sympleSEQ

A simple to build analogue step-sequencer, brought to you by

HEXINVERTER.NET
electronics projects for musical synthesis

(Assembly and Operation Manual rev.1)

Introduction

Hello, and thank you for your interest in or purchase of sympleSEQ!

This is the build manual for the project. You will find the schematics, as well as an assembly guide, printable drill template and tips on how to get the most out of sympleSEQ.

This project was born out of a serious lack of simple analogue hardware sequencers available to build. There are oodles of sequencer designs out there, ranging from simple to complex, but the actual construction of said devices is usually quite laborious, regardless of the electrical complexity/functionality. By design, a sequencer is a fairly simple device, so this tempts many people to not use printed circuit boards when assembling one and just build it all on perf-board. The actual logic circuitry for a sequencer is usually quite easy to understand - especially in the case of a design built around the CD4017 decade counter (known by many as the “Baby10”) like this one. Where builders encounter hardships however is in the extreme redundancy of wiring the controls for these devices. Each step in the sequence requires a potentiometer (knob), toggle switch and LED at minimum. If you have ever soldered wires to panel components, you probably know how delicate of a process it is -- it would be nice to have four hands when doing jobs like this, making just a handful of controls take a decent amount of time and endurance to complete -- let alone 8 or more sets of controls!

The general concensus is that in analogue synthesis, you can never have too many modulation sources. Despite its only providing 8 steps of control voltage, sympleSEQ can still satisfy many builders with 16 or 32 step sequencers already dedicated to pitch modulation by providing extra modulation sources for filters or anything else you can control via control voltage.

Most of all however, sympleSEQ’s target audience is people new to do-it-yourself (DIY) synthesis. When most people think “analogue synthesizer”, something like Pink Floyd’s famous tune “On the Run” probably comes to mind. So, it is only natural that nearly every beginner lusts for the sound of an analogue sequencer. The task of building one for a beginner up until now has been one of great difficulty for most, due to the lack of documentation and the extreme complexity of wiring such a thing to the panel for use afterwards.

Enter sympleSEQ: using a unique dual-board design, primarily board mount potentiometers, LEDs and switches, you can build yourself a simple analogue step-sequencer with little electronics knowledge.

Note that sympleSEQ does not do every single thing that a more complex, large and expensive sequencer does. There may be some features you miss (voltage quantizing, forward/back and other things are not included), but I think all of the essentials are here.

sympleSEQ was made possible only by the extensive online community of creative minds at the electro-music and Muffwiggler forums. Were it not for people contributing their excellent suggestions on feature enhancements for the prototype design and me trying to satisfy peoples’ requests, sympleSEQ would be nowhere near the product it is now!

Also, the cost of running such a product was made possible through pre-funding accumulated by trustworthy individuals who supported the project financially from the beginning!

Of particular mention are those who contributed start-up funding to the project. Seeing only videos and forum correspondence from me about the project and placing a great amount of trust into me is something sympleSEQ would not have survived if not for, so I would like to put out a “thank you” to everyone who placed their financial trust in this project! There is no way I could have funded the start up cost for this project myself, so it just goes to show what many people’s small effort can do when summed up together. Without the amazing do-it-yourself community we have, and the power of the internet, none of this would have been possible! It is a truly awesome spectacle to see so many unique projects realized with the power of communication technology. Thank you all for being a part of it!

If you have not heard, Ben of Re:Synthesis is going to be selling panels to complement this project directly from him. He is also offering fully assembled modules! Get on board that project by visiting the forums if you’re a regular there, or contacting him directly: http://www.resynthesis.co.uk

Anyway, enough of the babbling. Let us get onto the actual manual! If you are looking for quick facts, check out the FAQ (frequently asked questions) at the back of this manual.

- Stacy Gaudreau, aka *Hex Inverter*
hexinverter.net Electronics

Table of Contents

(note that all items are interactive - click to be taken to the page)

Technical Information

Power Supply Requirements:

5-15V DC supply @ ~15mA. sympleSEQ is very forgiving in terms of supply voltage as it uses CMOS circuitry. Of course the LED brightness will change with the voltage used, so you may need to adjust resistor values for odd voltages (say 8v or less perhaps?). The module has been built and tested at 9V and 12V, but should work at pretty much any other in the stated range. It is a great candidate for being put into a portable unit as it can run from a 9V battery with ease and does not require a dual voltage supply like much analogue gear does.

Control Voltage Output:

The output levels will be relative to the supply voltage you use, and should range from 0 to your supply voltage minus a diode voltage drop of about 1V -- ie: at 9V supply, you should see a maximum CV out of about 8V

Gate Output:

The gate will output a positive voltage excursion approximately equal to your supply voltage minus 1V for each step that has its switch at the "ON" (up) position. To use sympleSEQ with gear that wants specific gate levels (ie: 5V gate only) you must drop the voltage level by adding some circuitry to the gate output to your desired voltage level.

Onboard Clock:

sympleSEQ uses CMOS logic square waves operating at or near the supply voltage for a clock source. This clock should be useful for driving most other sequencer designs as well, but you may choose to run sympleSEQ from another more feature rich sequencer's clock. If this is the case, you will need a square wave that is close to the supply voltage you are running sympleSEQ at.

Control Pot. Knobs:

Due to sympleSEQ's extremely compact design, space is very tight on the panel. The maximum knob diameter you can use without mechanical issues is 19mm, but I highly recommend something more like 13 or 14mm. This entire module is very Eurorack in design, in that it packs maximum functionality into minimal space. One interesting feature of the Alpha 9mm PCB mount potentiometers used in this module is their pseudo-knob like shaft. The pots' shafts have a nice grippy surface as well as a pointer on them, thus, they function very much like knobs already! You may choose not to use knobs at all based on this if you wish to save cost/ panel space.

Assembly Guide

When putting your sympleSEQ/s together, I recommend following a set of steps I have synthesized in order to do things in a logical manner. Your mileage may vary, so feel free to experiment and come up with a better system. If you're a seasoned electronics person, you will probably find most of it does not apply to you, so feel free to just skim through it. Don't expect me to answer questions that have been answered already in this manual without giving you the stink-eye though!

Soldering Tips

If you have never soldered before, I highly recommend doing some research first before embarking on this project. I list a few videos below that should be all you need to get up and running making excellent soldering joints. A decent temperature controlled iron is expected for you to be able to complete this project. It is possible that you may damage some of the components while soldering them in if you are using a poor quality iron and have to hold the heat to the joint for too long.

Something I can add that I think is very beneficial: do NOT use sponges as tip cleaners!!! I NEVER use a sponge. Get yourself one of those wire/steel wool looking tip cleaners. I found a pair for $10 on eBay. There is simply no comparison. A sponge seriously damages your iron. Why? Well, first of all, when you plunge your nice hot tip into a sponge covered in cool water, it rapidly cools your tip. Metal does NOT like rapid heat changes, and it also causes your tip to cool off somewhat which makes it work to get back up to operating temperature (if you have a good iron, you probably won't notice it, but with a crappy iron it will probably mean you have to wait after cleaning your tip to get back to soldering!) The number one advantage of using a wire tip cleaner is the fact that you will pretty much never have to replace your tip due to it not damaging it! Seriously. I have been using the same tip for well over a year, and I am not lying when I say it is as good as new! I cannot tell it apart from a brand new tip. When I used to use sponges, I'd have to replace my tip every few weeks due to it getting caked up from the sponge burning onto it/oxidation/damage. Do yourself a favour and throw out that sponge. They are really inferior! If you have to use one for now, don't worry about it, but consider upgrading in the future!

