

# Laboratorio di architettura degli elaboratori

## CIRCUITI COMBINATORI

### Lezione 3

#### STRUMENTI SOFTWARE

- Logisim (<https://sourceforge.net/projects/circuit>)

#### CONTATTI

- Prof. F. Fontana ([federico.fontana@uniud.it](mailto:federico.fontana@uniud.it))
- Y. De Pra ([yuri.depra@uniud.it](mailto:yuri.depra@uniud.it) )

## Correzione esercizio 2.1

- a) Progettare un circuito che, ricevuti 4 segnali binari (bit) in ingresso, stabilisca se questi rappresentano nella notazione binaria un numero primo (consideriamo 1 non primo). Il circuito restituisce in uscita 1 se l'input rappresenta un numero primo, mentre restituisce 0 in caso contrario.
- b) Progettare un circuito che riceva in ingresso un numero binario di 4 bit. Il circuito restituisce in uscita 1 se l'ingresso è una cifra decimale (ossia un valore tra 0 e 9) divisibile per 2 o per 5; restituisce 0 se l'ingresso è una cifra non divisibile né per 2 né per 5. Infine, nel caso in cui l'ingresso non rappresenti alcuna cifra decimale, l'uscita può assumere un valore arbitrario.

Esercizio 2.1a

Numeri primi: 2,3,5,7,11,13

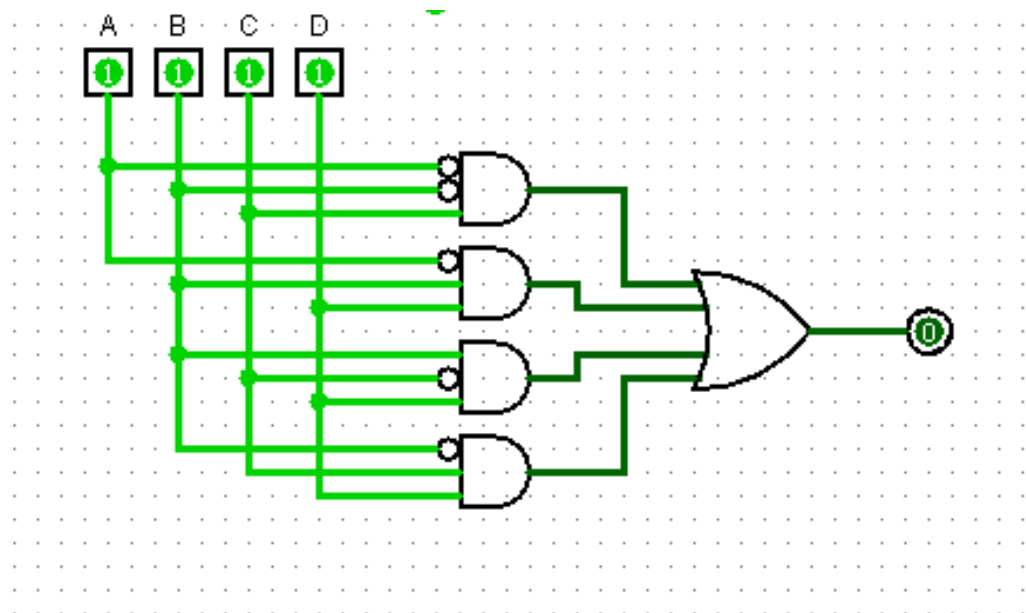
A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
...	...	...	...	...
1	1	1	1	0

NOTA: ogni  
quadrato deve  
avere almeno  
una casella non  
presente in  
altri quadrati

