

*Laboratorio di Architettura degli Elaboratori I*  
*Corso di Laurea in Informatica, A.A. 2019-2020*  
*Università degli Studi di Milano*



# Introduzione a Logisim

**Nicola Basilico**

Dipartimento di Informatica

Via Celoria 18- 20133 Milano (MI)

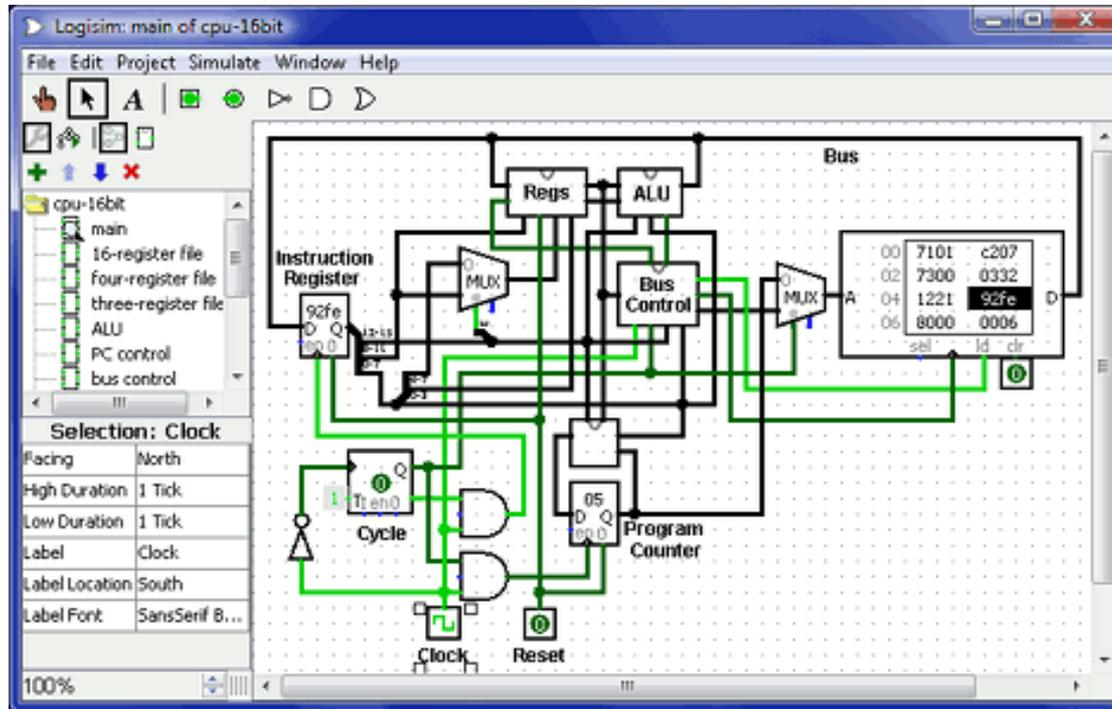
Ufficio 4008

[nicola.basilico@unimi.it](mailto:nicola.basilico@unimi.it)

