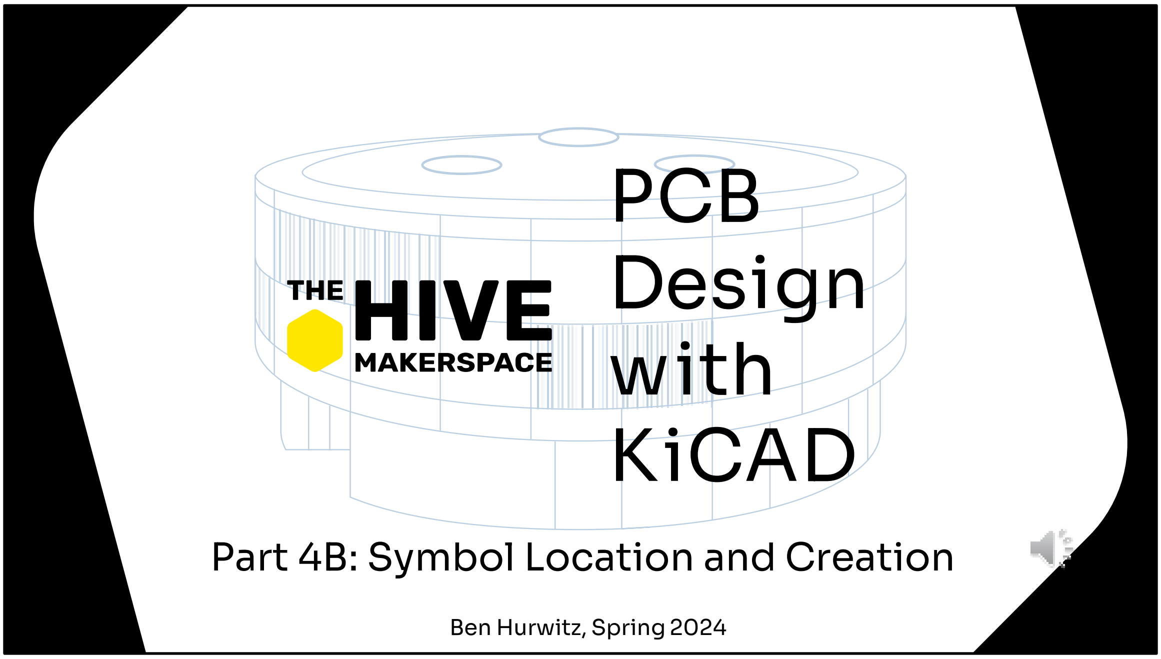
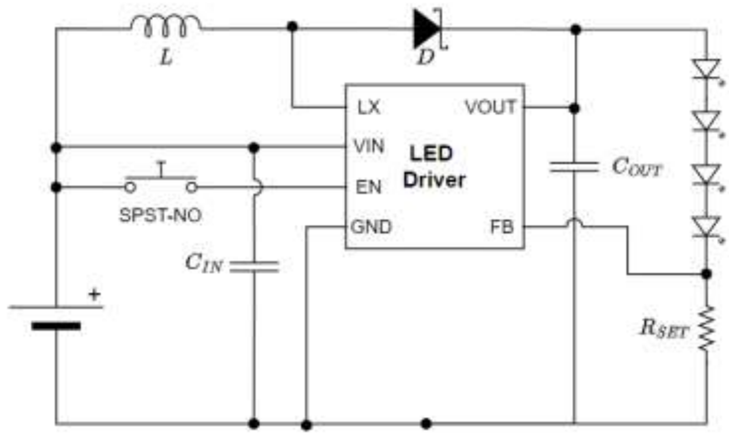


Hi, and welcome to part 4B of The Hive's PCB Design With KiCAD series. My name is Ben, and I'll be walking with you through this process. Part 4 as a whole will cover the entirety of the schematic creation. In this section, we'll cover how to locate device models online, and how to create symbols in KiCAD.

If you watched part 3, you might remember that in our design flow, there was a project setup and library creation step. Due to the relatively simplicity of this design, and that library management was not covered during the original workshop this tutorial is based off of, we're going to leave the library creations to parts 6 and 7. I strongly advise you to watch through those if you're considering doing design more seriously.

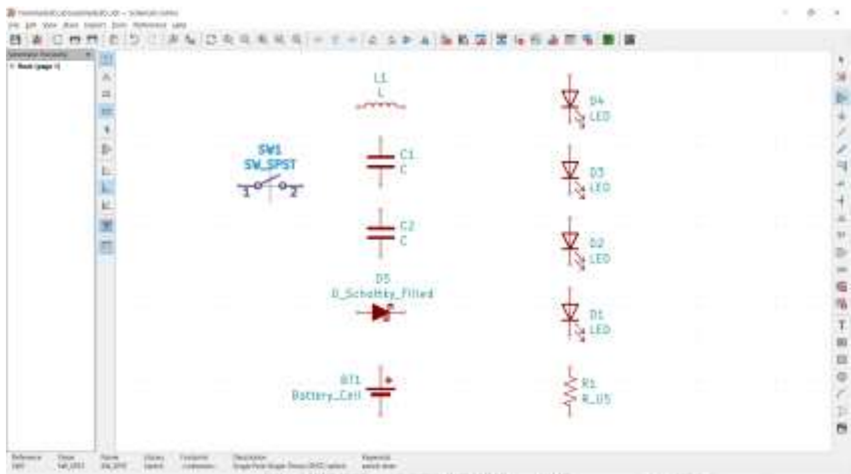
Anyway, let's get into KiCAD.

Circuit Reminder



Before we get into KiCAD, just a reminder of the flashlight circuit we're developing. Note that this image was not take from KiCAD, and therefore the symbols and graphics are different from those you are about to see.

Schematic Reminder



And this is a reminder of the schematic as it stood at the end of part 4A. We had added all the standard components, and were about to add the IC. If you've forgotten anything, I suggest you at least skim through that video (or the associated PDF).

Adding components – IC

- The last component to add is the IC
- The IC is slightly different than the other components we have put down, for two reasons:
  1. The symbol is “unique”, or at least, a symbol with the same pinout would be difficult to locate in the built-in libraries
  2. The footprint may or may not be in the libraries (although it is a standard package, so it likely is… but we’ll pretend.)

Adding the IC is the last step.

*It’s different than the previous symbols for two big reasons.

*First, the symbol itself is unique, or if it’s not unique, locating an IC with the same symbol that is already in KiCAD will likely be as much work as making a new symbol.

*Second, the footprint may or may not be in KiCAD either. It’s a standard package, so maybe, but we’d have to see.

Adding components – IC

- The first thing to do with ICs or other non-standard components is to see if they're already in the Symbol Libraries.
- Click "A" to add a symbol and filter for the RT4526.
- Did you find it?

It's always a good idea to check if they happen to be in one of the built in libraries though, so *go ahead and open the "Add Symbol" window and filter for the RT4526.

*Is it there?

