

ABCDE (F?)

Affrontiamo Brevi Concetti Di Elettronica
(Facile?)

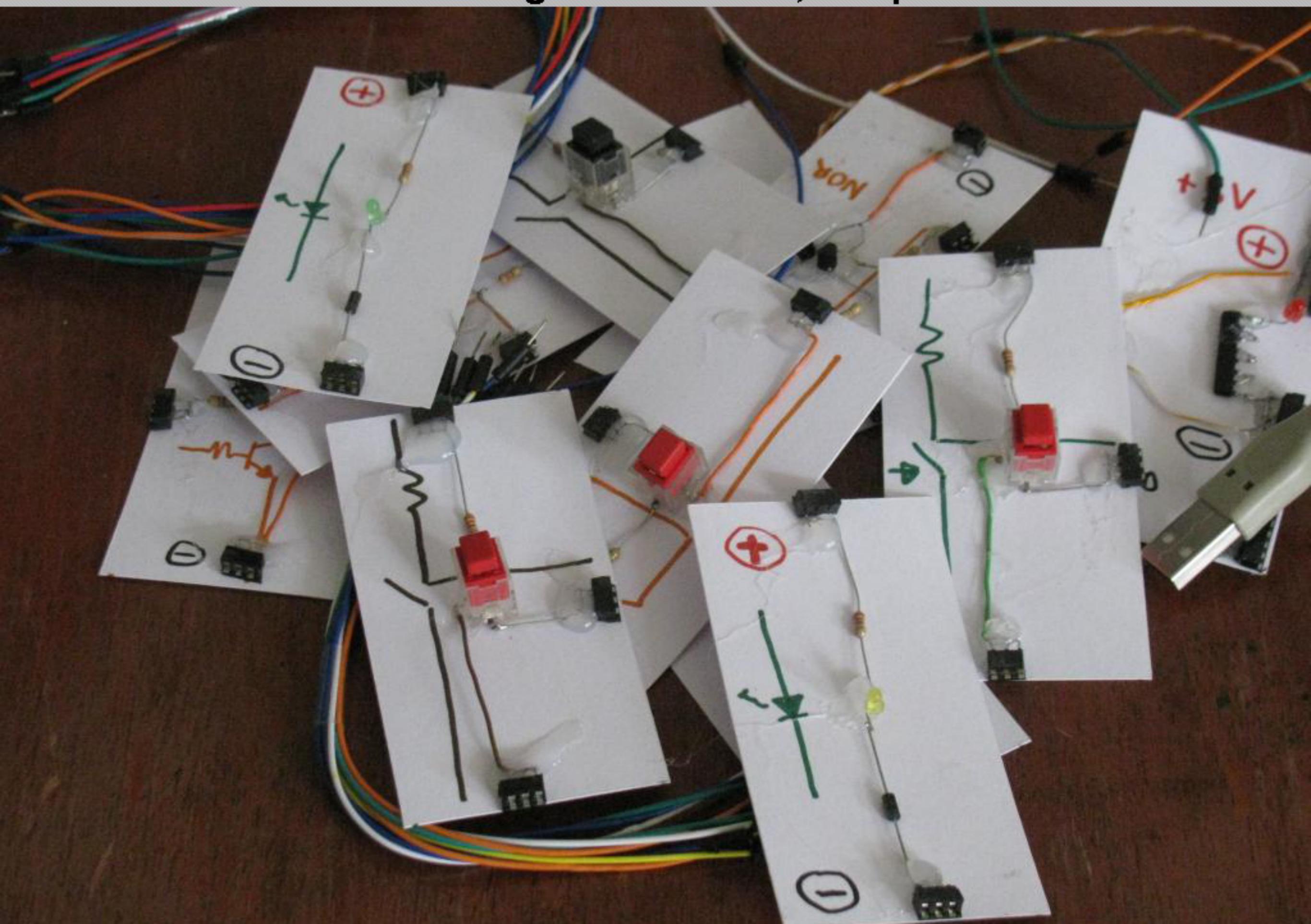
Ovvero: i primi 5 o 6 passi verso la costruzione di un computer



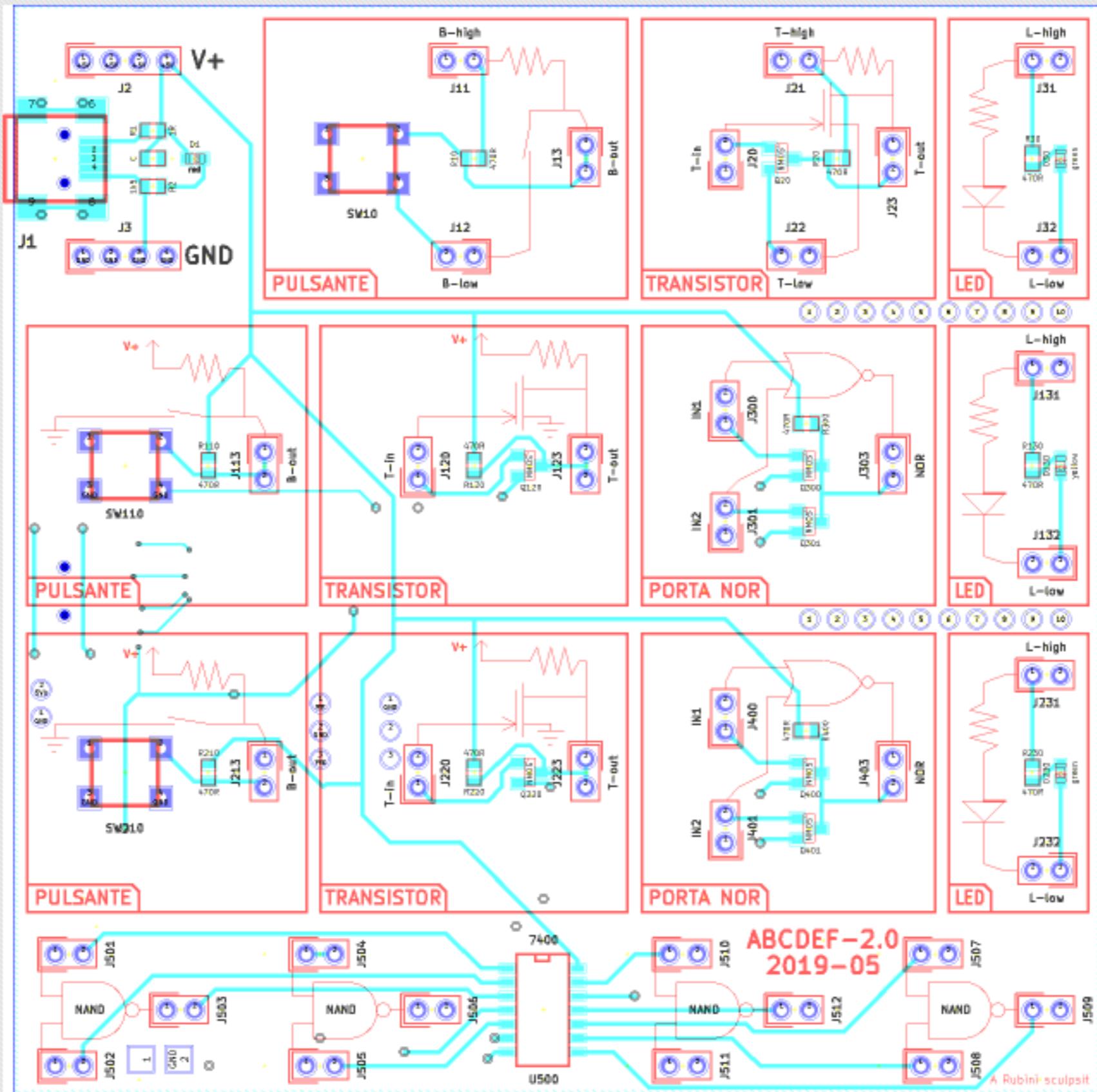
copyright © 2013,2019 - Alessandro Rubini - CC-BY-SA 3.0 Unported



E' stato un gioco di carte, a squadre



Ora e' un circuito stampato



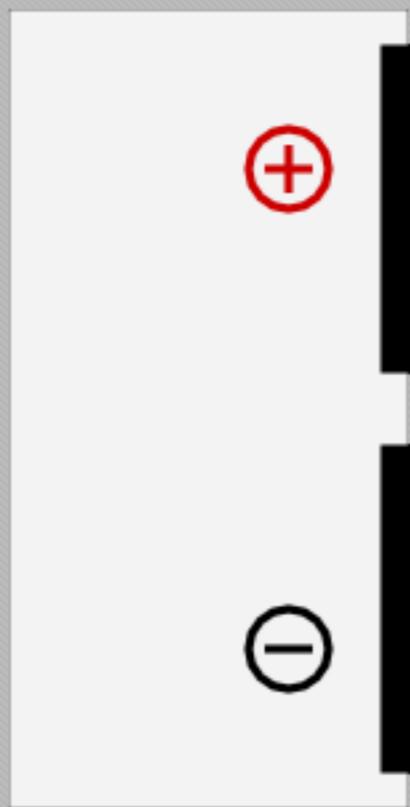
Obiettivo del gioco di squadra

Realizzare il circuito-tipo dei quiz televisivi d'epoca:

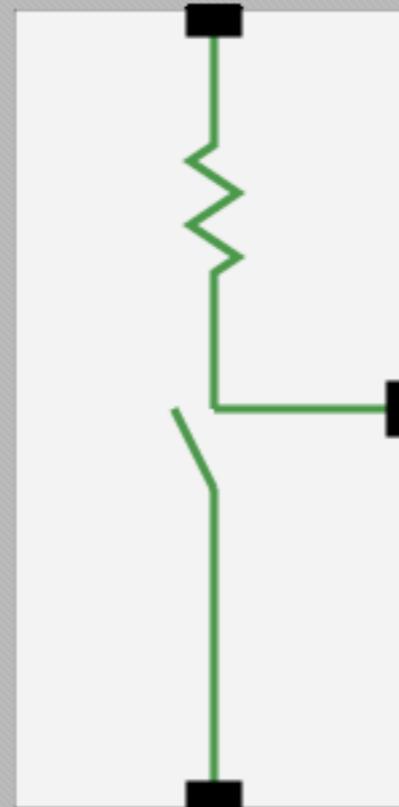
- Ci sono due concorrenti, due pulsanti due spie
- Il primo pulsante premuto accende la lampada
- Chi preme dopo non accende la lampada

Circuito realizzato con tecnologia d'epoca

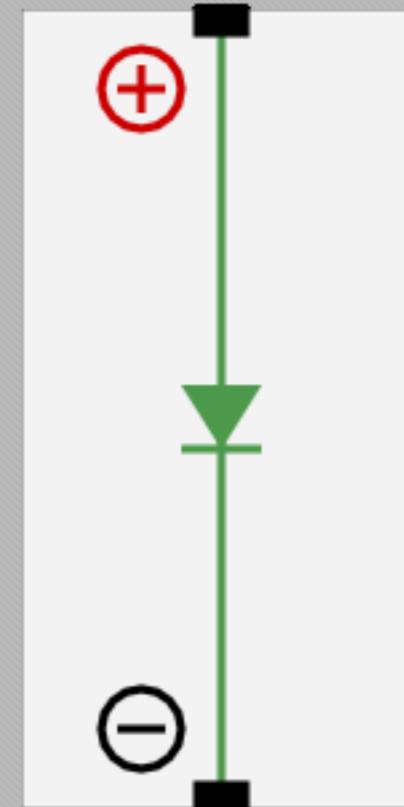
Strumenti iniziali (carte da gioco)