+39 02.503.16289

# Logisim

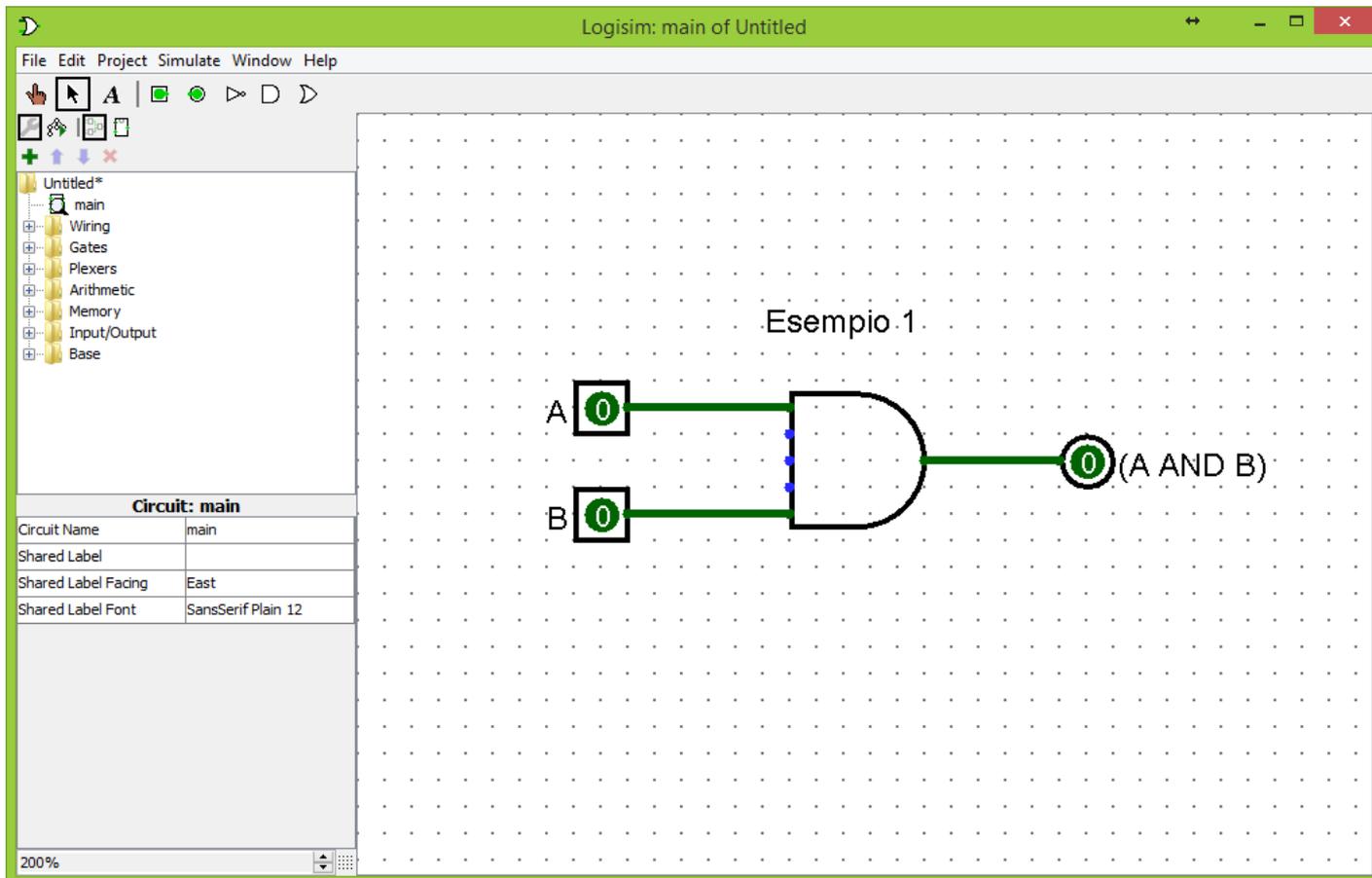
<http://www.cburch.com/logisim/>



- Strumento software che permette di progettare e simulare circuiti logici digitali

# Esempio

- Realizziamo un semplice circuito che, dati due segnali in ingresso  $A$  e  $B$ , calcoli  $(A \text{ AND } B)$



# Esempio

The screenshot shows the Logisim software interface. The title bar reads "Logisim: main of Untitled". The menu bar includes "File", "Edit", "Project", "Simulate", "Window", and "Help".

Annotations with red arrows point to various parts of the interface:

- Componenti di uso frequente:** Points to the toolbar containing icons for selection, deletion, and simulation.
- Libreria componenti:** Points to the component library on the left, which lists categories like Wiring, Gates, Plexers, Arithmetic, Memory, Input/Output, and Base.
- Proprietà componente selezionato:** Points to a table showing the properties of the selected component.
- Zoom area di lavoro:** Points to the zoom control at the bottom left, which is set to 200%.

The main workspace contains a circuit diagram titled "Esempio 1". It features two input components labeled "A" and "B", each with a "0" inside a square box. These inputs are connected to an AND gate. The output of the AND gate is connected to an output component labeled "0 (A AND B)".

