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!

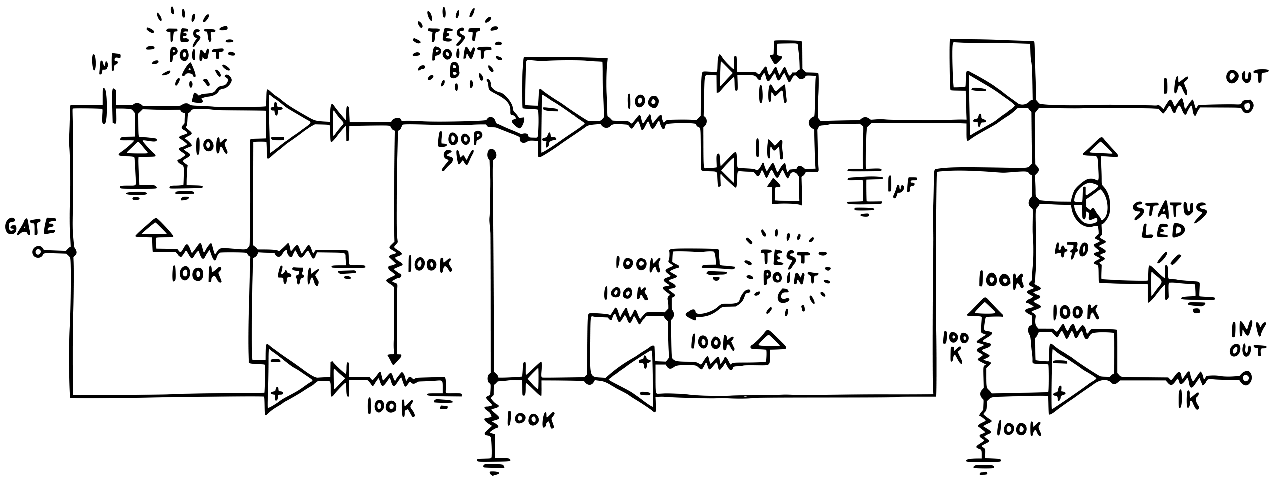
TABLE OF CONTENTS

CIRCUIT SCHEMATIC 2
BILL OF MATERIALS 3
POWERING YOUR BREADBOARD 5
CIRCUIT DESIGN CLOSE-UP 6
COMPONENTS & CONCEPTS APPENDIX 24
TOOLS APPENDIX 37
MODULE ASSEMBLY APPENDIX 40
SOLDERING APPENDIX 52

[1] Note that there's 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.

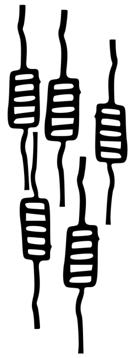
THE mki x es.edu ENVELOPE

Envelope generators are one of the most important sources of control voltage in any modular synthesizer. Though they might not generate audio signals themselves, many iconic sounds like the squelchy acid bass line, snappy techno kick drum and swelling dream pop pad wouldn't be possible if it weren't for envelope generators. So I knew I needed one for my DIY modular – which is why I created this wily little fake-ADSR.
Why fake? Because though it does send out a four-staged signal (**a**ttack/**d**ecay/**s**ustain/**r**elease), the duration of decay and release are tied together, controllable only via the same knob. To make up for this limitation, I decided to include both a looping function and an inverted output as extra goodies.



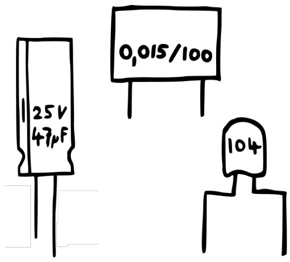
BILL OF MATERIALS

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:



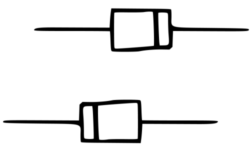
An array of resistors. The specific values (in ohms, which you should check for with a multimeter) are

100k	x10
47k	x1
10k	x1
1k	x2
470	x1
100	x1
10	x2



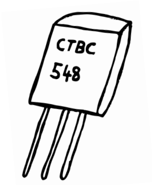
A few capacitors. The specific values (which are printed onto their bodies) are

47 µF (electrolytic)	x2
1 µf (1J/film)	x2
100 nF (104/ceramic)	x8



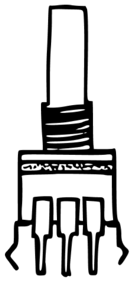
Some diodes. The specific model names (which are printed onto their bodies) are

SB140 (schottky)	x2
1N4148 (signal)	x6



A transistor. The specific model name (which is printed onto its body) is

BC548 (NPN)	x1



A few regular potentiometers. The specific values (which may be encoded & printed onto their bodies) are

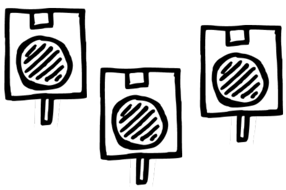
1M (A105) x2
100k (A104) x1

A switch. The specific model (which you can identify by the number of connectors on its underside) is

Single pole, double throw x1

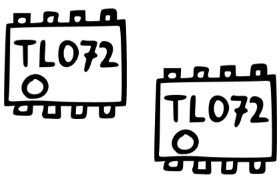
An LED (light emitting diode). The specific model (which you can identify by measuring the body's width) is

3mm (red) x1



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

Switched mono (black) x3



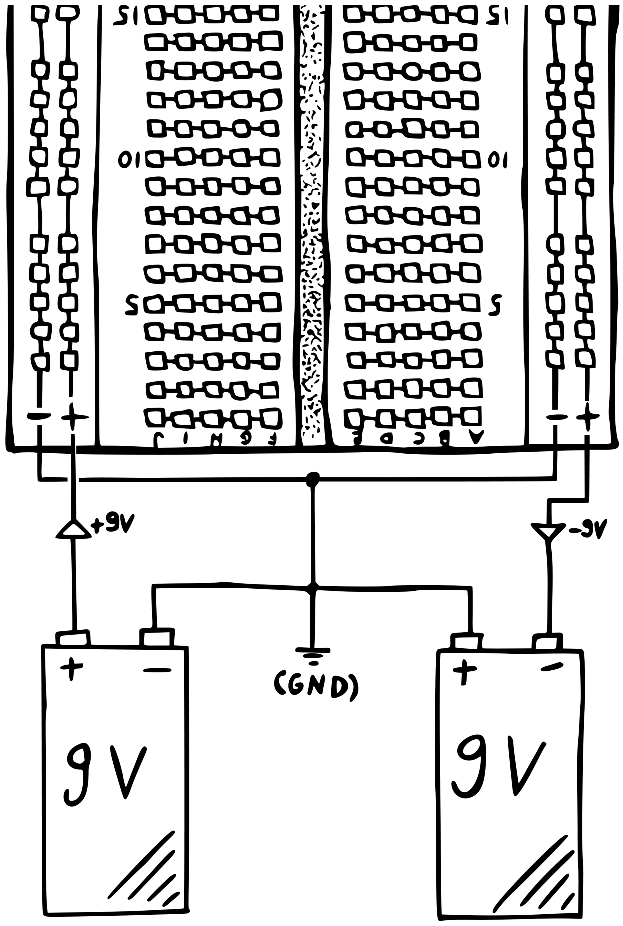
A couple of chips. The specific models (which are printed onto their bodies) are

TL072 (dual op amp) 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 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.** 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.[2] **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.

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

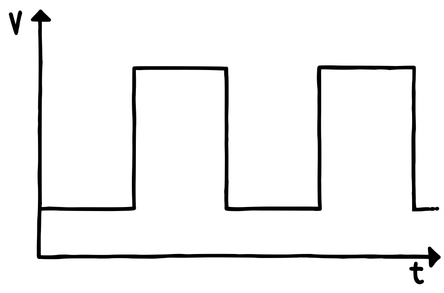
ENVELOPE BASICS

If you're new to modular synths (or even synths in general), you might ask: what's an envelope generator – and what do we use it for? Simple. We've only got two hands, best case. So there's a hard limit to how many of our synth's knobs we can turn and parameters we can tweak at the same time. Also, those hands are generally sluggish, sloppy and imprecise.

Wouldn't it be much better if we had some virtual, programmable robot hand to help us out? That would do what we tell it to do, reliably and precisely, whenever we want it to? If your answer is yes, then you might be a control freak. Also, you'd probably want an envelope generator. So let's build one!

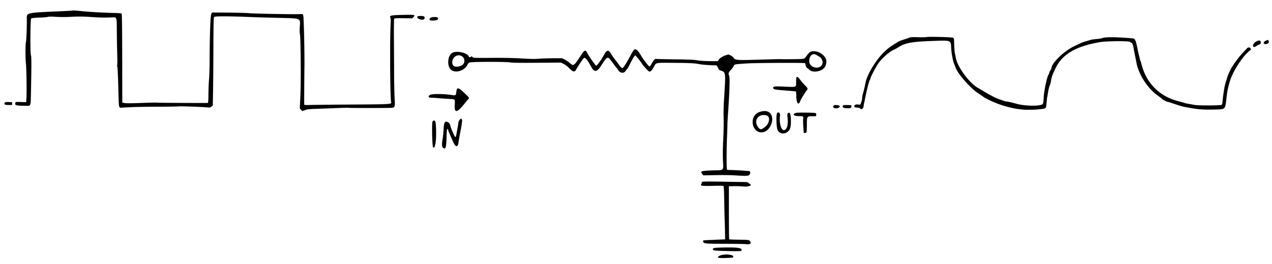
But before we can do that, we have to understand what an envelope does, exactly. How does it tweak parameters on another module? **Well, when designing oscillators, filters and amplifiers, you'll notice that people go to great lengths in order to make them voltage controllable**. This comes in handy for us. Because if we want a filter module to open up, for example, we can simply send a high level voltage into its CV (control voltage) input. Now of course, if we only want to open the filter (and keep it open), we could send in a fixed 12 V signal and be done with it. That's not all that useful, though. Normally, you'd want the filter to close at some point, too. So why not simply use a square wave oscillator here?

THE SIMPLEST ENVELOPE



Since a square wave is really just an oscillation between a high- and a low level voltage, this would indeed cause our filter to open up and close down. Even better – it would do so rhythmically as the oscillator cycles through its phases. And while this does work, it has two severe limitations. First: the opening and closing movements are always instant, with no way of making them any more gradual. This might be what you want in some contexts, but often, you'd need something less abrupt. And second: We can't change the rhythmic pattern – all we get is a constant staccato.

So what can we do about that? Well, the most obvious thing is to try and make the rising and falling edges less steep. **In envelope terms, we'd say that we want to slow down the attack and extend the release.** And while this may sound complicated, it's actually anything but.



All we need are two components: a resistor and a capacitor, set up like this.[3] Now, if you've worked with analog filters before, this setup should look strikingly familiar. **That's because this is really just a bog-standard passive low pass filter**. And what it does to our square wave is exactly what we said we're after: it takes the rising and falling edges and makes them less steep.

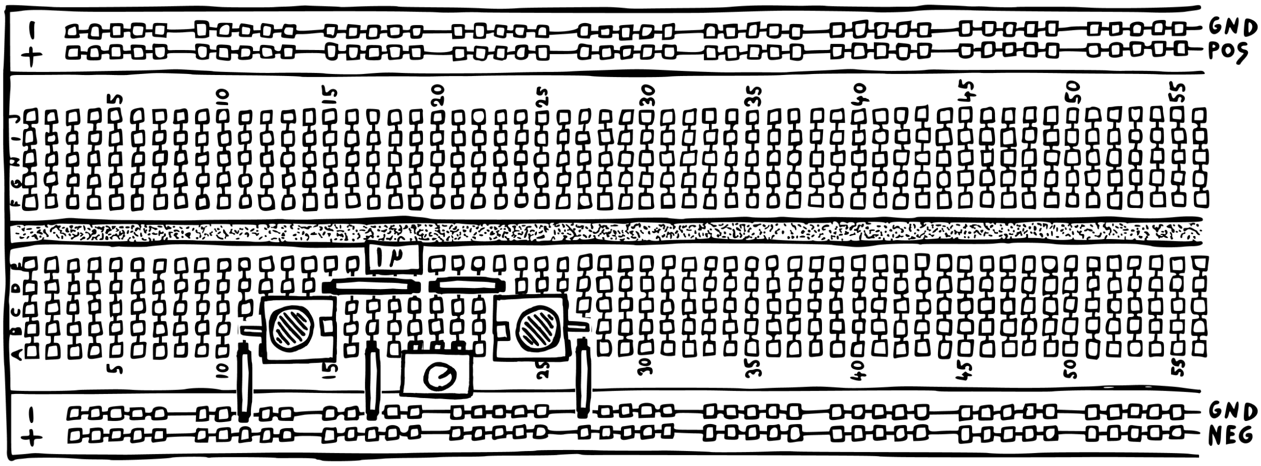
Here's how it works. Once the voltage at the input switches from low to high, a current will be forced through the resistor and into the capacitor, slowly filling it up. As the capacitor is being charged, the voltage at the output slowly rises until the cap is completely filled up, and the input- and output voltages align. Then, when the input voltage swings low, the whole process reverses. Now the capacitor will push its contents through the resistor and into the input, so to speak. This happens because the voltage on the right is much higher than the voltage on the left. As the capacitor empties out slowly,

[3] Read more about resistors and capacitors in the components & concepts appendix (page 26/27).

the two voltages align again. As you can see, this will turn our square wave input into something like a very basic attack-release envelope.[4]

But before we can try this, we'll have to think about appropriate values for the capacitor and resistor. Since we are dealing with a very, very slow square wave oscillation at the input, they'll need to be pretty big. Otherwise, the effect will be so minimal that we won't be able to tell the difference.

Also, we'll probably want to adjust the effect's intensity. For that, we've essentially got two options here: we could change either the capacitor- or resistor value. But since the former can only be done by hand, and switching components is not the most user-friendly strategy, **we're going to replace the fixed resistor with a potentiometer, set up as a variable resistor**.[5] This way, we can adjust the resistance (and thereby the steepness of our envelope's rising and falling edges) on the fly. A 1M pot should give us a decent enough range here.



To properly test this circuit, you'll need a square wave LFO[6] and a module with a CV input like a VCF. Send in the LFO via the right-hand socket, while connecting the other one to your filter. By turning the potentiometer's knob, you should be able to dial in a more or less intense effect. Great! **The only problem with this is that the attack- and release phases are not adjustable independently**. Changing one will always also change the other.

