



HARDWARE BASICS

build kits · khanin.info

Building instructions

HB-02

How a CPU Counts

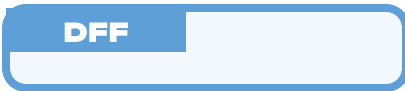
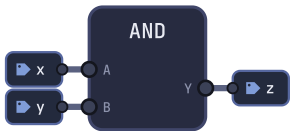
lessons 1-6 · from a bit to a calculator



simulator: cs.khanin.info

How to read these instructions

3



step number – follow strictly in order

parts for this step: how many to place

a part with labels on its pins

labels with the same name are connected

chip ports: input [and output]

blue frame – building a chip

grey tab – the main canvas

yellow ring – what has changed

remove a part together with its labels

double-click a label – new name

chip built in the lesson with this number

check: this is how it should work

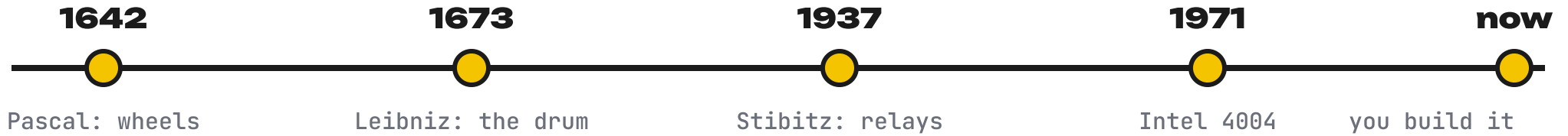
Keys

V	VCC (1)
G	GND (0)
N	transistor
I	switch
O	lamp
L	label
K	clock
U	button
[	port in
]	port out

Ctrl+G – into a chip
on a Mac Ctrl = Cmd



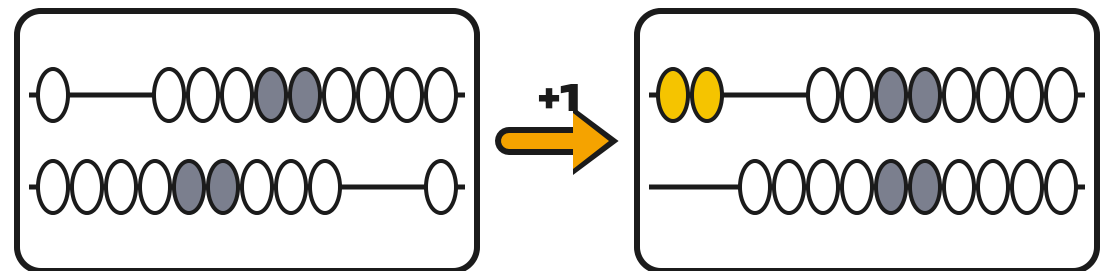
A little history



People added on the abacus too. The hard part is the carry: ten piled up – clear the digit and add one to the next. Pascal built a machine where a wheel going from 9 to 0 pushed the next one by itself. Leibniz's idea grew into arithmometers – in every office until the 1970s.

In 1937 George Stibitz built a binary adder on his kitchen table from two relays, a battery and light bulbs: lit is 1, dark is 0. Exactly what you build in the first lesson.

Relays gave way to vacuum tubes, tubes to transistors, but the adder stayed in every processor: it adds, subtracts, compares and computes addresses. The first microprocessor, the Intel 4004, was made for a calculator – so it could also do the decimal "+6" correction.

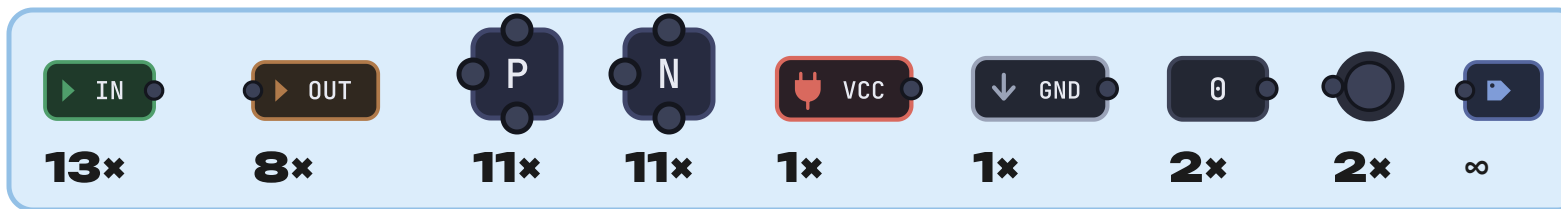
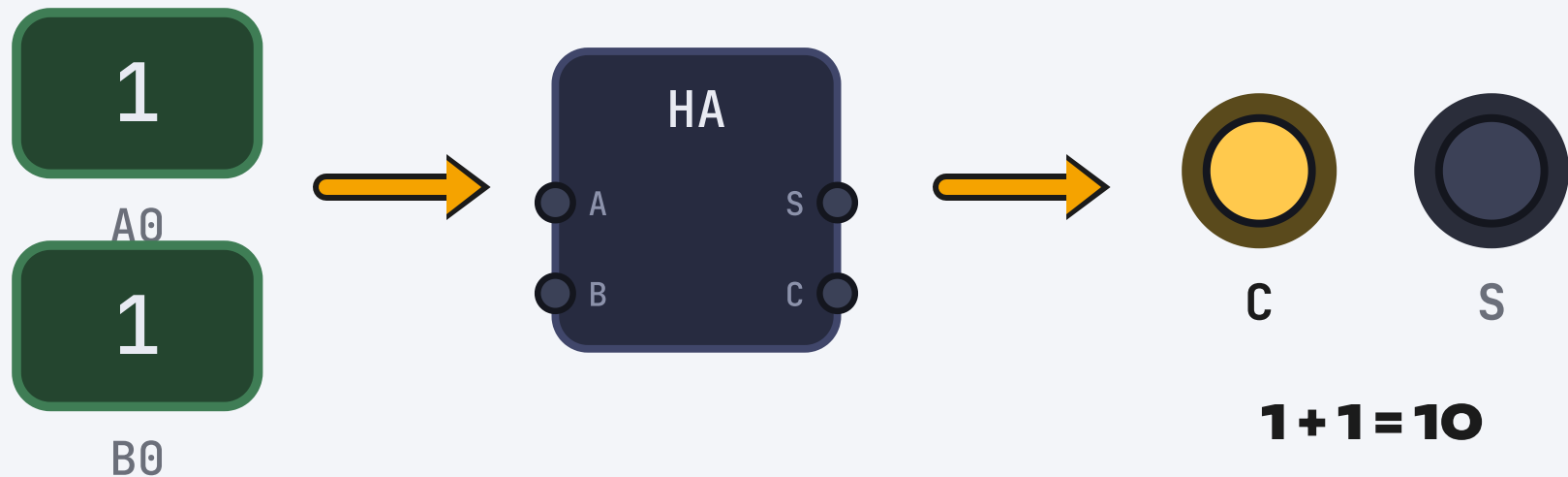


Intel and Bell Labs are trademarks of their respective owners. This kit is not affiliated with or endorsed by them.