Circuit: main	
Circuit Name	main
Shared Label	
Shared Label Facing	East
Shared Label Font	SansSerif Plain 12

# Operatori logici e proprietà

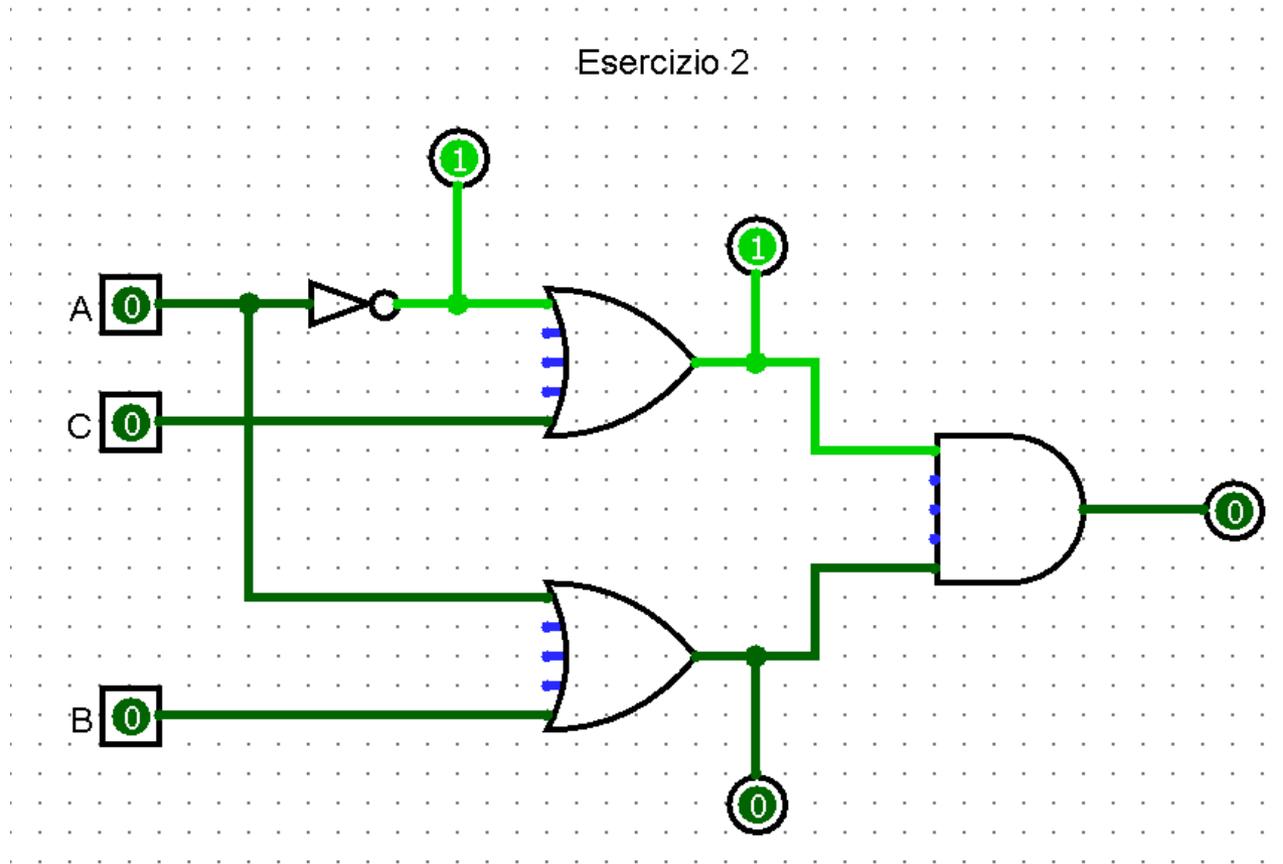
NOT	$\neg$	↓	Ordine di precedenza in assenza di parentesi
AND	$\wedge$		
OR	$\vee$		