Anyway, here are videos I highly recommend watching if you have never soldered before. If you watch these videos and follow what they are saying, I promise you will become a master of soldering in no time!

http://www.eevblog.com/2011/06/19/eevblog-180-soldering-tutorial-part-1-tools/

http://www.eevblog.com/2011/07/02/eevblog-183-soldering-tutorial-part-2/

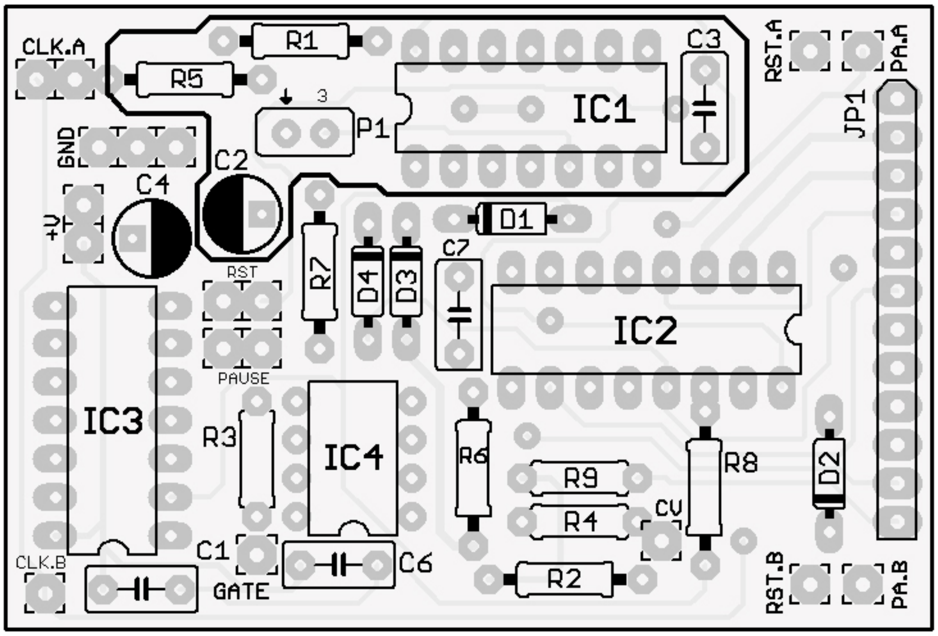
Once you've watched those, this video has some specific information about how to install certain components, tips/tricks, so I highly recommend watching it as well. I have watched all of these and therefore am not just blindly recommending them. They really are excellent videos:

http://tangentsoft.net/elec/movies/tt02.html

Logic Board Assembly

NOTE: It is necessary to install only one set of clock circuitry (outlined on the board silk-screen with a solid black border) if you are running multiple sympleSEQs off of the same clock.

Main Component Overlay



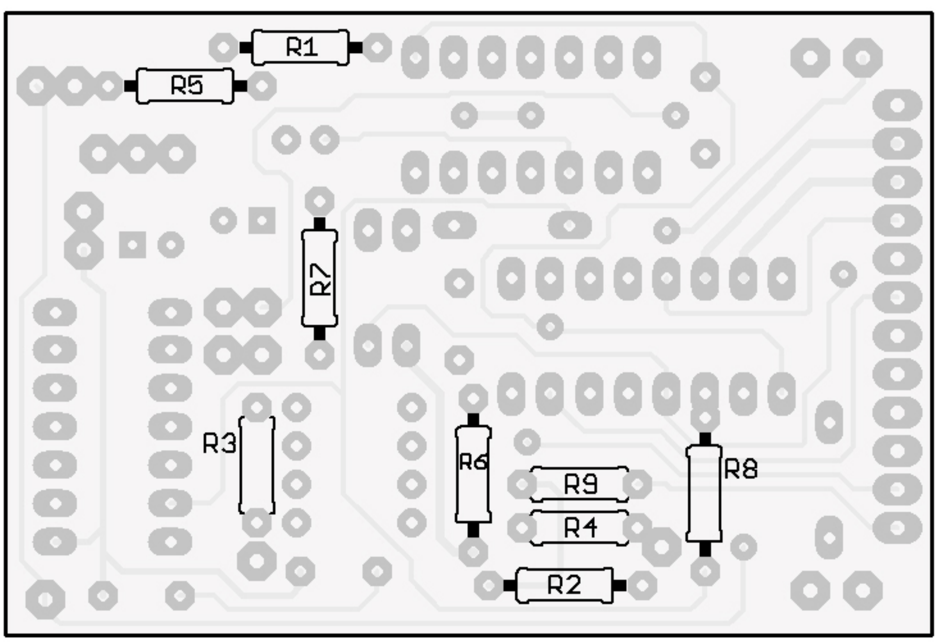
Values

R1, R2, R7, R9 - 100k
R5, R6, R8 - 10k
R3, R4 - 1k
P1 - 1M Linear Pot

C2-2.2uF
C4 - 10uF
C1, C3, C6, C7 - .1uF

IC1 - CD40106
IC2 - CD4017
IC3 - CD4011
IC4 - TLC272

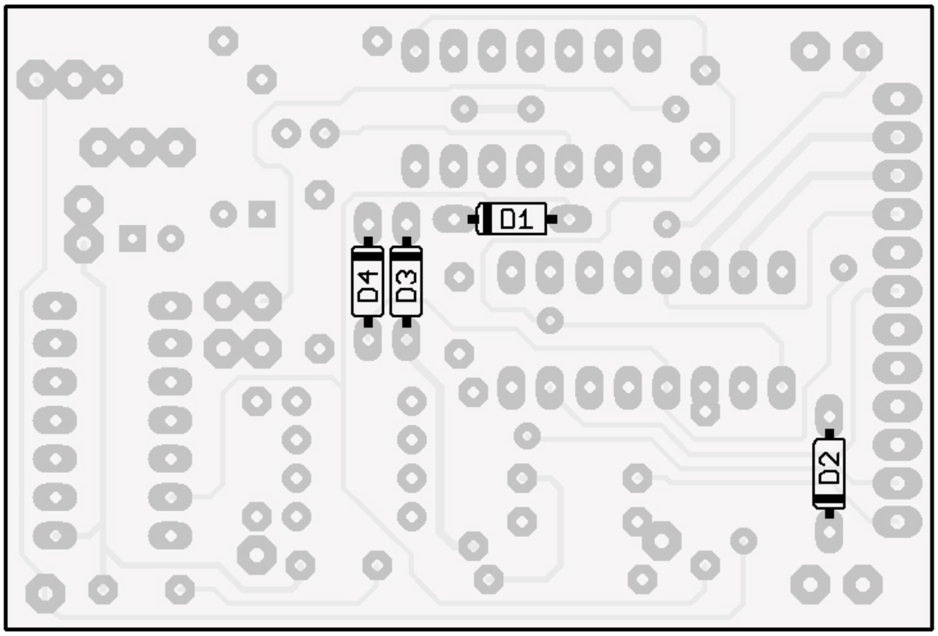
Step 1: Install Resistors



R1, R2, R7, R9 - 100k
R5, R6, R8 - 10k
R3, R4 - 1k

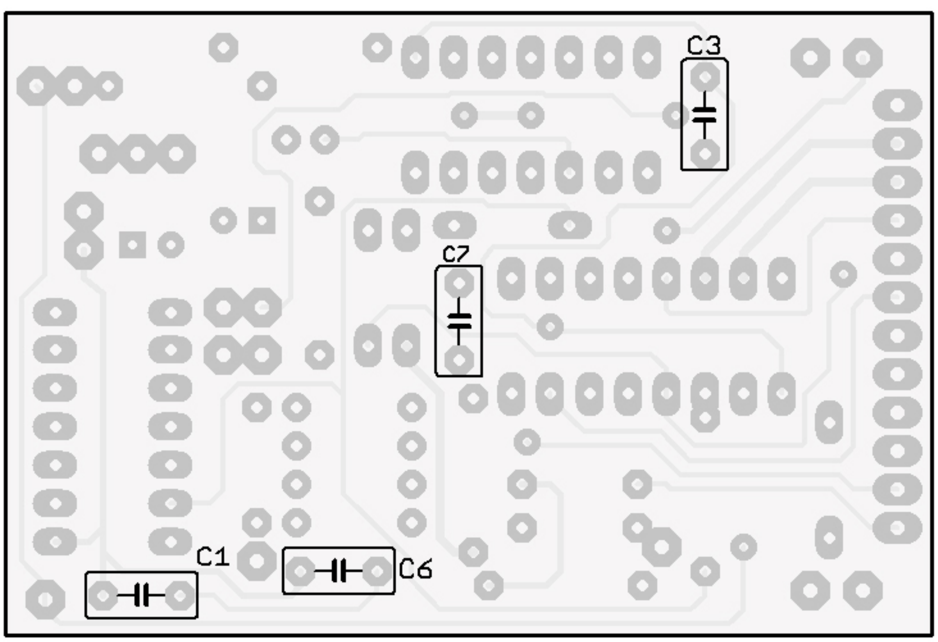
Pay attention of course to the value of resistor you are installing, and make sure that nothing is in the wrong place. It is probable that the board will not function correctly if you install a wrong value resistor somewhere. If you bought a full components kit, your resistors should have the value written on their tape-reel packaging to make it easier for you.