---

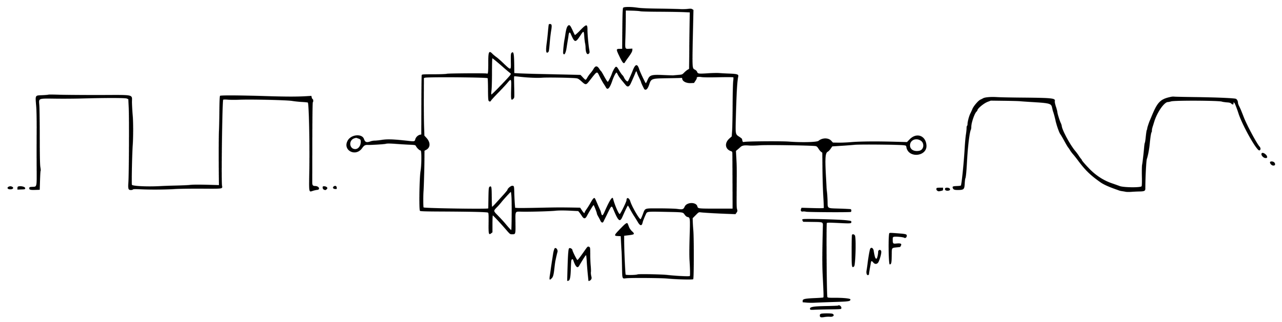
[4] You can try this chapter's circuits in a circuit simulator. I've already set them up for you right here: https://tinyurl.com/y7ea7j3a – you can change all values by double clicking on components.

[5] Read more about potentiometers in the components & concepts appendix (page 30).

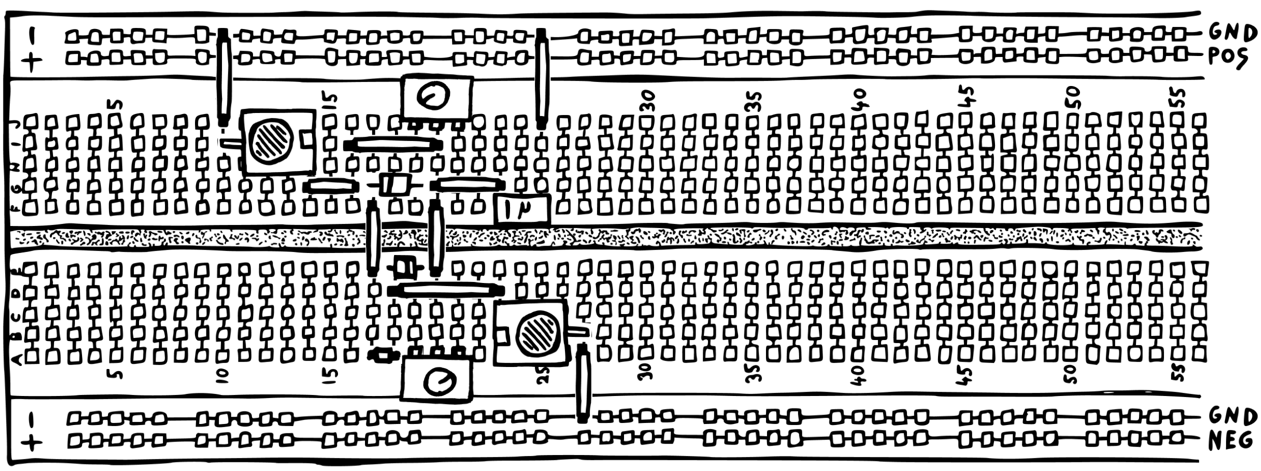
[6] A clock module or sequencer with a gate output will also work.

PASSIVE A/R ENVELOPE

So how can we separate the two? It's actually really easy. All we need are two diodes and another 1M potentiometer.



Here's how this works. **Diodes are basically one-way streets for electricity**.[7] **So by putting two of them in parallel, facing in opposite directions, we are taking a two-way street and splitting it into two one-way streets**. Where before, our capacitor was charged and discharged through the same resistor, now each phase gets their own.
So when the input signal swings high, a current will flow through the top diode – and only that diode – through the top potentiometer and into the capacitor. During the low phase, the current will take the other path. This means that one potentiometer now controls the attack – and the other controls the release.[8]



If you connect your LFO and VCF like before, you should be able to sculpt the filter movement much more freely by adjusting attack and release independently. So what we've built here is an ultra-simple passive attack-release envelope generator. Why passive? **Because it does not include any form of amplification**. It's getting all of its

[7] Read more about diodes in the components & concepts appendix (page 28).

[8] You can try this chapter's circuits in a circuit simulator. I've already set them up for you right here: https://tinyurl.com/y7xguhoo – you can change all values by double clicking on components.

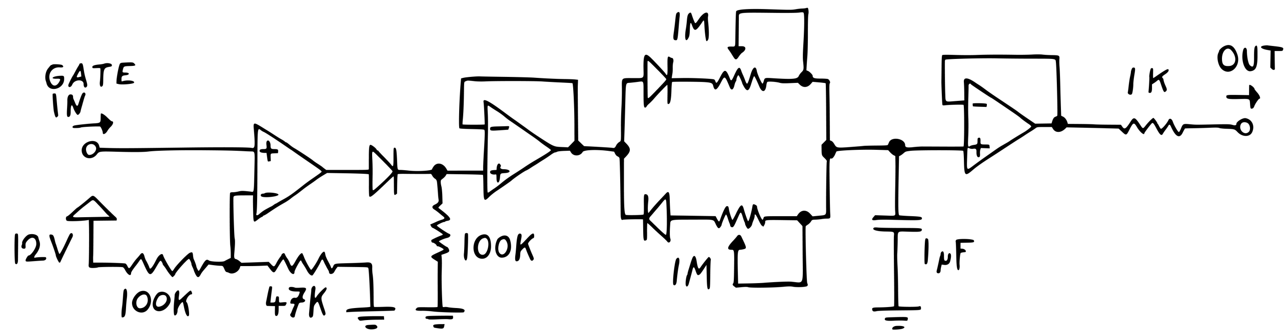
power from the square wave LFO. Is this a problem? That very much depends on what your goals & context are. In this current form, the envelope can only properly function if two external conditions are met: the circuit triggering it (in our case, the LFO) needs to be able to provide enough current. And the circuit we're controlling (in our case, the filter) needs to draw as little current as possible from our envelope.

Why? Let's imagine we put a big resistor between our LFO and the envelope. This way, we are severely limiting the amount of current flowing into our circuit. And that means that even if we dial our attack and release pots all the way down, charging and discharging our capacitor will not be instant – as we'd expect –, but instead would take a while.

On the other side, imagine our filter was also using the envelope to drive an LED. LEDs, if you don't know, are basically just diodes that light up when a current flows through them. Since they're pretty power-hungry, our filter's LED would eat up most of the current coming through the attack pot, preventing our capacitor from ever really being charged up. And that would severely restrict our envelope's range. Now granted, this is a worst case scenario. **Well-designed modules should always have high-current outputs and low- to no-current inputs**. Which, ironically, is a standard our envelope here does not live up to at all. It eats up our oscillator's signal, while providing barely anything for the filter.

ACTIVE A/R ENVELOPE

Thankfully, fixing that is really straightforward. **We'll simply set up op-amp buffers both at the in- and output.**[9] Buffers measure a voltage and provide an identical copy at their output, while being able to supply a decent amount of current.[10] Because of that, it no longer really matters how much current the input can provide – and how much the next circuit eats up.



Cool! But why the 1K resistor at the output, then? And what's all this additional stuff at the input? Well unfortunately, there are other worst case scenarios we need to consider. The first one of which being a classic user error. Imagine that user plugs our envelope's output into some other module's output by accident. If that other module also uses a buffer there, we'd basically create a short circuit, since buffers can not only source (i.e. send out), but also sink (i.e. absorb) plenty of current. **So by placing a 1k resistor before the output socket, we make sure that in this scenario, the maximum amount of current flowing is limited**. Saving our op amp from a potential early grave.

Okay, easy – so now, let's tackle the additional op amp on the left. What kind of problem does it fix? Well, while we did make sure that our circuit gets enough current, we haven't thought about the voltage we're feeding it yet. We should, though, because that voltage will determine the voltage range across which our envelope is operating.

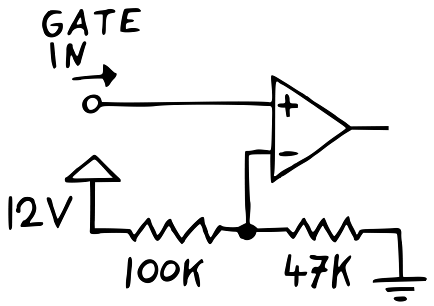
Think of it this way: **if all our envelope does is take the input signal and make the rising and falling edges less steep, then the maximum „height" of the resulting curve is completely determined by that input signal**. Why's that a problem? Because if the input signal would, for example, just swing between 0 and 1 V, that curve would be really small. And a smaller curve means a reduced range of effect. When controlling our filter, for example, this small curve would barely be able to move the cutoff point. So basically, our envelope would behave very differently depending on what kind of circuit we use to

[9] You can try this chapter's circuits in a circuit simulator. I've already set them up for you right here: https://tinyurl.com/ycz4qc3b – you can change all values by double clicking on components.

[10] Read more about op amps and buffers in the components & concepts appendix (page 33).

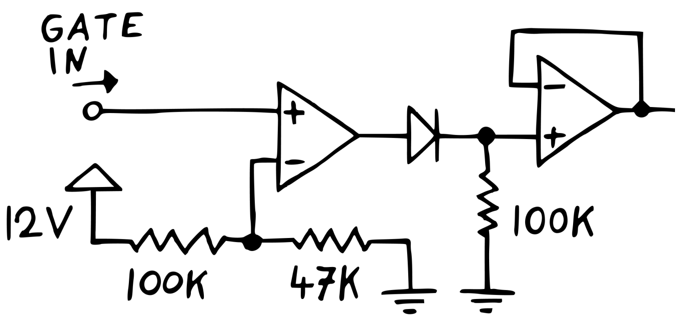
drive it. And for me, that would get very annoying very fast. But since it's very easy to eliminate this kind of external dependency, we'll get rid of it.

To do that, we use another op amp, which we set up in the comparator configuration. **A comparator, if you don't know, basically just looks at an input voltage, compares that input voltage to a reference voltage, and then tells us which one is higher**. How does it tells us? By either pushing its output voltage up to the positive or pulling it down to the negative supply rail. So in our case, that would be either + or –12 V.[11] Here's how this particular setup works in detail.



I've set up a voltage divider to get our reference voltage. A 100k/47k combination gives us approximately 3.8 V to work with. So whenever our input voltage is higher than that, the comparator's output will jump to +12 V. And if it's lower, it drops down to –12 V. Why did I choose that exact threshold? To be honest, mostly just because I had packs of 100k and 47k resistors lying on my table when I was testing this. But I still feel that 3.8 V is a decent value here. It's low enough so that any sequencer should be able to trigger the comparator, but definitely high enough to prevent it from firing randomly because of electromagnetic interference.

Okay, so now our envelope will always get the same 12 V to work with – as long as our input signal passes the 3.8 V threshold. But what about the comparator's low state output? Once the input drops below the threshold, it will swing down to –12 V. **This is not ideal, because traditionally, the base line for an envelope's output is supposed to be 0 V**. Which is why we'll put a diode, followed by a 100k resistor to ground, between our comparator's output and the buffer's input.



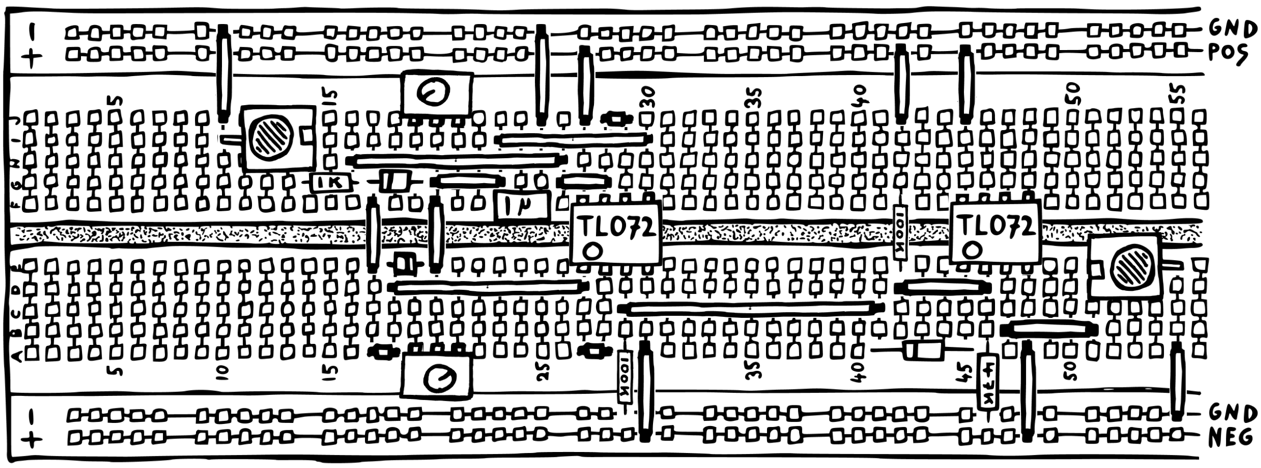
Here's what that does. Whenever the comparator is pushing out 12 V, the diode conducts and we also get about 12 V at the buffer's input. But once the voltage turns negative, the diode will block. Normally, the buffer's input would now be undefined (or „floating"). But since we have a 100k resistor to ground there, that input gets pulled down to 0 V instead.

---

[11] I keep talking about +/– 12 V as the positive/negative rail voltages because I use a +/– 12 V power supply myself. You can substitute this with +/– 9 V all throughout the manual if you're using batteries.

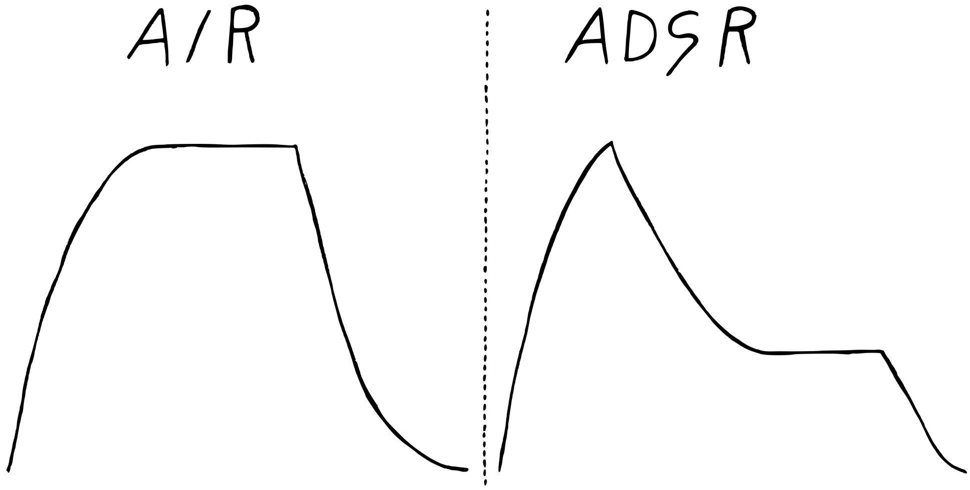
And with this, we've now forced the envelope's operating voltage range to always be between 0 and 12 V. Great! To see if that actually works, let's try it on the breadboard.

