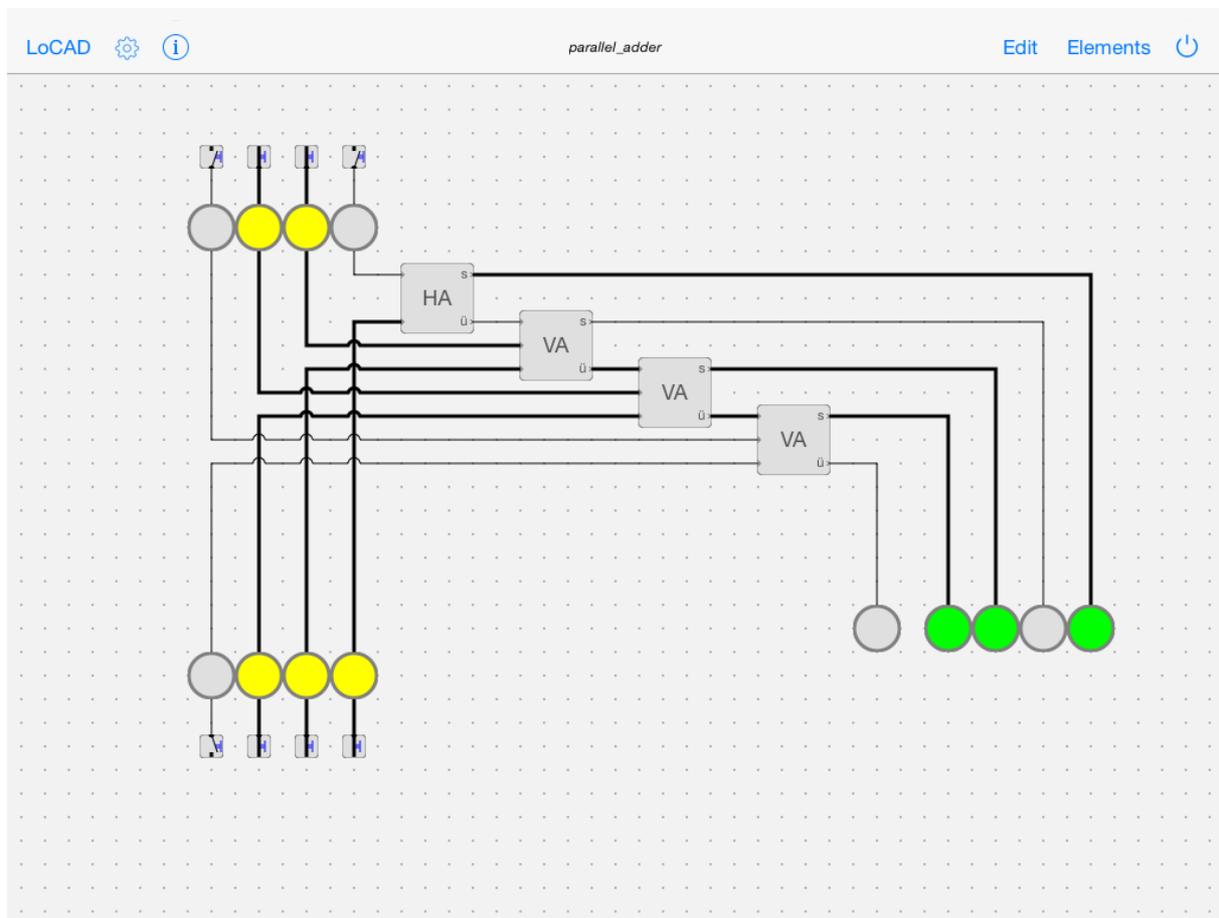


LoCAD Manual

CAD system for logic circuits

The workspace is available in landscape orientation.



Content

1	First Use & Configuration	3
2	Building a circuit	5
2.1	Elements.....	5
2.2	Wiring.....	8
3	Expanding the Circuit.....	11
3.1	Points of Connection	11
3.2	Deleting a Wire.....	13
3.3	Non-element.....	15
3.4	Negation point.....	17
4	Labeling a circuit.....	18
5	Editing selections / object groups	19
6	Saving a circuit	20
7	Setting the clock generator	21
8	Defining ICs.....	23

Translated by Katrin and Bastian Loch

1 First Use & Configuration

Upon first start you will get an empty workspace to build your circuit. If the workspace is not empty, you can create an empty project by using **LoCAD** → **New**.

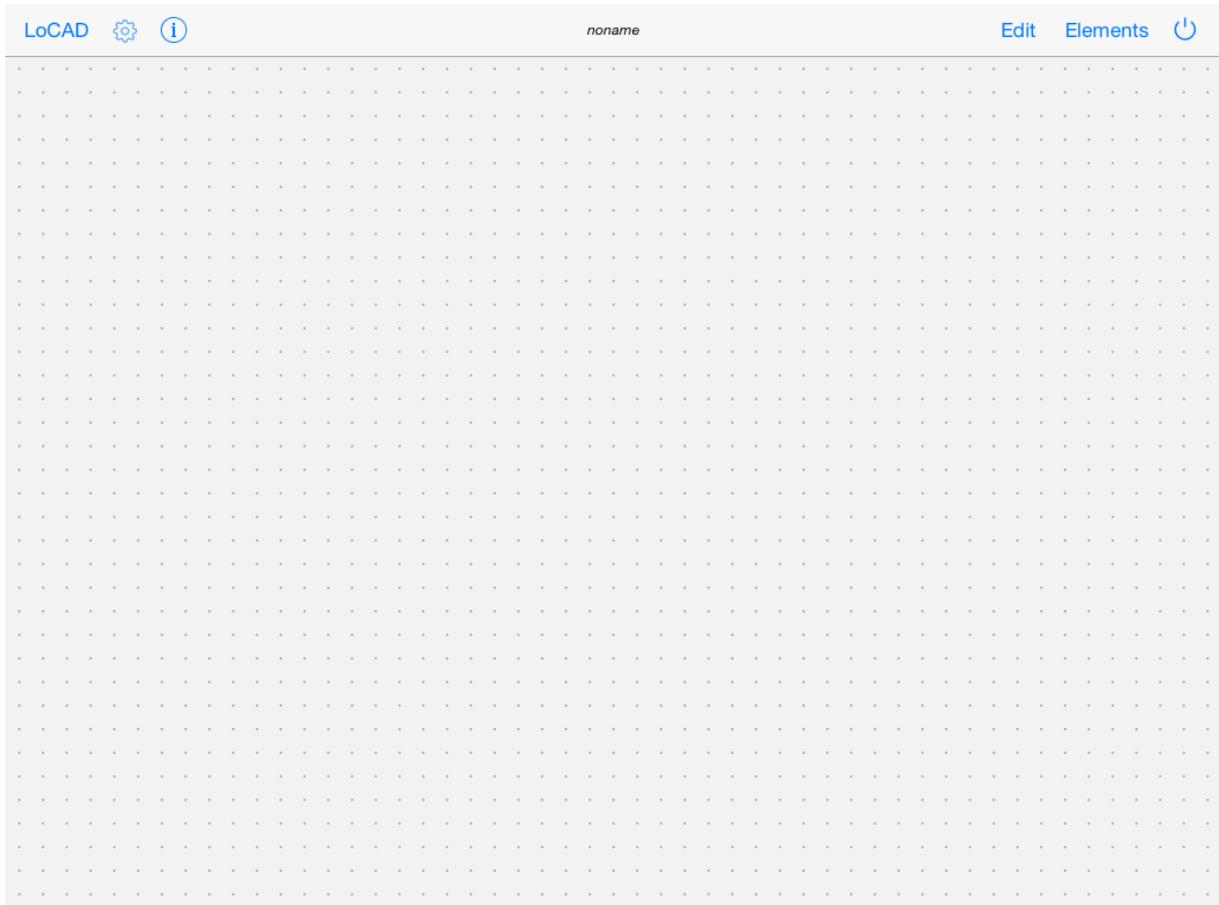


Image 1-1 empty workspace

The button  provides various options for settings, as depicted in the image below.

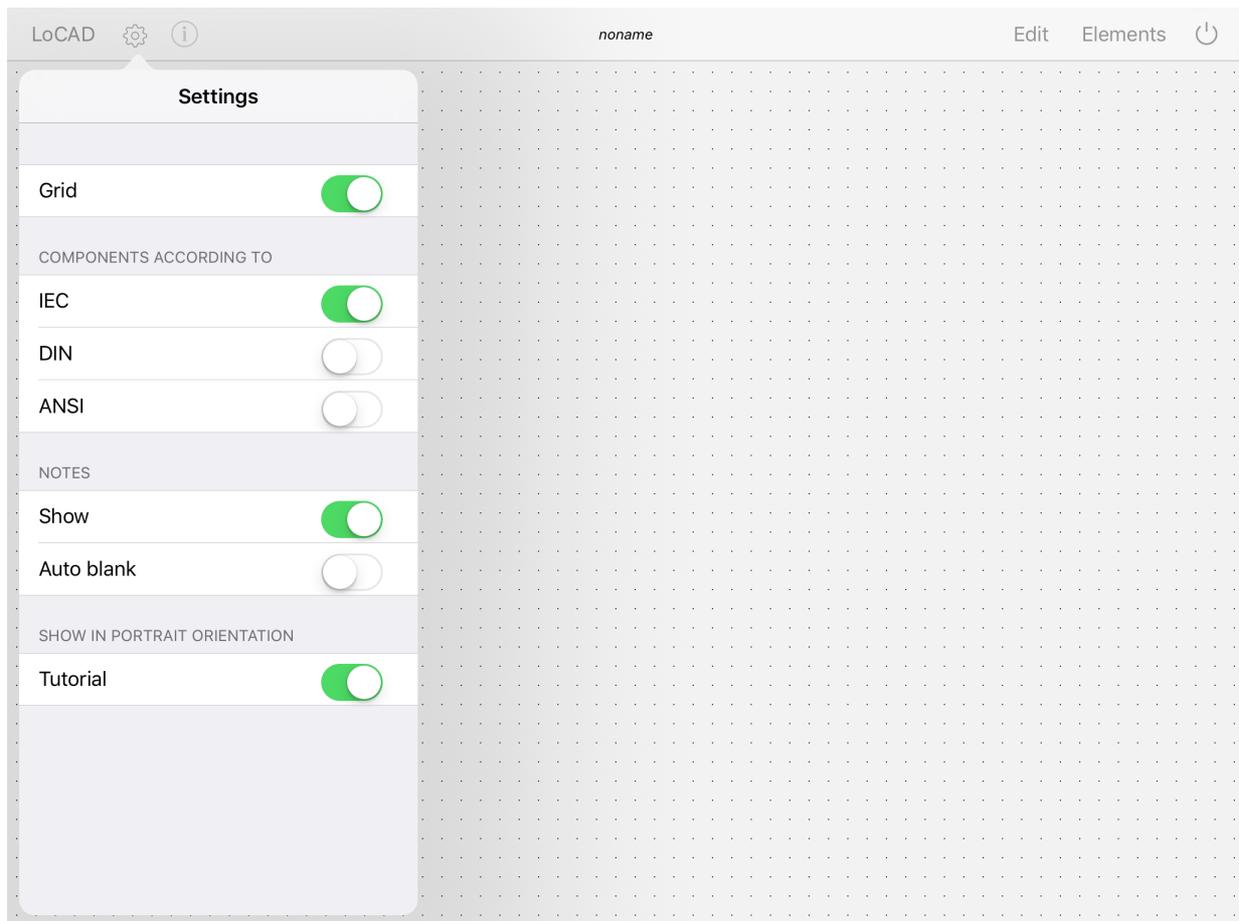


Image 1-2 settings

The grid-switch turns the dotted grid on and off.

You can choose how the elements of your circuits are presented. There are three standards you can choose from:

- IEC - IEC 60617-12
- DIN - DIN 40700 (before 1976)
- ANSI - US ANSI 91-1984

This manual uses the IEC standard.

Notifications, which need to be confirmed with OK, can be turned on and off. There is also an option to automatically fade out a notification after a short period of time without confirming by OK.

The last toggle enables or disables the display of the manual in portrait mode.

2 Building a circuit

2.1 Elements

Touch **Elements** to display the toolkit. Another touch makes it disappear again.