Alimentazione

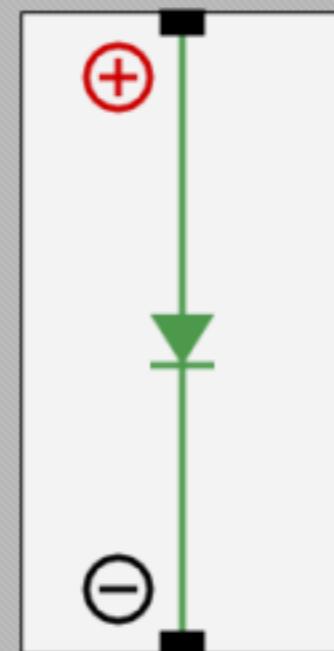
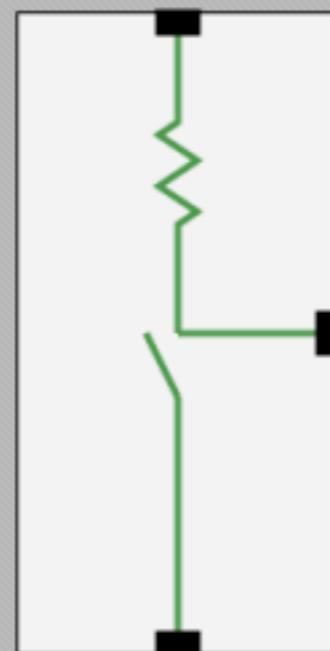
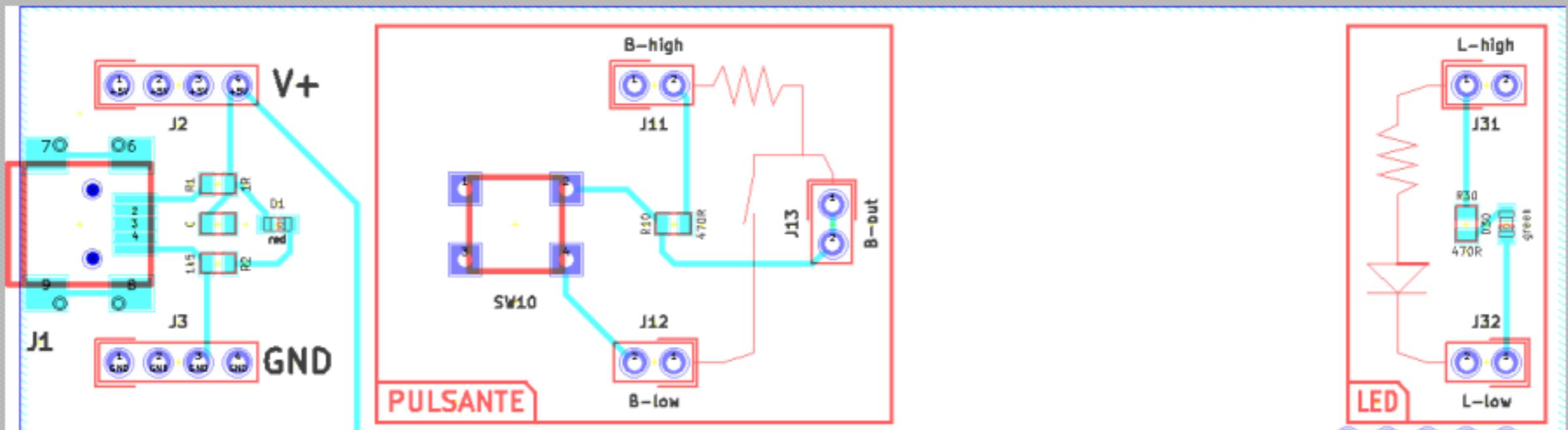


Pulsante



Lucina (LED)

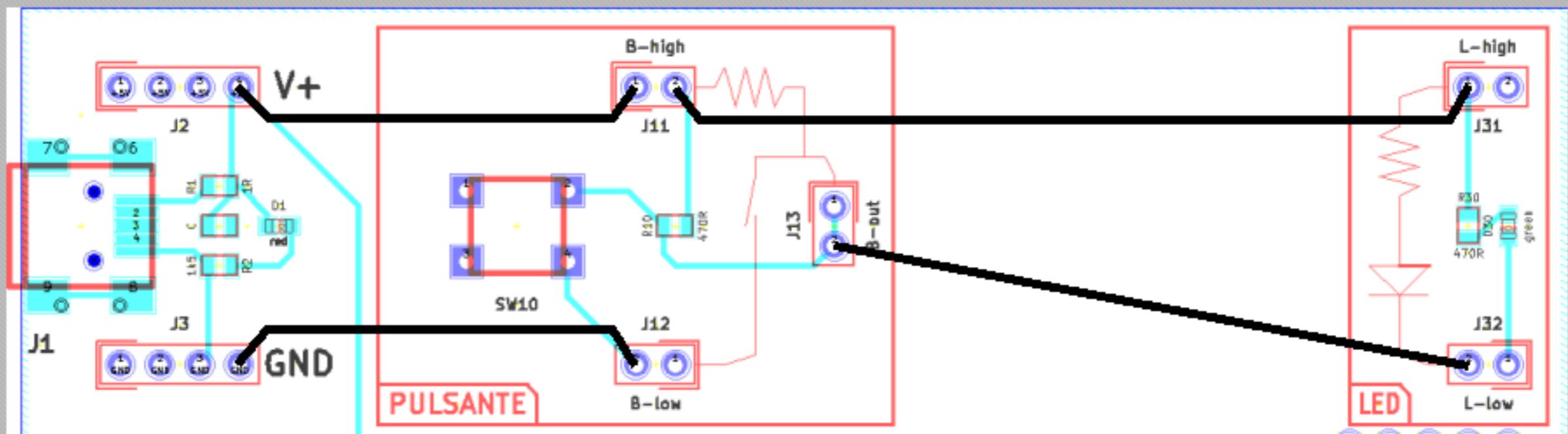
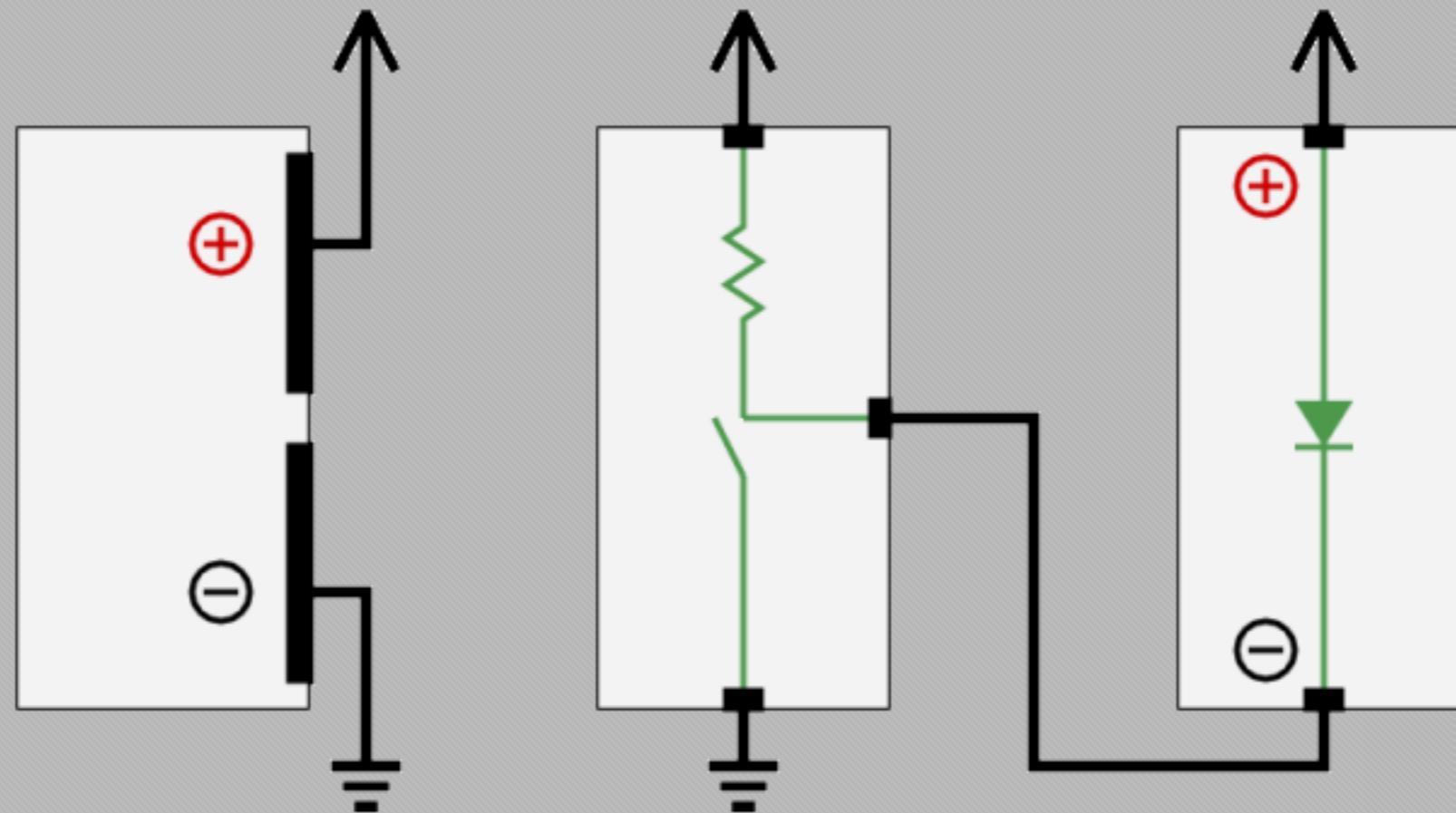
Strumenti iniziali (sul circuito)



Sfida 1: accendere un led

Accendere il LED quando viene premuto il pulsante

Soluzione 1



Teoria: logica positiva e logica negativa

In questo caso premendo il pulsante spengo il segnale

- Questa viene chiamata "logica negativa"
- Posso dire "c'e` il sole" in positivo
- Ma in negativo posso dire "non e` coperto"

