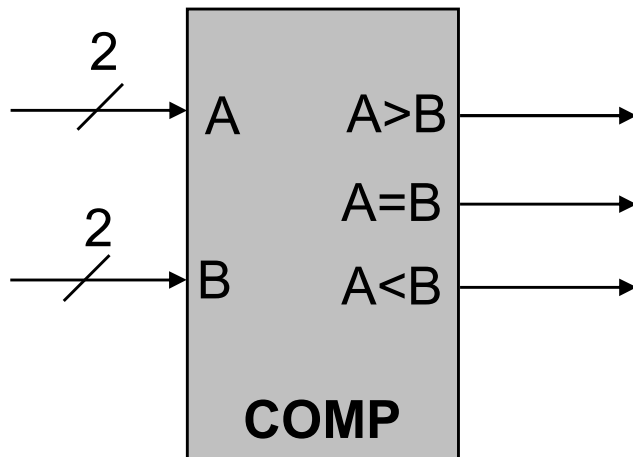


# Confrontatore

---

- Il blocco funzionale confrontatore ha:
  - ▶ due ingressi A e B da  $n \geq 1$  bit ciascuno
  - ▶ tre uscite:
    - $A < B$
    - $A = B$
    - $A > B$
- Il blocco confronta i due numeri binari A e B da n bit presenti sui due gruppi di ingressi, e
  - ▶ attiva (a 1) l'uscita corrispondente all'esito del confronto
  - ▶ azzerà le uscite corrispondenti alle condizioni false

# Confrontatore di numeri a 2 bit



Esempi:

- Se  $A = 01$  e  $B = 10$ 
  - ▶  $A > B = 0$
  - ▶  $A = B = 0$
  - ▶  $A < B = 1$
- Se  $A = 10$  e  $B = 00$ 
  - ▶  $A > B = 1$
  - ▶  $A = B = 0$
  - ▶  $A < B = 0$
- Se  $A = 10$  e  $B = 10$ 
  - ▶  $A > B = 0$
  - ▶  $A = B = 1$
  - ▶  $A < B = 0$

# Confrontatore di numeri a 2 bit

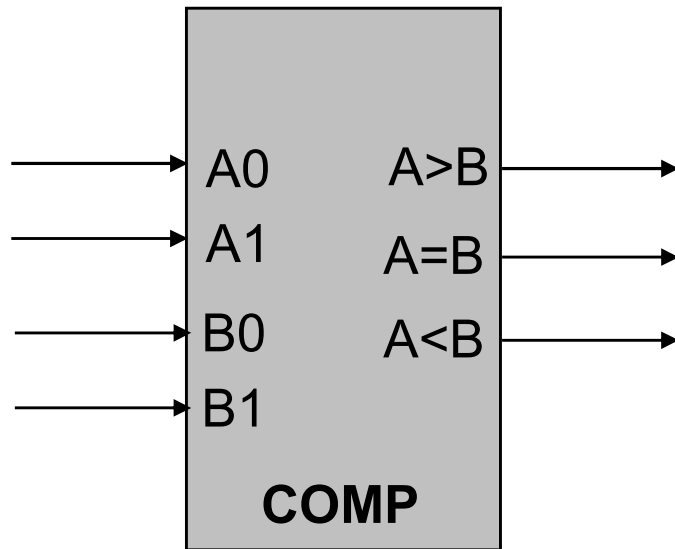


Tabella delle verità

A1	A0	B1	B0	A < B	A = B	A > B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