AB \ CD					
CD \ AB		00	01	11	10
	00	0	0	0	0
	01	0	1	1	0
	11	1	1	0	1
	10	1	0	0	0

$Y = A'B'C + A'BD + BC'D + B'CD$

Circuito corrispondente



## Esercizio 2.1b

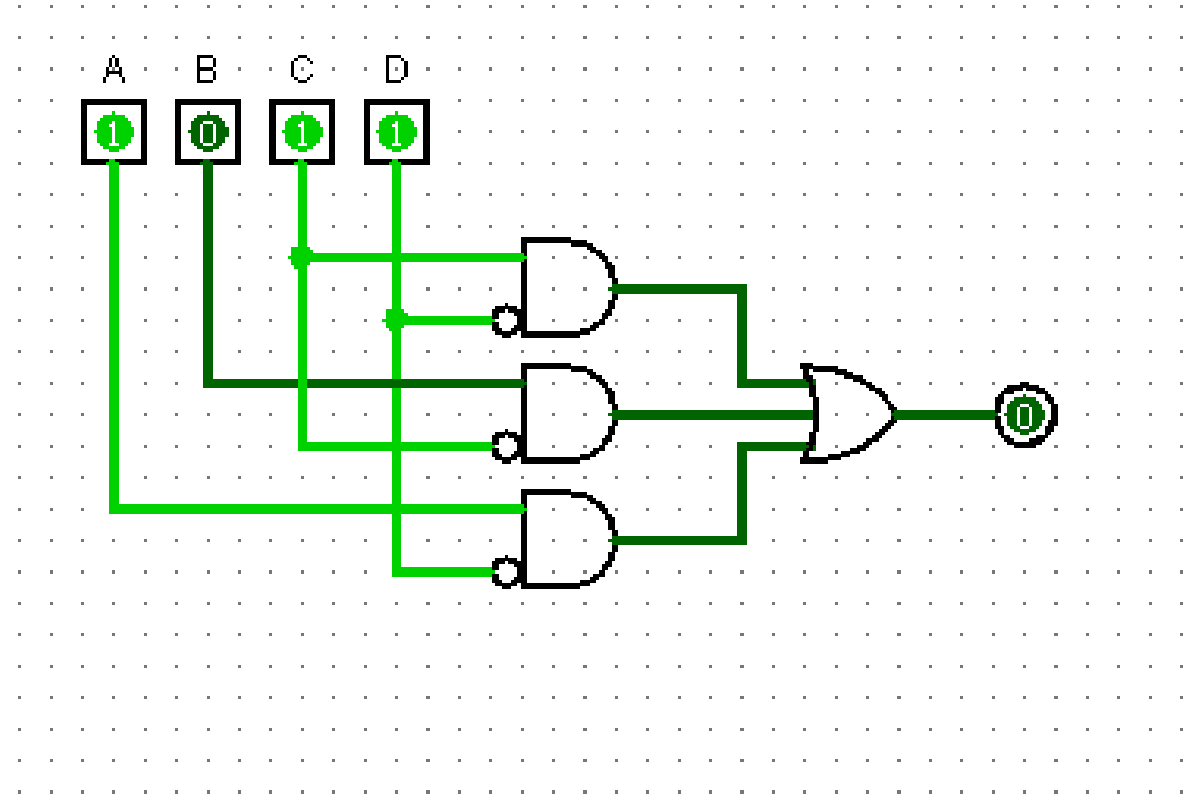
2, 4, 5, 6, 8  $\rightarrow$  1 (divisibili per 2 o 5)

0, 1, 3, 7, 9  $\rightarrow$  0

10, 11, 12, 13, 14, 15  $\rightarrow$  X (0 oppure 1)