Make sure that the TL072-chips (which house two standard op amps each) are set up exactly as shown here – if you reverse the power connections, they will heat up and die!



If you now go ahead and test this, everything should be working pretty much like before – though the range of filter movement will be significantly increased.

AR VS. ADSR ENVELOPES

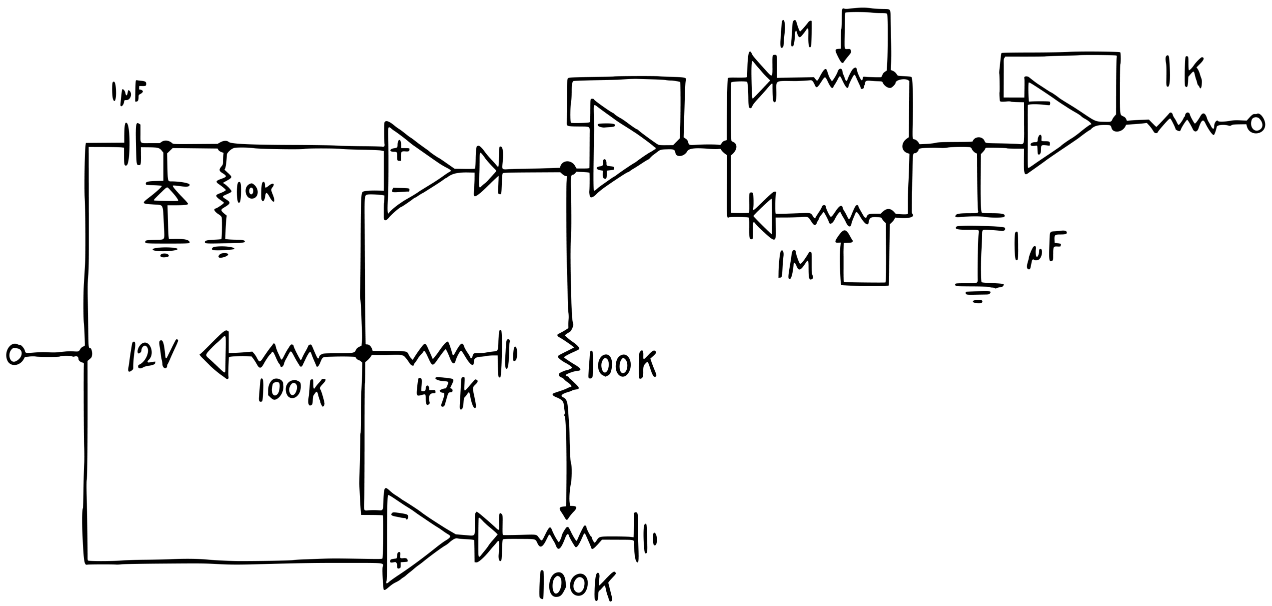


And while we could leave it there, I'd rather put in a bit more effort in order to give us finer control over the envelope curve's shape. On the left side here, I've drawn up what our current circuit is capable of producing: a simple attack-release-curve. On the other side, we have a more complex attack-decay-sustain-release curve. What's the difference? Well, while both curves have an attack and a release phase, **the one on the right adds a decay phase and the ability to set a specific sustain level**. The idea here is this. If the sustain is set to a lower value than the envelope's peak, we get a drop after the attack. This is the decay phase. Once that's through, the curve settles on the set sustain level while the input signal stays high. From here, we enter the release phase once the input swings low.

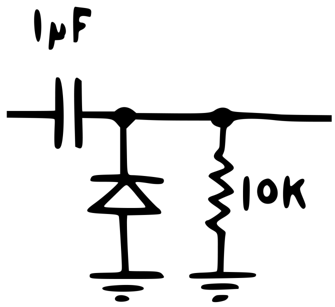
What's the benefit of this added complexity? Simple: we can produce a wider variety of sounds. Personally, I really like short, plucky, percussive hits – and also gliding acid bass lines. Both of them are not really doable with a simple attack-release envelope.

Now, turning the circuit we have into a proper ADSR envelope is somewhat out of scope for this project. **Still, with very little extra effort, we can can build something that approximates it**.

THE PSEUDO-ADSR ENVELOPE



Here's how that would work. As you can see, I've basically just copied our comparator and pasted it down at the bottom. Both the new and the original one get the input signal, and they share the same reference voltage, but I've placed a high-pass filter before the top one. That high-pass would normally turn a square wave cycle into two short voltage spikes.



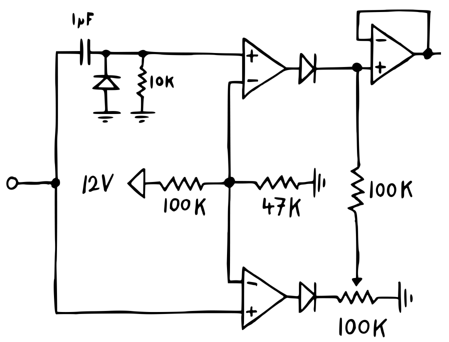
First a positive one, when the input transitions from low to high – and then a negative one when it drops from high to low. Now, since we're not interested in the negative spike, and it could cause our comparator to glitch out under certain circumstances, I've decided to eliminate it using another diode. Whenever the voltage at the comparator's input (i.e. after the 1 µF capacitor) tries to go negative, the diode will open and thereby neutralize it.[12]

Okay, but what do we need the positive spike for? Simple. **By feeding this positive spike into our comparator, we get a quick 12 V burst right when the envelope is triggered**. So with a fast attack, the envelope's curve will always start at the peak level.

---

[12] You can try this chapter's circuits in a circuit simulator. I've already set them up for you right here: https://tinyurl.com/yc2c8u5h – you can change all values by double clicking on components.

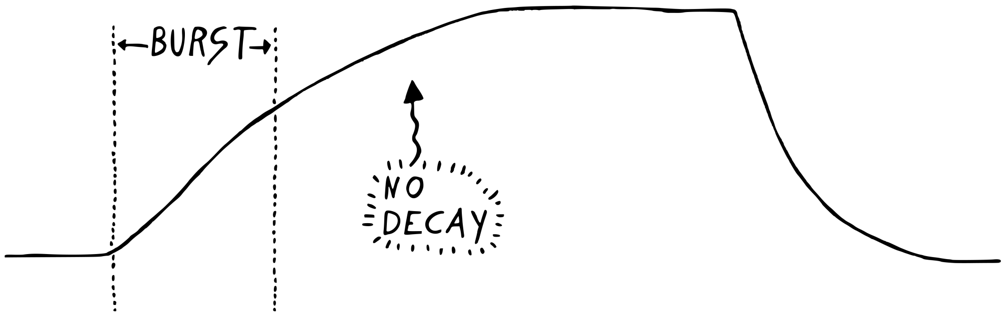
Why's that important? To answer this, we'll first have to talk about the other comparator. Since it doesn't have a high-pass at its input, it will simply behave like the comparator in our previous iteration. Whenever the input voltage is above 3.8 V, we'll get a constant 12 V at its output. The diode afterwards serves the same purpose as the one up top – it blocks the comparator's low state.
After this, I've set up another potentiometer as a variable voltage divider. This allows us to take the 12 V during the comparator's high phase and freely scale them to any value between those 12 and 0 V. Whatever voltage we dial in here will be our sustain level. Why? **Because the 100k resistor at the input buffer doesn't connect straight to ground like before, but rather to our sustain level voltage**. If that sounds confusing, let's break it down step by step.



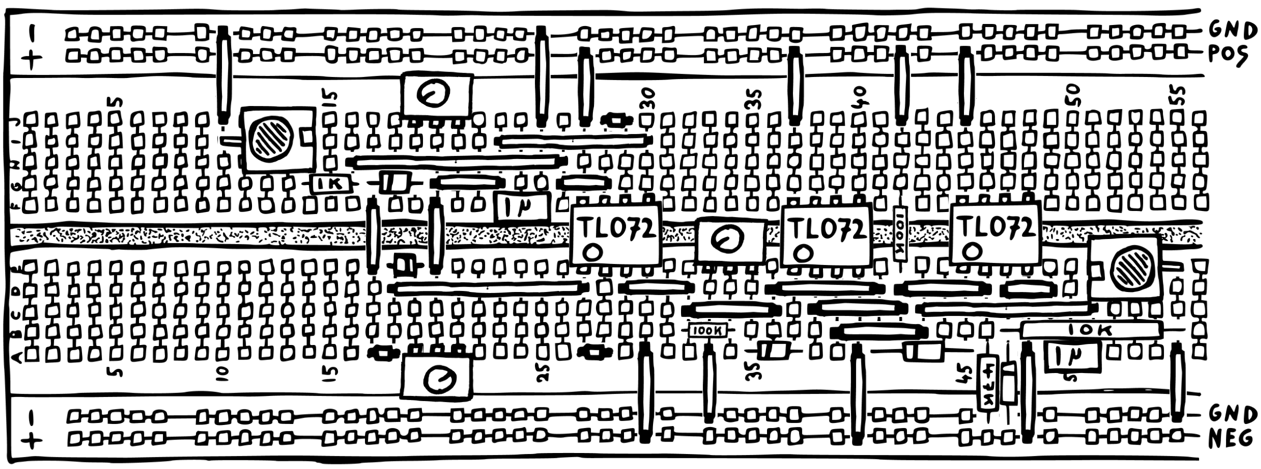
Our input starts out low. This means that both our comparators' outputs sit at –12 V. But because of the two diodes, this doesn't propagate, and so our buffer's input gets pulled down to 0 V through the 100k resistor and potentiometer. Giving us 0 V at the envelope's output as well. Next, let's assume that the input goes high. This will do two things: we'll get a voltage spike after our high pass, which gets converted into a short 12 V burst by the top comparator. Simultaneously, the other comparator pushes out a constant 12 V that get scaled down by our sustain potentiometer.
Let's assume we've set it to about 50 %. This means that at the buffer's input, we've got our 12 V burst coming from the top, and a constant 6 V coming from the bottom comparator. Since there's no resistor in the top path, but a 100k in the bottom one, the burst will „win" and push the overall voltage at the buffer's input up to about 12 V.
Our buffer – being a buffer – will copy those 12 V and push them through the attack potentiometer. Again assuming that we've dialed in a fast attack, there will be a pretty low resistance in its path, allowing the capacitor to be charged up to about 12 V before the burst is over. So at this point, our envelope's output sits somewhere around 12 V – its peak value. But because the burst is a burst, it'll quickly die down and the top comparator's output will drop to –12 V. **So suddenly, with the top diode blocking, the only voltage applied to the buffer's input is our sustain level: 6 V**. This means that our buffer's output will drop from around 12 to those 6 V, allowing our capacitor to partially discharge through the release path. Because remember – the burst charged it up to around 12 V.
Once the voltage at the capacitor has dropped to the sustain level, it will stabilize, giving us a constant 6 V at the envelope's output. Until the input signal swings low, our buffer's output drops to 0 V, and the capacitor is allowed to complete its discharging process. The result is an output curve with 4 distinct phases: attack, decay, sustain and release. Now as you might have noticed, there are two rather big caveats here. **First, both in the decay- and the release phase, the capacitor discharges through the same**

potentiometer. This means that you can't control those two phases individually – changing one will also change the other. So we can never have a curve with a long decay and super short release.

Second, the decay phase is directly dependent on the set attack. Why's that? Simple. If we dial in a slow attack, the capacitor will not be charged up to the peak level during the short initial burst. Maybe it won't even reach the set sustain level, let alone surpass it.



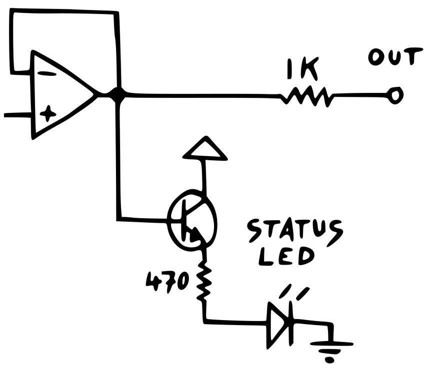
In effect, we basically skip the decay phase, as we are never dropping down to the sustain level. Now granted, a proper ADSR envelope shouldn't have these two problems. But for how simple our circuit is, and how few components it uses, I think we can book this as a worthwhile tradeoff. So let's set this up on the breadboard!



Try playing around with the new sustain knob, and test how it interacts with the other two. You should be able to dial in a pretty wide range of filter movements now. Great! But it might be nice if we had some visual indication of the voltage our envelope is sending out, right?

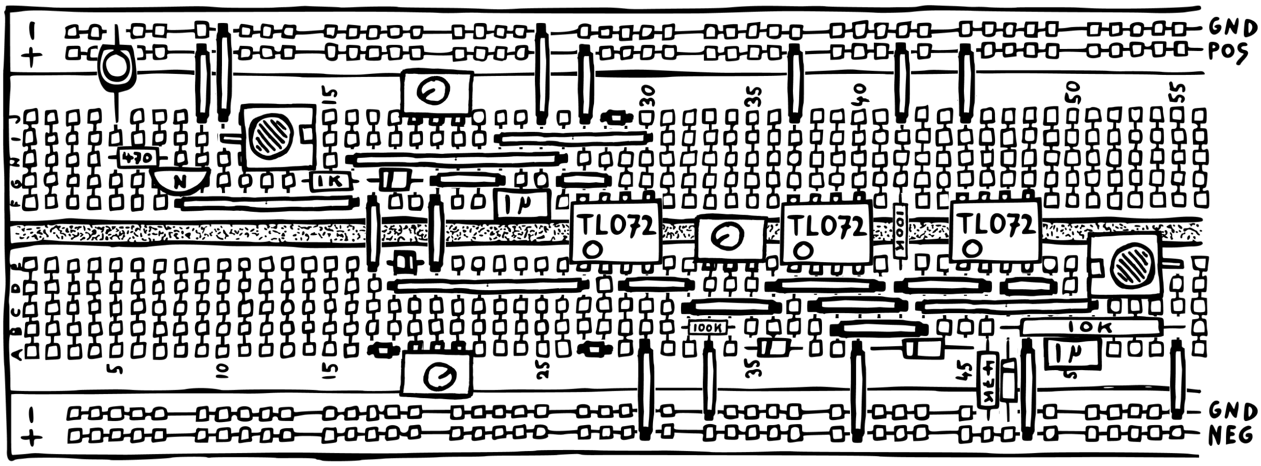
THE STATUS LED

Thankfully, there's a quick and easy way to achieve this: by implementing a simple status LED, which we'll drive using our envelope's output. This way, the LED will tell us what voltage level the envelope is currently pushing out out by shining more or less bright. So far, so simple. But there are two slight problems with this. First, LEDs are (as I said before) pretty power hungry. And though our op amp buffer is capable of providing enough current to light one up, this would pretty much completely occupy it. Second, LEDs are super quick to burn out if we push too much current through them. **So we need to make sure we provide enough, but not too much current for our status LED**. Doing this is much easier than it might sound, though. We only need three components in total: our LED, an NPN transistor and a small resistor.[13]



Here's how this works. Whenever our buffer sends out a non-zero voltage, a tiny current will flow into the transistor's base. This will cause that transistor to open up, resulting in a much bigger current flowing from the positive rail through the current limiting resistor and then our LED, lighting it up.[14]

The nifty thing about this setup is that the higher the voltage coming from our buffer, the more the transistor will open up – and the brighter the LED will get. All while the 470 Ω resistor prevents the current from increasing too much. Note that it's not really the op amp that's providing the power to light up the LED here – instead, we're getting it pretty much straight from the power supply.