# Confrontatore a 2 bit: sintesi ( $A < B$ )

## Tabella delle verità

A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

**A<B**

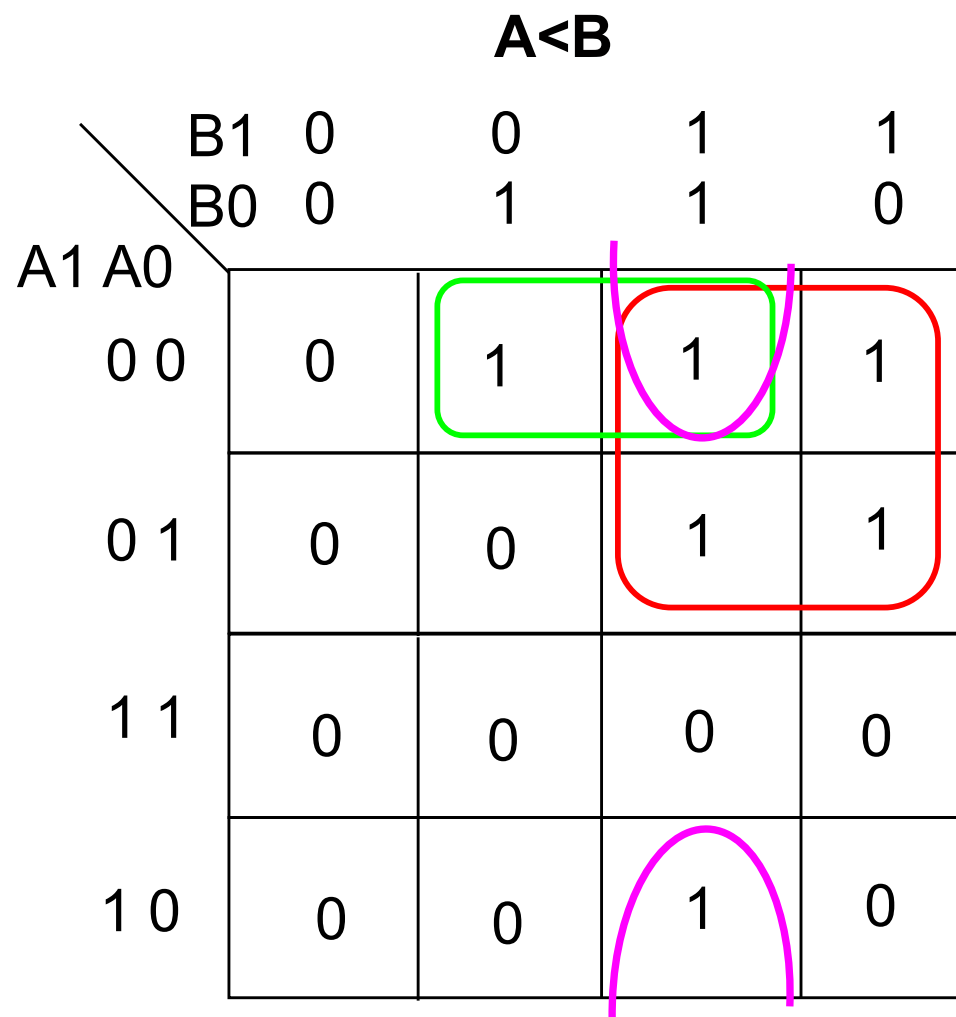
	B1	0	0	1	1
	B0	0	1	1	0
A1 A0					
0 0		0	1	1	1
0 1		0	0	1	1
1 1		0	0	0	0
1 0		0	0	1	0

**A<B = ?**

# Confrontatore a 2 bit: sintesi ( $A < B$ )

## Tabella delle verità

A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0



$$A < B = \neg A1 B1 + \neg A1 \neg A0 B0 + \neg A0 B1 B0$$

# Confrontatore a 2 bit: sintesi (A=B)

## Tabella delle verità

A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

		B1	0	0	1	1
		B0	0	1	1	0
A1	A0					
0	0	1	0	0	0	
0	1	0	1	0	0	
1	1	0	0	1	0	
1	0	0	0	0	1	

$$A=B = \neg A1 \neg A0 \neg B1 \neg B0 + \neg A1 A0 \neg B1 B0 + A1 A0 B1 B0 + A1 \neg A0 B1 \neg B0$$