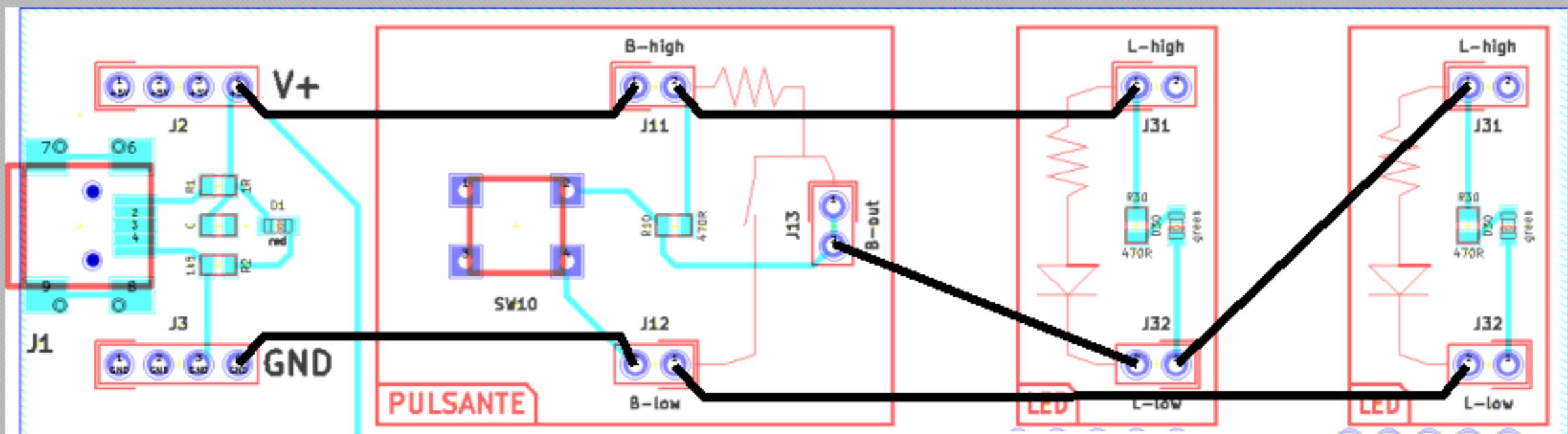
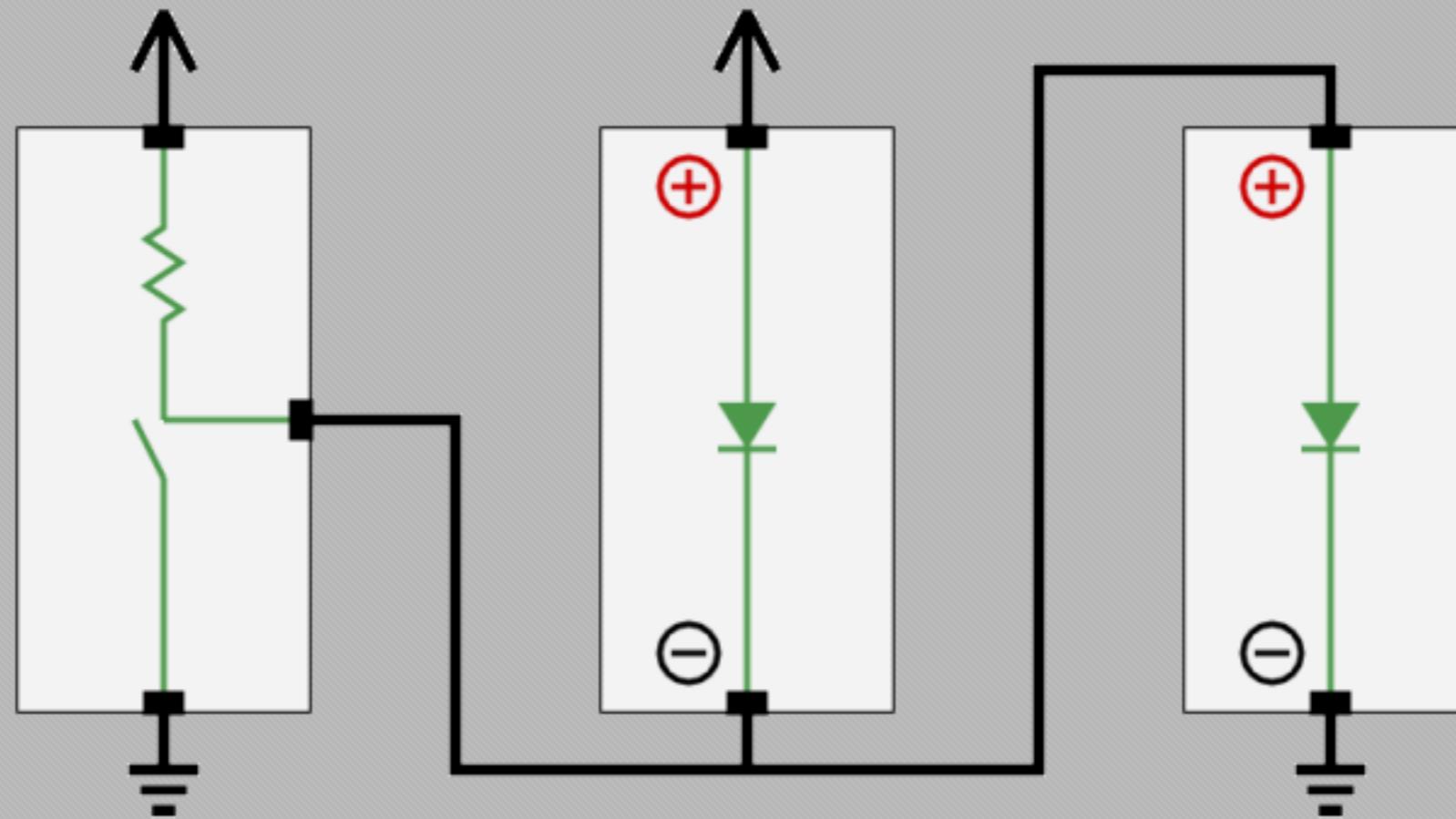
Ribaltando il pulsante si puo` fare in logica positiva

- E` consentito: il led e` polarizzato, il pulsante no
- Normalmente non si fa, vedremo perche`

Sfida 2: accendere due led

Accendere un LED e spegnere l'altro, con un pulsante

Soluzione 2



Teoria: attivo alto e attivo basso

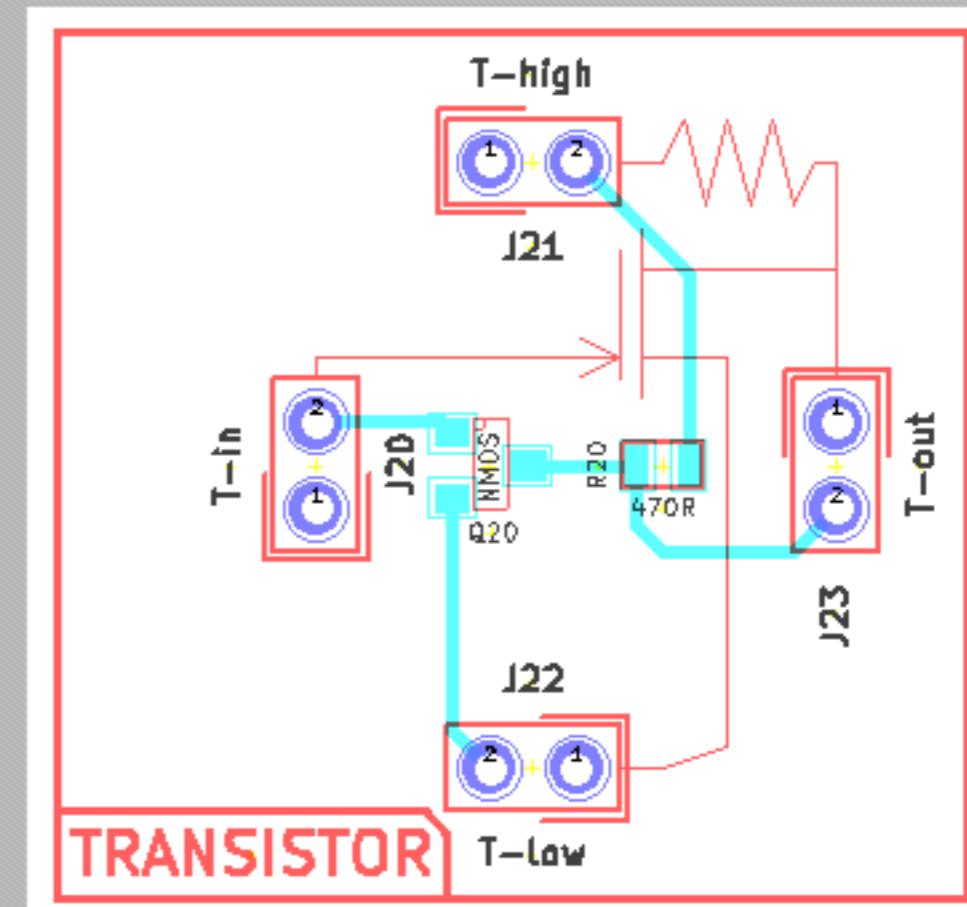
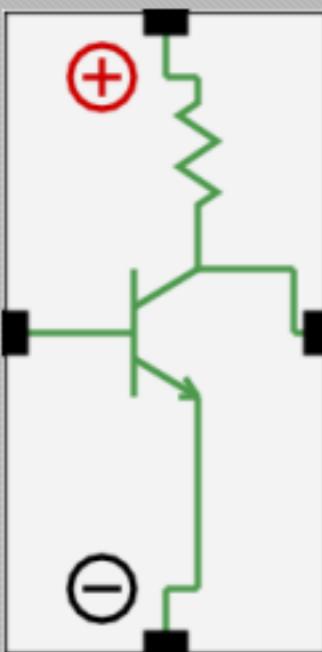
In questo caso, uno dei led e` a logica positiva

- Ma e` quello che rimane normalmente acceso
- Il pulsante invece rimane in logica negativa

Se ribaltiamo il pulsante, il risultato non cambia

- Comunque il pulsante puo` pilotare entrambe le uscite
- Si dice anche "segnale attivo alto" e "attivo basso"
- Qui lo stesso segnale fa entrambe le cose, una per led.

Strumenti: il transistor



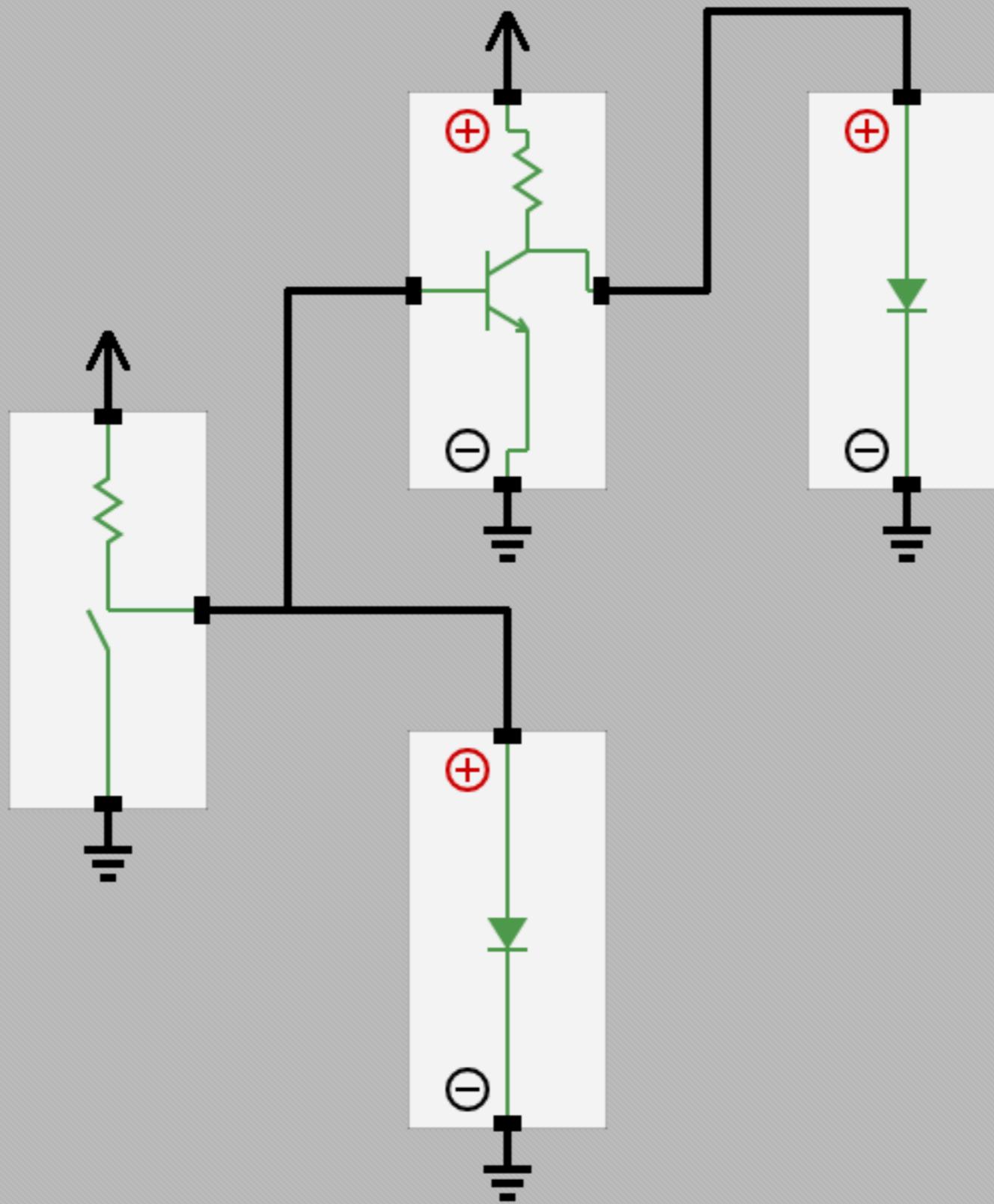
Il transistor e` come un pulsante ma senza il dito
Invece del dito si usa un segnale in "base", in logica positiva.
Gli altri piedini si chiamano "collettore" ed "emettitore"
(oppure gate, drain, source, ma evitiamo complicazioni)