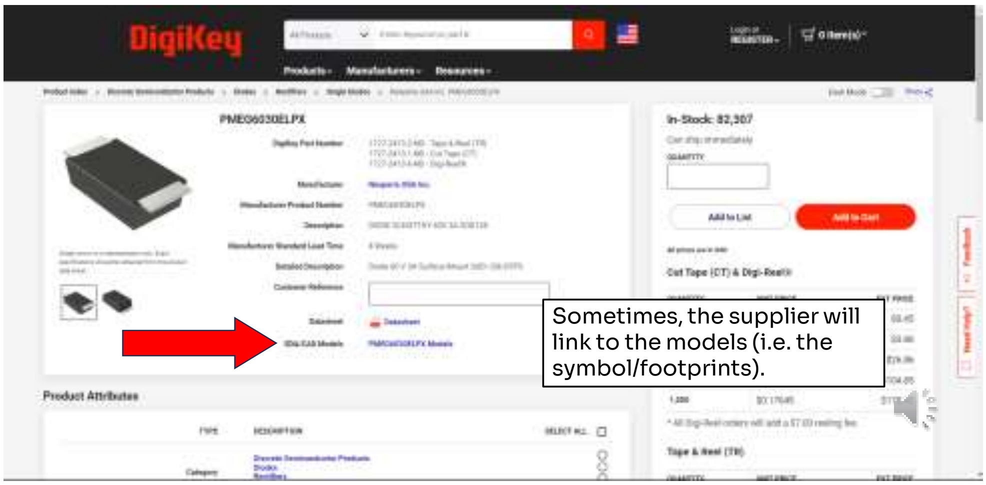
Locating a model for the IC

- You shouldn't have actually found anything, which is pretty normal for most ICs and non-standard parts.
- That's okay!
- We're now going to look to see if someone else has already done the work of generating a symbol and footprint.

Totally okay that it's not.

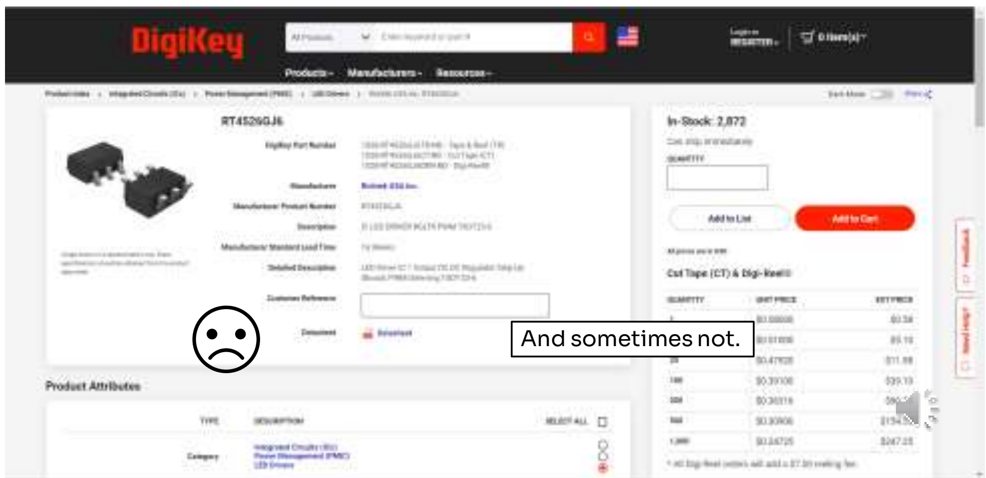
The next check is to see if the models have been professionally generated already. Always try to work smarter, not harder.

Locating a model for the IC

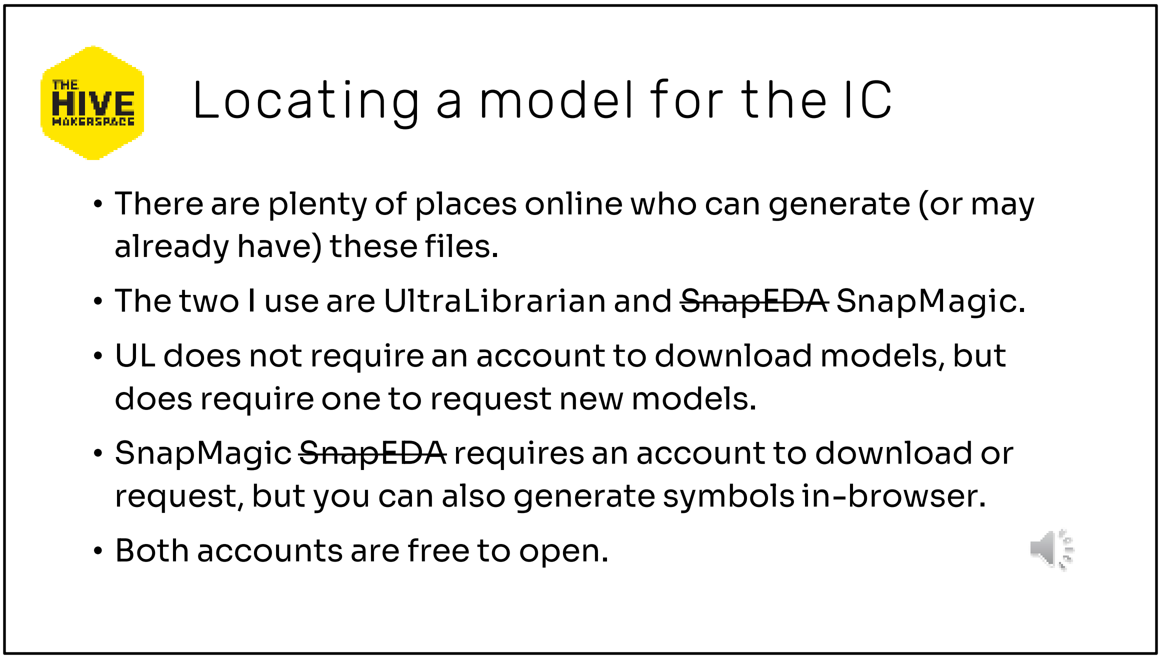


Sometimes, the suppliers will link to models.

Locating a model for the IC



Sometimes not.



If the supplier doesn't link, we can go look for them manually.

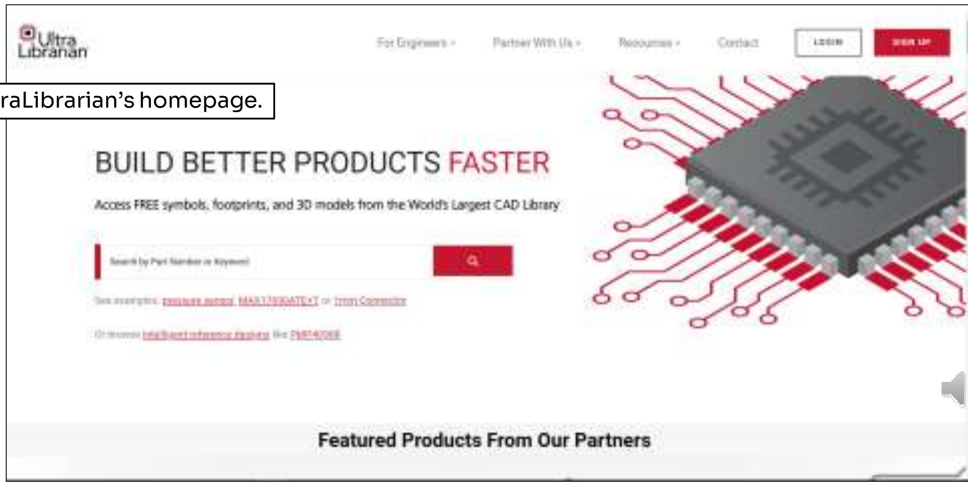
*The two places I have had good results with are Ultra Librarian and SnapMagic, which used to be called SnapEDA.

*Both require free accounts to request new models, but Ultra Librarian allows you to download pre-made ones without one.

*SnapMagic has a few additional tools that are available for registered accounts as well, like in-browser symbol and footprint generation.