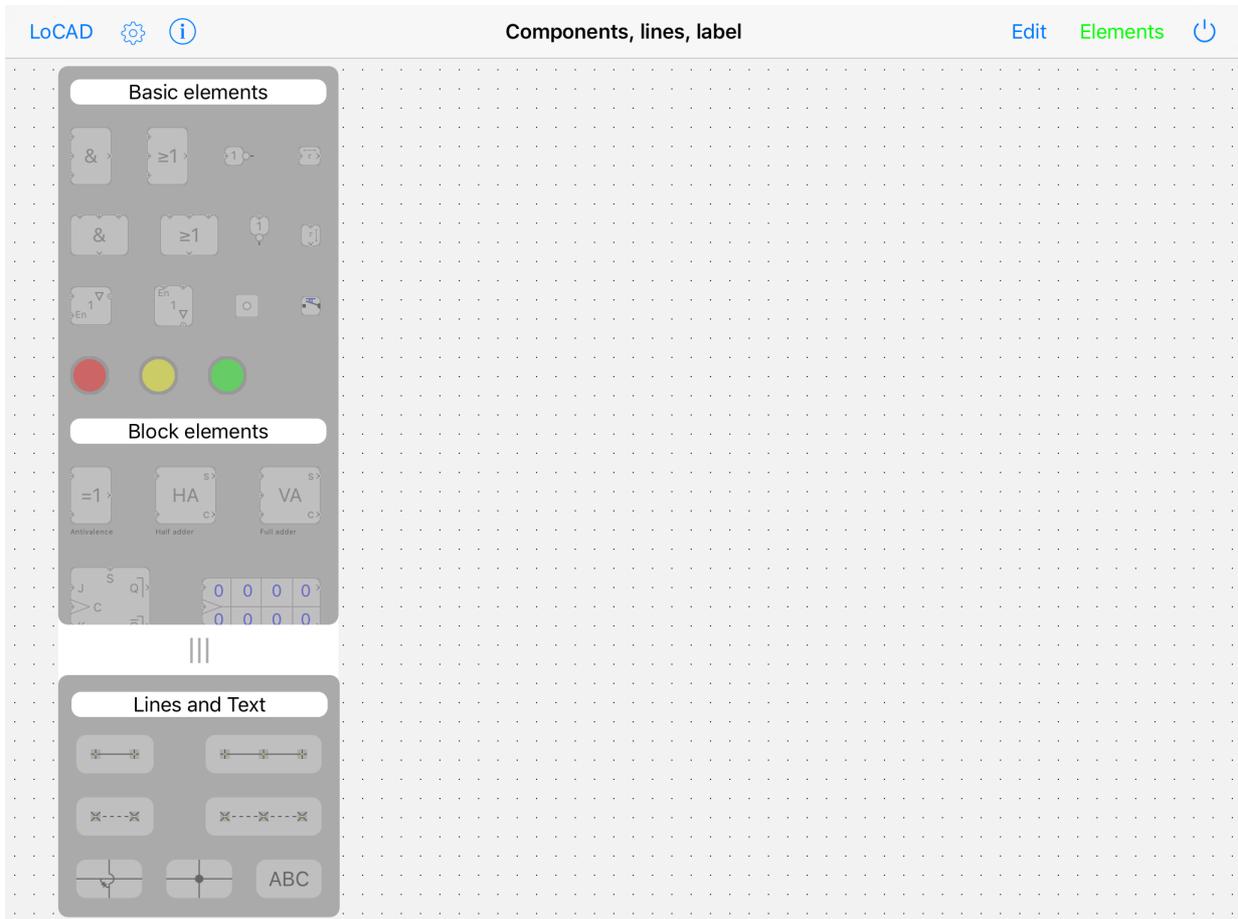


Image 1.2-1 toolkit

In the upper (scrollable) part of the toolkit, you will find *basic elements*, *block elements* and user-defined ICs (Integrated Circuits).

Basic elements are:

- And (in two different positions: inputs left and output right, resp. inputs above and output below)
- Or (“)
- Inverter (“)
- Delay element (“)
- Tristate element (“)
- Negation point
- Switch
- colored LEDs

Block elements are:

- Antivalence (exclusive or)
- Half-adder
- Full-adder
- Flipflop (double-edge-triggered JK-master-slave-flipflop)

Shifting register
 Multiplexer
 Demultiplexer
 Clock generator
 8-bit digital display (with two's complement display option)
 4-bit digital display (with two's complement display option)
 7-segment display
 3-to-8 decoder
 8-bit binary counter
 16 x 4-bit RAM

The explanatory note "*How do computers work? - Introduction to the basic principles of computer hardware*" is currently in preparation.

It explains how the individual components can be interconnected to form functional units.

Upon first start, *user-defined ICs* will be empty. You can store your own ICs here.

The lower part of the toolkit contains tools for wiring and labeling:



Draws wire from start to end.



Draws wire from start to end. The end will be the new starting point.



Deletes wire from start to end.



Deletes wire from start to end. The end will be the new starting point.



Sets or deletes a coupling diode,



Sets or deletes a soldering point.



Adds labeling.

You can move the toolkit horizontally by touching and moving the ||| marks between the upper and lower part.

Select an **and-element** (element with &-sign). The display in the toolbar will change accordingly. Place it on the workspace by touching the workspace. You can move it around until you lift your finger.

As long as you have selected an element, you can place as many of the same kind (in the same position) as you like onto your workspace.

Now place additional elements on your workspace.

For the next exercise, we will delete all elements on your workspace. Touch **Edit** and then touch all set elements on the workspace successively. The marked elements are indicated with a shade. By touching the trashcan, the marked elements will be deleted.

Place a new **and** on the workspace.

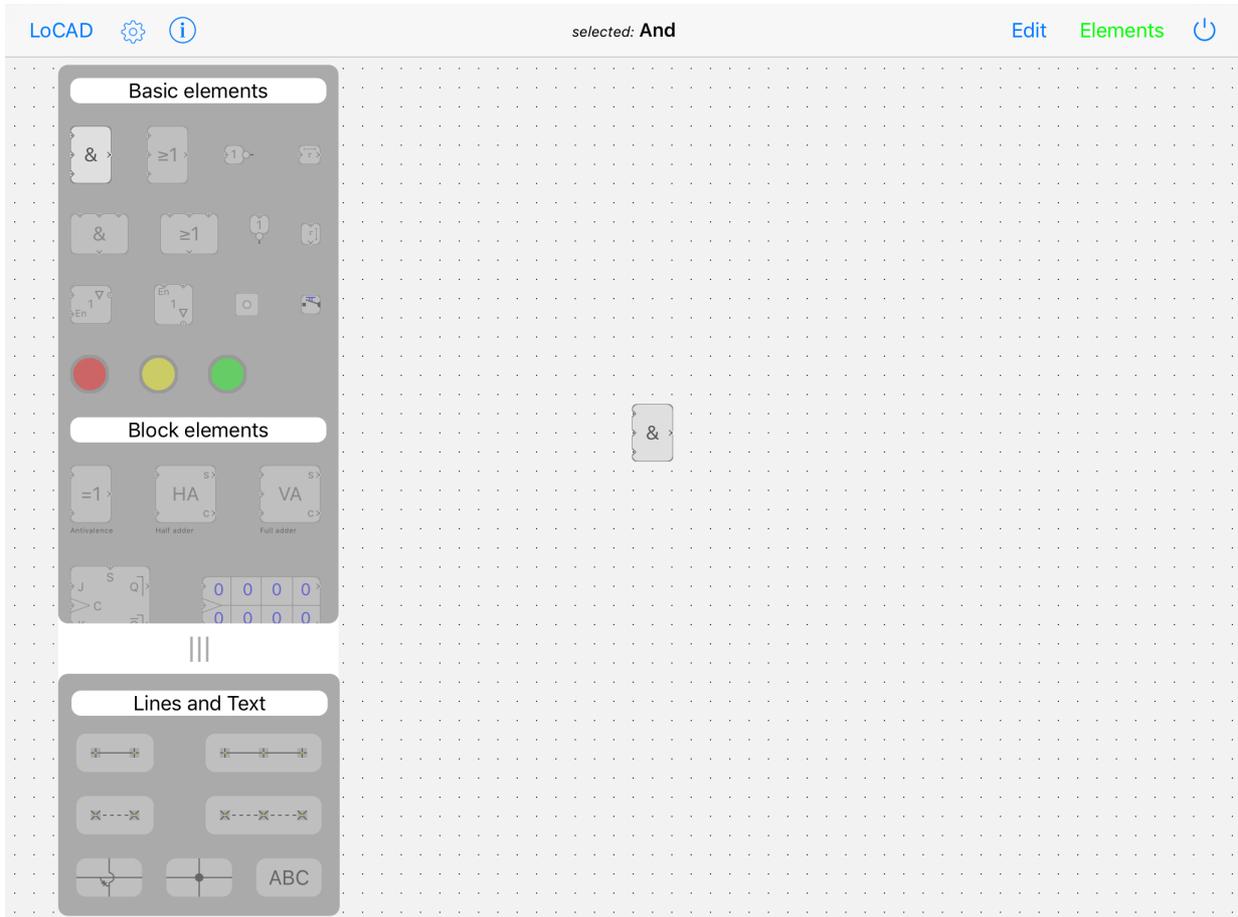


Image 2.2-1 And on workspace

The **and** has three input connections and one output connection. They are marked with >. This way you are able to distinguish better between output and input, which is especially useful in *user-defined ICs*.

Note:

To keep the circuits simple, power lines are not displayed.

2.2 Wiring

Having placed an element on our workspace, we will now wire it.

Select the first wiring tool  (from start to end) from the lower part of the toolkit.

Wiring (Start → End) will be displayed in the toolbar. Touch the output of the and-element. For more accuracy, you can touch and hold. A lens will appear for easier placement.

Note:

It is not necessary that you hit the spot exactly. The system adjusts automatically to the next grid point for each touch.

Furthermore, it tries to draw a direct short connection wire on its own. If this is not possible, you will need to draw the wire yourself with the according detour.

Draw a wire like shown below.

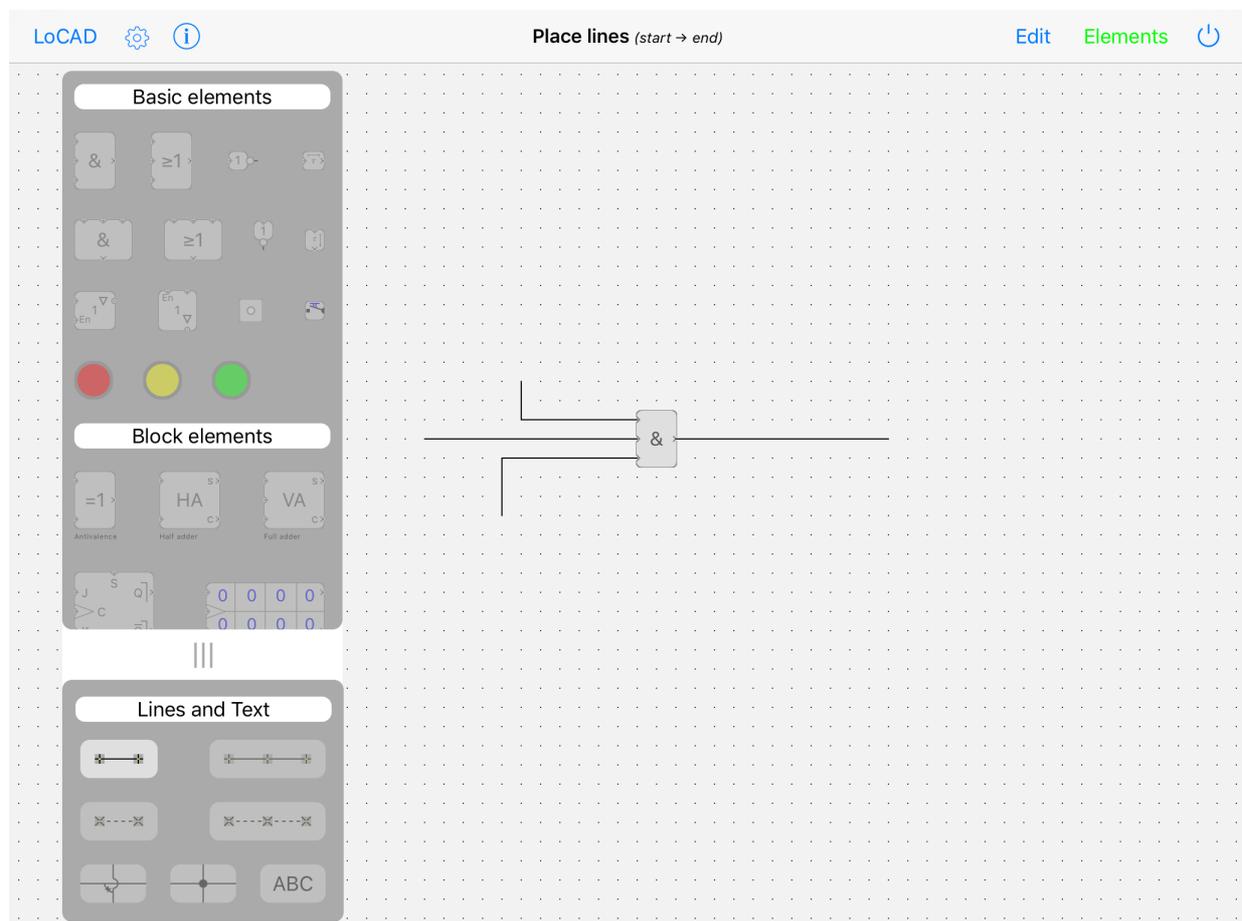


Image 2.2-1 And-element with input and output wires

From the basic elements, select the switch and place switches onto the ends of the input wires.

They will align themselves according to the wire direction. Touch the red LED and place it on the end of the output wire.

Note:

Unlike the switch, the LED cannot be placed directly onto the ending of a wire. The center of the LED needs to be at least one grid unit away from the end of the wire.

