

# Esercitazione 01

---

## Introduzione a Logisim

Gianluca Brilli  
*gianluca.brilli@unimore.it*



# Ambiente di Lavoro (1)

---

- › **Logisim**: Simulatore di Circuiti Logici.
- › disponibile il pacchetto jar da <https://sourceforge.net/projects/circuit/> , ma anche il pacchetto deb nelle maggiori distribuzioni GNU/Linux derivate da Debian.
- › *\$ sudo apt-get install logisim*



# Ambiente di Lavoro (2)

Logisim: main of Untitled

File Edit Project Simulate Window Help

+

- ▼ Untitled\*
  - main
  - ▶ Wiring
  - ▶ Gates
  - ▶ Plexers
  - ▶ Arithmetic
  - ▶ Memory
  - ▶ Input/Output
  - ▶ Base

Pin	
Facing	North
Output?	No
Data Bits	1
Three-state?	No
Pull Behavior	Unchanged
Label	
Label Location	West
Label Font	SansSerif Plai...

100%



# Porte Elementari (1)

---

1. Cliccare sulla porta AND. Un'ombra di una porta AND ora segue il cursore. Cliccare sulla basetta del circuito principale per posarlo.
2. Cliccare il pulsante del piedino di ingresso ("Input Pin") e piazzarne due da qualche parte a sinistra della porta AND.
3. Cliccare il pulsante del piedino di uscita ("Output Pin") e posizionarne uno da qualche parte a destra della porta AND.



# Porte Elementari (2)

---

1. Cliccare ora il pulsante di selezione "Select tool" e connettere i due piedini di input con la porta AND. Per posizionare un filo, basta cliccare e trascinare dal punto sorgente a quello destinazione.
2. È possibile ridurre il numero di ingressi della porta cliccandola con lo strumento di selezione e modificandone le proprietà nel pannello situato in basso a sinistra della finestra. Tale impostazione può essere applicata anche prima che il componente sia stato posato.



# Porte Elementari (3)

---

1. Infine, cliccare lo strumento di manipolazione ("Poke") e smanettare con i piedini di ingresso osservando gli effetti. Controllare che il comportamento della porta corrisponda con le aspettative.



# Creazione di Sottocircuiti (1)

---

- › Creazione di reti logiche in maniera “modulare”.
- › Creare un nuovo schema: Menu -> File -> New.
- › Creare un nuovo sotto-circuito: Menu -> Project -> Add Circuit. Lo si chiami NAND.
- › creare un circuito NAND con due piedini di ingresso a sinistra e uno di uscita a destra. Per farlo, usare solo porte AND, OR o NOT

# Creazione di Sottocircuiti (2)

---

---



- > Si ritorni al circuito principale doppio-cliccando "main" nel prospetto sinistro dei circuiti. Si clicchi ora (una sola volta) l'elemento NAND posizionarlo nel circuito principale. Lo si collaudi aggiungendovi piedini di ingresso e di uscita.





# Creazione di Sottocircuiti (3) - Soluzione

The screenshot displays a circuit simulation software window with a menu bar (File, Edit, Project, Simulate, Window, Help) and a toolbar. The left sidebar shows a project tree for 'Untitled 2\*' with a 'Wiring' section containing various components like Splitter, Pin, Probe, Tunnel, Pull Resistor, Clock, Constant, Power, Ground, Transistor, and Transmission Gate. Below the sidebar is a table with the following data:

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

The main workspace shows a circuit diagram on a grid background. It consists of two input pins labeled 'A' and 'B' connected to a NAND gate. The output of the NAND gate is connected to an inverter, which is then connected to an output pin labeled 'OUT'. The circuit is drawn with blue lines on a white grid.