Sfida 3: la porta NOT

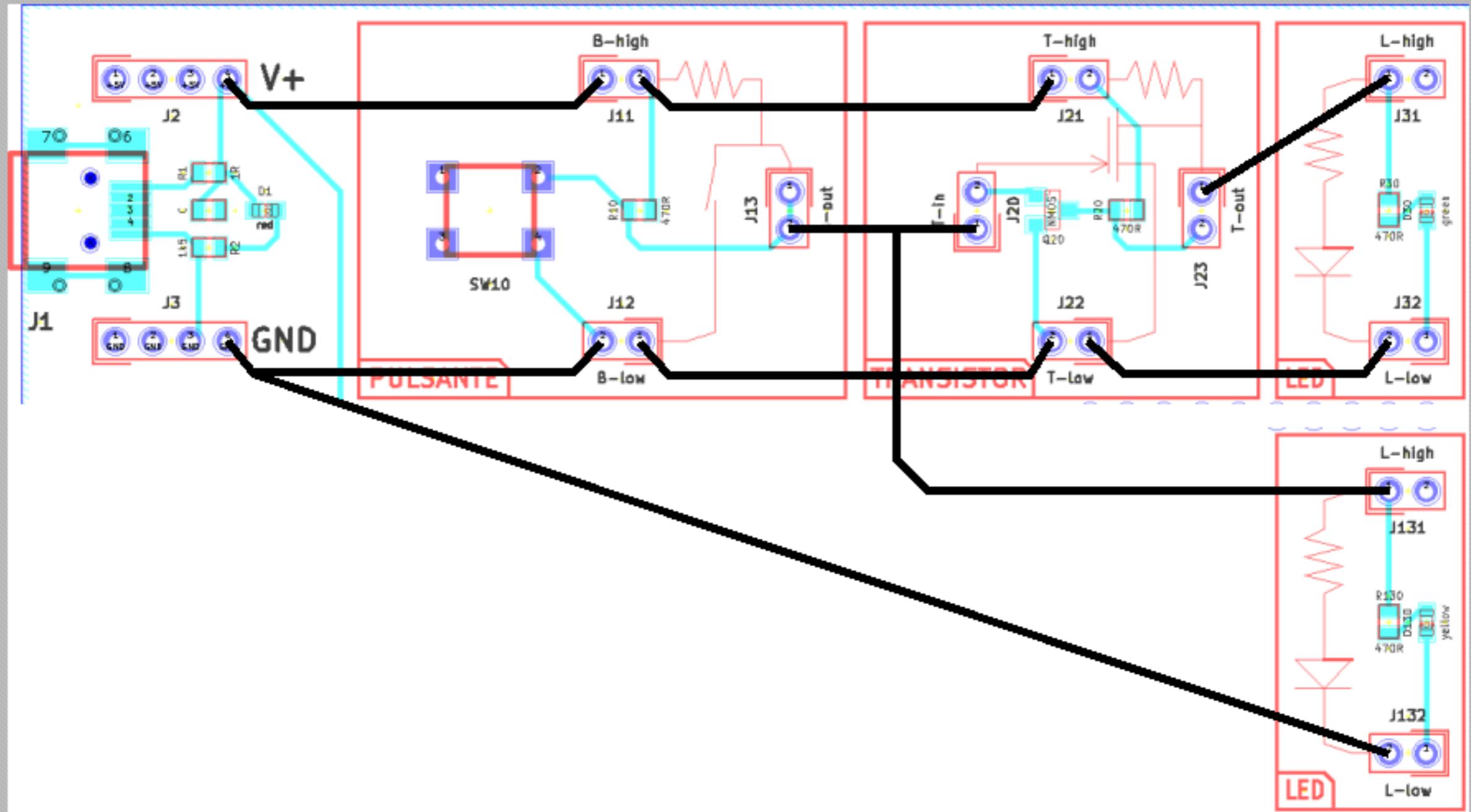
Costruire un circuito che inverta un segnale

**Usare il pulsante per accendere un led e spegnere l'altro,
utilizzando due led collegati in logica positiva**

Soluzione 3 (con le carte)

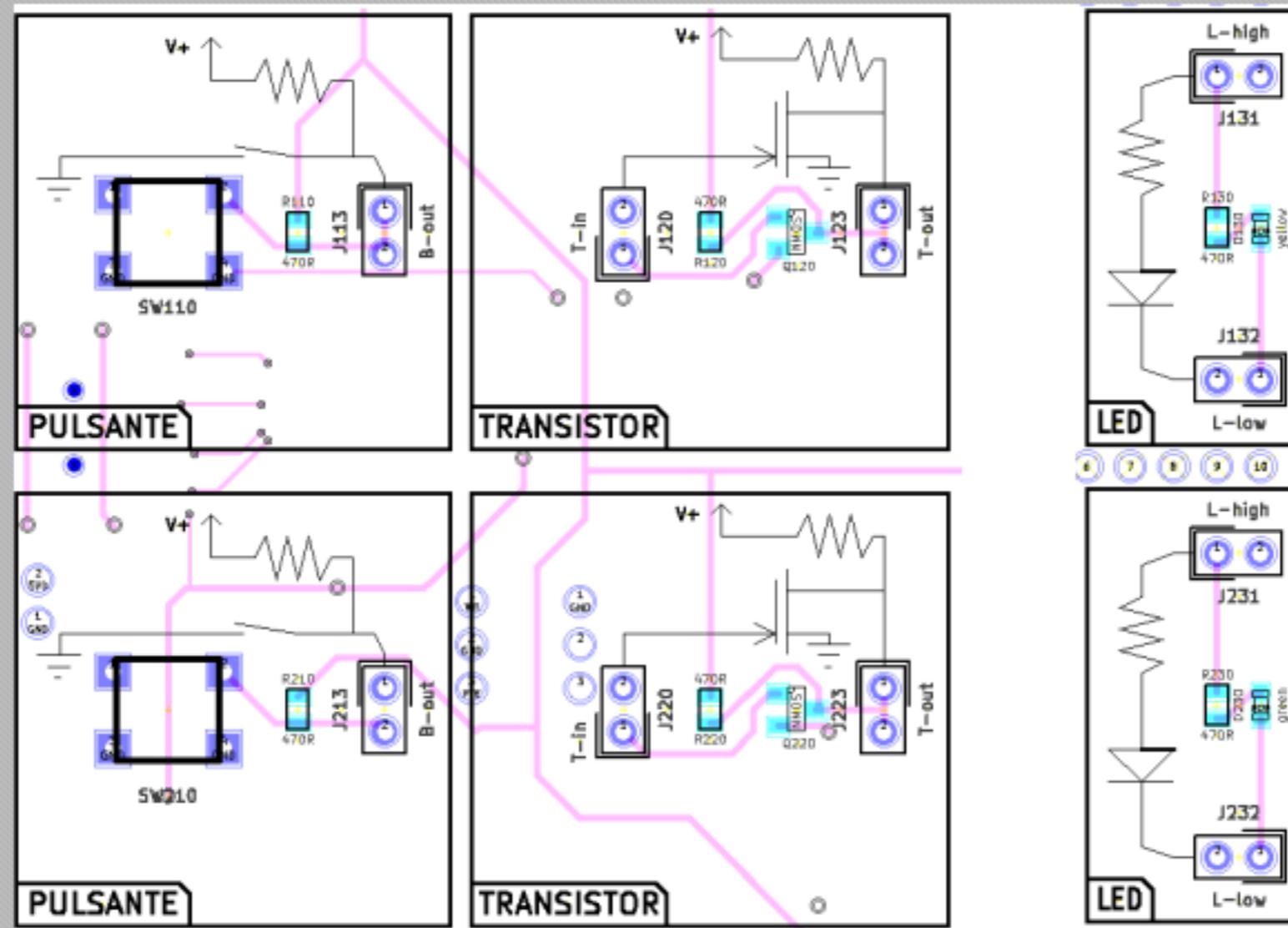


Soluzione 3 (con il circuito)



Nuovi strumenti

Nella soluzione precedente, molti fili servono solo per alimentare (GND, 5V). Ora useremo componenti già alimentati (seconda e terza riga del circuito).



Teoria: la porta logica

La porta logica e` un circuito che combina due segnali

- La porta NOT appena vista e` il caso piu` semplice

La porta AND e` un circuito fondamentale

- Se ho tempo e c'e` il sole esco in bicicletta

La porta OR e` l'altro circuito fondamentale

- Se c'e` freddo o piove mi vesto pesante

La porta logica e` in pratica una tabella 2x2

0	0
0	1

0	1
1	1

Teoria: la porta invertente

Posso definire funzioni logiche invertenti

La porta NAND (not and): se A e B allora nego

- Se ho tempo e c'e` il sole non resto in casa

La porta NOR (not or): se A oppure B, allora nego

- Se c'e` freddo o piove non mi vesto leggero

Altre due tavelle 2x2

1	1
1	0

1	0
0	0

Teoria: tutte le porte logiche

Ci sono 16 possibili tabelle per due possibili ingressi

- Solo 8 tabelle simmetriche sui due ingressi
- Due sono banali (tutti zeri, tutti uni), ne restano 6
- (N)AND, (N)OR, già viste, e due che qui non trattiamo