# Confrontatore a 2 bit: sintesi ( $A > B$ )

## Tabella delle verità

A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

		B1	0	0	1	1
		B0	0	1	1	0
A1	A0					
0	0	0	0	0	0	
0	1	1	0	0	0	
1	1	1	1	0	1	
1	0	1	1	0	0	

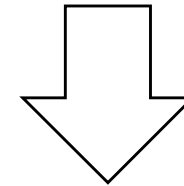
$$A > B = A0/B1/B0 + A1/B1 + A1A0/B0$$

# Confrontatore a 2 bit: sintesi (A=B)

## Tabella delle verità

A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	<b>1</b>	0
0	0	0	1	1	<b>0</b>	0
0	0	1	0	1	<b>0</b>	0
0	0	1	1	1	<b>0</b>	0
0	1	0	0	0	<b>0</b>	1
0	1	0	1	0	<b>1</b>	0
0	1	1	0	1	<b>0</b>	0
0	1	1	1	1	<b>0</b>	0
1	0	0	0	0	<b>0</b>	1
1	0	0	1	0	<b>0</b>	1
1	0	1	0	0	<b>1</b>	0
1	0	1	1	1	<b>0</b>	0
1	1	0	0	0	<b>0</b>	1
1	1	0	1	0	<b>0</b>	1
1	1	1	0	0	<b>0</b>	1
1	1	1	1	0	<b>1</b>	0

$$A=B = \neg(A1 \oplus B1) \neg(A0 \oplus B0)$$



$$A>B = \neg(A<B) \neg(A=B)$$

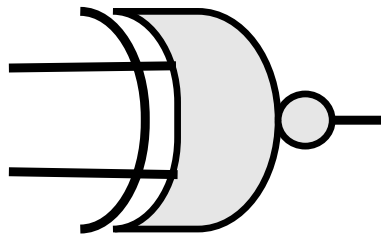


# Ripasso: operatore booleano NXOR

## NXOR

(«fa il contrario di XOR»)