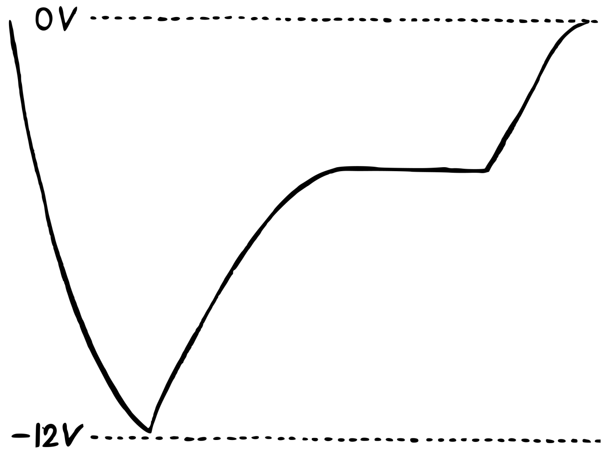
[13] Read more about transistors in the components & concepts appendix (page 35).

[14] You can try this chapter's circuits in a circuit simulator. I've already set them up for you right here: https://tinyurl.com/y7gr9hnj – you can change all values by double clicking on components.

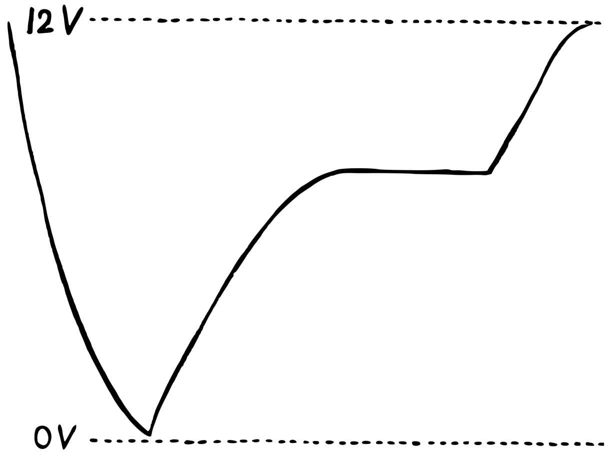
When adding this to your breadboard, make sure you double (or even triple) check all connections **– it's just too easy to accidentally burn an LED.**

THE INVERTED OUTPUT

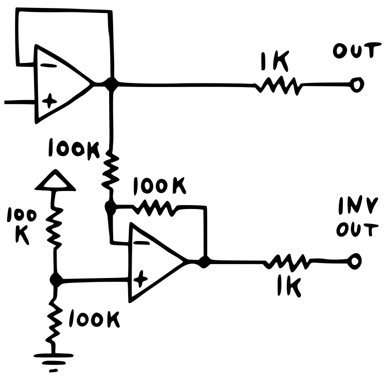
At the beginning of this manual, I promised two extra goodies to make up for our envelope's shortcomings. One of which was an inverted output. But before we can add one, we'll have to clarify what an inverted output is, exactly. If we take the name purely at face value, we might imagine the signal coming from our envelope's inverted output to look like this.

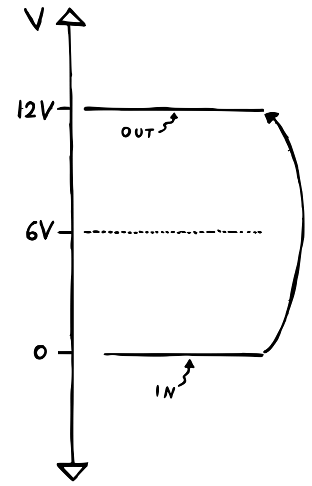


So where our regular output would start off at a 0 V baseline and go up from there, the inverted output would drop below that baseline instead. And though this is totally doable (and is sometimes done this way in other modules), I think it'd be much more useful if the output looked like this instead.



It's the same core principle, but the baseline is now our high level voltage. Meaning that the curve is no longer dipping into the negatives, but rather „peaking“ at ground level. This way, it'll do the exact opposite of what we've seen before when controlling a VCF with it. The filter will be wide open by default, and then close down as the envelope is cycling through its phases. Cool! But how do we actually implement this?

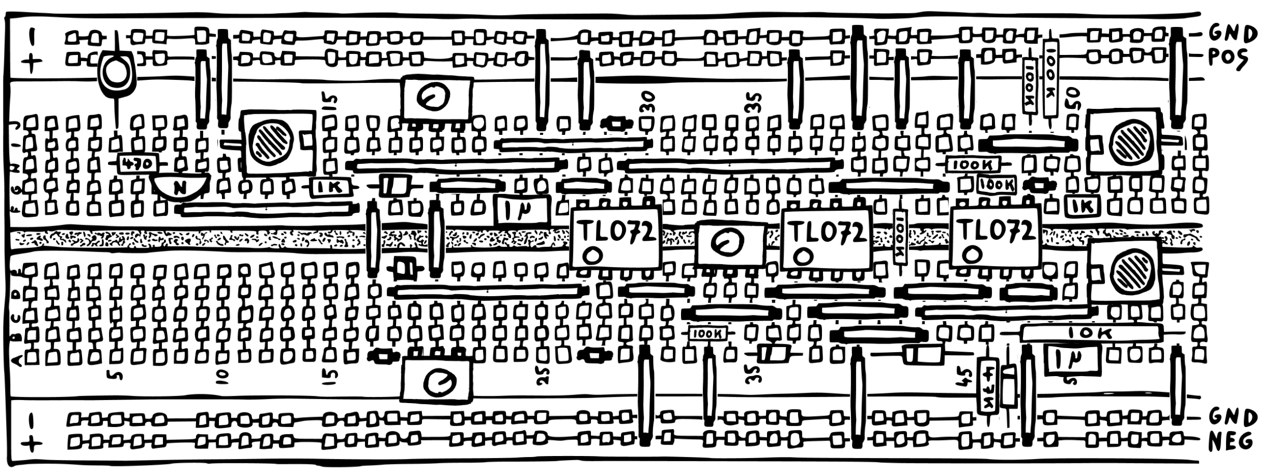




Turns out it's doable quite easily with another op amp, which we'll set up as an inverting buffer.[15] But instead of connecting the non-inverting input to ground – as you would normally do –, we give it a constant 6 V from a simple 50 % voltage divider.

To understand how this works, I find it very helpful to imagine that the voltage we apply to the non-inverting input is the mid-axis against which we are inverting (or mirroring) the signal. If that mid-axis is 0 V, then an input of 12 V is going to give us an output of –12 V. But if the mid-axis is 6 V, then the same input will give us an output of 0 V instead. And conversely, a 0 V input results in a 12 V output.[16]

Since our envelope is operating squarely in the range between 0 and 12 V, the output curve will simply be flipped on its head while staying within that range. Which is exactly what we said we're after. So let's add this to our breadboard and see how we fare.



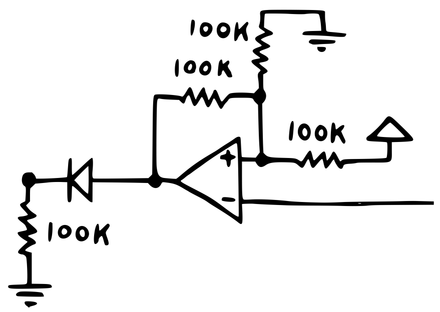
Try driving your VCF with the newly added inverted output. You should be able to verify that it now starts off open and then closes down as the envelope cycles through its phases.

[15] Read more about inverting buffers/amplifiers in the components & concepts appendix (page 34).

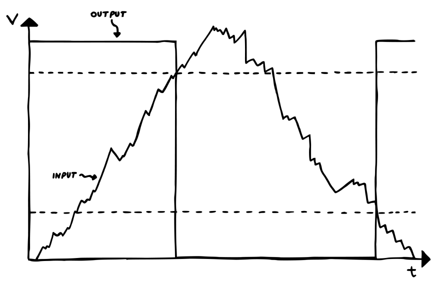
[16] You can try this chapter's circuits in a circuit simulator. I've already set them up for you right here: https://tinyurl.com/y9rrt8pp – you can change all values by double clicking on components.

LOOPING THE ENVELOPE

The second extra goodie I promised was a loop-mode for our envelope. Here's how we'll implement that. You can turn pretty much any regular envelope into a looping envelope by using this little circuit.



You just patch it in between output (on the right) and input (on the left) – and the envelope will re-trigger continuously. To make that happen, we're configuring yet another op amp as a Schmitt trigger inverter.[17] You can think of that like a watchdog which constantly monitors the envelope's output voltage. **Whenever that output drops to 0 V, our watchdog will re-trigger the envelope and then sit back and wait until it's time to strike again.**[18]



This works because a Schmitt trigger inverter has two thresholds against which it compares the voltage we apply to its input. The three resistors up top (or rather the relation between them) set those thresholds. In our case, the bottom threshold is set at 0 V, while the top threshold is placed at around 8 V. So when the input drops to 0 V, the op amp's output will jump to 12 V. Then once the input rises above the 8 V line, the output will drop down to –12 V.

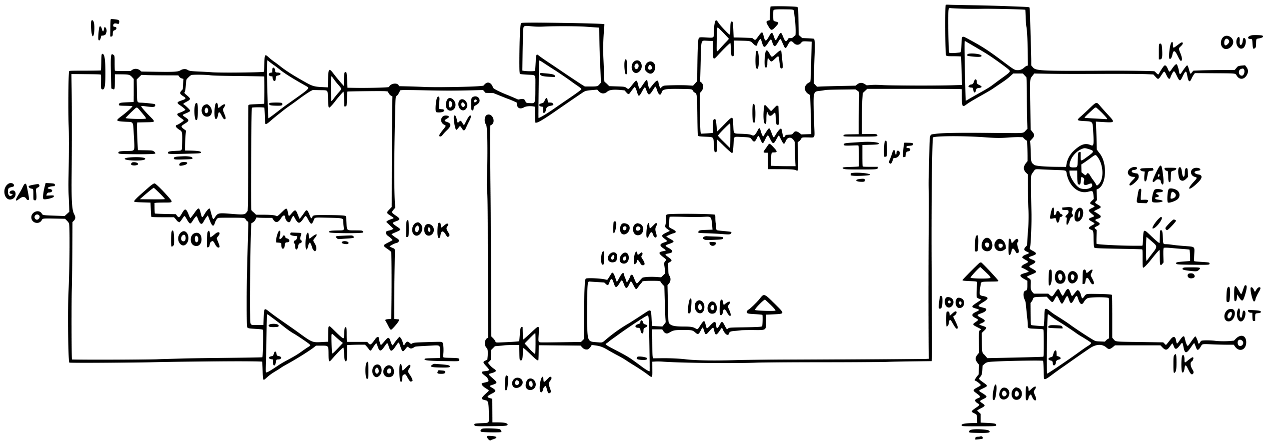
Now, you might've noticed the diode and 100k pulldown-resistor after our op amp's output. Why do we need these? Don't we already block any negative voltages coming from our two input comparators? Yes, we do – but for the loop mode, we'll have to circumvent them. This is because we need to make sure that our Schmitt trigger inverter doesn't get stuck. Because as we know, it will only change states if the envelope's output rises above 8 V. If we use our envelope's regular input, the maximum output voltage depends on the set attack and sustain levels. Dialing in a slow attack and low sustain level might cause the voltage curve to peak at less than 8 V – which will break the loop by not triggering the inverter. The solution to this problem is connecting the inverter's output

---

[17] Read more about Schmitt trigger inverters in the components & concepts appendix (page 28).

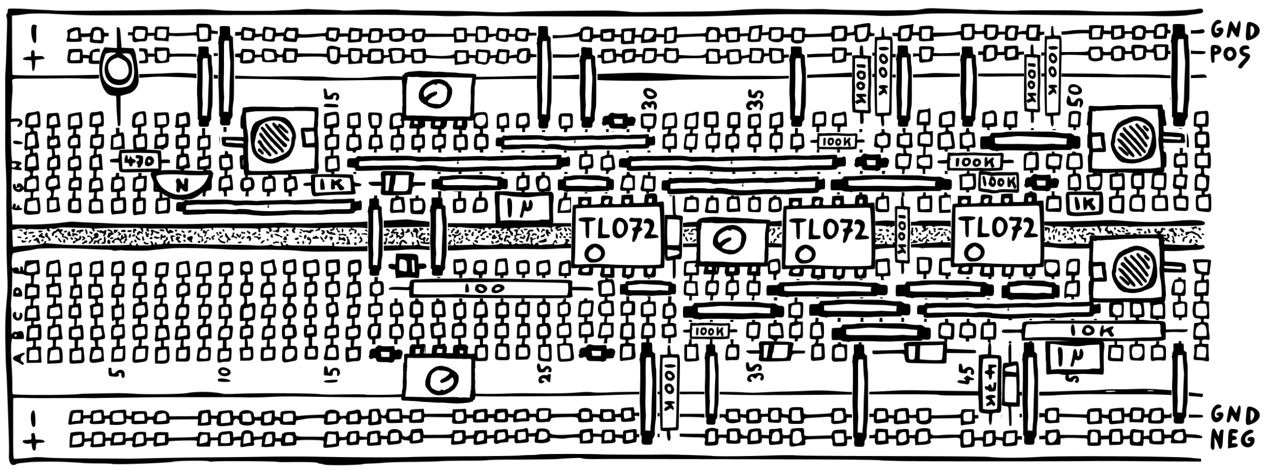
[18] You can try this chapter's circuits in a circuit simulator. I've already set them up for you right here: https://tinyurl.com/y7wh2o6u – you can change all values by double clicking on components.

directly to the input buffer – ideally through a simple switch that allows us to toggle between the regular mode and our new loop-mode.