Le 4 porte logiche sono tra loro molto simmetriche

0	0	0	1
0	1	1	1

1	1	1	0
1	0	0	0

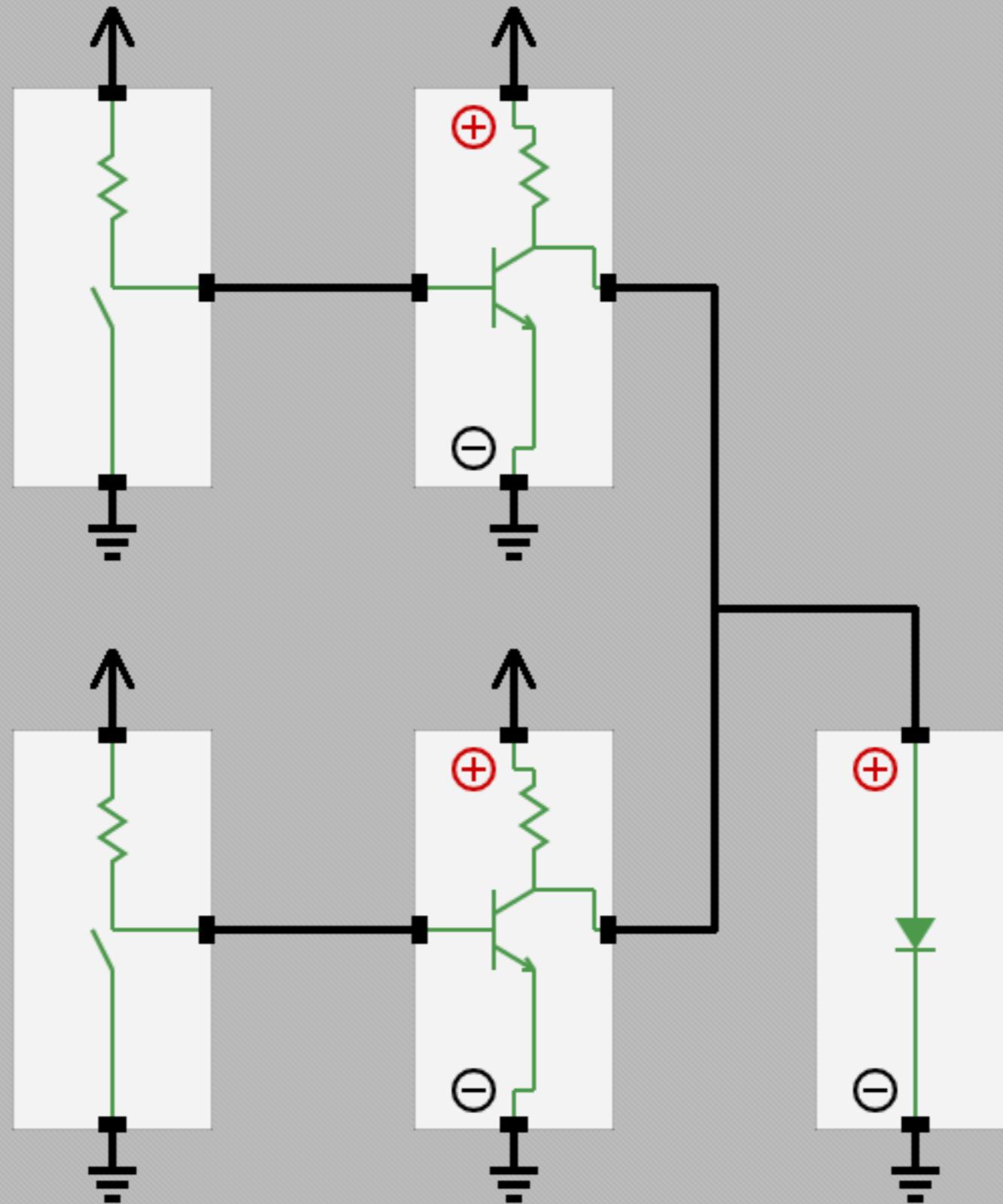
E, cambiando la polarità, una vale l'altra

- NOR: Se devo lavorare o c'è brutto, non esco in bici

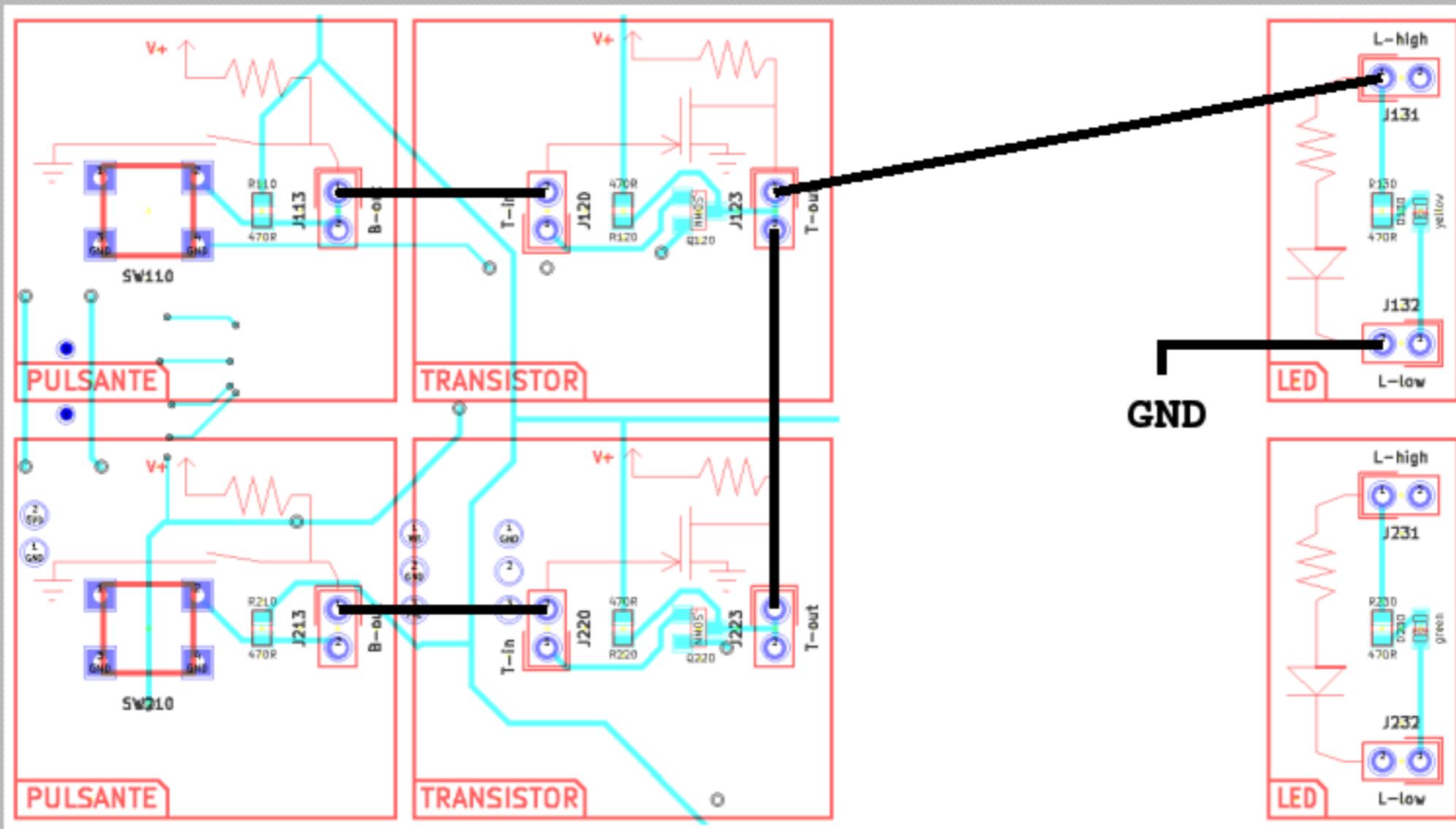
Sfida 4: la porta NOR

**Costruire una porta NOR con due transistor
Usare due pulsanti per l'attivazione**

Soluzione 4



Soluzione 4 (con il circuito)



Teoria: analisi della soluzione

Una porta NOR e` fatta da due transistor

- I due ingressi vanno alle due basi
- Le due uscite vanno collegate in comune

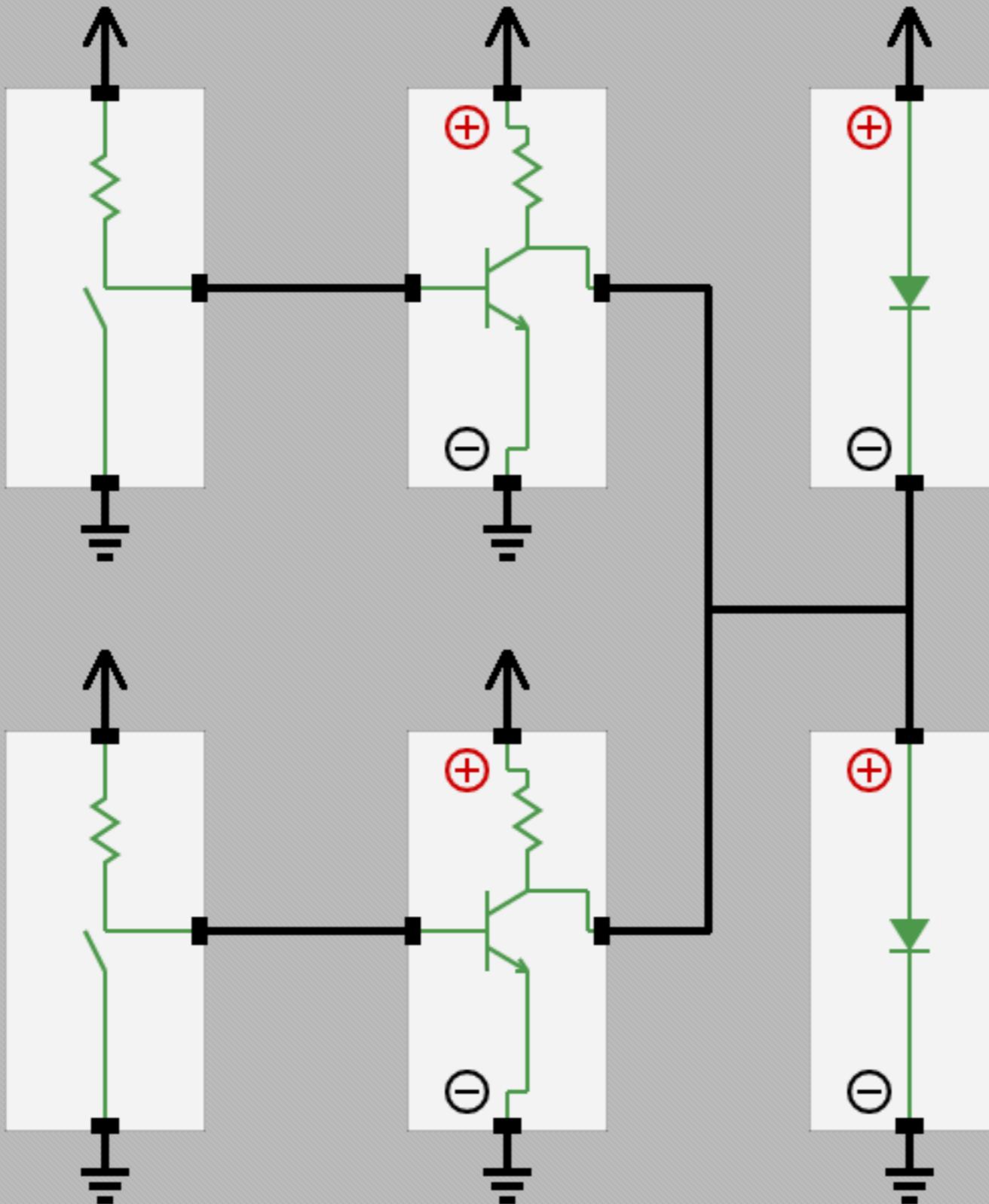
I pulsanti sono in logica negativa, l'uscita positiva

- Abbiamo una porta AND (accesso se entrambi premuti)

Se invertiamo anche il led, otteniamo una NAND

- Quindi una NOR in logica negativa e` una NAND
- Anche OR diventa AND, e viceversa.

Soluzione 4b: che porta logica e`?



like an infinite resistance. This causes the output of the circuit, V_{out} , to take on a value close to V_{cc} , an externally regulated voltage, typically +5 volts for this type of transistor. When V_{in} exceeds the critical value, the transistor switches on and acts like a wire, causing V_{out} to be pulled down to ground (by convention, 0 volts).

