 619 p. ISBN: 9788535234657.
- ◇ WAGNER, Flávio R.; REIS, André I.; RIBAS, Renato P. **Fundamentos de circuitos digitais**. Porto Alegre: Bookman, 2008. 166 p. (Série Livros Didáticos, n. 17) ISBN: 9788577803453.
- ◇ VAHID, Frank. **Sistemas digitais: projeto, otimização e HDLS**. Rio Grande do Sul: Artmed Bookman, 2008. 558 p. ISBN: 9788577801909.
- ◇ WAKERLY, John F. **Digital design: principles and practices**. 4. ed. Upper Saddle River, N.J.: Pearson/Prentice Hall, c2006. xxiv, 895 p. ISBN: 0131863894.

MATERIAL TEMPORÁRIO
[[[NÃO REVISADO]]]