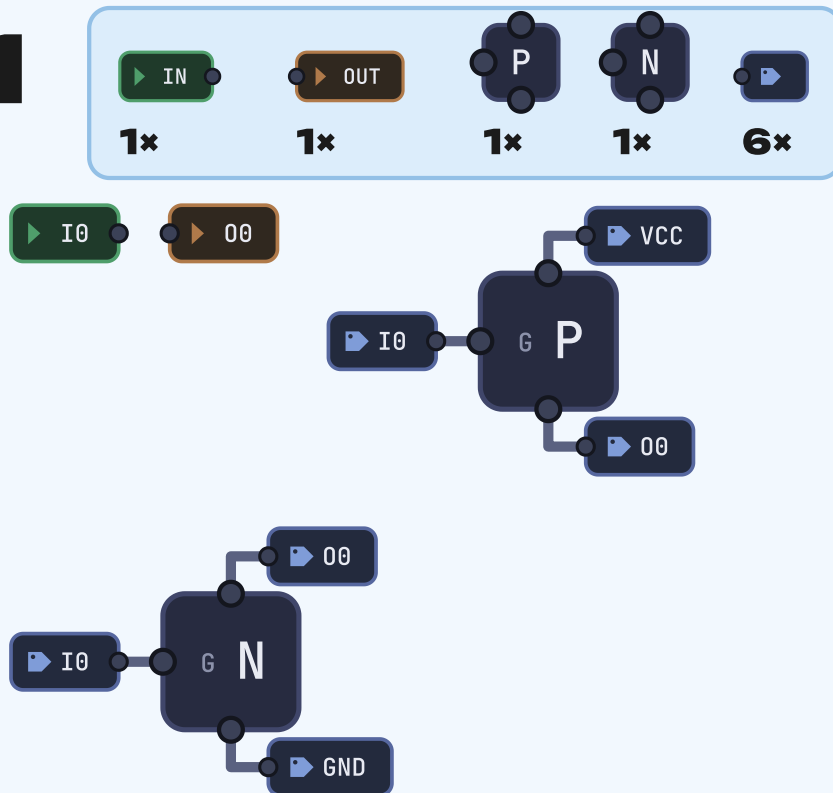
Half adder

gates from transistors, then $1 + 1 = 10$

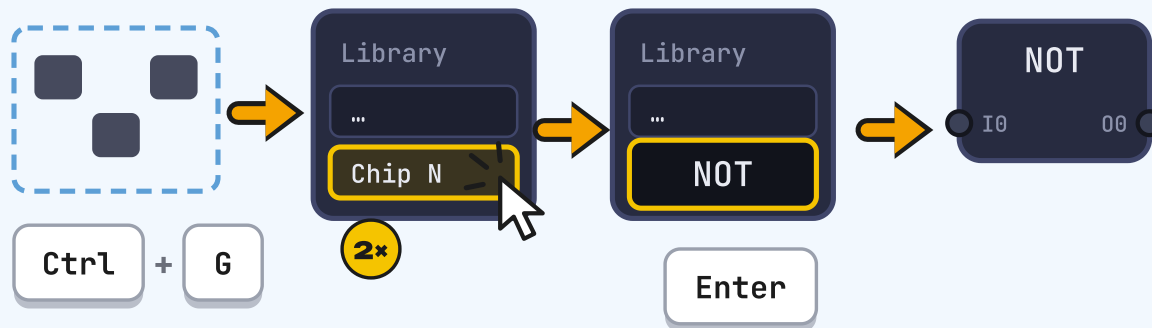


NOT

1

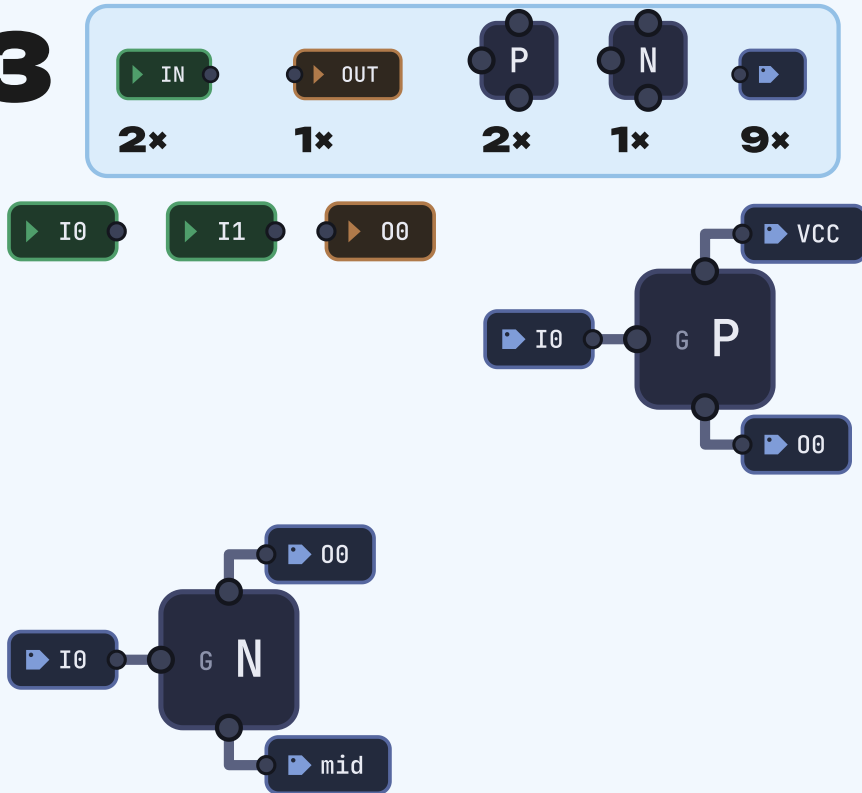


2

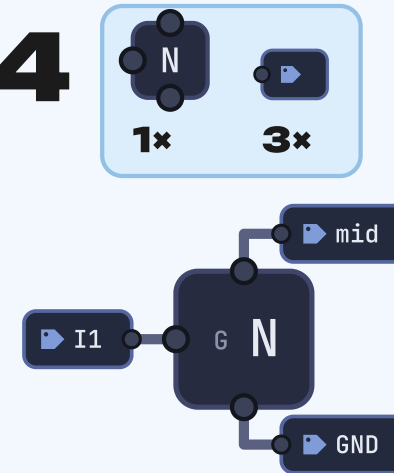


NAND

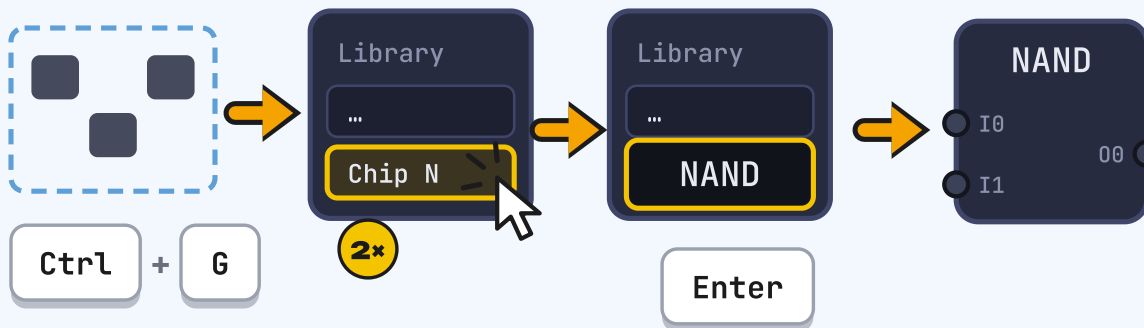
3



4

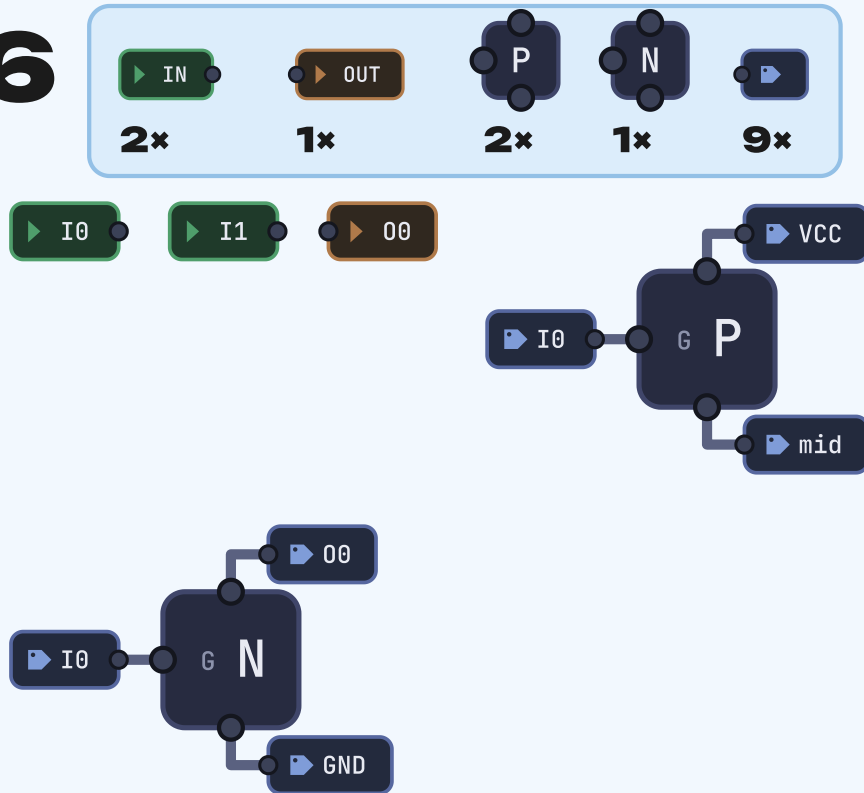


5

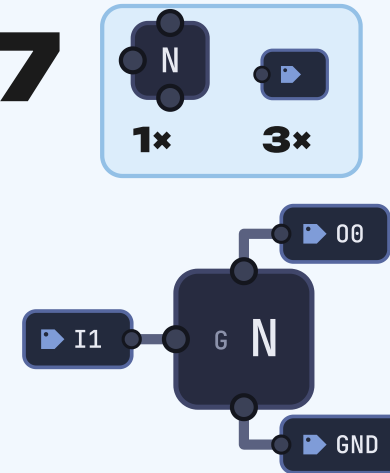


NOR

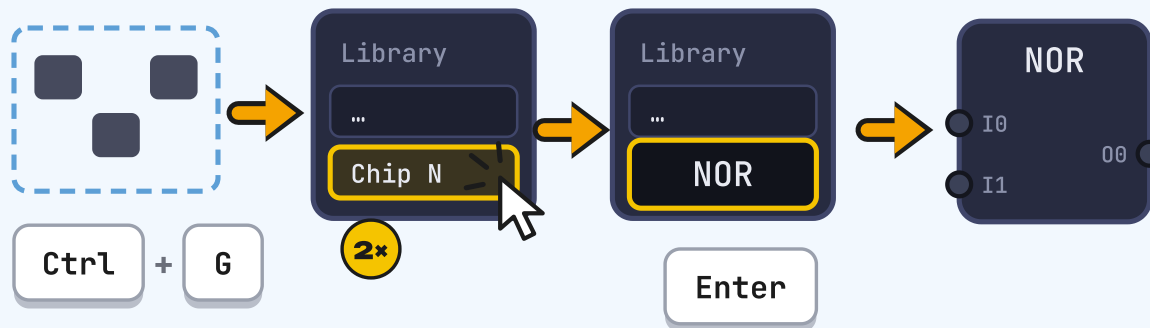
6



7

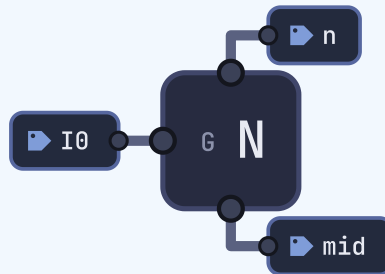
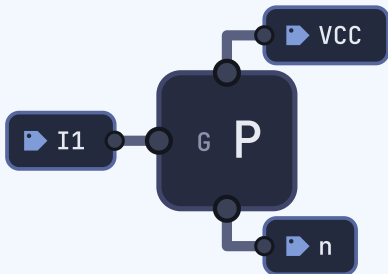
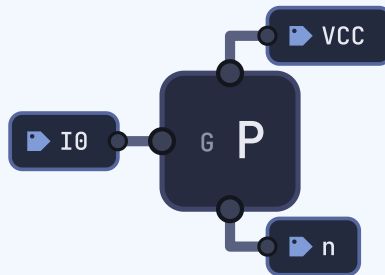


8

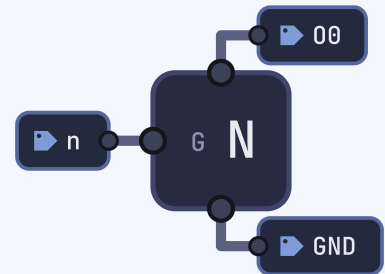
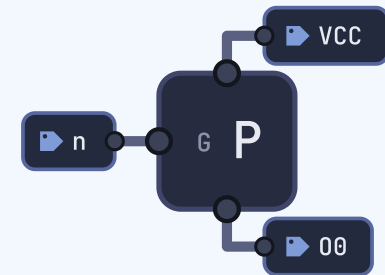
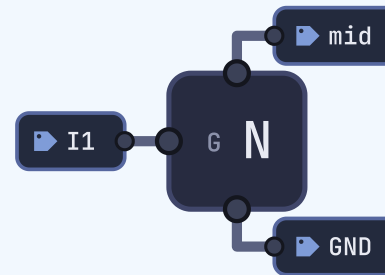
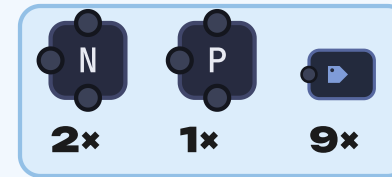