# Esercizio 01

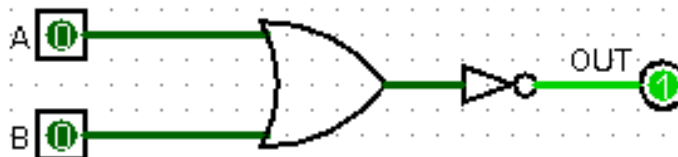
---

- > Costruiamo e testiamo in logisim un sottocircuito per:
- > NOR
- > XOR

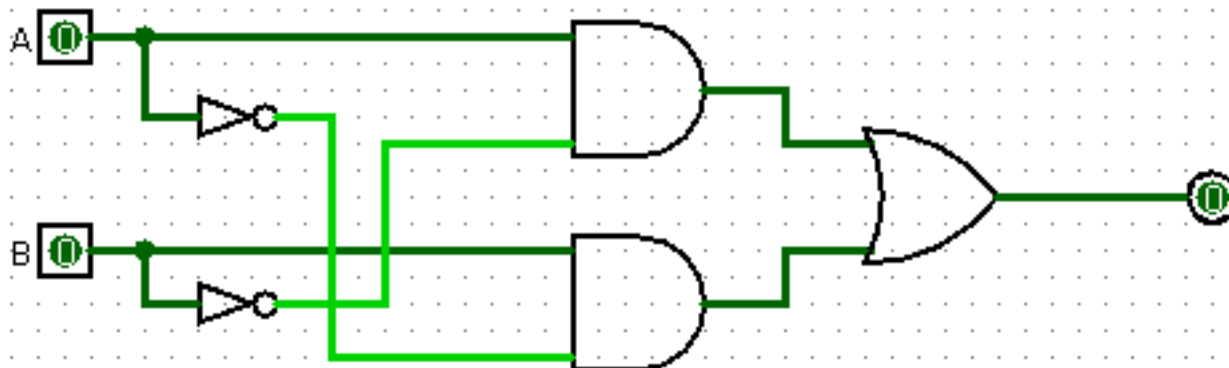


# Esercizio 01 - Soluzione

> NOR



> XOR





# Tablelle di Verità e Funzioni Logiche

- > Nel prospetto dei circuiti cliccare destro sul sotto-circuito NAND ed analizzare il circuito selezionando "Analyze circuit".

Combinational Analysis  
File Edit Project Simulate Window Help

Inputs Outputs Table Expression Minimized

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

Build Circuit

Combinational Analysis  
File Edit Project Simulate Window Help

Inputs Outputs Table Expression Minimized

Output: x  
(A + B) (A + B)

(A + B) (~A + ~B)

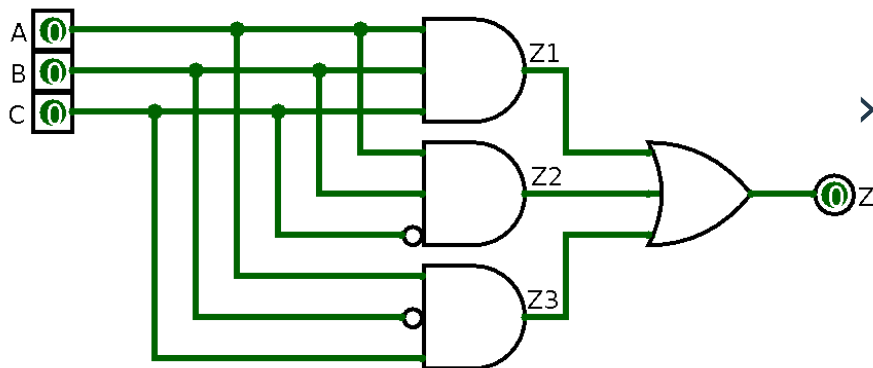
Clear Revert Enter

Build Circuit

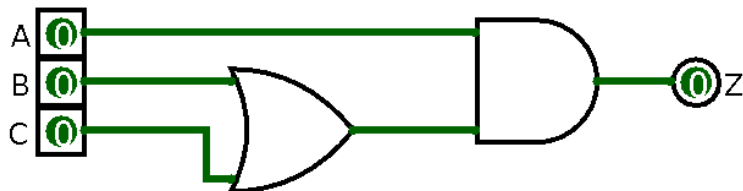


# Esercizio 02

> Testiamo in Logisim il seguente circuito:



> Versione non minima



> Versione minimizzata applicando alcuni teoremi.



# Esercizio 02 - Soluzione

- > Confrontiamo le tabelle di verità, per mezzo del tool di generazione automatica di Logisim:

A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

> Circuito partenza

di

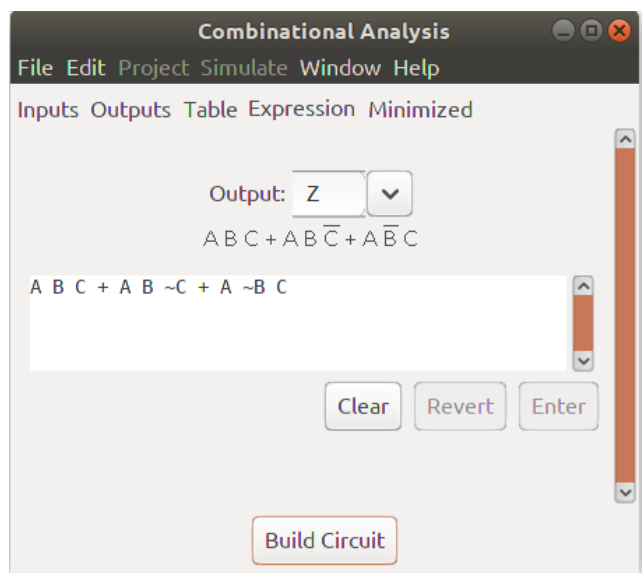
A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

> Versione minimizzata



# Esercizio 02 - Soluzione

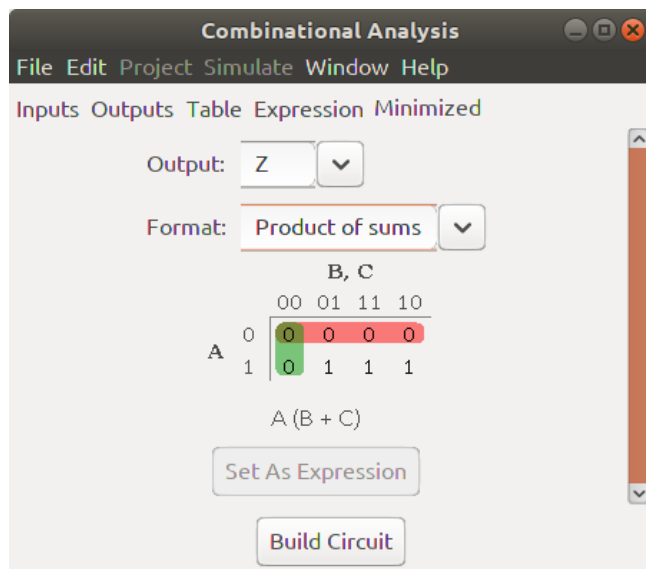
- > Relativamente al primo circuito ( la versione non minimizzata ), aprendo la voce Analyze Circuit notiamo la sua espressione logica:





# Esercizio 02 - Soluzione

- > Cliccando in basso su “Minimized” possiamo automaticamente ottenere la versione minimizzata che ci siamo calcolati tramite i teoremi dell’algebra di Boole.



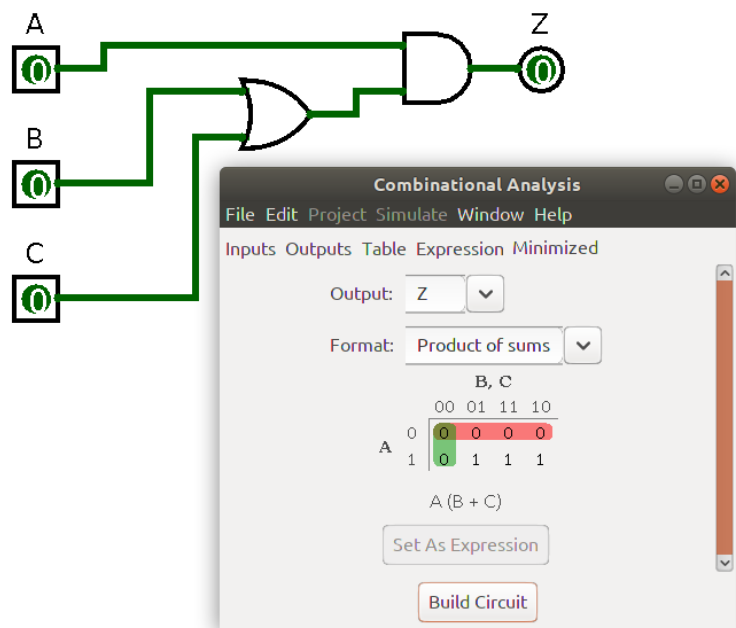
$$> Z = A ( B + C )$$





# Esercizio 02 - Soluzione

- > Cliccando infine su “Build Circuit” possiamo generare il circuito logico della versione minimizzata in automatico.