Richiamo delle proprietà fondamentali

	AND	OR
Identità	$1 \wedge X = X$	$0 \vee X = X$
Elemento nullo	$0 \wedge X = 0$	$1 \vee X = 1$
Idempotenza	$X \wedge X = X$	$X \vee X = X$
Inverso	$X \wedge \neg X = 0$	$X \vee \neg X = 1$
Commutativa	$X \wedge Y = Y \wedge X$	$X \vee Y = Y \vee X$
Associativa	$(X \wedge Y) \wedge Z = X \wedge (Y \wedge Z)$	$(X \vee Y) \vee Z = X \vee (Y \vee Z)$
Distributiva	(di AND risp. ad OR) $X \wedge (Y \vee Z) = X \wedge Y \vee X \wedge Z$	(di OR risp. ad AND) $X \vee (Y \wedge Z) = (X \vee Y) \wedge (X \vee Z)$
Assorbimento I	$X \wedge (X \vee Y) = X$	$X \vee (X \wedge Y) = X$
Assorbimento II	$X \wedge (\neg X \vee Y) = X \wedge Y$	$X \vee (\neg X \wedge Y) = X \vee Y$
De Morgan	$\neg(X \wedge Y) = \neg X \vee \neg Y$	$\neg(X \vee Y) = \neg X \wedge \neg Y$