**

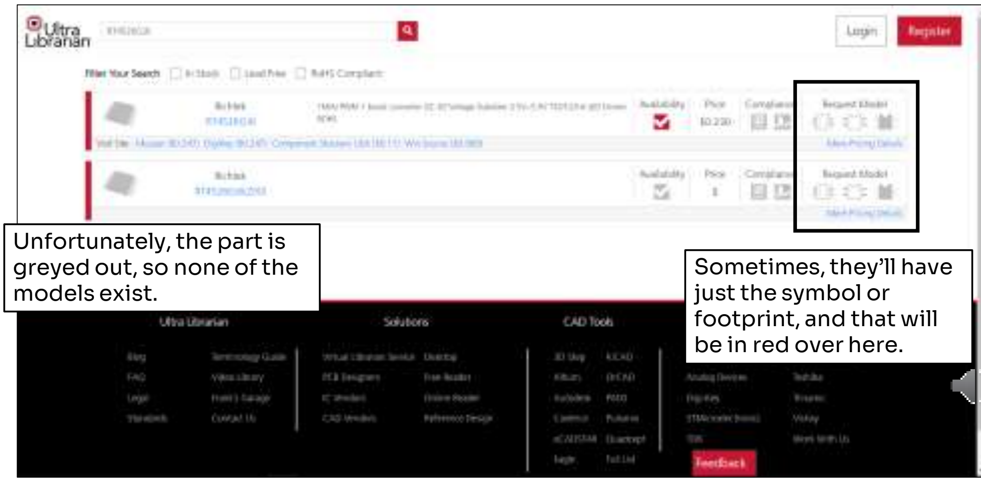
Locating a model for the IC



UltraLibrarian's homepage.

We'll start arbitrarily with Ultra Librarian. Search for the symbol you'd like a model for.

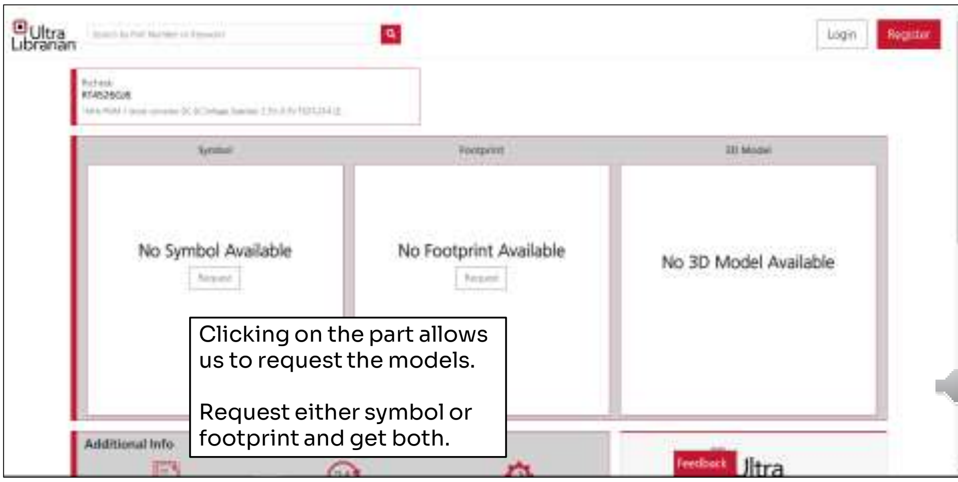
Locating a model for the IC



Unfortunately, the part is grayed, so no models exist.

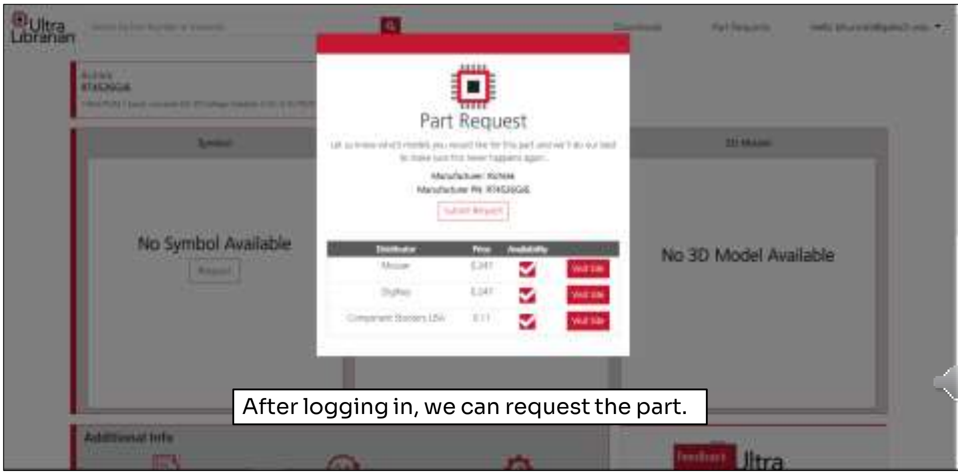
*Sometimes, the icons on the right will be filled in, indicating that they have a footprint or a symbol, but not today.

Locating a model for the IC



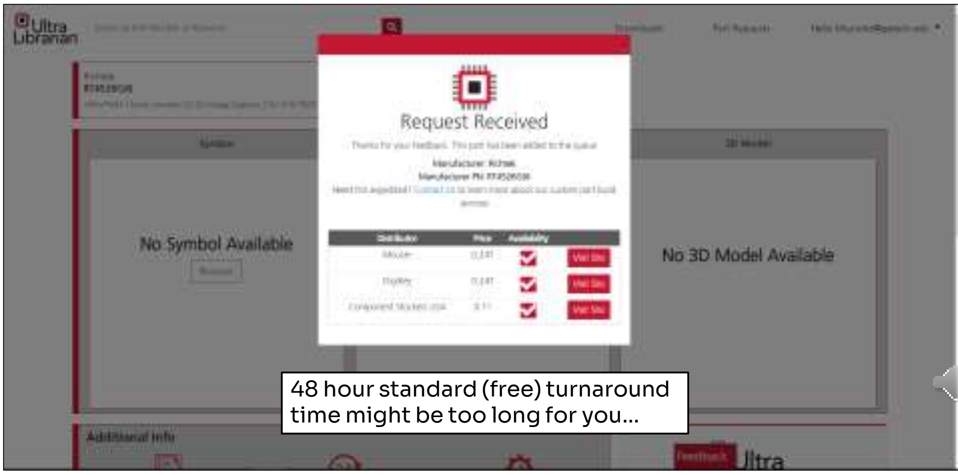
Clicking the part gets us to the part page, where we can ask them to create a model for us. Request either to get both.

Locating a model for the IC

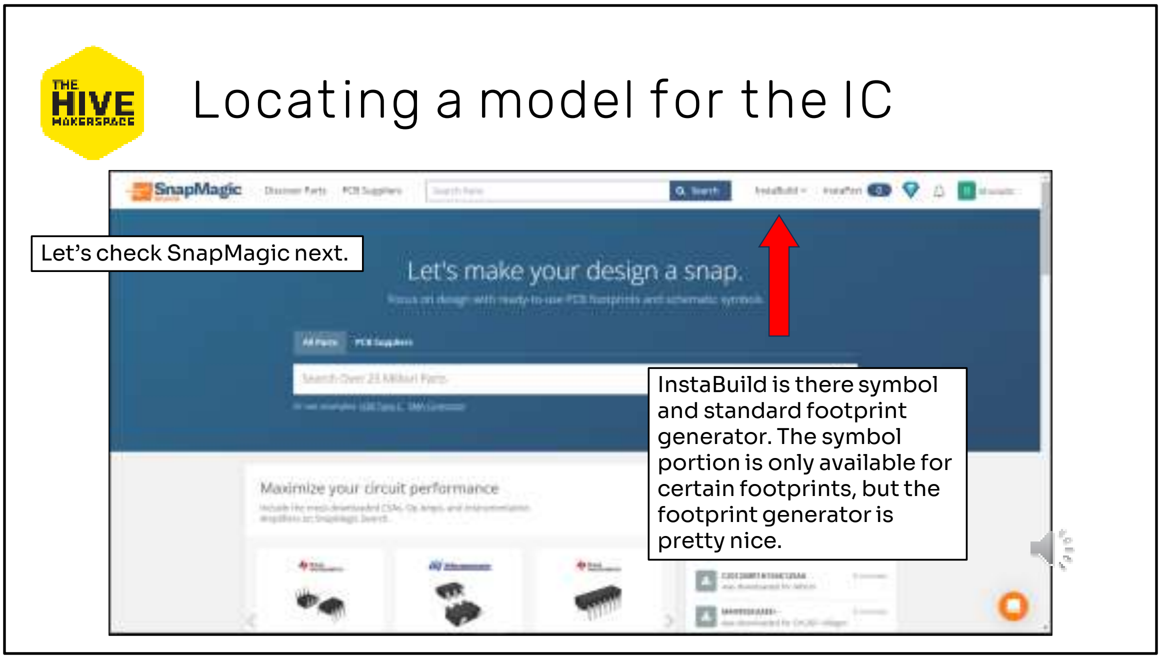


Looks like this

Locating a model for the IC



Standard turnaround guarantee is 48 hours, which may or may not be too long for you.