AB \ CD		AB			
		00	01	11	10
CD	00	0	1	X	1
	01	0	1	X	0
	11	0	0	X	X
	10	1	1	X	X

$$Y = CD' + BC' + AD'$$



## Correzione esercizio 2.2

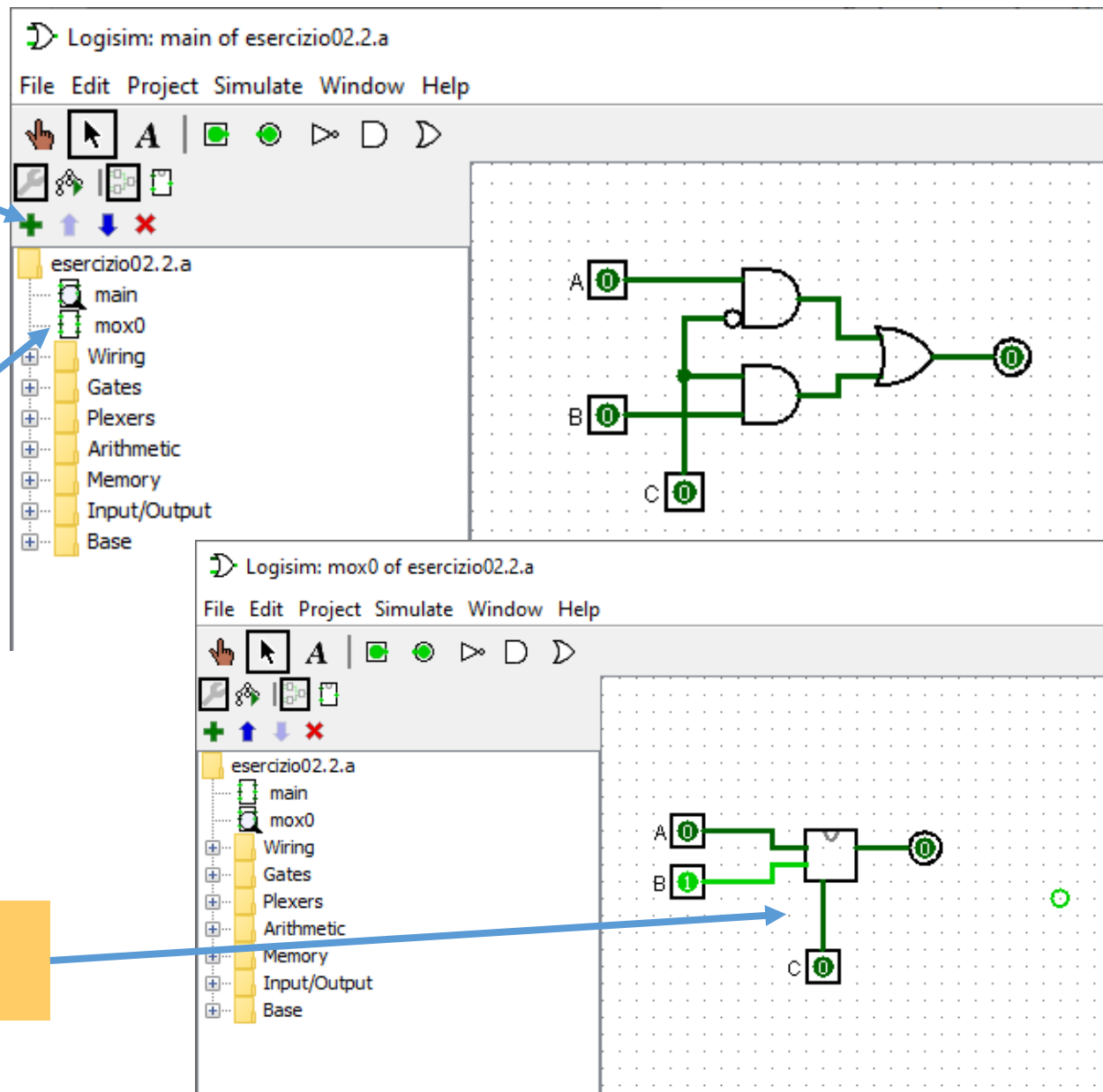
- a) Costruire un multiplexer con 1 ingresso di controllo, e realizzarlo come modulo Logisim.
- a) Utilizzando tre multiplexer con 1 ingresso di controllo realizzare un multiplexer con 2 ingressi di controllo.