Step 2: Install Diodes



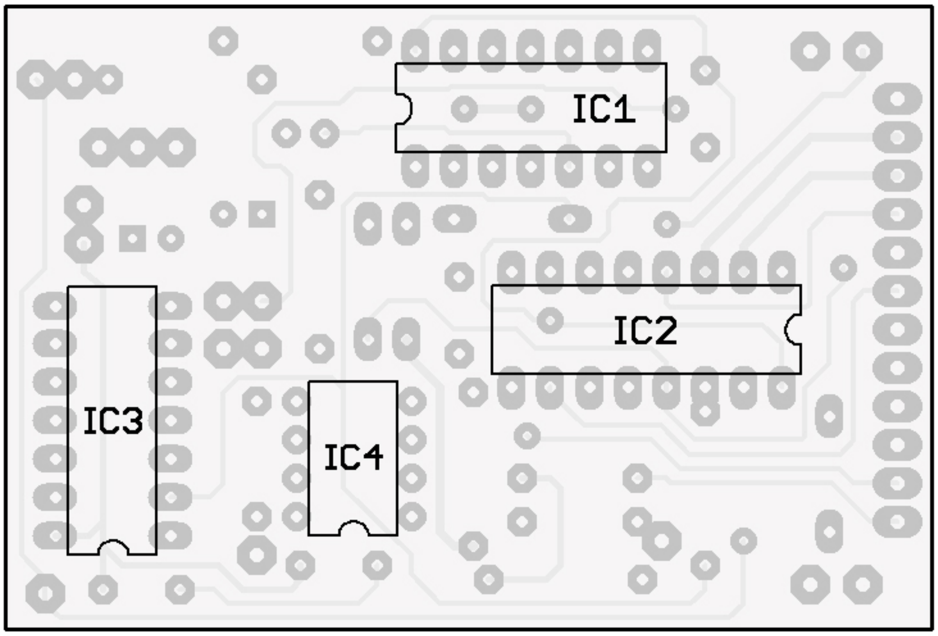
Next up is the diodes. Be careful not to apply heat for too long, as diodes can be damaged from excessive heat. Make sure that they are installed in the correct direction. The band on the diode is the negative side, and should be installed on the side with the band on the board silk-screen.

Step 3: Install IC Bypass Capacitors



Now, install the four ICs' .1uF ceramic bypass capacitors, C1, C3, C6, C7

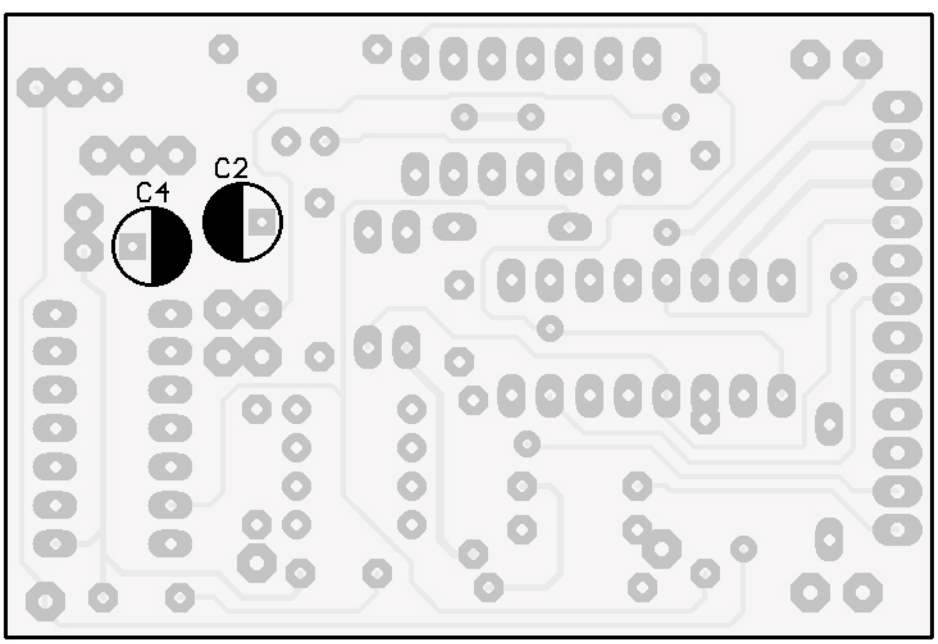
Step 4: Install IC Sockets



IC1 - CD40106
IC2 - CD4017
IC3 - CD4011
IC4 - TLC272

Install the four IC sockets, being careful to match the notch's orientation to the silk-screen's orientation. A trick you can use for seating them nicely is to place the socket in the board, flip the board over so that it's resting on the socket, and then "tack" only one leg in place. Now, touch the soldering iron to the one leg while flattening the board onto the socket with your hand, seating it flush against the board (being careful not to burn yourself, of course) -- that way, you can get the socket "tacked" in place perfectly before soldering up all of the pins.