Now your workspace should look like this:

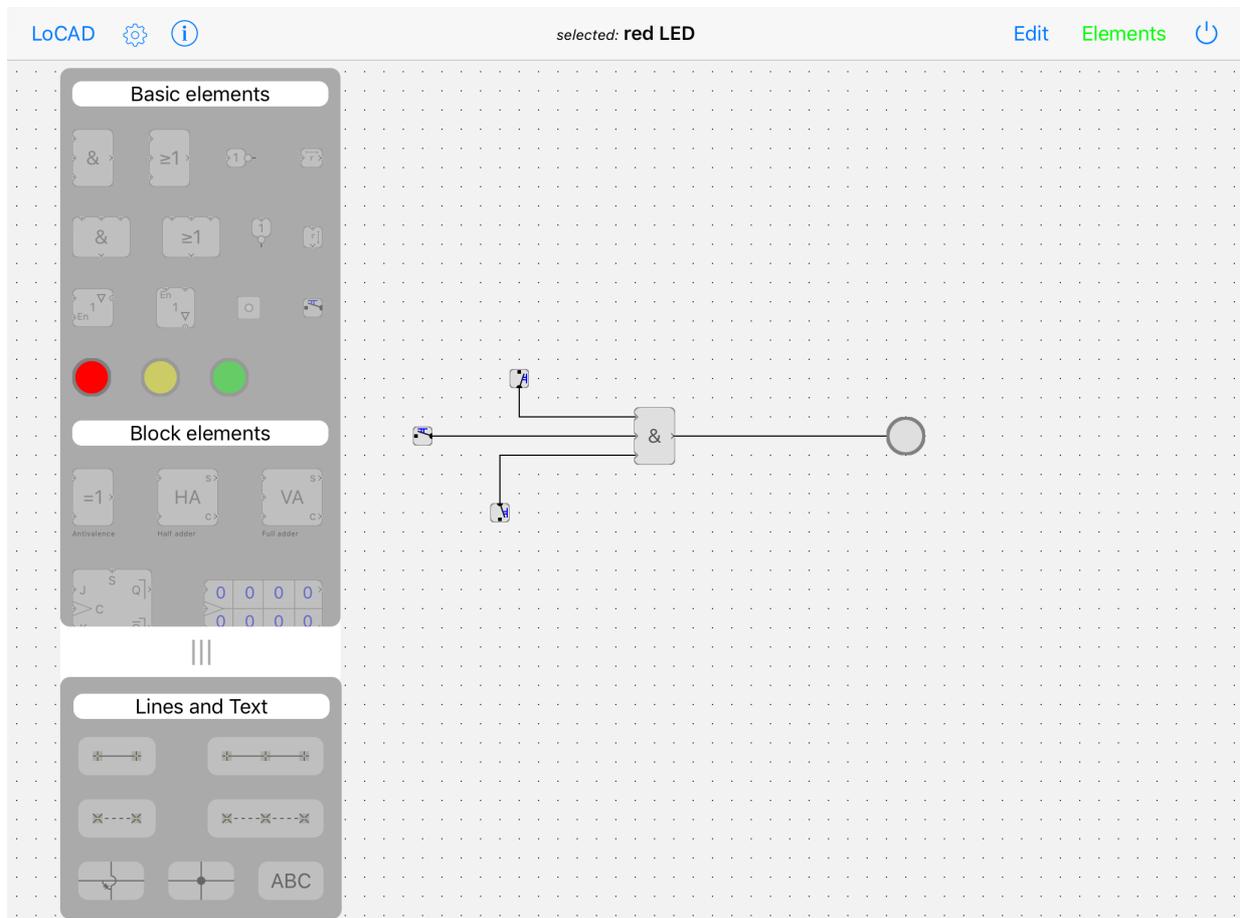


Image 2.2-2 And-element with switches and LED

This one element with connected switches, wires and an LED will suffice for the moment. We will now start up this circuit. Therefore, touch the power switch  in the upper right corner.

Touching one of the three switches will change its status and turn the connected wire into 1-state. The 1-state of a wire is indicated graphically by a broadening of the respective wire.

You can also watch the signal dispersion in slow motion by touching the  button on the upper left and then moving the appearing shifting register to the right.

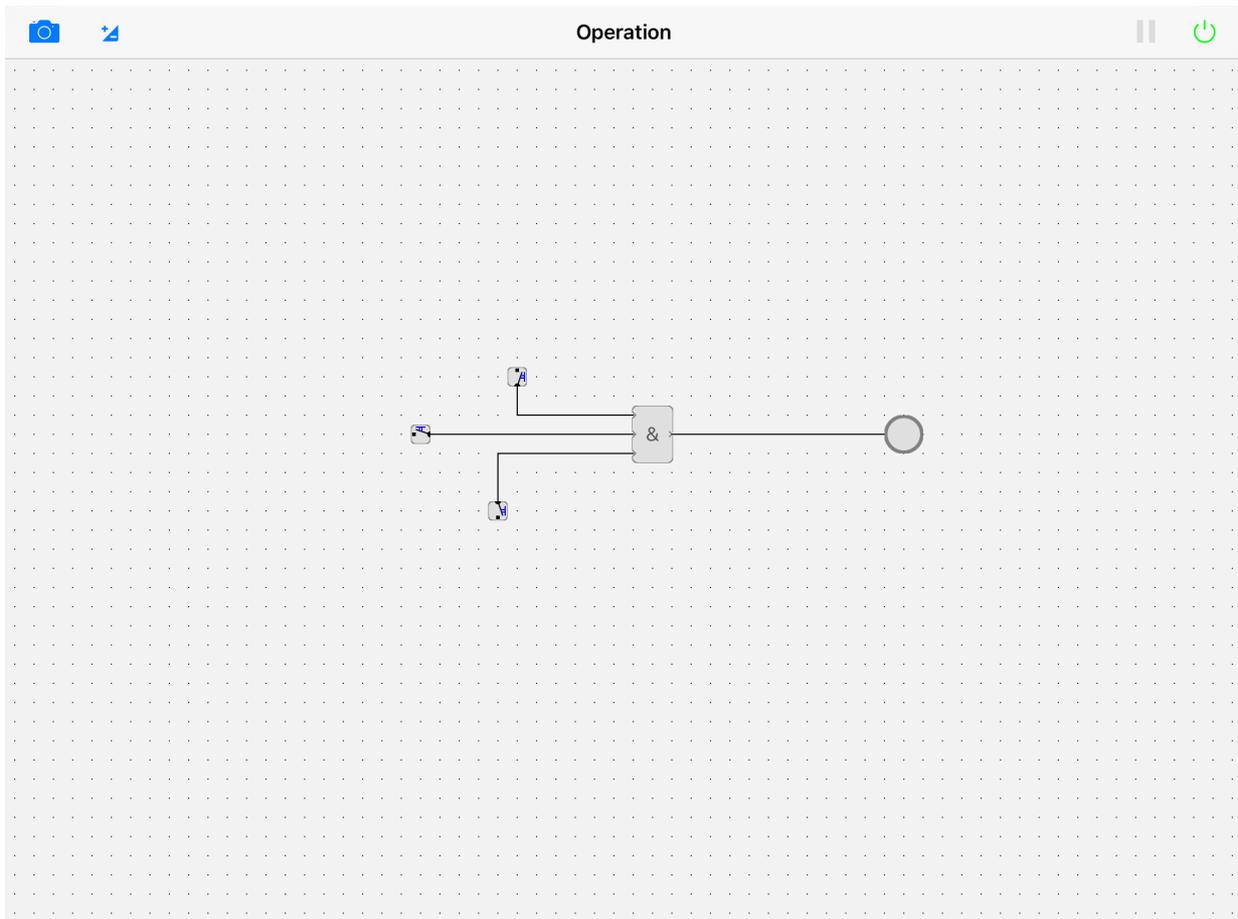


Image 2.2-3 operation mode

Set a deceleration value of about 2 and touch the upper switch again. Now you will see a slowed-down dispersion of the 0-signal.

Touching the switch again will put it into 1-state again.

The upper input of the and-element is now in 1-state, while the output is still in 0-state. An and-element only has the output state 1 if all inputs are in state 1 as well. So, switch all inputs into state 1 and watch the output. As soon as any input is switched back into 0-state, the output will also display 0-signal again.

We will now expand this simple example of a circuit a little bit. Therefore, switch all three inputs back to 0-signal.

3 Expanding the Circuit

Switch off the circuit by touching the power switch and touch *elements* again to open the toolkit.

Choose the or-element (element with the sign ≥ 1) and place one of these above and one below the and-element. To connect the elements, you will need to draw more wires now. You already know how to draw wires. Wire the elements as shown in the circuit below.

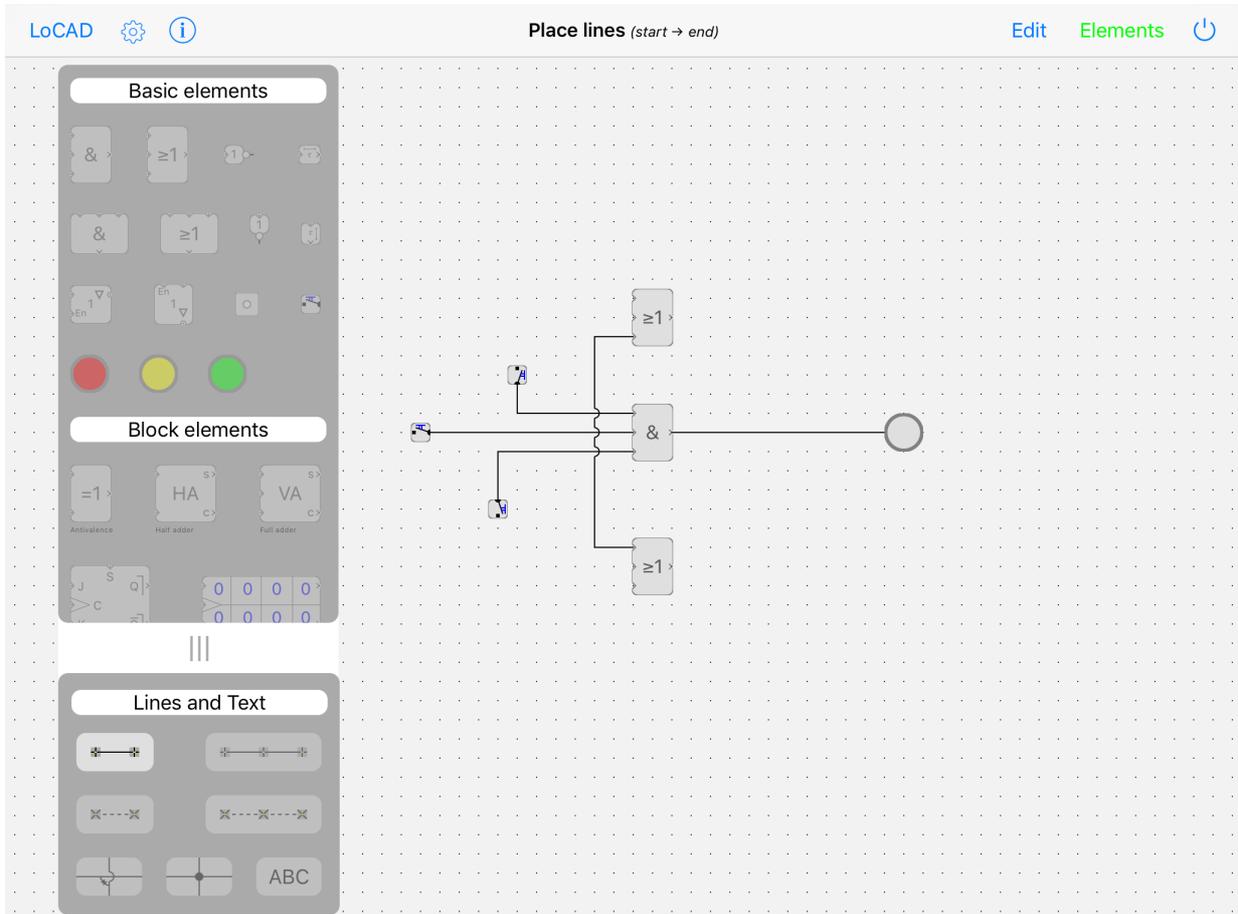


Image 3-1 circuit with and- and or-elements

3.1 Points of Connection

In the exemplary circuit above, the vertical wire crosses the input wires of the and-element. If we want the wires leading to the two or-elements to receive the same signal as the middle input wire of the and-element, we need to solder the wires. For this purpose, touch the soldering point in the middle of the bottom row in the lower part of the toolkit, and place it on the respective points on the wires with a touch. With yet another touch, you can remove the soldering point again.