AND

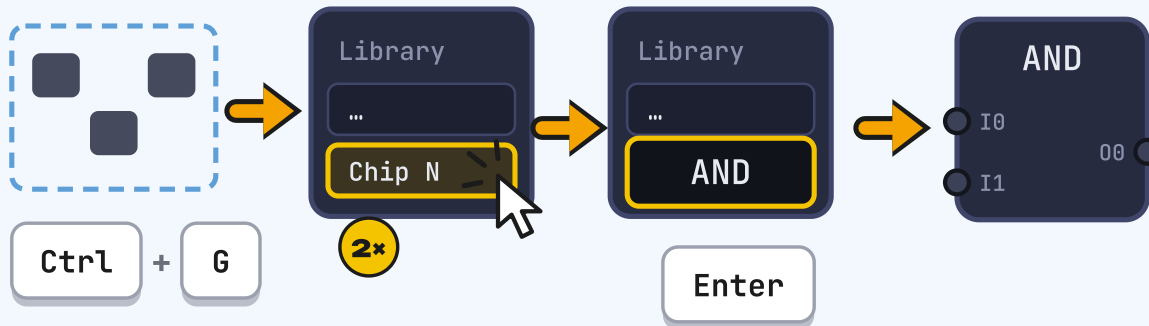
9



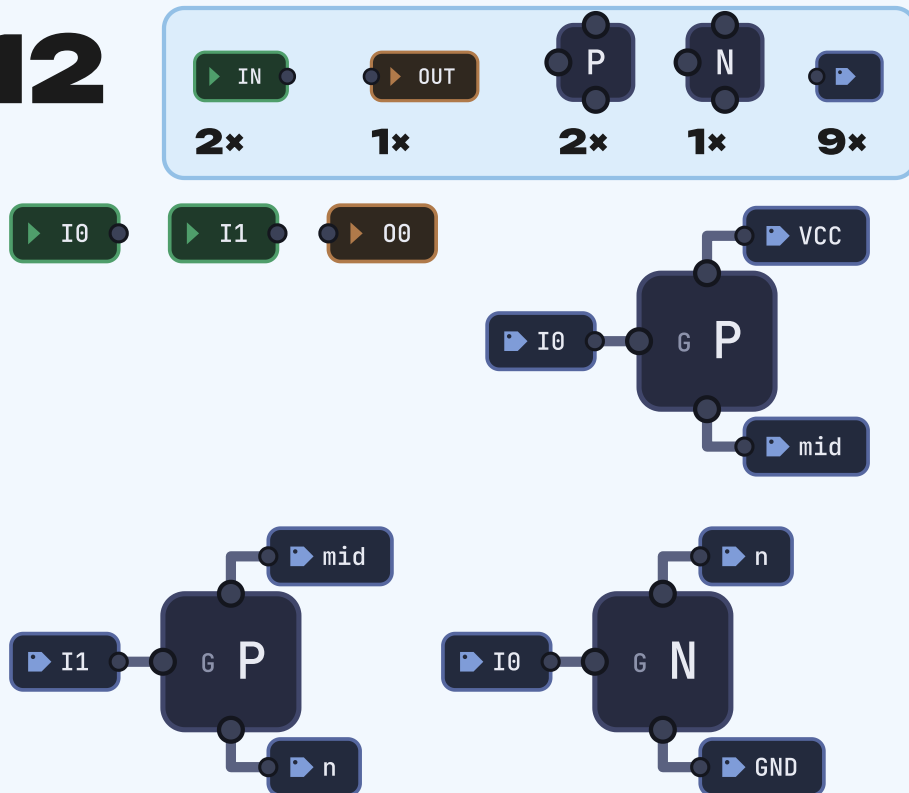
10



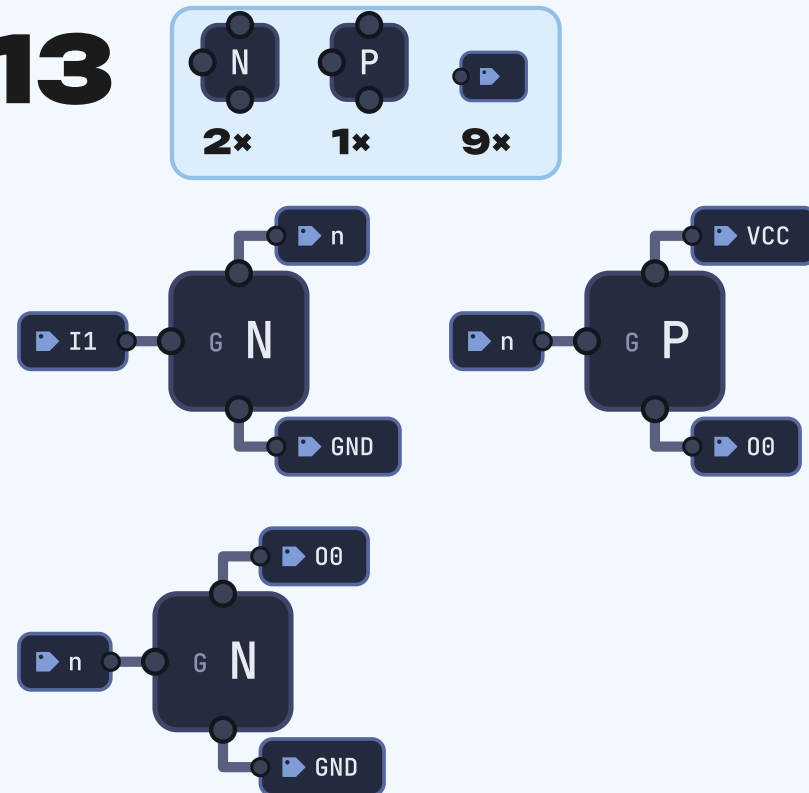
11



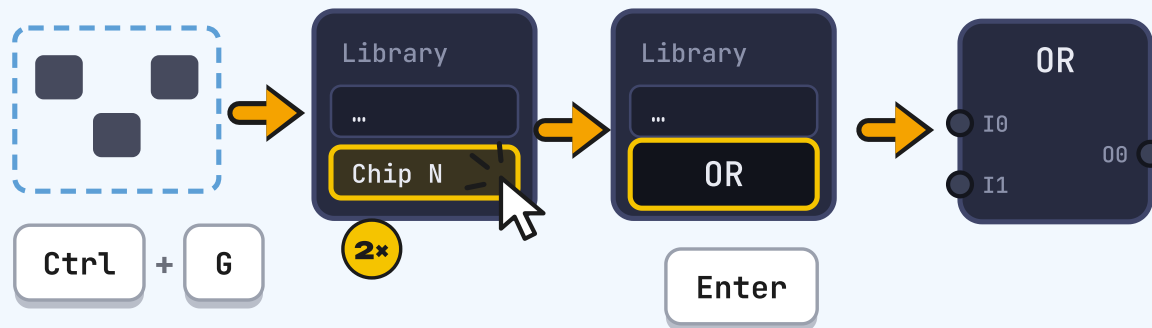
12



13

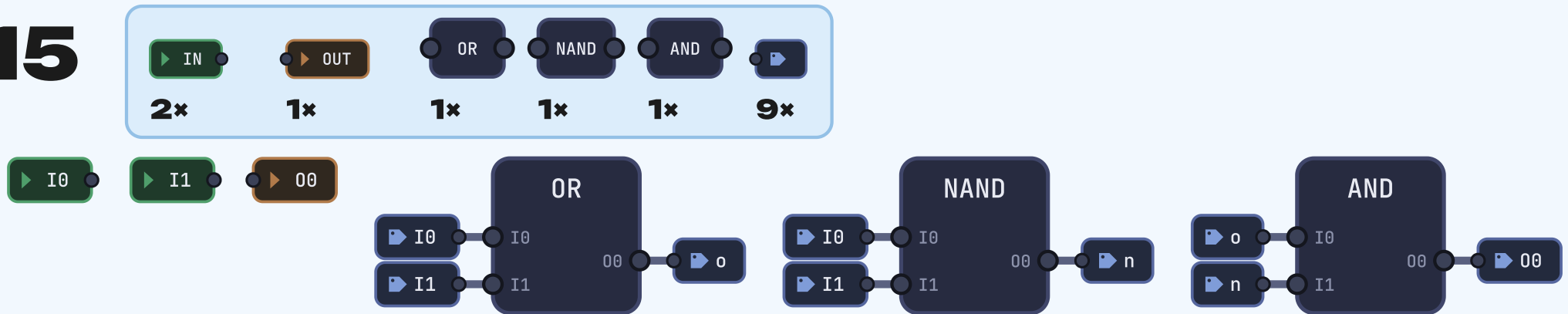


14

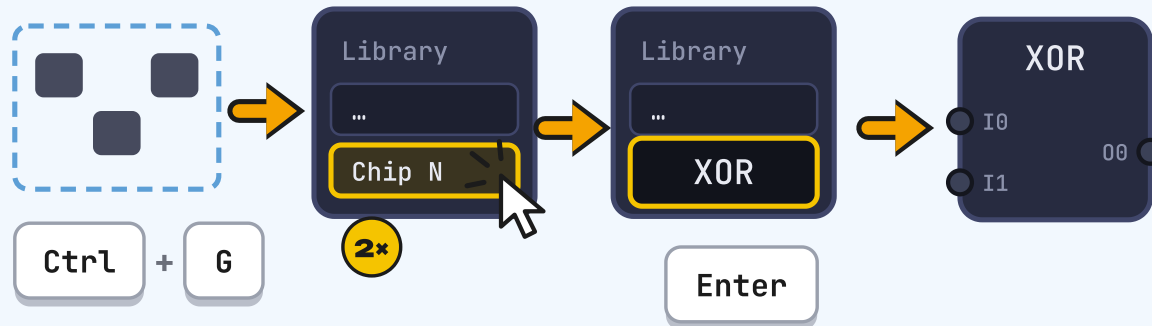


XOR

15

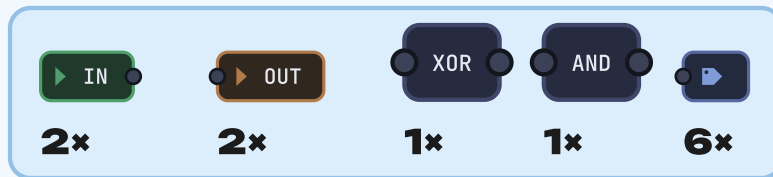


16

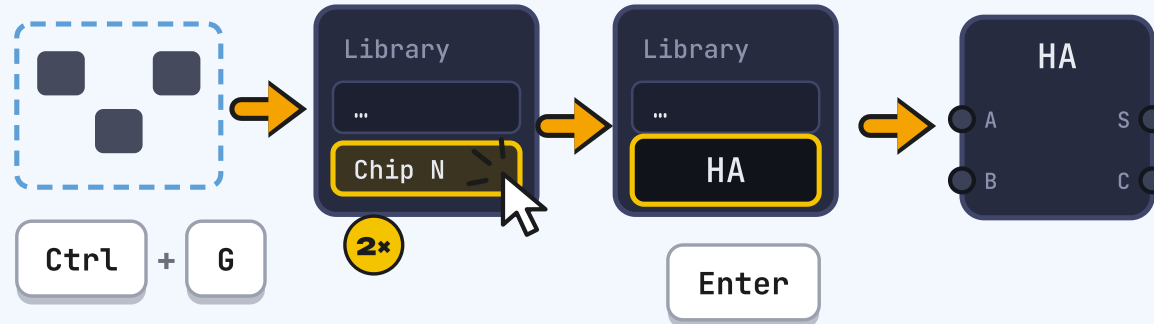


HA

17

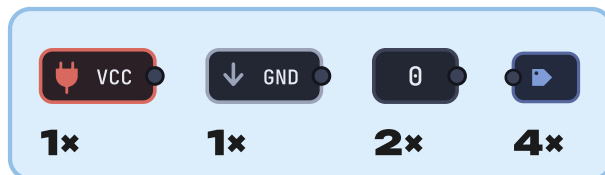


18

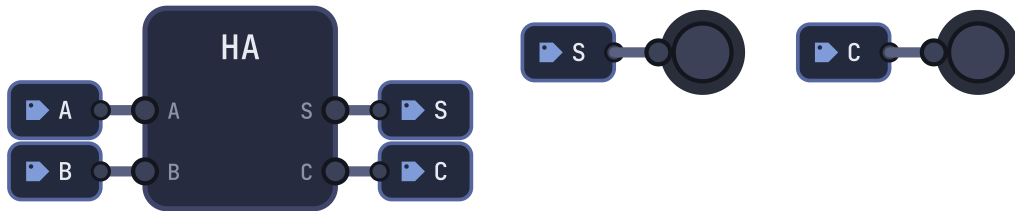


CANVAS

19



20



A0



B0



C



S

$$0 + 1 = 01$$



A0



B0



C



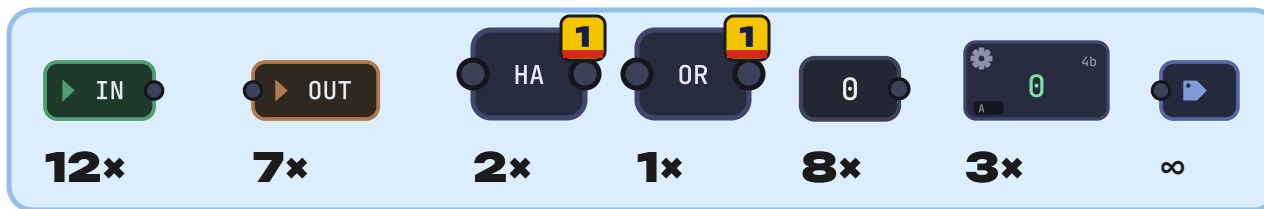
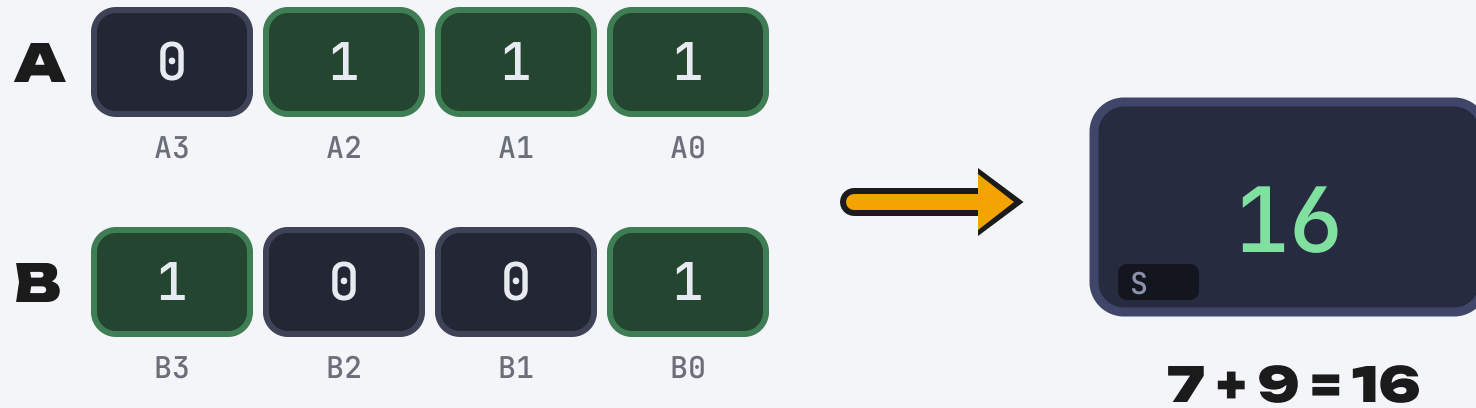
S

$$1 + 1 = 10$$



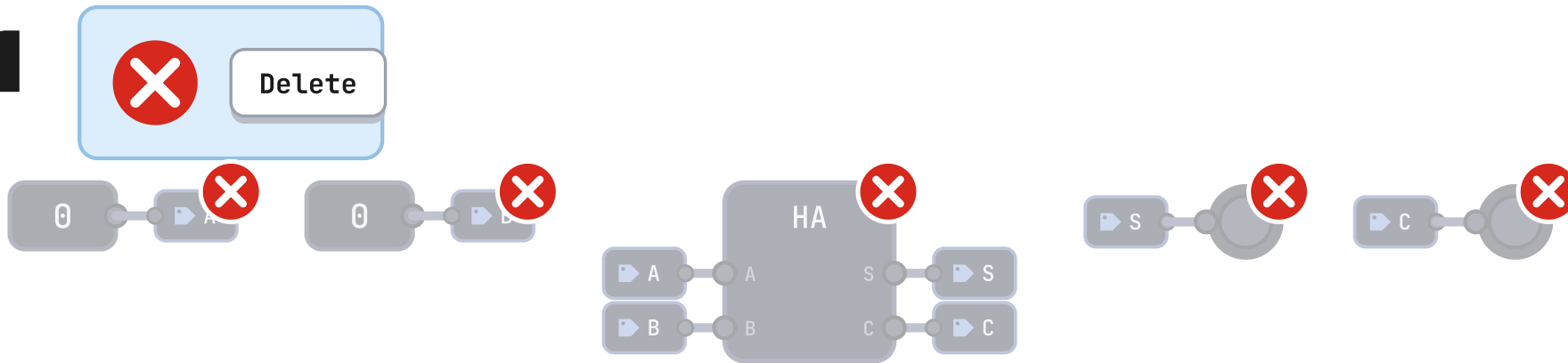
4 bits

the full adder FA and four digits in a chain



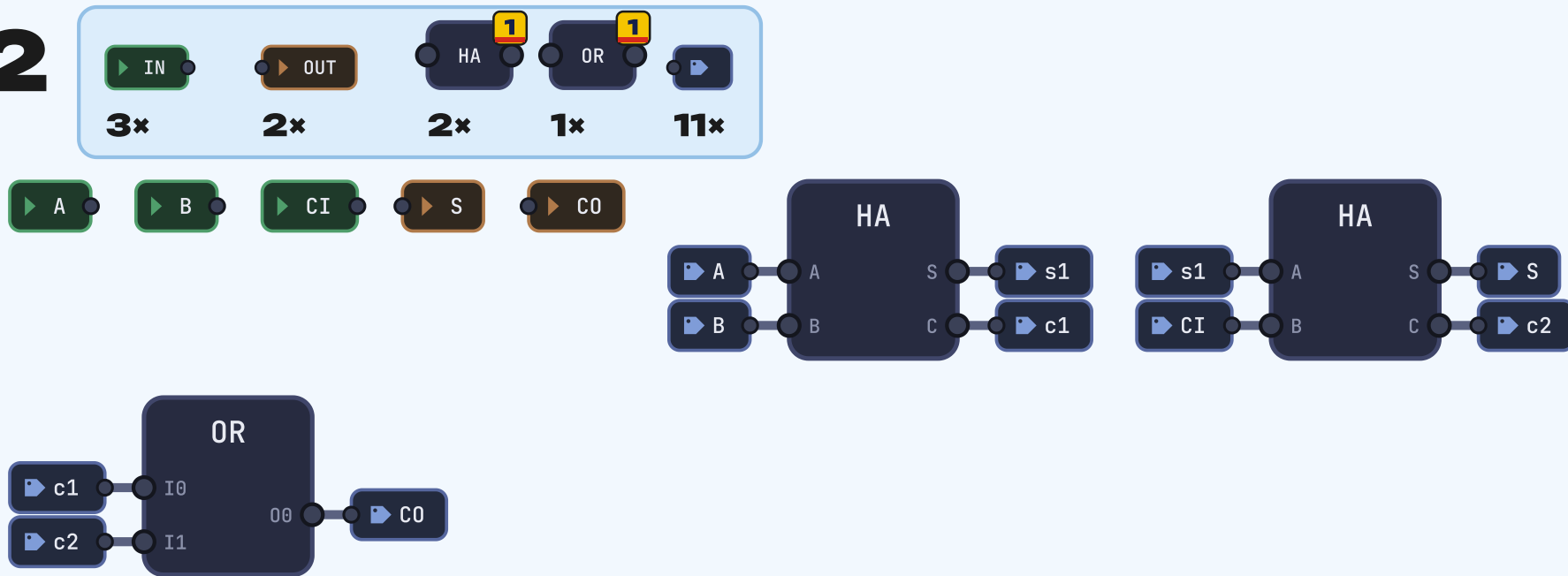
CANVAS

1



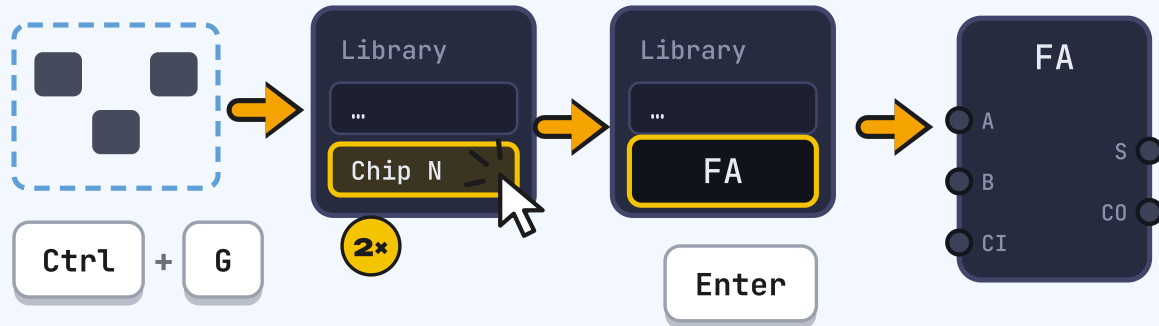
FA

2



FA...