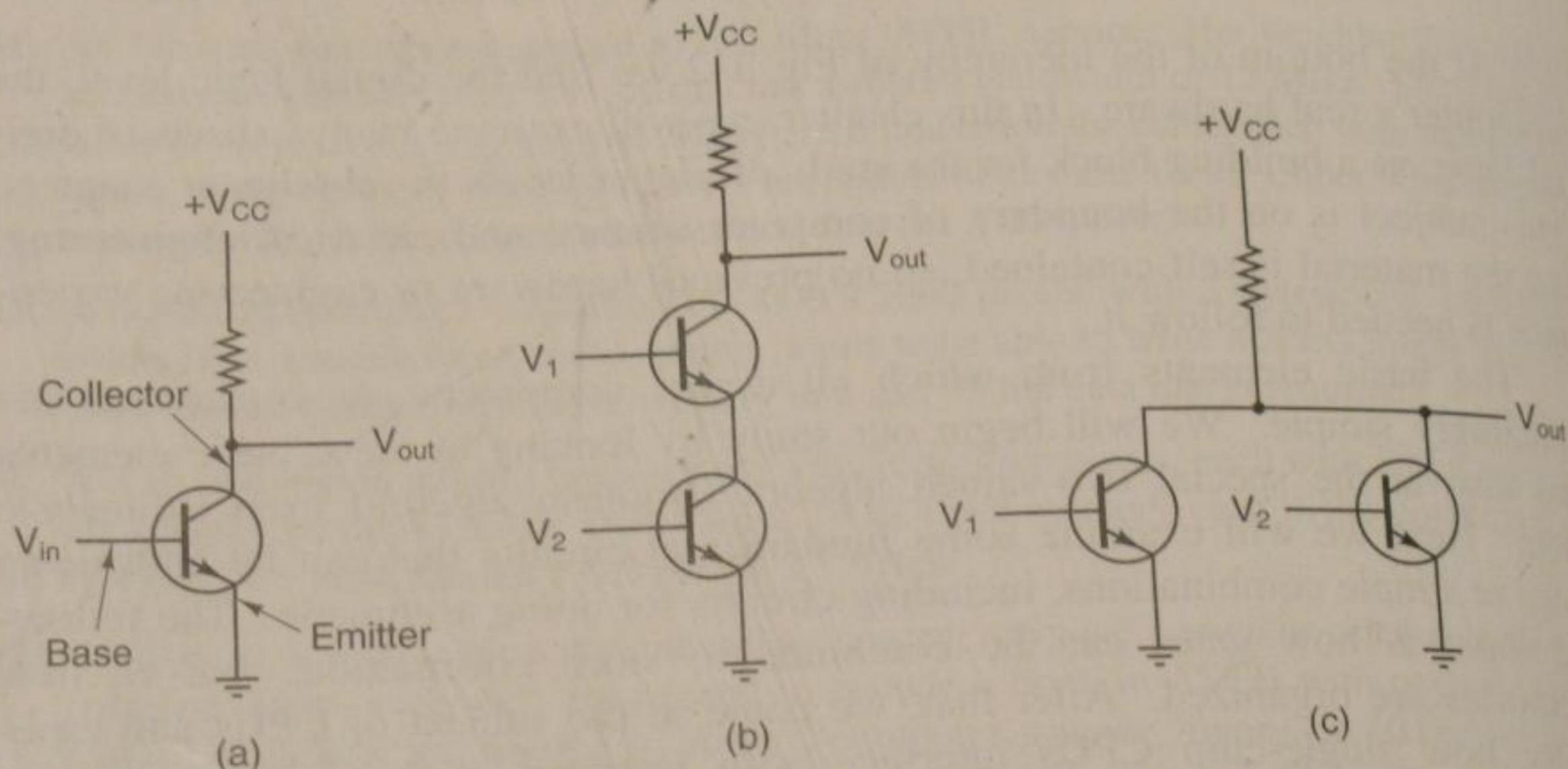
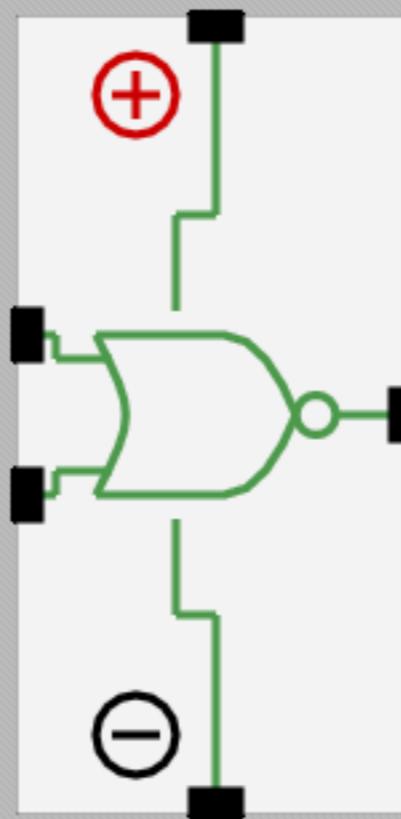


Figure 3-1. (a) A transistor inverter. (b) A NAND gate. (c) A NOR gate.

Queste porte logiche sono "da manuale"

The important thing to notice is that when V_{in} is low, V_{out} is high, and vice versa. This circuit is thus an inverter, converting a logical 0 to a logical 1, and a logical 1 to a logical 0. The resistor (the jagged line) is needed to limit the amount of current drawn by the transistor so it does not burn out. The time

Strumento: la porta NOR

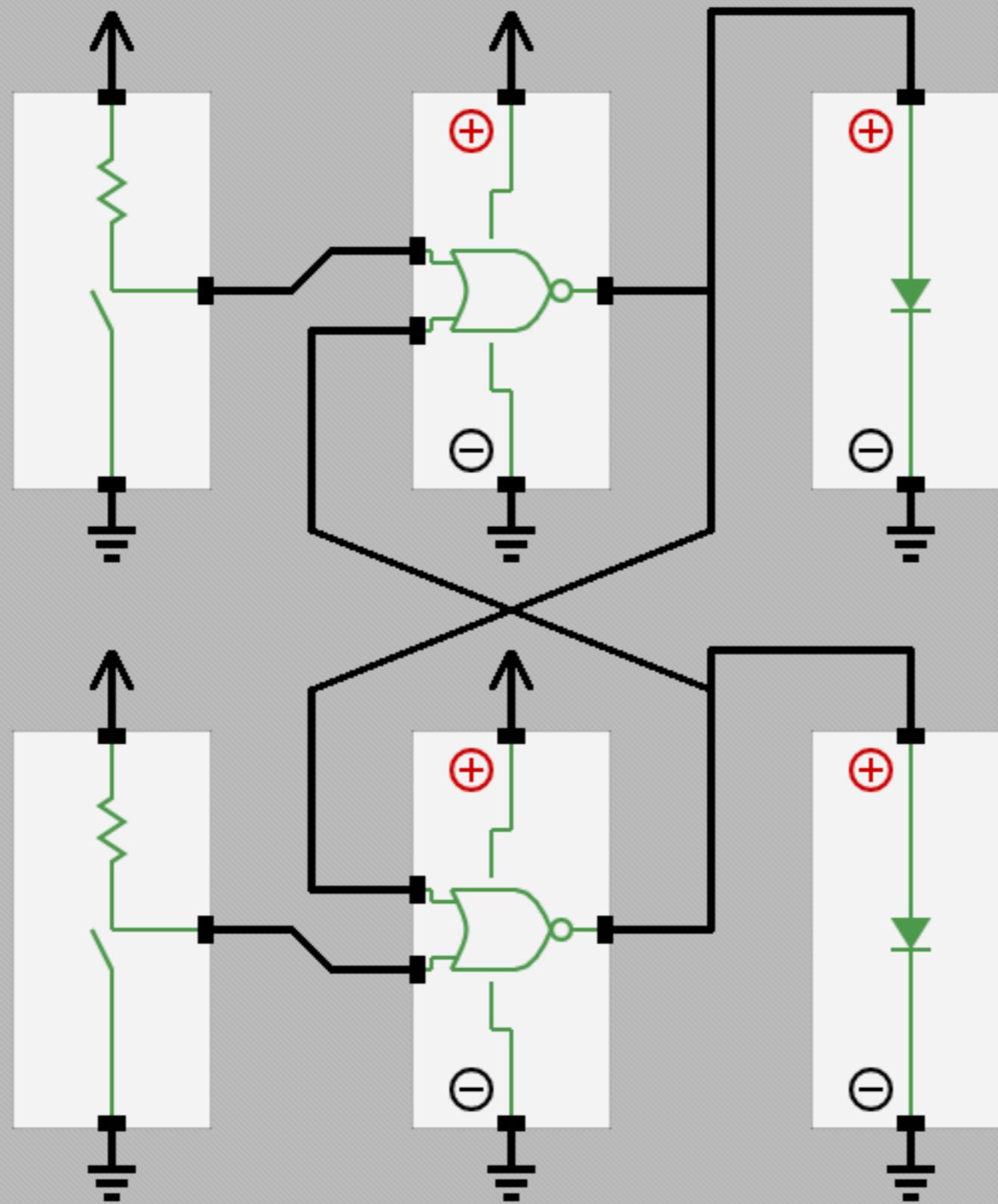


L'ultima carta del gioco e` una porta NOR gia` fatta
(e due di tali porte sono presenti nel circuito equivalente)

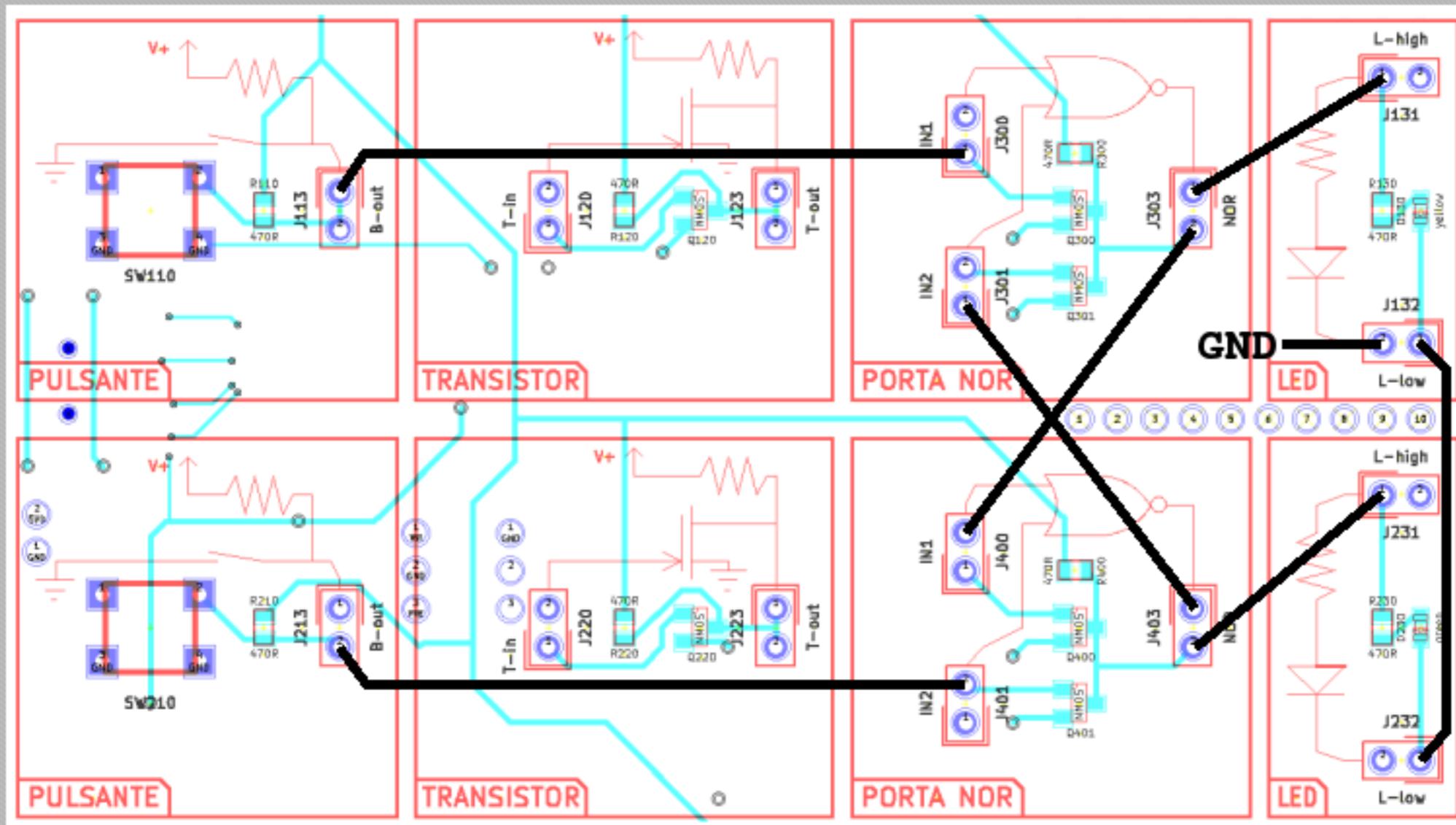
Sfida 5: il circuito del quiz

Realizzare "il primo accende" usando due porte NOR.