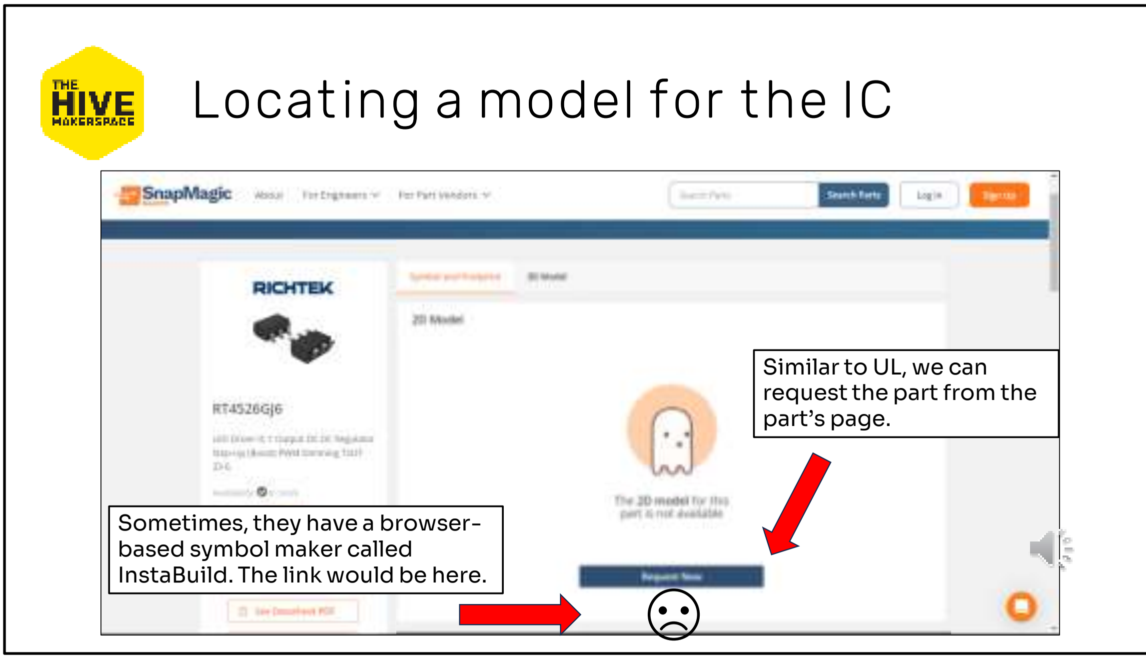
Let's check our other source, SnapMagic.

*SnapMagic's in-browser model generator is linked on their homepage, but while most standard IC footprints can be generated, the symbols are limited to select footprints only.

Locating a model for the IC

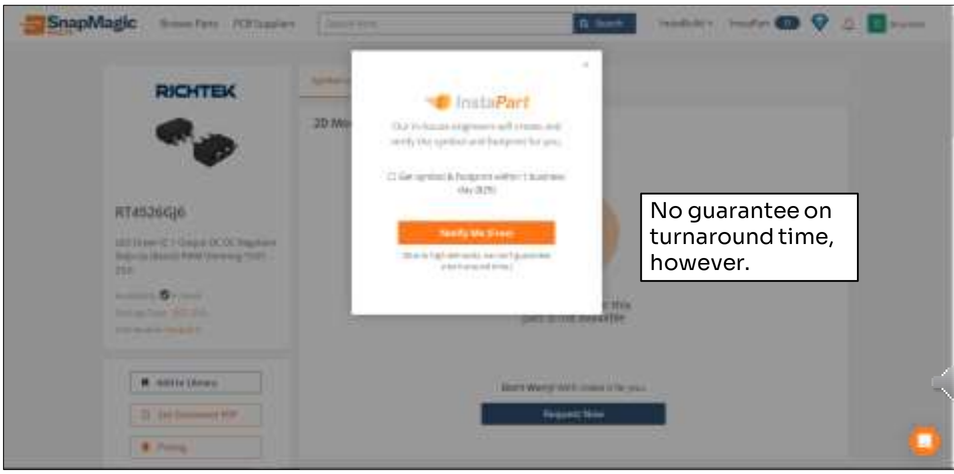
Searching for our part brings us to this page, *but be careful! Sometimes they recommend a part at the top that isn't right.

*Still, no models available – we can see this with the empty icons on the right.



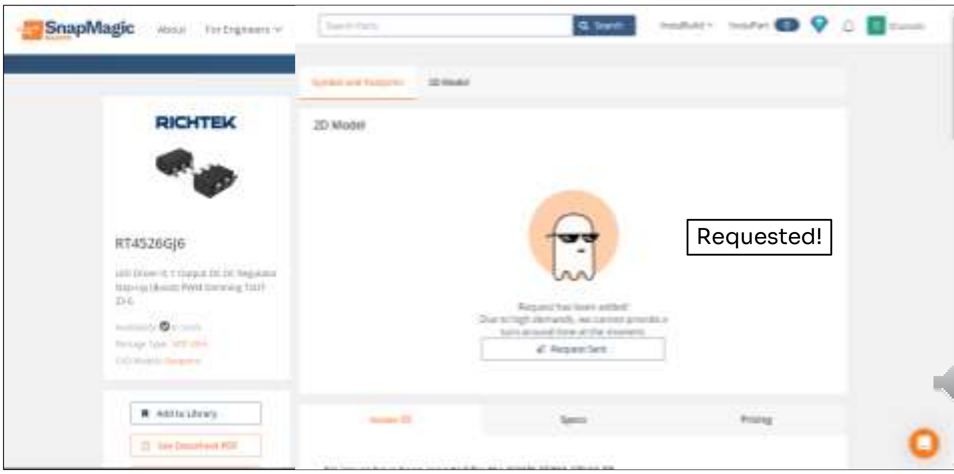
We can request the models from the parts page. *There would also be a link here if the in-brower symbol generator was available for this part, but it's not. Sad.

Locating a model for the IC



Unfortunately, SnapMagic doesn't have a guarantee on turnaround time.

Locating a model for the IC



But you get a cool ghost when you request the part.

~~Locating~~ Creating a model for the IC

- Struck out thrice. What to do?
- Can keep searching (someone *must* have made this before, right?), but returns may be diminishing.
- Because KiCAD stores symbols and footprints in separate libraries, we'll just go ahead and create our own symbol directly in KiCAD.
  - No need to create our own footprint (yet)

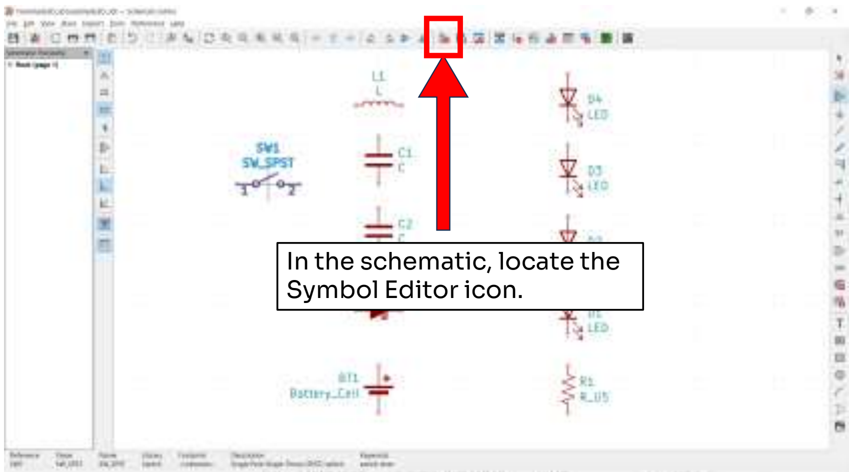
So three strikes. Are we out?