# Esercizio 2

1. Si riproduca in Logisim il seguente circuito:

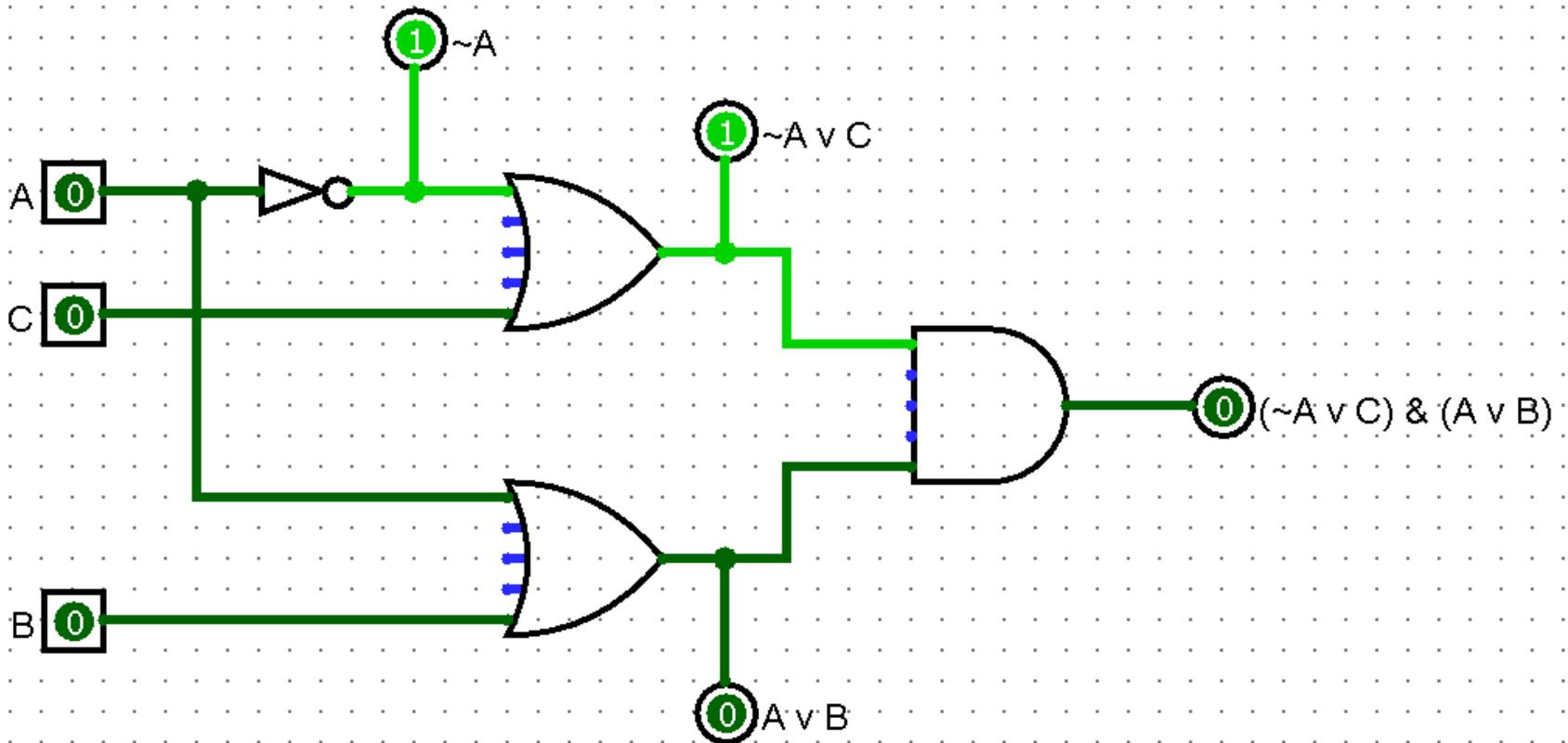


2. Si determini l'espressione logica di tutte le uscite (intermedie e finale)
3. Si scriva la tabella di verità del circuito

# Esercizio 2

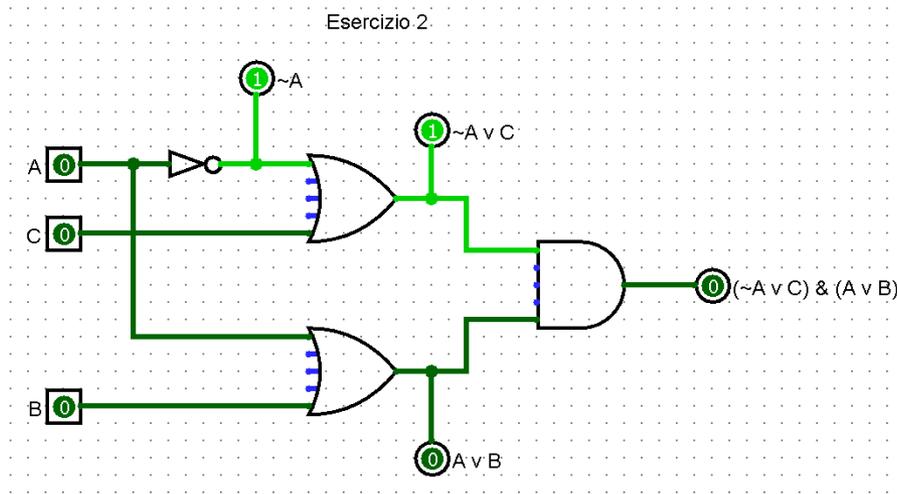
Label sui segnali (intermedi e finale)

Esercizio-2.



# Esercizio 2

Tabella di verità



$A$	$B$	$C$	$(\neg A \vee C) \wedge (A \vee B)$
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	0
1	1	1	1