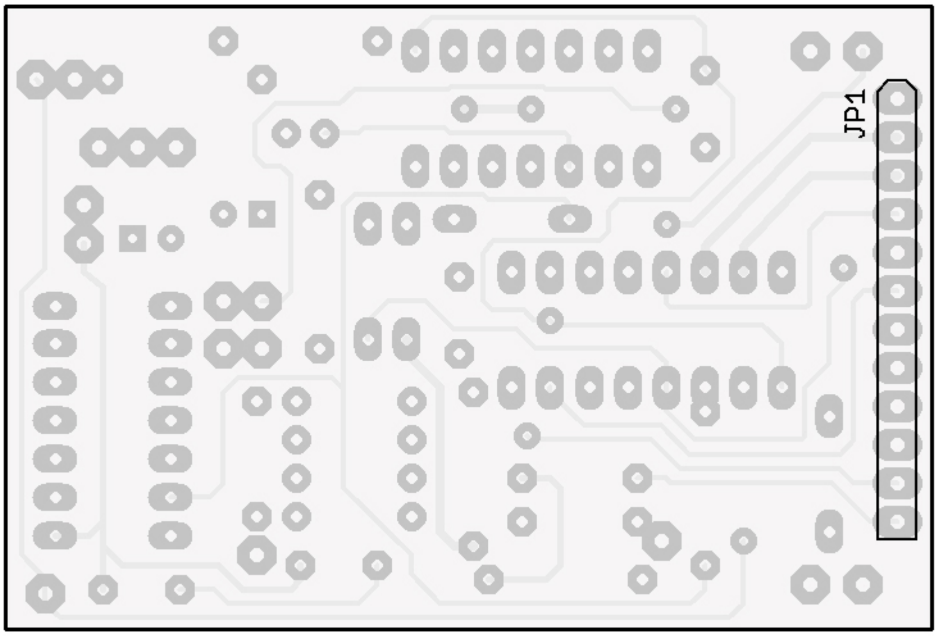
Step 5: Install Electrolytic Capacitors



C2-2.2uF
C4 - 10uF

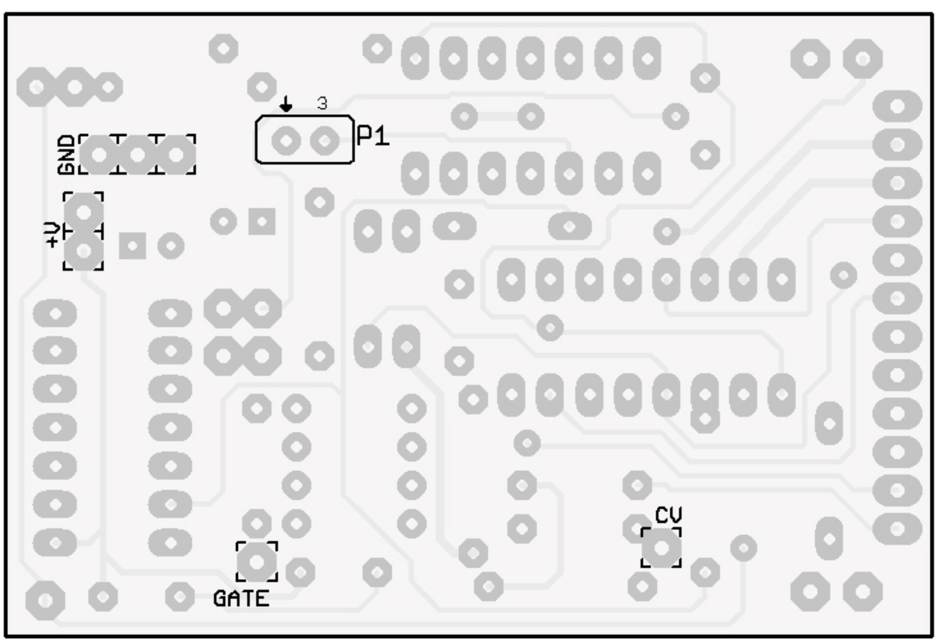
Install the remaining two electrolytic capacitors, being careful to note the correct polarity. The shaded half on the silk-screen legend indicates the negative lead of the capacitor.

Step 6: Install Board-to-board Header



The last soldering step is to install the male header into the OPPOSITE side of the board. By doing this, you are making it so that the logic board will plug in to the back of the control board. So, just to make that clear: when you are looking at the component side of the board we have been working on, the header should be facing AWAY from you on the other side of the board when installed correctly.

Step 7: Wiring and Tempo Pot



Install the crucial wires that lead to the logic board. Highlighted are the ones integral to the base model for standard functionality. If you want to chain multiple modules or use external clocking, you can continue on to the optional steps after the control board assembly. I recommend electrically assembling all of the logic boards before tying them together or performing other modifications on them. It makes it easier to work with them when you aren't dealing with a nest of wires yet.

P1 is a 1M linear potentiometer. It sets the tempo of the clock and should be mounted to the panel wherever you choose to put it. The "3" on the board legend designates the right-most pin on the potentiometer when the shaft is pointing towards you. The arrow indicates the wiper (centre pin). +V and GND are pretty self-explanatory. GATE is the buffered GATE out, and is ready for connection to a VCA or drum module. CV is the buffered CV out and is ready for connection to a voltage controlled module. Don't forget to connect GND to the negative part of the jacks you use for carrying GATE/CV out. That is what the extra GND pad is for!

Done!

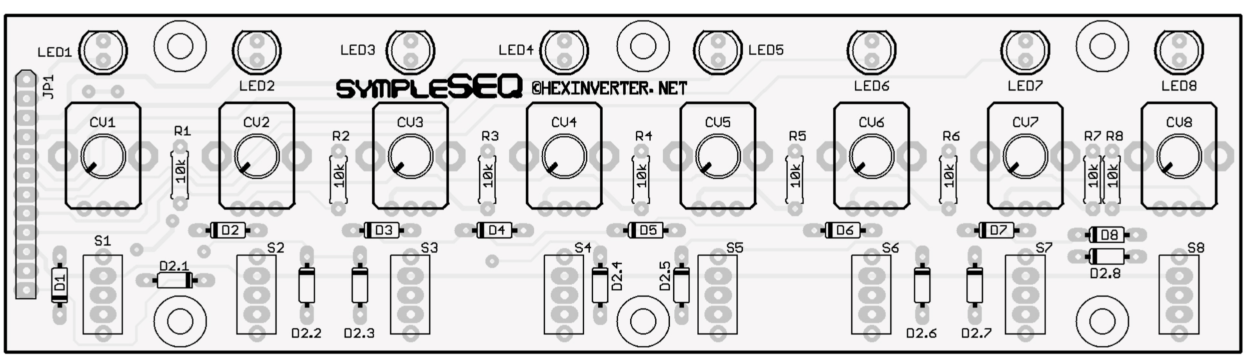
You are now done the logic board assembly. If you are not chaining multiple sequencers or doing any external pause/reset/clock mods, then feel free to install the ICs into their sockets now.

Continue on to the control board assembly below.

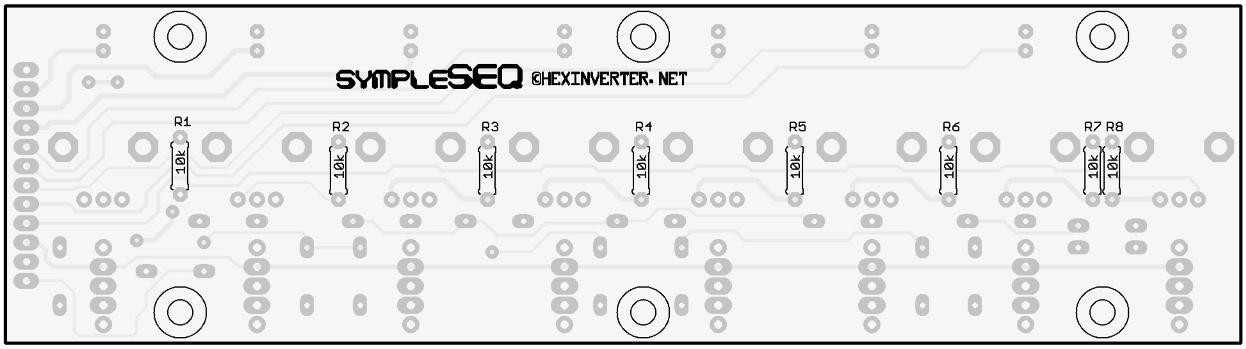
Control Board Assembly

The control board is regretfully pretty redundant to put together, but if you find yourself getting bored, just comfort yourself by imagining how awful it would be to wire all of these parts by hand! It doesn't seem so bad now, does it?

Main Component Overlay

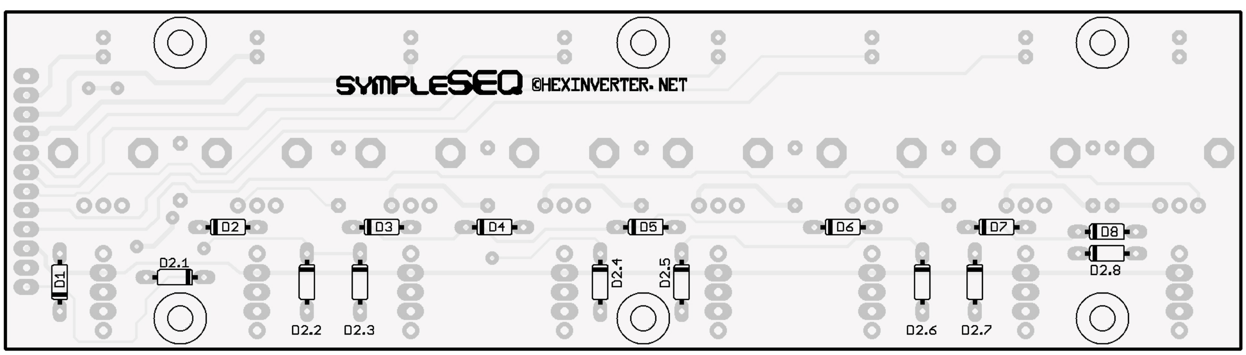


Step 1: Resistors



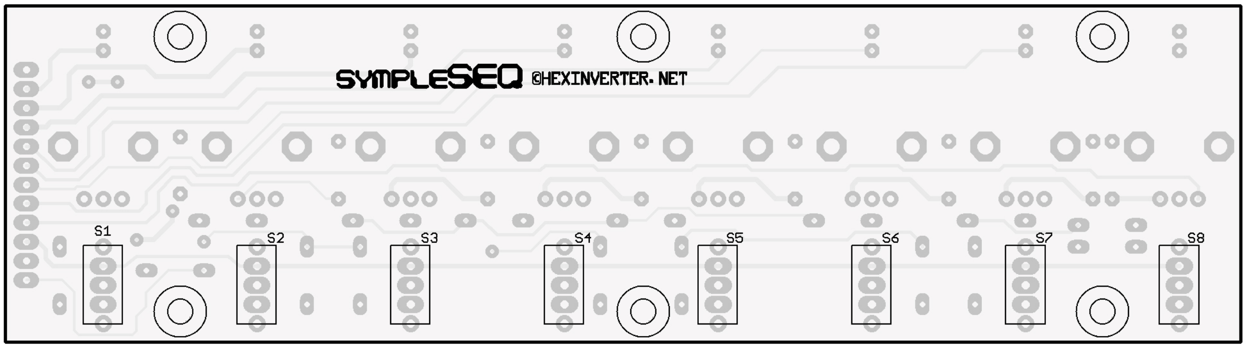
Install resistors R1-R8 which are all 10k current limiting resistors for the LEDs. Note that depending on the colour and intensity of LEDs you have used, you may need to alter this value. If you did not listen to me and used “normal” LEDs (not “super-bright”s), you’ll probably have to greatly reduce the value of these resistors just to get the LEDs to light. Blue LEDs tend to need more current than a green LED, and thus need a smaller resistor value to produce the same amount of intensity. If you are using a kit, you can ignore this part as the resistors should work good with the LEDs in the kit.