By circumventing the input comparators, the relation between our Schmitt trigger inverter's state and the envelope's output voltage becomes a lot simpler. Because now, there's no envelope setting that can prevent its output from peaking at a value below 8 V. The inverter will simply keep pushing out a high level voltage until we cross that threshold.

Great! But before we try this on the breadboard, there's another small thing I snuck in here: the 100 Ω resistor after our input buffer. This is really more of an added bit of polish. **It ensures that when our attack- or release-pots are dialed down all the way, (dis-)charging the capacitor isn't instant.** If we leave this out, we might get an ugly clicking-noise from our filter, for example – simply because the change in cutoff frequency is so abrupt.



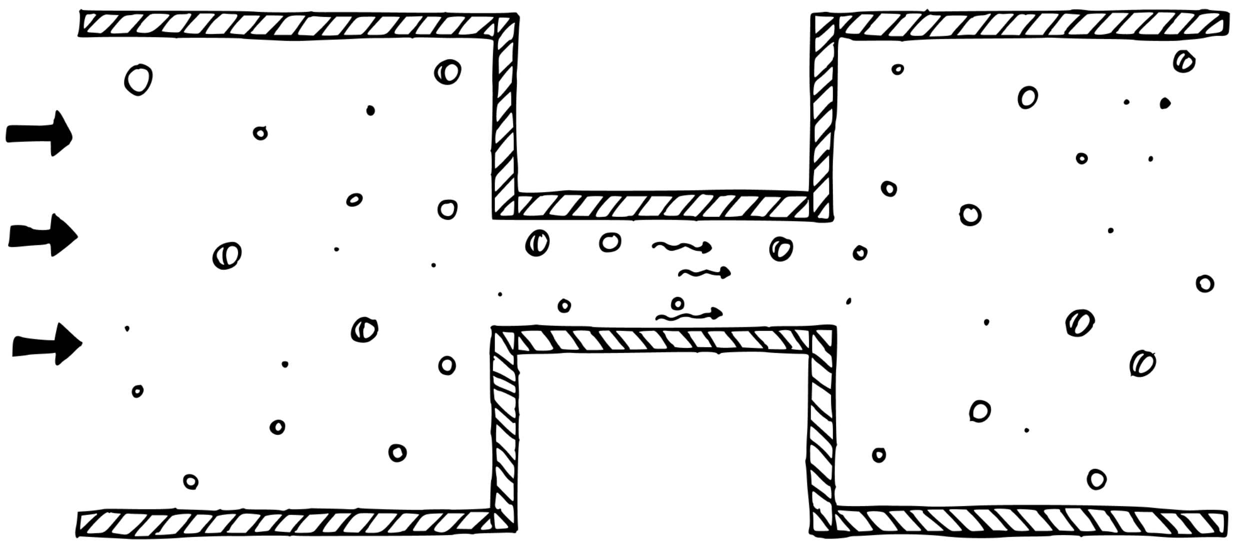
And with that, our envelope is done. If you now want to make your creation permanent, dig out the panel and PCB from the kit, heat up your soldering iron and get to building! You can find more information on how to populate the board & how to solder in the enclosed appendix.

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

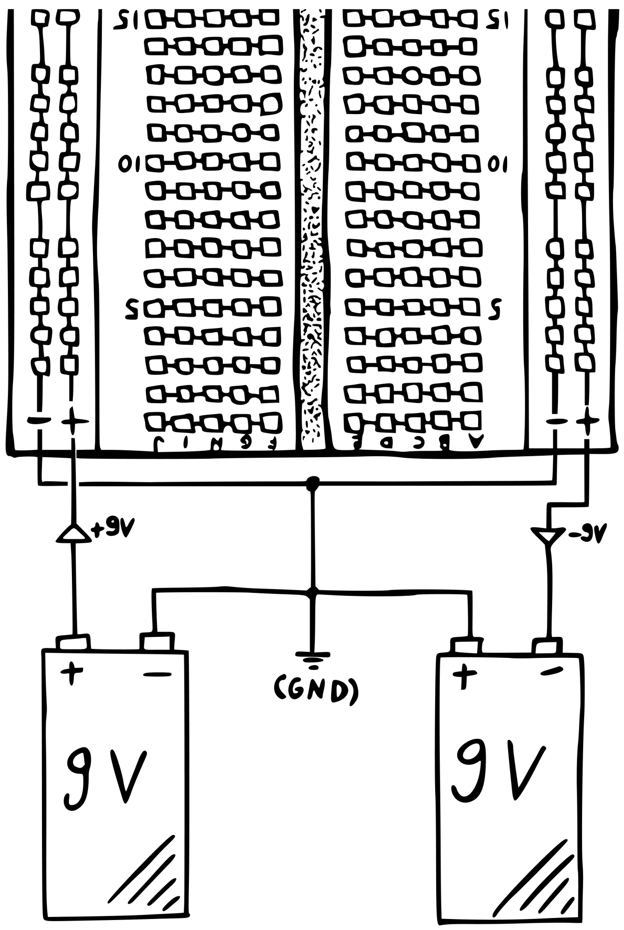
There are three main properties we're interested in when talking about electronic circuits: resistance, voltage and current. To make these less abstract, we can use a common beginner's metaphor and compare the flow of electrons to the flow of water through a pipe.



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.

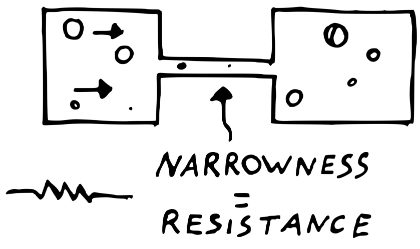


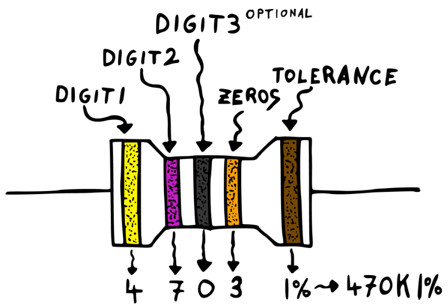
By connecting the batteries like this, the positive terminal of the left battery becomes your +9 V, while the negative terminal of the right is now your –9 V, and the other two combine to become your new ground.[19] **Please make sure you disconnect the batteries from your breadboard when you make changes to the circuit!** Otherwise you run the risk of damaging components.

[19] If you're struggling with setting this up, you can watch me do it here: https://youtu.be/XpMZoR3fgd0?t=742

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[20] – 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.

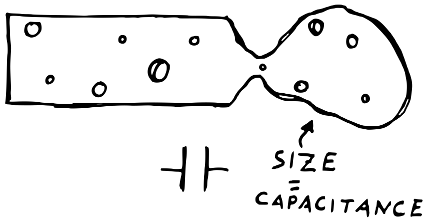
For the resistors in this kit, a yellow body tells you that the actual resistance value might be ±5 % off. A dark blue body indicates ±1 % tolerance. Some kits will also contain light blue ± 0.1% resistors to avoid the need for manual resistor matching.

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.

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

CAPACITORS

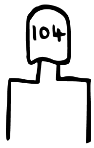
A capacitor is a bit like a balloon that you can attach to the open end of a pipe. If there's some pressure in the pipe, the balloon will fill up with water until the pressure equalizes. (Since the balloon needs some space to expand into, both of the capacitor's legs need to be connected to points in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



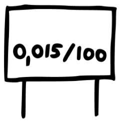
Then, should the pressure in the pipe drop, the balloon releases the water it stored into the pipe. The maximum size of the balloon is determined by the capacitor's capacitance, which we measure in farad (F). There are quite a few different types of capacitors: electrolytic, foil, ceramic, tantalum etc. 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.

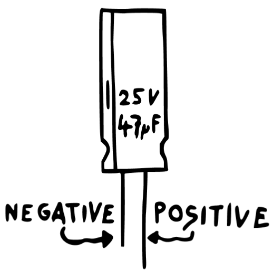
Unlike resistors, capacitors have their capacitance value printed onto their casing, sometimes together with a maximum operating voltage. **Be extra careful here!** That voltage rating is important. Your capacitors can actually explode if you exceed it! So they should be able to withstand the maximum 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.[21] 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.[22]



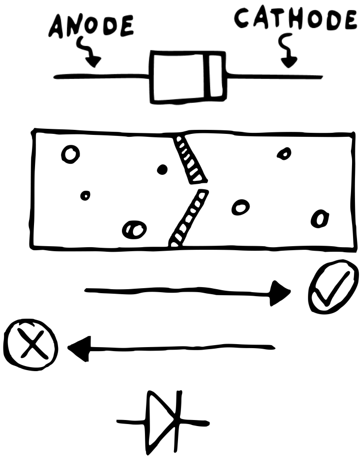
Electrolytic capacitors can be identified by their cylinder shape and silver top, and they usually directly indicate their capacitance value and their voltage rating. They are polarized – so make sure you put them into your circuit in the correct orientation.

---

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

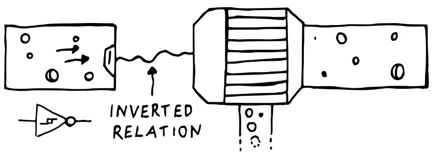
22 If yours do encode their values, same idea applies here – look up film capacitor value code.

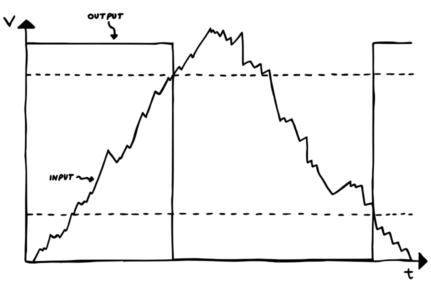
DIODES



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.

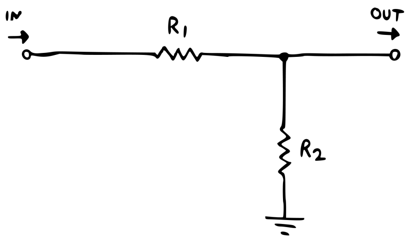
SCHMITT TRIGGER INVERTERS



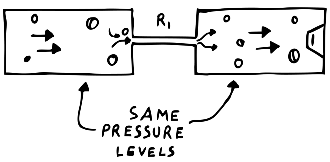


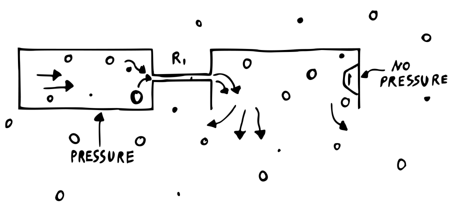
You can think of a Schmitt trigger inverter as two separate things. On the left, there's a sensor that measures the pressure inside an attached pipe. On the right, there is a water pump. This pump's operation is controlled by the sensor. Whenever the pressure probed by this sensor is below a certain 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.

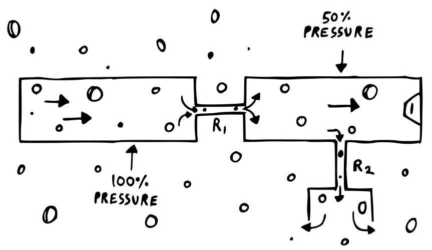
VOLTAGE DIVIDERS



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.



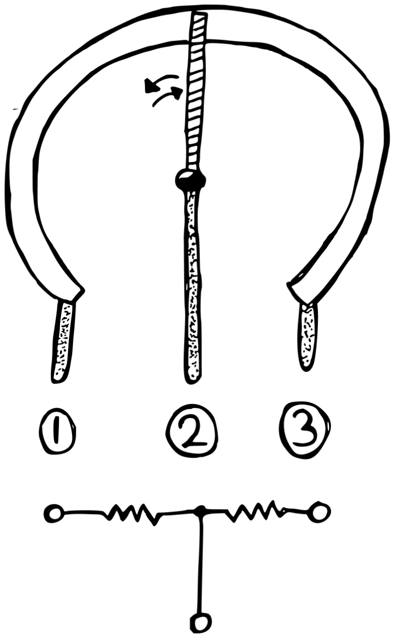




First, think about what would happen if R2 was completely sealed off. Our sensor would tell us that **the pressure on the right side is exactly the same as the pressure on the left**. Because the pushing force has nowhere else to go.
On the other hand, imagine R2 would just be a wide opening. Then **the 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

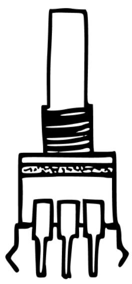
Potentiometers can be used as variable resistors that you control by turning a knob. But, and that's the handy part, they can also be set up as variable voltage dividers. To see how that works, let's imagine we open one up.

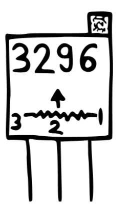


Inside, we would find two things: a round track of resistive material with connectors on both ends plus what's called a wiper. This wiper makes contact with the track and also has a connector. It can be moved to any position on the track. Now, the resistance value between the two track connectors is 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.