3

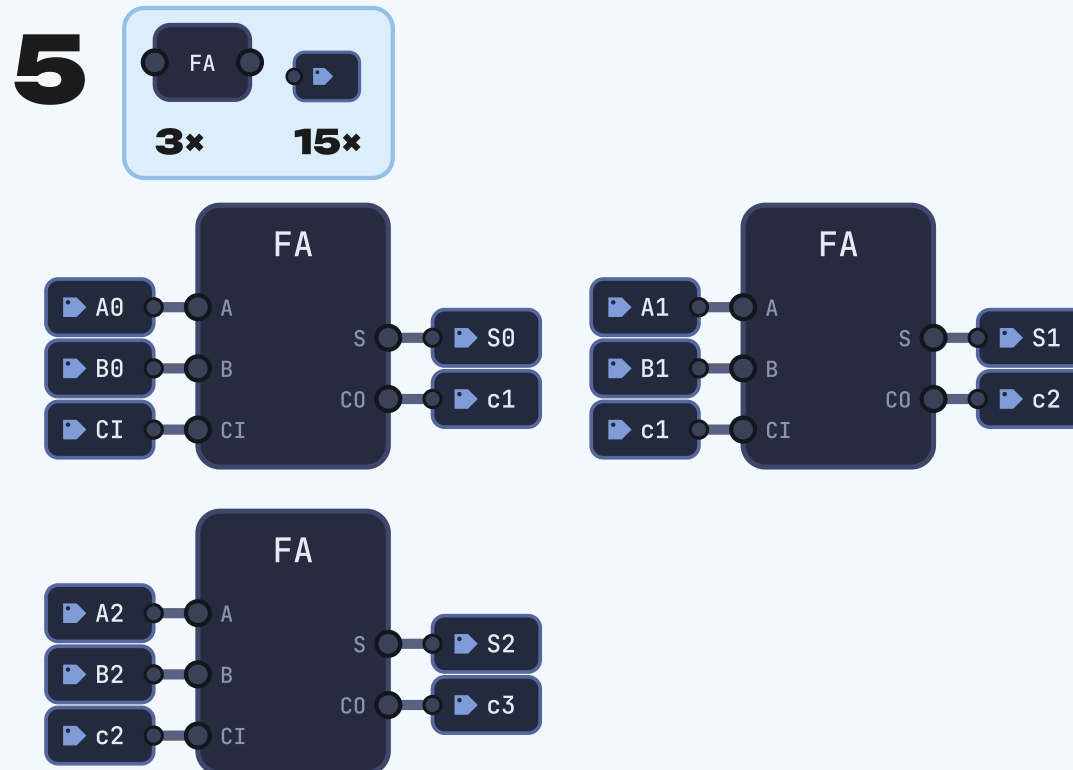


ADDER4

4

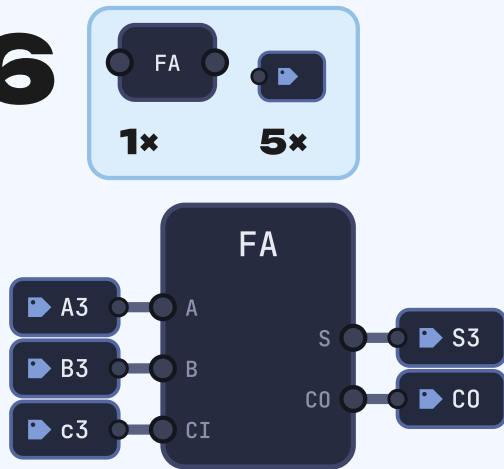


5

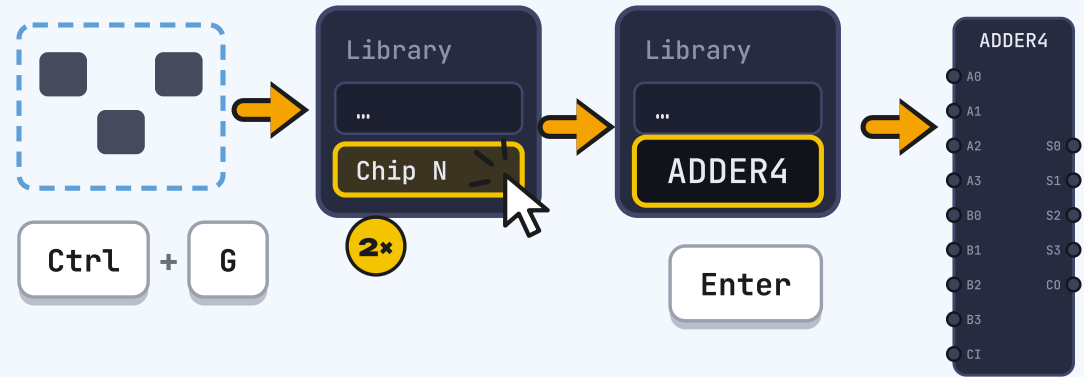


ADDER4 ...

6

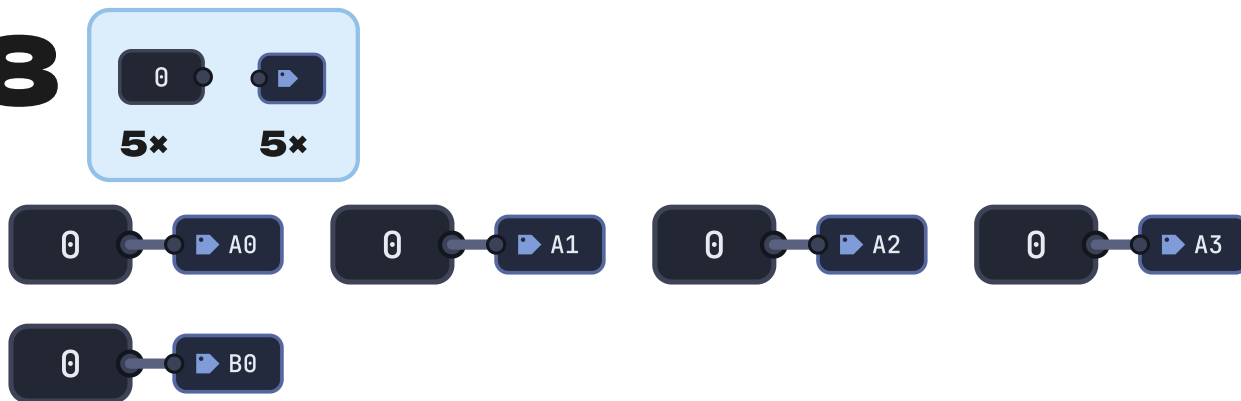


7



CANVAS

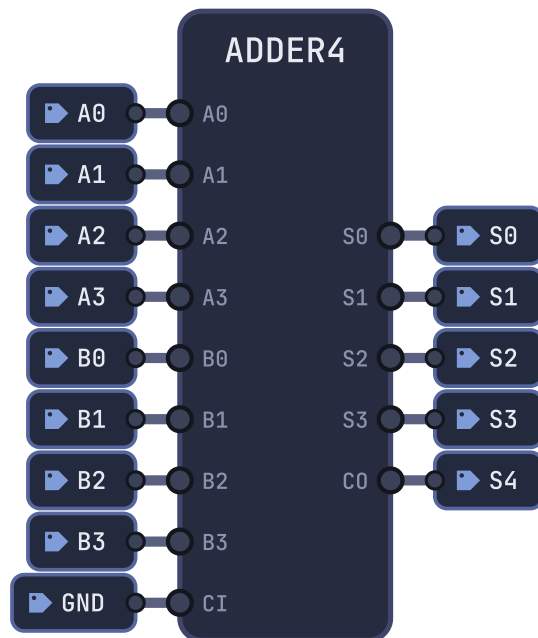
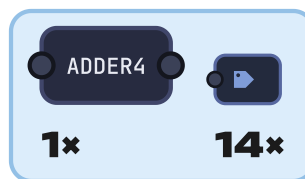
8



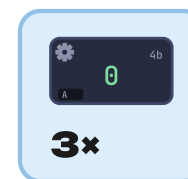
9



10



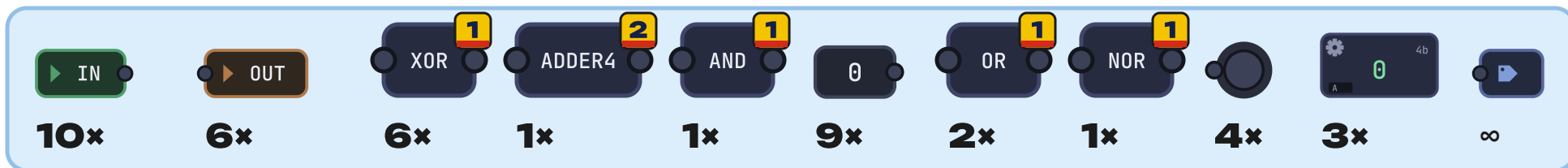
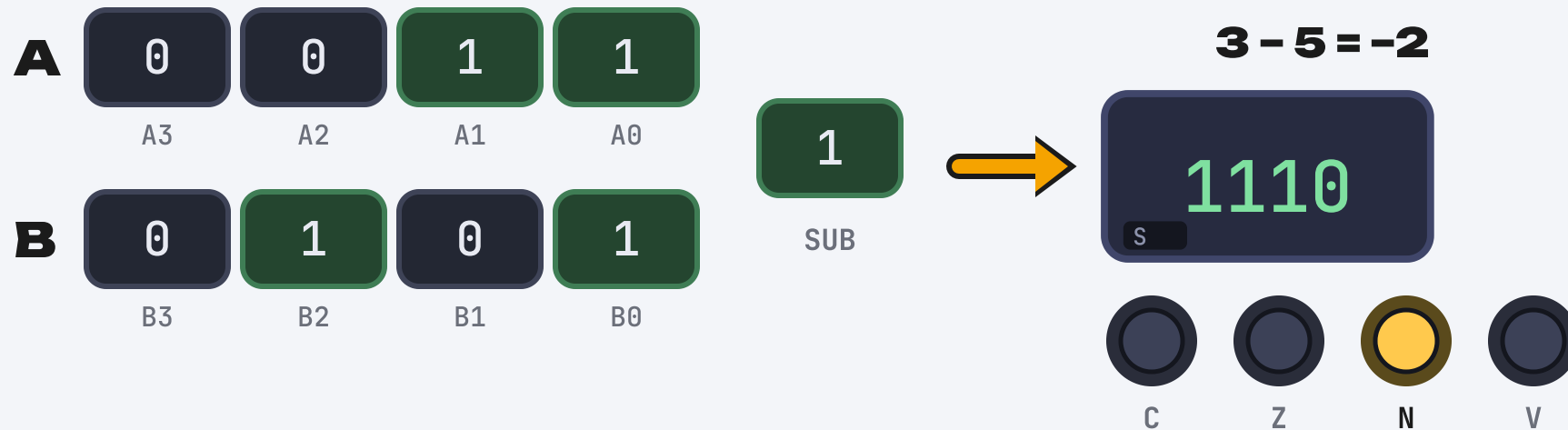
11



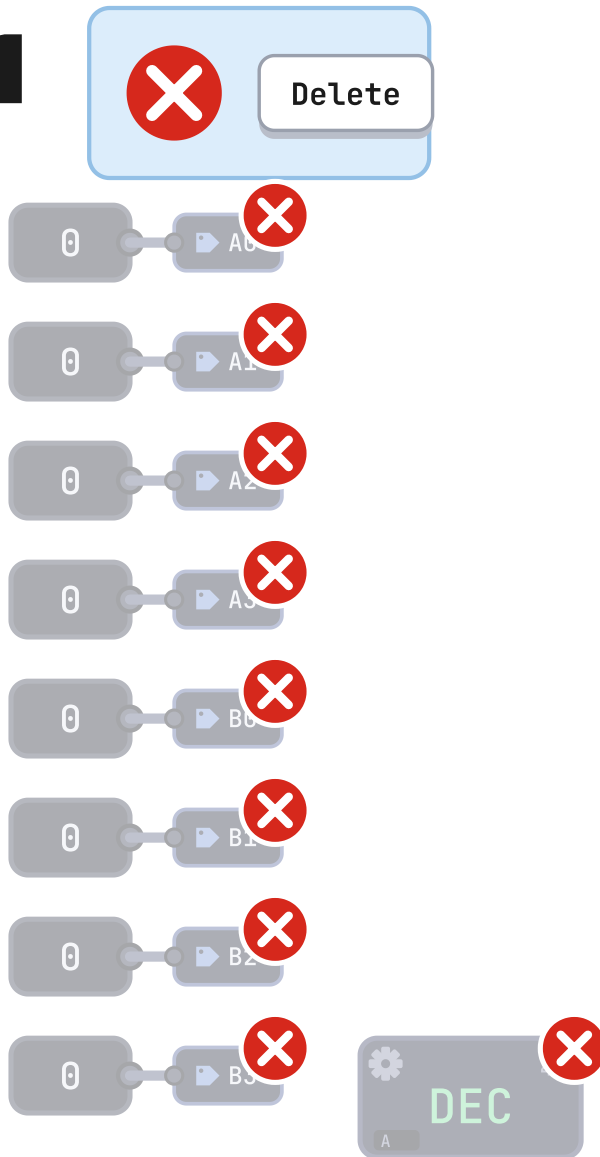


Subtraction and flags

two's complement, the SUB switch, lamps C Z N V



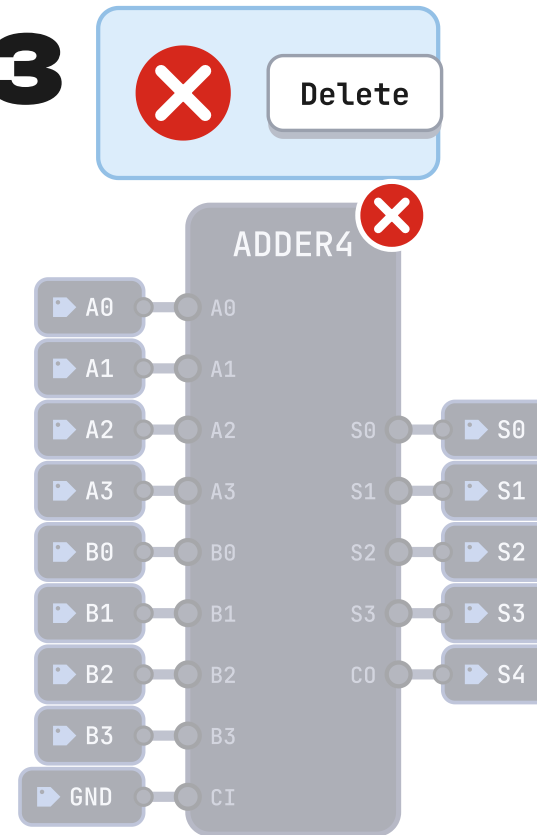
1



2

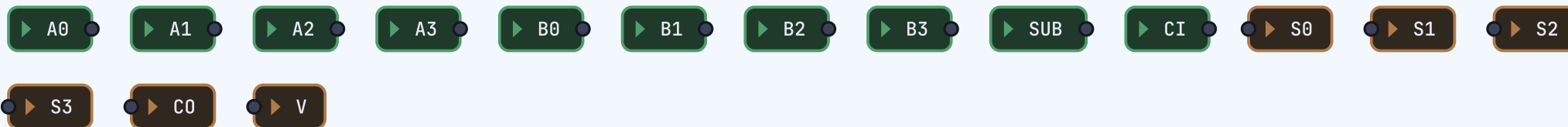
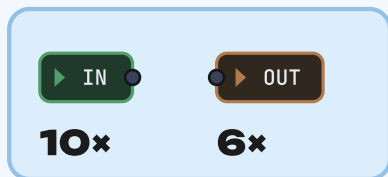