*We can keep searching randomly on the internet, but the returns will likely be diminishing, and the quality might be degraded.

*Instead, we can just go ahead and make the symbol in KiCAD directly.

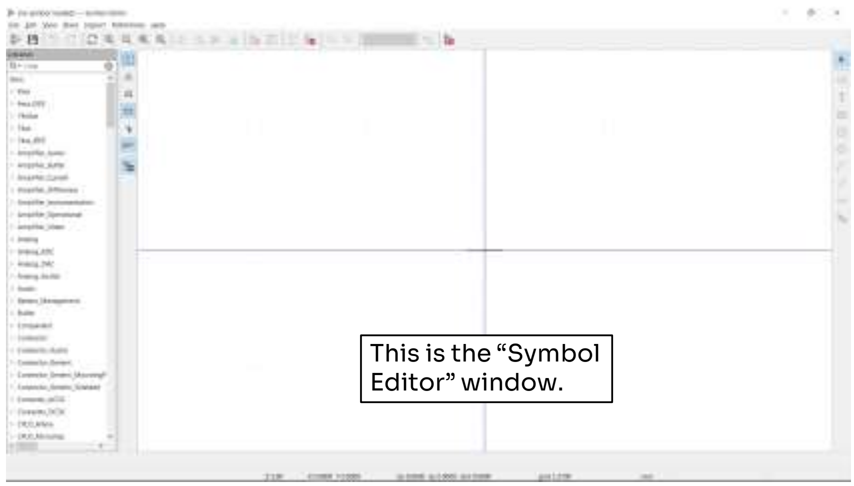
Symbols are relatively easy to make, and are less prone to errors than a custom footprint, though they can be tedious with many pins, so it's totally doable to make them without waiting for Ultra Librarian or Snap Magic to be done.

Creating a symbol for the IC

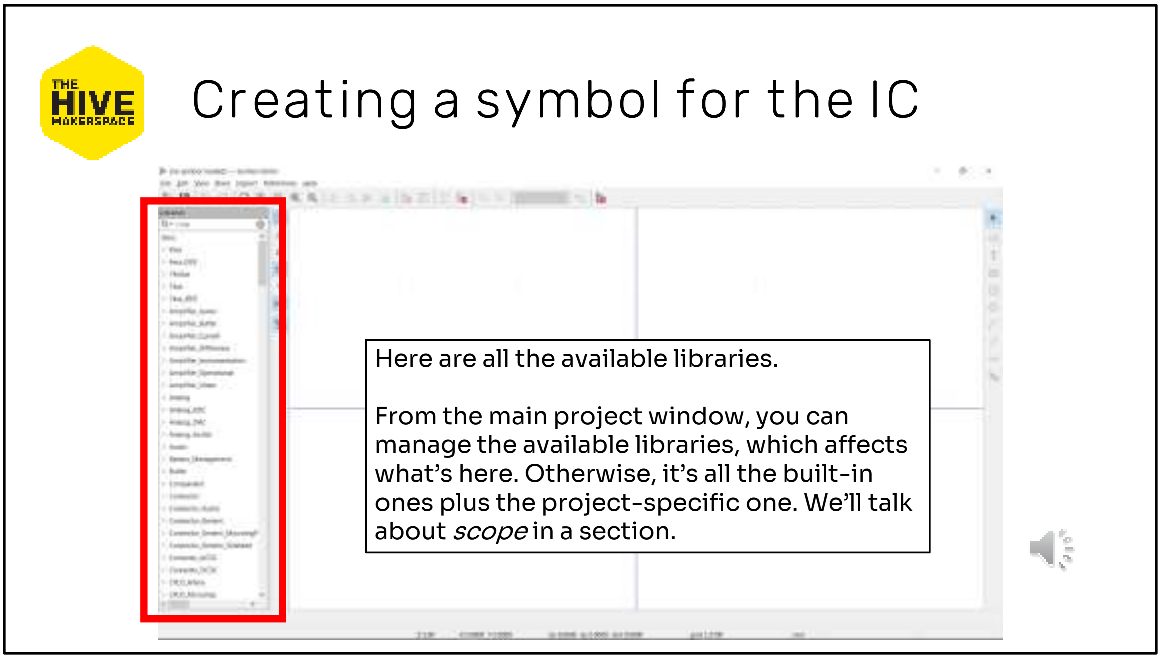


To make a symbol, we need to go into the Symbol Editor, which can be access from the project window or directly from the schematic using this icon.

Creating a symbol for the IC



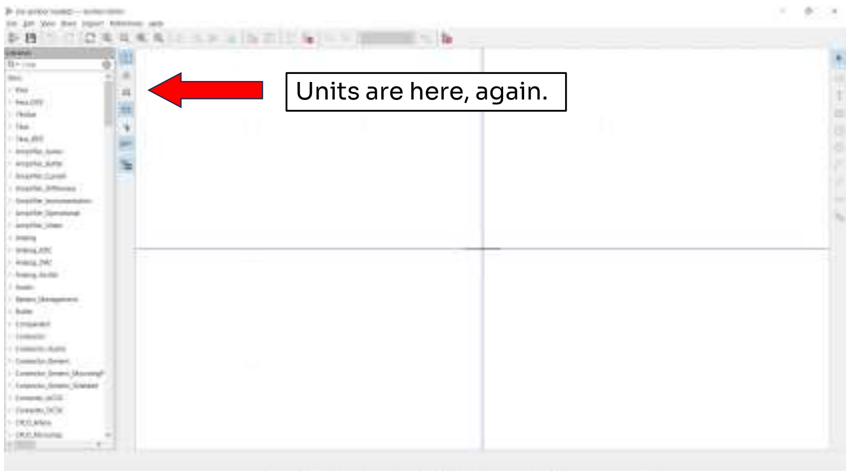
This is the blank symbol editor window.



On the left are all the available libraries. You can manage which are available through the project window under preferences.

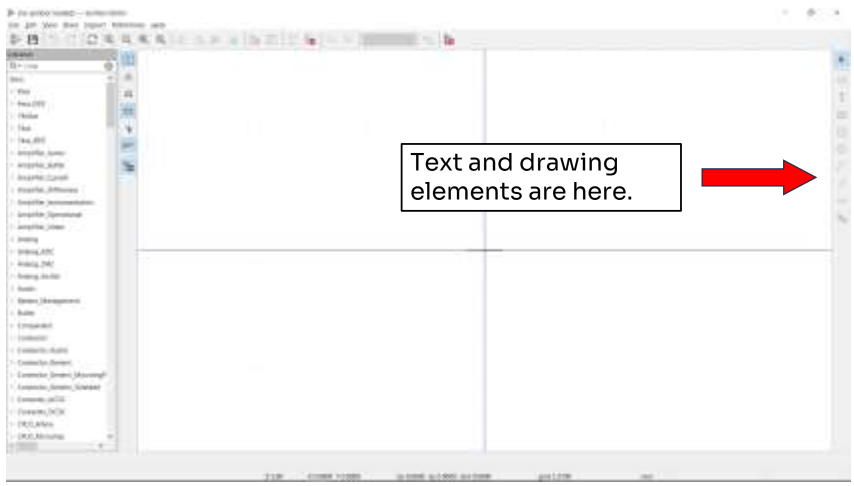
Note that all the built in libraries are read-only, so we'll need to make a new library for our new footprints.

