INTRO | mki x es.edu

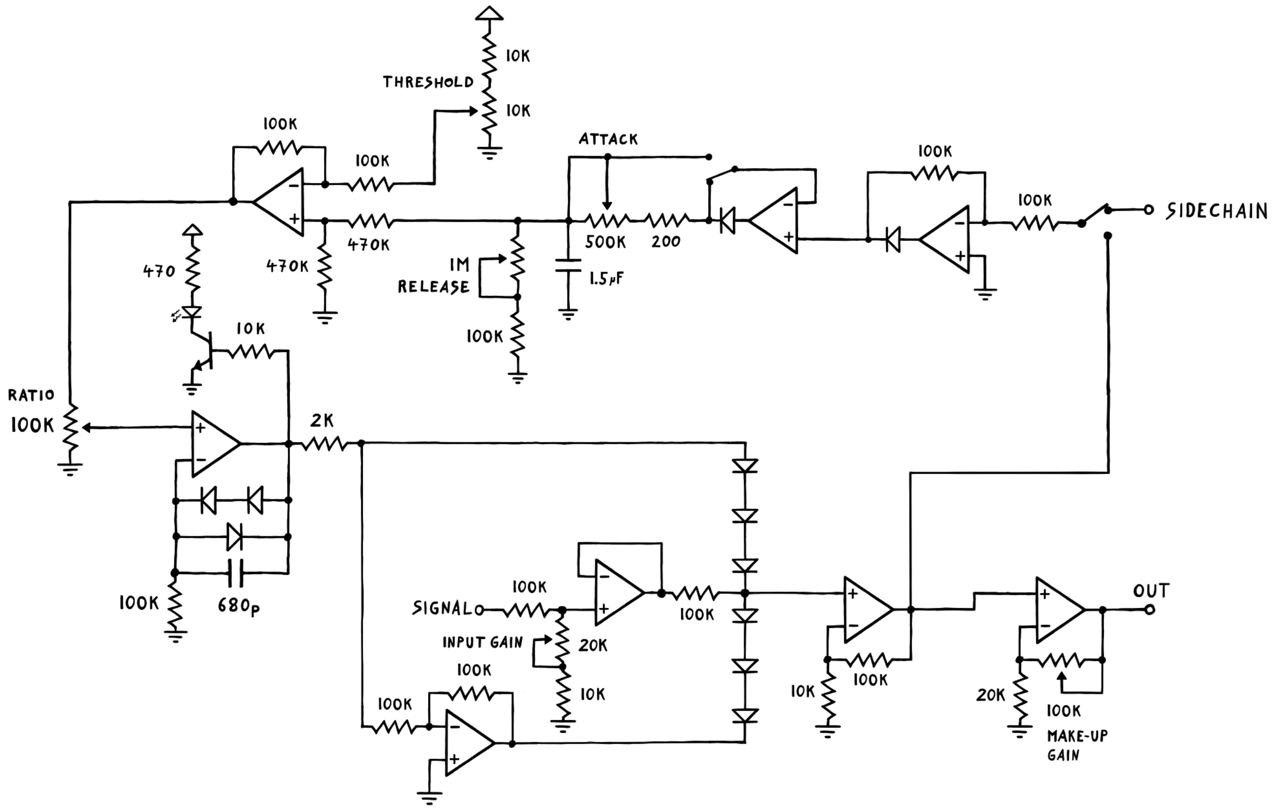
Hey there, thanks for buying this DIY kit! We – **Erica Synths** and **Moritz Klein** – have developed it with one specific goal in mind: teaching people with little to no prior experience how to design analog synthesizer circuits from scratch. So what you'll find in the box is not simply meant to be soldered together and then disappear in your rack.
Instead, we want to take you through the circuit design process step by step, explaining every choice we've made and how it impacts the finished module. For that, we strongly suggest you follow along on a **breadboard[1],** which is a non-permanent circuit prototyping tool that allows you to experiment and play around with your components. To help you with this, we've included suggested breadboard layouts in select chapters.
In addition to this, you can also play around with most of the chapter's circuits in a **circuit simulator** called CircuitJS. CircuitJS runs in your browser. You'll find weblinks in the footnotes which will direct you to an instance that already has example circuits set up for you. We strongly encourage you to fiddle with the component values and general structure of those circuits to get a better understanding of the concepts we're laying out.
Generally, this manual is intended to be read and worked through front to back, but there were a few things we felt should go into a dedicated appendix. These are general vignettes on electronic components & concepts, tools, and the process of putting the module together once you're done experimenting. Don't hesitate to check in there whenever you think you're missing an important piece of information. Most importantly though: have fun!

TABLE OF CONTENTS

[1] Note that there is no breadboard included in this kit! You will also need a pack of jumper wires and two 9 V batteries with clips. These things are cheap & easy to find in your local electronics shop.

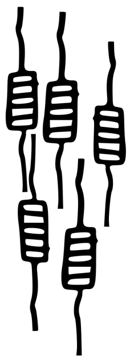
THE mki x es.edu COMPRESSOR

Need some professional polish in your modular system? With this simple diode-based compressor, you can shape the dynamics of your sound, adding punch and consistency to your patches. Whether you're looking to beef up your basslines, add sustain to your leads, or tame the peaks of your percussion, this circuit's got you covered.



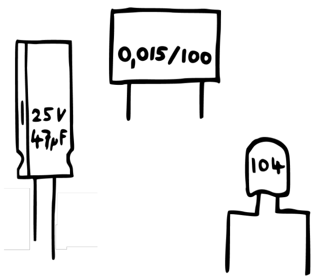
BILL OF MATERIALS

Before we start, please check if your kit contains all of the necessary components. In addition to a PCB, panel and power cable, your box should also contain:



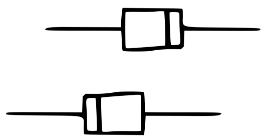
An array of resistors. The specific values (in ohms, which you should check for with a multimeter) are

470k	x2
330k	x1
220k	x1
120k	x1
100k	x11
47k	x1
33k	x4
20k	x1
10k	x4
3k3	x4
2k	x1
470Ω	x 1
200Ω	x 1
10Ω	x2



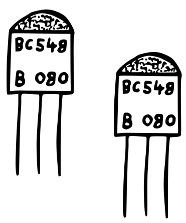
A bunch of capacitors. The specific values (which are printed onto their bodies) are

47µF (electrolytic)	x2
1.5µF (1.5J63/foil)	x3
100nF (104/ceramic)	x8
680pF (681/ceramic)	x1



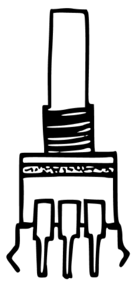
Some diodes. The specific model names (which are printed onto their bodies) are

1N4148 (signal) x13
SB140[2] (schottky) x2



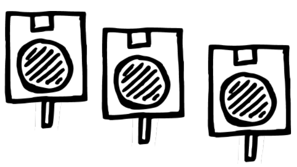
A couple of transistors. The specific model names (which are printed onto their bodies) are

2N3904 (NPN) x5



A handful of regular potentiometers. Their specific values (which may be encoded & printed onto their bodies) are

1M (A105) x1
500k (A504) x1
100k (A104) x2
20k (A203) x1
10k (A103) x1

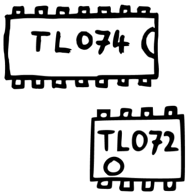


A few jack sockets. The specific models (which you can identify by their color) are

Switched mono (black) x3

---

[2] Please note that these could also be a different model (e.g. 1N5818).



A couple chips. Their specific models (which are printed onto their bodies) are

TL074 (quad op amp)	x1
TL072 (dual op amp)	x2

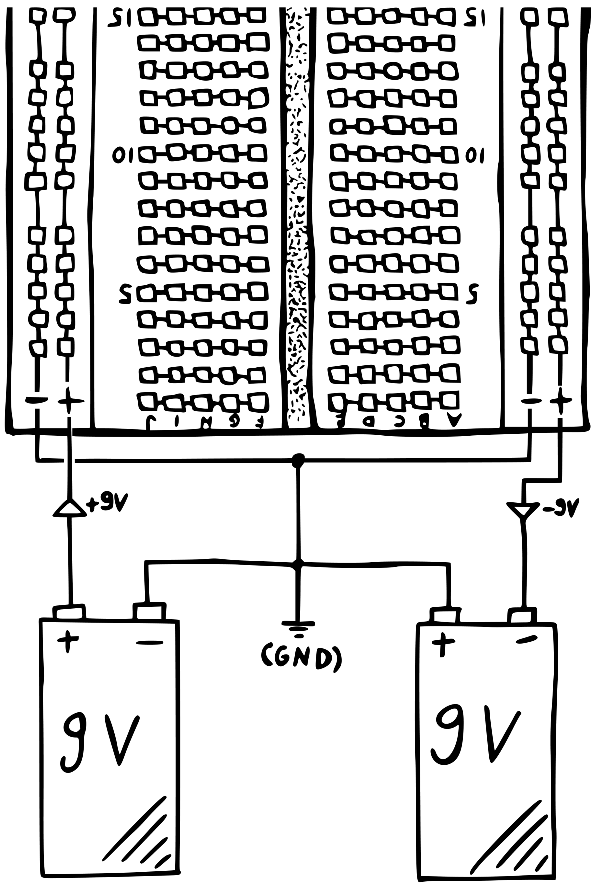
A few LEDs (light emitting diodes). The specific models (which you can identify by measuring their body's width) are

3mm (red)	x2
3mm (green)	x3

You will also find a few sockets that are only relevant when assembling the module in the end.

POWERING YOUR BREADBOARD

Before we can start building, you'll need to find a way of providing your breadboard with power. Ideally, you'd use a dual 12 V power supply for this. Dual power supplies are great – and if you want to get serious about synth design, you should invest in one at some point. But what if you're just starting out, and you'd like to use batteries instead? Thankfully, that's totally doable. **You just need to connect two 9 V batteries to your breadboard like shown here.**[3] For this, you should use 9 V battery clips, which are cheap & widely available in every electronics shop.



By connecting the batteries like this, the row on the left side labeled **+** becomes your positive rail, the row on the right side labeled **+** becomes your negative rail, and both rows labeled **–** become your ground rails.[4]

Please make sure you disconnect the batteries from your breadboard when you make changes to the circuit! Otherwise you run the risk of damaging components.

[3] Since the circuits in this manual were designed for a 12 V power supply, we assume that to be the default. Everything will still work roughly the same with 9 V, though.

[4] This is a bit awkward because breadboards weren't really made with dual supply voltages in mind.

COMPRESSION BASICS

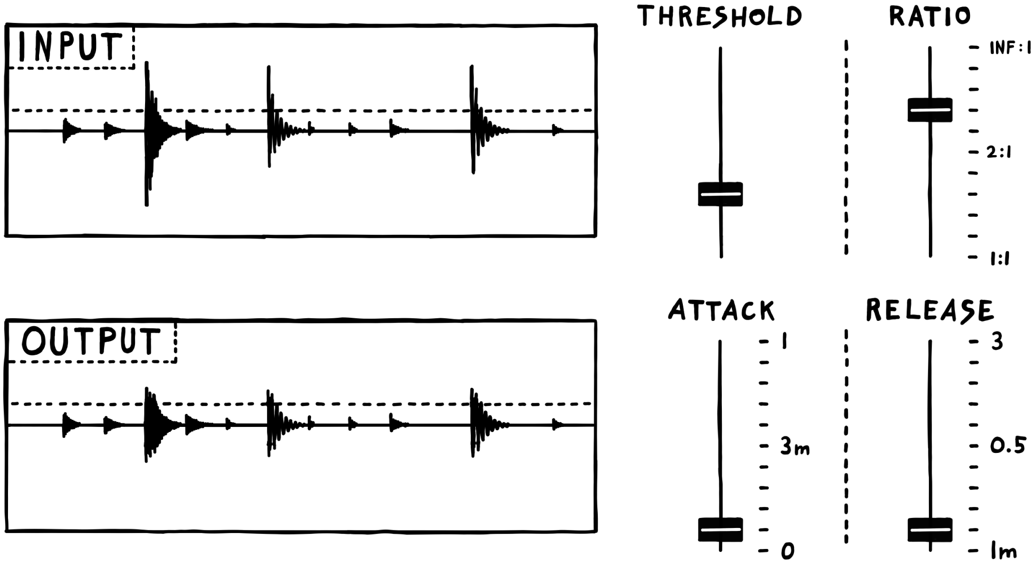
Compression is an effect that I personally only know from my DAW. I've never touched a hardware compressor, much less an analog one. Which made it all the more intriguing to try and design one myself. Since I never bothered to really understand the plug-in I was using (I just tweaked the settings until I liked what I heard), the obvious first step was to take a closer look at what it actually does to a waveform.
For that, I recorded a beat from my drum machine, which you can see here as a visual representation.

UNCOMPRESSED

Unfortunately, the levels are all messed up – the kick and snare are way too loud, while the hats & toms are super quiet. This is visible in the waveform itself: these big, wide bumps are the kick and snare, while hats and toms only show up as tiny blips. And here's what that same clip looks like after I've run it through the compressor.

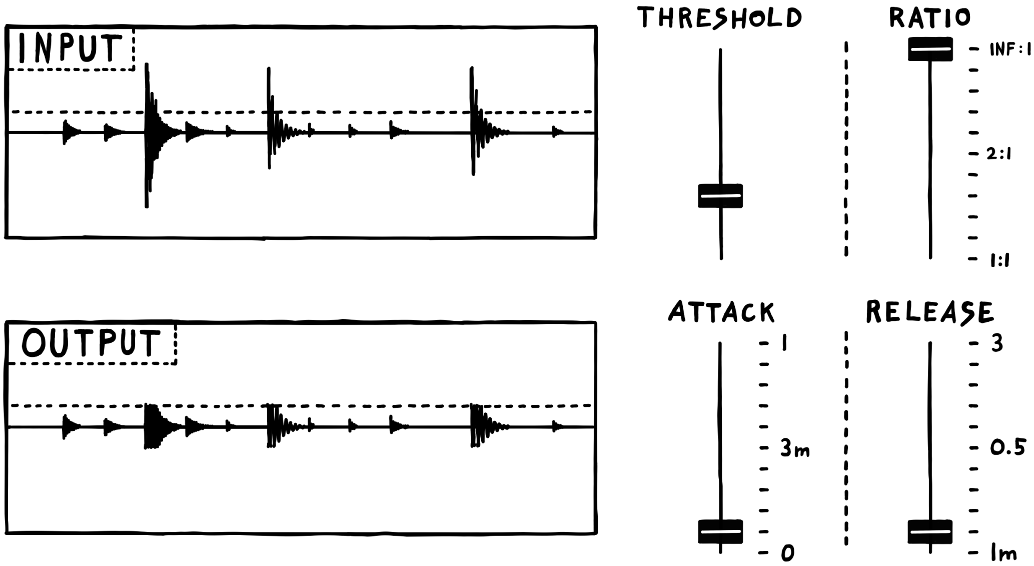
COMPRESSED

Suddenly, the kick and snare aren't that loud anymore. You can clearly see this in the waveform: the bumps are much smaller than before. Curiously, the small blips stayed the same volume, though. So it's not like we simply reduced the overall volume of the clip.
Instead, we somehow specifically reduced only the volume of the kick and snare hits. Making our beat's mix much more balanced as a result.
Cool, but how does this work? How does the compressor know which parts of the waveform are too loud – and how does it turn them down selectively? **The magic control for this is the compression threshold, which it uses to decide if a sound is too loud, and subsequently, whether it needs to turn down the output volume**. If you're having trouble visualizing this, here's a quick illustration.



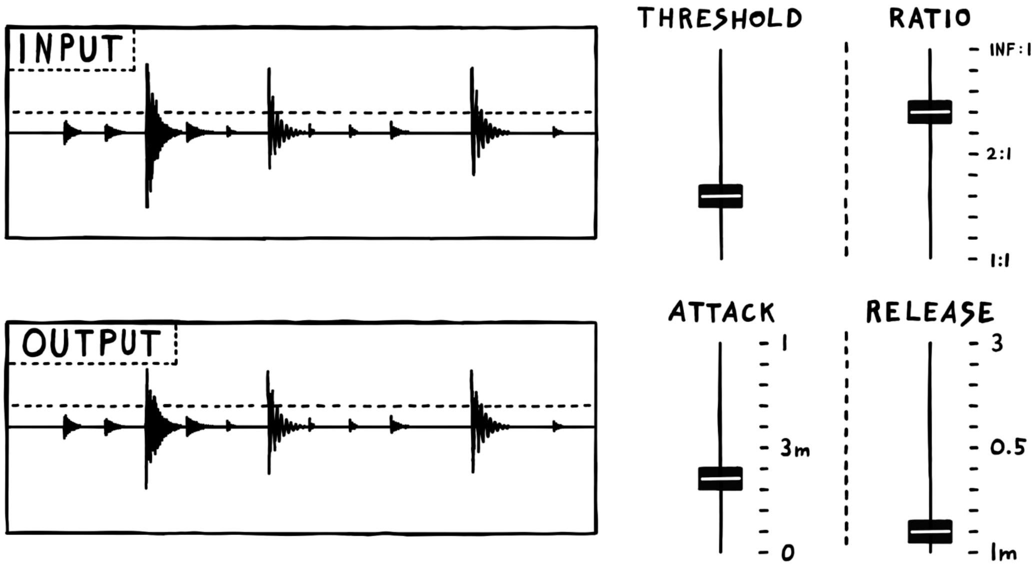
In the box up top, we have our input waveform and the original compression threshold – and below it, the compressor’s output. As you can see, all the parts that cross the threshold in the input are significantly reduced in volume in the output. Though interestingly, they still cross it in the result!

Why is that? **Because of the compressor’s ratio setting, which tells it by how much it should reduce the volume when the input crosses the threshold**. In the example above, that setting is dialed to a pretty moderate level. That’s why the compressor doesn’t simply cut off the signal at the threshold. If we crank it up all the way, though, that’s precisely what happens: the threshold turns into a brick wall that the output cannot cross.

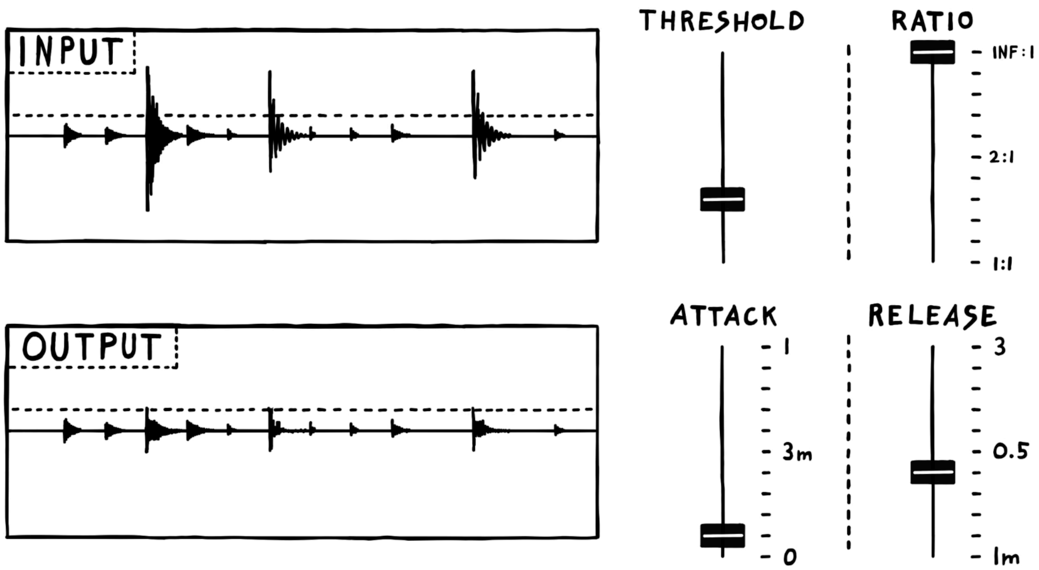


There are two more controls that I regularly use when working with a compressor. **First up, there’s the attack time, which tells the compressor how long it should wait**

before reducing the output volume. In our examples above, it was turned down completely, so the volume was reduced the instance the signal crossed the threshold. Let's turn it up and see what happens.



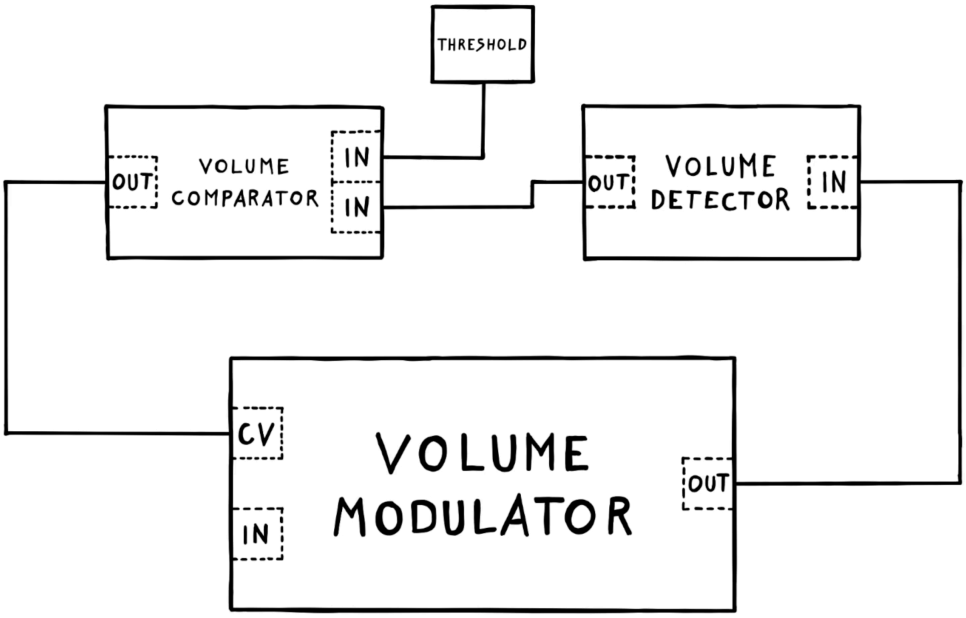
As expected, the compressor now lets the signal push beyond the threshold for a moment before proceeding to squash it.
Finally, let's look at the release control, which is basically the symmetrical counterpart to the attack control. **It tells the compressor how long it should wait before restoring the output volume.** In our previous examples, it was turned down completely. So the volume was restored the instance the signal fell back below the threshold. Let's try bump this one up, too.



As you can see, the parts of the waveform that directly follow the big bumps are now affected – even though they don't cross the threshold.

COMPRESSION THRESHOLD

Cool! So for a very basic little compressor circuit, these are the four features I'd definitely want to replicate: compression threshold, compression ratio, attack and release. To make our lives a bit easier, we'll start by singling out the compression threshold. Conceptually, here's how I think it could be implemented.