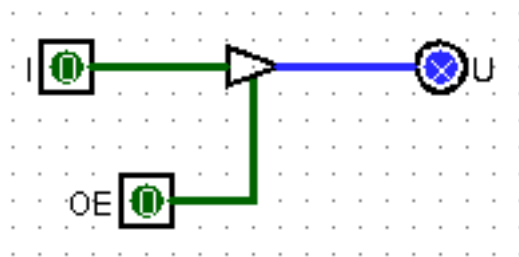




# Buffer Three State (1)

---

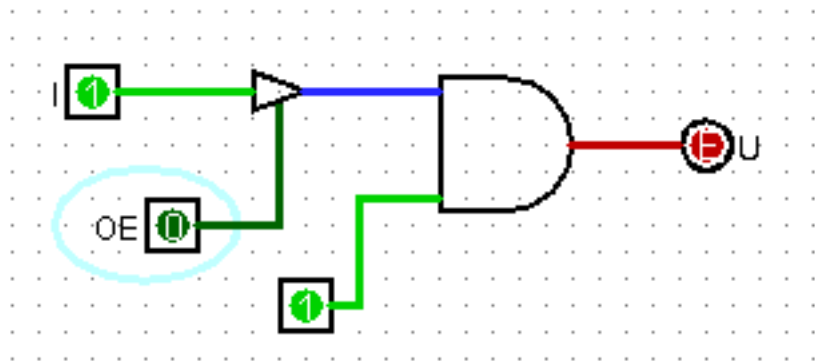
- > Elemento che introduce un terzo stato differente da 0 e 1, chiamato **Alta Impedenza** (indicato con Z).
- > Modella la condizione di **Circuito Aperto e Cortocircuito**.



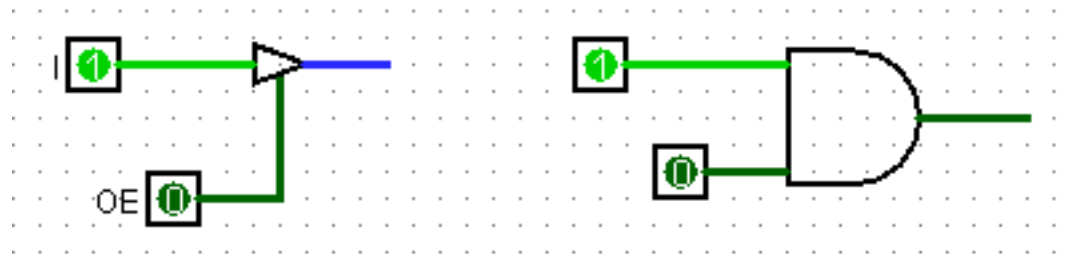


# Buffer Three State (2)

> Qual'è il valore dell'uscita?



> Cosa cambia tra queste due reti?





## Esercizio 2

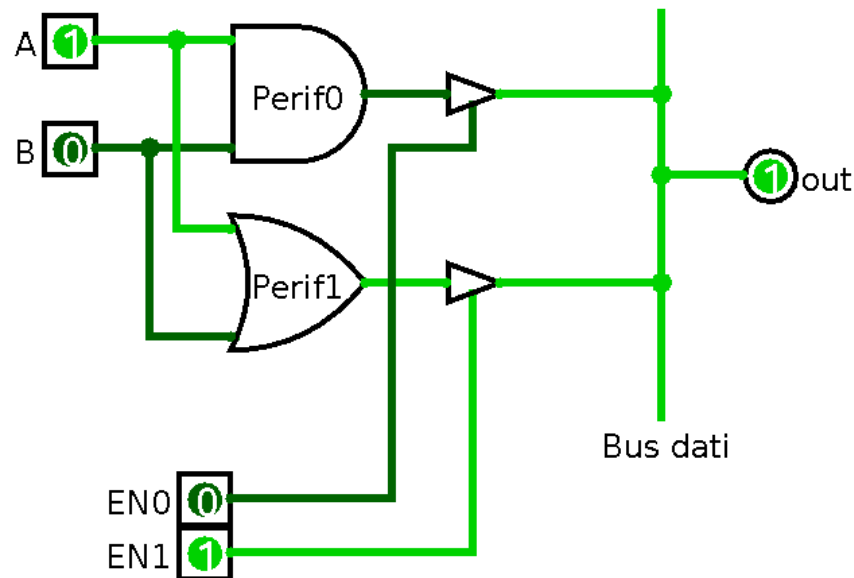
---

---

- › Realizziamo un semplice circuito basato su **buffer three-state** che vada a simulare il funzionamento di un **bus dati** con diverse **periferiche collegate** ad esso.
- › Per semplicità supponiamo che queste periferiche siano delle semplici porte logiche.