Creating a symbol for the IC



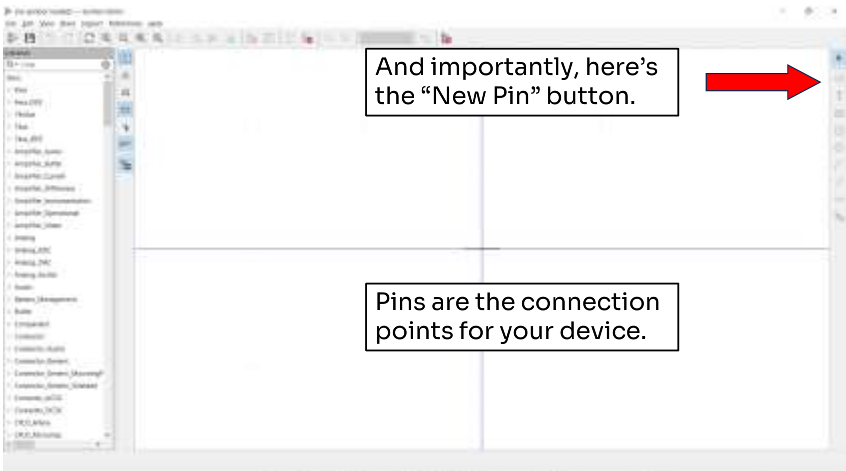
Units in the top left again.

Creating a symbol for the IC



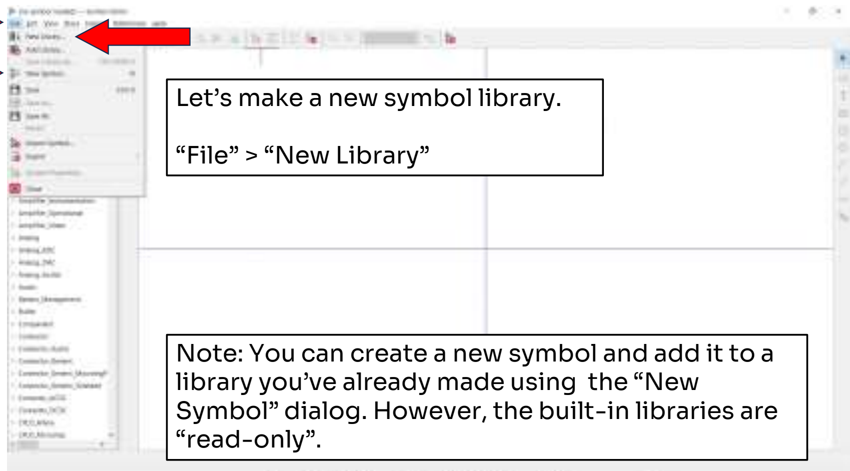
Text and drawing actions on the right.

Creating a symbol for the IC



Top right has the new pin button, which will be the most used button here, likely, since pins are the connection points.

Symbol Library – Creating Models



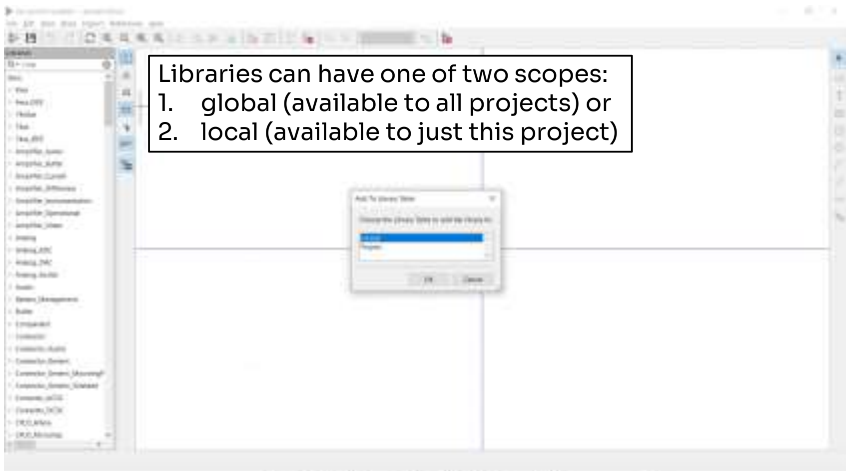
Let’s make a new symbol library.

“File” > “New Library”

Note: You can create a new symbol and add it to a library you’ve already made using the “New Symbol” dialog. However, the built-in libraries are “read-only”.

As mentioned, we need to make a new library to add our symbol into, since all the built-in libraries are read-only, and we don’t have any writeable libraries of our own.

Creating a symbol for the IC



New libraries can be set to have one of two scopes: they can be set to available for all projects, called global scope, or only be available for this project, called project or local. Project libraries can actually be made available in other projects by adding them in manually though the main project window, under preferences.

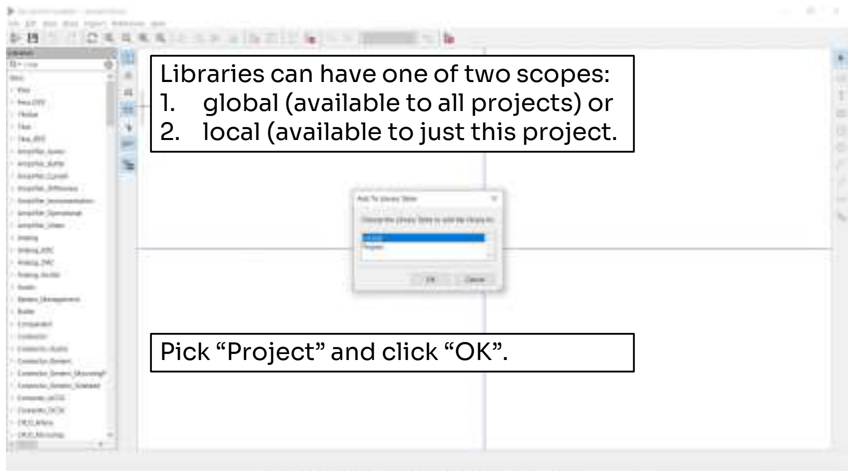
Aside: Library Management

- Library management is a whole topic, especially with larger teams with different backgrounds integrating automated backups or version control
- For KiCAD, it is strongly preferable to use local (i.e. project-specific) libraries for all projects.
  - Library objects can easily be copied into new libraries, making global libraries mostly a liability (e.g. external updates breaking your design)

Library management is an important but under-appreciated aspect of PCB design work. Or, at least, under-appreciated by beginning designers. There's a world of discussion out there about the best methods, especially when considering backups, version control, and cloud connectivity, but for KiCAD, it's nearly always better to use project-scope libraries rather than global ones. Why? Global libraries are not under your control, can change or fluctuate between revisions, and make transferring an entire project more difficult. Thus, it's strongly preferred to create a single project library and copy global parts into it.

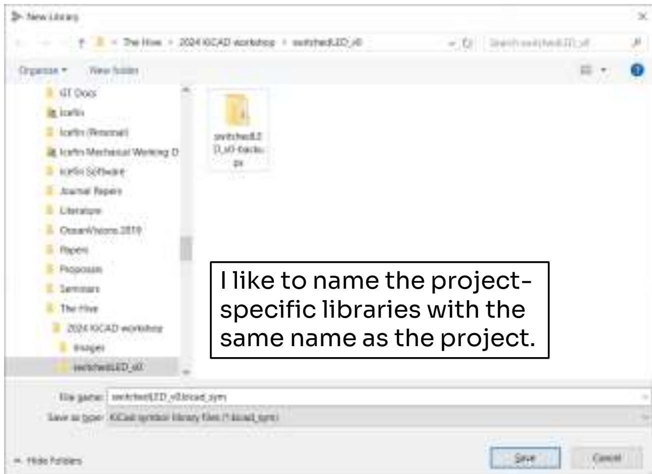
We've obviously not done that with this tutorial for a few reasons, primarily because it adds a lot of tutorial time and that it's a very simple design. If you'd like to understand how this process would work in a real design, we go through them from the beginning in parts 6 and 7.