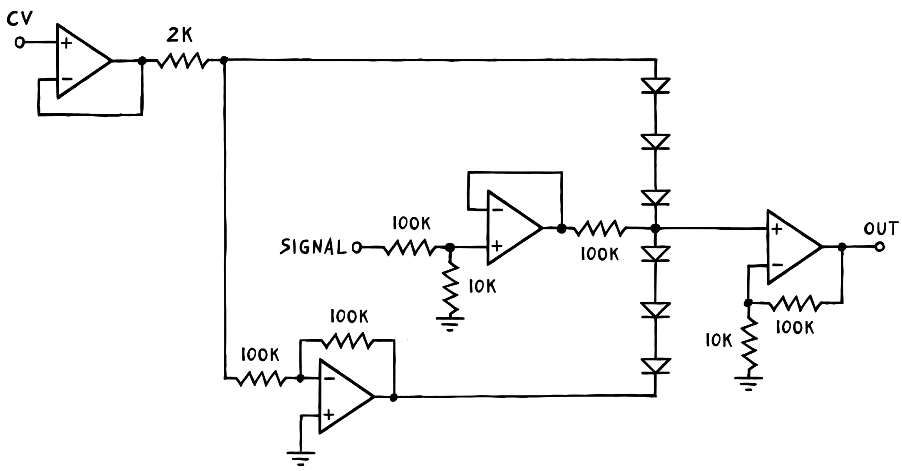
First up, we need a block that allows us to manipulate the volume of our input signal. Since this has to happen automatically, we'll want a solution that can be controlled with a voltage. Next, we need to know how loud the output of that first block currently is. This might sound like a pretty simple question, but it's not exactly trivial to answer. We'll get to that in a bit.

After this, we need to compare the current volume level to a threshold and determine if we're above or below it – and by how much. Finally, we want to use this information to control the output volume of our first block. This way, we establish a feedback loop that works something like this: **At any given moment, we ask: how loud is the output signal? Then: is it louder than the set threshold? If yes, then reduce the output volume. If no, leave it as-is.**

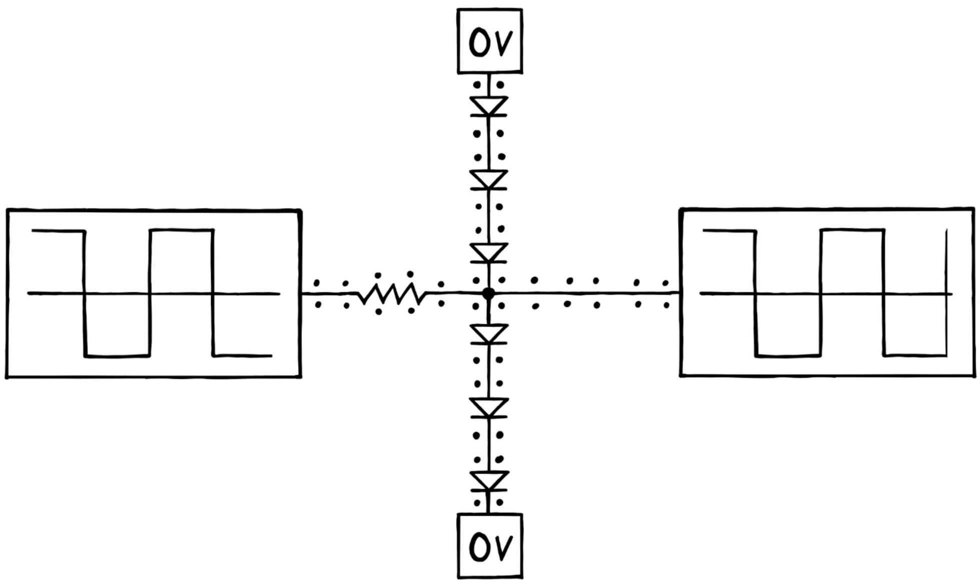
This way, we'd get something akin to the brick wall compression we've seen earlier –– where the compression threshold is basically a hard volume ceiling. (This is also called limiting, by the way.) Cool! Now, in order to turn this idea into an actual circuit, we'll need to come up with implementations for these four individual blocks. Let's start with volume modulation.

DIODE-BASED VCA

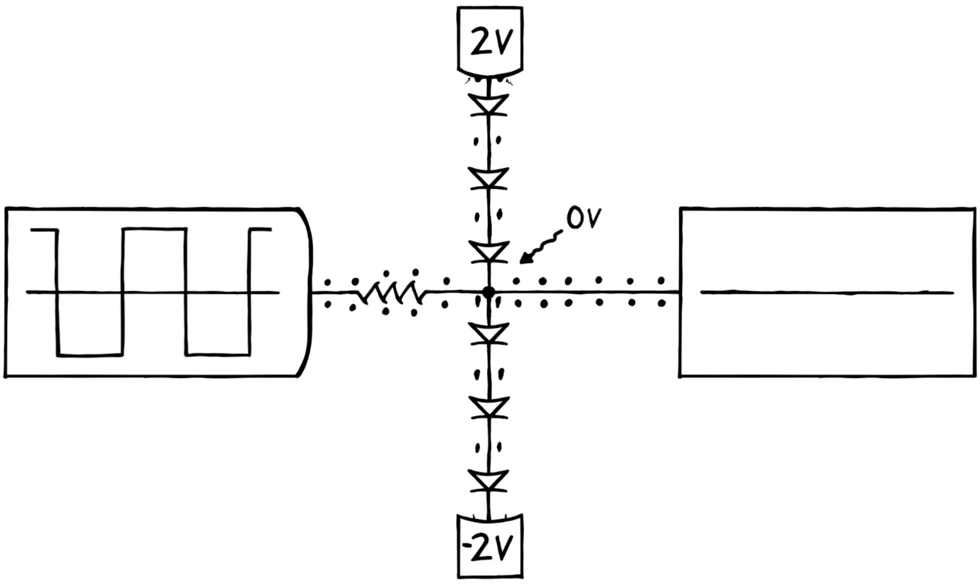
If you've already built a couple of our DIY kits, you might already be thinking in a certain direction here. As was I – which is why I decided to go with a VCA-based solution. **VCA, if you don't know, is short for voltage controlled amplifier**. It does exactly what we're looking for: adjusting the volume of an input signal in response to a given control voltage. Thankfully, I did already come up with a couple VCA designs, so we can just use one of those. My diode-based implementation seems like a great fit, as it is really quite simple. Here's what it looks like after I tinkered with it a little to make it fit our specific needs.



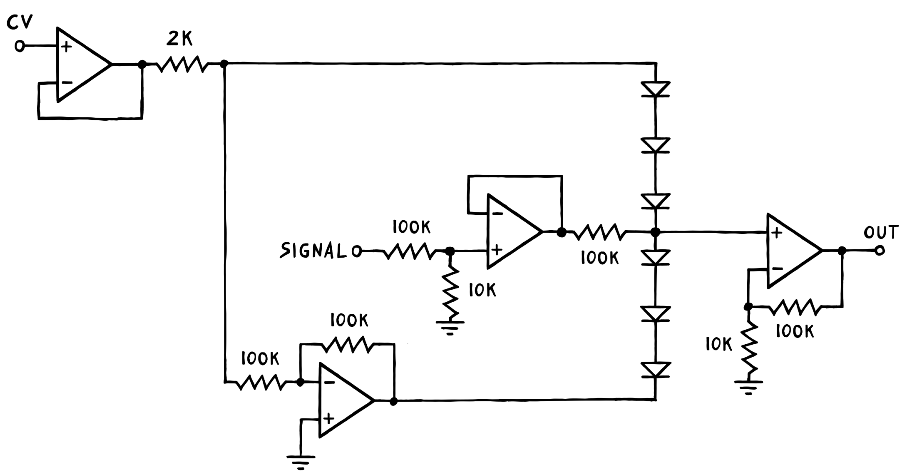
And while this might seem like a lot at first glance, the functional core is just this string of six diodes with a 100k resistor attached to its central node. Here's how it works. We apply a scaled-down version of our input signal to the resistor, the control voltage to the top of the diode string – and an inverted copy to the bottom.



The output can then be picked up from the central node. **If the control voltage is 0, the output will be identical to the input**. That's because there is no current flowing: neither through the diodes, nor through the resistor, since the input is not strong enough to push or pull the diodes open. And if a voltage is not allowed to transform into current, it's preserved as-is. This is why we need to scale the input down, by the way. Because if it's too loud, it opens the diodes in this scenario, which would result in noticeable distortion.
Okay, now if we bump the control voltage to 2 V, the diodes are wide open and a large current flows from top to bottom. Since the diode string is essentially a 50% voltage divider, the central node will sit at 0 V – exactly halfway between top and bottom voltage.



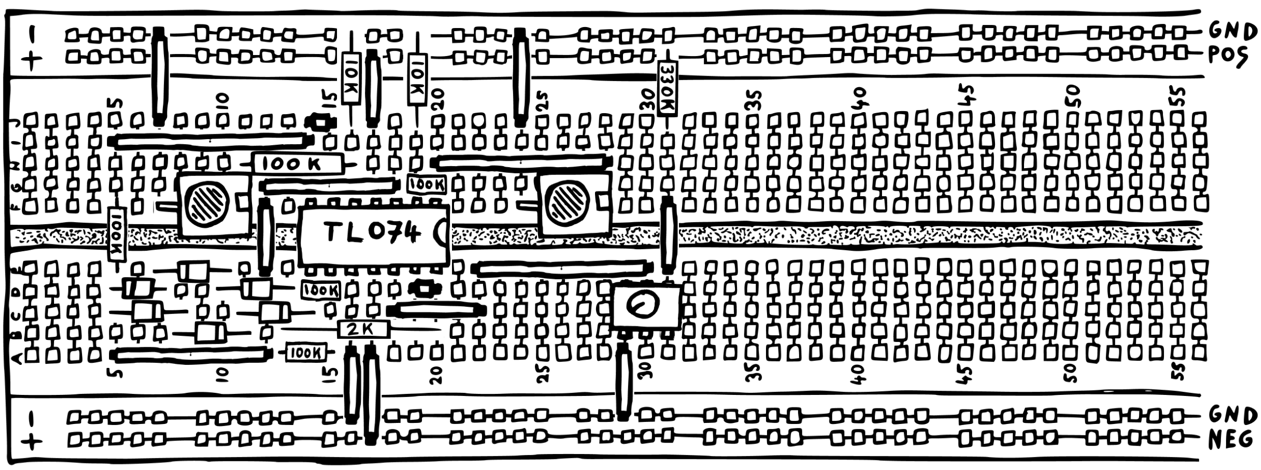
Now, if the input signal goes above or below 0 V, a current flows through the resistor. But since that current is so much smaller than the one flowing through the diodes, the central node voltage will remain almost unaffected. The signal is basically washed away, if you will. And the output stays silent. **So in summary: the more current flows through the diodes, the more we reduce the output volume.** Looking at the circuit as a whole again, we can see that everything else is just there to support this mechanism.



First, we scale the input signal down with a 100k/10k voltage divider. Then, we buffer it and apply it to the 100k resistor. Next, we buffer the incoming CV before inverting it and then applying both versions to our string of diodes. (The 2k resistor after the CV buffer is just to limit the maximum amount of current we push through the diodes. Wouldn't want to overdo it, right?)

Finally, we pick up the output from the central node and then amplify it to reverse the downscaling at the input. If you've been wondering why we use a string of six diodes instead of just two – here's the reason: w**e want to keep the gain of the output amplifier as low as possible to prevent random noise from creeping in**. Multiple diodes in series are much harder to open, which means that we can apply a louder input signal without risking distortion. In turn, the output stage doesn't need as much gain to restore the signal.

And that's it. Now by increasing the CV, we can decrease the output volume – which is exactly what we were looking for.[5] Great! So grab your breadboard and put this to the test.

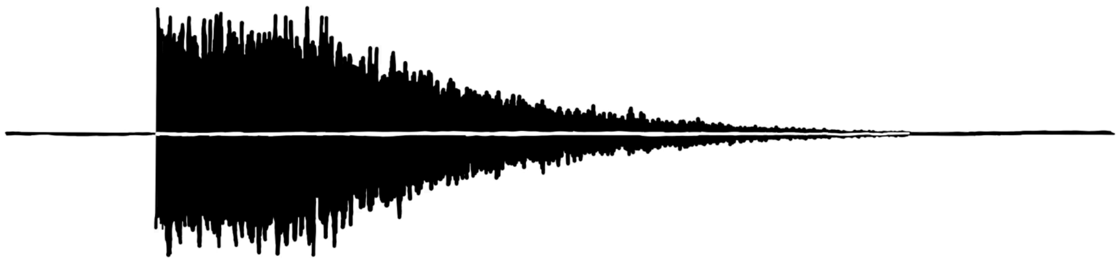


If you now send any audio signal (oscillator, drum machine, …) into the signal input on the left, you should be able to change the output volume by turning the 100k potentiometer's knob, which produces a control voltage between 0 and 3 V.

[5] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

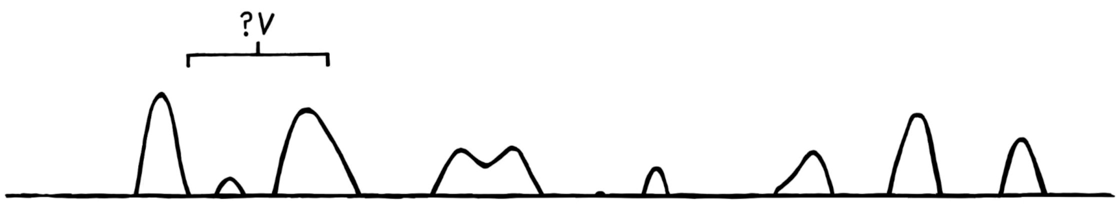
SIMPLE PEAK DETECTOR

Great, so we've got the first block down. Next, we'll tackle the volume detector. For that, we'll first have to define what we mean when we talk about volume in this case. Here's what a single snare drum hit looks like when viewed on an oscilloscope.

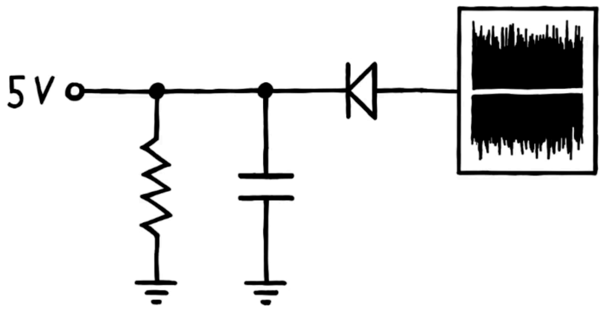


The line in the middle marks ground level – 0 V. As is typical for audio signals, the snare oscillates around that line. Now, strictly speaking, you'd determine the volume of this sound by measuring the height of the waveform at any given point, top to bottom. But since audio signals, this one included, are often roughly symmetrical, we can take a little shortcut here.
Instead of measuring top to bottom, we'll simply discard the lower half and treat the distance between the current peak and ground as the absolute volume reading for now. This will greatly simplify the first iteration of our circuit.

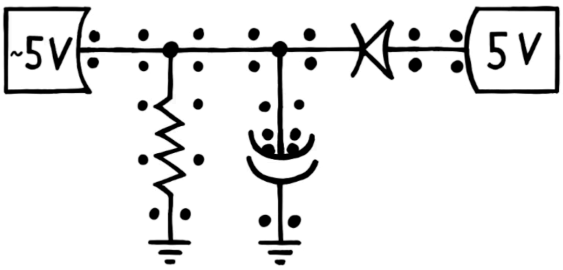
Okay, so that's the y-axis. We also need to talk about the x-axis, though. That's because the answer to the question "how loud is this sound currently" depends heavily on what we mean by "currently". At this scale, for the first fifth of the waveform, the approximate answer would be 5 V. But if we zoom in on the x-axis by a lot, that answer changes.



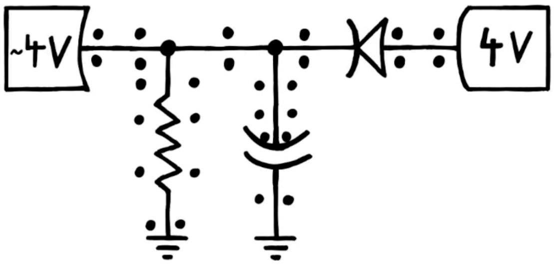
Because now, we get wildly different volume levels directly next to each other that we can't simply round up to an approximate value. That's of course due to the fact that sounds are oscillations. **And it means that in order to get a useful answer to our question, we have to measure the sound's volume over a significant period of time**. How do we pull that off? Easy: with a circuit called a peak detector, which consists of just three components: a diode, a capacitor and a resistor.



If you set them up like this and send your audio signal into the diode, you can pick up its current volume level on the other side. Here's how it works. Whenever the input rises significantly above the 0 V-line, a large current is pushed through the diode and into the capacitor, filling it up. **Once the voltages on both sides are (almost) the same, the current flow will stop**. We've sampled the current volume level.



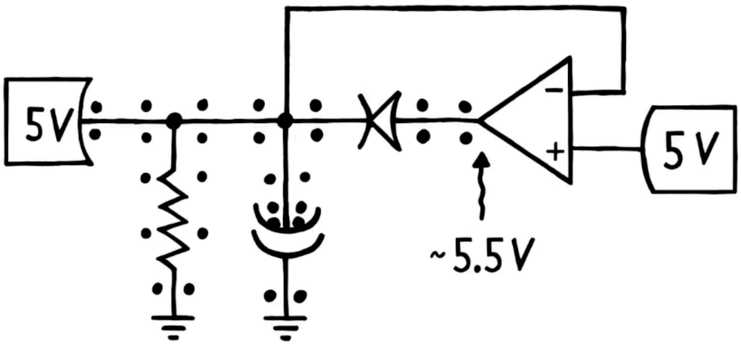
Next, when the input voltage drops, the charge inside the cap will very slowly drain out through the resistor. This is how we measure the voltage level over an extended period of time.



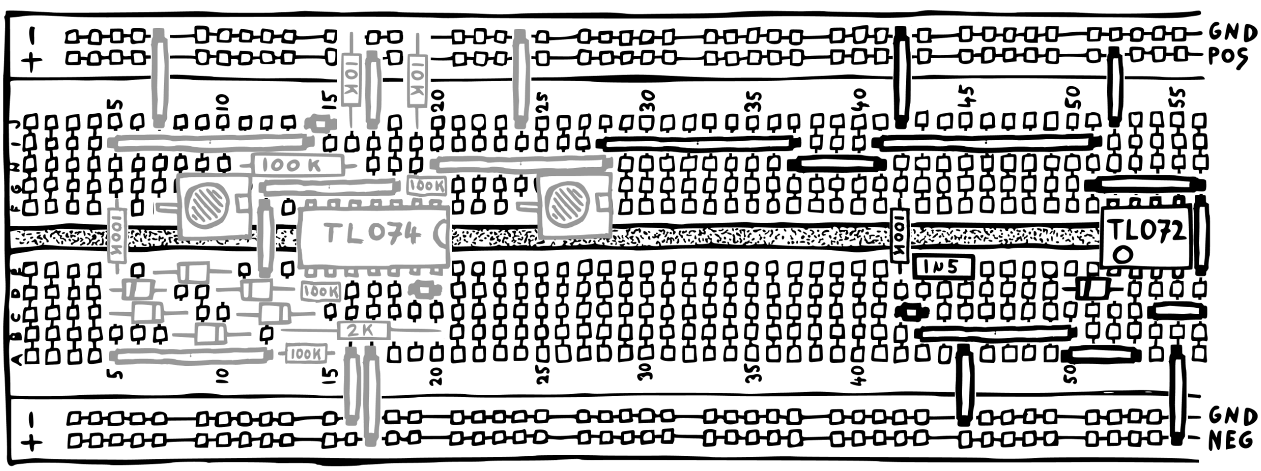
Because if the input signal is returning to the same peak in rapid succession, the voltage at our capacitor will stay roughly constant. And once that peak drops for a while, the excess current will drain out and the voltages equalize. For this to work properly, we'll need to find the right values for both the cap and resistor, though. If they're very small, the time interval we're looking at will be very short – and vice versa. In my experiments, combining a 100k resistor with a 1.5 µF capacitor gave me the best results.

Still, if you'd set this up on the breadboard and compared the peak detector's output to the input waveform using an oscilloscope, you'd notice that the volume we detect is a good deal too low. Why is that? Well, we can actually blame the diode here. Because as we saw earlier, diodes will only let current pass through if the input voltage is significantly higher than the output voltage. This means that our capacitor can never be charged to the input signal's actual peak level. Simply because the current flow will stop prematurely.

To fix this, we can make use of a nifty trick involving another op amp. If we combine it with our diode like this, we get rid of the gap between the signal's peak and the capacitor voltage.



That's because the op amp will simply increase its output voltage until the two input voltages are equal, neutralizing the voltage drop across the diode in the process. Great! So let's set this up on the breadboard and see how we fare.[6]



[6] You can try this chapter's circuits in a simulator. I've already set them up for you right here. You can change all values by double clicking on components.

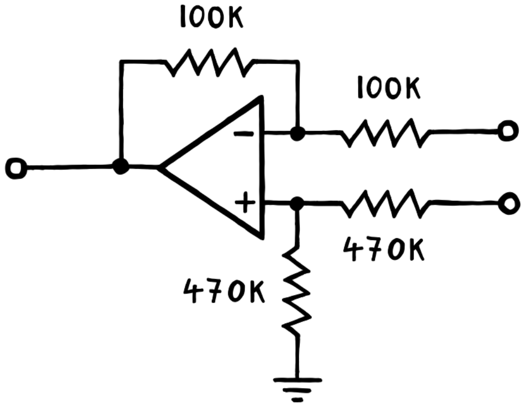
To test this properly, you'll need a 2-channel oscilloscope. Connect one channel to the VCA's output, and the other to the peak detector's output. Apply the same vertical & horizontal scaling to both, and you should be able to see that the waveform's peaks are now properly traced by the peak detector.[7]

[7] If you don't have a 2-channel oscilloscope, you can watch me do this experiment here.

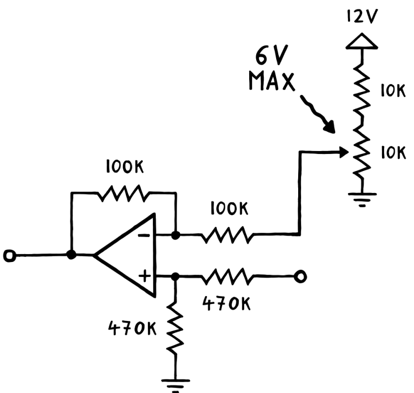
VOLTAGE SUBTRACTOR

So that's the second block down – just one more to go! Here, we want to compare the detected volume level to a variable threshold and decide if it's above or below it – and by how much. This sounds more tricky than it actually is, though. Because all we need is simple subtraction. **If we take the detected volume and subtract the threshold from it, the result will tell us all we need to know**. A positive result means we're above the threshold. A negative one means we're below it. With the absolute value telling us by how much in both cases.