3

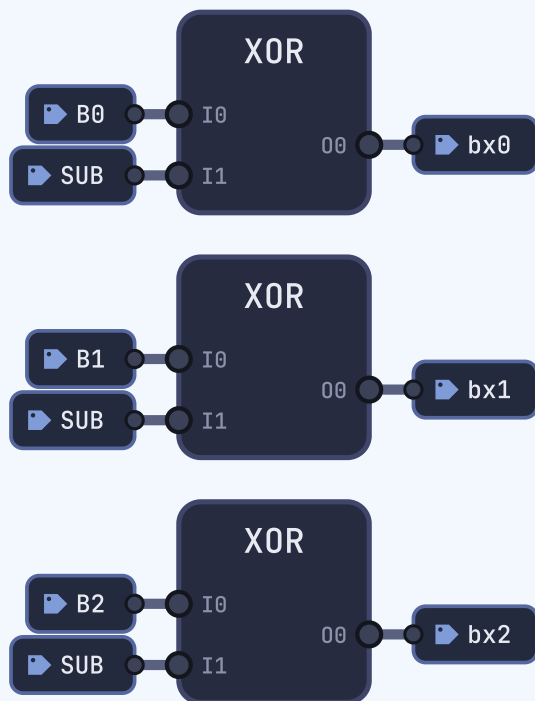
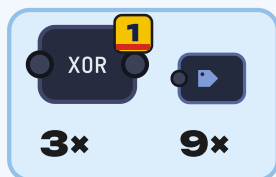


ADDSUB4

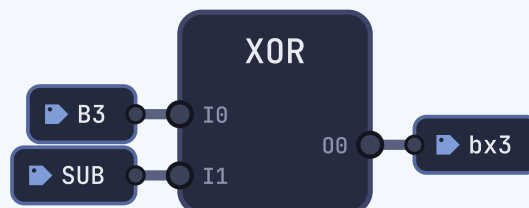
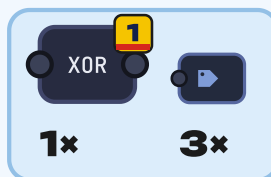
4



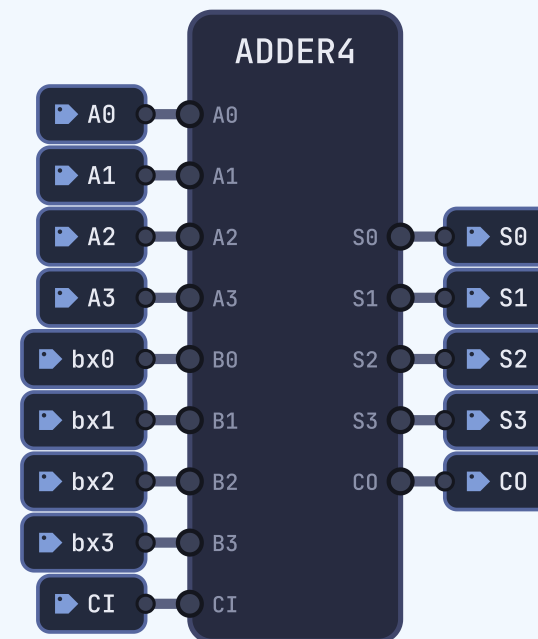
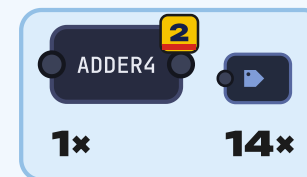
5



6

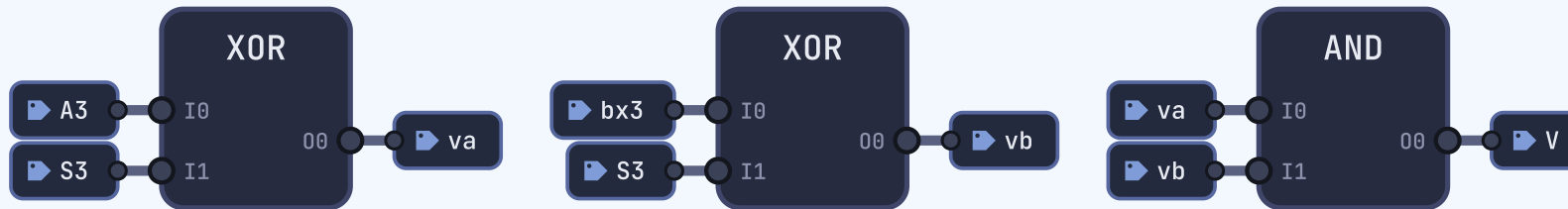
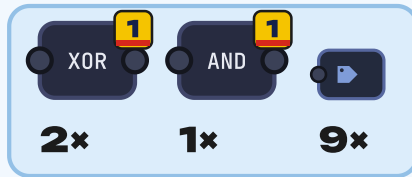


7

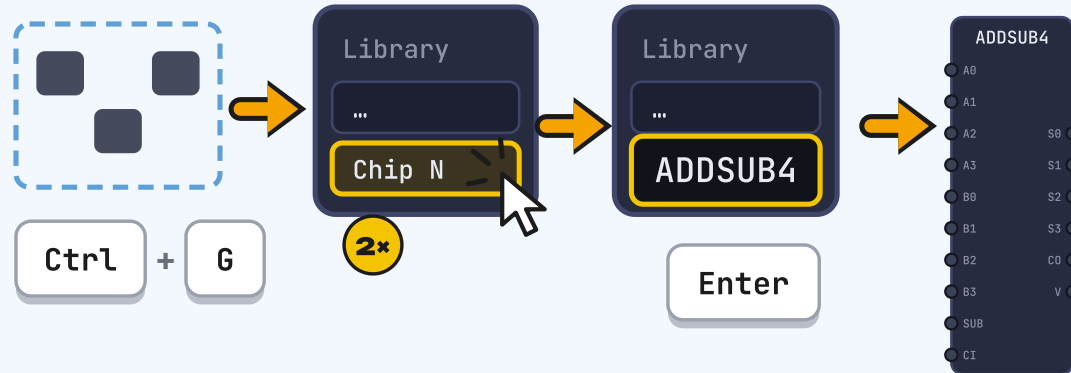


ADDSUB4 ...

8



9



CANVAS

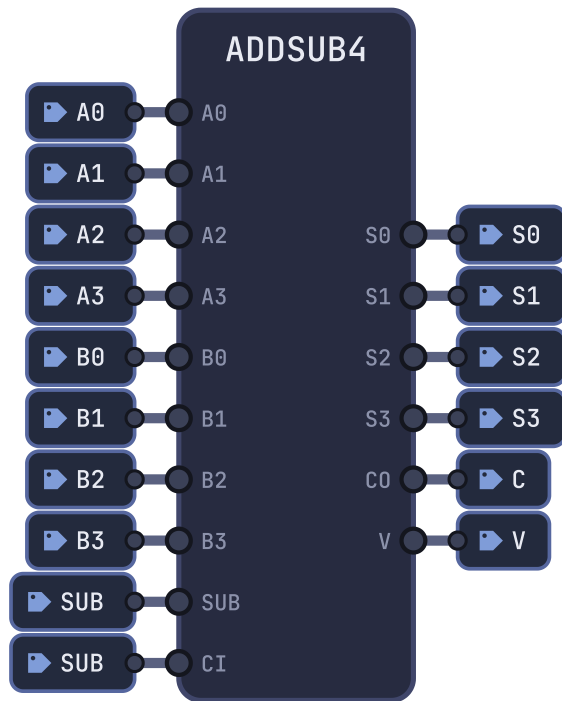
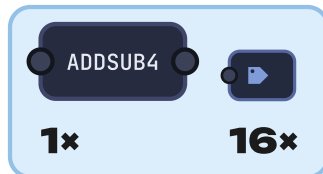
10



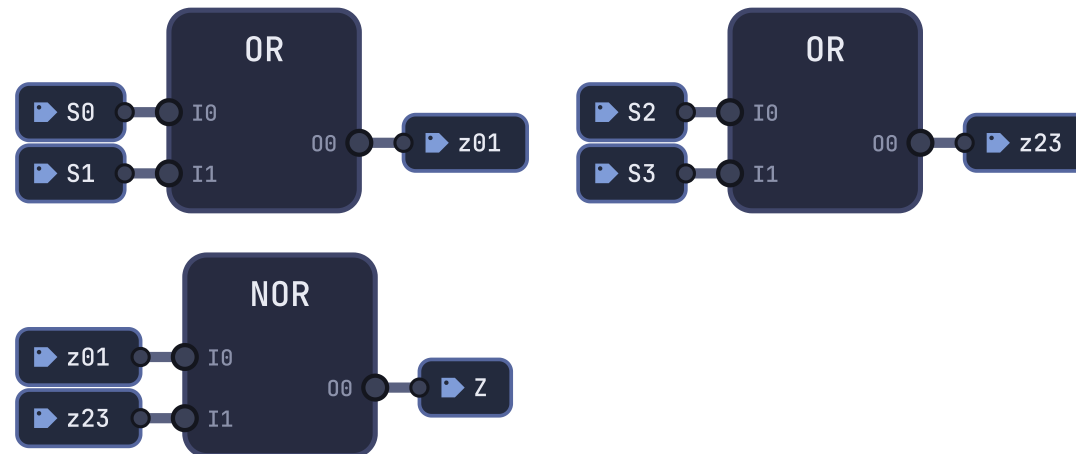
11



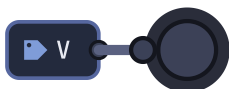
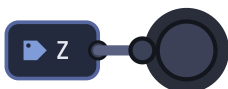
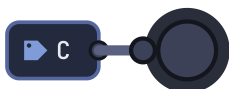
12