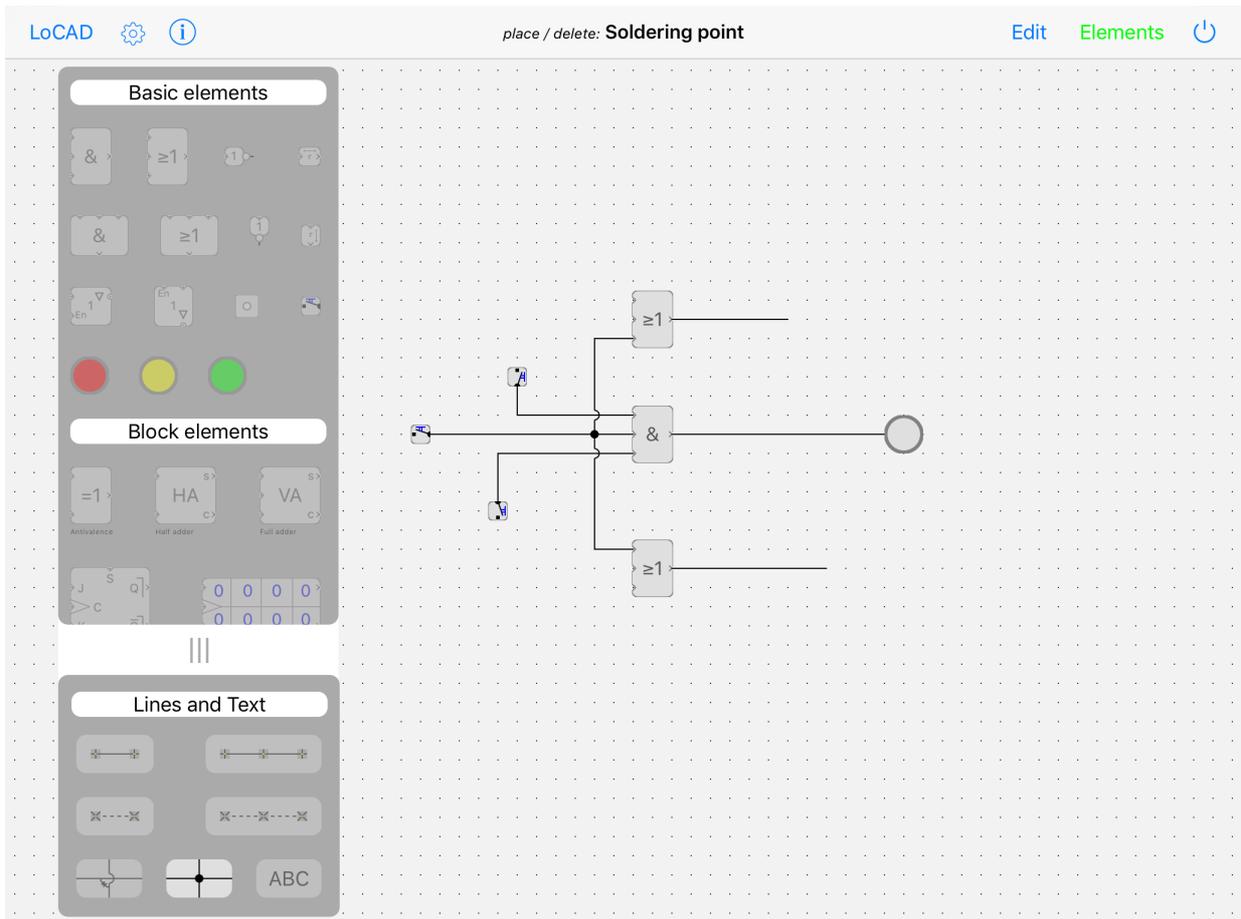


Image 3.1-1 expanded circuit with soldering point

Now put the circuit into operation again with a touch on the power switch, and switch the middle input wire of the and-element to 1-signal.

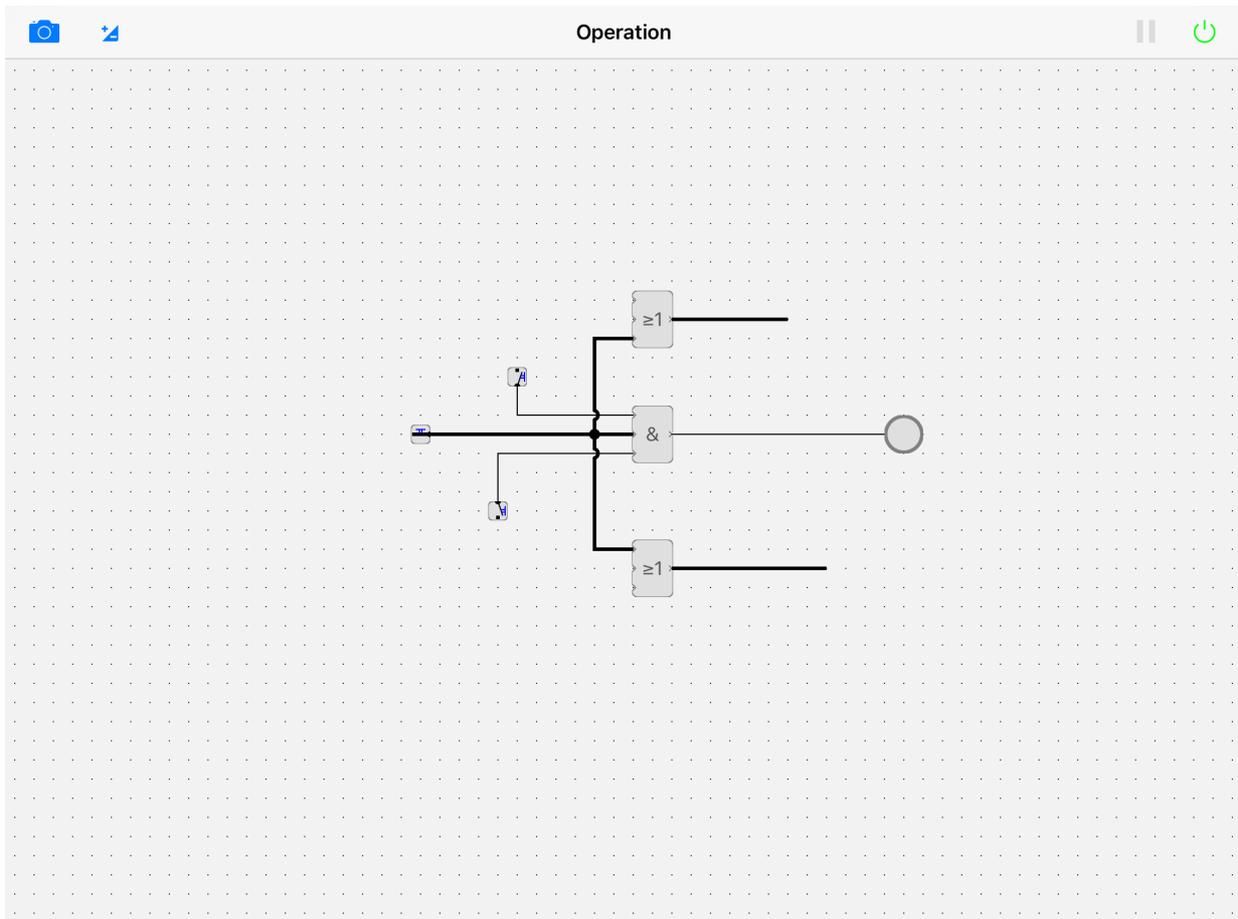


Image 3.1-2 expanded circuit with soldering point in operation

Note:

You can also interrupt the dispersion of the signal by touching the pause button on the right of the power switch. By touching the appearing play button, you let the signal continue.

3.2 Deleting a Wire

We want to change the circuit above now in a way that the upper or-element is connected with the upper input line of the and-element, and the lower or-element is connected to the lower input line of the and-element. To achieve this, a part of the vertical wire needs to be deleted.

Deleting a wire is only possible if the line is in 0-state. So, switch the middle input line back to 0-state, open the toolkit, and touch the  tool in the lower part of the toolkit. Then touch the start and the end of the wire part you want to delete.

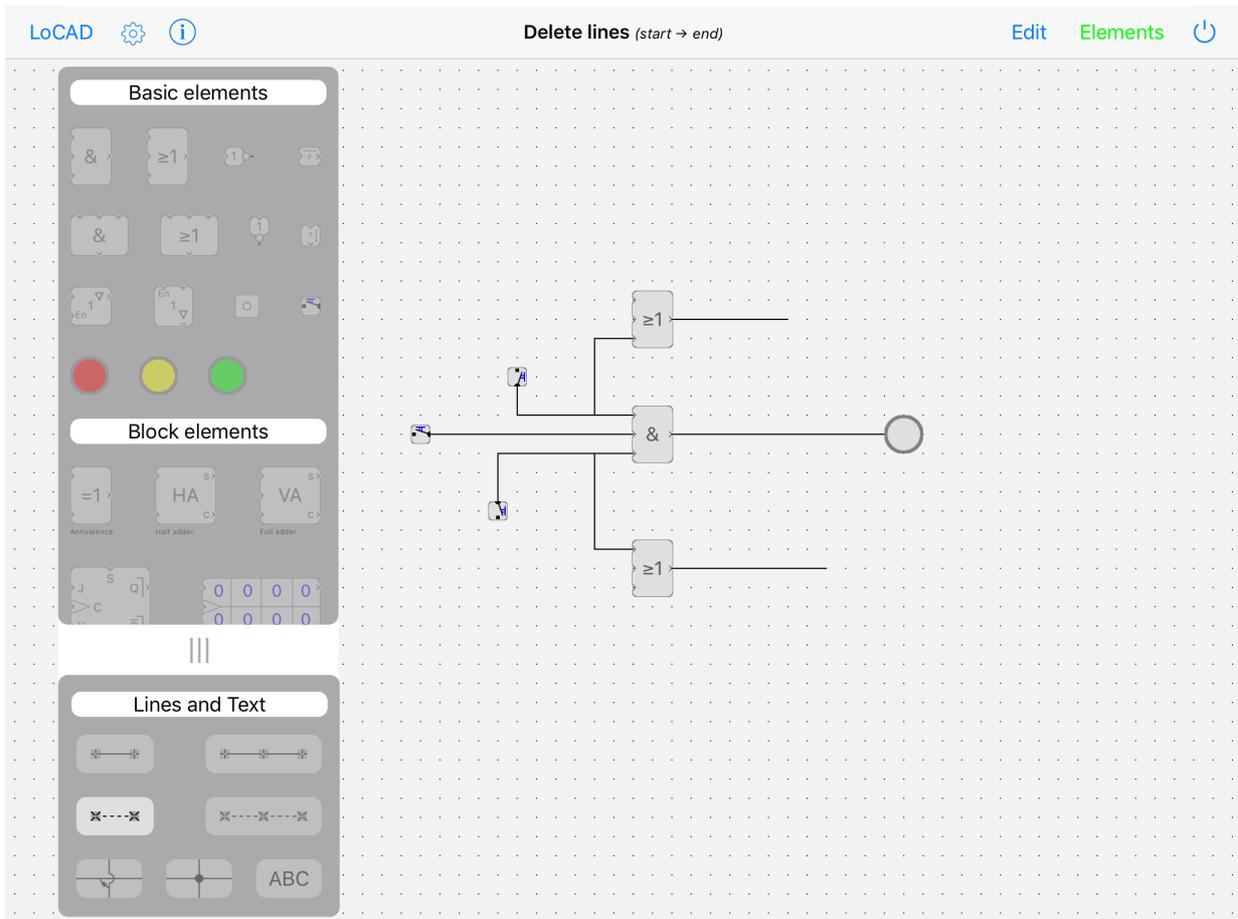


Image 3.2-1 expanded circuit with deleted part of a wire

Note:

Deleting a part of a wire this way is only possible if the start and the end point are horizontal or vertical to each other.

A whole wire transmission (with all possible branching) can be deleted by changing into the edit mode with the **Edit** tool and touching the respective wire on any point.

The wire transmission will then be indicated with a shade and can be deleted with a touch on the wastebin.

3.3 Non-element

Now we want to supply the lower and-element with the inverted signal of the lower input line. The first possibility to achieve this is to place a non-element ( resp. ) on the wire leading to the upper input of the or-element.