Soluzione 5



Soluzione 5 (con il circuito)



Il quiz a tre pulsanti, hardware

Se affrontassimo il problema con tre pulsanti

- Dovremmo usare 9 transistor
- Ogni led dipende dal suo pulsante
- Ma dipende anche da entrambe le altre uscite

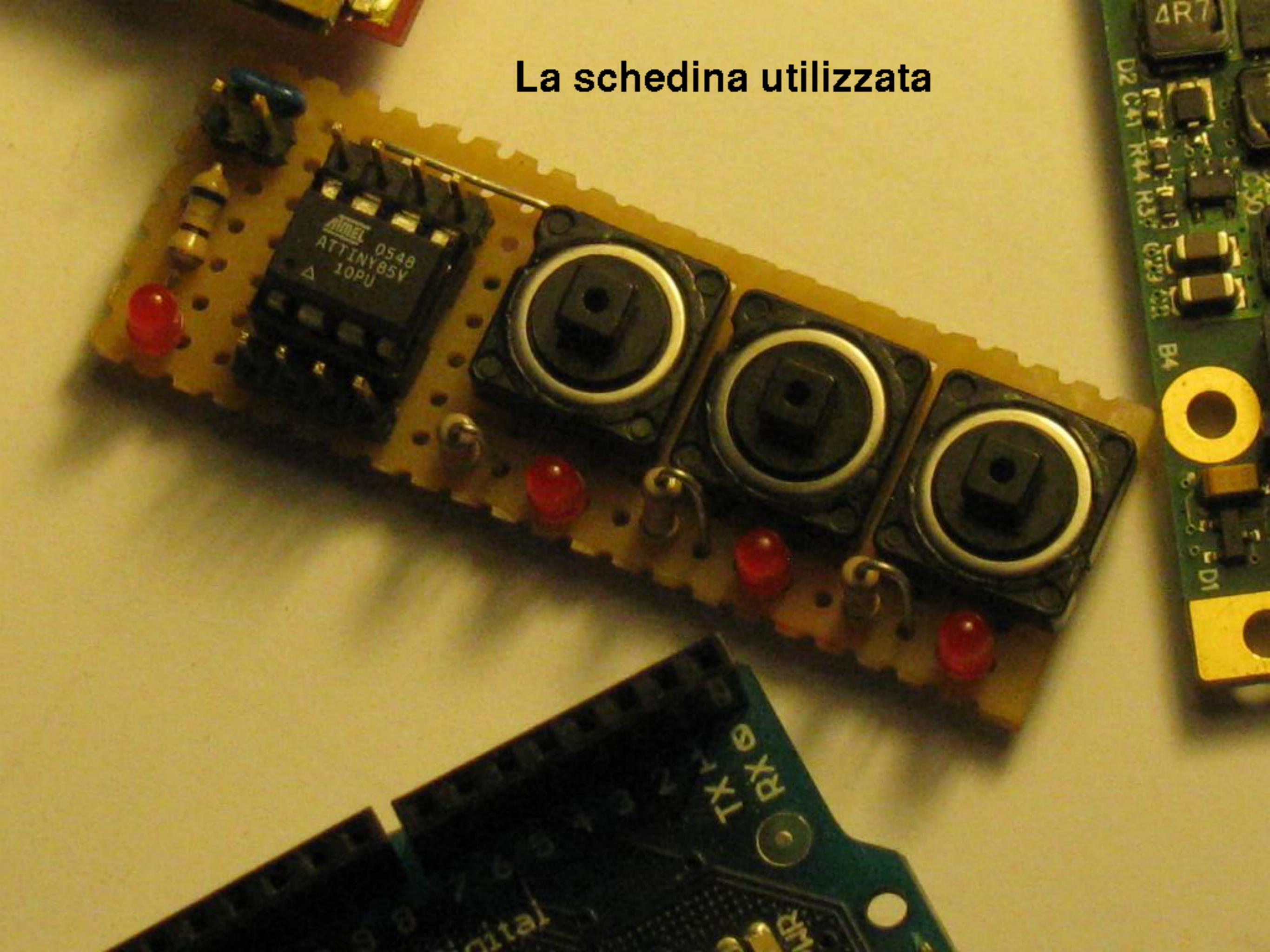
Stiamo facendo circuiti con retroazione

- L'uscita dipende dallo stato interno del sistema
- E lo stato interno dipende dalla storia passata
- Non e` un problema combinatorio ma sequenziale

Risulta, perciò, piu` facile usare un microcontrollore

- Non sono nove transistor, ma decine di migliaia
- Il sistema puo` svolgere un programma, nel tempo.

La schedina utilizzata



Il quiz a tre pulsanti, software

```
while (1) {
    /* read input */
    buttons = ~PINB;

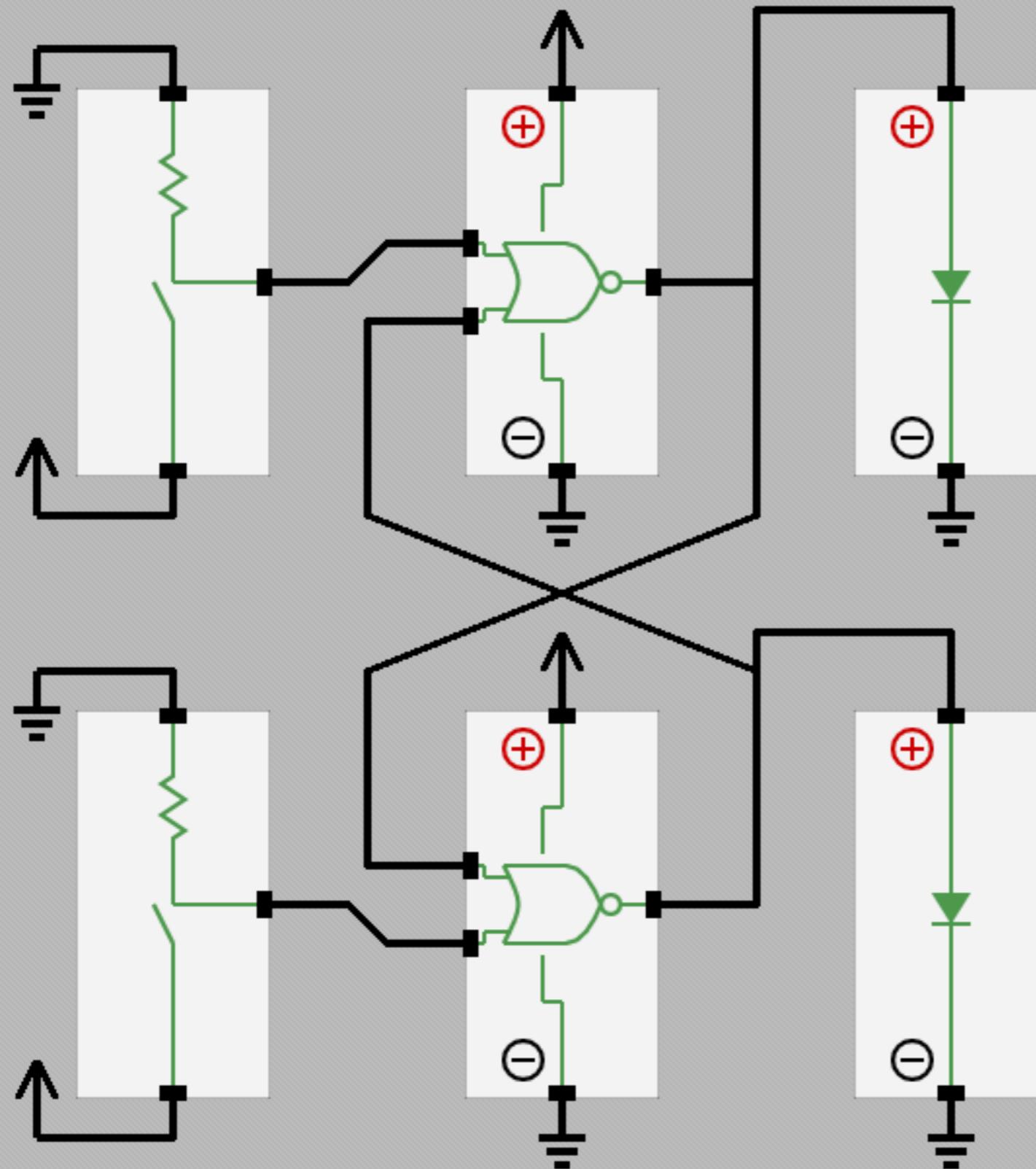
    /* calculate */
    if (!status) {
        if (buttons & IN1) status = OUT1;
        if (buttons & IN2) status = OUT2;
        if (buttons & IN3) status = OUT3;
    } else {
        if ((buttons & (IN1 | IN2 | IN3)) == 0)
            status = 0;
    }

    /* write output */
    PORTB = status;
}
```