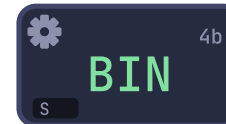
13



14



15



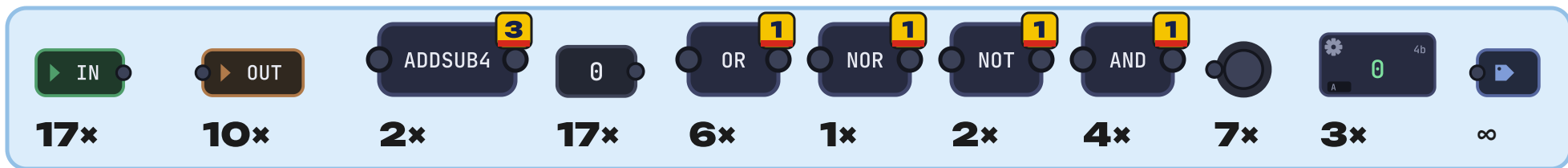


8 bits and comparison

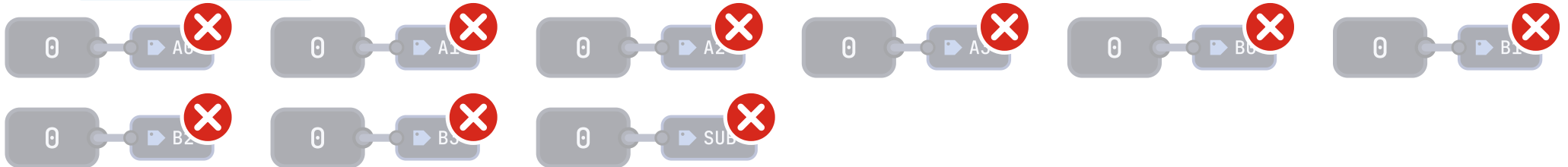
two ADDSUB4 in a chain, lamps = < >



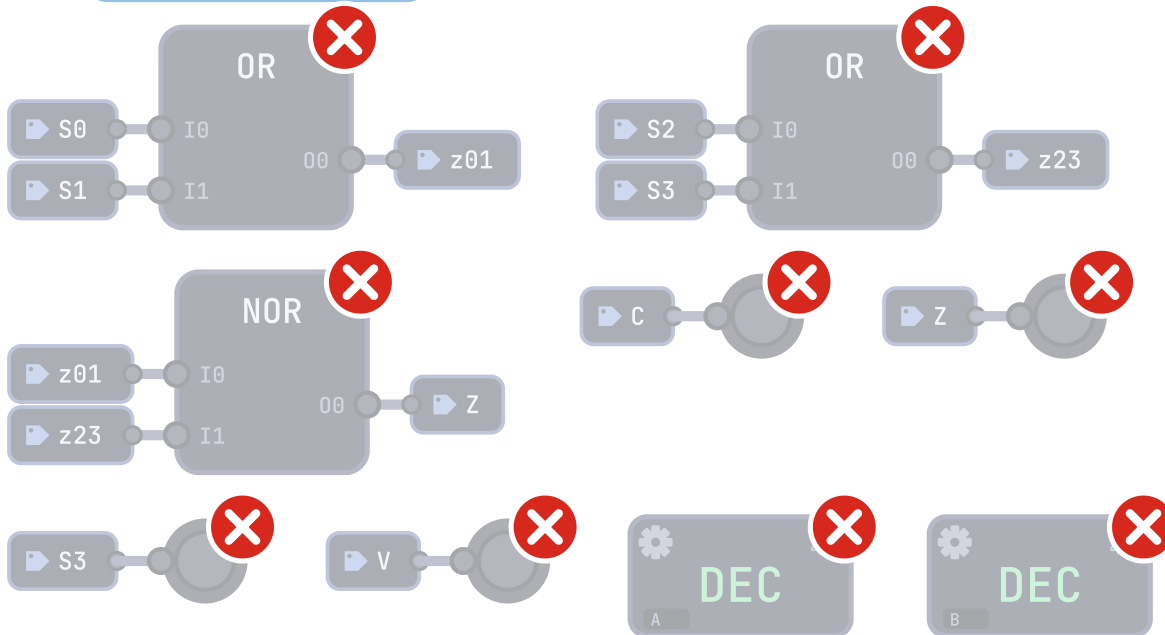
5 - 9: A < B



1



2



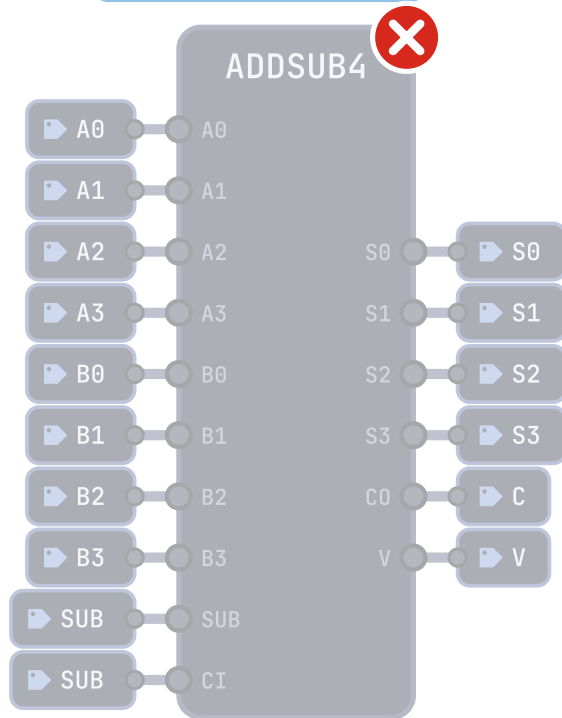
3



4



Delete



ADDSUB8

5

IN

17x

OUT

10x

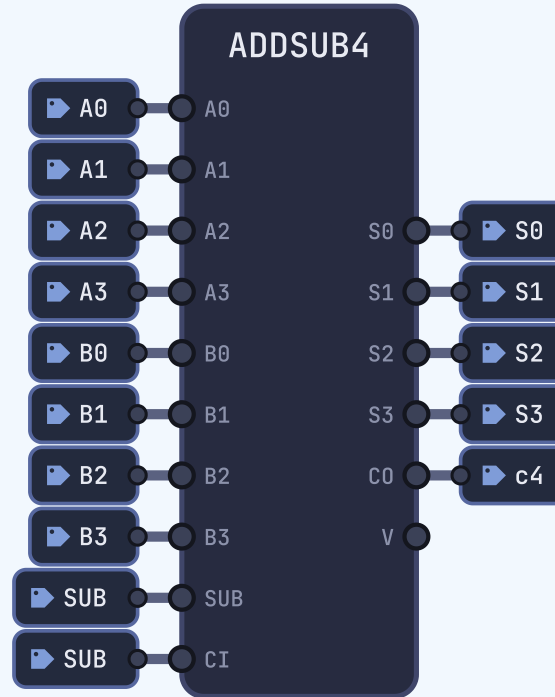


6

ADDSUB4

1x

15x

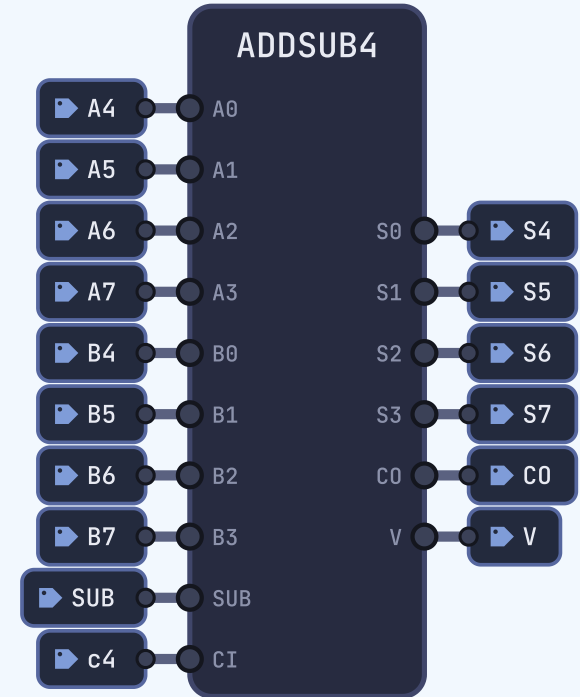


7

ADDSUB4

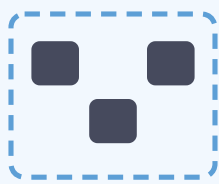
1x

16x



ADDSUB8 ...

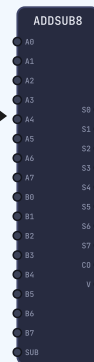
8



Ctrl + G

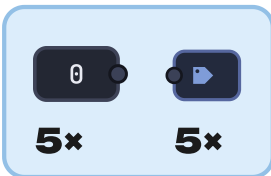


Enter

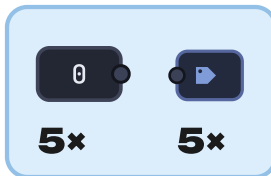


CANVAS

9



10



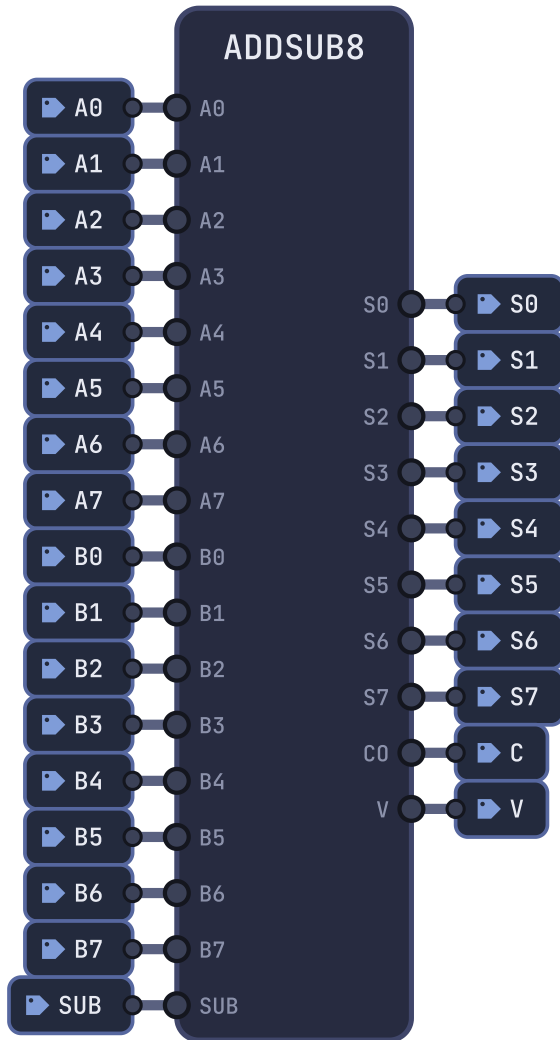
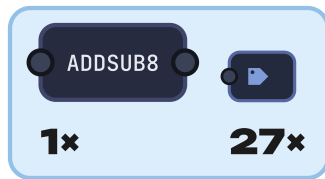
11



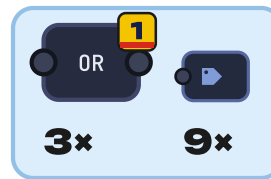
12



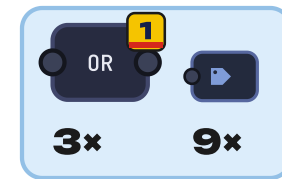
13



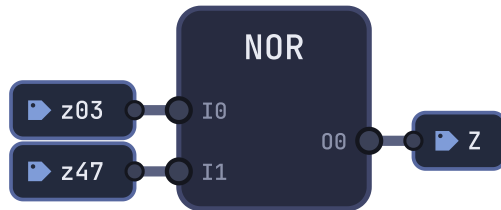
14



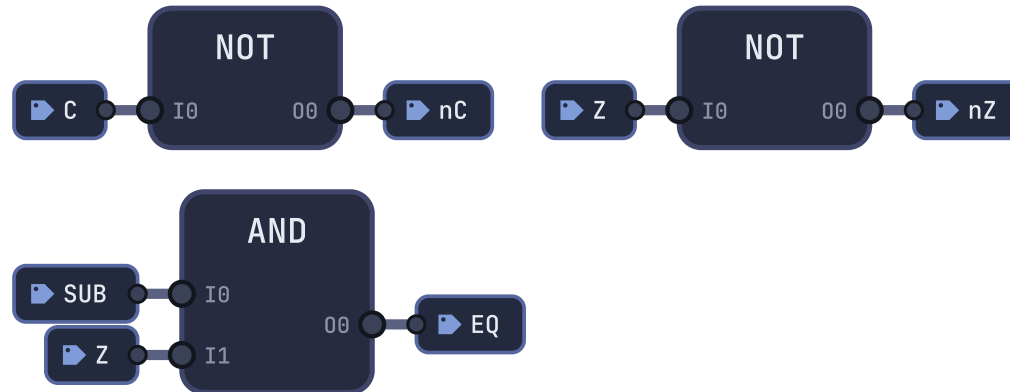
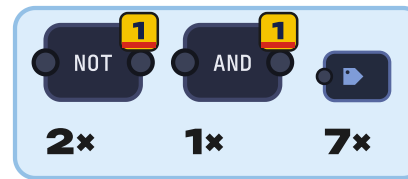
15



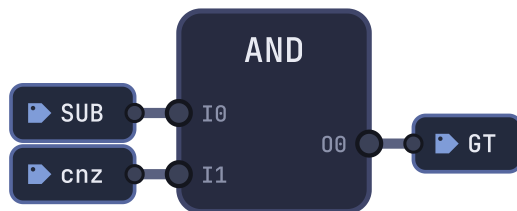
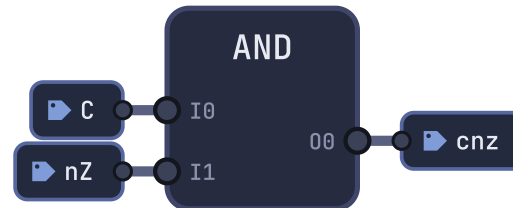
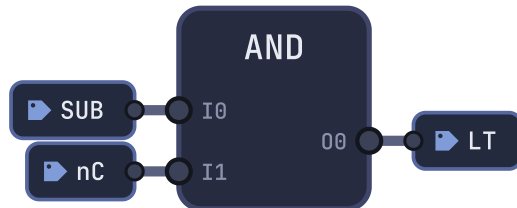
16



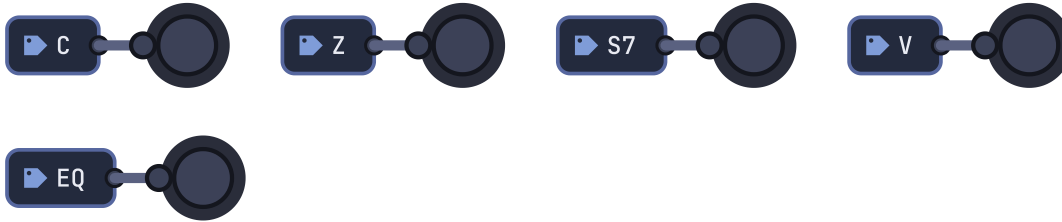
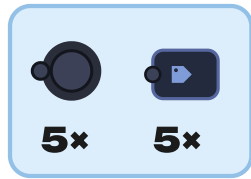
17



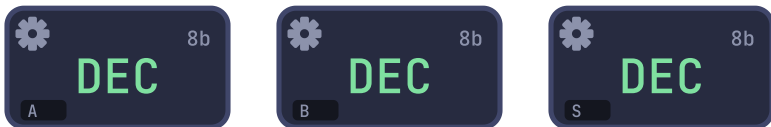
18



19



21



A

0	0	0	0	0	1	1	1
A7	A6	A5	A4	A3	A2	A1	A0

B

0	0	0	0	0	1	1	1
B7	B6	B5	B4	B3	B2	B1	B0

1



S 0



SUB

A

0	0	0	0	0	1	0	1
A7	A6	A5	A4	A3	A2	A1	A0

B

0	0	0	0	1	0	0	1
B7	B6	B5	B4	B3	B2	B1	B0

1



S 252



SUB

A

1	1	0	0	1	0	0	0
A7	A6	A5	A4	A3	A2	A1	A0

B

1	0	0	1	0	1	1	0
B7	B6	B5	B4	B3	B2	B1	B0

1

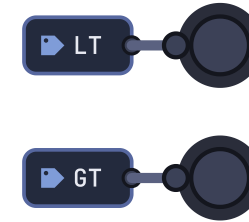
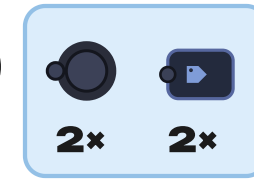


S 50



SUB

20



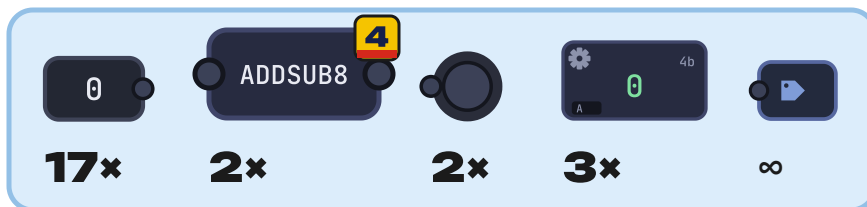


The $\pm$ calculator

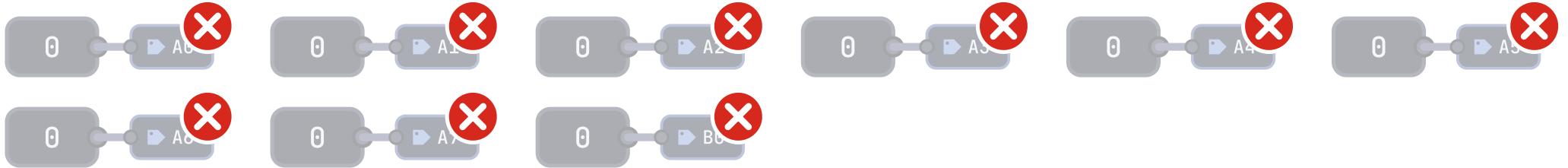
the minus on a lamp, the magnitude on the display