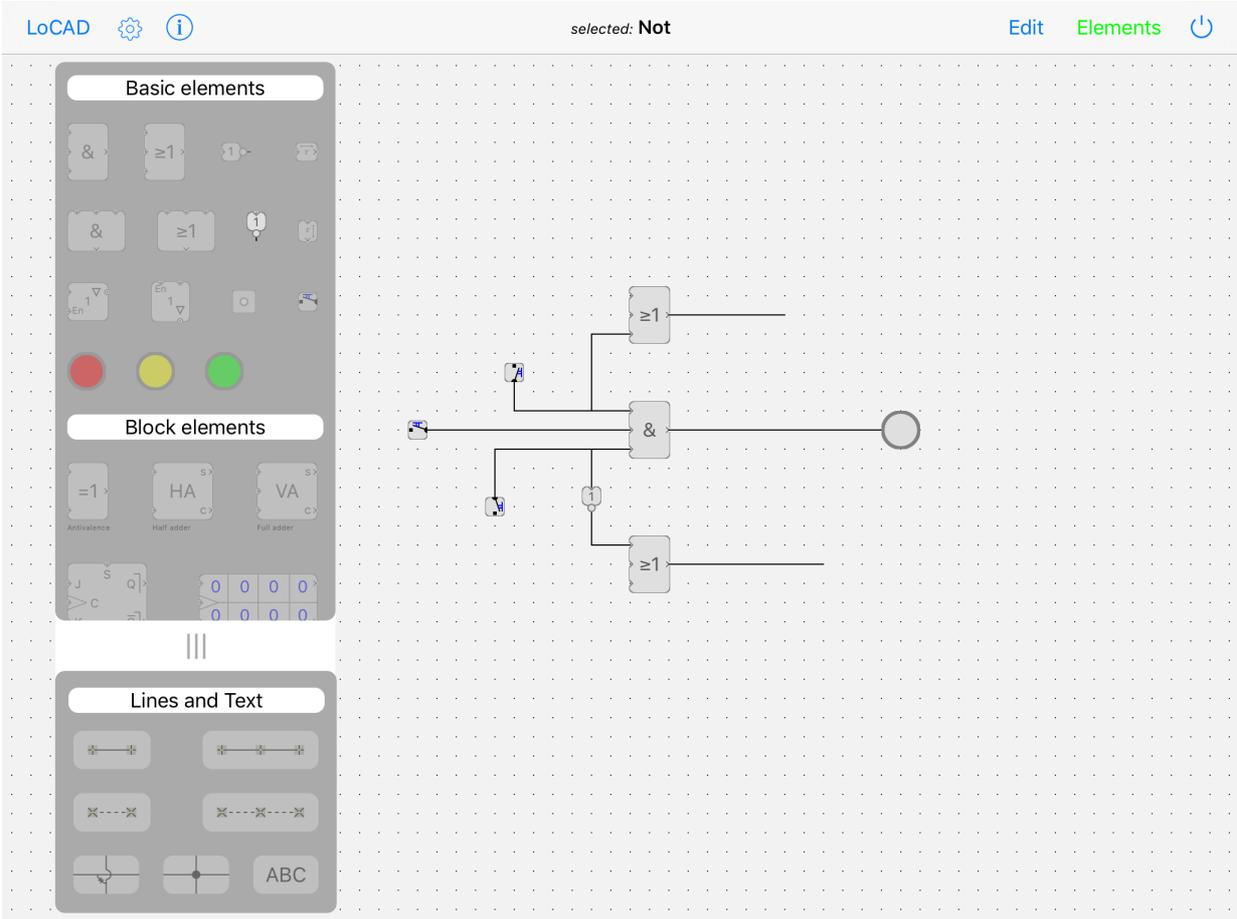


Image 3.3-1 expanded circuit with non-element

After switching the power on, the state shown in the image below will arise.

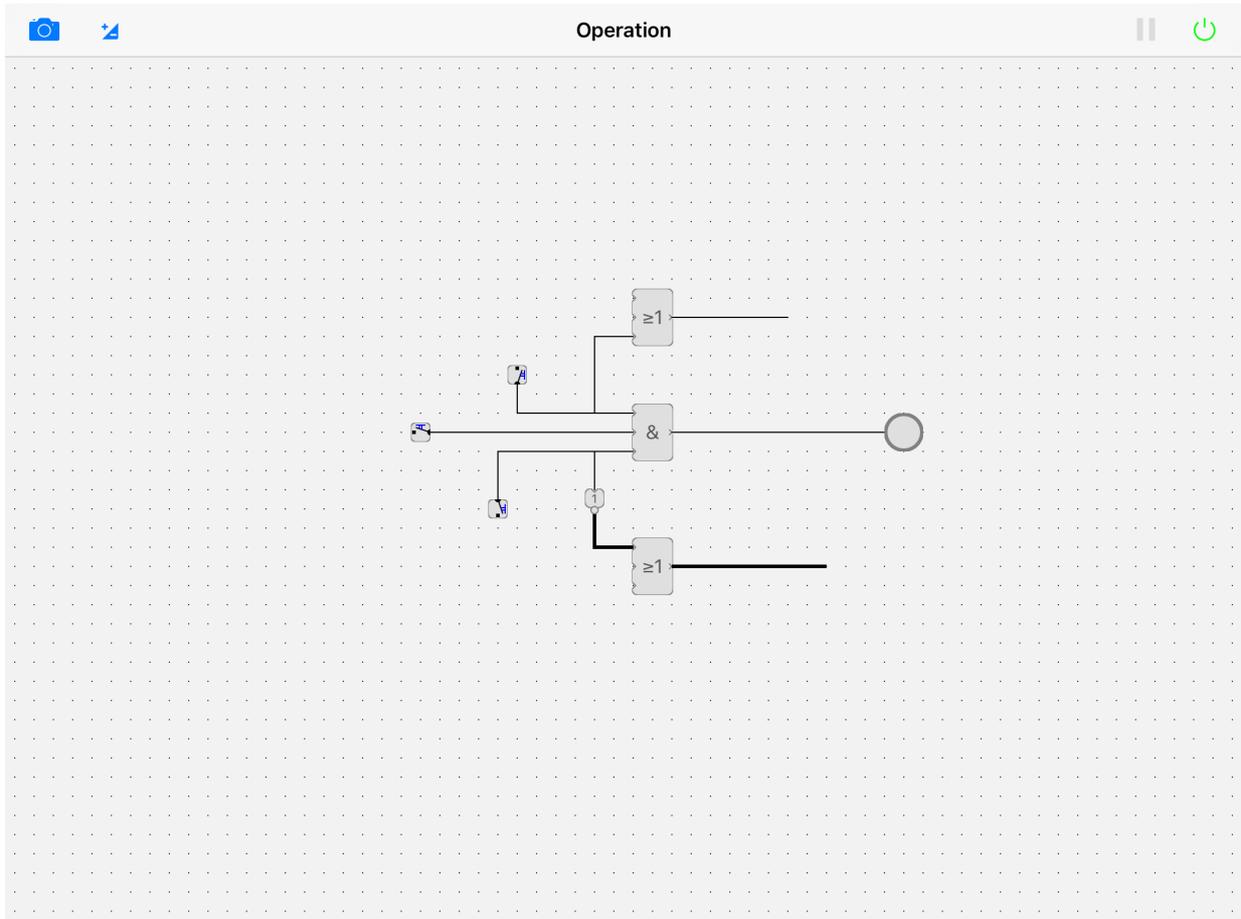


Image 3.3-2 expanded circuit with non-element in operation mode

Alternatively and more space-saving, you can also use a negation point.

3.4 Negation point

Negation points can be placed at all inputs and outputs of the basic and block elements. By placing a negation point at its output, an and-element becomes a non-and-element (NAND element), and an or-element becomes a non-or-element (NOR element).

If you want to delete the non-element and replace it by a negation point, you need to consider the following:

If a non-element is deleted, a gap arises in the line, which now conducts a 1-signal "at a loose end". Trying to close the gap with a piece of wire, the note "*make line currentless first*" appears.

There are several solutions for this problem:

Place a switch at the wire with the 1-signal. When you touch the power switch, the wire will be in 0-state. Now you can delete the switch and close the gap.

or

Switch the input of the non-element into 1-state (with the switch at the lower input line of the and-element) before you delete the non-element. Then delete the non-element and switch the input line back into 0-state.

Now that the line is continuous again, you can choose the negation point  from the basic elements in the toolkit (third element in the third row) and place it at the input of the or-element. If you touch the negation point at the element again, it is deleted.

 **Note:**

If an input or output is not wired, you cannot place a negation point there.

4 Labeling a circuit

Touch the tool  at the lower right in the toolkit. Then touch the place on the workspace where you want to write.

After entering the text, fade out the keyboard by touching the tool  on the lower right.

Having marked a text through *edit*, you can move, edit and/or change the size of the text.

5 Editing selections

Let's suppose the circuit that we have built so far was placed impractically and should better be placed in a different area of the screen. Instead of deleting all wires and elements and building them up in a different place again, you can mark the circuit as an object group and move the selection to another place as a whole.

Choose the tool  for selecting an object group under **edit**. Now you can mark the area of the object group that you want to edit.

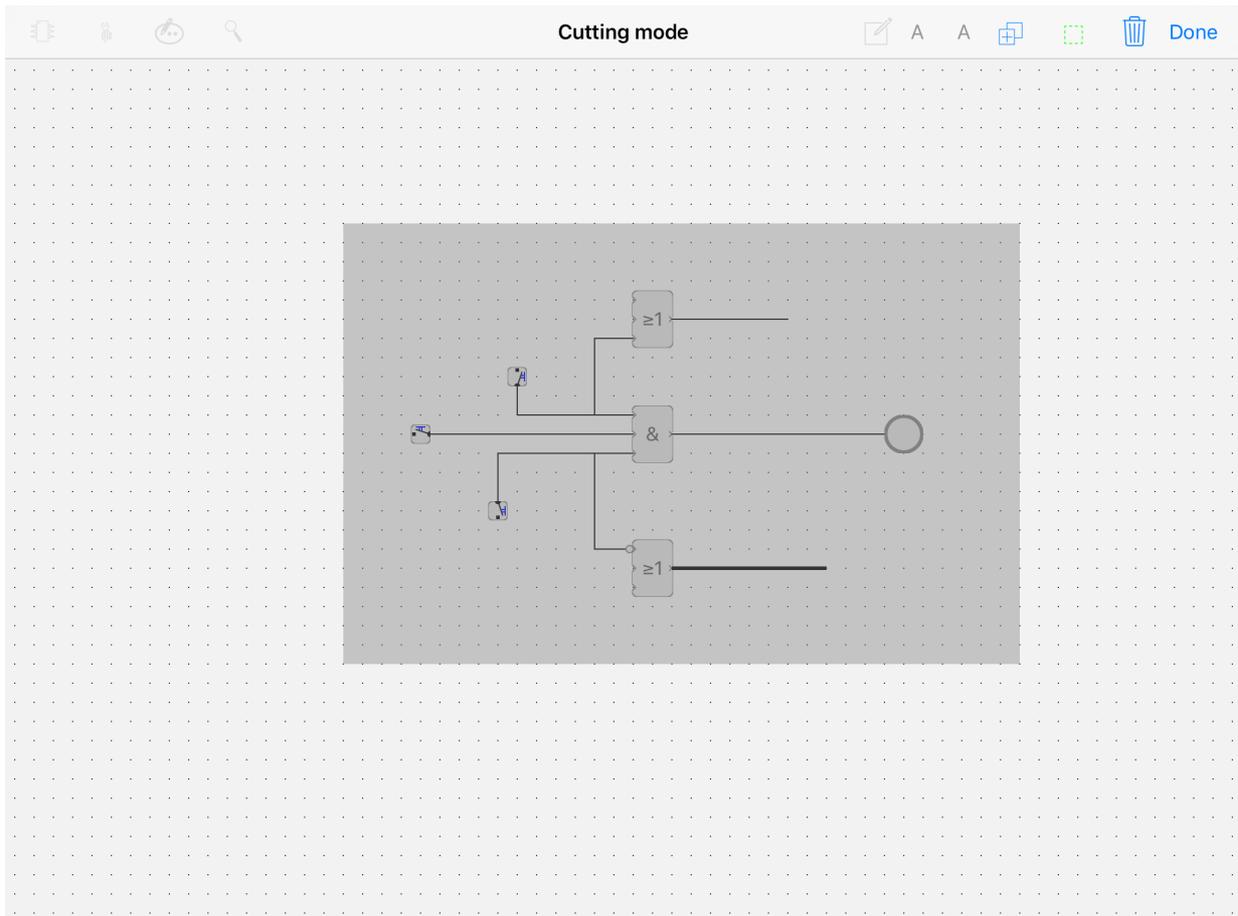


Image 5-1 marked area

The selection can now be moved, copied or deleted. For moving, put your finger down on the selection and shift it to another place.

The selection will still stay marked, so you can move it again or make adjustments in case you are not yet quite content with its new position.

Now we want to copy this marked selection to another place. Touch the tool . This copies the elements that are in the scope of the selection. This process is indicated through the selection's color becoming lighter. Now you can move the selection to another place.

6 Saving a circuit

For preserving a built circuit, you can find the menu items **save** and **save as** under the **LoCAD** button. A circuit that has already got a name will be saved by clicking **save** without further notice.

If the circuit has not yet got a name, as in our case (it still says *noname* in the headline), a dialog window will pop up. The circuit is only saved after you have entered a name and clicked **OK**.

By choosing **save as** you can also save the circuit under a different name.

It is possible to forward, resp. transfer saved circuits as mail attachment.

The following chapter explains the possibility of generating signals automatically with a clock generator.

7 Setting the clock generator

Instead of laying a signal on a line by hand with the help of a switch, you can also use a clock generator.

We will make a new circuit so you can directly comprehend the setting options of the clock generator. Start the following new circuit via **LoCAD** → **New** and operate it with the power switch.