# Esercizio 3

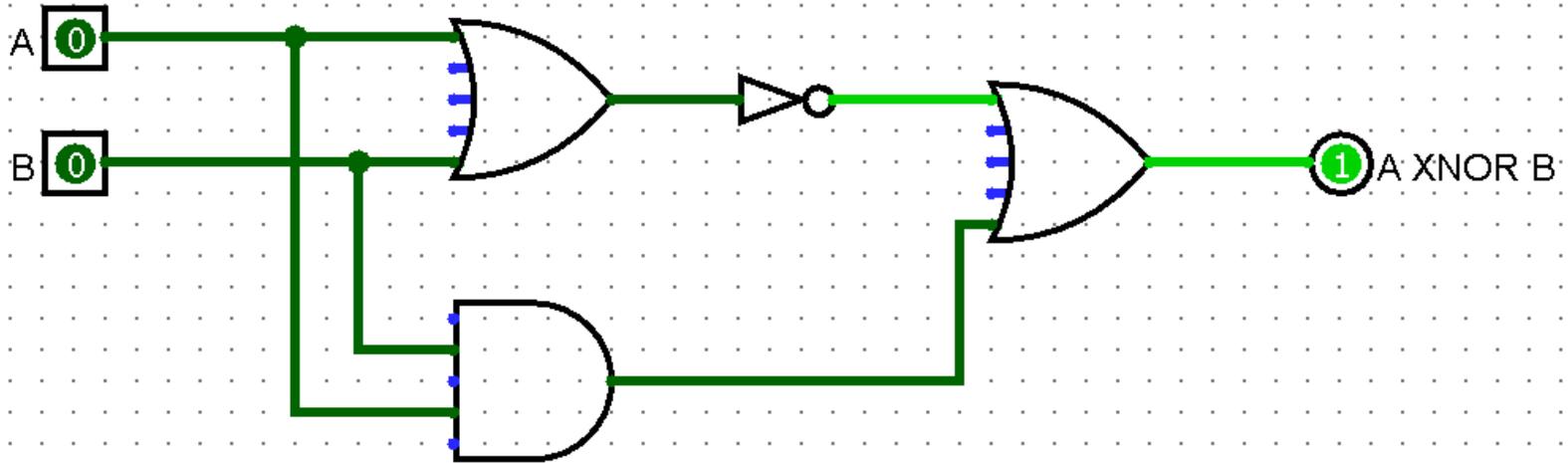
1. Dati due segnali  $A$  e  $B$ , si implementi un circuito che calcoli  $A \text{ XNOR } B$  senza usare porte composte ( $\text{NAND}$ ,  $\text{NOR}$ ,  $\text{XOR}$ ,  $\text{XNOR}$ )
2. Si derivi la tabella di verità e si osservi la funzione logica risultante

# Esercizio 3

1. Dati due segnali  $A$  e  $B$ , si implementi un circuito che calcoli  $A \text{ XNOR } B$  senza usare porte composte ( $\text{NAND}$ ,  $\text{NOR}$ ,  $\text{XOR}$ ,  $\text{XNOR}$ )
2. Si derivi la tabella di verità e si osservi la funzione logica risultante