Now, how do we implement this? Thankfully, voltage subtraction is yet another operation that op amps can help us with. Even if the setup is a little more involved than, say, a simple buffer.



Since I've already explained the mechanics behind this circuit in detail in a different manual, I'll skip it here. You can find a link to that manual in the footnotes.[8] The basic gist is this, though: **the op amp will subtract the voltage at the inverting input from the voltage at the non-inverting input and then set its output to the result of that calculation**. So all we need now is a variable threshold voltage.



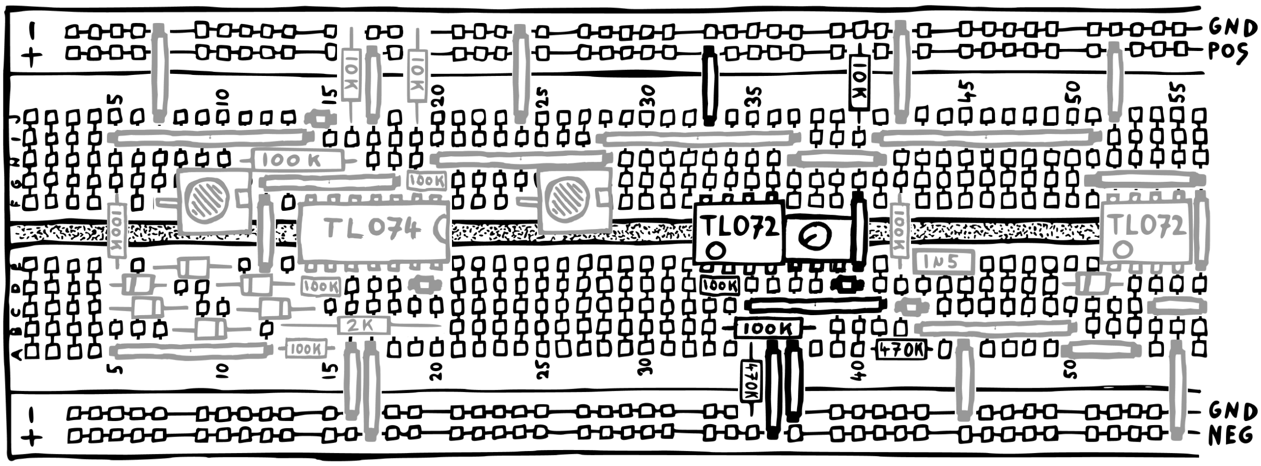
For that, we'll simply set up another potentiometer as a variable voltage divider.[9] Since we don't expect our input signal peaking at anywhere near 12 V, it makes sense to restrict its range somewhat. Otherwise the knob would have a huge dead zone.

If we put a 10k resistor between the potentiometer and the positive rail, we pin the maximum threshold to 6 V. Which should give us a comfortable amount of headroom, considering that the highest expected value in a eurorack synthesizer is 5 V.

[8] The manual is available here, and the explanation begins on page 25.

[9] The threshold is effectively setting a volume ceiling, so we should use a logarithmic potentiometer to make it feel natural/linear.

Great, so let's see how this pans out when we set it up on the breadboard.[10]



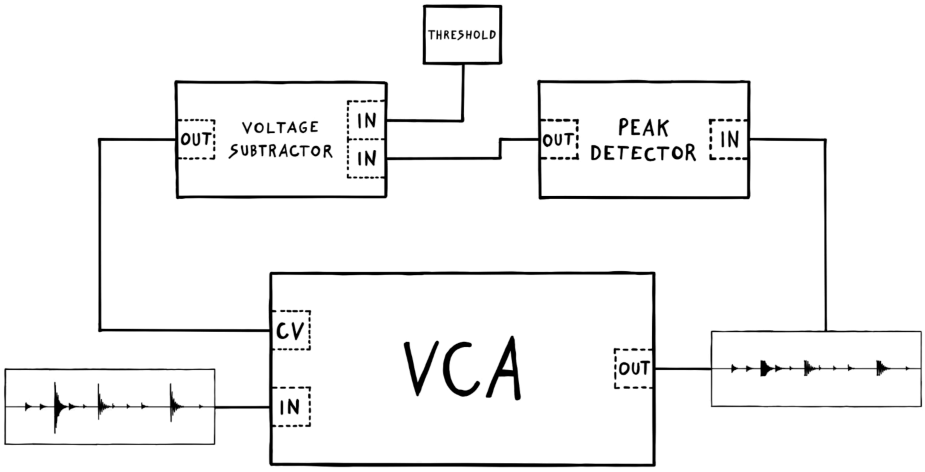
You'll again want to monitor both the VCA's output and the voltage subtractor's output on two separate oscilloscope channels.[11] Dial in a relatively low threshold, and you should get positive bumps whenever the input signal crosses it – and negative ones when it doesn't.

[10] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

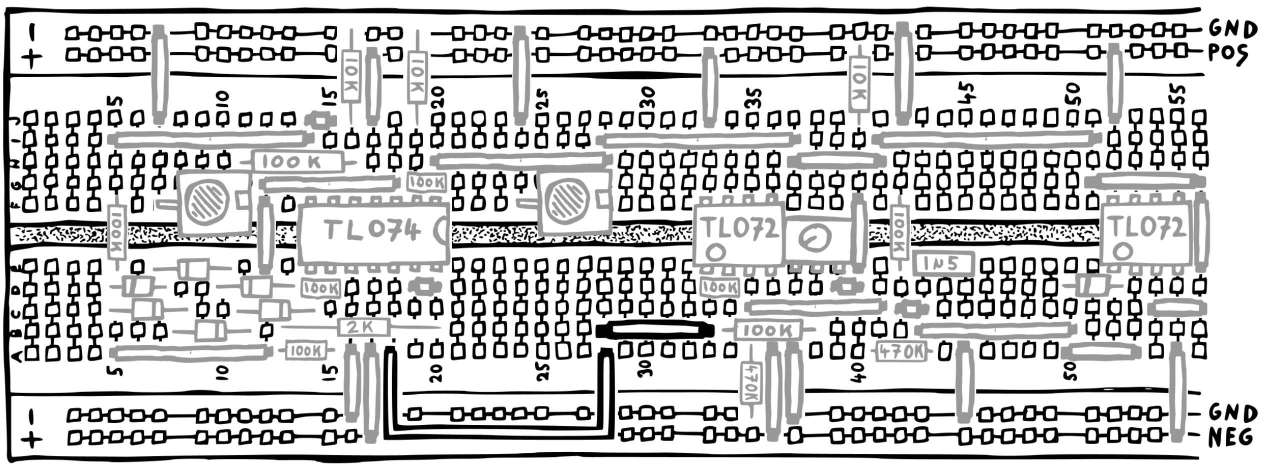
[11] Again, you can watch me do this here if you don't have a 2-channel oscilloscope.

CLOSING THE LOOP

So that's the final block done. Now we just need to send the subtractor's output into the VCA's CV input.



Because then, whenever the input signal crosses the set threshold, the subtractor will send out a positive voltage, which in turn reduces the VCA's gain. Conversely, if the output is below the threshold, the subtractor sends out a negative voltage – which allows the input to pass through the VCA unchanged. **That's because current cannot flow through the VCA's diode string in reverse – so it basically just ignores negative CV inputs.** Great! Let's see if this works as expected.[12]



Ideally, you'll want to monitor both the compressor's output and the set threshold voltage on two separate oscilloscope channels.[13] That way, you should be able to see that lowering the threshold does indeed reduce the volume of big peaks. Though weirdly, they

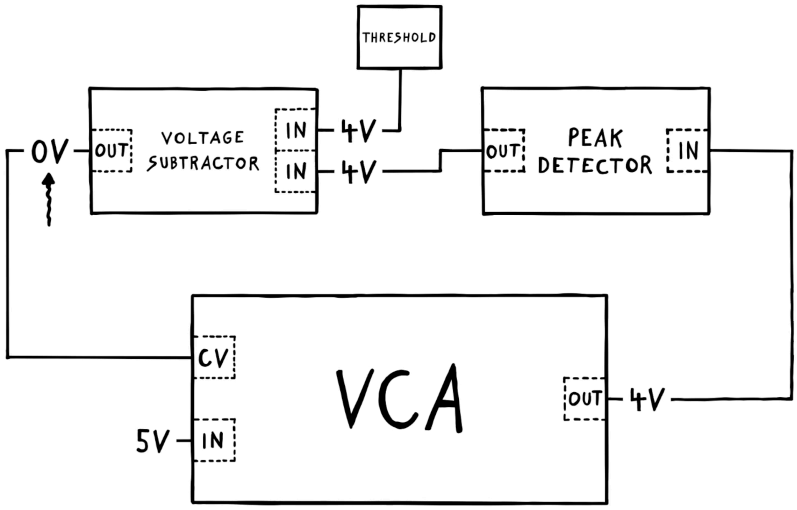
[12] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components

[13] You can again watch me do this here if you don't have a 2-channel oscilloscope.

won't be capped at the threshold. Instead, you'll see a prominent gap between the threshold and the peaks. What's up with that?

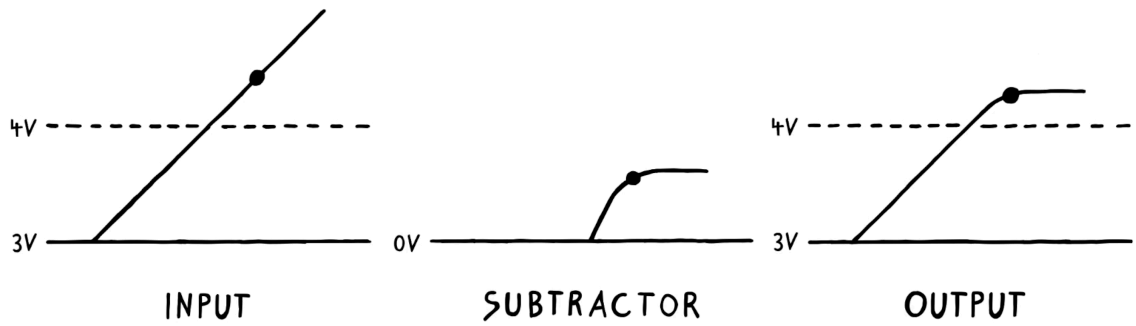
CLOSING THE GAP

Well, this is (again) the diodes' fault. To understand why, let's analyze the feedback loop we created here. We'll assume that the signal input is fixed at 5 V, while the threshold is set to 4 V. Now, what we'd hoped to see is that the circuit reduces the VCA's gain until the output is equal to the threshold. But there's a rather glaring issue with that.



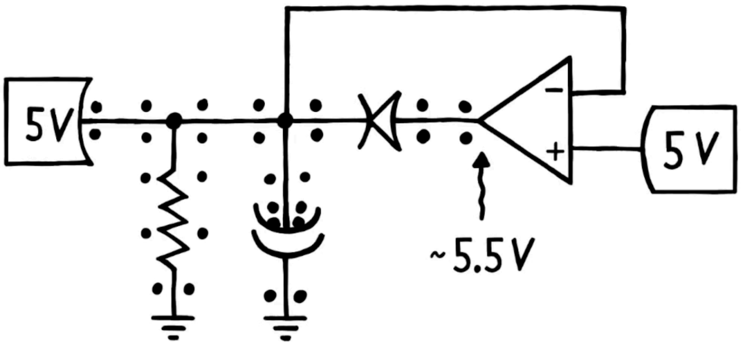
Because if the currently detected peak was the same as the set threshold, our subtractor's output would be 0 V. Which, in turn, means that the VCA just allows the signal to pass through unchanged. But this clashes with our previous assumption. If the output volume is not reduced, it's obviously not equal to the 4 V threshold.

So what happens in reality is that the feedback loop settles on a kind of compromise: where the difference between detected peak and threshold is just big enough to drive a substantial amount of current through our VCA's diode string. If that sounds confusing, let's do a little thought experiment. For that, we'll plot the VCA's input, the subtractor's output, and the VCA's output side by side.

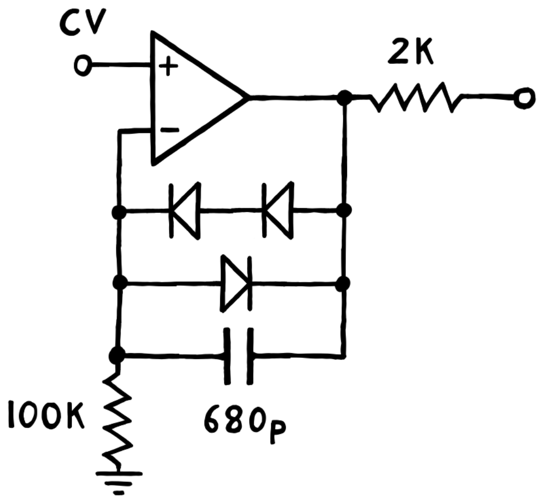


Let's say we slowly raise the input voltage from 3 to 5 V, while the threshold stays fixed at 4 V. As soon as the input crosses the 4 V-line, the subtractor's output starts going positive. But since the diodes in our VCA don't conduct at very low CV levels, there won't

be any impact on its output voltage. Only once the subtractor's output is high enough to drive a significant amount of current through those diodes will we see an effect on the VCA's output. And from that point on, it quickly stagnates – along with the subtractor's output. Here, we've reached the compromise I mentioned before.
So this explains the gap between threshold and signal we saw on the oscilloscope. But is there anything we can do about it? Thankfully, there is: **we just have to make sure that our VCA's diodes turn on instantly when the CV input starts going positive**. For that, we can fall back on what we've learned last time a diode gave us trouble.



Remember how we put this diode inside the buffer's feedback path to neutralize the voltage drop across it? This worked because the op amp has to turn on the diode before the voltage on the other end can rise by any significant amount.
We can apply the same idea to our VCA's CV input buffer and force it to turn on the entire diode string almost instantly. Here's how that works. We add in a string of two diodes followed by a 100k resistor to ground between the buffer's output and its inverting input. Also, we'll add a single diode facing the other way, and a tiny 680 pF capacitor in parallel.

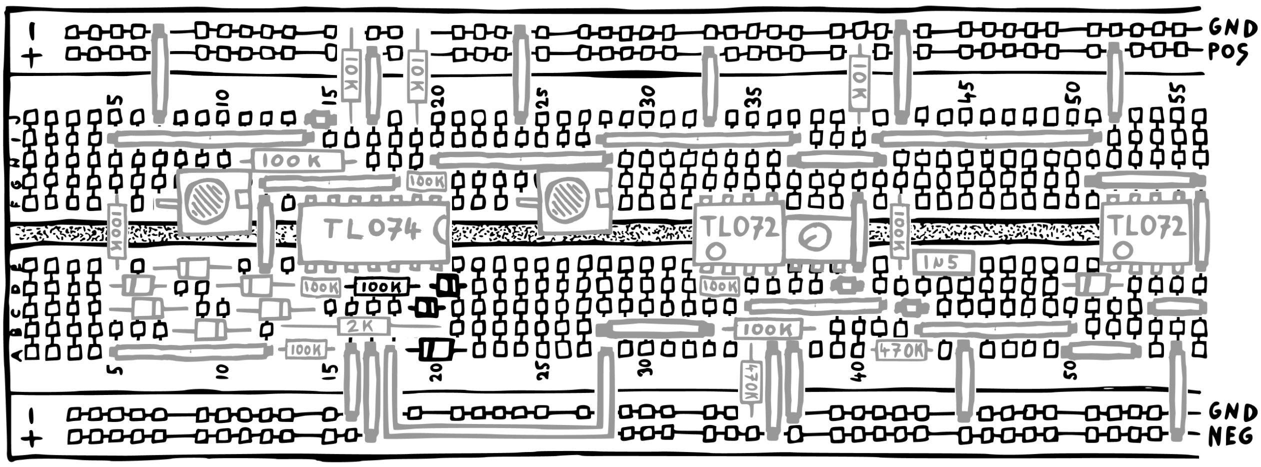


Then, when the voltage at the non-inverting input rises slightly above ground level, the op amp is forced to increase its output voltage enough to turn on the two series diodes. Okay, but why two diodes and not six? First, because we don't want to make our input buffer too sensitive to small changes in input voltage, because that can lead to unwanted distortion.

And second, because our dual buffer setup effectively doubles the voltage applied across the diode string, and the turn on voltages of diodes in series simply get summed together. So two times the turn on voltage for two diodes in series is equal to the turn on voltage of four diodes in series.

Alright, what about the diode facing the other direction, though? **Well, without it, the buffer would crash down to the negative rail once the CV input goes below 0 V, resulting in ugly distortion**. To prevent this, we simply give our buffer a back channel to pull current through.

Cool, and the capacitor? **This is more of a safety measure to make sure the op amp doesn't start oscillating in certain scenarios.** Since this is more likely to happen on a PCB than on a breadboard, we can omit the capacitor for now, though.[14]

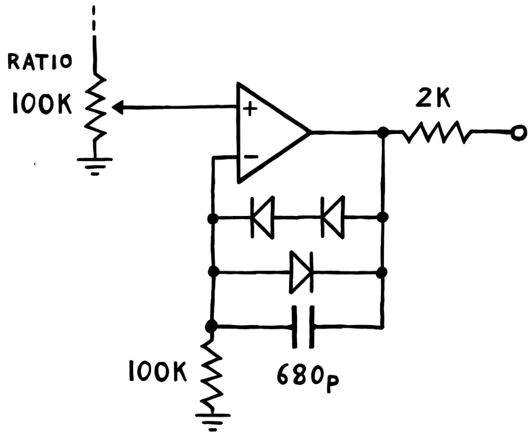


Once you've set this up, you'll see that the gap between threshold and peaks is greatly reduced. Unfortunately, it won't sound super great. This shouldn't surprise us, though, since our compressor now reacts really violently when the signal crosses the threshold.

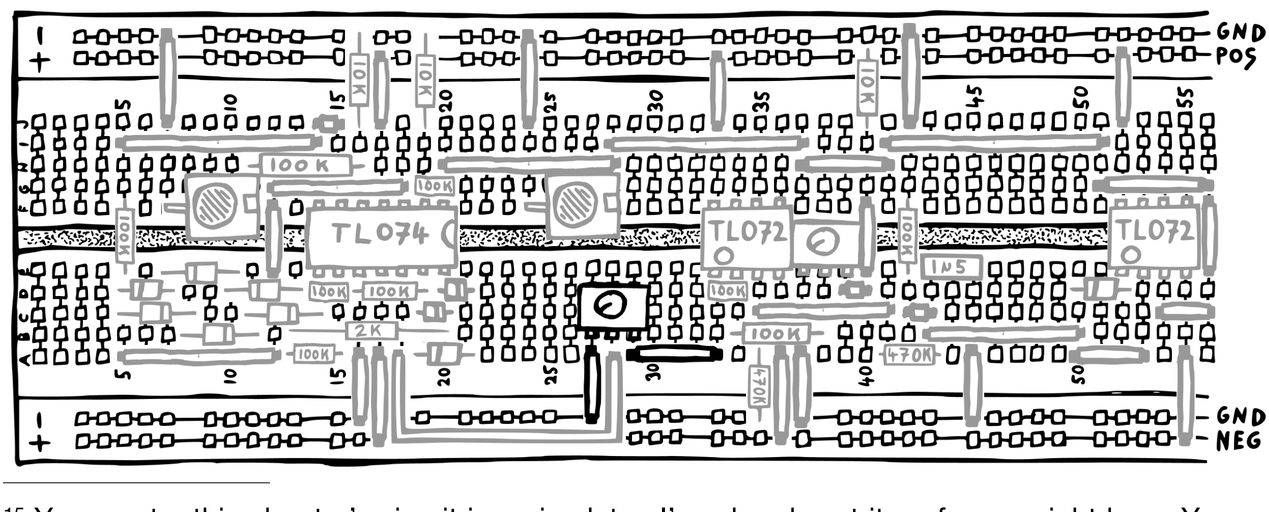
[14] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

VARBIABLE COMPRESSION RATIO

So what's the remedy? Easy: relax a bit on the compression ratio. **To do that, we just have to scale down the feedback we apply to the VCA.** Because if we cut it down by, say, 50%, then the gap between detected peak and threshold has to be twice as big before the VCA fully clamps down on the output volume.
To implement this, we'll simply set up a variable voltage divider at the VCA's CV input. This will allow us to adjust the compression ratio on the fly by turning a knob.



There's a catch, though: if you'd use a regular (i.e. linear) potentiometer here, you'd get a huge dead zone in the pot's range. This actually makes a lot of sense, though. Because as we said before, setting the voltage divider to 50% means that the initial, tiny gap between peak and threshold is allowed to double.
But doubling a tiny value will give you a (still) tiny result. Only once we get up to factors of 20 and above will we see a noticeable change. **Luckily, we can counteract this exponential relation if we use a logarithmic pot as our variable voltage divider – instead of a linear one.**[15]



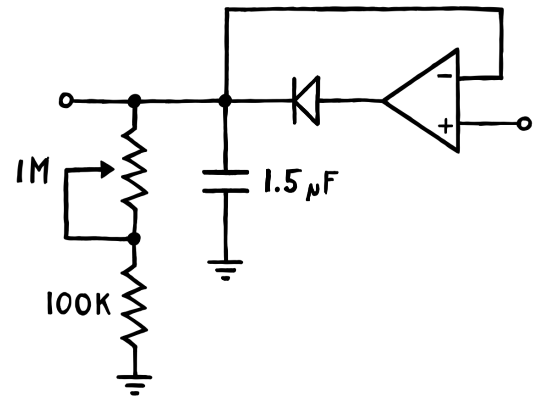
[15] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

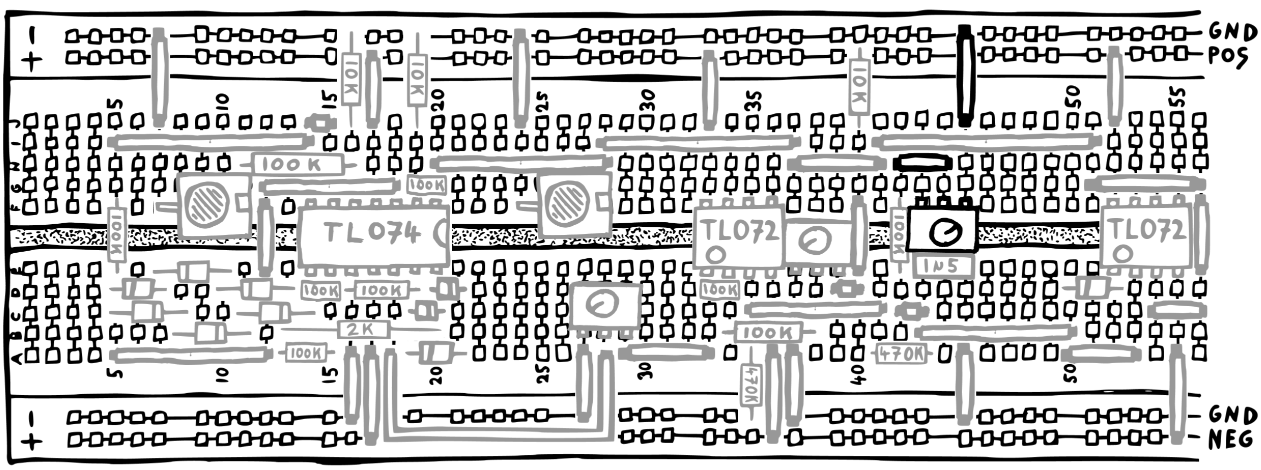
VARIABLE RELEASE

Now there's just two more controls to figure out: attack and release. Thankfully, both are really straightforward to implement in our current setup. Let's start with the release dial. As we said before, the release tells the compressor how long it should wait before restoring the output volume once the input drops back below the threshold. Conveniently, we've already set a release time for our compressor when we constructed the peak detector.