## Esercizio 2.2a

1. 'Add circuit',  
scegliere il nome

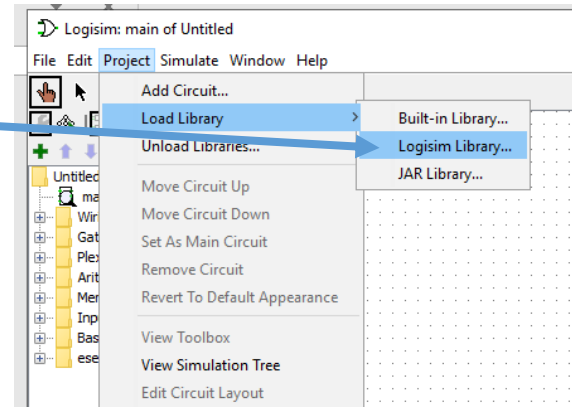
2. 'Set as  
main circuit'

3. Usare come  
modulo Logisim



## Esercizio 2.2b

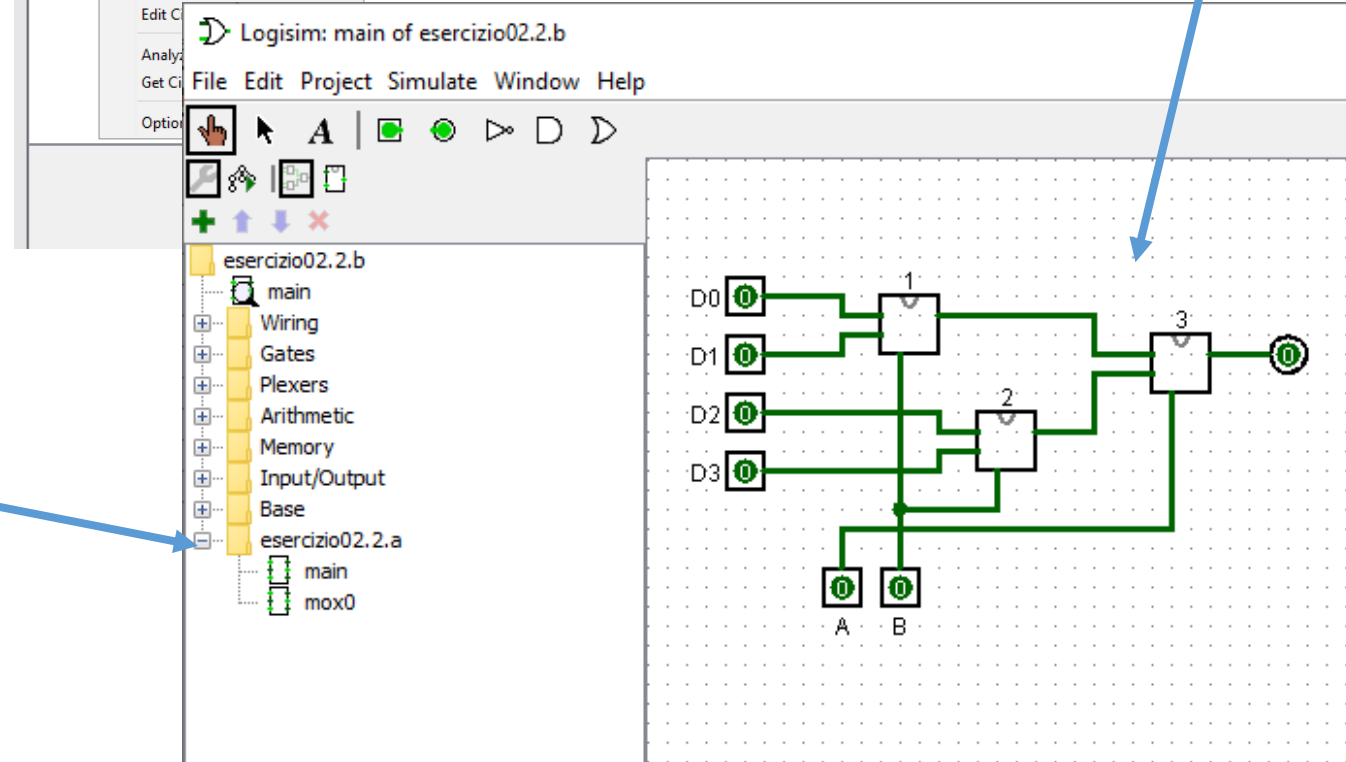
1. Carico il circuito precedente come libreria