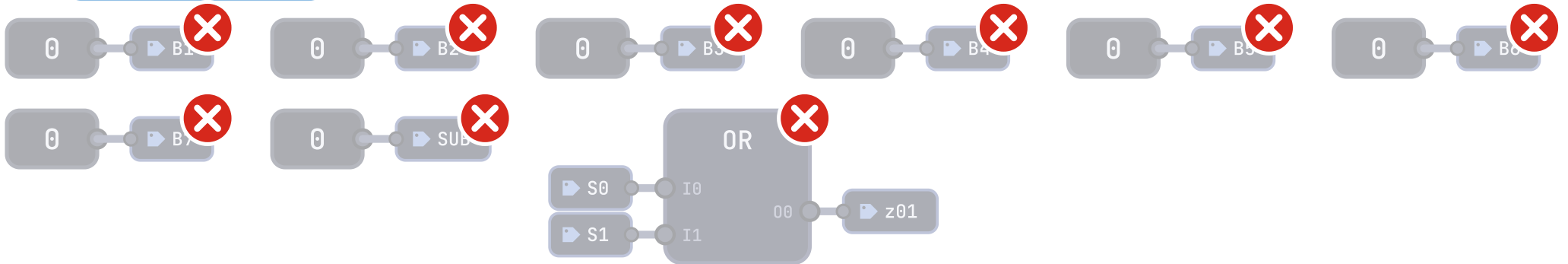
$$3 - 5 = -2$$



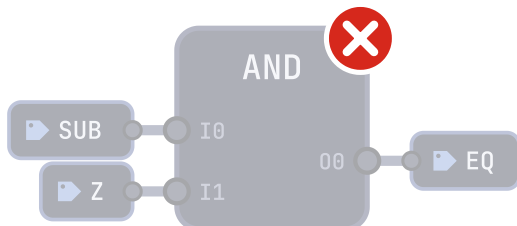
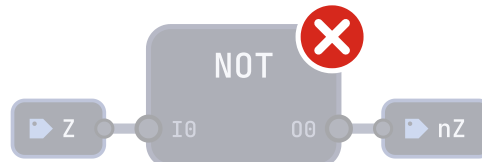
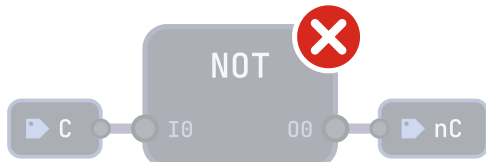
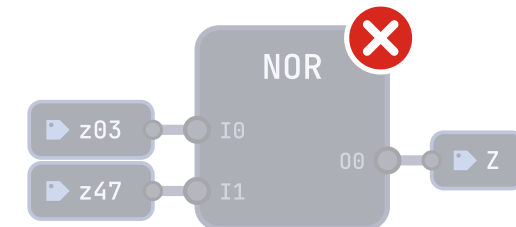
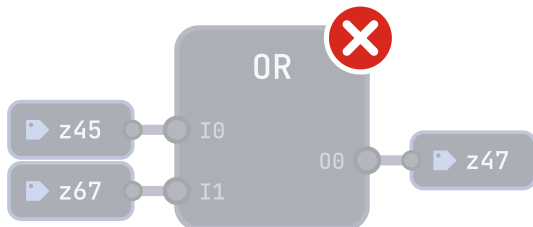
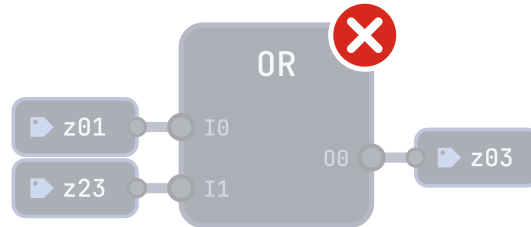
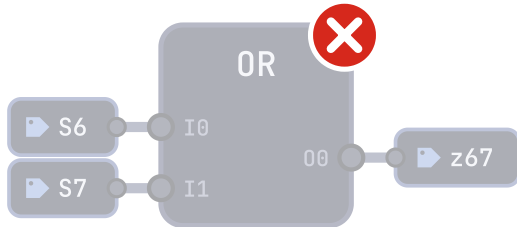
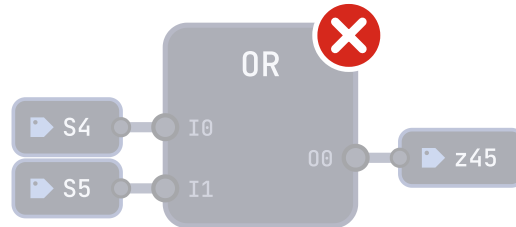
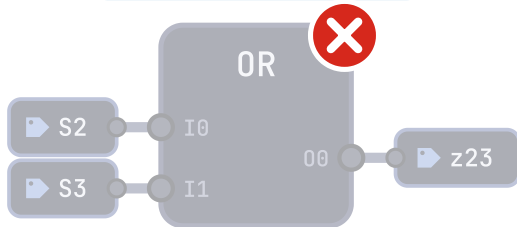
1



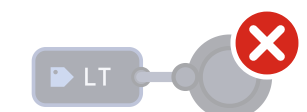
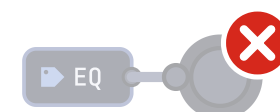
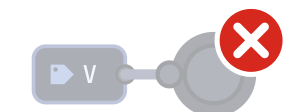
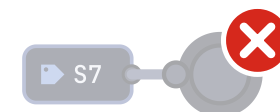
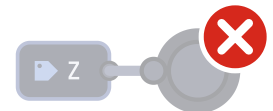
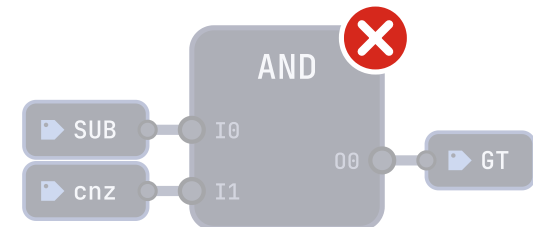
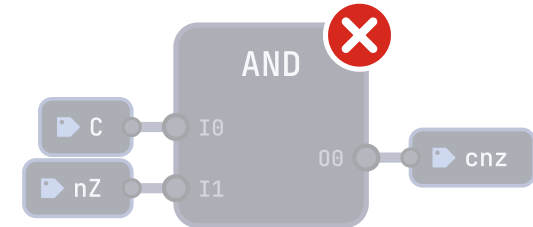
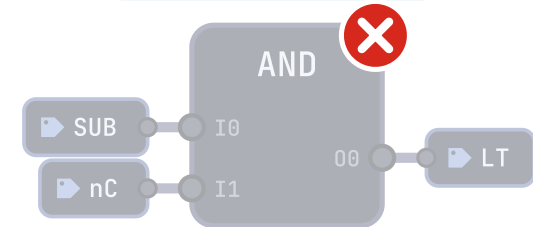
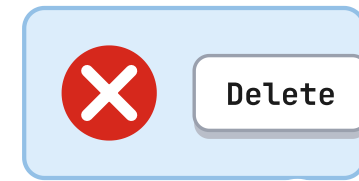
2



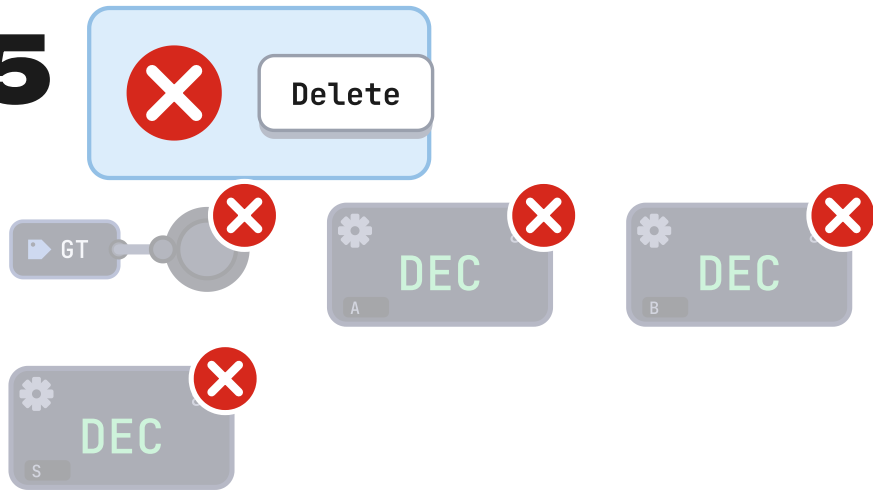
3



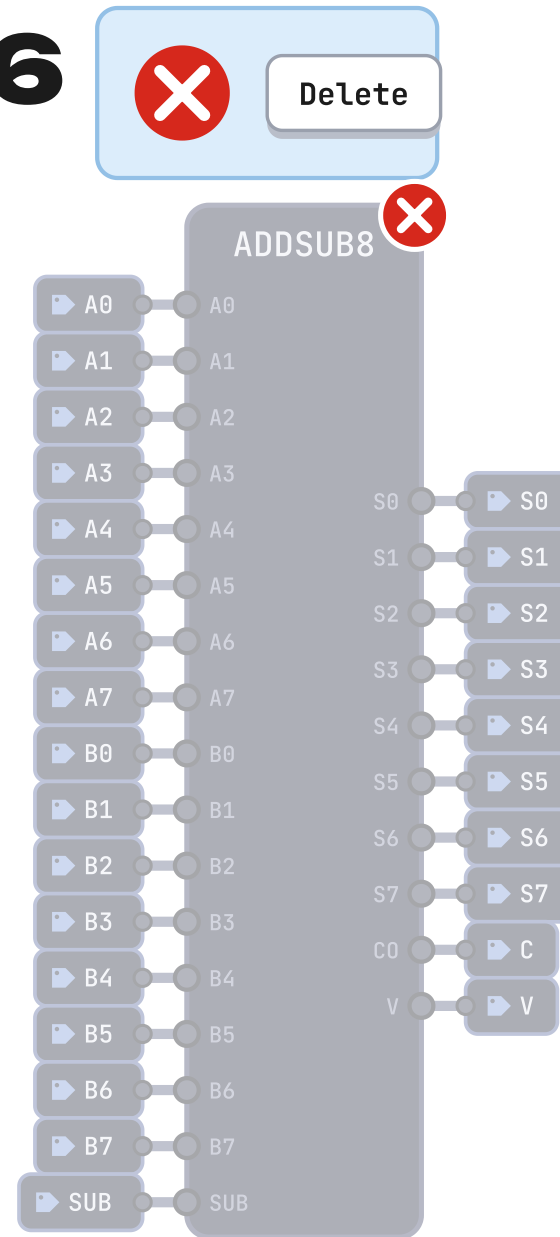
4



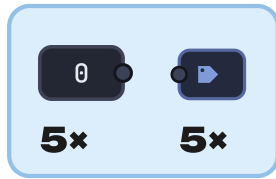
5



6



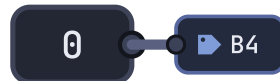
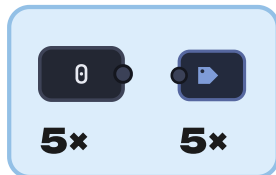
7



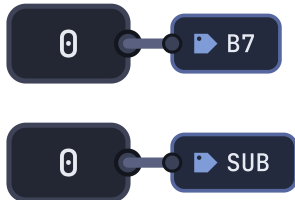
8



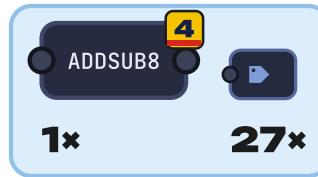
9



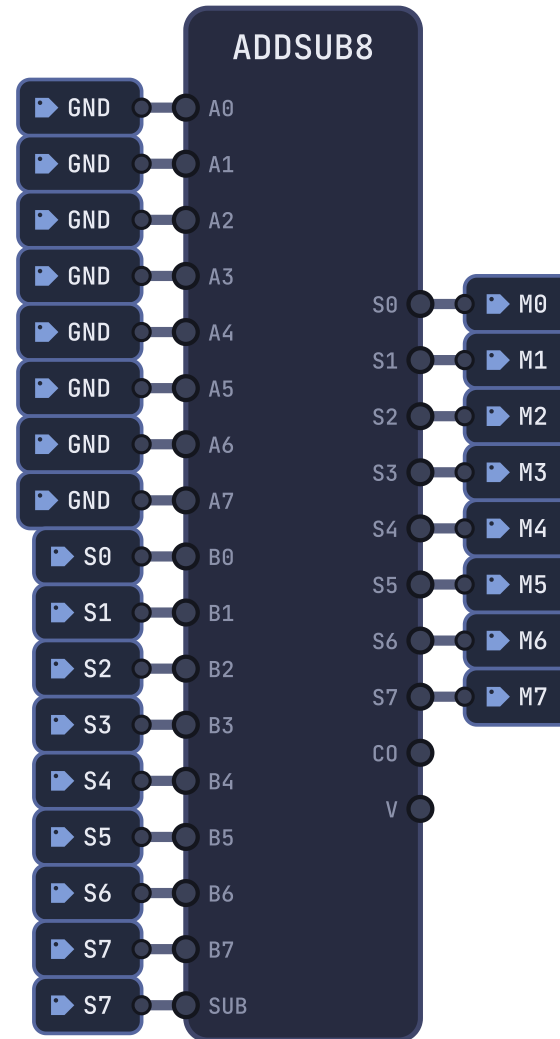
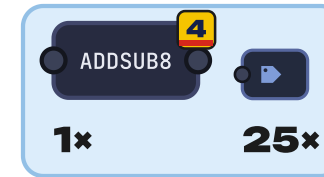
10



11



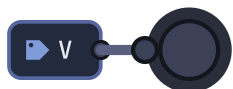
12



13



14



A



B



1



SUB

A



B



0



SUB

A



B



1

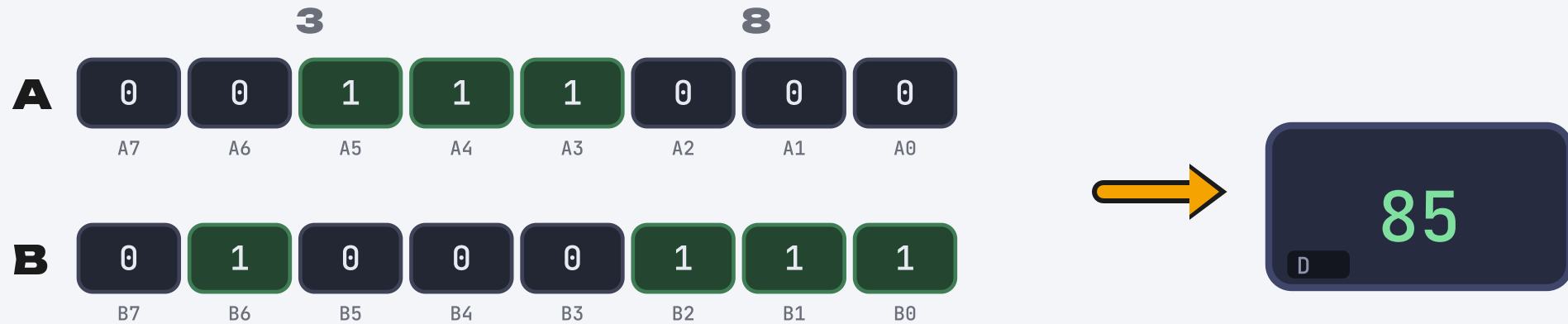


SUB

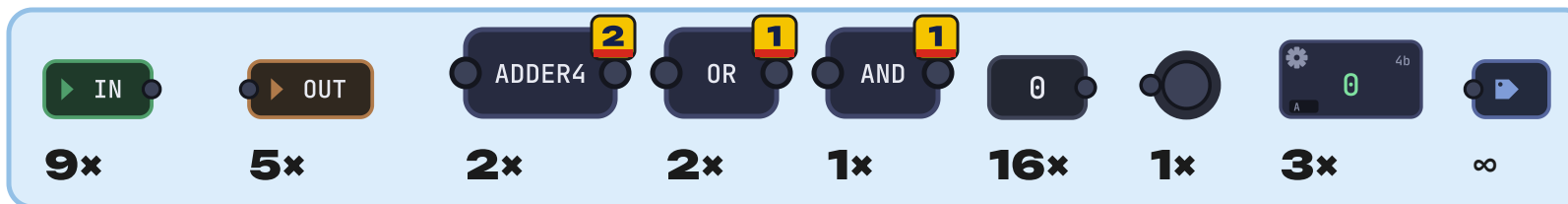


Bonus: decimal math

the +6 correction: $38 + 47 = 85$

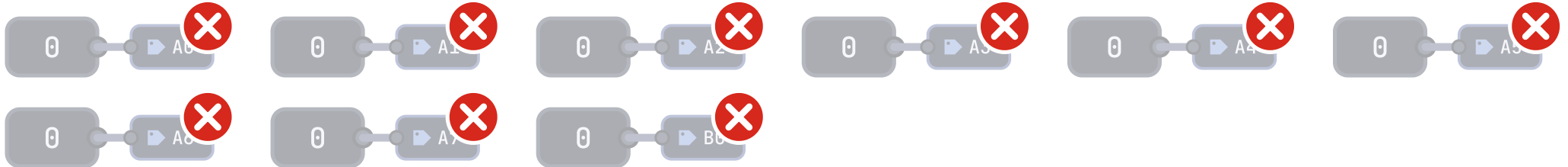


$$38 + 47 = 85$$



CANVAS

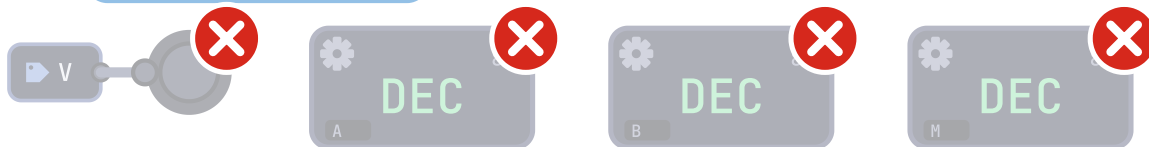
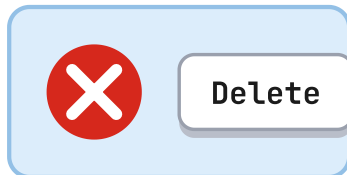
1