Remember how we added that 100k resistor to ground to measure volume over an extended period of time? This works because it delays the drop of the detected peak down to the actual output volume. **And since the detected peak is what determines the amount of compression we apply, a delayed drop there directly results in an extended release time**.

So all we have to do is replace this 100k resistor with a 1M potentiometer and we've got our release dial – right? Well, almost. We also need to set a minimum resistance, because otherwise we'd create a short circuit at the pot's minimum setting. In my experiments, a 100k resistor worked best here: go any lower, and the release is so short that the output gets noticeably distorted.[16]

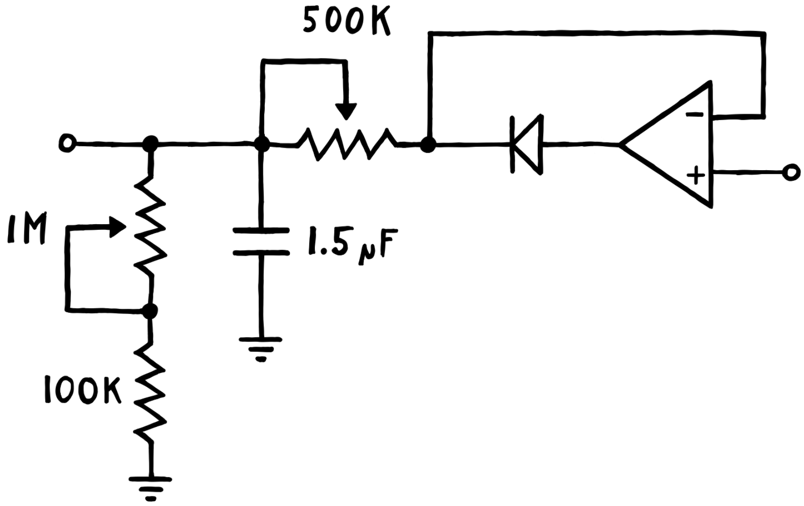




[16] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

VARIABLE (SNAPPY) ATTACK

Now, for the attack dial, the same idea applies in reverse. **If we want the compression to kick in later, we just have to slow down the capacitor's charging process**. And we do that (you guessed it) by putting a potentiometer between the diode and the capacitor.

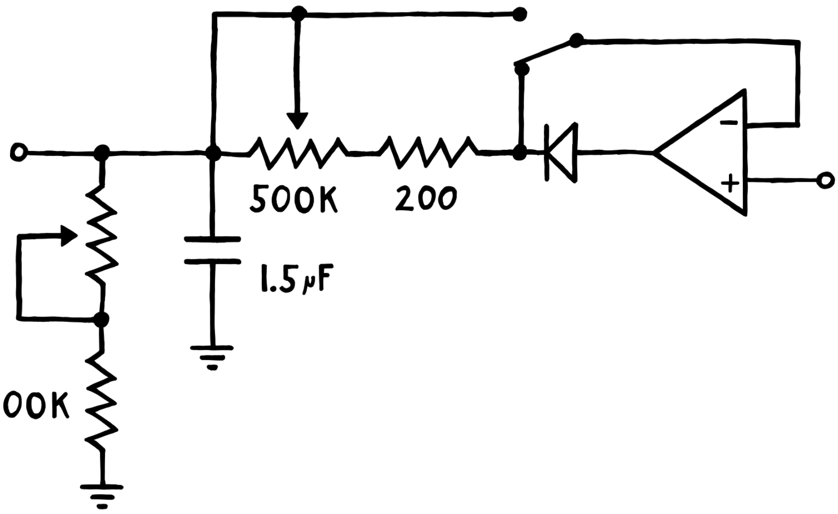


There are a couple caveats here, though. First, asking the op amp to fill up a 1.5 µF capacitor without any resistance in the charging path isn't the wisest idea (it can lead to component failure in the long run). So we should also add a small 200 ohms series resistor.

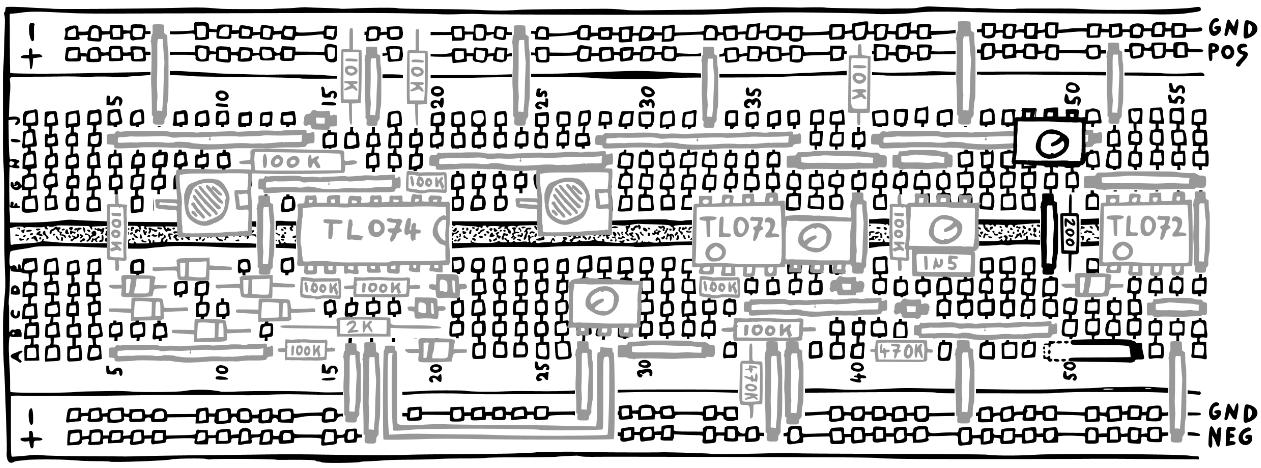
And second, while this setup does work in principle, fast attack times won't sound quite as snappy as you might hope. Thankfully, there's a simple hack for that. **The problem is that in our setup, with a non-zero attack time dialed in, the speed of the charging process is not static**. Instead, it gets slower exponentially as the cap fills up. So our compression kicks in gradually – which causes the lack of snappiness.

To fix this, we can include the potentiometer in the buffer's feedback path. Then, the op amp will blast the diode and resistor with 12 V until the capacitor is charged to the input voltage. This brute force approach results in an almost linear-looking charging process. Which should cause the compression to kick in more sharply.

Alright, but what if you encounter a situation where the regular behavior would actually sound better? In that case, you'd probably want to turn off the snappy behavior. **For that, all we need to do is take the potentiometer out of the feedback loop**. And we can do that with a simple SPDT switch. If we set it up as shown below, flipping the switch puts the potentiometer either into or out of the feedback loop. And that's it!



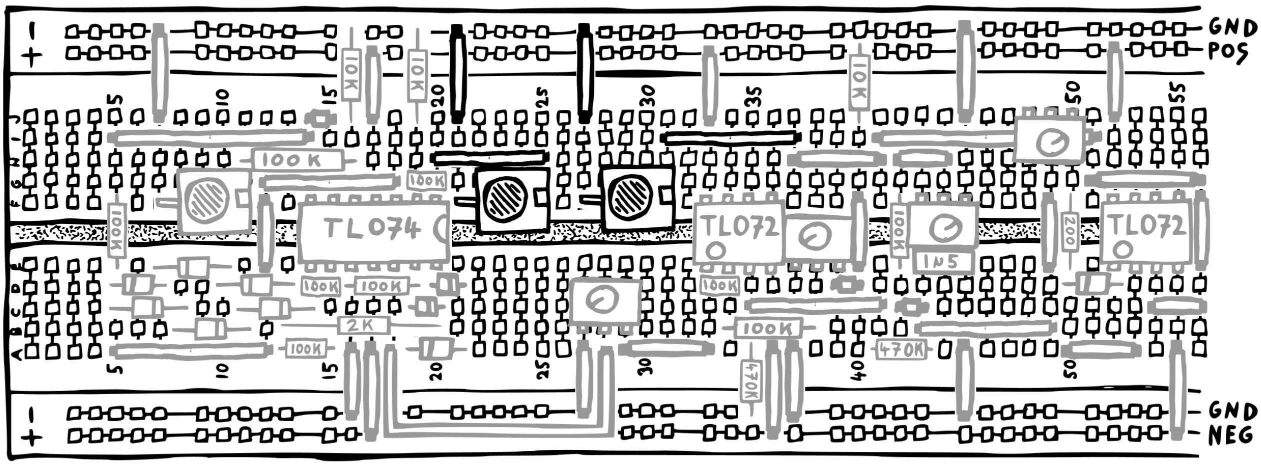
Now, because the switch in your kit can't be plugged into a breadboard, you can only test the snappy vs. regular attack behavior by using two different lengths of jumpers (as shown in the breadboard layout).[17]



[17] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

SIDE CHAINING

With the basic compressor functionality done, we'll now get into fun bonus feature-territory. First up: side chaining. What's side chaining? **It's when you control the compressor with a secondary signal instead of its own output**. So peaks in that secondary signal will cause the main signal to drop in volume.
Trying this with our circuit is as easy as breaking the feedback loop and attaching a jack socket to the peak detector. Then, the compression is triggered by whatever we feed into that socket.



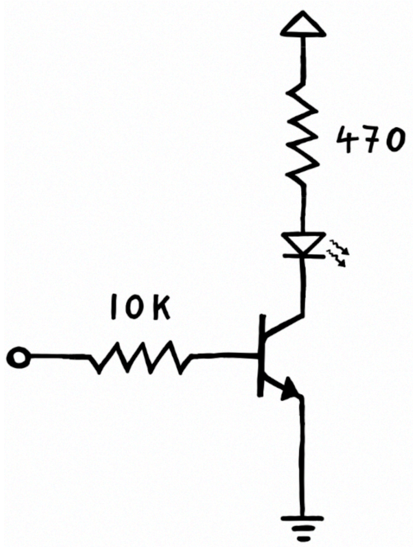
To test this, send something like a kick drum into the side chain-input on the right. You should then hear the main signal duck away every time the kick hits.
Attention: since real estate on the breadboard is limited and we've got a lot more to set up, we'll get rid of the side chain-input in the next breadboard layout!

GAIN REDUCTION INDICATOR

While testing the different functionalities, you might have noticed that it's a slight bit difficult to gauge when the compressor is actually active. So let's do something about that! We'll implement a simple LED-based gain reduction indicator. Here's the basic idea: **we want the LED to light up once the compressor starts reducing the output volume.** And ideally, the LED's brightness should be tied to the amount of gain reduction we apply.

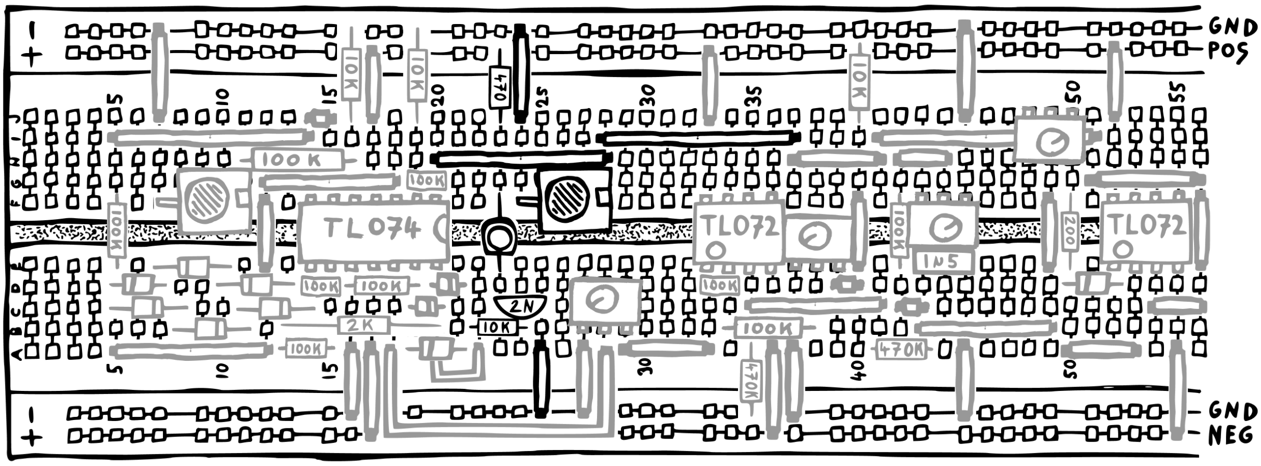
Alright, but how do we know when the compressor is active? Simple: by looking at what's going on inside the VCA. As we discussed previously, that VCA works like this: the more current flows through the diode string, the more the volume of the input signal is reduced. And the amount of current flowing is directly determined by the voltage coming from the CV input buffer. So if we use that voltage to light up our LED, we should already get what we were looking for, right?

Well, almost. There's two problems with this idea. First, a standard LED eats up much more current than an op amp is able to push out. And second, the voltages we apply to the diode string are not high enough to properly light up that LED anyways. **So we'll need to amplify our op amp's output somehow.** Thankfully, we can do that really easily using a standard NPN transistor and two resistors. If we set them up like this and connect the input node directly to the CV buffer's output, our LED will light up as intended.



Here's how it works. Once the buffer voltage rises significantly above 0 V, a small current is pushed into the transistor's base, which in turn allows a much larger (but proportional) current to flow from the positive rail, through the collector resistor, the LED, the transistor, and to ground.

And while this mechanism would in principle work without the two resistors, we need the collector resistor to prevent the LED from burning up – and the base resistor to make the LED light up gradually instead of having hard on/off states.[18]



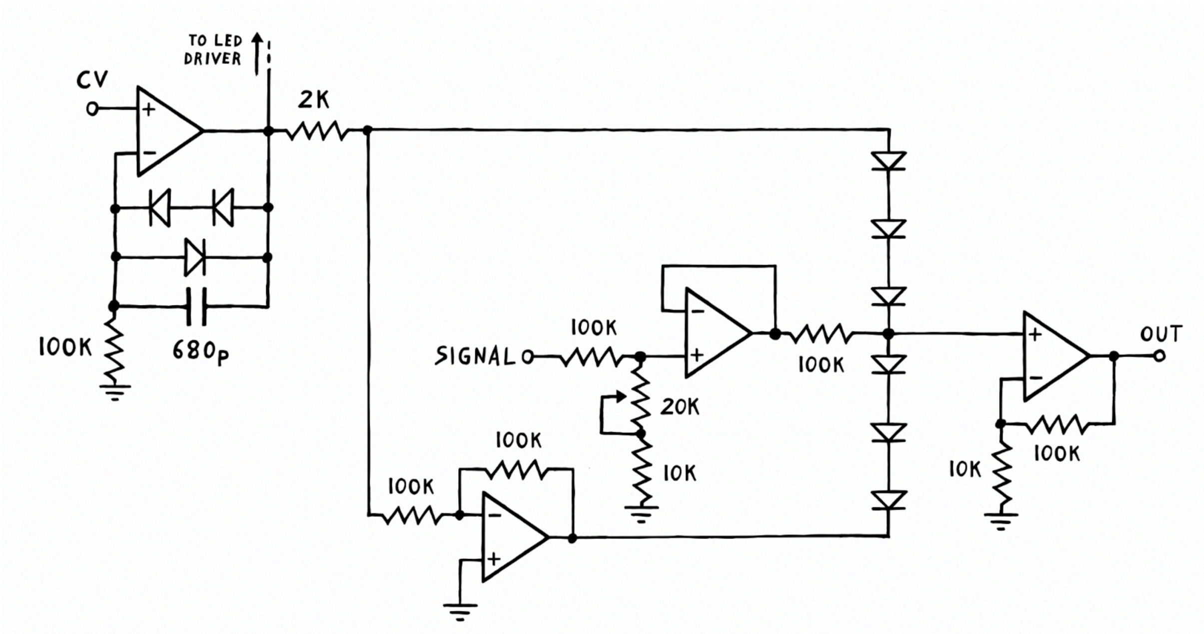
To test this out, send a loud signal into the signal input and then gradually push down the threshold. You should see the LED get brighter the more the compressor reduces the gain. But what if your input signal is actually really quiet – and you barely see the LED light up?

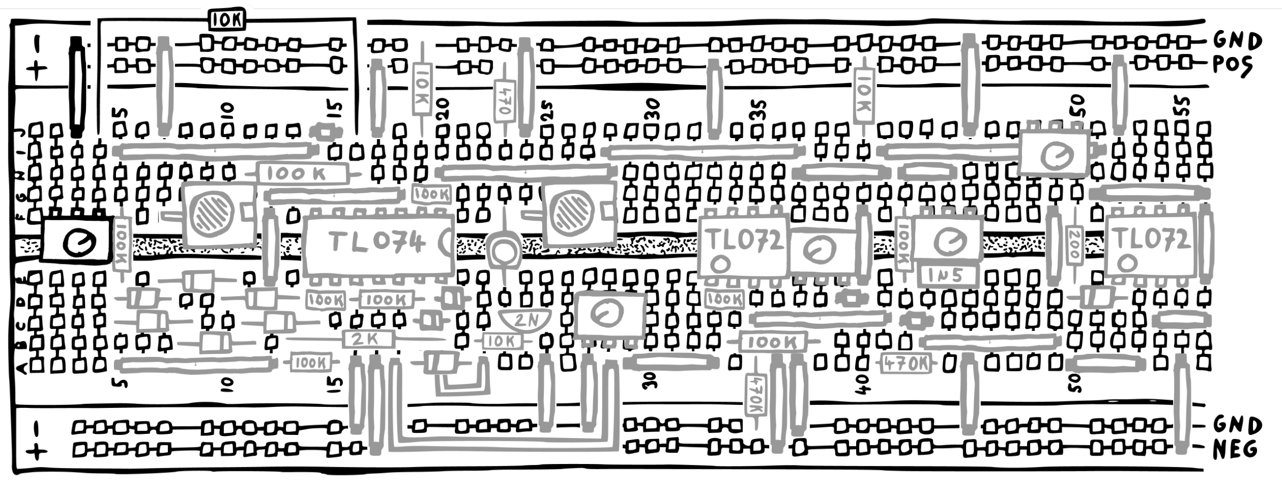
[18] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

VARIABLE INPUT GAIN

In that case, whatever is producing your sound is probably sending out a line level signal. What can we do about that? Easy: amplify the input! Or, in our case: don't cut it down so aggressively. Because as you might remember, we had to cut it down to prevent our VCA from distorting the signal.

But if the input is very low in volume anyways, we don't have to cut it down as much. **And because the amplifier on the other side of the VCA is tuned to restore the signal up to eurorack levels (which is super loud), the input will leave the VCA louder than it actually entered.** Now, to give us some flexibility, I decided to implement this using a potentiometer. If we set it up like this, the minimum setting will leave the input gain untouched – and we can increase it up to 3x from there by turning the knob.[19]

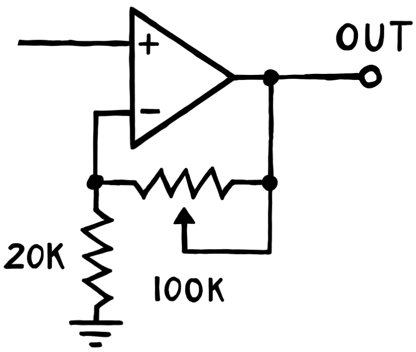




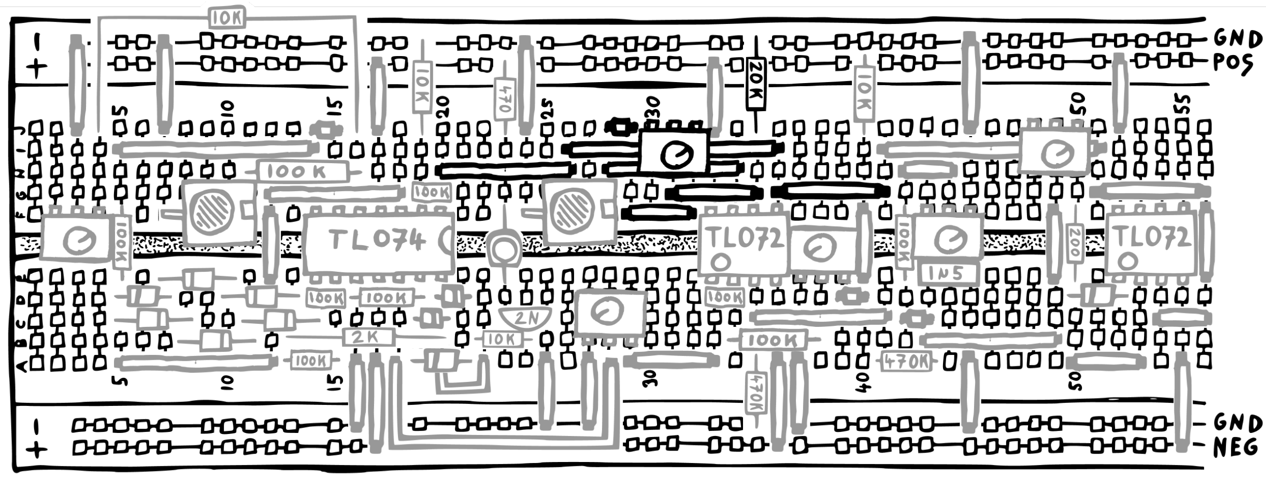
[19] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

VARIABLE MAKE-UP GAIN

Now, while playing around with threshold & ratio, you probably noticed that the output volume can get really low. Why is that? **Simple: if you compress a signal, you turn the loud parts down to reduce its dynamic range**. Because of this, the overall output volume is going to be lower.
To compensate, most compressors include a make up gain stage, which allows you to push the overall output volume back up. Implementing this is pretty straightforward: we just set up a non-inverting amplifier with variable gain and connect it to the compressor's output.