A	B	input
0	0	D0
0	1	D1
1	0	D2
1	1	D3

n ingressi di controllo gestiscono  $2^n$  input

2. Compare nel menù ad albero



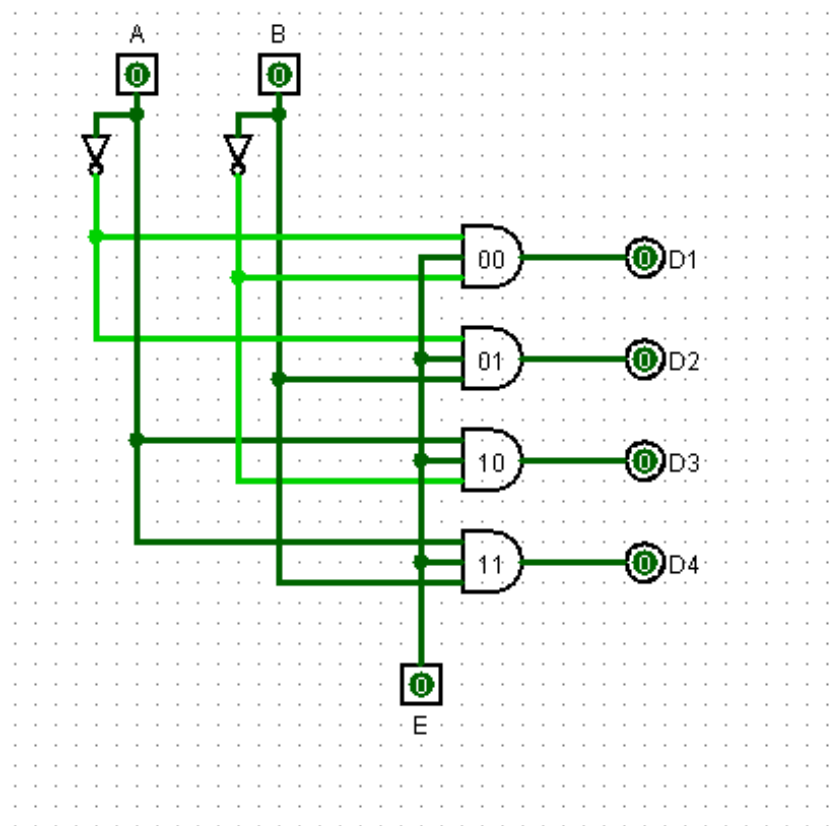
## Correzione esercizio 2.3

- a) Progettare un decoder a 2 ingressi dotato di un segnale aggiuntivo di Enable. Se il segnale Enable vale 0 tutte le uscite valgono 0; se Enable vale 1 si comporta come un circuito decoder. Realizzare il circuito come modulo.
- b) Utilizzare il modulo del punto precedente per realizzare un decoder a 3 ingressi e
- c) uno a 4 ingressi.