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

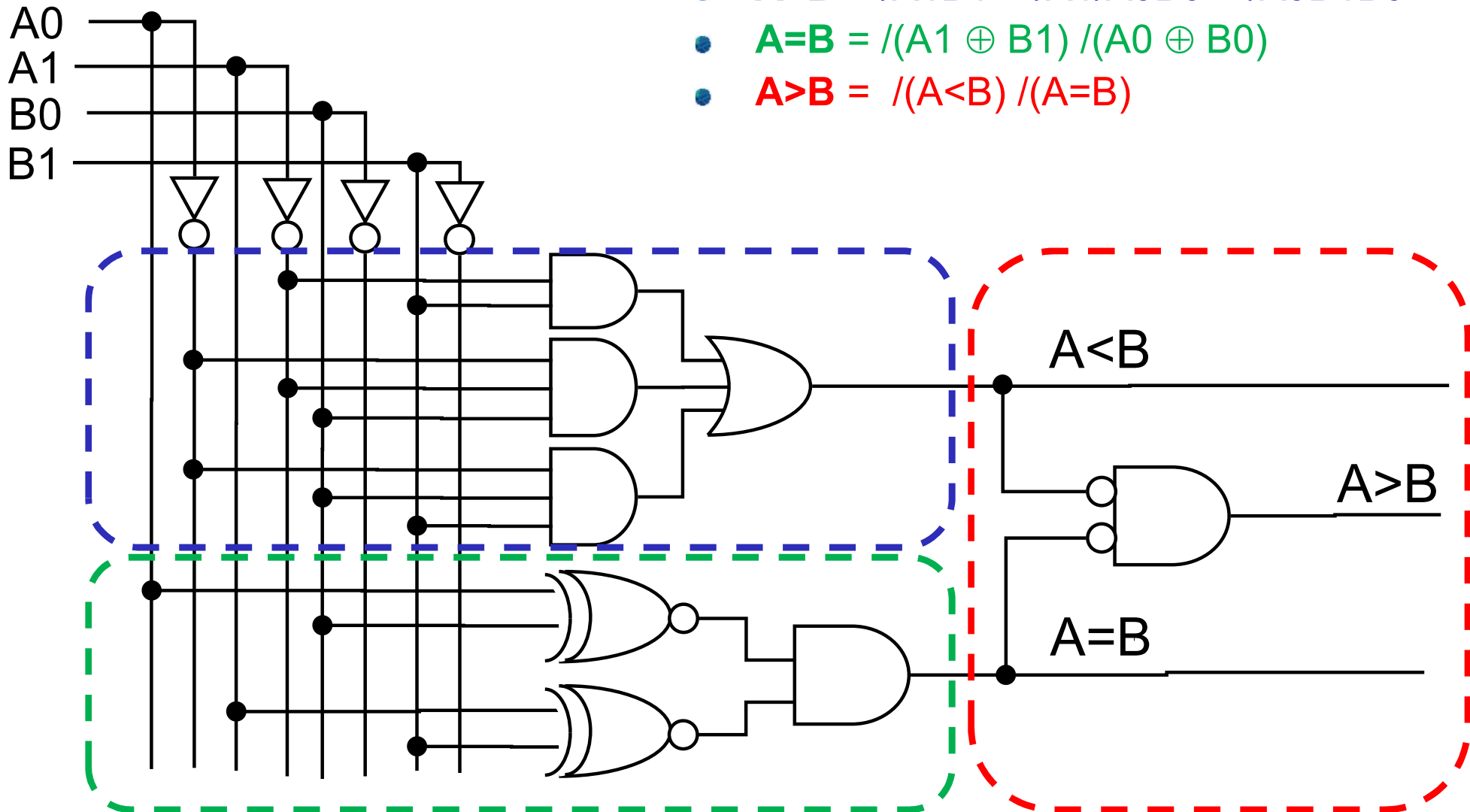


Significati intuitivi di A **NXOR** B:

- uno **XOR** seguito da un NOT  
 $A \text{ NXOR } B = \neg (A \text{ XOR } B)$
- cioè...  
 entrambi veri oppure  
 entrambi falsi
- cioè...  
 $AB + \neg A \neg B$
- cioè...  
**operatore di uguaglianza fra A e B**  
 vero se A e B sono uguali.  
 falso se sono diversi

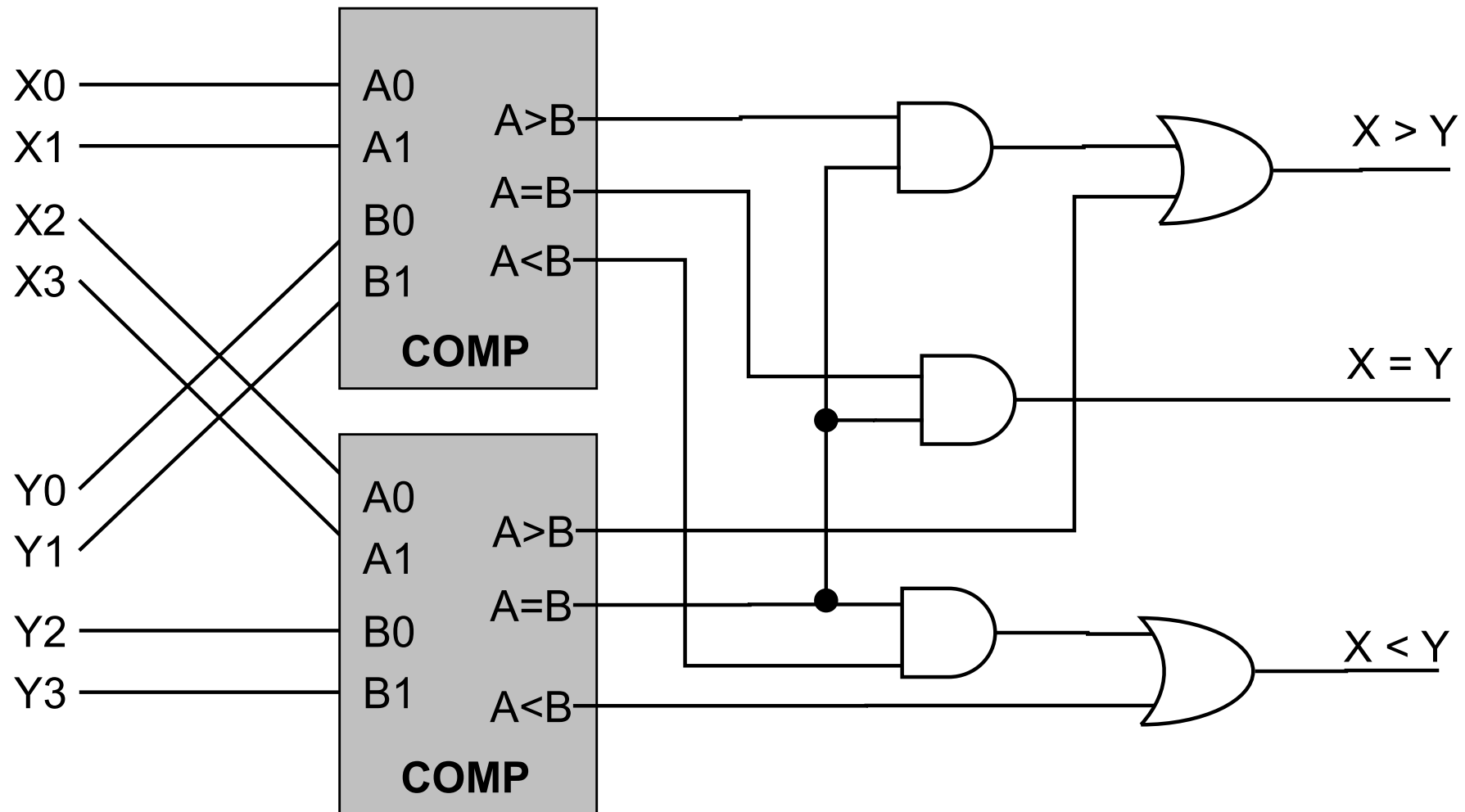
# Confrontatore a 2 bit: possibile implementazione