Combining a 100k potentiometer with a 20k resistor to ground in the op amp's feedback path gives us a gain range of 1x to 6x, which should be plenty.

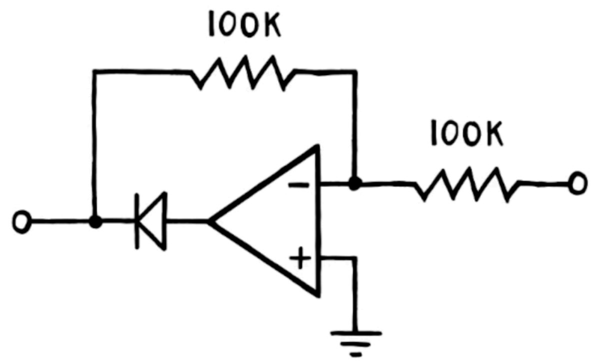


To test this out, first dial in a very aggressive threshold & ratio. Then, counteract the loss of volume by turning up the make-up gain. And while this should work just fine, you might notice that the output is getting pretty noisy. **This is the unfortunate side-effect of amplification – it doesn't discriminate between signal and noise**.
In our particular case here though, we can pretty much ignore the noise – simply because it's coming from the breadboard itself. Breadboards, if you don't know, are notoriously

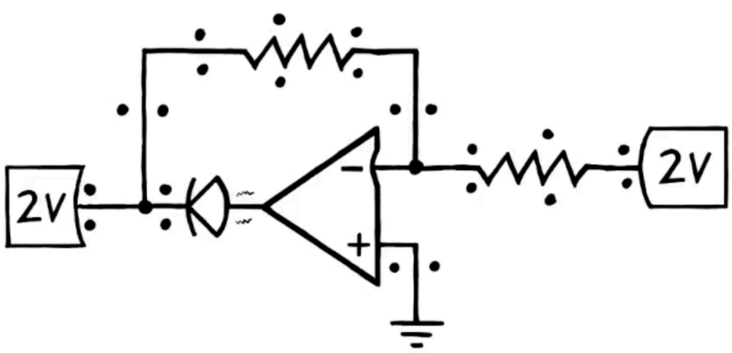
bad when it comes to baseline noisiness. Once you assemble the module, things will sound a lot cleaner.

FULL WAVE RECTIFIER

And while our compressor is perfectly usable already, there's one last improvement we should make. Remember how we decided to measure the output signal's volume by simply discarding its lower half? We did that based on the assumption that most audio signals are symmetrical, and so we wouldn't actually lose much information.
In reality though, there are a ton of sounds that are asymmetrical, especially those coming from outside a synthesizer. In those cases, we do lose plenty of information. So how do we fix this? With something called a full wave rectifier. The idea here is this. **What if instead of throwing away the waveform's lower half, we'd simply fold it upwards?** Then, our peak detector could read both sides of the waveform and react to any asymmetries. Cool idea in theory – but how do we pull this off? Well, there's a couple different ways to do it, but I decided to go for a rather simplistic approach using only an op amp, two 100k resistors and a diode.

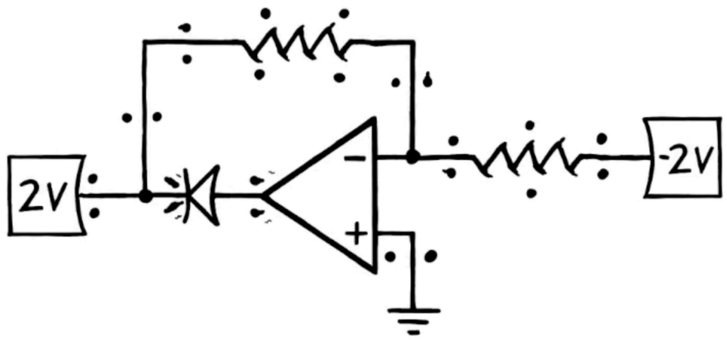


If we set them up like this, any signal we feed into the input on the right will be folded upwards along the 0 V line. Here's how it works. If the input voltage goes positive, the op amp will try to pull the voltage at the inverting input down to ground level. This won't work though, because the diode is blocking that.

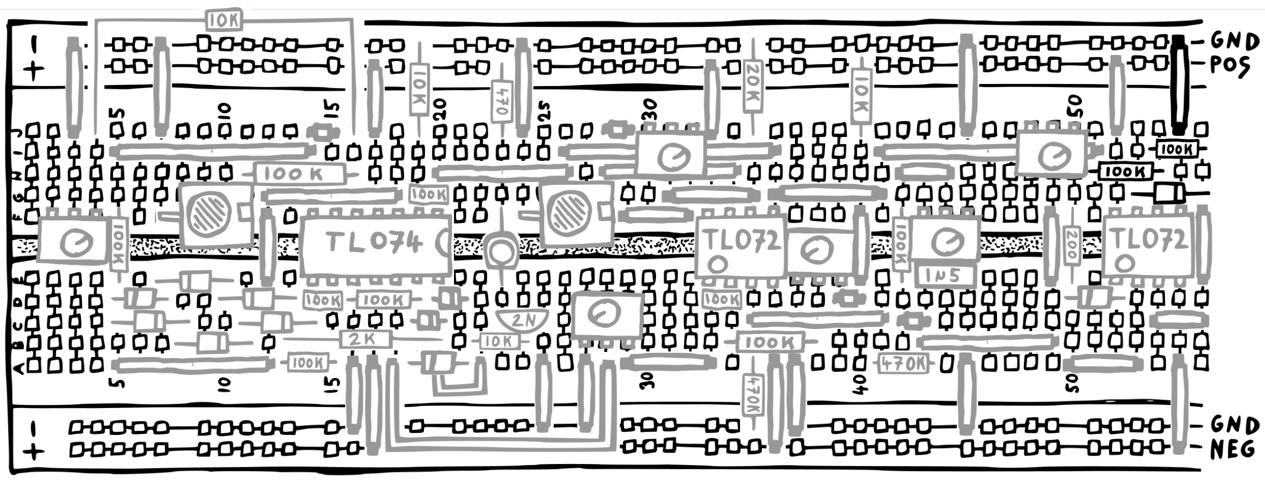


So ultimately, the output voltage will be identical to the input voltage – because no current is actually flowing through those resistors. This means that the upper part of our waveform will stay untouched.

Once the input goes negative, though, things get interesting. Because now, the op amp will try to push the voltage at the inverting input up to ground level – and actually succeed. For that, it has to raise the voltage after the diode to the exact inverse of the input voltage.



And that's because the two resistors now form a 50% voltage divider. So we'll only get 0 V at the inverting input if 0 V is the the exact midpoint between the input voltage and the voltage after the diode. **In effect, this means that the lower part of our waveform will be inverted.**[20] Great! There is just one downside to this setup.
It will only work properly as long as we don't draw any current from the output node. That's because during the input signal's high phase, the output voltage will only be identical to the input if no current flows through the resistors. In our case, this is not an issue, though, since the peak detector we'll be feeding is input buffered and doesn't draw any current from the rectifier.

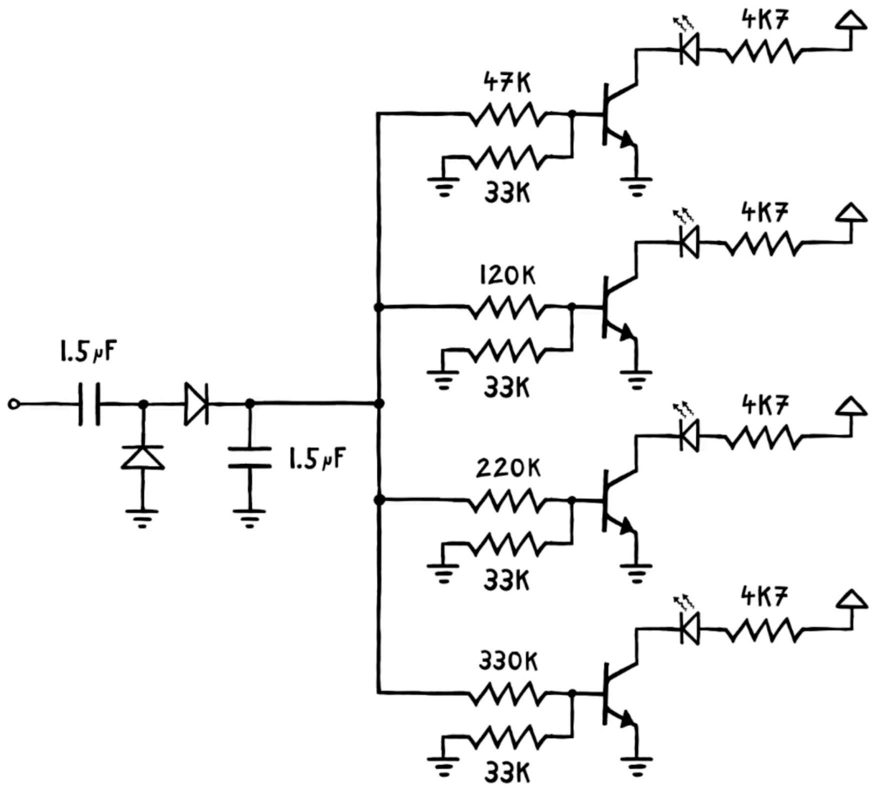


If you now test this with a sound like a human voice, electric guitar or similar, you should be able to notice an improvement in response time.

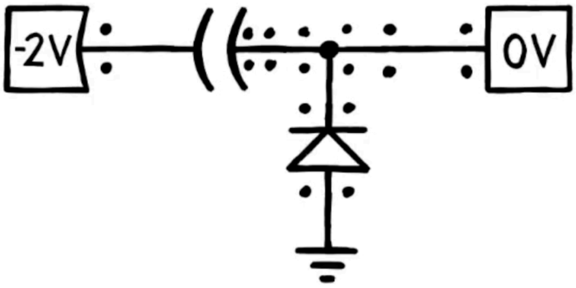
[20] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components.

LED-BASED VU METER

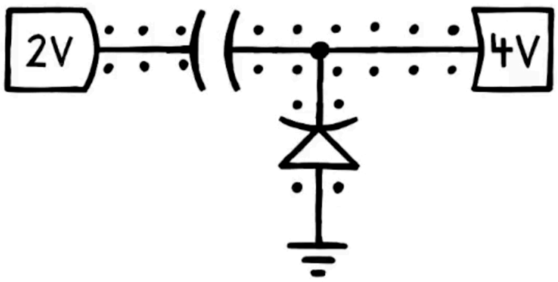
Now, when I sent the above circuit over to my friends at Erica Synths and asked them to turn it into a prototype module, they said: “Sure, but why don’t we add a couple more LEDs? People love LEDs!”, and sent back schematics for a super simple LED-based VU meter.



Since I liked this idea a lot, I decided to add it to my design – of course, only after taking a closer look at how it works. Here’s what I found. So first up, we’ve got an input stage consisting of two capacitors and two diodes. To be honest, I haven’t seen this design pattern before, so I was initially pretty confused at what it does, exactly.
It becomes more clear if we think about the expected input, though. A VU meter is supposed to visualize the volume of an audio signal. **So it’s safe to assume that the second diode and capacitor work as a peak detector**. It’s not immediately obvious, because the resistance to ground that a peak detector normally has is sort of hidden away: the voltage dividers on the right work double duty for that.
Okay, but what about the other capacitor & diode? This is where it gets a little tricky: **they’re used to push the audio input above the 0 V-line, so that the peak detector can actually measure its volume top to bottom**.
Here’s how it works. Let’s imagine we isolate the sub-circuit and apply a simple square wave to the input node. During the wave’s first low phase, current gets pulled out of the cap and flows into it on the other side.



This will leave us with -2 V on the left and about 0 V on the right. Then, when the input swings high, the charge on the other side has nowhere to go, since the diode is blocking.
Because of this, the absolute amount of voltage we add at the input is stacked on top of the 0 V on the right. So we'll read almost 4 V there.



And that's it: we've pushed the signal above the 0 V-line. Now if we reassemble the circuit, it still works roughly the same way – and so the peak detector gets to measure the signal's volume top to bottom. Its output is then used to feed four LED drivers very similar to the one we set up earlier for the gain reduction indicator. The kicker here are the four voltage dividers before those, though.
Since their divide-down ratios ramp up almost exponentially, the diode at the top will light up at much lower voltages than the one at the bottom. This way, the louder the input signal, the more LEDs light up in total.[21] And that's all there is to this super simple VU meter. Now, since we already have an indicator for the amount of gain reduction we apply, Erica and I decided that it'd make the most sense to use the VU meter for visualizing the compressor's output volume.[22]
And with this, our compressor is complete. Once you're done experimenting, dig out the panel and PCB from the kit, heat up your soldering iron and get to building. You can find more information on how to populate the board & how to solder in the enclosed appendix.

[21] You can try this chapter's circuit in a simulator. I've already set it up for you right here. You can change all values by double clicking on components

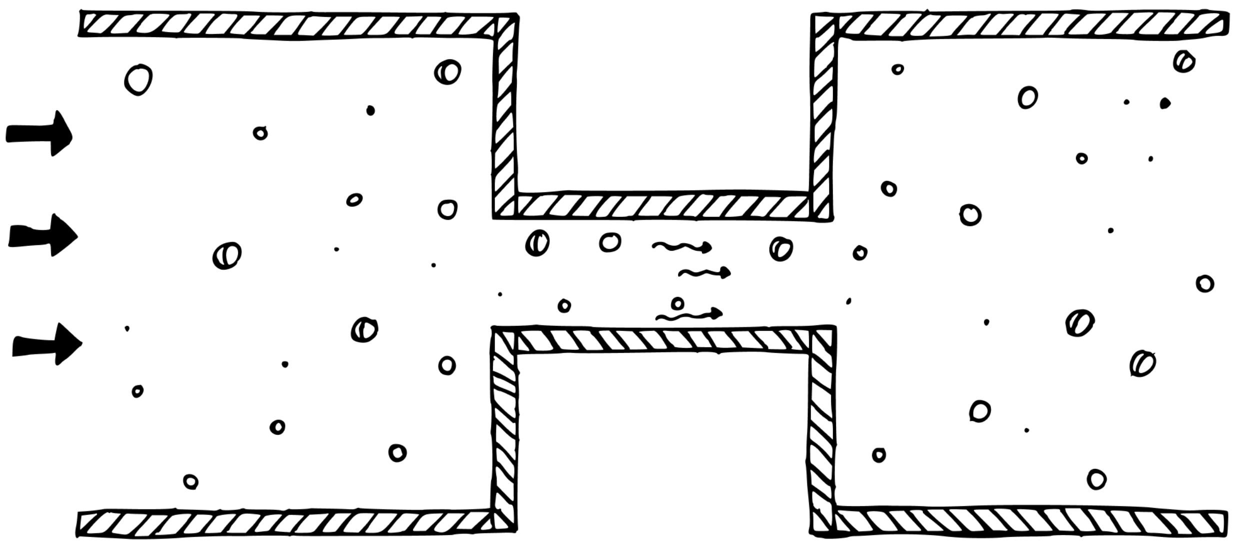
[22] Unfortunately, there's no more space left on the breadboard, so we won't be able to set the VU meter up there.

COMPONENTS & CONCEPTS APPENDIX

In this section, we'll take a closer look at the components and elemental circuit design concepts we're using to build our module. Check these whenever the main manual moves a bit too fast for you!

THE BASICS: RESISTANCE, VOLTAGE, CURRENT

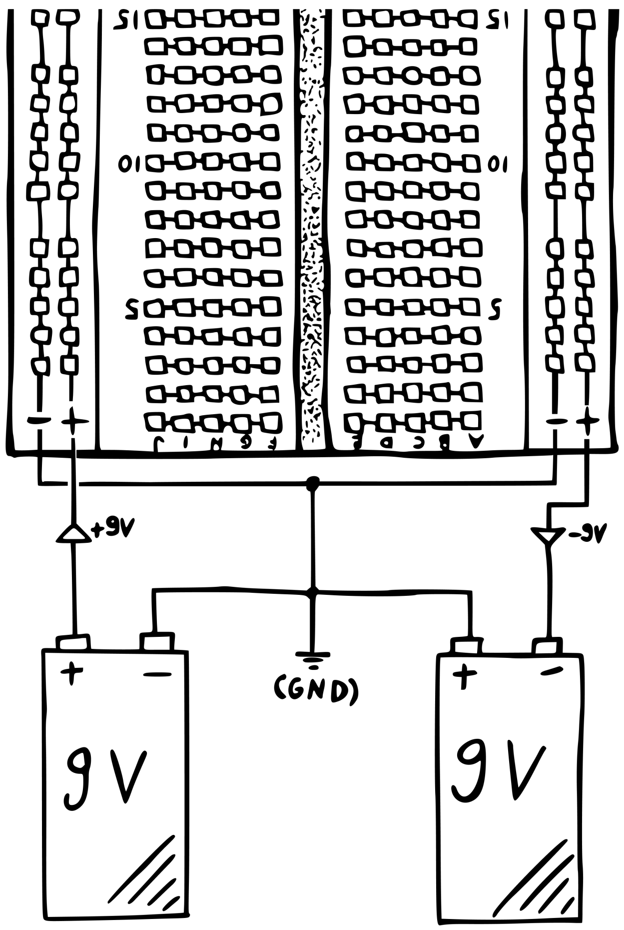
There are three main properties we're interested in when talking about electronic circuits: resistance, voltage and current. To make these less abstract, we can use a common beginner's metaphor and compare the flow of electrons to the flow of water through a pipe.



In that metaphor, resistance would be the width of a pipe. The wider it is, the more water can travel through it at once, and the easier it is to push a set amount from one end to the other. Current would then describe the flow, while voltage would describe the pressure pushing the water through the pipe. You can probably see how all three properties are interlinked: **more voltage increases the current, while more resistance to that voltage in turn decreases the current**.

USING TWO 9 V BATTERIES AS A DUAL POWER SUPPLY

Dual power supplies are great – and if you want to get serious about synth design, you should invest in one at some point. But what if you're just starting out, and you'd like to use batteries instead? Thankfully that's totally doable. **You just need to connect two 9 V batteries like shown here.** For this, you should use 9 V battery clips, which are cheap & widely available in every electronics shop.

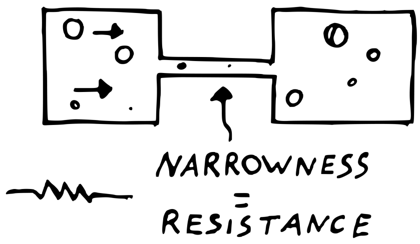


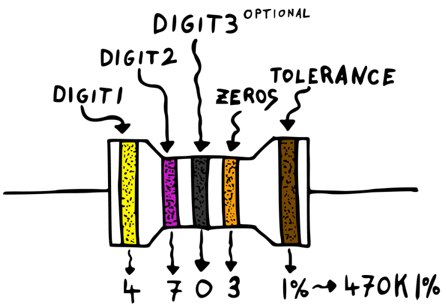
By connecting the batteries like this, the positive terminal of the left battery becomes your +9 V, while the negative terminal of the right is now your –9 V, and the other two combine to become your new ground.[23] **Please make sure you disconnect the batteries from your breadboard when you make changes to the circuit!** Otherwise you run the risk of damaging components.

[23] If you're struggling with setting this up, you can watch me do it here.

RESISTORS

While a conductive wire is like a very big pipe where lots of water can pass through, **a resistor is like a narrow pipe that restricts the amount of water that can flow**. The narrowness of that pipe is equivalent to the resistance value, measured in ohms (Ω). The higher that value, the tighter the pipe.





Resistors have two distinctive properties: linearity and symmetry. Linearity, in this context, means that for a doubling in voltage, the current flowing will double as well. Symmetry means that the direction of flow doesn't matter – resistors work the same either way.

On a real-life resistor, you'll notice that its value is not printed on the outside – like it is with other components. Instead, it is indicated by colored stripes[24] – along with the resistor's tolerance rating. In addition to that, the resistor itself is also colored. Sometimes, depending on who made the resistor, this will be an additional tolerance indicator.

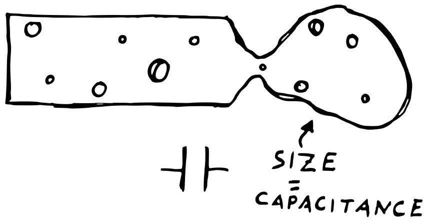
For the resistors in this kit, a yellow body tells you that the actual resistance value might be ±5 % off. A dark blue body indicates ±1 % tolerance. Some kits will also contain light blue ± 0.1% resistors to avoid the need for manual resistor matching.

While in the long run, learning all these color codes will be quite helpful, you can also simply use a multimeter to determine a resistor's value.

[24] For a detailed breakdown, look up resistor color coding. There are also calculation tools available.

CAPACITORS

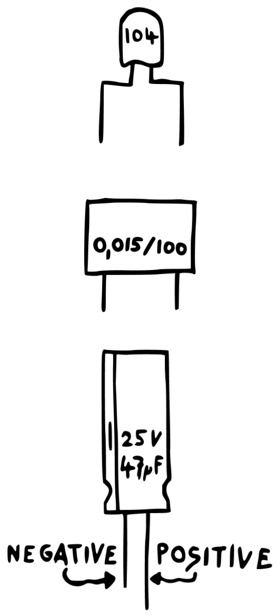
A capacitor is a bit like a balloon that you can attach to the open end of a pipe. If there's some pressure in the pipe, the balloon will fill up with water until the pressure equalizes. (Since the balloon needs some space to expand into, both of the capacitor's legs need to be connected to points in your circuit.)



Then, should the pressure in the pipe drop, the balloon releases the water it stored into the pipe. The maximum size of the balloon is determined by the capacitor's capacitance, which we measure in farad (F). There are quite a few different types of capacitors: electrolytic, foil, ceramic, tantalum etc. They all have their unique properties and ideal usage scenarios – but the most important distinction is if they are polarized or not.

You shouldn't use polarized capacitors against their polarization (applying a negative voltage to their positive terminal and vice versa) – so they're out for most audio-related uses like AC coupling, high- & low-pass filters etc.

Unlike resistors, capacitors have their capacitance value printed onto their casing, sometimes together with a maximum operating voltage. **Be extra careful here!** That voltage rating is important. Your capacitors can actually explode if you exceed it! So they should be able to withstand the maximum voltage used in your circuit. If they're rated higher – even better, since it will increase their lifespan. No worries though: the capacitors in this kit are carefully chosen to work properly in this circuit.



Ceramic capacitors usually come in disk- or pillow-like cases, are non-polarized and typically encode their capacitance value.[25] Annoyingly, they rarely indicate their voltage rating – so you'll have to note it down when buying them.