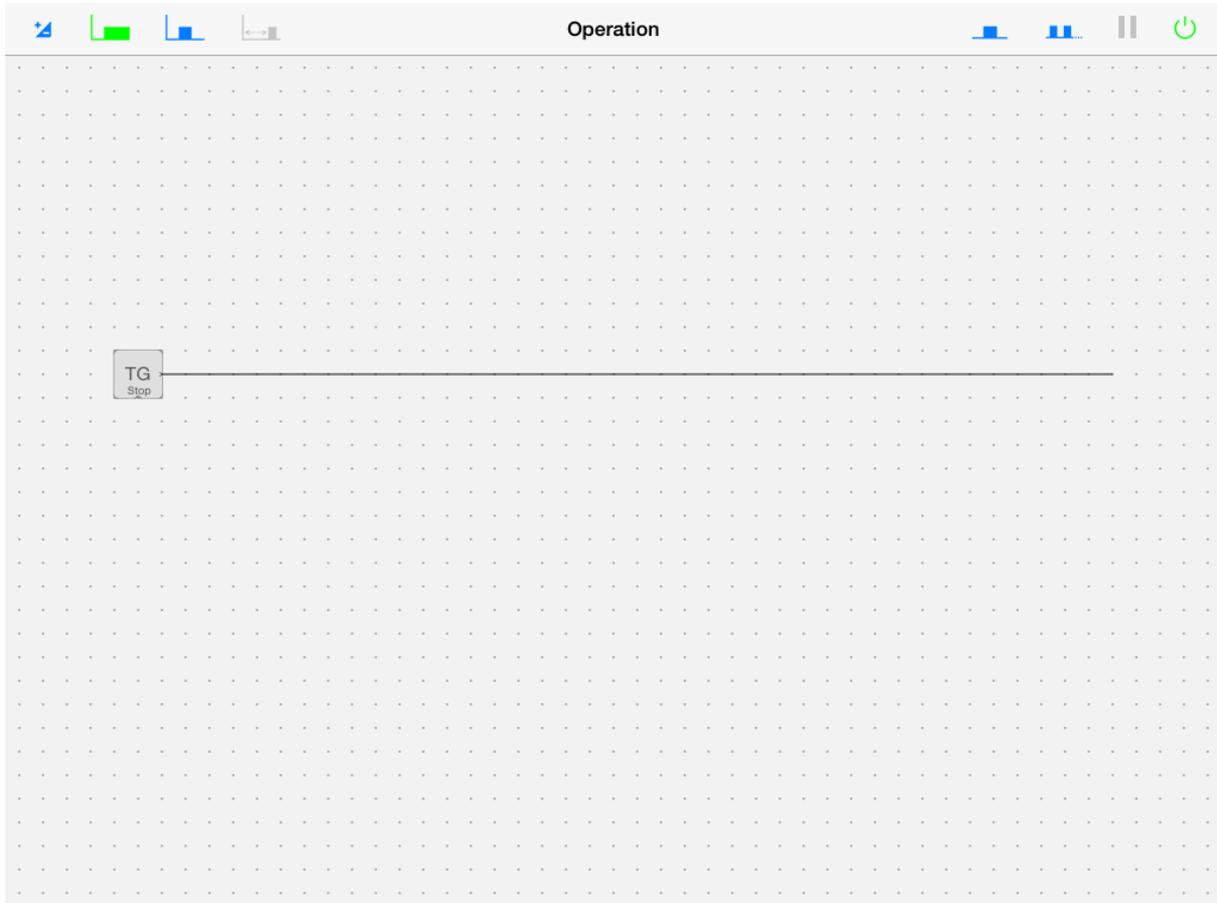


Image 7-1 circuit with clock generator

On the upper right, you will find two new tools:

■ for single clock pulse

■■ for continuous pulse

Touching the single pulse button, the clock generator will generate a single clock pulse, while with the continuous pulse button, the clock generator will keep generating clock pulses until you hit *pause* or switch the whole circuit off.

If you now watch the signal distribution carefully – maybe with a big delay – you will notice that the clock generator does not submit a new signal state until the actual state has reached the end of the line.

This working mode of the clock generator is not quite realistic, but it will surely not disturb a sensible circuit. In any case, it simplifies the simulation and makes sure that no unclear states arise in the circuit. It is compatible with reality insofar as, also in reality, a

clock signal may only come in when all elements involved are in a stable state. This mode of the clock generator is set with the button  (second from upper left).

If it is necessary that the clock generator work independently from current signals, you can make it submit independent clock pulses with the button .

When setting the clock pulses, you can change the phase relationship between 1- and 0-phase with the tool .

Once chosen, a setting stays valid until a new setting is made.

Now set the phase relationship between 1-phase and 0-phase to 4:12.

After starting the clock generator, you get the following situation with independent clock pulses on the line:

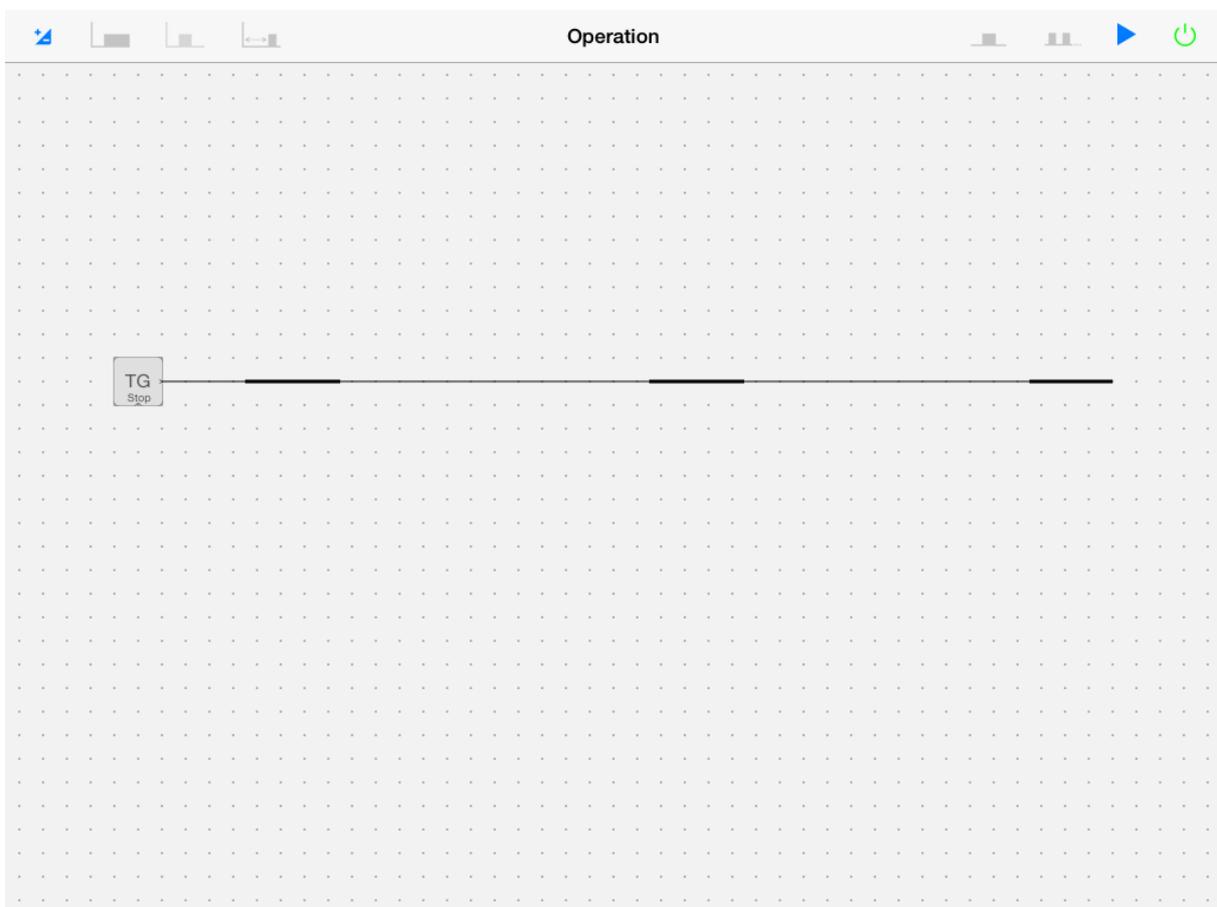


Image 7-2 independent clock pulses

Note:

The clock generator has an input on the bottom, which is labeled with “stop”. If this input receives a 1-signal, the clock generator is switched off. It can only be re-activated again when the stop input has 0-signal.

8 Defining ICs

In the following, we explain the possibility to “shrink” extensive circuits into integrated circuits (abbr. **IC**). This eliminates the constraint of the workspace to one screen and, through successive integration of circuits, allows you to build up complex systems, e.g. a complete microprogrammed computer.

We will explain the definition process with the example of the integrated circuit SN7408 from the SN74xx series by Texas-Instruments. The integrated circuit SN7408 contains four and-elements with two inputs each. It has the following terminal assignment:

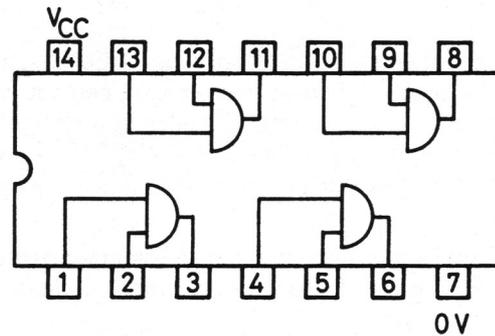


Image 8-1 terminal assignment of SN7408

In order to define this circuit as an autonomous element with the name "SN7408", you first need to build the following circuit by placing four and-elements and wiring them.

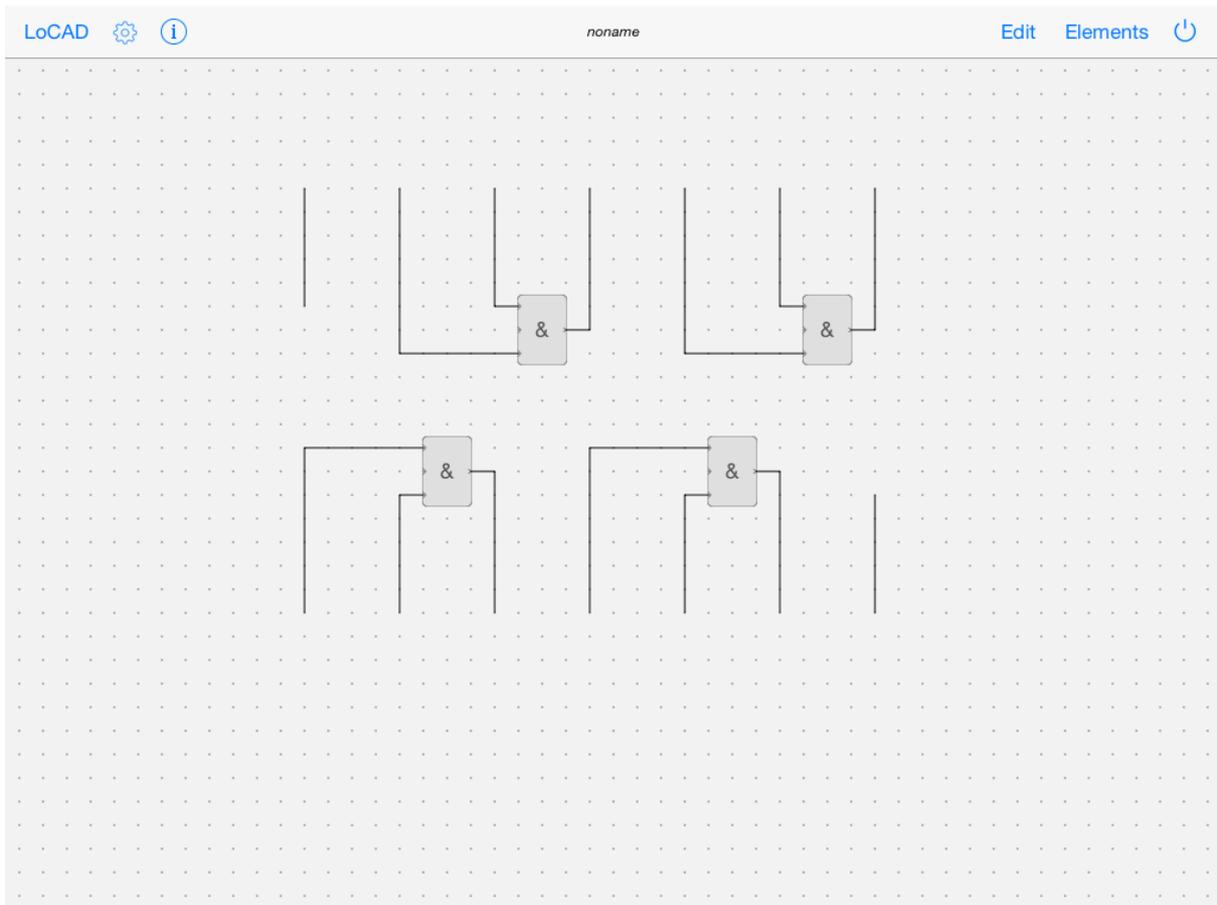


Image 8-2 inner circuit of SN7408

The wire pieces on the upper left and lower right that are not connected to an element have been inserted so that, with the IC definition later on, also pins no. 14 and 7 appear.