Creating a symbol for the IC



Based on the principles of good KiCAD library management, make the new library a project-level library.

Creating a symbol for the IC



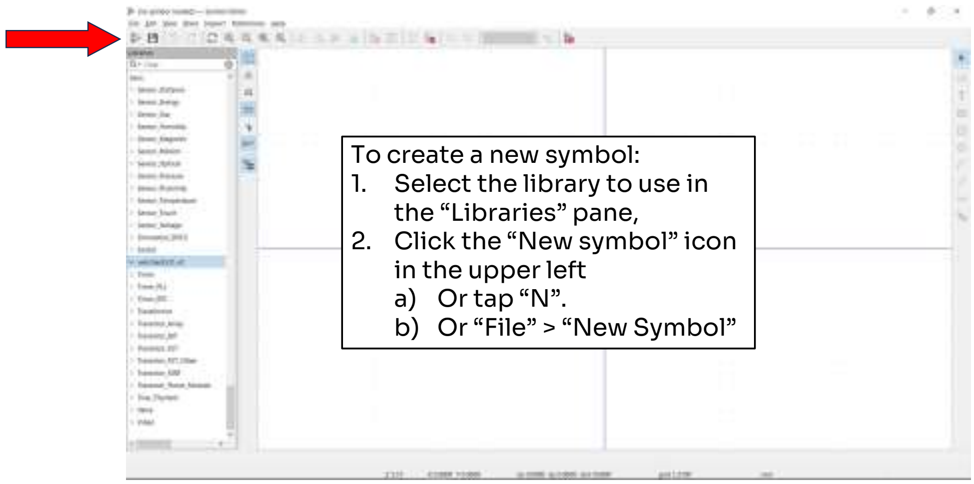
I like to name the project-specific libraries with the same name as the project.

Save the library. Typically, you would save it as the same name as the project itself, and in the project directory, so that everything related to the project is nicely packaged together.

Creating a symbol for the IC

We can see the new library on the left here.

Creating a symbol for the IC

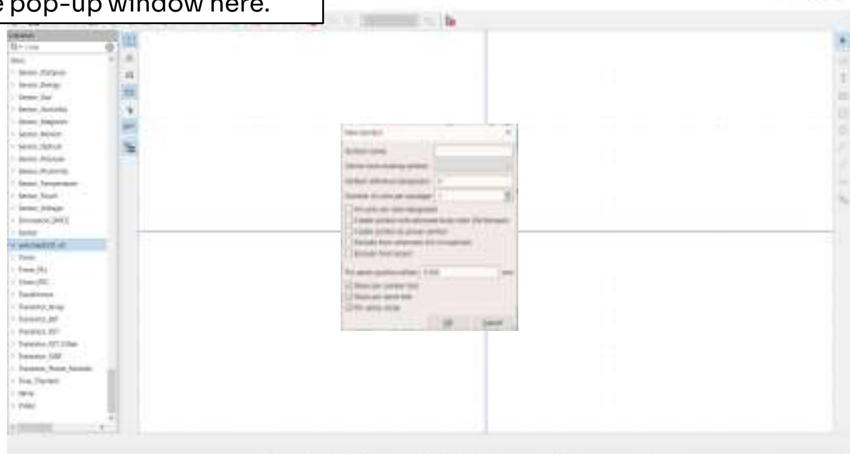


To create a new symbol:
1. Select the library to use in the “Libraries” pane,
2. Click the “New symbol” icon in the upper left
   a) Or tap “N”.
   b) Or “File” > “New Symbol”

Next, we can create a new symbol just hitting the “N” key, the icon in the upper-left, or under the “File” menu.

Creating a symbol for the IC

Fill in the pop-up window here.

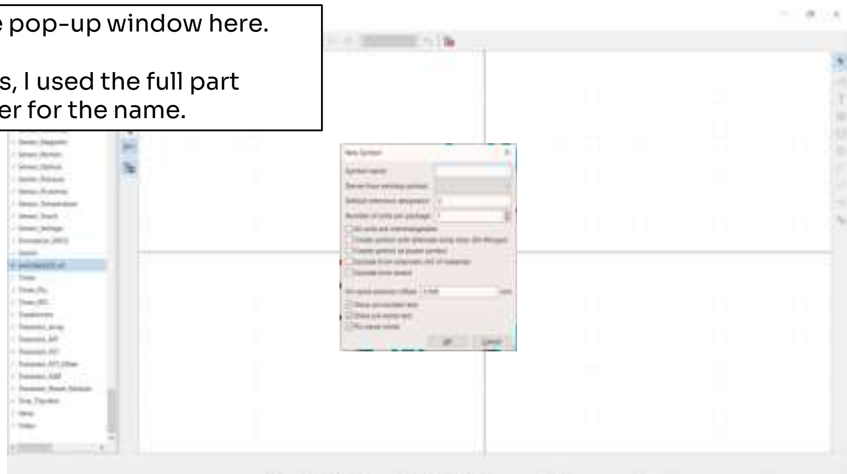


Let’s fill in the window here.

Creating a symbol for the IC

Fill in the pop-up window here.

- For ICs, I used the full part number for the name.

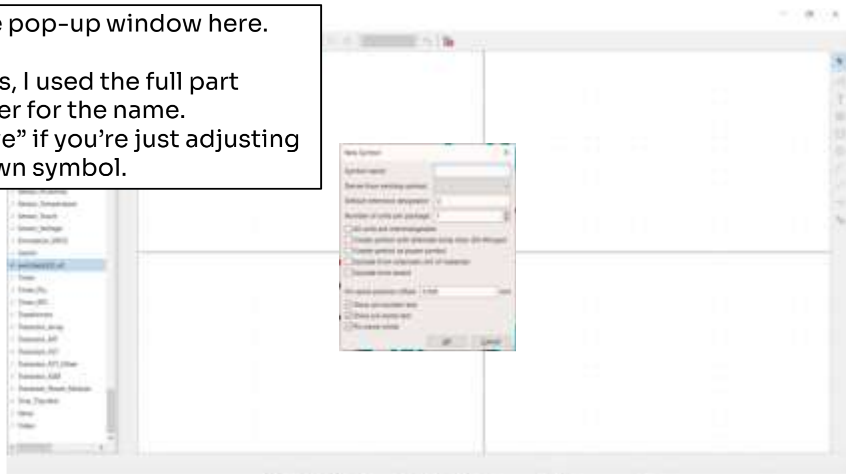


Generally, good policy is to name the symbol with the part number, which in this case would be RT4526. You can leave off the GJ6 in this case because, if you read the datasheet, those are the only options for the final three digits.

Creating a symbol for the IC

Fill in the pop-up window here.

- For ICs, I used the full part number for the name.
- "Derive" if you're just adjusting a known symbol.

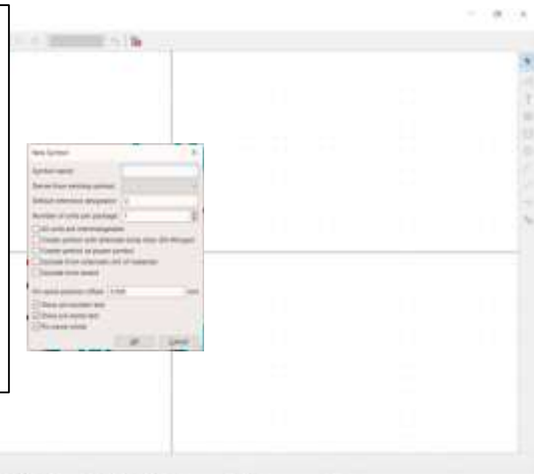


"Derive" refers to if you're adjusting a known symbol, like creating a polarized capacitor by copying the basic capacitor.