Step 2: Diodes



Install and solder the 16 diodes onto the board, again making sure that they are installed the correct way. Things will not work properly if they are installed incorrectly.

Step 3: Switches



The switches have a flattened edge on one side of the post that you may wish to orient the same on all of the switches for aesthetic reasons. The switches themselves are uni-directional, so exact orientation does not matter. It's up to you whether the flattened part points up or down.

Put 3 or 4 switches into the board and then flip it over such that the board rests on the switches. You may not be quick enough to get them all in place before gravity takes over when flipping the board around, so don't sweat it if a few fall out.

Each switch has 5 posts - the centre 3 are actual electric contacts, and the outer 2 are mechanical support.

Now, solder only one of the outermost mechanical posts of each switch. Flip the board back over so you can see the switches. Apply heat to each single "tack" joint you made one at a time, while seating the switches with your fingers or pliers (being careful not to burn yourself) such that they are perfectly flush with the board and not lopsided. This is an absolutely crucial step, as it is important that the switches be perfectly aligned when putting the board behind the panel holes.

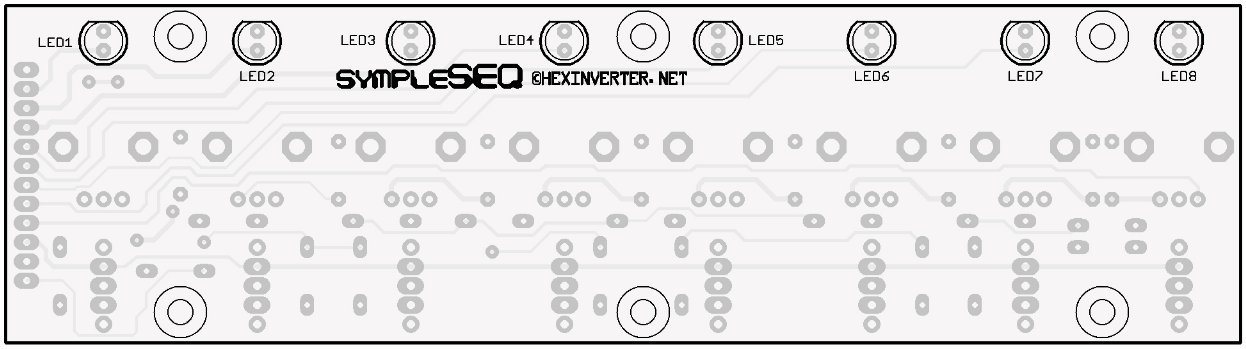
Repeat until all of the switches are "tacked" in place with only one solder joint.

Now you can do all of the remaining outermost mechanical contacts for the 8 switches.

Now comes the most critical part of the entire assembly process. In case you have not noticed, these switches are extremely small. Due to their size, they are VERY sensitive to heat. You can easily damage them if you hold the soldering iron to them for too long (which is why we did the mechanical contacts first, as they don't actually heat up the insides too much). You should be looking at a bunch of un-soldered sets of 3 inner pins - one set for each switch. What you are going to do is go from switch to switch soldering ONLY ONE pin at a time. If you take more than 2 or 3 seconds on one pin, STOP and go to the next whether it's done or not!

Let the pin cool off for at least 30 seconds before going to it again. Do 1 pin for every switch, then go onto something else for a minute before returning for the next set of pins. This way there is virtually no way you'll ruin a switch. I have included one extra switch for every kit just in case you mess one up. Once all of the switches are done, pat yourself on the back! The hard part is done.

Step 4: LEDs



This part is a little funny. In hindsight, I should have developed the project with LED spacers in mind. For now, we'll have to space the LEDs manually. Once again, we're using the "tacking" method where we solder only one lead and then position it where we want using that one joint.

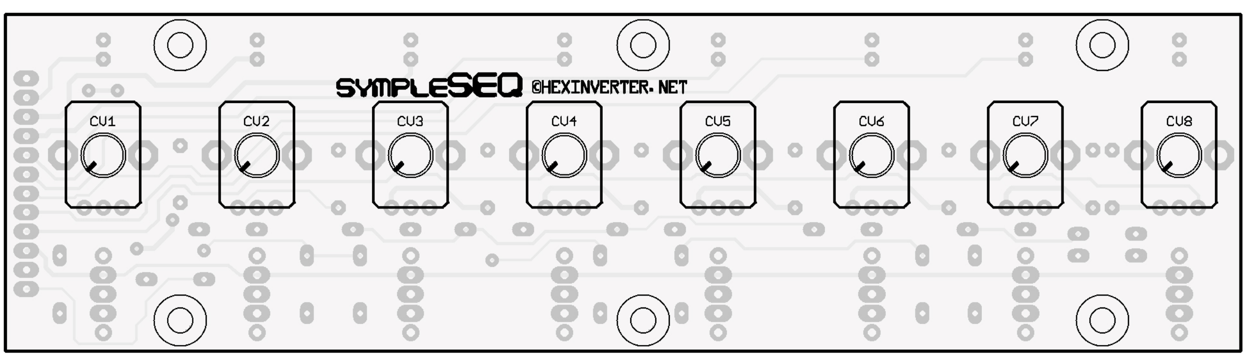
Place all of the LEDs in their spaces, noting polarity. How far you put them in really does not matter as we will adjust it in a moment. The flat edge indicates the negative side, and should be pointing up on the board when looking at it from the component side.

Solder only one leg of each LED, being careful not to overheat them. LEDs can be damaged by heat pretty easily.

Now, look at the board from the side so you have a good view of the LEDs and the switches you installed in step #3. Compare the base of the LED you are positioning to the area where the cylindrical post on the switch turns into the rectangular base (ie: the area where the panel will be resting on the switches at), and carefully heating the LED for only a few moments, position the LED's base to be slightly lower than the toggle switch's reference point. This will mean that most of the LED will be sticking out of the panel, but not all of it.

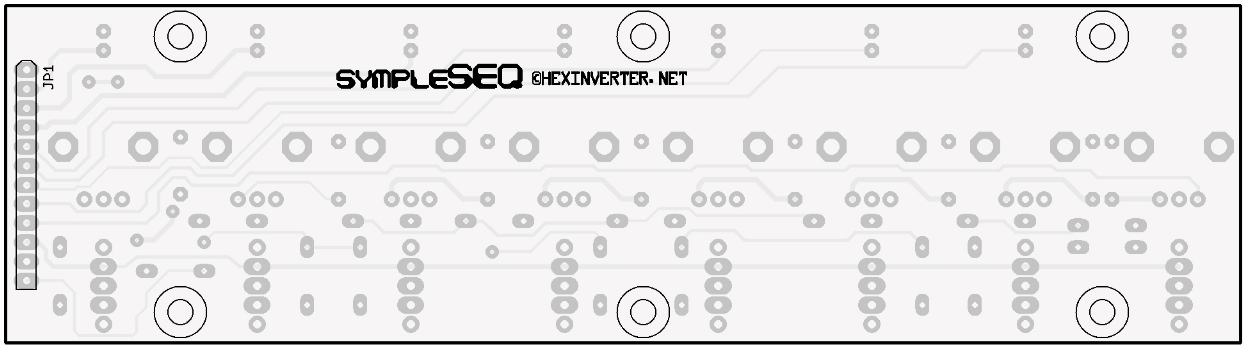
Repeat for all of the LEDs. This may sound confusing but is really quite easy once you've done one or two LEDs.