- $A < B = \neg A_1 B_1 + \neg A_1 \neg A_0 B_0 + \neg A_0 B_1 B_0$
- $A = B = \neg(A_1 \oplus B_1) \neg(A_0 \oplus B_0)$
- $A > B = \neg(A < B) \neg(A = B)$



# Confrontatore a 4 bit realizzato con confrontatori a 2 bit

- Si confrontano separatamente le parti più significative, e le parti meno significative, di X e Y.



# Compito a casa

---

- Si noti che circuiti come Decoder e Multiplexer non fanno alcuna ipotesi sul significato dei segnali.
- Il confrontatore invece ipotizza che i dati di ingresso siano dei numeri naturali codificati in binario
  - ▶ (e non in CP2 o in virgola mobile...).
- Realizzare un confrontatore tra numeri di tre bit in complemento a 2.
- Es. se  $X = 011$  e  $Y = 111$ , il confrontatore deve mettere a uno l'uscita  $X > Y$  e a zero le altre. Infatti in complemento a 2  $011=3$ , mentre  $111=-1$ .

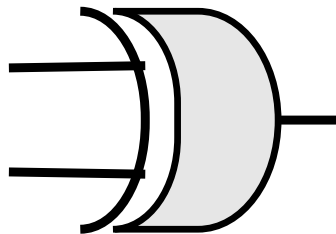
- Nello stesso modo, posso costruire un confrontore per  $2n$  bit con due confrontatori da  $n$  bits
  - ▶ Due confrontatori a 2 bit == 1 confrontatore a 4 bit
  - ▶ Due confrontatori a 4 bit == 1 confrontatore da 1 byte
  - ▶ Due confrontatori a 1 byte == 1 conf. da 2 byte (per short int!)
  - ▶ Due confrontatori a 2 byte == 1 conf. da 4 byte (per int!)

# Ripasso: operatore XOR

**XOR**  
(«or esclusivo»)

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

uno o l'altro,  
ma  
**NON entrambi**



Significati intuitivi di **A XOR B**:

- A oppure B, ma non entrambi  
(in latino: A **aut** B)
- vero se A e B diversi  
falso se A e B uguali
- vale  $\neg A$  se  $B = 1$   
vale  $A$  se  $B = 0$ 
  - ▶ (e viceversa)
- il contrario di A, se B vale;  
A immutato, altrimenti
  - ▶ (e viceversa)
- ...
- **somma naturale di A e B  
come numeri di... 1 bit!  
(ignorando il bit di riporto)**

# Semisommatore

---

- Dati due numeri naturali rappresentati su un solo bit, il circuito ne calcola la somma (compreso il riporto).

## Tavola delle verità

# Semisommatore

- Dati due numeri naturali rappresentati su un solo bit, il circuito ne calcola la somma (compreso il riporto).

## Tavola delle verità

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



# Semisommatore

- Dati due numeri naturali rappresentati su un solo bit, il circuito ne calcola la somma (compreso il riporto).

## Tavola delle verità

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