Film capacitors come in rectangular, boxy cases, are non-polarized and sometimes, but not always, directly indicate their capacitance value and their voltage rating without any form of encoding.[26]

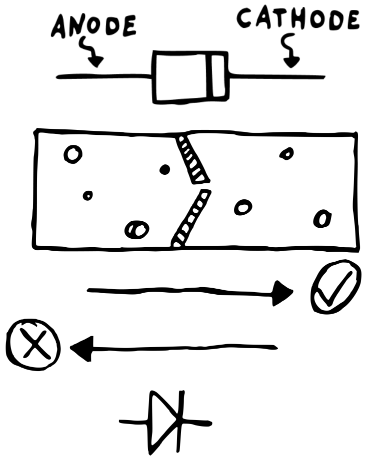
Electrolytic capacitors can be identified by their cylinder shape and silver top, and they usually directly indicate their capacitance value and their voltage rating. They are polarized – so make sure you put them into your circuit in the correct orientation.

---

[25] For a detailed breakdown, look up ceramic capacitor value code. There are also calculation tools available.

[26] If yours do encode their values, same idea applies here – look up film capacitor value code.

DIODES

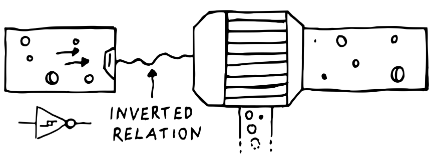


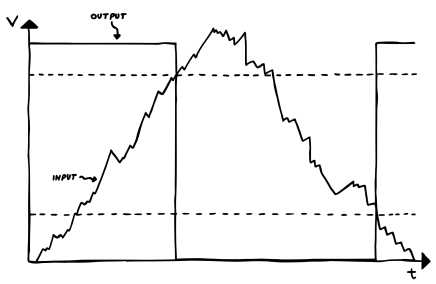
Diodes are basically like one-way valves. Current can only pass through in one direction – from anode to cathode. That direction is indicated by the arrow in the diode symbol and by a black stripe on the diode's casing. So any current trying to move in the opposite direction is blocked from flowing.

There are a few quirks here, though. For one, the diode will only open up if the pushing force is strong enough. Generally, people say that's 0.7 V**,** but in reality, it's usually a bit lower. Also, diodes don't open up abruptly – they start conducting even at much lower voltages, although just slightly.

There are a lot of different diode types: Zener, Schottky, rectifier, small signal etc. They all have their unique properties and ideal usage scenarios – but usually, a generic 1N4148 small signal diode will get the job done.

SCHMITT TRIGGER INVERTERS

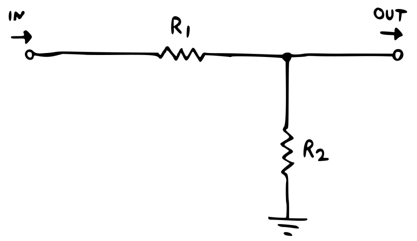




You can think of a Schmitt trigger inverter as two separate things. On the left, there's a sensor that measures the pressure inside an attached pipe. On the right, there is a water pump. This pump's operation is controlled by the sensor. Whenever the pressure probed by this sensor is below a certain threshold, the pump will be working. If the pressure is above a second threshold, the pump won't be working.

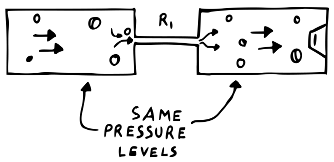
Here's a quick graph to visualize that. The squiggly line represents the voltage at the input, while the dotted line shows the voltage at the output. So every time we cross the upper threshold on our way up, and the lower one on our way down, the output changes its state. One thing that's very important to keep in mind: no current flows into the sensor! It's really just sensing the voltage without affecting it.

VOLTAGE DIVIDERS

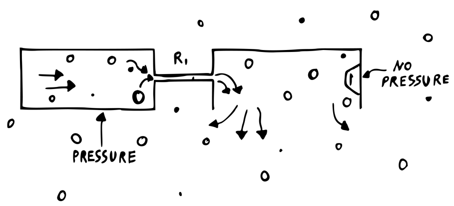


A voltage divider is really just two resistors set up like this: input on the left, output on the right. If R1 and R2 are of the same value, the output voltage will be half of what the input voltage is. How does it work?

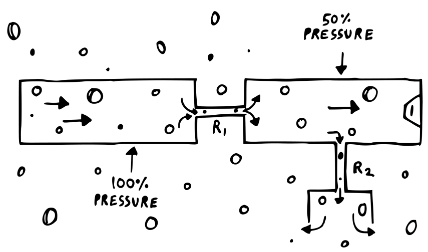
Let's use our analogy again: so we have a pipe on the left, where water is being pushed to the right with a specific amount of force. Attached to it is a narrow pipe, representing R1, followed by another wide pipe. Then at the bottom, there's another narrow pipe, representing R2, where water can exit the pipe system. Finally, imagine we've set up a sensor measuring the voltage in the right hand pipe.



First, think about what would happen if R2 was completely sealed off. Our sensor would tell us that **the pressure on the right side is exactly the same as the pressure on the left**. Because the pushing force has nowhere else to go.



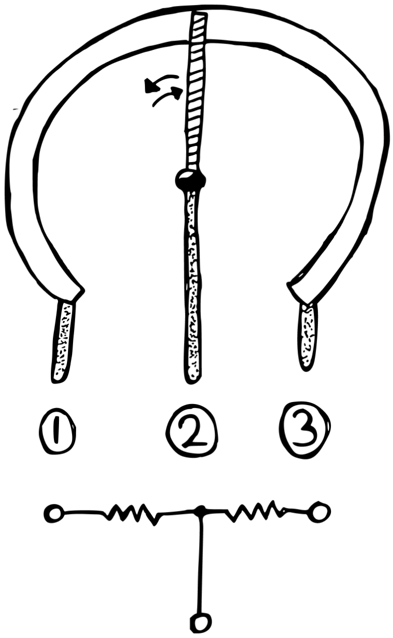
On the other hand, imagine R2 would just be a wide opening. Then **the pressure on the right would be 0**, because it'd all escape through that opening. But what happens if R2 is neither completely closed off nor wide open? Then the pressure would be retained to varying degrees, depending on the narrowness of the two resistor paths.



If pipe R1 is wide and pipe R2 is narrow, most of the pressure will be retained. But if it's the reverse, the pressure level will be only a tiny fraction. And if R1 and R2 are identical, **the pressure will be exactly half of what we send in**.

POTENTIOMETERS

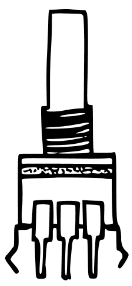
Potentiometers can be used as variable resistors that you control by turning a knob. But, and that's the handy part, they can also be set up as variable voltage dividers. To see how that works, let's imagine we open one up.

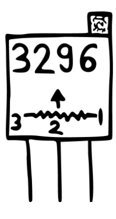


Inside, we would find two things: a round track of resistive material with connectors on both ends plus what's called a wiper. This wiper makes contact with the track and also has a connector. It can be moved to any position on the track. Now, the resistance value between the two track connectors is always going to stay exactly the same. That's why it's used to identify a potentiometer: as a 10k, 20k, 100k etc. But if you look at the resistance between either of those connectors and the wiper connector, you'll find that this is completely dependent on the wiper's position.

The logic here is really simple: **the closer the wiper is to a track connector, the lower the resistance is going to be between the two**. So if the wiper is dead in the middle, you'll have 50 % of the total resistance between each track connector and the wiper.

From here, you can move it in either direction and thereby shift the ratio between the two resistances to be whatever you want it to be. By now, you might be able to see how that relates to our voltage divider. If we send our input signal to connector 1 while grounding connector 3, we can pick up our output signal from the wiper. Then by turning the potentiometer's knob, we can adjust the voltage level from 0 to the input voltage – and anything in between.





In these kits, you will encounter different types of potentiometers. First, there's the regular, full-size variant with a long shaft on top. These are used to implement user-facing controls on the module's panel and they usually – but not always – indicate their value directly on their casing. Sometimes, they'll use a similar encoding strategy as capacitors, though.[27]

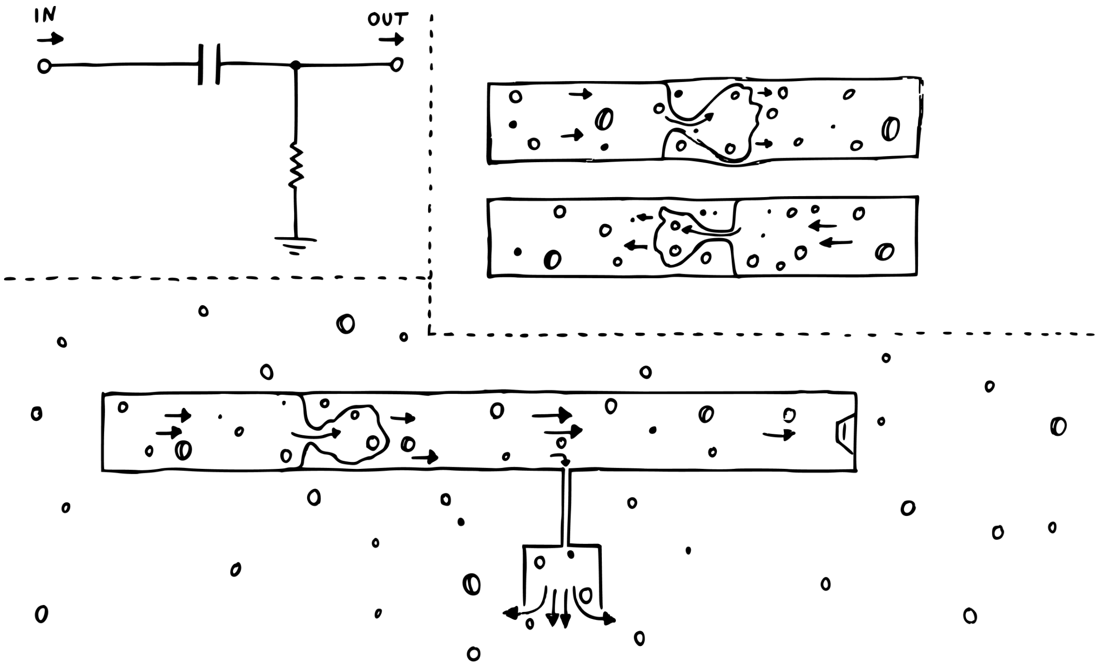
Second, we've got the trimmer potentiometer, which is usually much smaller and doesn't sport a shaft on top. Instead, these have a small screw head which is supposed to be used for one-time set-and-forget calibrations. Trimmers usually encode their value.

---

[27] Look up potentiometer value code for a detailed breakdown.

AC COUPLING

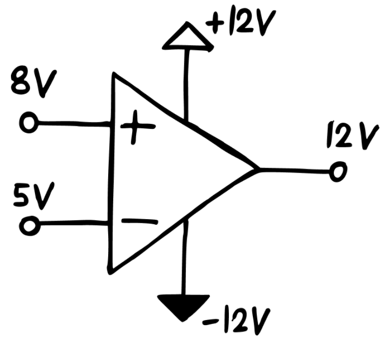
What is AC coupling – and how does it work? Imagine two adjacent pipes with a balloon between them. Now, no water can get from one pipe into the other, since it's blocked by the balloon. But, and that's the kicker, **water from one side can still push into the other by bending and stretching the balloon, causing a flow by displacement**.



Next, we'll bring in a resistor after the coupling point, going straight to ground. **This acts like a kind of equalizing valve**. Now imagine we apply a steady 5 V from one side. Then on the other side, we'll read 0 V after a short amount of time. Why? Because we're pushing water into the balloon with a constant force, causing it to stretch into the other side, displacing some water. If we didn't have the equalizing valve there, we'd simply raise the pressure. But since we do have it, the excess water can drain out of the system. Until the pressure is neutralized, and no water is actively flowing anymore.
Okay, so now imagine that the voltage on the left hand side starts oscillating, let's say between 4 V and 6 V. When we start to go below 5 V, the balloon will begin contracting, basically pulling the water to the left. This will create a negative voltage level in the right hand pipe – like as if you're sucking on a straw, making the voltage there drop below 0 V. Then, once the pressure on the other side rises above 5 V, the balloon will inflate and stretch out again, pushing water to the right. And the pressure in the right hand pipe will go positive, making the voltage rise above 0 V. **We've re-centered our oscillation around the 0 V line.** Okay, but what about the resistor? If current can escape through it, doesn't that mess with our oscillation? Well, technically yes, but practically, we're choosing a narrow enough pipe to make the effect on quick pressure changes negligible!

OP AMPS

Op amps might seem intimidating at first, but they're actually quite easy to understand and use. The basic concept is this: every op amp has two inputs and one output. Think of those inputs like voltage sensors. You can attach them to any point in your circuit and they will detect the voltage there without interfering. **No current flows into the op amps inputs – that's why we say their input impedance is very high**. Near infinite, actually. Okay, but why are there two of them?



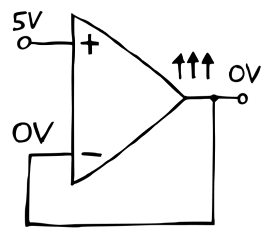
The key here is that op amps are essentially differential amplifiers. This means that they only amplify the difference between their two inputs – not each of them individually. If that sounds confusing, let's check out a quick example. So we'll imagine that one sensor – called the non-inverting input – is reading 8 V from somewhere. The other sensor – called the inverting input – reads 5 V. Then, as a first step, the op amp will subtract the inverting input's value from the non-inverting input's value. Leaving us with a result of 3. (Because 8 minus 5 is 3.) **This result then gets multiplied by a very large number – called the op amp's gain**. Finally, the op amp will try to push out a voltage that corresponds to that multiplication's result.

But of course, the op amp is limited here by the voltages that we supply it with. If we give it –12 V as a minimum and +12 V as a maximum, the highest it can go will be +12 V. So in our example, even though the result of that multiplication would be huge, the op amp will simply push out 12 V here and call it a day.

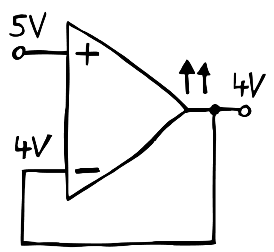
The handy thing though about op amp outputs is that they draw their power directly from the power source. This means that they can supply lots of current while keeping the voltage stable. **That's why we say an op amp has a very low output impedance**.

OP AMP BUFFERS/AMPLIFIERS

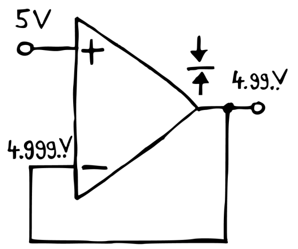
Buffering, in the world of electronics, means that we provide a perfect copy of a voltage without interfering with that voltage in the process. With an op amp-based buffer, the buffering process itself works like this. We use the non-inverting input to probe a voltage, while the inverting input connects straight to the op amp's output. **This creates what we call a negative feedback loop**. Think of it this way. We apply a specific voltage level to the non-inverting input – let's say 5 V.



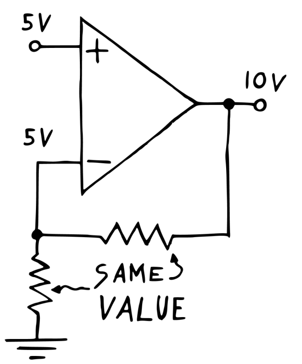
Before the op amp starts processing the voltages at its inputs, the output will be switched off. This means that **output and inverting input sit at 0 V at first**. So then, the op amp will subtract 0 from 5 and multiply the result by its gain. Finally, it will try and increase its output voltage to match the calculation's outcome.



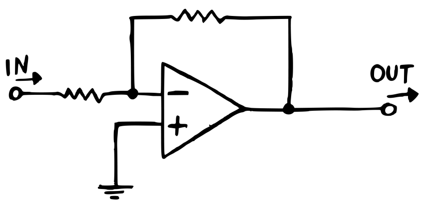
But as it's pushing up that output voltage, the **voltage at the inverting input will be raised simultaneously**. So the difference between the two inputs is shrinking down. Initially, this doesn't matter much because the gain is so large. As the voltage at the inverting input gets closer to 5 V though, the difference will shrink so much that in relation, the gain suddenly isn't so large anymore.



Then, the output will **stabilize at a voltage level that is a tiny bit below 5 V**, so that the difference between the two inputs multiplied by the huge gain gives us exactly that voltage slightly below 5 V. And this process simply loops forever, keeping everything stable through negative feedback. Now if the voltage at the non-inverting input changes, that feedback loop would ensure that the output voltage is always following. So that's why this configuration works as a buffer: the **output is simply following the input**.



How about amplifying a signal though? To do that, we'll have to turn our buffer into a proper non-inverting amplifier. We can do that by replacing the straight connection between inverting input and output with a voltage divider, forcing the op amp to work harder. Here's how that works. Say we feed our non-inverting input a voltage of 5 V. Now, **the output needs to push out 10 V in order to get the voltage at the inverting input up to 5 V**. We call this setup a non-inverting



amplifier because the output signal is in phase with the input.

For an inverting buffer/amplifier, the input signal is no longer applied to the non-inverting input. Instead, that input is tied directly to ground. So it'll just sit at 0 V the entire time. The real action, then, is happening at the inverting input. Here, we first send in our waveform through a resistor. Then, the inverting input is connected to the op amp's output through another resistor of the same value.

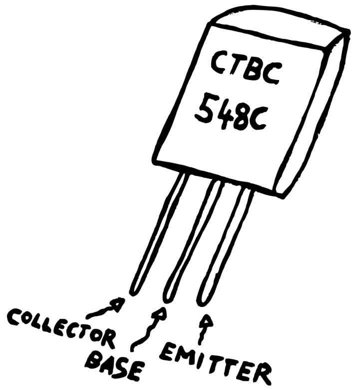
How does this work? Well, let's assume that we're applying a steady voltage of 5 V on the left. Then, as we already know, the op amp will subtract the inverting input's voltage from the non-inverting input's voltage, leaving us with a result of –5 V. Multiply that by the huge internal gain, and the op amp will try to massively decrease the voltage at its output.

But as it's doing that, an increasingly larger current will flow through both resistors and into the output. Now, as long as the pushing voltage on the left is stronger than the pulling voltage on the right, some potential (e.g. a non-zero voltage) will remain at the inverting input. Once the output reaches about –5 V though, we'll enter a state of balance. Since both resistors are of the same value, the pushing force on the left is fighting the exact same resistance as the pulling force on the right. **So all of the current being pushed through one resistor is instantly being pulled through the other**.

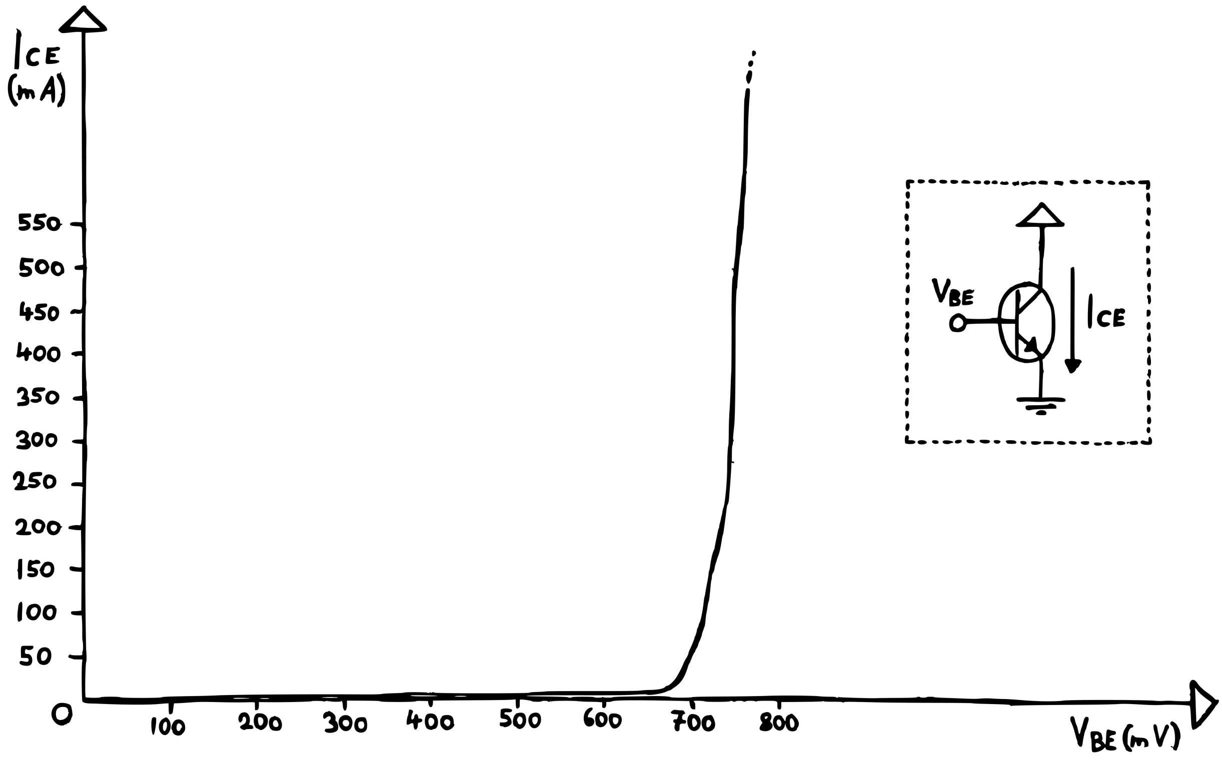
And that means that the voltage at the inverting input will be lowered to about 0 V, allowing our op-amp to settle on the current output voltage level. So while we read 5 V on the left, we'll now read a stable –5 V at the op amp's output. Congrats – we've built an inverting buffer! **If we want to turn it into a proper amplifier, we'll simply have to change the relation between the two resistances**. By doing this, we can either increase (if you increase the right-hand resistor's value) or reduce (if you increase the left-hand resistor's value) the gain to our heart's content.

BIPOLAR JUNCTION TRANSISTORS

Bipolar junction transistors (or BJTs for short) come in two flavors: NPN and PNP. This refers to how the device is built internally and how it'll behave in a circuit. Apart from that, they look pretty much identical: a small black half-cylinder with three legs.



Let's take a look at the more commonly used NPN variant first. Here's how we distinguish between its three legs. **There's a collector, a base and an emitter.**[28] All three serve a specific purpose, and the basic idea is that you control the current flow between collector and emitter by applying a small voltage[29] to the base. The relation is simple: **more base voltage equals more collector current**. Drop it down to 0 V and the transistor will be completely closed off. Sounds simple – but there are four important quirks to this.