Step 5: Potentiometers



Install the 8 board mount potentiometers in their places, starting with the outer mechanical posts and adjusting them using the "tack method" as necessary to make them perfectly flat on the circuit board. It is crucial that they are aligned accurately so as to go through the panel in the correct drill spaces. Then, solder up all of the other pins!

Step 6: Board-to-board Header



Install the female 12 pin header on the OTHER side of the board (ie: if you are looking at the board from the component side, the header should be facing away from you on the other side). This will mean that the logic board can “plug-in” from the back.

Done!

You are now finished the control board assembly. Pat yourself on the back for a job well done! If you are making only one sequencer, the only remaining electrical step is to power up your sequencer and see if you assembled it without error. If you are constucting more than one or wish to have RESET/PAUSE or external clock functionality, we will go over those things next.

To use your sympleSEQ, it’s as easy as any other sequencer. The tempo knob controls tempo, and each step’s pot controls the CV for that step. The gate only outputs on a step if its switch is in the UP position for that step. You can make the sequencer reset at a different point other than 8 steps by putting the switch for that step into the DOWN position. If you put the 7th step’s switch to the reset (DOWN) position, you will have a sequence of 6 steps. You may notice that if you have more than one point selected as a reset point, the sequencer does strange things. This is a limitation of the low parts count/design implementation of the sequencer. Just keep this in mind when playing live -- only ever select one reset point at a time (unless you want chaotic behaviour of course).

Wiring Multiples

I just want to make it clear for everyone that intense documentation of how to do every possible interface configuration Ben (the panel maker) is offering will likely not be provided by me. It will be up to you to determine how to wire things based on the schematics.

You will have to bend it to your will if you wish to do something out of the scope of the base model. Don't feel left in the dust -- a few forum members have been developing/theorizing modifications since the initial funding campaign for the project!

Due to the way the electronics work (and please keep in mind that the entire module is designed around lowest parts count/easiest solution in mind), there are some limitations on how you can set things up in terms of multiple modules.

sympleSEQ was designed a certain way and thus is not universal to everyone's desired configuration right out of the box. I will try and outline the way the design intends to be used in the clearest form possible:

Option 1: Master Clock w/ Slaves (one big sequencer)

One master clock with "x" number of slaves. RST chains from master, PAU chains from master, CLK chains from master. One RST/PAU input/switch. All modules are synced. (Recommended if you have no other sequencers).

Install the components/potentiometer for one clock only (or none if choosing to be externally clocked), and then chain the logic boards to each other using the pre-defined pads at the top and bottom of each logic board. It should connect like this:

CLK.A -> CLK.B
RST.A -> RST.B
PAU.A -> PAU.B

...and of course you want your slave boards to be connected to the +/GND power connection.

Be aware that not all of the pads on the logic boards will necessarily be used.

Thus, there is only one clock. When one sequencer pauses, all of them pause. When one sequencer resets, all of them reset. Get it?

Option 2: Multiples Without Clocks (external clocking)

All separate without clocks (external clocking)

RST/PAU/CLK are all independent and rely on external input. Sync depends on clock/pause/reset sources. (Recommended if you have any other sequencers you plan to use this with).

This is how I would recommend running things since it would allow you to do things like divide the clock frequency of one of the channels to get different step lengths (this could be useful if making a rhythm sequencer to get more out of the 8 steps).

This may sound confusing, but it really is not. Just run multiples of your clock source to control each sequencer independently. If you want them to sync, make sure it is the same clock source. If you use stackable banana jacks like me, this is a pretty easy procedure. Be aware that strange things may happen if you wire all of the PAU/RST chaining pads together yet have different clock sources feeding in. This is not supported out of the box and you will have to experiment on your own if you want to do that. It may or may not work as you expect.

It will be up to you to modify it so that you can switch between clock sources via switch/switching jack if you want to be able to change modes. I would have had to require jumpers on the PCB to implement this out of the box, which is unacceptable as it would complicate first time builders' assembly, and not everyone wants this feature.

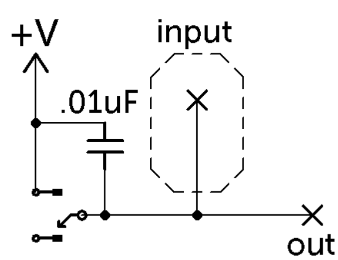
Reset, Pause and External Clocking

The baseline sympleSEQ does not require these modifications to function, but based on requests from people, the functionality has been incorporated into the PCB design. Implementing them will mean you will need more inputs, and possibly switches if you want (as per the bill of materials), but in general they should be pretty trivial to implement thanks to the circuitry being mostly on board.

Reset/Pause Configuration

You have some choice here in terms of which way you want to trigger pause (PAUSE) or reset (RST) functionality. There are two ways to trigger a pause or reset. You can choose to trigger the feature by running an input jack and then supplying a +V signal whenever you want it to happen, or you can wire a switch (toggle or momentary - your choice) that when "on", ties the pad on the circuit board for that function to +V. You can also wire it up to accept either.

Here is the diagram of how to wire it up. The part surrounded by the dashed line ("input") is for a manual external input with a jack, whereas the capacitor and switch form the switch trigger. The output ("out") ties directly to either "RST" or "PAUSE" on the logic board with some wire of your choice. Now, you can trigger reset or pause with either the flick (or push) of a switch, or an external +V signal routed into the sequencer.



External Clocking

When choosing to feed an external clock signal into sympleSEQ, you must send in a positive square wave at or very close to the +V supply that sympleSEQ is running on. When clocking externally you may omit the circuitry on the logic board that is surrounded by the black border, but do not omit R1 which is a pull-down resistor necessary for stable functioning. Only omit R5 when you have multiple sympleSEQs chained off the same clock. To input a clock signal into sympleSEQ, it's as easy as running a lead from either the CLK.A or CLK.B pad on the logic board to a jack for interfacing.

ADVANCED USER TIP: Please note that an interesting feature of sympleSEQ is that when the duty cycle of the clock changes, so does the length of the gate on the output. This is how the "Baby10 gate tie problem" was solved. This means that you could input a PWM controllable square wave to sympleSEQ in place of the stock clock and have gate length control. If you do not know what I am talking about here, don't fret. I'd expect only a seasoned DIYer to know how to implement this.

When you want to use sympleSEQ's clock to control other gear, make sure first that the gear you want to clock is compatible with CMOS level clock circuitry. Most are, but it helps to verify. Clocking other gear is as easy as running a lead from either the CLK.A or CLK.B pad on the logic board to a jack which goes to the sequencer you want to sync.

Mechanical Assembly

Well, you have gotten this far, and hopefully the electrical assemblage of your sympleSEQ is complete and without fault. Now it's time to finish the project up by putting it in an enclosure that you would like to use it in. If you have purchased ready-made panels from Re:Synthesis, you can of course skip the whole drilling step.

I will start off by saying that I highly recommend you have a drill press for this. There are quite a few holes to drill on sympleSEQ and while it is not impossible with a hand drill, it will be pretty fatiguing.

Start off by printing out the drill template in the appendix. It has been made such that it is easy to affix it to your panel and drill the holes for the control board to go through. I will leave it up to you to locate and drill the positions for tempo, output jacks and any other additional features you wish to include. Everyone's configuration will be different, of course.

It is crucial that you try to get the holes as accurate as possible, as all 8 steps of the board need to fit through the holes nicely.

Once your holes are drilled, it's just a matter of installing the 6 machine screws holding the control panel in place, and then affixing the logic board to the back. Congrats on finishing your sympleSEQ!