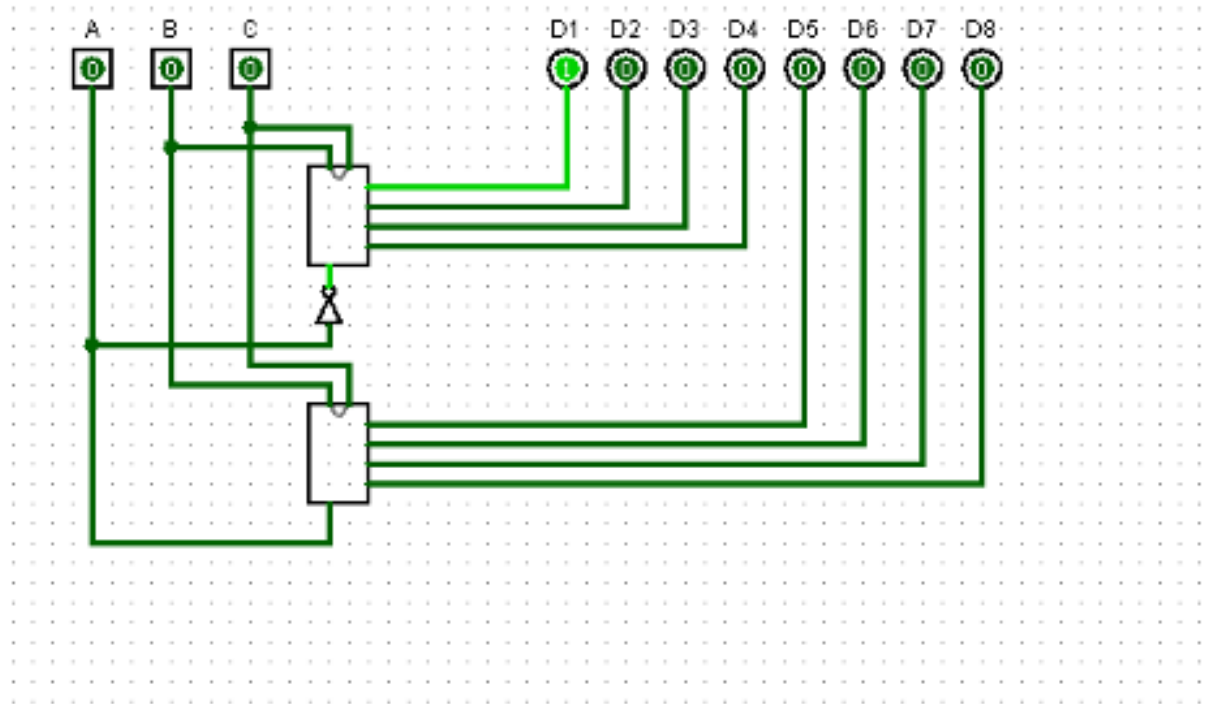
## Esercizio 2.3a -- Decoder a 2 ingressi con segnale di Enable