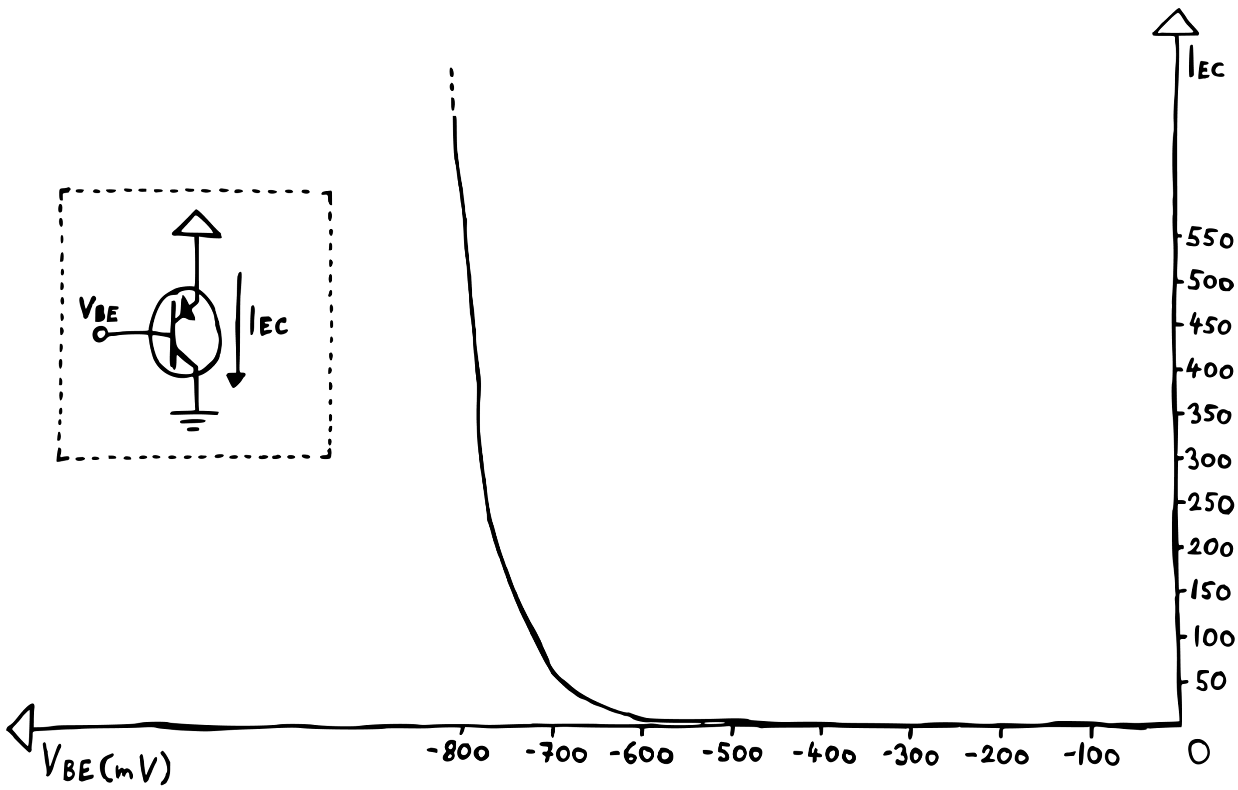


First, the relation between base voltage and collector current is exponential. Second, unlike a resistor, a BJT is not symmetrical – so we can't really reverse the direction of the

[28] Please note that the pinout shown here only applies for the BC series of transistors. Others, like the 2N series, allocate their pins differently.

[29] The voltage is measured between base and emitter. So „a small voltage“ effectively means a small voltage **difference** between base and emitter!

collector current. (At least not without some unwanted side effects.) Third, also unlike a resistor, a BJT is not a linear device. Meaning that a change in collector voltage will not affect the collector current. And fourth, the collector current is affected by the transistor's temperature! The more it heats up, the more current will flow.
Now, for the PNP transistor, all of the above applies, too – except for two little details. Unlike with the NPN, **the PNP transistor decreases its collector current when the voltage at its base increases**[30]. So you have to bring the base voltage below the emitter to open the transistor up. Also, that collector current flows out of, not into the collector!

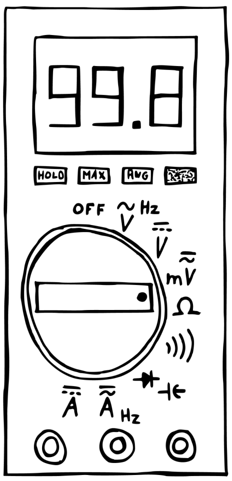


[30] Again, the voltage is measured between base and emitter.

TOOLS APPENDIX

There are two types of tools that will help you tremendously while designing a circuit: multimeters and oscilloscopes. In this appendix, we'll take a quick look at each of these and explore how to use them.

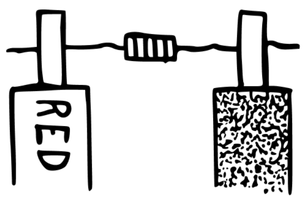
MULTIMETERS

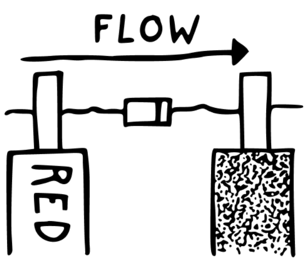


Multimeters come in different shapes and sizes, but the most common type is probably the hand-held, battery powered variant. It can measure a bunch of different things: voltage, current, resistance, continuity. Some have additional capabilities, allowing you to check capacitance, oscillation frequency or the forward voltage drop of a diode.

When shopping for one, you'll probably notice that there are really expensive models boasting about being TRUE RMS multimeters. For our purposes, this is really kind of irrelevant, so don't feel bad about going for a cheap model!

Using a multimeter is actually really straightforward. Simply attach two probes to your device – the one with a black cable traditionally plugs into the middle, while the red one goes into the right connector. Next, find whatever you want to measure and select the corresponding mode setting.



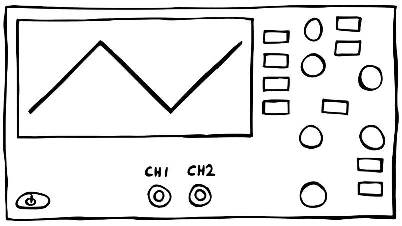


In some cases, it doesn't matter which probe you connect to which component leg or point in your circuit. This is true for testing resistors, non-polarized capacitors (foil/film, ceramic, teflon, glass etc.), continuity[31] or AC voltage.

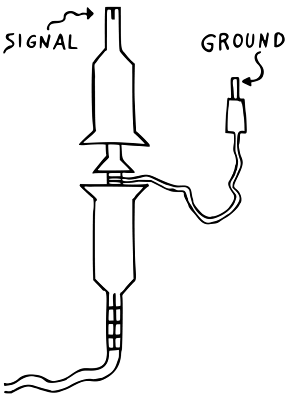
In others, you'll have to be careful about which probe you connect where. For testing the forward voltage drop of a diode, for example, **the multimeter tries to push a current from the red to the black probe**. Here, you'll have to make sure the diode is oriented correctly, so that it doesn't block that current from flowing. For testing a DC voltage, you want to make sure the black probe is connected to ground, while you use the red one to actually take your measurement.

[31] Just a fancy word for saying that two points are electrically connected.

OSCILLOSCOPES



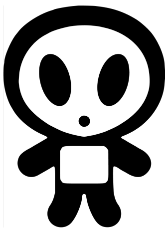
While multimeters are fairly cheap and compact, oscilloscopes are usually somewhat pricey and bulky. **If you're willing to make the investment, they are a huge help with the troubleshooting process, though**. Using one is, again, surprisingly straightforward – if you manage to work your way through the sometimes quite convoluted UI, especially on digital models.



To start using your scope, simply attach a probe to one of the channel inputs. These probes usually have two connectors on the other end: a big one that you operate by pulling the top part back – and a smaller one, which is usually a standard alligator clip. The latter needs to be connected to your circuit's ground rail, while you probe your oscillation with the former. Now what the oscilloscope will do is **monitor the voltage between the two connectors over time and draw it onto the screen as a graph**. Here, the x-axis is showing time, while the y-axis is showing voltage. You can use the device's scaling controls to zoom in on a specific part of your waveform.

Usually, digital oscilloscopes will also tell you a couple useful things about the signal you're currently viewing: minimum/maximum voltage level, oscillation frequency, signal offset. Some even offer a spectrum analyzer, which can be useful to check the frequencies contained in your signal.

BUILD GUIDE



MODULE ASSEMBLY APPENDIX

Before we start building, let's take a look at the complete **mki x es.edu Compressor** schematics (see next page) that were used for the final module's design and PCB fabrication. Most components on the production schematics have denominations (a name – like R1, C1, VT1, VD1, etc.) and values next to them. Denominations help identify each component on the PCB, which is particularly useful during **calibration, modification** or **troubleshooting.**

XS1 is the **Audio input** jack socket, **XS2** is the **Sidechain input** jack socket; it basically duplicates clock input, so you can use it to synchronize other modules to the clock of the Sequencer. **XS3** is the **Audio Output** jack socket. In our designs, we use eurorack standard 3,5mm jack sockets (part number WQP-PJ301M-12).

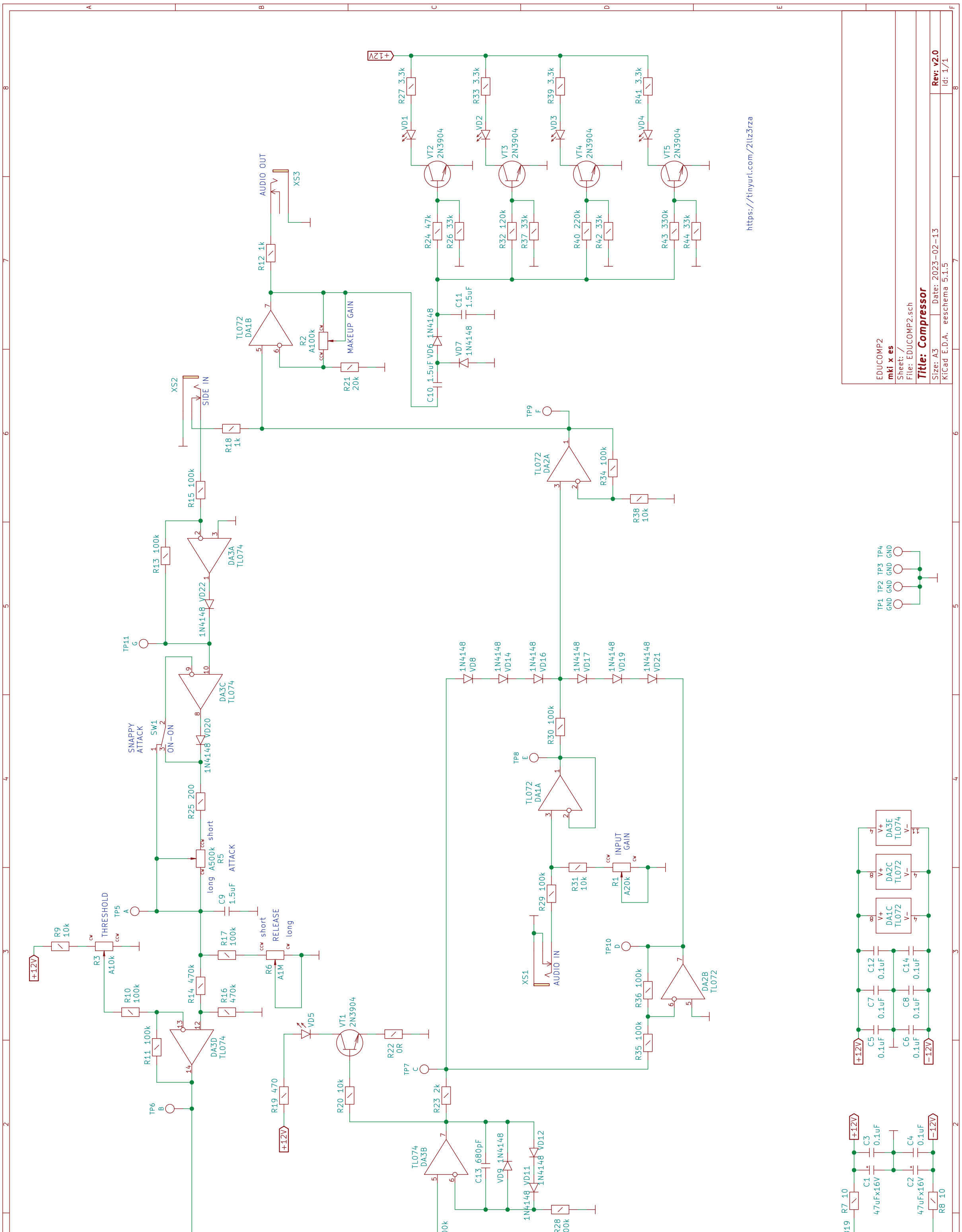
XP1 is a standard eurorack **power connector.** It's a 2x5 male pin header with a key (the black plastic shroud around the pins) to prevent accidental reverse polarity power supply connection. This is necessary because connecting the power incorrectly will permanently damage the module.

VD15 and **VD18** are **schottky diodes** that double-secure the reverse polarity power supply protection. Diodes pass current only in one direction. Because the anode of VD15 is connected to +12 V on our power header, it'll only conduct if the connector is plugged in correctly. If a negative voltage is accidentally applied to the anode of VD15, it closes, and no current passes through. The same goes for VD18, which is connected to -12 V. Because schottky diodes have a low forward voltage drop, they are the most efficient choice for applications like this.

Next, we have two **10 Ohm resistors (R7** and **R8)** on the + and – 12 V rails, with **decoupling** (or **bypass-**) capacitors **C3** and **C4**. These capacitors serve as energy reservoirs that keep the module's internal supply voltages stable in case there are any fluctuations in the power supply of the entire modular system. In combination with R7 and R8, the large 47 microfarad pair (C1 and C2) compensates for low frequency fluctuations, while C3 and C4 filter out radio frequencies, high frequency spikes from switching power supplies and quick spikes created by other modules. Often another component – a **ferrite bead** – is used instead of a 10 Ohm resistor and there's no clear consensus among electronic designers which works best, but generally for analogue modules that work mostly in the audio frequency range (as opposed to digital ones that use microcontrollers running at 8 MHz frequencies and above), resistors are considered to be superior.

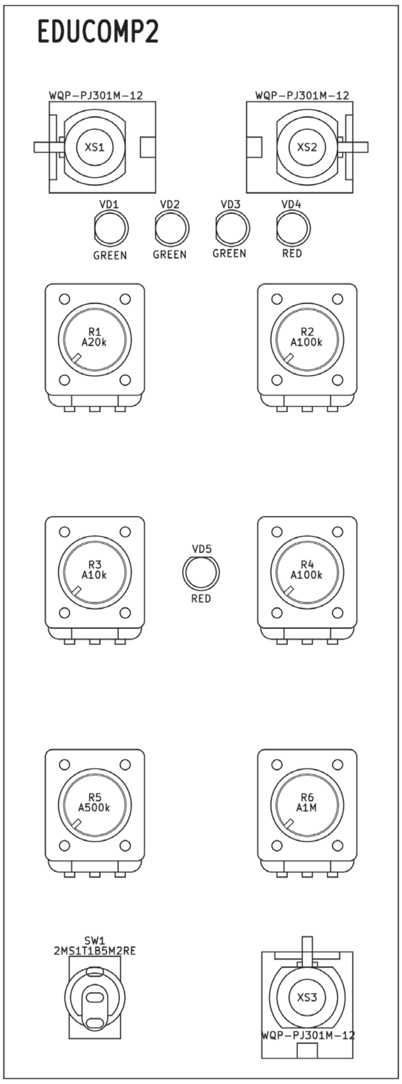
Another advantage of 10 Ohm resistors is that they will act like **slow "fuses"** in case there's an accidental short circuit somewhere on the PCB, or an integrated circuit (IC) is inserted backwards into a DIP socket. The resistor will get hot, begin smoking and finally break the connection. Even though they aren't really fuses, just having them there as fuse substitutes is pretty useful - **you'd rather lose a cent on a destroyed resistor than a few euros on destroyed ICs.**

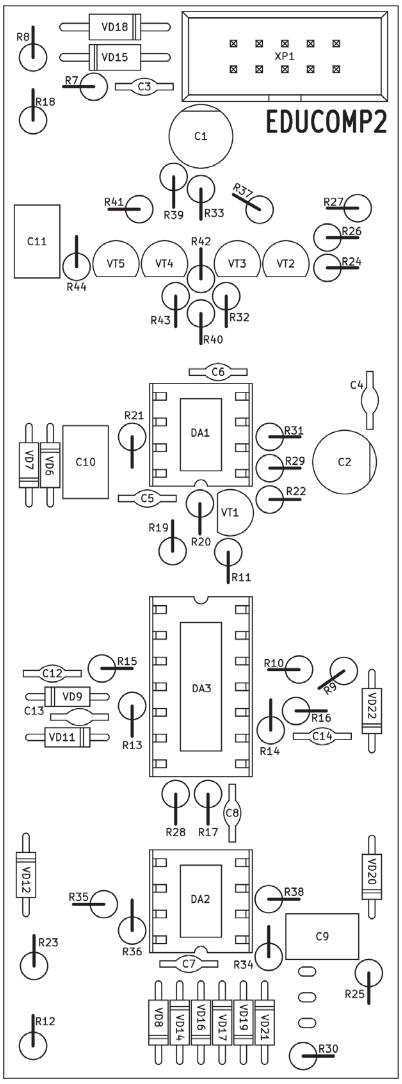
Capacitors **C5 – C8, C12, C14** are additional decoupling capacitors. If you inspect the PCB, you'll see that these are placed as close to the power supply pins of the ICs as possible. For well-designed, larger PCBs you will find decoupling capacitors next to each IC. Like the others, their job is to simply compensate for any unwanted noise in the supply rails. If the input voltage drops, then these capacitors will be able to bridge the gap to keep the voltage at the IC stable. And vice-versa - if the voltage increases, then they'll be able to absorb the excess energy trying to flow through to the IC, which again keeps the voltage stable. Typically, 0.1 uF capacitors are used for this purpose.

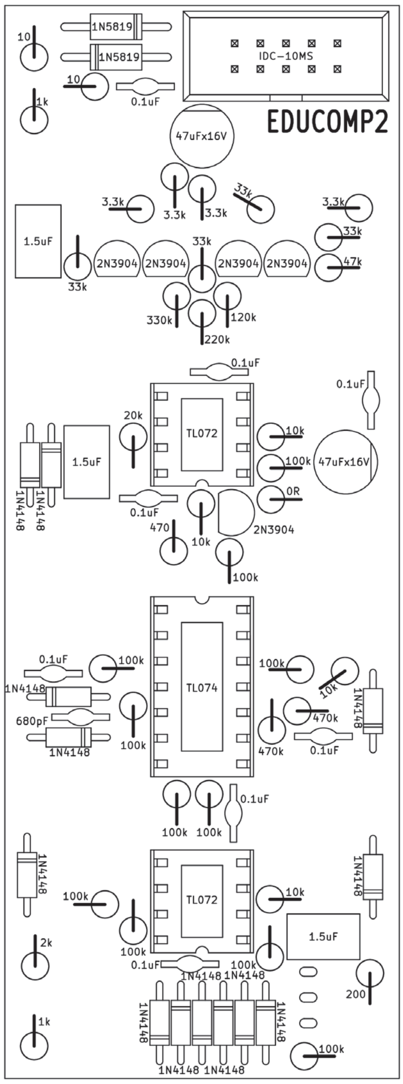

EDUCOMP2
mki x es
Sheet: /
File: EDUCOMP2.sch
Title: Compressor
Size: A3
Date: 2023-02-13
KiCad E.D.A. eeschema 5.1.5
Rev: v2.0
Id: 1/1
https://tinyurl.com/2llz3rza
AUDIO OUT
SIDE IN
AUDIO IN
MAKEUP GAIN
INPUT GAIN
THRESHOLD
SNAPPY ATTACK
ATTACK
RELEASE
RATIO

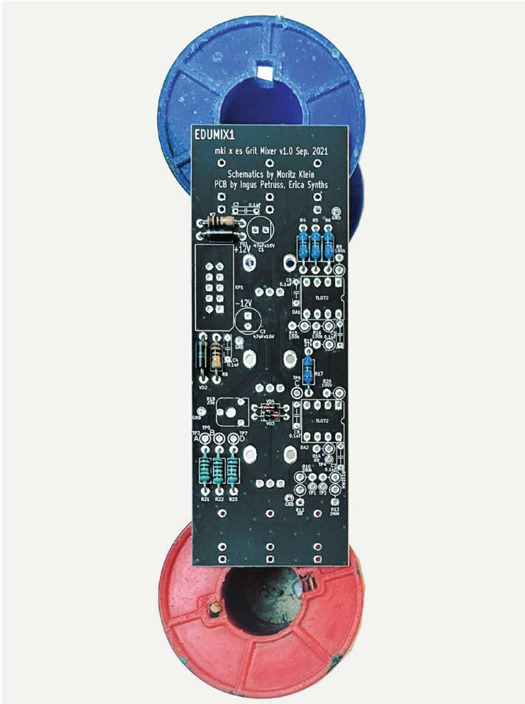
mki x es.edu

Before you start soldering, we highly recommend printing out the following part placement diagrams with designators and values. Because some of our PCBs are rather densely populated, this will help you to avoid mistakes in the build process.