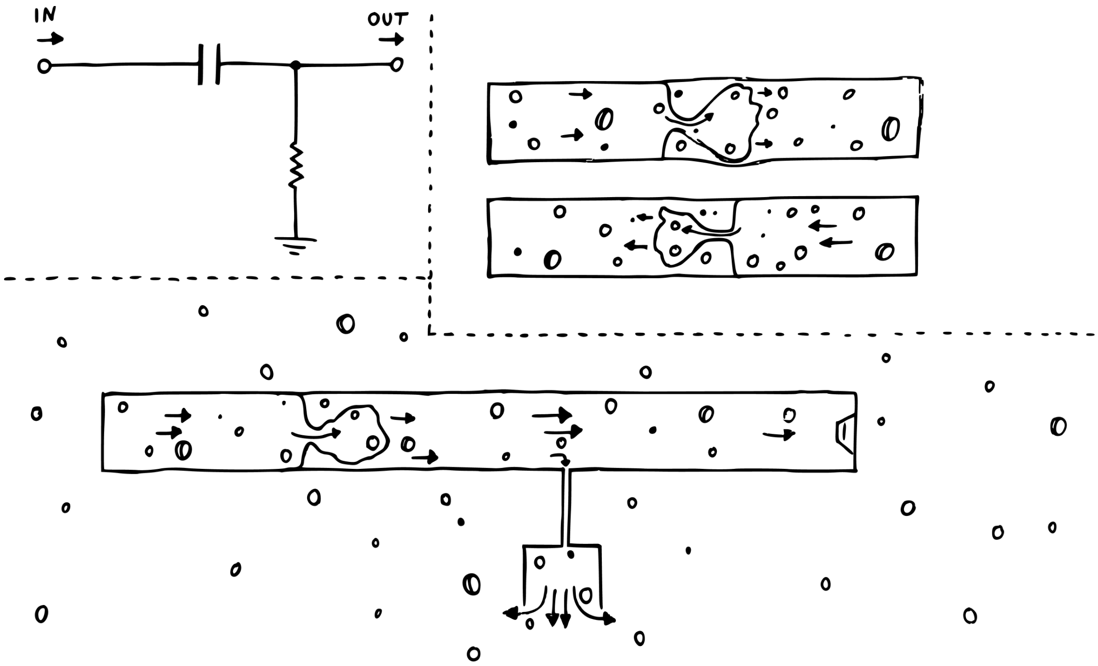
In these kits, you will encounter different types of potentiometers. First, there's the regular, full-size variant with a long shaft on top. These are used to implement user-facing controls on the module's panel and they usually – but not always – indicate their value directly on their casing. Sometimes, they'll use a similar encoding strategy as capacitors, though.[23]

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.

[23] 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**.

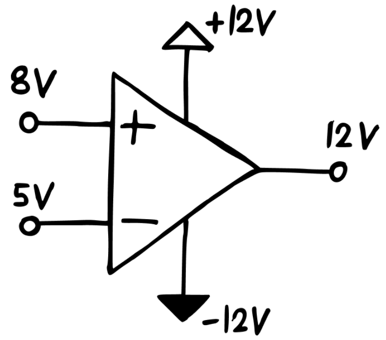


Next, we'll bring in a resistor after the coupling point, going straight to ground. **This acts like a kind of equalizing valve**. Now imagine we apply a steady 5 V from one side. Then on the other side, we'll read 0 V after a short amount of time. Why? Because we're pushing water into the balloon with 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?



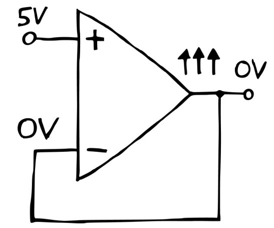
The key here is that op amps are essentially differential amplifiers. This means that they only amplify the difference between their two inputs – not each of them individually. If that sounds confusing, let's check out a quick example. So we'll imagine that one sensor – called the non-inverting input – 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.

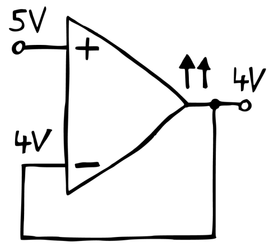
The handy thing though about op amp outputs is that they draw their power directly from the power source. This means that they can supply lots of current while keeping the voltage stable. **That's why we say an op amp has a very low output impedance**.

OP AMP BUFFERS/AMPLIFIERS

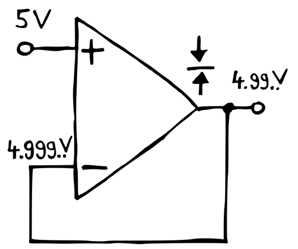
Buffering, in the world of electronics, means that we provide a perfect copy of a voltage without interfering with that voltage in the process. With an op amp-based buffer, the buffering process itself works like this. We use the non-inverting input to probe a voltage, while the inverting input connects 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.



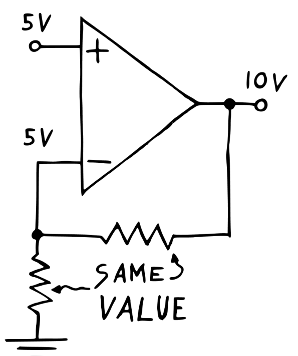
Before the op amp starts processing the voltages at its inputs, the output will be switched off. This means that **output and inverting input sit at 0 V at first**. So then, the op amp will subtract 0 from 5 and multiply the result by its gain. Finally, it will try and increase its output voltage to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



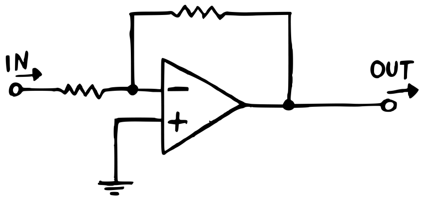
But as it's pushing up that output voltage, the **voltage at the inverting input will be raised simultaneously**. So the difference between the two inputs is shrinking down. Initially, this doesn't matter much because the gain is so large. As the voltage at the inverting input gets closer to 5 V though,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

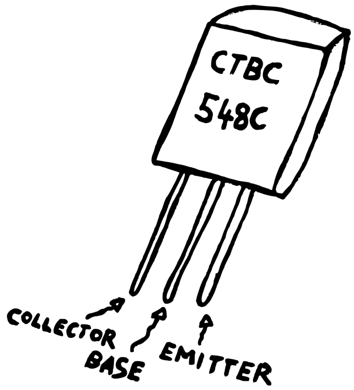
How does this work? Well, let's assume that we're applying a steady voltage of 5 V on the left. Then, as we already know, the op amp will subtract the inverting input's voltage from the non-inverting input's voltage, leaving us with a result of –5 V. Multiply that by the huge internal gain, and the op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

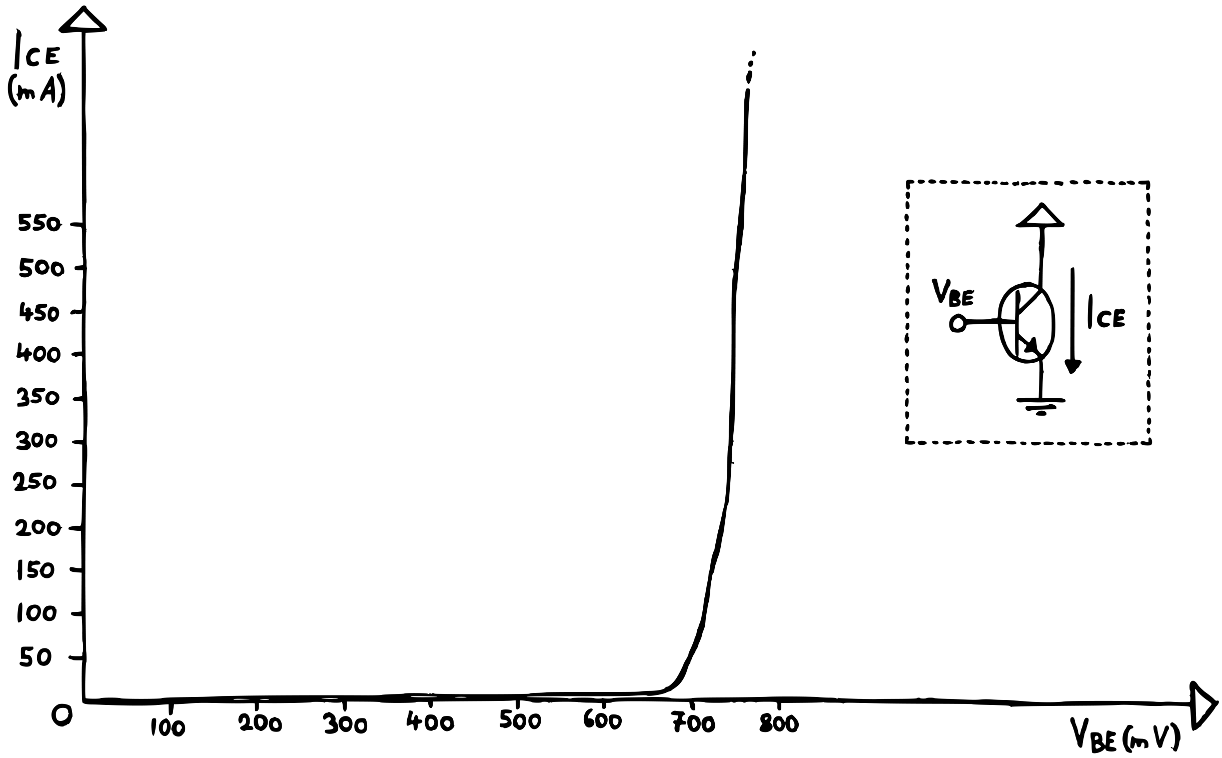
And that means that the voltage at the inverting input will be lowered to about 0 V, allowing our op-amp to settle on the current output voltage level. So while we read 5 V on the left, we'll now read a stable –5 V at the op amp's output. Congrats – we've built an inverting buffer! **If we want to turn 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.**[24] 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[25] 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.



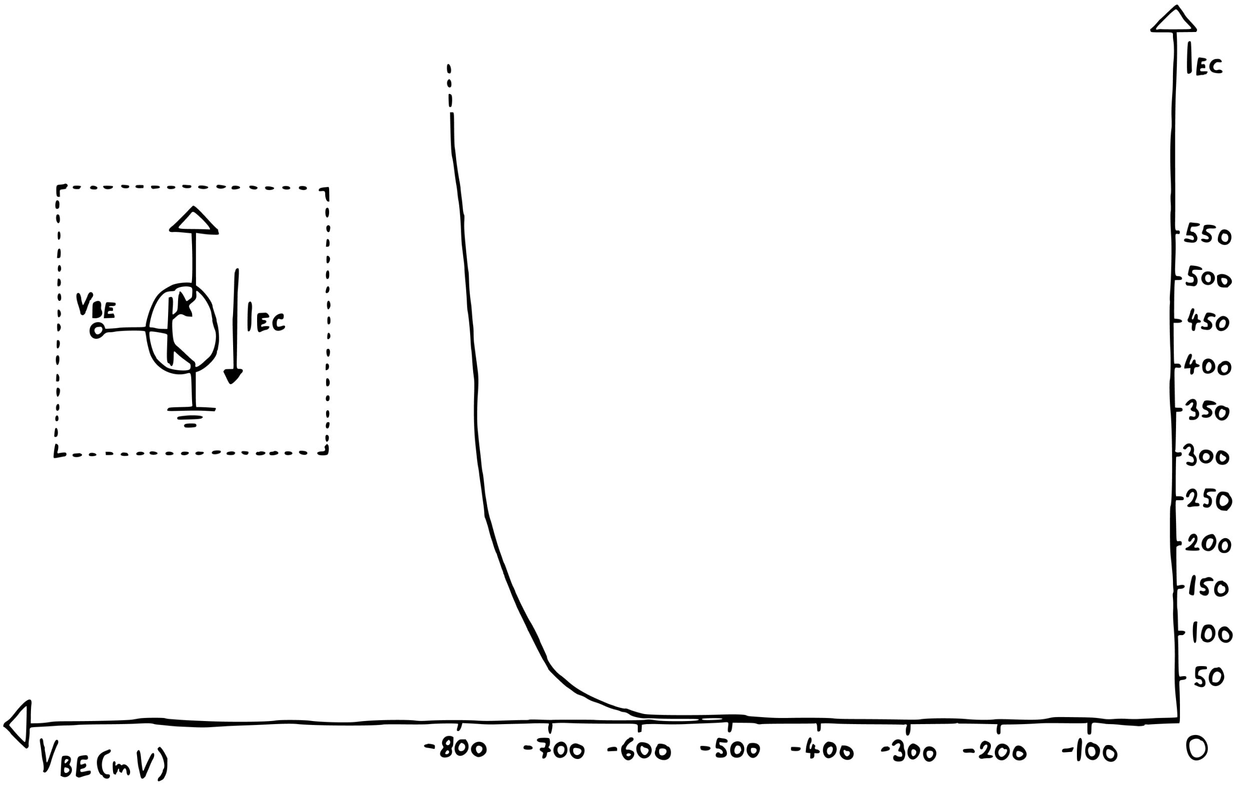
First, the relation between base voltage and collector current is exponential. Second, unlike a resistor, a BJT is not symmetrical – so we can't really reverse the direction of the

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

25 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**[26]. 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!

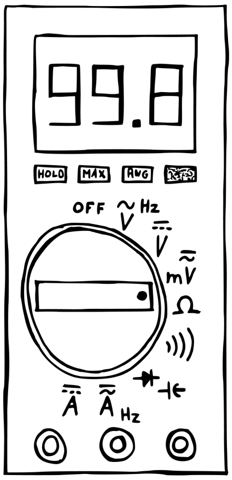


[26] Again, the voltage is measured between base and emitter.

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.

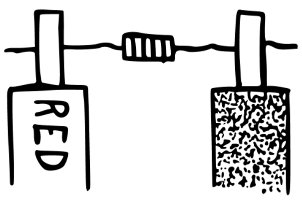
MULTIMETERS

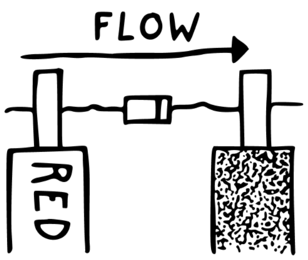


Multimeters come in different shapes and sizes, but the most common type is probably the hand-held, battery powered variant. It can measure a bunch of different things: voltage, current, resistance, continuity. Some have additional capabilities, allowing you to check capacitance, oscillation frequency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



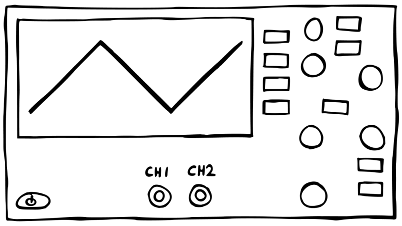


In some cases, it doesn't matter which probe you connect to which component leg or point in your circuit. This is true for testing resistors, non-polarized capacitors (foil/film, ceramic, teflon, glass etc.), continuity[27] or AC voltage.

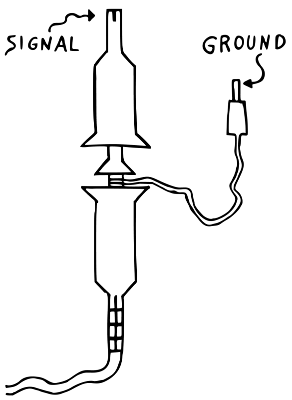
In others, you'll have to be careful about which probe you connect where. For testing the forward voltage drop of a diode, for example, **the multimeter tries to push a current from the red to the black probe**. Here, you'll have to make sure the diode is oriented correctly, so that it doesn't block 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.