A	B	output
0	0	D1
0	1	D2
1	0	D3
1	1	D4



### Esercizio 2.3b -- Decoder a 3 ingressi, usando 2 decoder a 2 ingressi

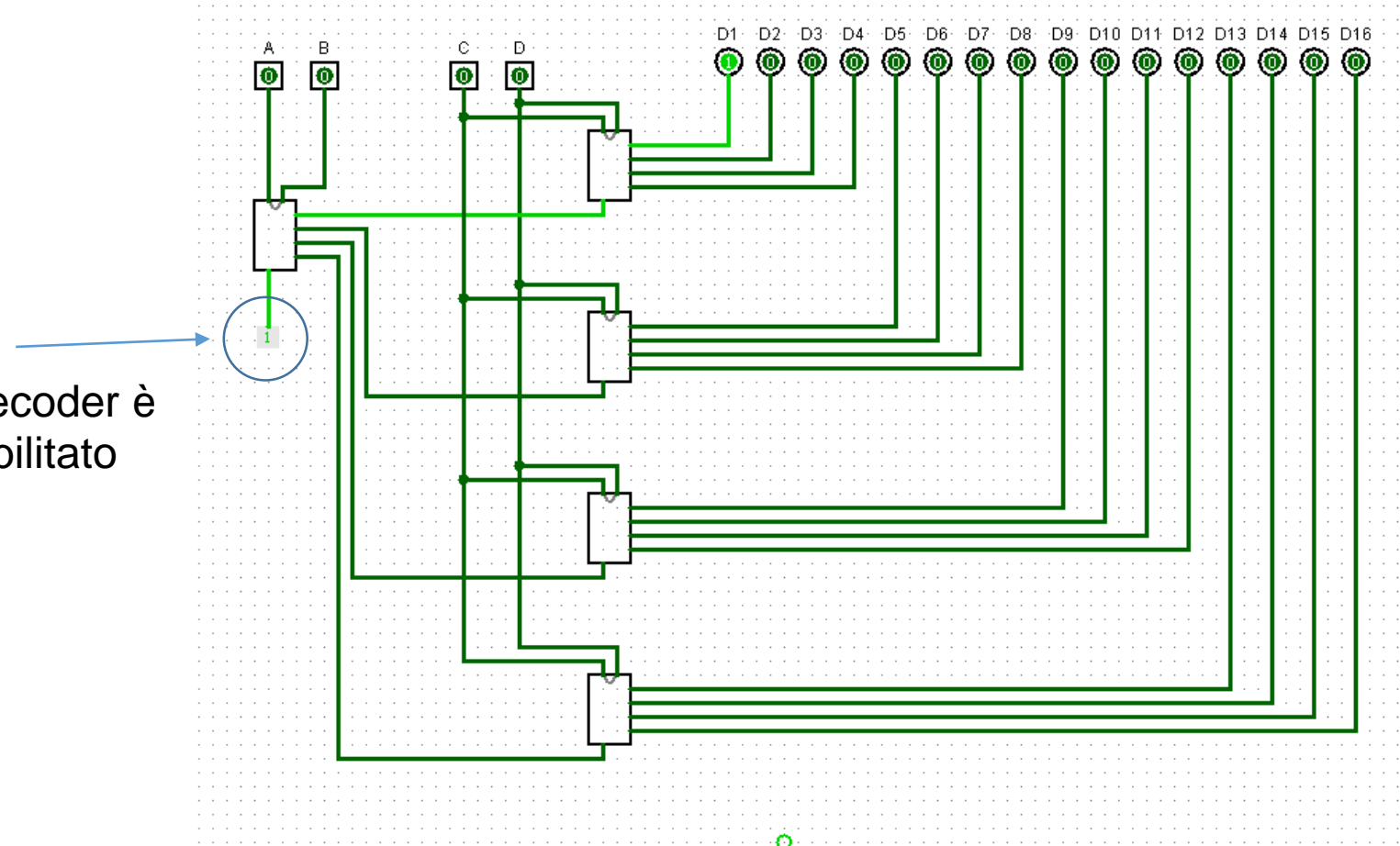
A	B	C	
0	0	0	D1
0	0	1	D2
0	1	0	D3
0	1	1	D4
1	0	0	D5
1	0	1	D6
1	1	0	D7
1	1	1	D8



A è collegato al segnale di Enable dei decoder e abilita  
alternativamente i due decoder che si comportano in maniera  
identica rispetto agli input B e C

## Esercizio 2.3c -- Decoder a 4 ingressi e $2^4=16$ uscite.

Ci servono 5 decoder a 2 ingressi



A e B tramite il primo decoder selezionano quale dei successivi 4 decoder è abilitato tramite i segnali di Enable. C e D entrano in ognuno dei successivi decoder, che hanno comportamento identico.

## Esercizio 3.1

- a) Progettare un half-adder, ossia un circuito combinatorio che somma due bit e genera il bit risultato ed un riporto. L'half-adder va realizzato come modulo (sottocircuito) Logisim.
- b) Utilizzando 4 moduli half-adder, costruire un circuito che ricevuto in ingresso un numero binario di 4 cifre, restituisca in uscita il numero binario successivo.

## Esercizio 3.2

- a) Utilizzando due moduli half-adder, progettare un full-adder, ossia un circuito combinatorio che somma due bit ed un riporto in ingresso. Il circuito genera un bit risultato ed un eventuale riporto in uscita. Realizzare il full-adder come modulo Logisim.
- b) Progettare un circuito che calcoli la somma di due numeri binari di 4 bit ciascuno.