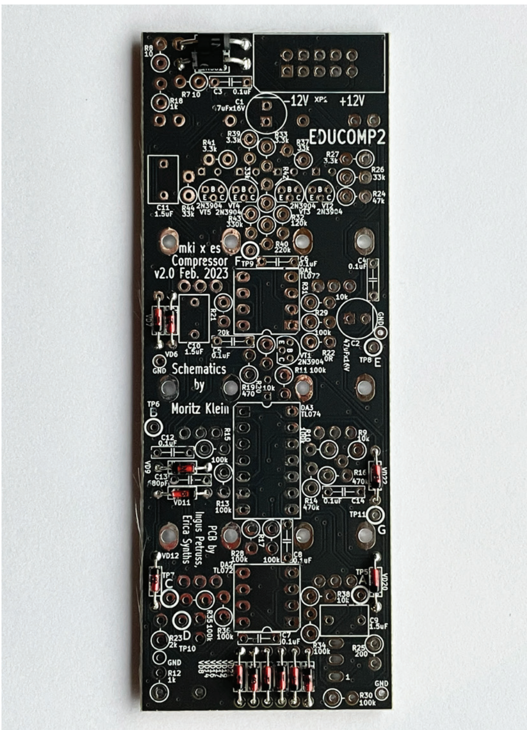




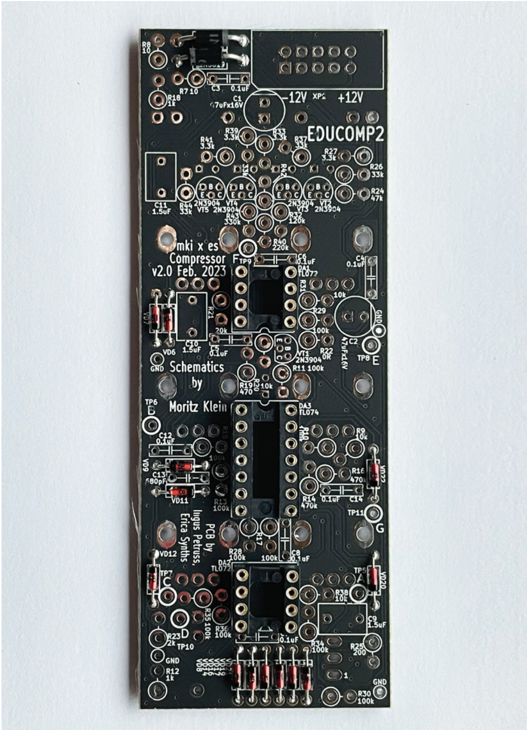




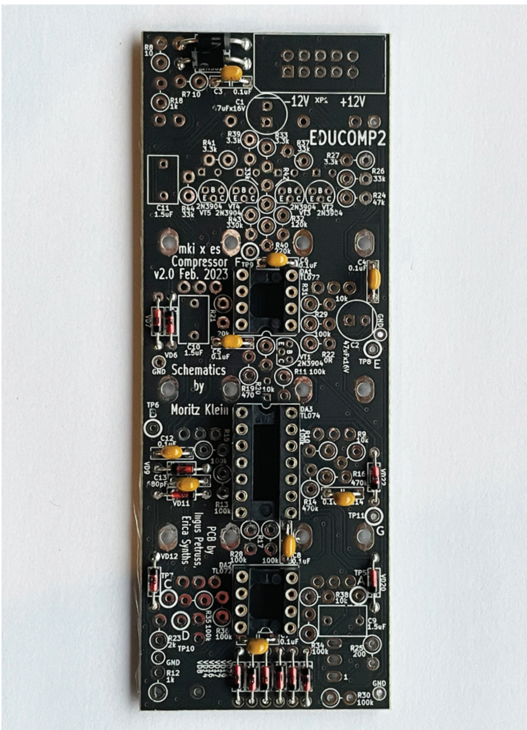
Place the Compressor PCB in a PCB holder for soldering or simply on top of some spacers (I use two empty solder wire coils here).



I usually start populating PCBs with lower, horizontally placed components. In this case, these are **switching diodes** and the **power protection diodes**. Bend the diode leads and insert them in the relevant places according to the part placement diagram above. All components on the PCB have both their value and denomination printed onto the silkscreen. Remember – **when inserting the diodes, orientation is critical!** A thick white stripe on the PCB indicates the cathode of a diode – match it with the stripe on the component. Flip the PCB over and solder all components. Then, use pliers to cut off the excess leads.

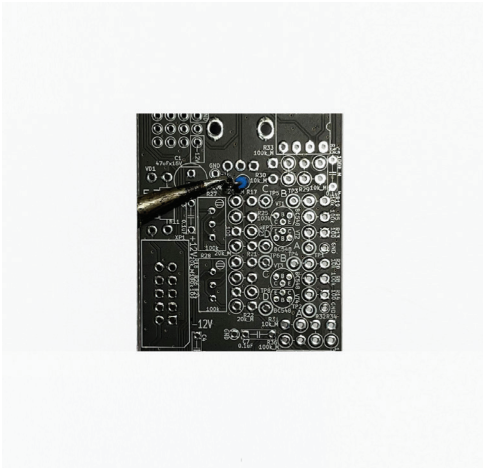


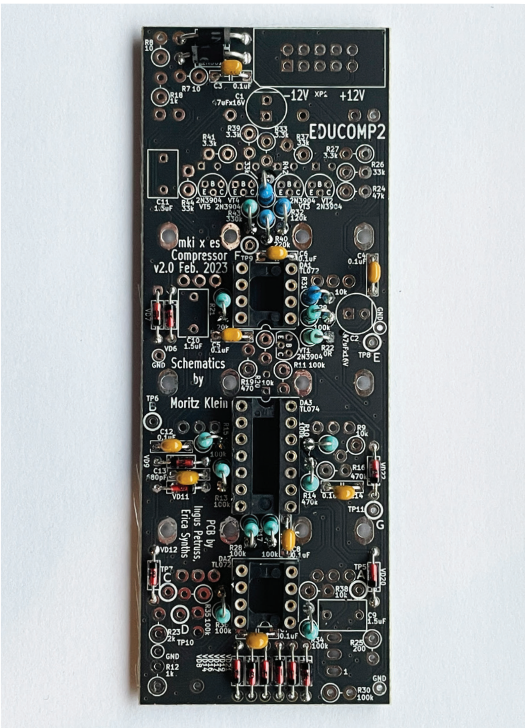
Next, insert the first DIP socket, hold it in place and solder one of the pins. Continue with the **next DIP socket**. Make sure the DIP sockets **are oriented correctly** – the notch on the socket should match the notch on the PCB's silkscreen. Now, turn the PCB around and solder all remaining pins of the DIP sockets.



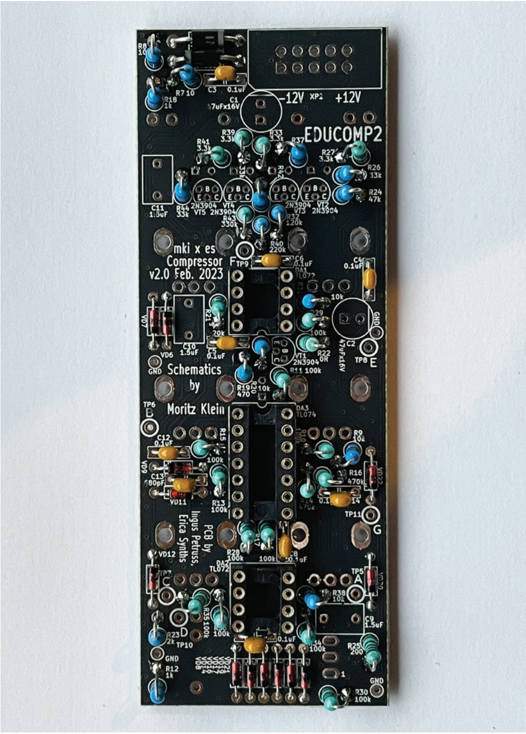
Then proceed with **the ceramic capacitors.** Place the PCB in your PCB holder or on spacers, insert the capacitors and solder them like you did with the resistors & diodes before. Now your PCB should look like this:

In order to save space on the PCB, some of our projects, including the Compressor, have **vertically placed resistors**. The next step is to place & solder those. We will solder them in two rounds – first, we'll solder those that will be hard to access later, then we'll solder rest of them. Bend a resistor's legs so that its body is aligned with both legs and insert it in its designated spot. Then solder the longer lead from the top side of the PCB to secure it in place, turn the PCB around and solder the other lead from the bottom. You can insert several resistors at once. Once done with soldering, use pliers to cut off excess leads.

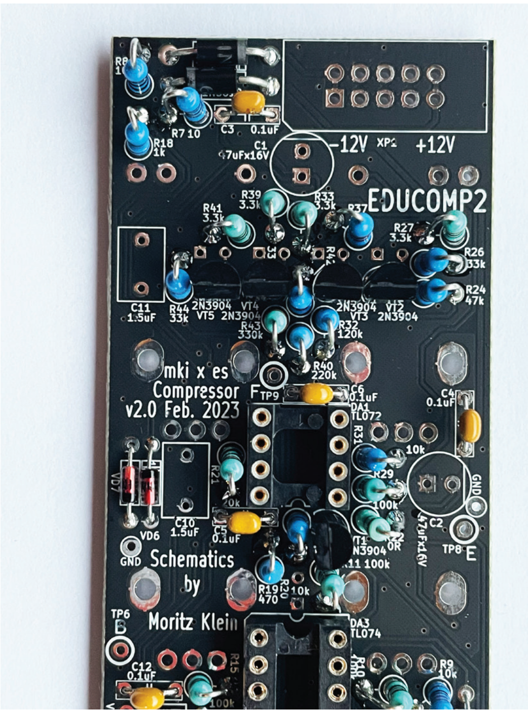




Solder the first set of resistors as on the photo below.

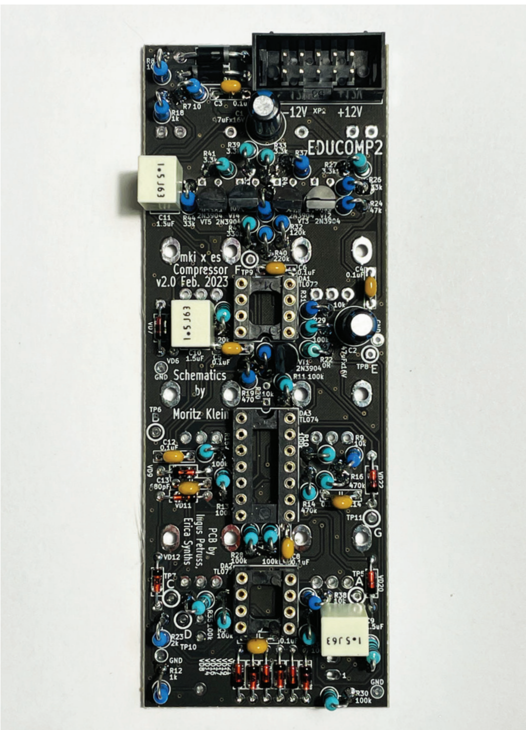


Then solder the rest of resistors. Once you are done with soldering all resistors, your PCB should look like this:

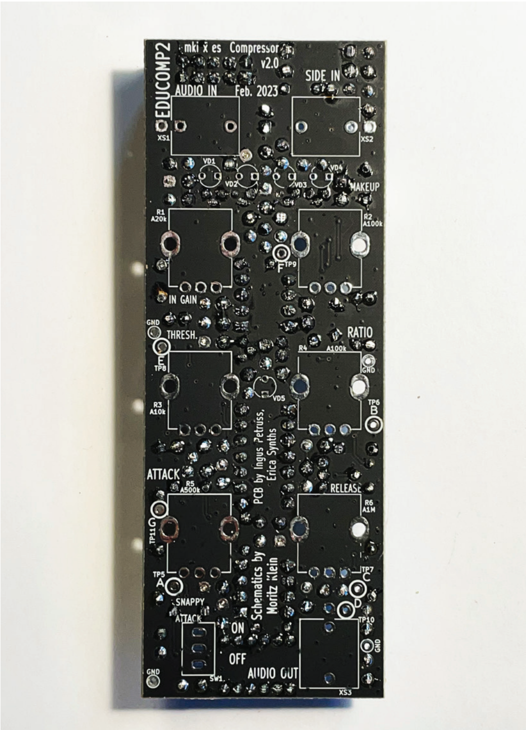


Next up: inserting & soldering transistors. Make sure you align the transistor with the marked outline on the silkscreen – **orientation is critically important here.**

Now, insert **film capacitors** and solder them. Then, insert **the electrolytic capacitors.** Electrolytic capacitors are bipolar, and you need to mind their orientation. The positive lead of each electrolytic capacitor is longer, and there is a minus stripe on the side of the capacitor's body to indicate the negative lead. On our PCBs, the positive pad for the capacitor has a square shape, and the **negative lead** should go into the pad next to **the notch** on the silkscreen.



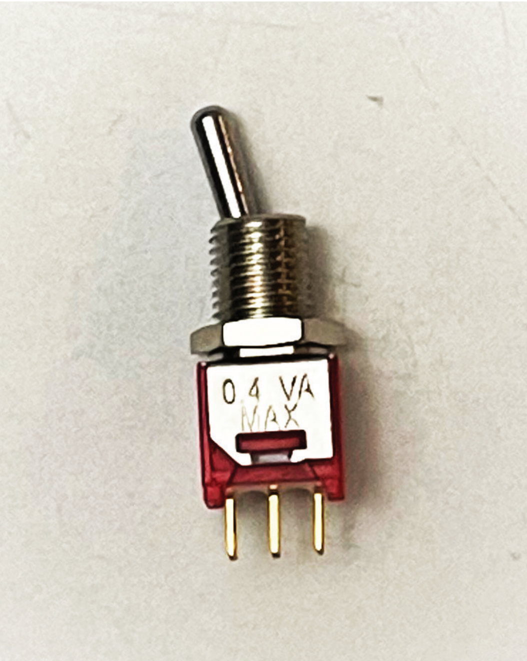
Then complete the component side of the Compressor PCB by soldering the **PSU socket.** Make sure the orientation of the socket is as shown in the picture below – the arrow pointing to the first pin is aligned with a notch on the silkscreen. **The key** on the socket will be **facing down.** Now your PCB should look like this:



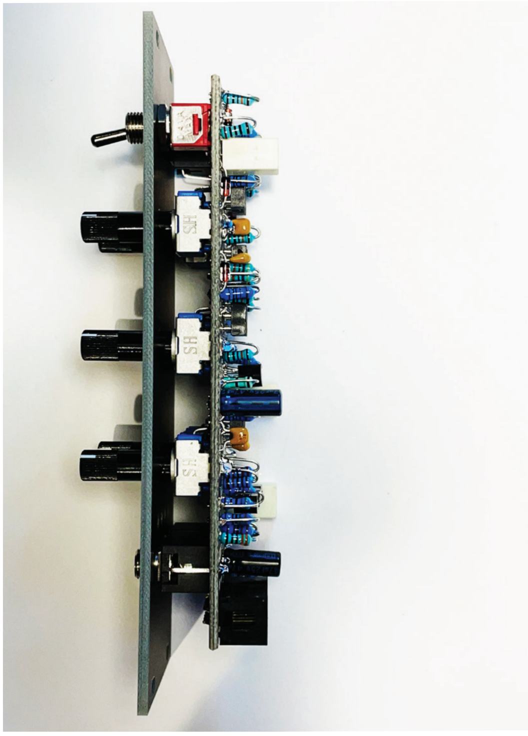
Now, turn the PCB around and inspect your solder joints. **Make sure all components are soldered properly and there are no cold solder joints or accidental shorts.** Clean the PCB to remove extra flux, if necessary.

Insert the jack sockets and solder them. Insert **the potentiometers** (make sure, you have right values!), **but don't solder them yet!**

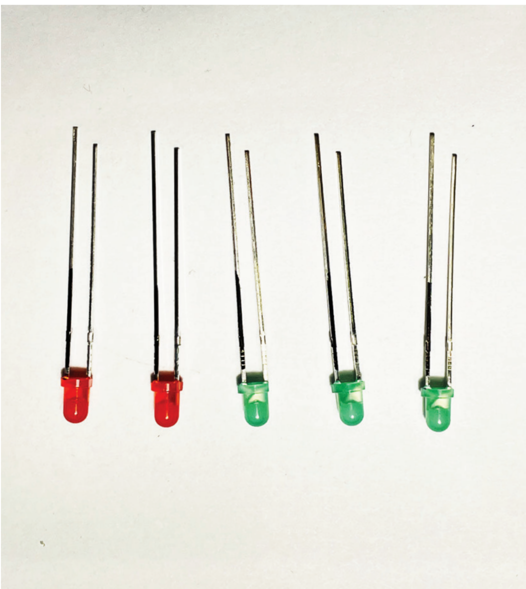
Fit the front panel (you may want to fix it with the nuts on the jack sockets) and make sure that the potentiometer shafts are aligned with the holes in the panel – and that they're able to rotate freely. Now, go ahead and solder the potentiometers.



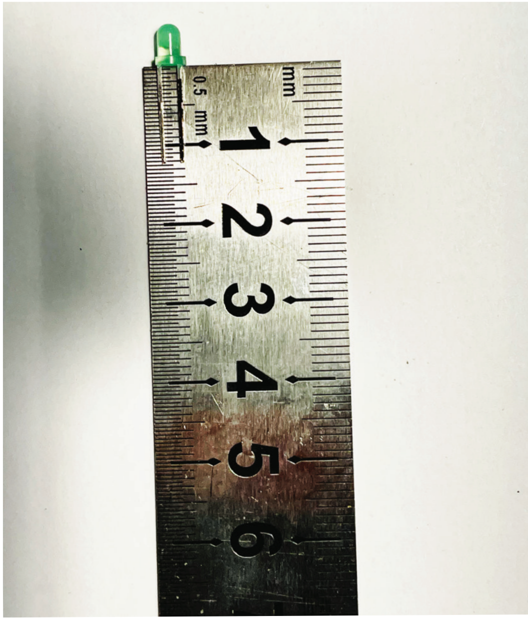
The **switches** require special attention. There are two nuts for the switch (they look identical to the jack socket nuts, but the thread is different). Screw on one of the nuts until it fixes itself on the bottom of the thread.



Now, insert the switch in the relevant place on the PCB, place the front panel, fix it with few nuts on the jack sockets and **solder the switch.**



The LEDs require special attention. Because of tight placement of components on the PCB, it's very challenging to solder them and cut off excess leads after soldering. Therefore we'll cut the leads to relevant length before soldering. First, use a permanent marker to note the anode (longer lead) of the LED.



Then cut the leads 12mm long.

Now, insert the LEDs in the relevant place on the PCB, but do not solder them, yet! Orientation of the LEDs is important – check the silkscreen! A notch on the silk-screen indicates the **cathode of the LED** (a shorter lead next to a notch on the LED) and the longer lead – the **anode of the LED**, we just **marked black**, – has to go into a hole with square-like polygon on the PCB.

Fit the front panel again and **fix it** with some nuts on jack sockets and one nut on the switch. Push the LEDs through relevant holes on the panel and solder them. We are almost done!

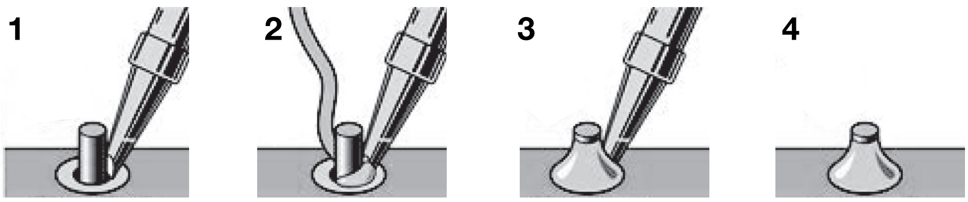
Now, **insert the ICs into their respective DIP sockets.** Mind the orientation of the ICs – match the notch on each IC with the one on its socket.

Congratulations! **You have completed the assembly of the mki x es.edu Compressor module!** Connect it to your eurorack power supply and switch it on. If there’s no "magic smoke”, it’s a good sign that your build was successful. The module does not require calibration, and you can use it straight away. It’s time to enjoy some musical results!

SOLDERING APPENDIX

If you've never soldered before – or if your skills have become rusty – it's probably wise to check out some **THT** (through-hole technology) **soldering tutorials on YouTube**. The main thing you have to remember while soldering is that melted solder will flow towards higher temperature areas. So, you need to make sure you apply equal heat to the component you are soldering and the solder pad on the PCB. The pad will typically absorb more heat (especially ground-connected pads which have more thermal mass), so keep your soldering iron closer to the pad on the PCB. It's critically important to dial in the right temperature on your soldering station. I found that about **320°C** is the optimal temperature for most parts, while for larger elements like potentiometers and sockets, you may want to increase the temperature to **370°C.**

Here's the recommended soldering sequence:

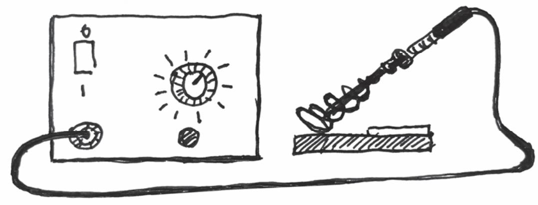


Heat part and pad 2 - 3 sec	Add solder	Continue heating 1 -2 sec.	Let cool

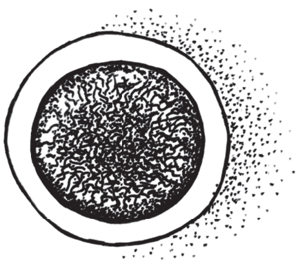
After you have completed soldering, inspect the solder joint:

Perfect	Too much solder	Not enough solder	Cold joint	Too much heat	Short

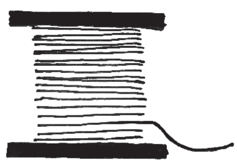
DIY electronics is a great (and quite addictive) hobby, therefore we highly recommend you invest in good tools. In order to really enjoy soldering, you'll need:



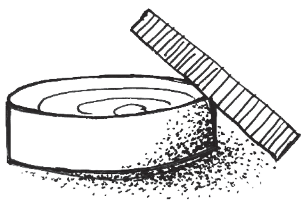
A decent soldering station. Top-of-the-line soldering stations (brands like Weller) will cost 200€ and above, but cheaper alternatives around 50€ are often good enough. Make sure your soldering station of choice comes with multiple differently-sized soldering iron tips. The most useful ones for DIY electronics are flat, 2mm wide tips.



When heated up, the tips of soldering irons tend to oxidize. As a result, solder won't stick to them, so you'll need to clean your tip frequently. Most soldering stations come with a **damp sponge for cleaning the iron tips** – but there are also professional solder tip cleaners with **golden curls** (not really gold, so not as expensive as it sounds). These work much better because they do not cool down the iron.

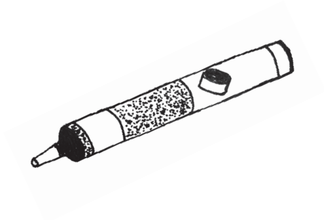


Solder wire with flux. I find 0,7mm solder wire works best for DIY projects.

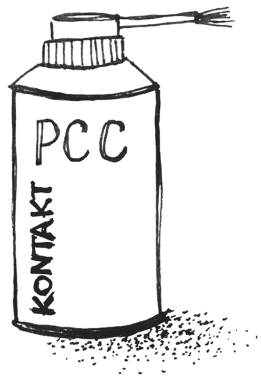


Some **soldering flux** paste or pen will be useful as well.

Cutting pliers. Use them to cut off excess component leads after soldering.



A solder suction pump. No matter how refined your soldering skills are, you will make mistakes. So when you'll inevitably need to de-solder components, you will also need to remove any remaining solder from the solder pads in order to insert new components.



Once you have finished soldering your PCB, it's recommended to remove excess flux from the solder joints. **A PCB cleaner** is the best way to go.

All of these tools can be found on major electronic components retailer websites, like Mouser, Farnell and at your local electronics shops. As you work your way towards more and more advanced projects, you'll need to expand your skillset and your tool belt – but the gratification will be much greater.

"I've been DJing a little bit, so you get used to the fact that music sounds brilliant when it's loud."

– Sophie Ellis Bextor