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

OSCILLOSCOPES



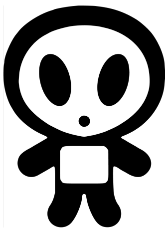
While multimeters are fairly cheap and compact, oscilloscopes are usually somewhat pricey and bulky. **If you're willing to make the investment, they are a huge help with the troubleshooting process, though**. Using one is, again, surprisingly straightforward – if you manage to work your way through 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 Envelope** 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 **Gate input** jack socket, **XS2** is the envelope **signal output** jack socket and **XS3** is the **inverted envelope signal output** jack socket – these are the very same we've already been using on the breadboard for interfacing with other devices. In our designs, we use eurorack standard 3,5mm jack sockets (part number WQP-PJ301M-12).

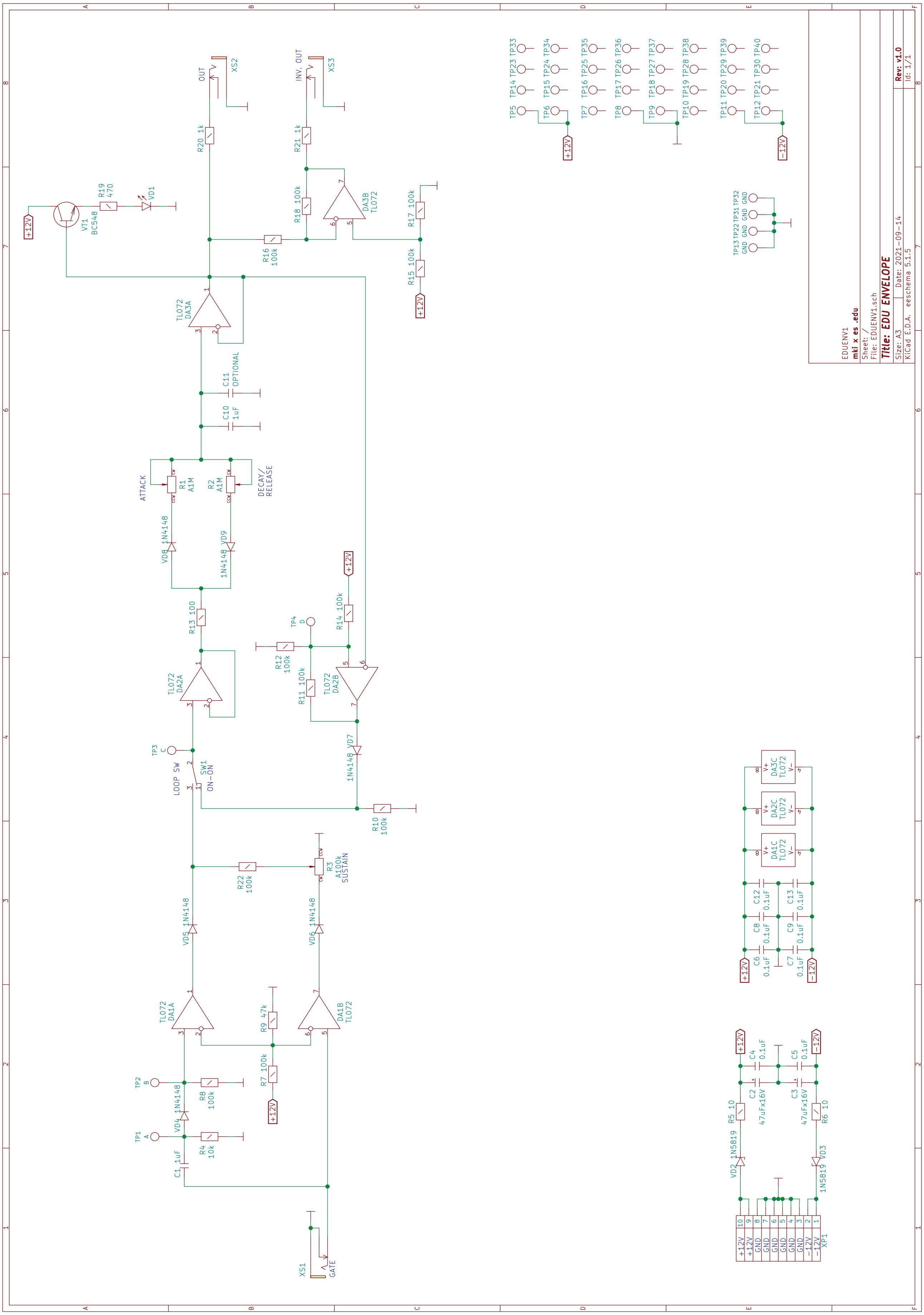
XP1 is a standard eurorack **power connector**. It's a 2x5 male pin header with a key (the black plastic shroud around the pins) to prevent accidental reverse polarity power supply connection. This is necessary because connecting the power incorrectly will permanently damage the module.

VD2 and **VD3** are **schottky diodes** that double-secure the reverse polarity power supply protection. Diodes pass current only in one direction. Because the anode of VD2 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 VD2, it closes, and no current passes through. The same goes for VD3, 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.

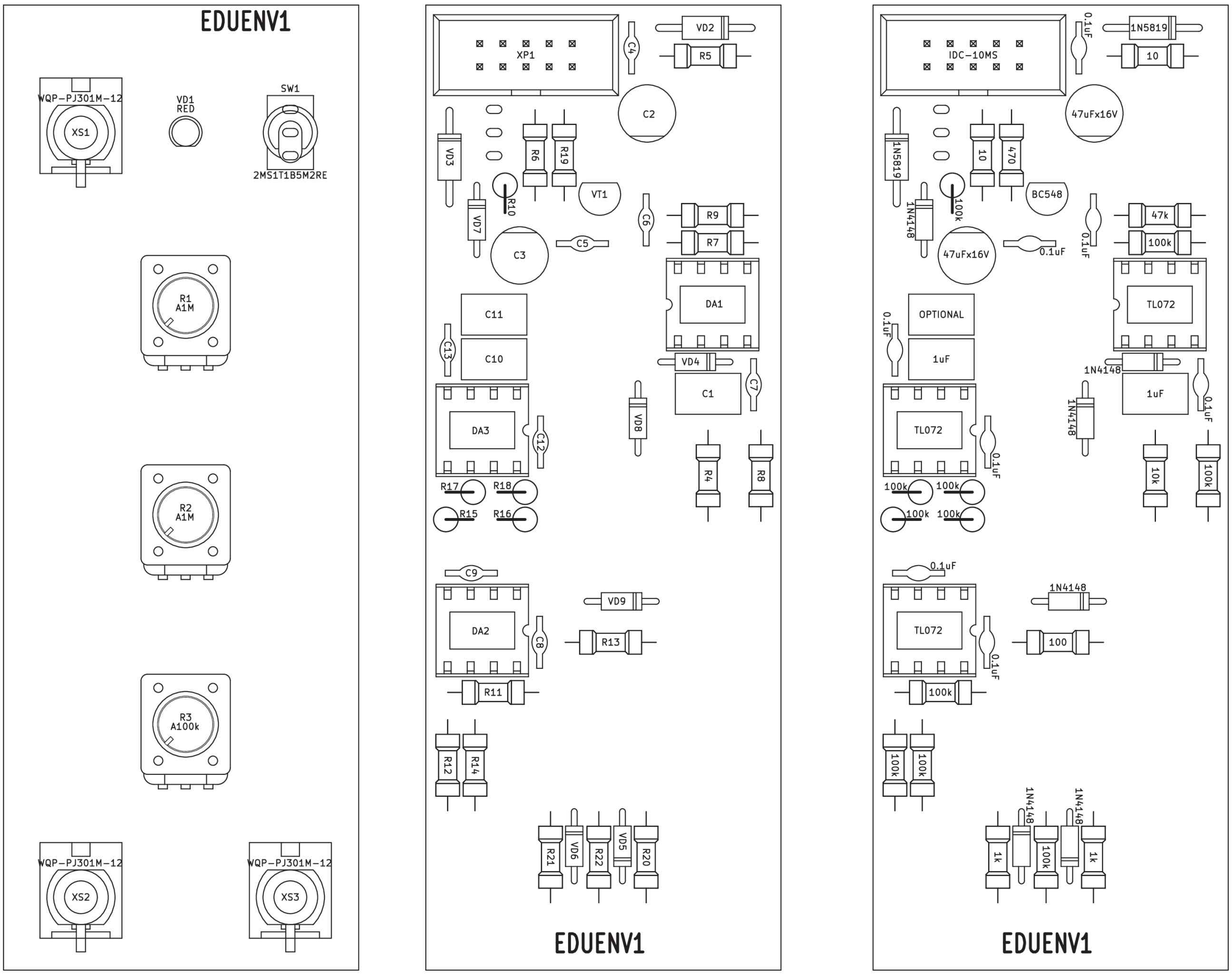
Next, we have two **10 Ohm resistors** (**R5** and **R6**) on the + and – 12 V rails, with **decoupling** (or **bypass**-) capacitors **C2 – C5**. These capacitors serve as energy reservoirs that keep the module's internal supply voltages stable in case there are any fluctuations in the power supply of the entire modular system. In combination with R5 and R6, the large 47 microfarad pair (C2 and C3) compensates for low frequency fluctuations, while C4 and C5 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.

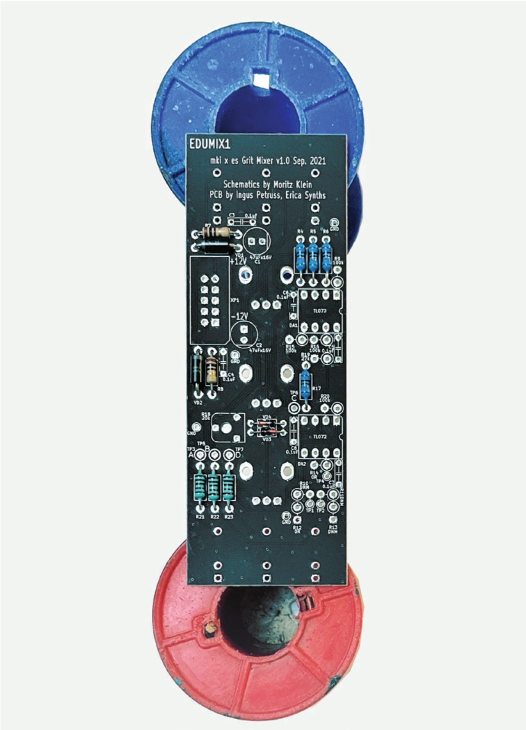
Another advantage of 10 Ohm resistors is that they will act like **slow "fuses"** in case there's an accidental short circuit somewhere on the PCB, or an integrated circuit (IC) is inserted backwards into a DIP socket. The resistor will get hot, begin smoking and finally break the connection. Even though 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 **C6 – C13** 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.

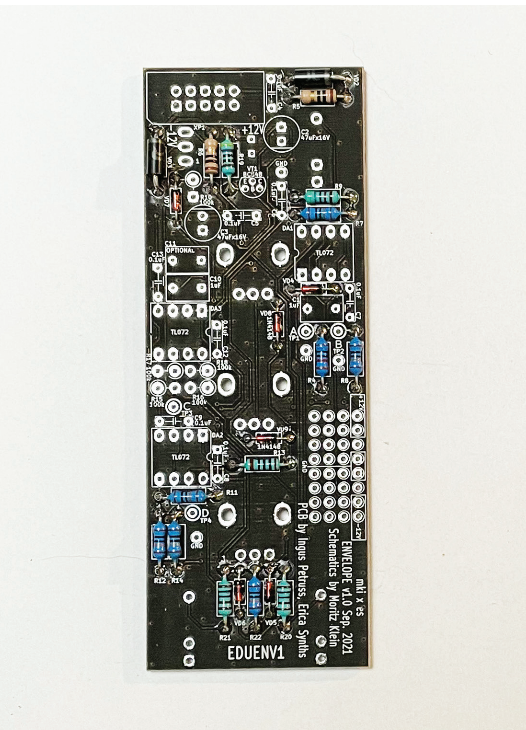

EDUENV1
mki x es .edu
Sheet: /
File: EDUENV1.sch
Title: EDU ENVELOPE
Size: A3
Date: 2021-09-14
Rev: v1.0
KiCad E.D.A. eeschema 5.1.5
Id: 1/1

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.

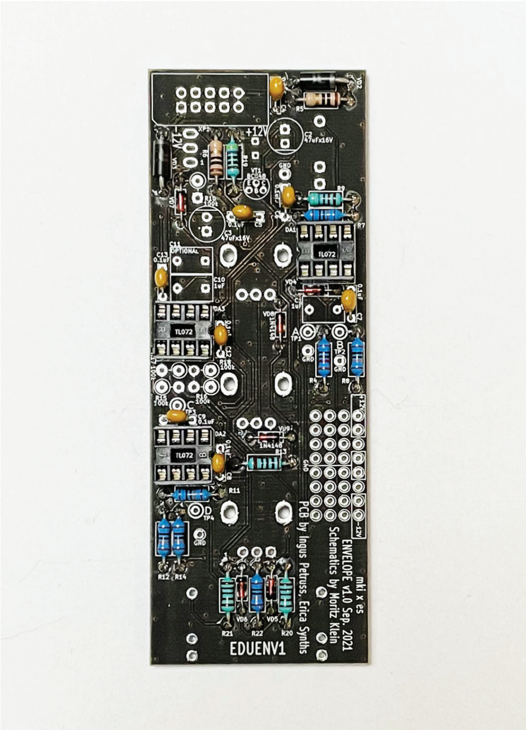




Place the Envelope PCB in a PCB holder for soldering or simply on top of some spacers (I use two empty solder wire coils here).