2



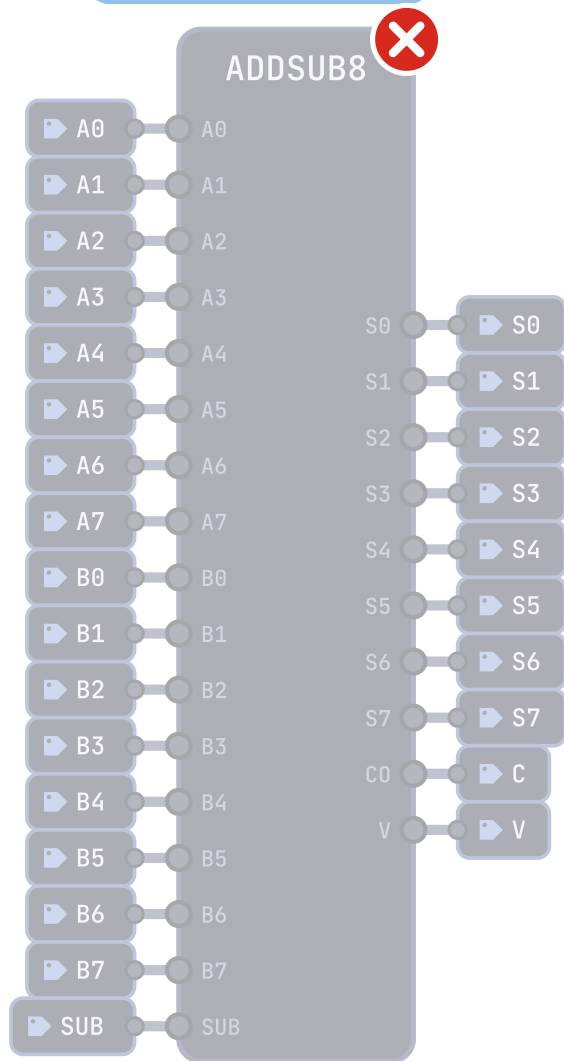
3



4



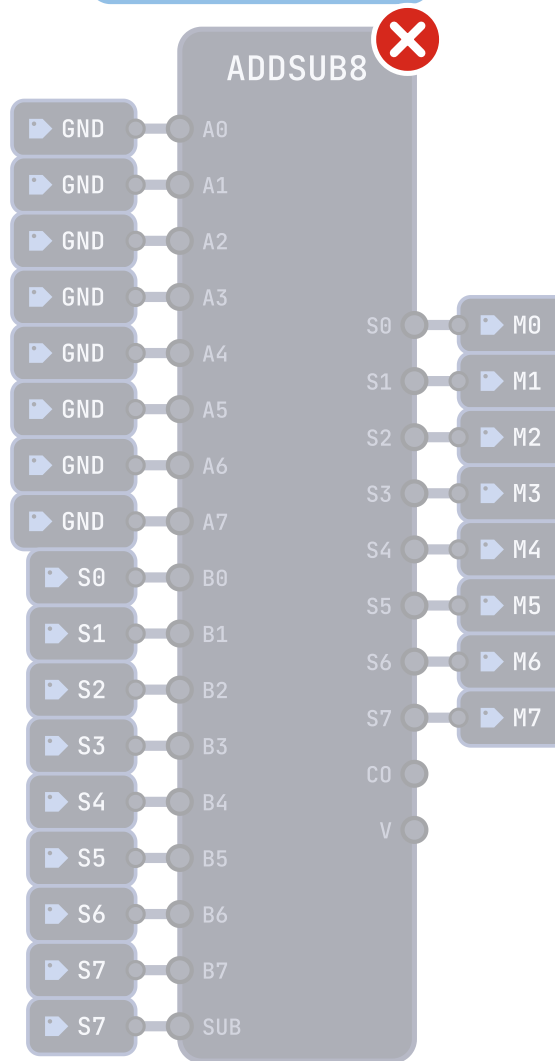
Delete



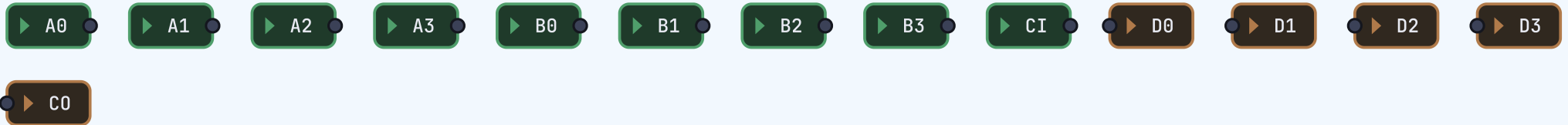
5



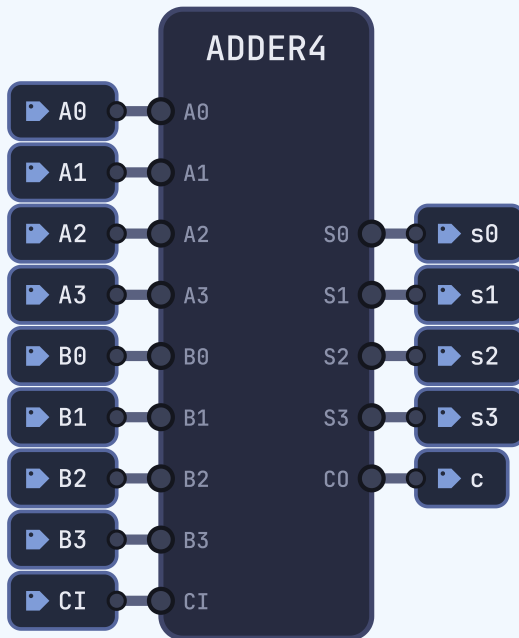
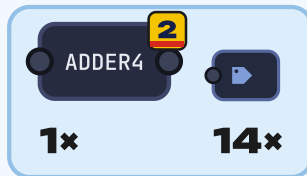
Delete



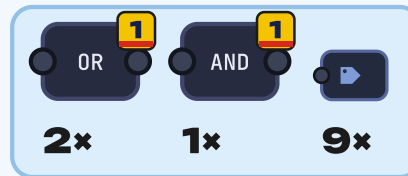
6



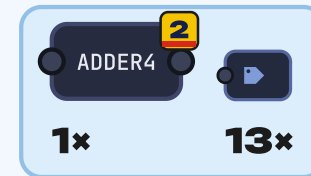
7



8

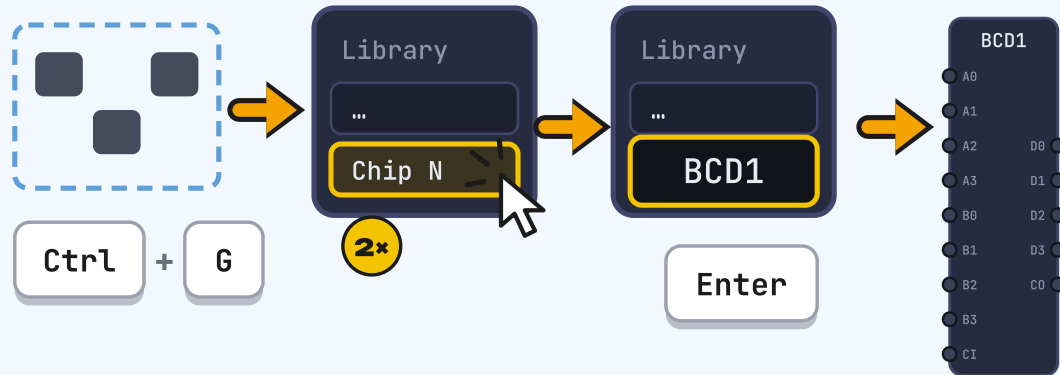


9



BCD1...

10

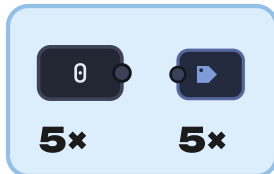


CANVAS

11



12



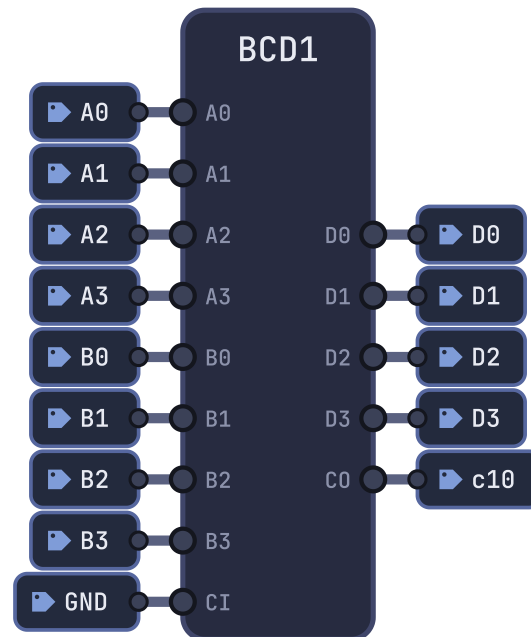
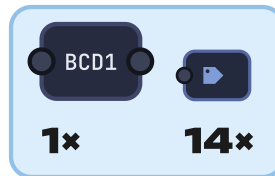
13



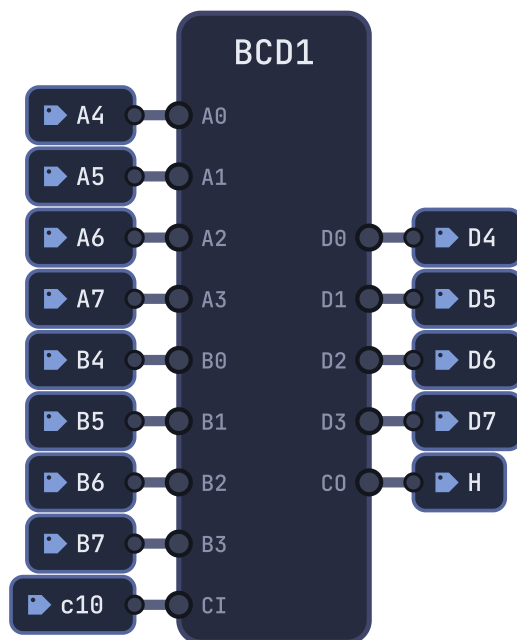
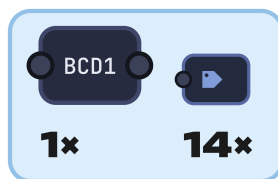
14



15



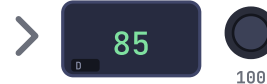
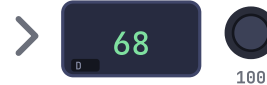
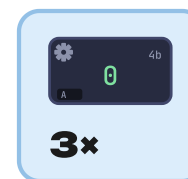
16



17



18



Kit parts

how many times you place each part over all the lessons



11×



11×



1×



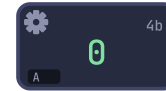
1×



69×



16×



15×



61×



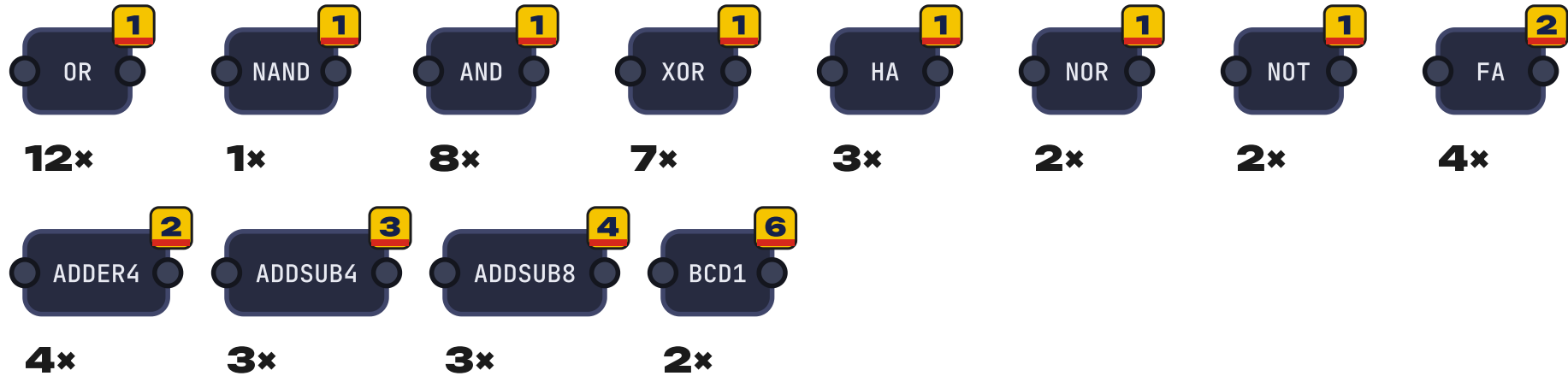
36×



488×

Kit chips

you build them yourself: each one is made of the ones before



Done!



a $\pm$ calculator of 1,208 transistors

set A and B, click SUB – the display shows the magnitude, the "-" lamp the sign

Copyright

© 2026 Daniil Khanin, khanin.info. All rights reserved.

These instructions, the drawings and circuits are original work.
Reprinting, publishing online, selling or passing it on to others,
in whole or in part, is prohibited without the written permission of the author.

The latest version is at khanin.info.



khanin.info

HB-02