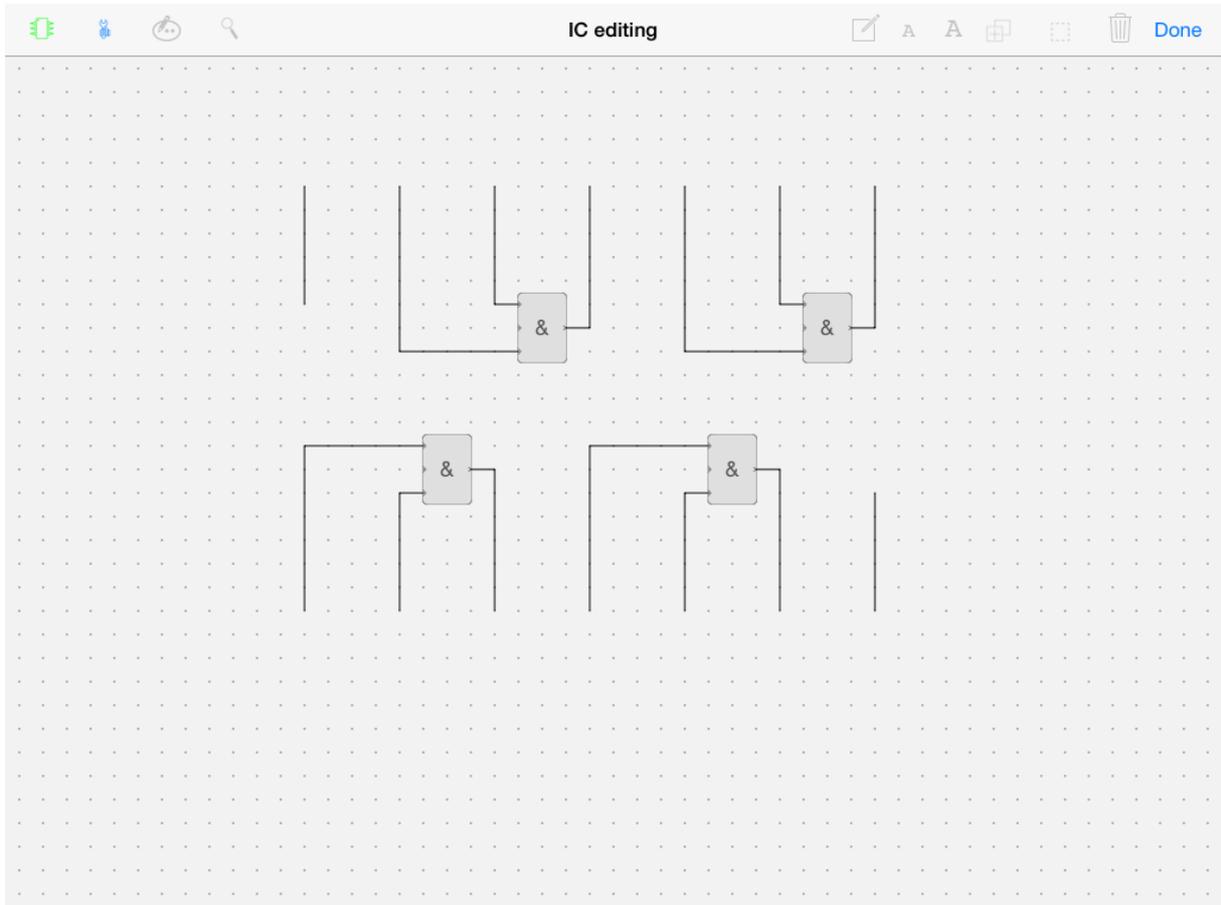


Image 8-3 IC editing

Now select the IC edit mode by touching **edit** and then the symbol  on the upper left.

A touch on  now enables you to define the area that shall be shrunk to an IC. Open up this area as shown in the following image.

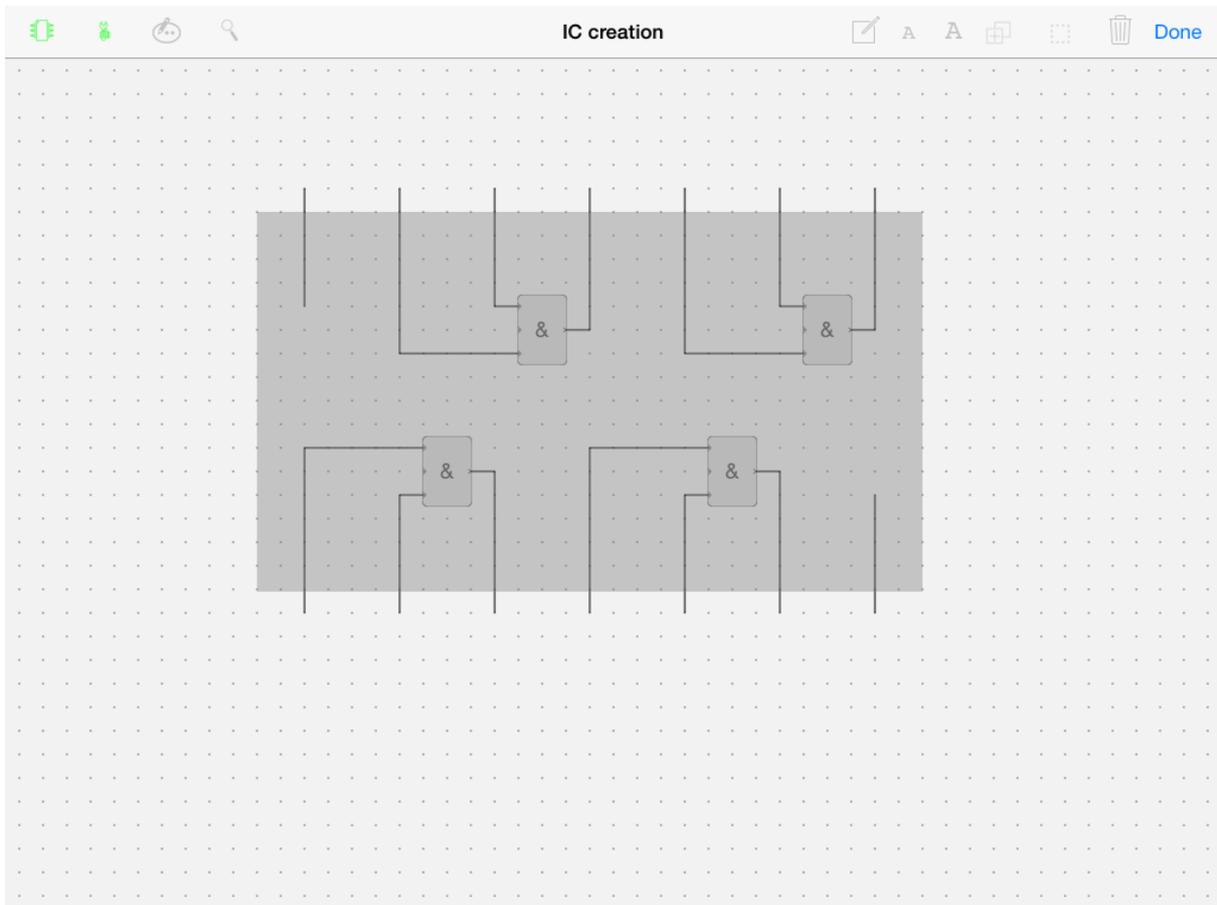


Image 8-4 IC area definition

Then enter a short name for the IC and a significant file name for the circuit. In our case, it makes sense to use „SN7408“ for both the IC and the file name. After touching **OK**, the underlying inner circuit will automatically be saved as „SN7408.stg“, and the integrated circuit will be saved as „SN7408.icc“. It will be available in the toolkit from now on.

If you build a new circuit now and place the IC just defined, you will see that the inputs are automatically numbered with *e1* to *e10*, and the outputs with *a1* to *a4*.

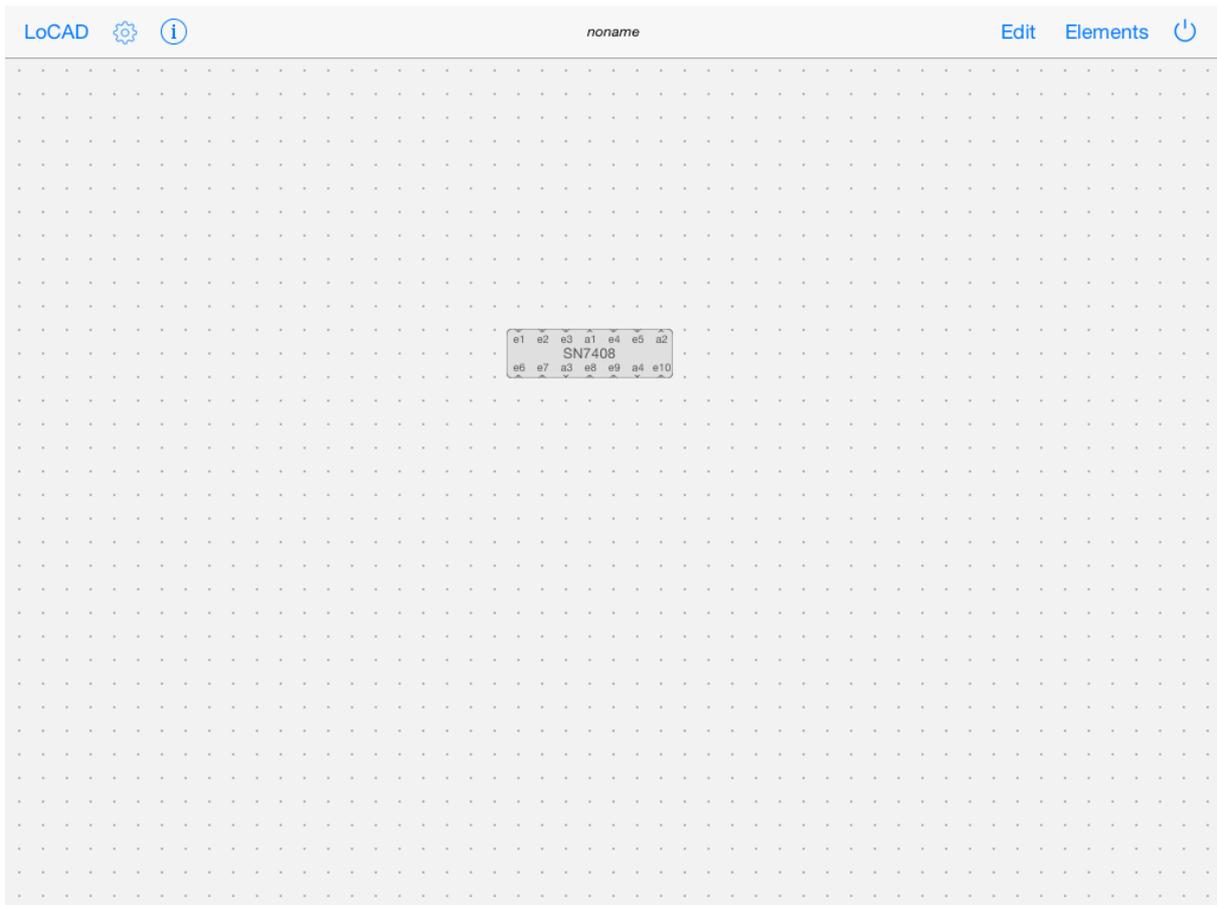


Image 8-5 defined IC element

In order to obtain the same pin labeling as given in the data sheet, you can open a labeling list through **edit** →  and touching the IC with the tool . In this list, you can make the desired changes.