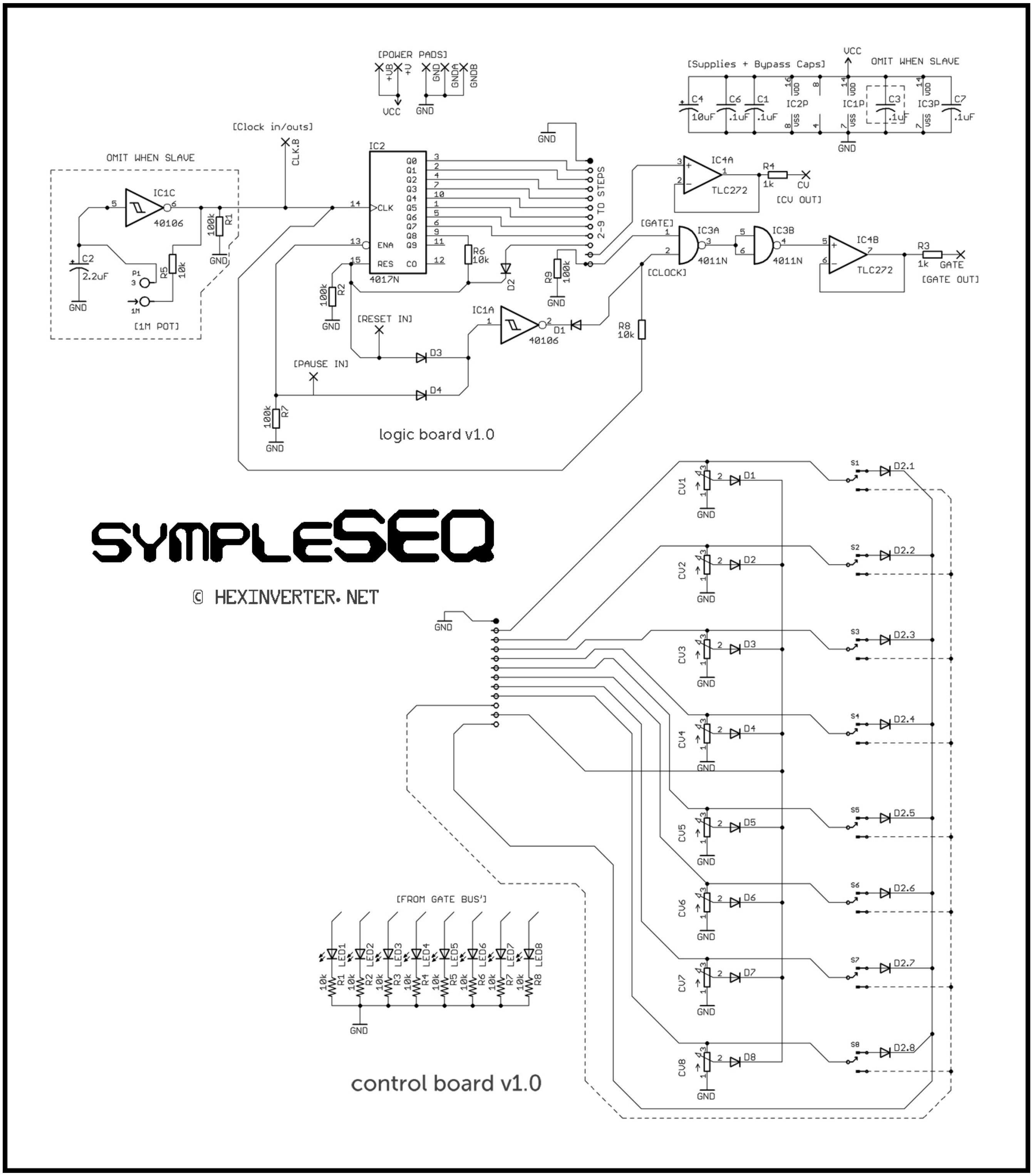
Now, go make some music!

Appendix

Here you will find resources that will aid you during your build.

Schematics

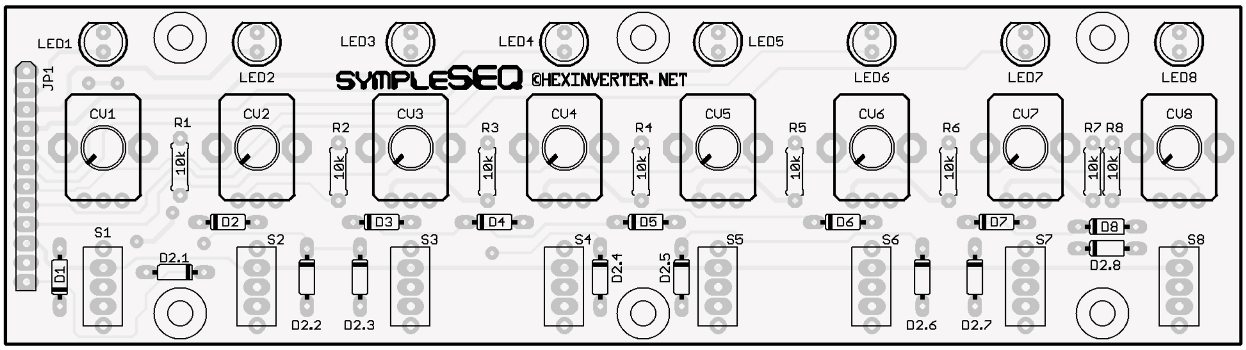
You can find a larger online version of the schematics at: hexinverter.net



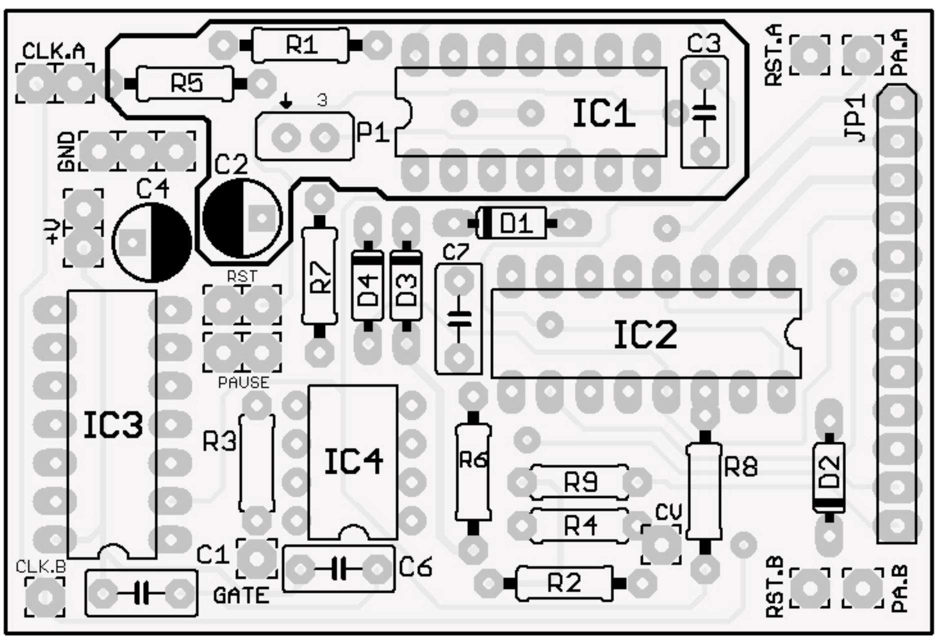
Board Overlays

These are duplicated in the assembly guide, but you may find it useful to print them out on one page here for reference. Think of the trees before you do so, though!

Control Board



Logic Board



Bill of Materials

You can also find the online version of the bill of materials on Google Docs.