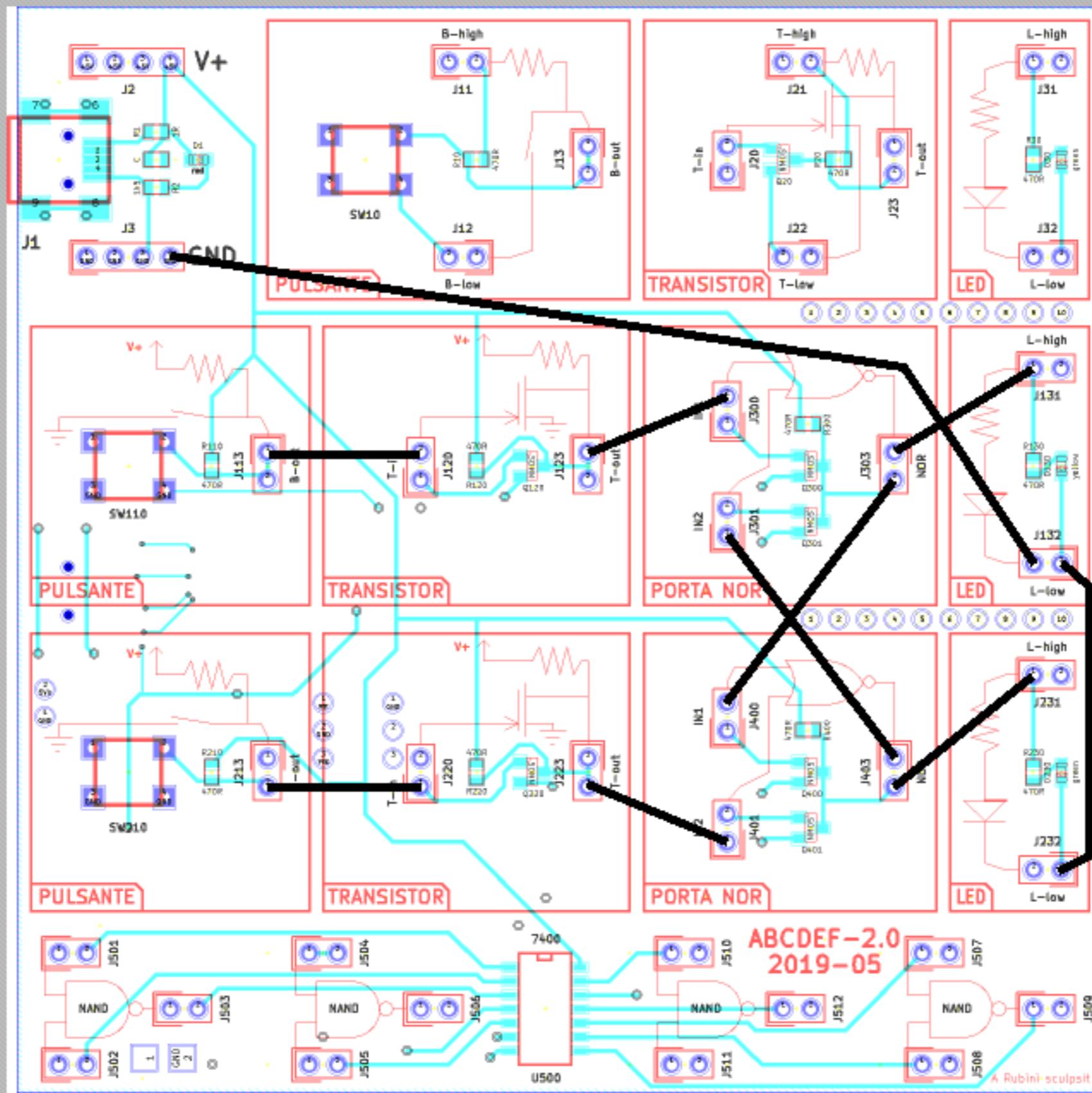
Sfida 6: il circuito del ziug

Proviamo ad invertire, i pulsanti,
usando il circuito del quiz con ingressi positivi

Soluzione 6



Soluzione 6 (con il circuito)



Teoria: il flip flop

Il circuito appena realizzato e' un flip-flop

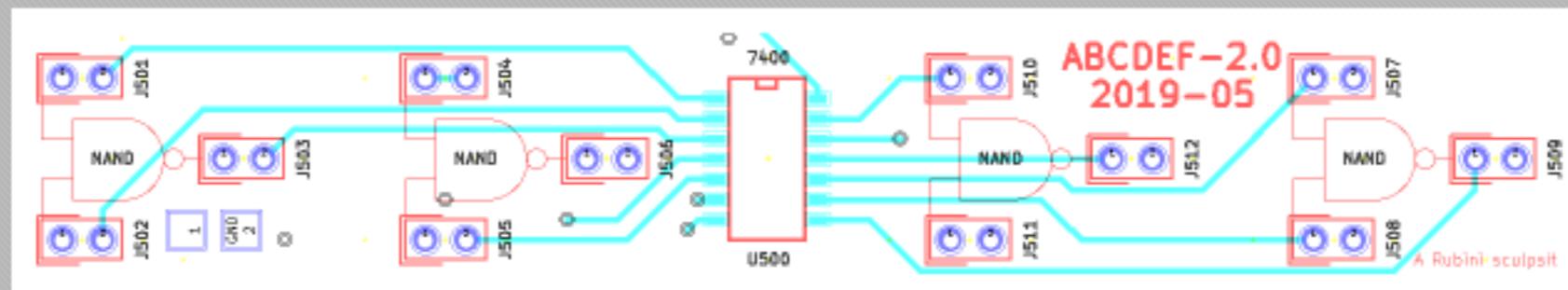
- Si tratta di una cella di memoria da 1 bit
- Uno dei pulsanti accende, l'altro spegne

Il flip-flop della letteratura e' diverso

- E` realizzato con due porte NAND, ingressi attivi bassi
- Che corrisponde al nostro NOR con ingressi attivi alti

Il "primo" circuito logico, codice 7400, sono 4 porte NAND

- Il circuito di ABCDEF contiene all'ultima riga un 7400
- E` possibile realizzare due flip-flop, come compito a casa



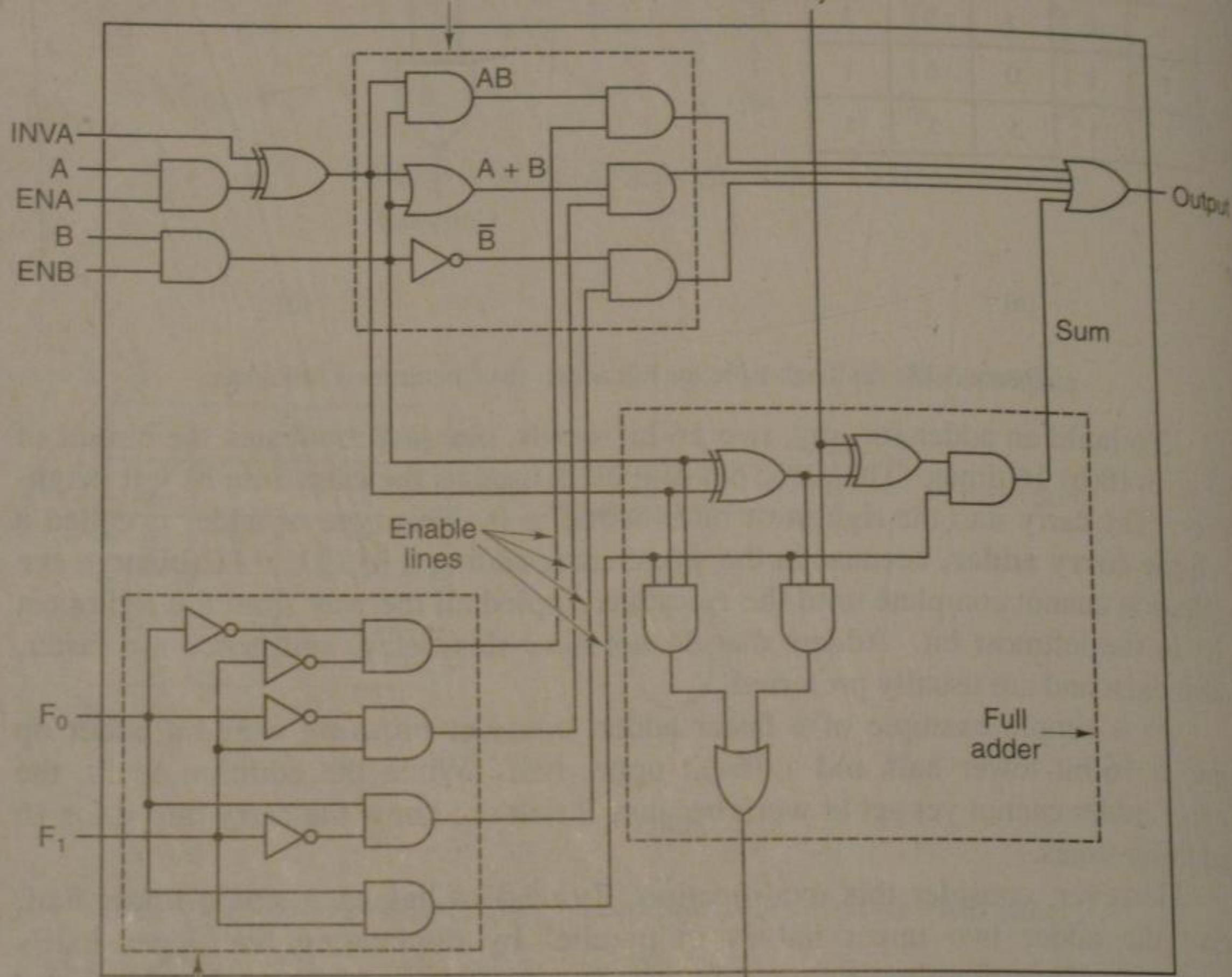
Dal flip-flop al computer

Il flip-flop e` una cella di memoria da un bit

- 8 flip-flop fanno un byte
- 1024 byte fanno un kB, eccetera

Ma un calcolatore deve anche fare i calcoli

- Quello che ci manca e` una unita` "aritmetico-logica"
- Potremmo pensare, come il flip-flop, di partire da 1 bit



Con 20 porte logiche si fa una ALU 1-bit

Decoder

Carry out

The lower right-hand corner of the ALU contains a full adder for computing the sum of A and B , including handling the carries, because it is likely that several of these circuits will eventually be wired together to perform full-word operations. Circuits like Fig. 3-19 are actually available and are known as **bit slices**. They allow the computer designer to build an ALU of any desired width. Figure 3-20 shows an 8-bit ALU built up of eight 1-bit ALU slices. The INC signal is only useful for addition operations. When present, it increments (i.e., adds 1 to) the result, making it possible to compute sums like $A + 1$ and $A + B + 1$.

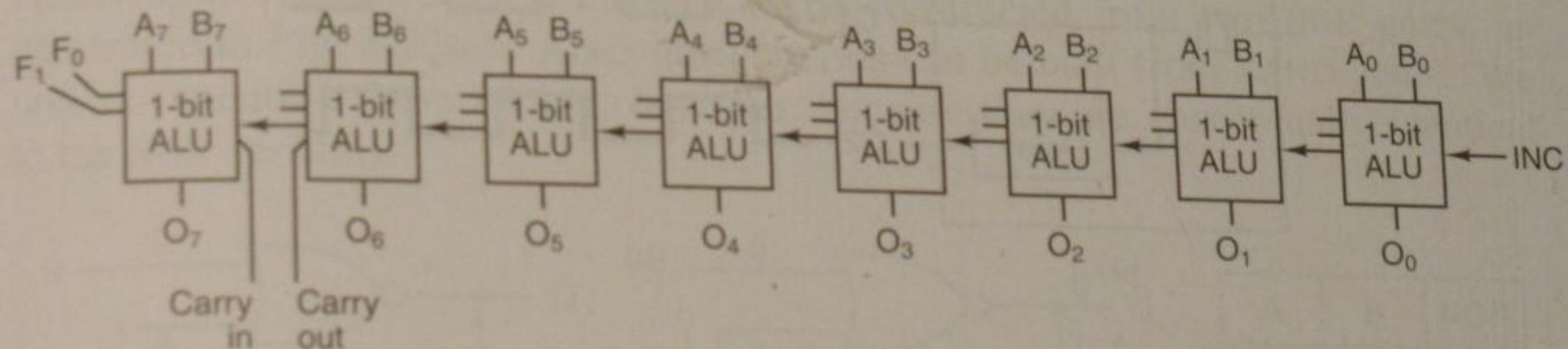


Figure 3-20. Eight 1-bit ALU slices connected to make an 8-bit ALU. The enables and invert signals are not shown for simplicity.



Con pochi registri si fa un computer

Nandputer: fatto con sole porte NAND



<http://blog.kevtris.org/?p=62>

Volume 2 / Seminumerical Algorithms

Poi, programmare e' un'altra arte

DONALD E. KNUTH Stanford University

THE ART OF
COMPUTER PROGRAMMING

THIRD EDITION

Boston · San Francisco · New York · Toronto · Montréal
London · Munich · Paris · Madrid
Singapore · Tokyo · Singapore · Mexico City