Suggerimento:  $A \text{ XNOR } B = \neg(A \vee B) \vee (A \wedge B)$

# Esercizio 3

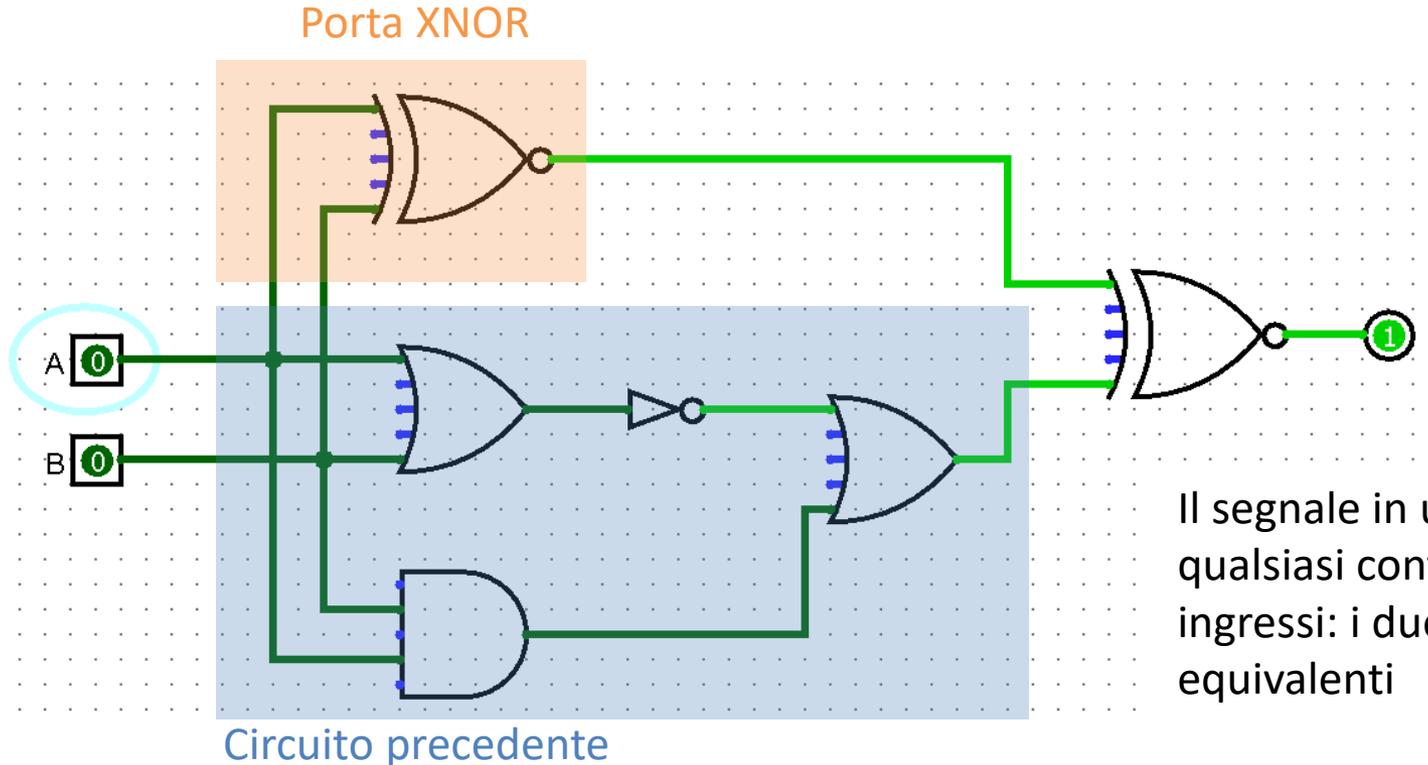


$A$	$B$	$\neg(A \vee B) \vee (A \wedge B)$
0	0	1
0	1	0
1	0	0
1	1	1

La funzione risultante è l'**uguaglianza logica**: possiamo usare **XNOR** per valutare l'uguaglianza del segnale in uscita a due diversi circuiti

# Esercizio 3

Confronto il circuito prodotto precedentemente con la singola porta XNOR utilizzando un'ulteriore porta XNOR:



Il segnale in uscita è 1 per qualsiasi configurazione degli ingressi: i due circuiti sono equivalenti

# Esercizio 4

Sia data la seguente espressione logica:

$$X = \neg A \vee \neg(B \vee \neg C)$$

1. Si derivi la tabella di verità (si indichino anche alcune sotto-espressioni)
2. Si realizzi il circuito corrispondente e si verifichi la correttezza della tabella

# Esercizio 5

Dimostrare tramite manipolazioni algebriche (specificando le proprietà usate) che:

$$E_1 = E_2$$

dove:

$$E_1 = \neg(\neg A \wedge B \wedge \neg C \vee A \wedge B \wedge \neg C) \wedge A$$

$$E_2 = (\neg B \wedge A) \vee (A \wedge C)$$

Si implementino i circuiti di  $E_1$  e  $E_2$  e si verifichi l'equivalenza tramite la porta **XNOR**

# Esercizio 6

Si consideri la seguente espressione:

$$E_1 = (A \text{ NOR } B) \wedge (C \vee \neg B)$$

1. Si implementi il circuito corrispondente usando la sola porta **NAND**
2. Si mostri, con passaggi algebrici e confronto tra circuiti, che è equivalente a

$$E_2 = \neg A \wedge \neg B$$

<i>A</i>	<i>B</i>	<i>A NAND B</i>
0	0	1
0	1	1
1	0	1
1	1	0

*Laboratorio di Architettura degli Elaboratori I*  
*Corso di Laurea in Informatica, A.A. 2019-2020*  
*Università degli Studi di Milano*



**Nicola Basilico**

Dipartimento di Informatica

Via Celoria 18- 20133 Milano (MI)

Ufficio 4008

[nicola.basilico@unimi.it](mailto:nicola.basilico@unimi.it)

+39 02.503.16289

*Hanno contribuito alla  
realizzazione di queste slides:*

*Dr. Iuri Frosio*

*Dr. Massimo Marchi*

*Dr. Alberto Quattrini Li*

*Dr. Matteo Re*