## Esercizio 2 - Soluzione

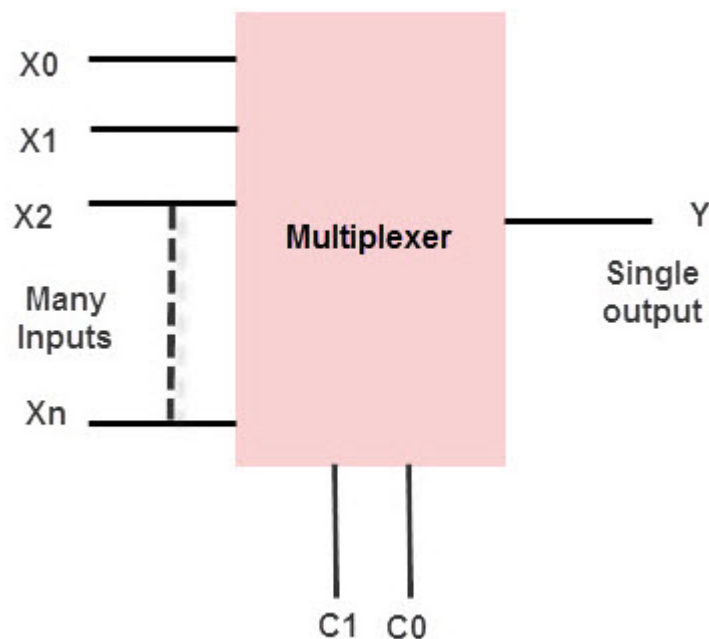


- > I due output enable abilitano un ingresso piuttosto che l'altro.
- > Ovviamente se più di uno sono abilitati si crea conflitto.



# Esercizio 03

- > Realizziamo un Multiplexer 2 a 1 in Logisim
- > Analizziamo Tabella d. V. e F. Logica del MUX

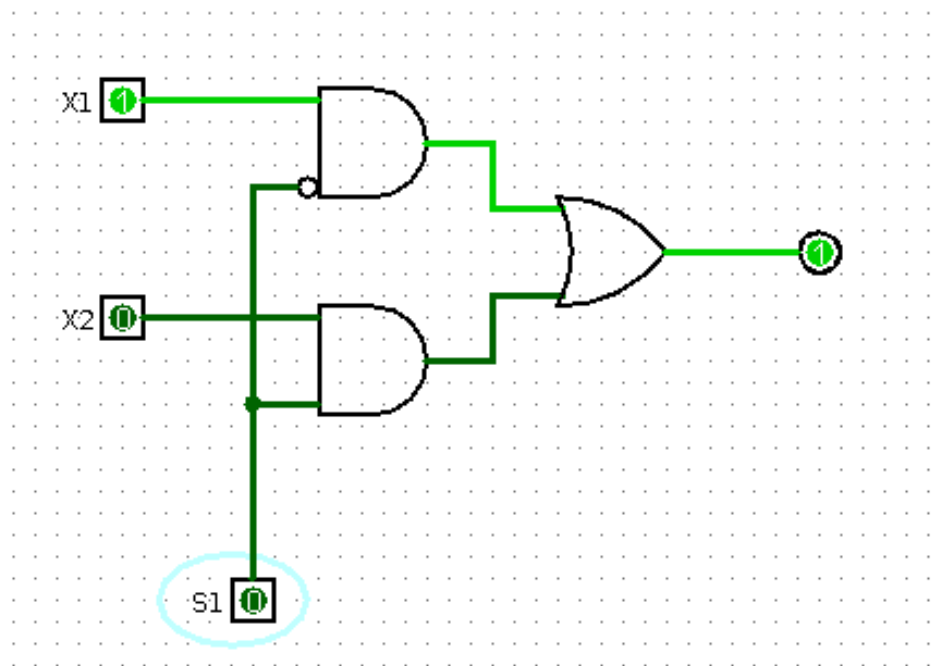


- > Se ci rimane tempo
- > Estendiamo il MUX a 4 linee di ingresso.



# Esercizio 02 - Soluzione A

- > linee di ingresso:  $n = 2$
- > linee di selezione:  $k = \log_2(n) = 1$





# Esercizio 02 - Soluzione B

- › MUX a 4 ingressi: necessario aggiungere due ingressi ed un selettore.