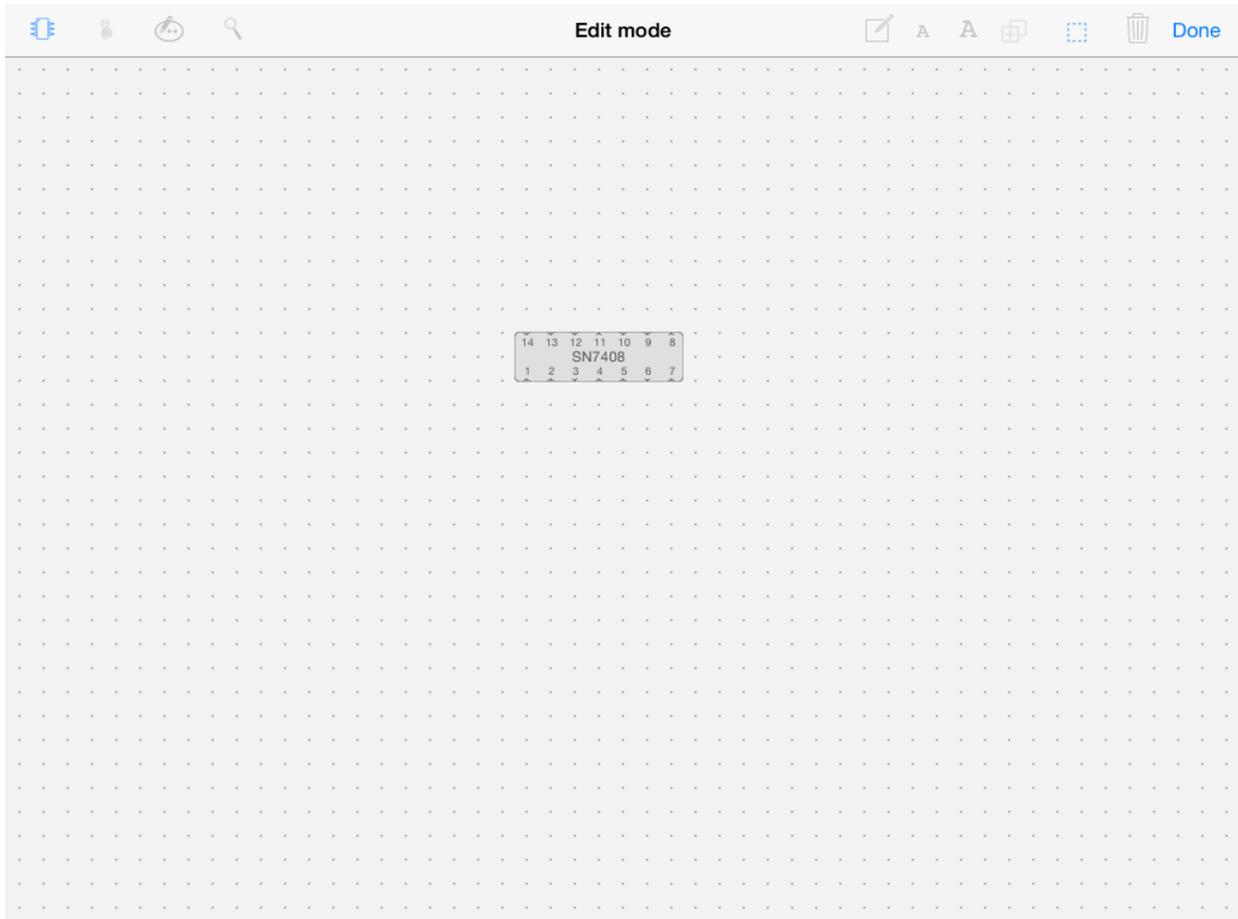


Image 8-6 IC with adapted pin labeling

Now connect a few wires to the IC (see following image).

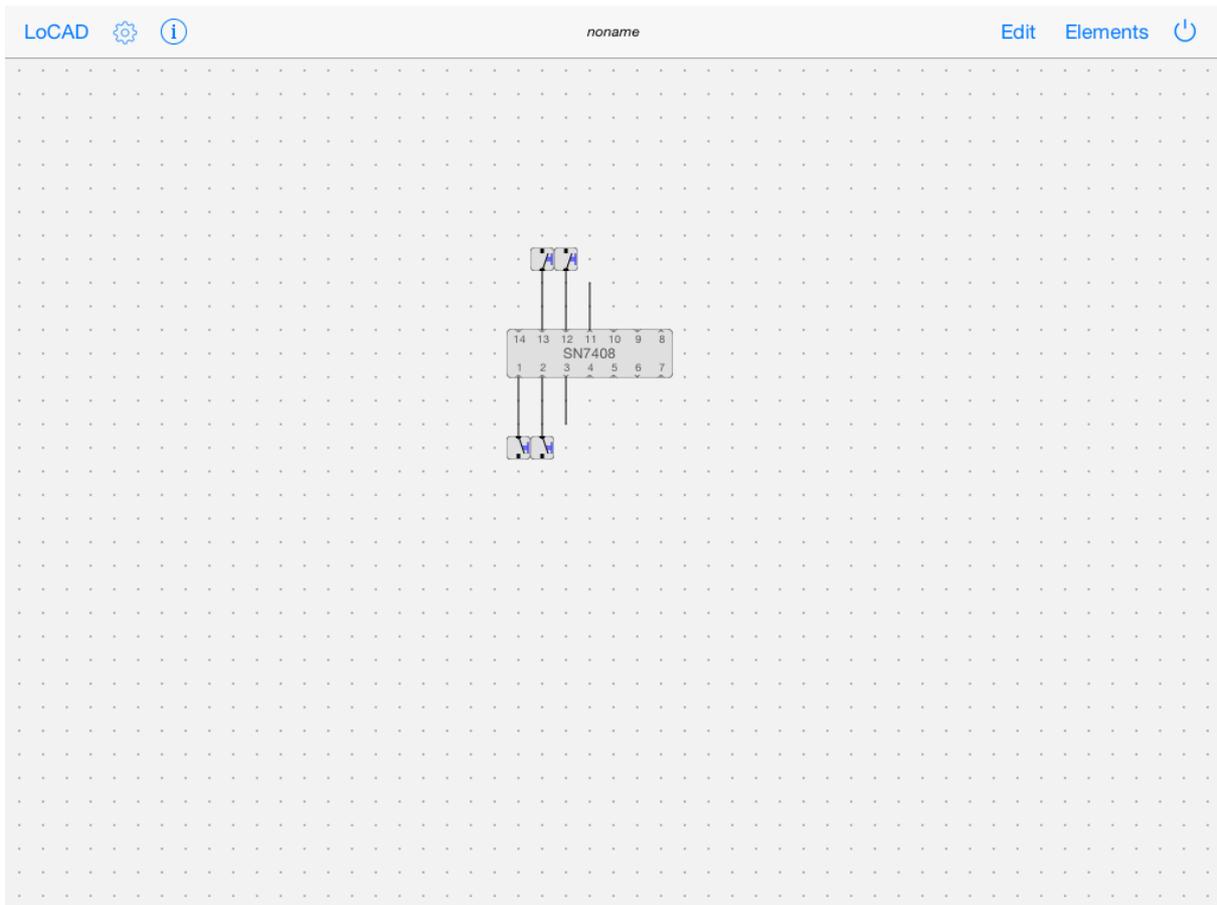


Image 8-7 IC with connecting lines

Operate the circuit with the power switch.

If you put 1-signal on the inputs, you will see the according states at the outputs of the inner circuit.

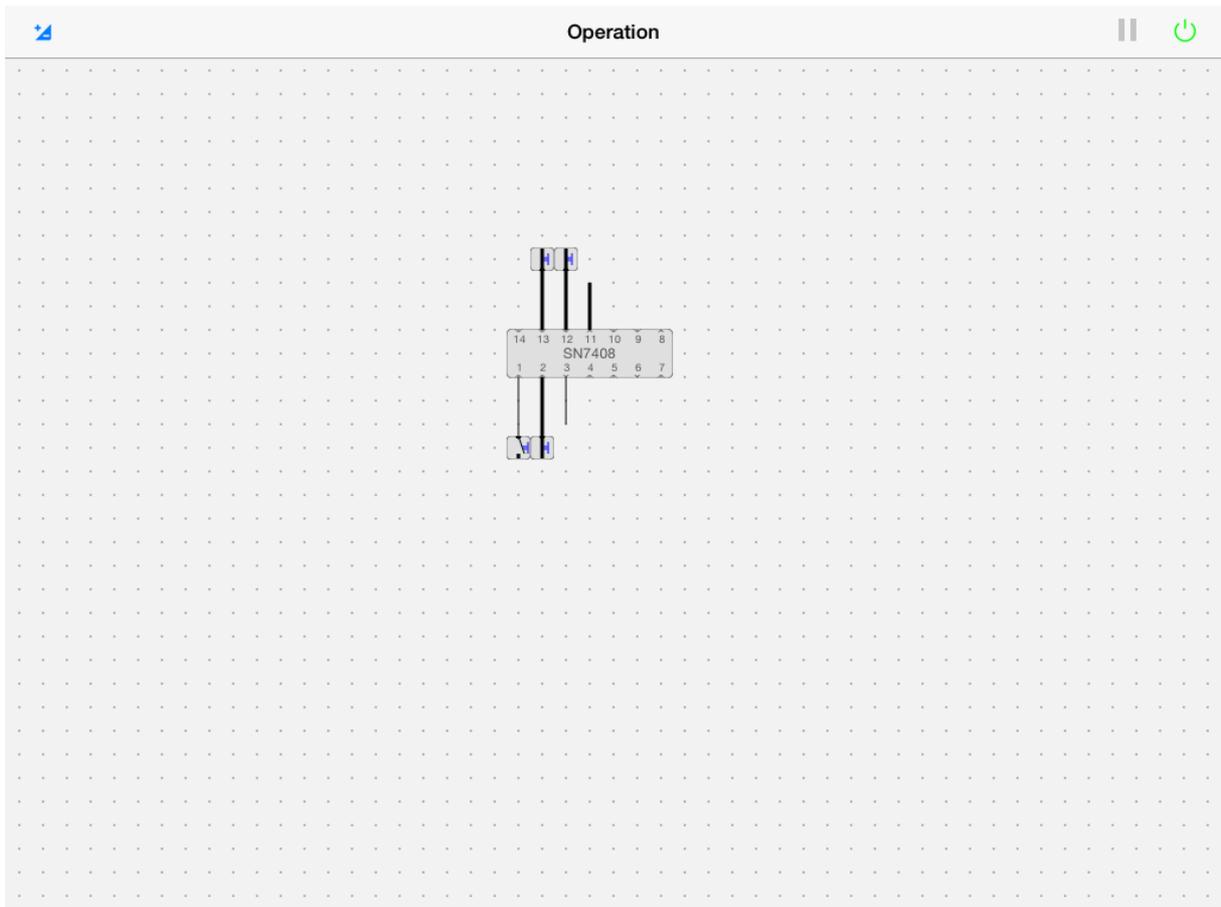


Image 8-8 IC with connecting lines in operation mode

You can watch the state inside the element anytime by touching it.

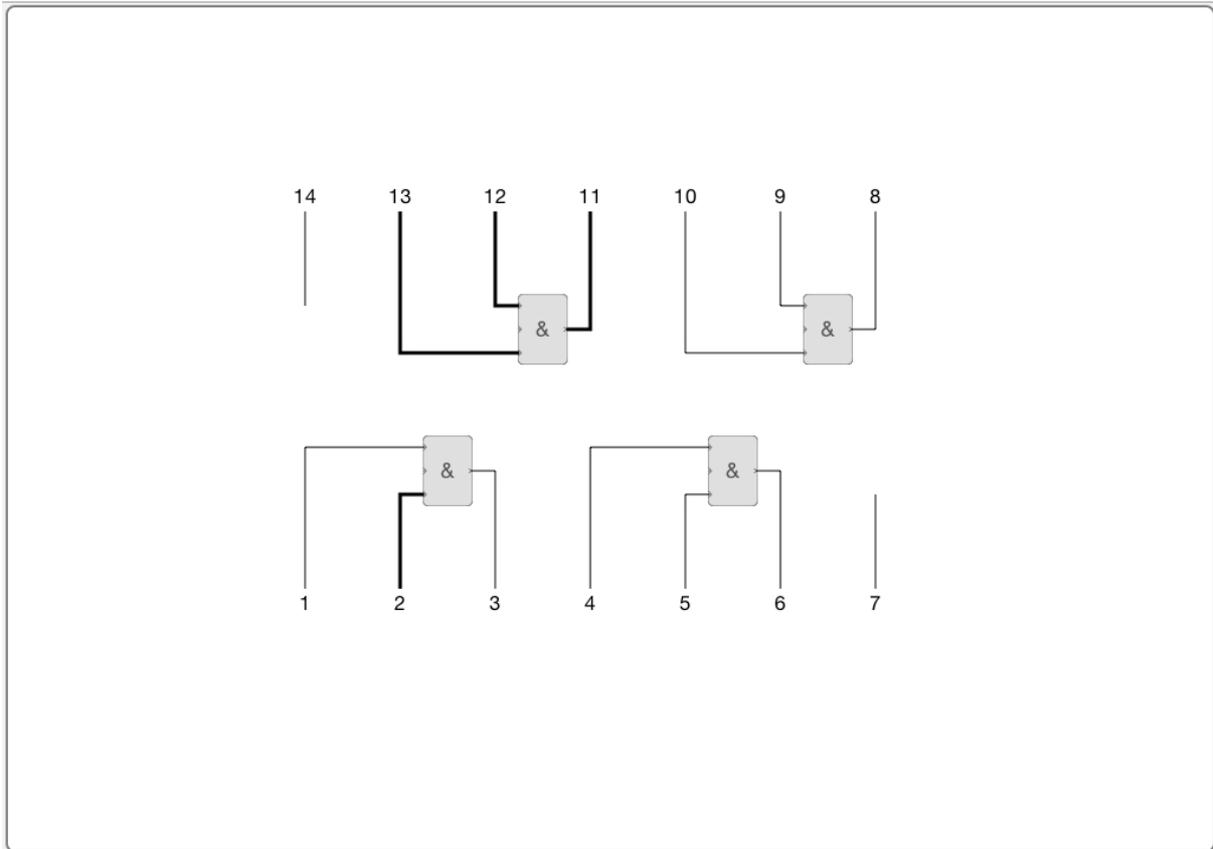


Image 8-9 inner circuit of SN7408 with signals

If an inner circuit contains further ICs, you can open these step by step by touching on them. With the button **< back** you can always go back one step.

In principle, you can always change a once defined element in a way that you load the underlying inner circuit, make the necessary changes and define the element anew.

With the help of the possibility of integration thus shown, we can very well illustrate the concept of modules, resp. the integration principle of modern digital technology. Through nested integration, we can develop extremely complex circuits that still stay manageable.