Part	Quantity	Designator/s	
[BASE MODEL] - included in full kit.			
		Logic Board	**Control Board**
ICs			
CD40106 Hex Inverter	1	IC1	
CD4017 Decade Counter	1	IC2	
CD4011 Quad NAND Gate	1	IC3	
TLC272 Dual Op-amp **	1	IC4	
*** You can also use TLV272, or pretty much any other TL-series dual linCMOS single supply op-amp with the same pinout.*			
NOTE: I highly recommend using IC sockets. You will need 2x14 pin, 1x16 pin and 1x8 pin DIP sockets if you are sourcing your own parts. The full kits include these sockets :)			
Capacitors			
.1uF Ceramic Capacitor (5mm spacing)	4	C1, C3, C6, C7	
2.2uF Electrolytic Capacitor	1	C2	
10uF Electrolytic Capacitor	1	C4	
Diodes			
5mm High Intensity LED	8		LED1-8
1N914 Small Signal Diode	21	D1, D2, D3, D4, D5	D1-8, D2.1-8
Resistors			
Alpha 9mm PCB Mount 100k Potentiometer	8		CV1-8
1M Linear Potentiomer (panel mount)	1	P1	
100k Resistor	4	R1, R2, R7, R9	
10k Resistor	11	R5, R6, R8	R1-8
1k Resistor	2	R3, R4	
5% should be fine for all resistors. You may wish to use 1% to increase temperature stability for the clock resistor (R5), but it's not absolutely crucial. If you have them around, you may as well use 1% for everything, but if you don't then don't worry about it!			
Switches			
PCB Mount super mini ON-OFF-ON Vertical Toggle Switch	8		S1-8
[PARTS YOU MUST GET] - NOT included in full kit!			
Output Jacks (1/4", Banana, 1/8", etc.)	2	CV Out, GATE Out	
SPDT Toggle Switch (only if building standalone!)	1	Power	
9V Battery Clip (only if building standalone!)	1	GND, +V	
[OPTIONAL PARTS] - NOT included in full kit!			
Input Jacks (1/4", Banana, 1/8", etc.)	3	RESET In, PAUSE In, CLOCK In	
SPDT Toggle Switch	2	RESET, PAUSE	
.01uF Ceramic Capacitor (soldered across switch leads)	2	RESET, PAUSE	
Knobs 6mm shaft/19mm max. diameter	9	CV1-8, P1	

Drill Template

Print out this page as a drill template for drilling your control board's panel holes. Tape the cut out sheet to the panel and drill away! Easy. (there are multiples here to save the number of sheets you need to print if making multiples)

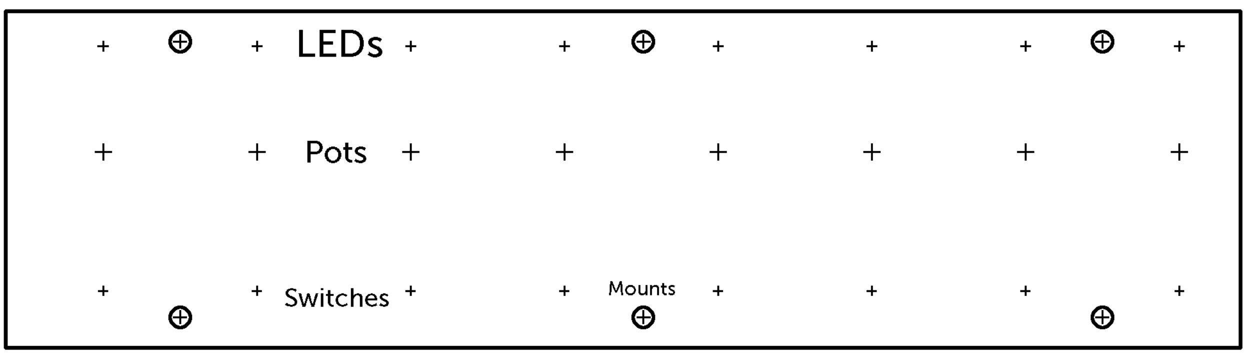
LED holes: 5mm drill (13/64")

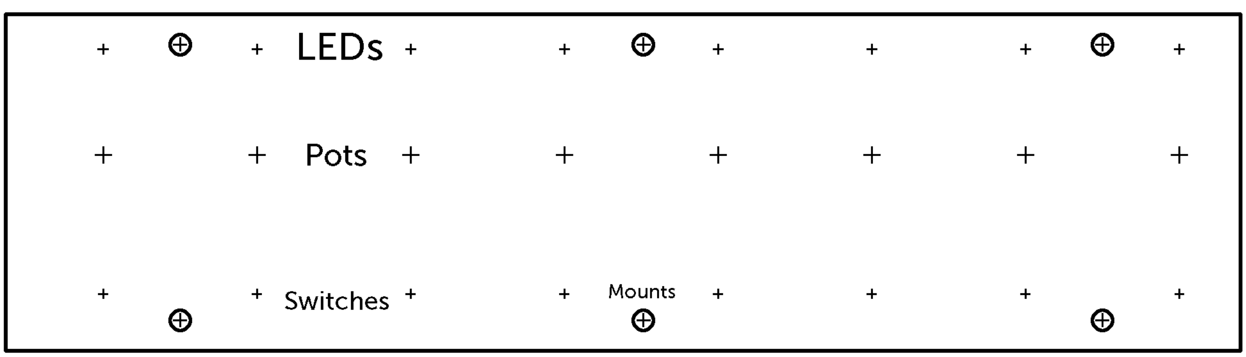
Pots: 6.35mm drill (1/4")

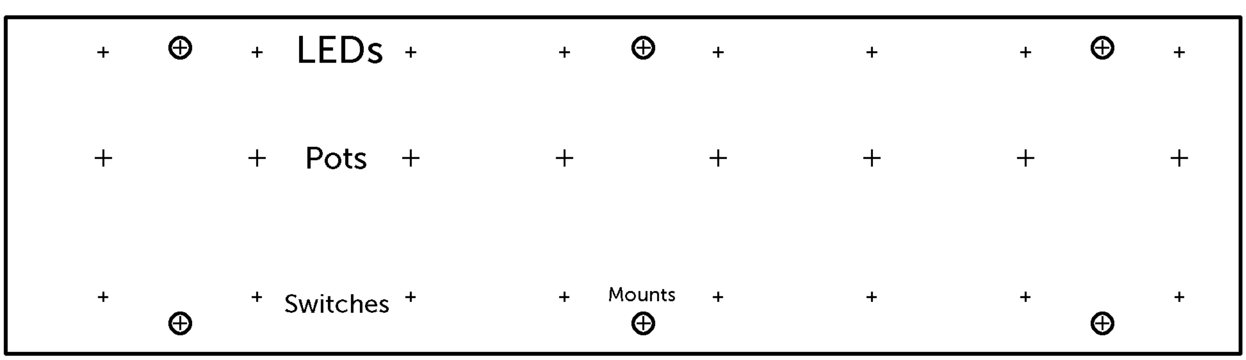
Switches: 5mm drill (13/64")

Mounts: 3mm drill (1/8")

Note that you may need to file out some of the holes to account for irregularities in your soldering job on the board. An offset switch may prevent the rest of the components from passing through their holes perfectly. Again, this can be greatly simplified by purchasing a ready-made panel from Ben at Re:Synthesis :)







Frequently Asked Questions

Q: Can I use sympleSEQ with my other sequencers?

A: Yes! You can use sympleSEQ either as a clock source, or clock it from other master sequencers.

Q: I think mine is buggy. Why does it act funny when I select more than one step to reset at with the switches?

A: This actually is not a bug. This is a side-effect of the minimalistic circuitry in sympleSEQ and is a known thing. It is important that the user understand they need to undo one reset point before setting another, or chaotic patterns may result! Unfortunately you can not have everything perfect when working with very limited components. This shouldn't harm the module to use it in this way if you enjoy the effect for musical purposes.

Q: How long does it take to make one?

A: This is all dependent on the builder's skill and experience level, and also whether you purchased a ready-made panel from Ben at Re:Synthesis or not. Generally speaking, you should be able to complete the electrical assembly in about an hour if you are experienced. A beginner will want to carefully follow the instructions, however, and thus it will take a while longer. Drilling and panel making will add a fair bit of time to this if you did not order a panel!

Q: I haven't built anything before. Am I capable of building sympleSEQ?

A: Yes! You are the target audience this project was originally conceptualized for. I have made every effort to make it possible for beginners to get themselves a sequential voltage source to start making sounds with. This is not to say that you will find all of the knowledge you need in this manual -- I have listed other resources in the "Soldering Tips" section that I strongly encourage you to check out. I have no doubt in my mind that if you are interested in electronics you are probably smart enough to learn along the way -- this alone should ensure you will be successful in your assembly. Of course, if you run into problems, me and others are available to help!

Q: I missed out on kits. Where do I find the board mount switches/potentiometers?

A: The Futurlec part number is listed in the Bill of Materials, right below the part listing. The Mouser part number for the potentiometers is also listed in the Bill of Materials.