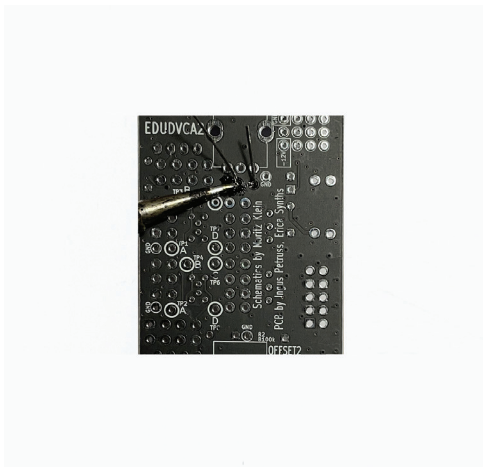


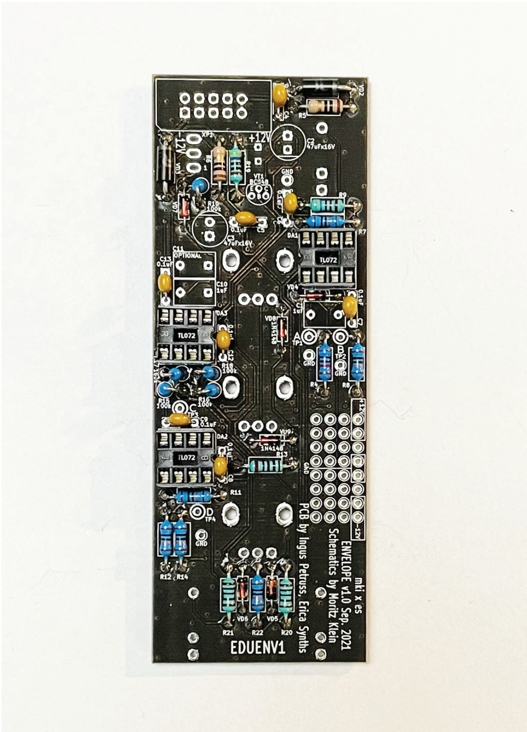
I usually start populating PCBs with lower, horizontally placed components. In this case, these are **most of the resistors**, **switching diodes** and the **power protection diodes**. Bend the resistor leads and insert them in the relevant places according to the part placement diagram above. All components on the PCB have both their value and denomination printed onto the silkscreen. If you are not sure about a resistor's value, use a multimeter to double-check. Next, insert the diodes. 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:

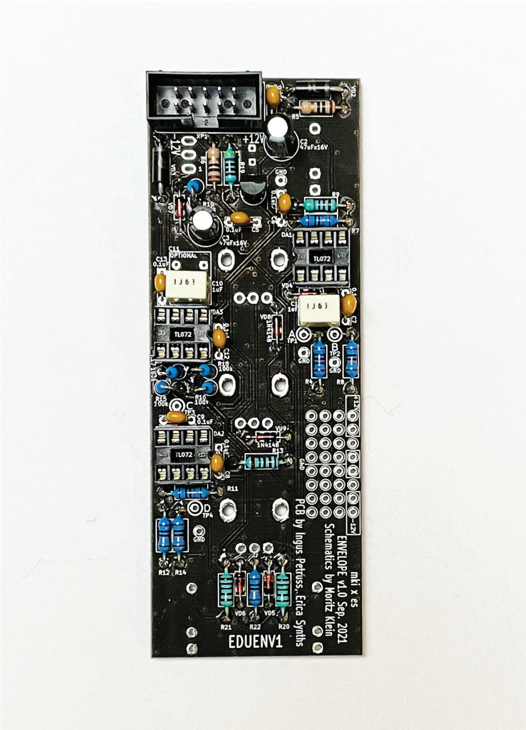
In order to save space on the PCB, some of our projects, including the dual VCA, have **vertically placed resistors**. The next step is to place & solder those. Bend a resistor's legs so that its body is aligned with both legs and insert it in its designated spot. Then solder the longer lead from the 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.





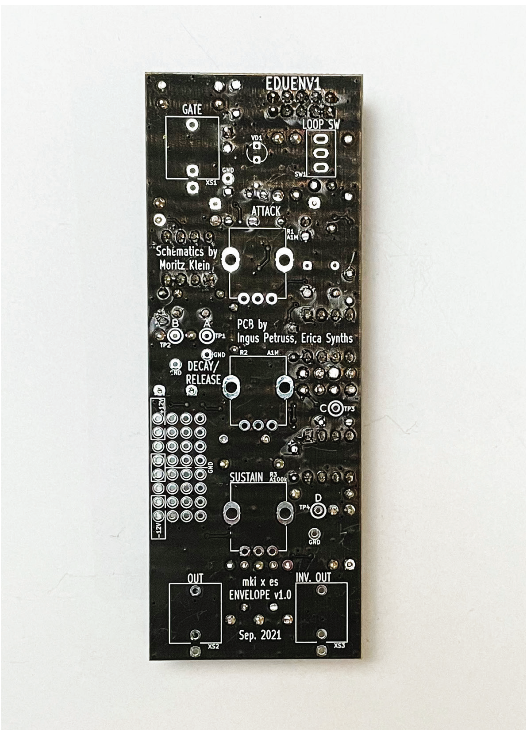
Once you are done with soldering all resistors, your PCB should look like this:

Next, insert & solder 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.

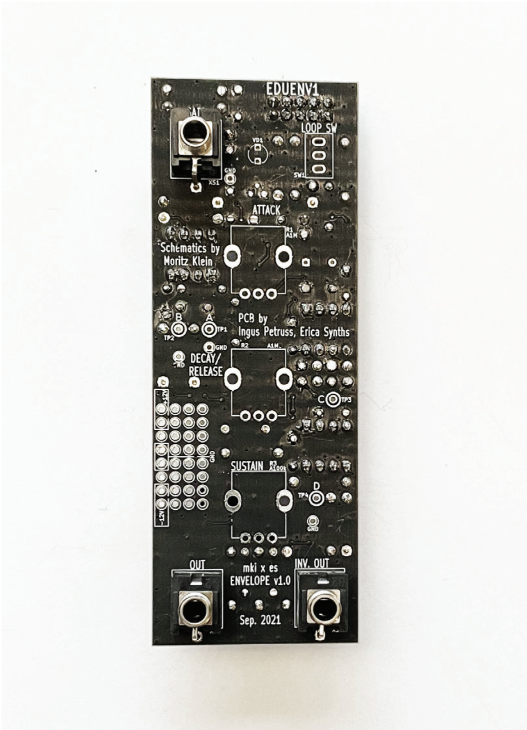


Next up: inserting & soldering the transistor. Make sure you align the transistor with the marked outline on the silkscreen – **orientation is critically important here**. Also, insert film capacitors and solder them.

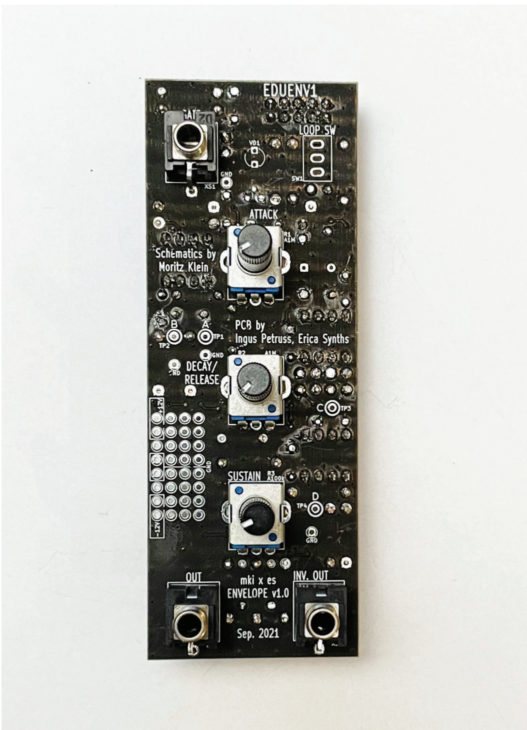
Then complete the component side of the Envelope PCB by soldering the **PSU socket.** Make sure the orientation of the socket is as shown in the picture below – the arrow pointing to the first pin is aligned with a notch on the silkscreen. The key on the socket will be **facing inwards towards the PCB**.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



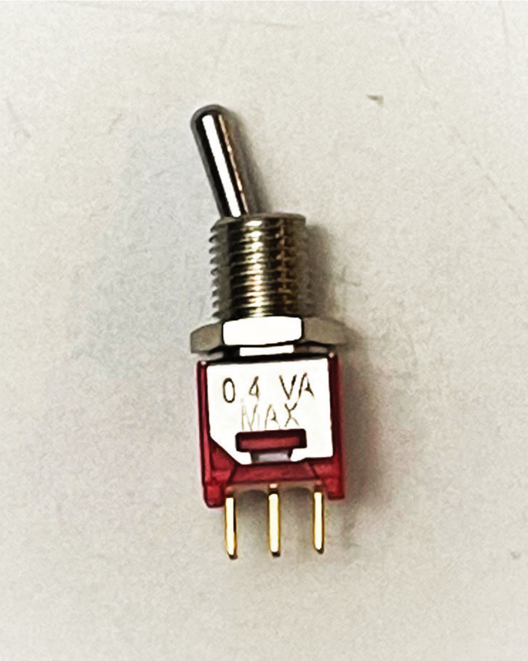
Now, turn the PCB around and inspect your solder joints. **Make sure all components are soldered properly and there are no cold solder joints or accidental shorts.** Clean the PCB to remove extra flux, if necessary.



Insert the jack sockets and solder them.



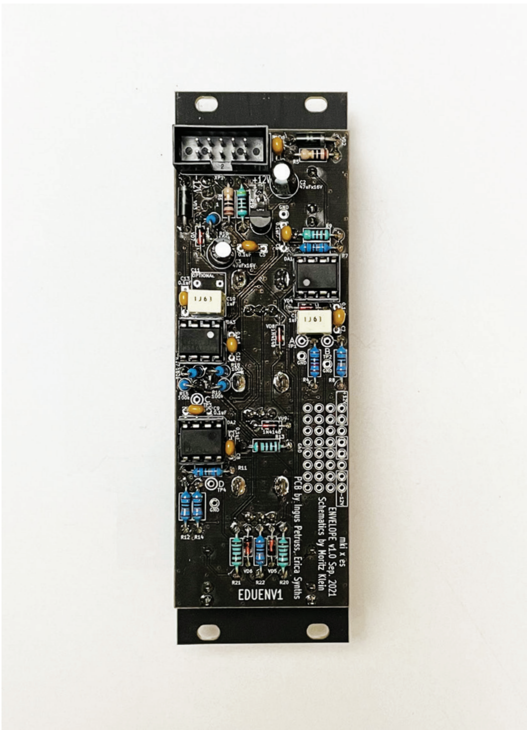
Insert the potentiometers, but don't solder them yet! Fit the front panel and make sure that the potentiometer shafts are aligned with the holes in the panel – and that they're able to rotate freely. Now, go ahead and solder the potentiometers.



The **single/loop switch** requires 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**.

Insert the LED in the relevant place on the PCB, but do not solder it, yet! Orientation of the LED is important – check the silkscreen! A notch on the silkscreen 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** – has to go into a hole with square-like polygon on the PCB. **Fit the front panel** again and **fix it** with 3 nuts on jack sockets and one nut on the switch. Now, solder the LED. We are almost done!



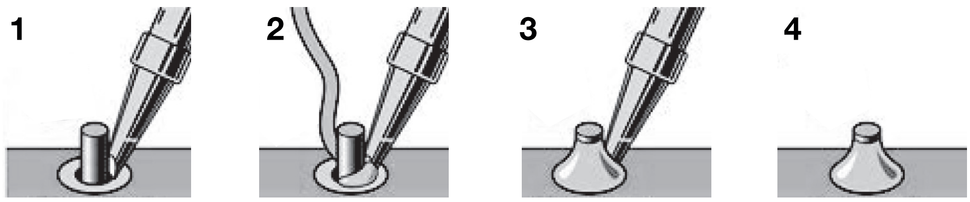
Now, **insert the ICs into their respective DIP sockets.** Mind the orientation of the ICs – match the notch on each IC with the one on its socket.

Congratulations! **You have completed the assembly of the mki x es.edu Envelope module!** It does not need any calibration and, if assembly is correct, it should work straight away. 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. Flip the **switch to the LOOP mode** and check if the LED is blinking. If it is, adjust Attack and Release settings to see if turning the potentiometers clockwise **decreases the LED blinking** frequency. If you consider making slowly evolving drone music, you can install an optional C11 (not included in the kit), which will double the Attack and Release time. **Enjoy!**

SOLDERING APPENDIX

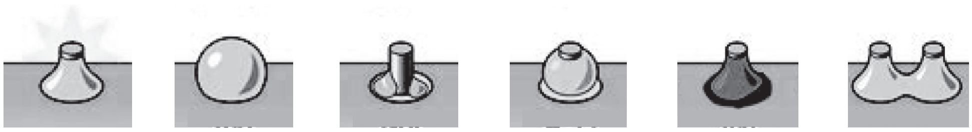
If you’ve never soldered before – or if your skills have become rusty – it’s probably wise to check out some **THT** (through-hole technology) **soldering tutorials on YouTube**. The main thing you have to remember while soldering is that melted solder will flow towards higher temperature areas. So you 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 of parts, while for larger elements like potentiometers and sockets, you may want to increase that temperature to **370 °C.**

Here’s the recommended soldering sequence:



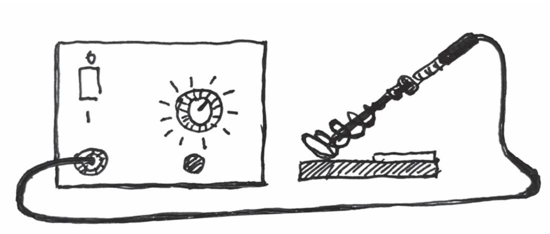
Heat part and pad 2 - 3 sec | Add solder | Continue heating 1 -2 sec. | Let cool

After you have completed soldering, inspect the solder joint:

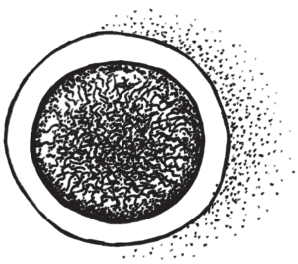


Perfect | Too much solder | Not enough solder | Cold joint | Too much heat | Short

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:

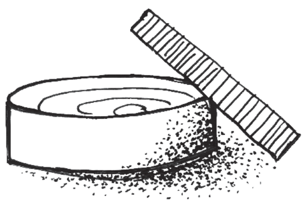


A decent soldering station. Top-of-the-line soldering stations (brands like Weller) will cost 200€ and above, but cheaper alternatives around 50€ are often good enough. Make sure your soldering station of choice comes with multiple differently-sized soldering iron tips. The most useful ones for DIY electronics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



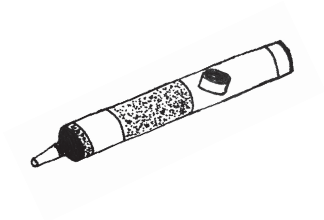
When heated up, the tips of soldering irons tend to oxidize. As a result, solder won't stick to them, so you'll need to clean your tip frequently. Most soldering stations come with a **damp sponge for cleaning the iron tips** – but there are also professional solder tip cleaners with **golden curls** (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.

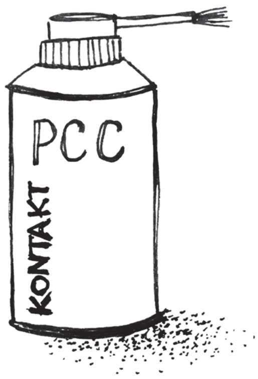


Some **soldering flux** paste or pen will be useful as well.

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.

"Twenty years from now you will be more disappointed by the things that you didn't do than by the ones you did do. Explore. Dream. Discover."

– Mark Twain