Creating a symbol for the IC

Fill in the pop-up window here.

- For ICs, I used the full part number for the name.
- “Derive” if you’re just adjusting a known symbol.
- Each instance of this symbol in a schematic will be identified by <Reference Designator><Number>, e.g. R5 for the fifth resistor. “U” is typical for ICs, but you can choose something else.



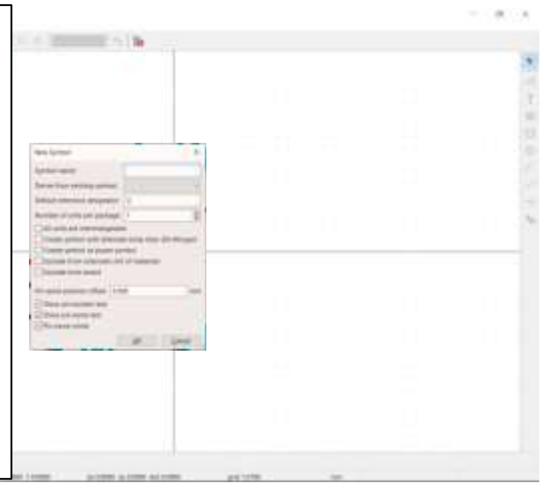
The reference designator is the letter that KiCAD uses to identify all instances of this symbol. Each instance will also get a number, so a resistor might be designated R5 for the fifth resistor of the schematic. “U” is very common for ICs, though you could choose something else if you’d like. It’s fine for multiple symbols to have the same designator character, like U or R; it just means they’re of the same type, so to speak.

Creating a symbol for the IC

Fill in the pop-up window here.

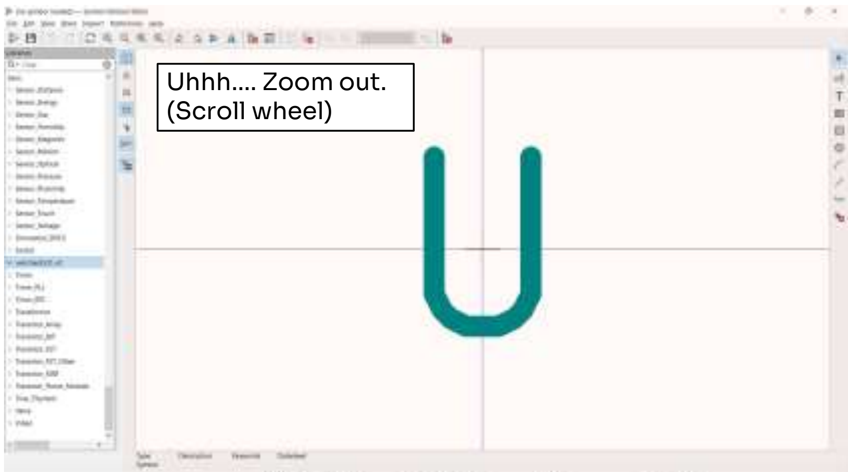
- For ICs, I used the full part number for the name.
- “Derive” if you’re just adjusting a known symbol.
- Each instance of this symbol in a schematic will be identified by <Reference Designator><Number>, e.g. R5 for the fifth resistor. “U” is typical for ICs, but you can choose something else.

The rest is not relevant now. Click OK to continue.



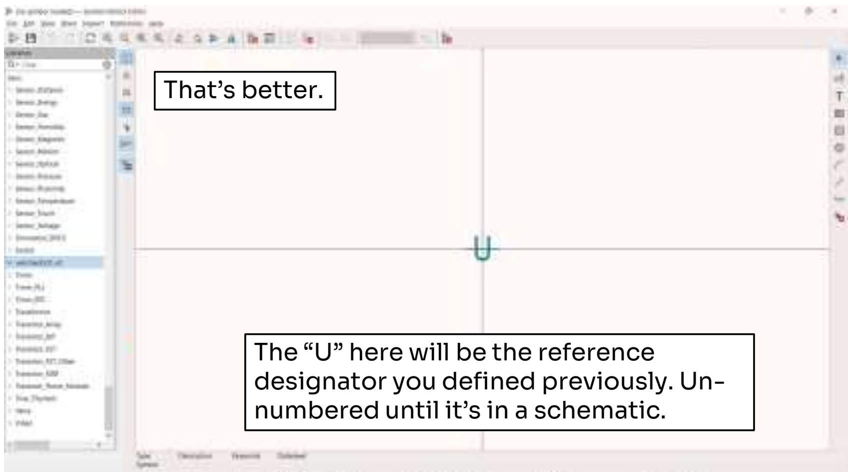
The rest doesn’t matter to us, so click “OK”.

Creating a symbol for the IC

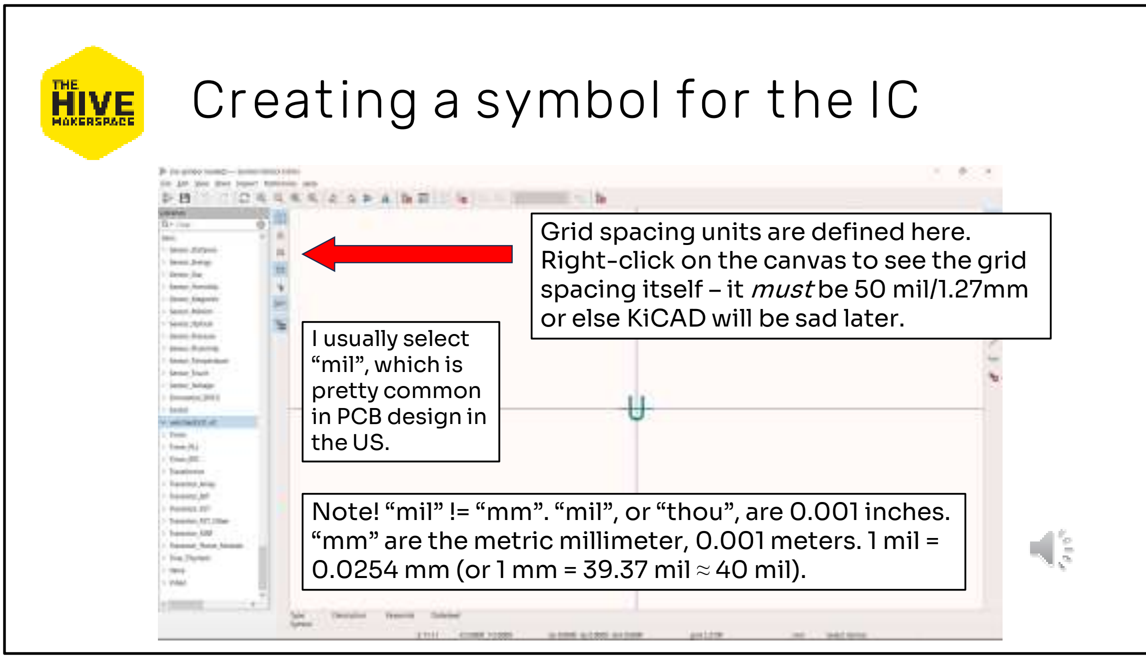


Since there's nothing on the editor now, it will zoom automatically in to the reference designator.

Creating a symbol for the IC



Something like this is better. The “U” is currently placed at the anchor point, which is the point at which the symbol will be attached to the mouse. We’ll move those both later.

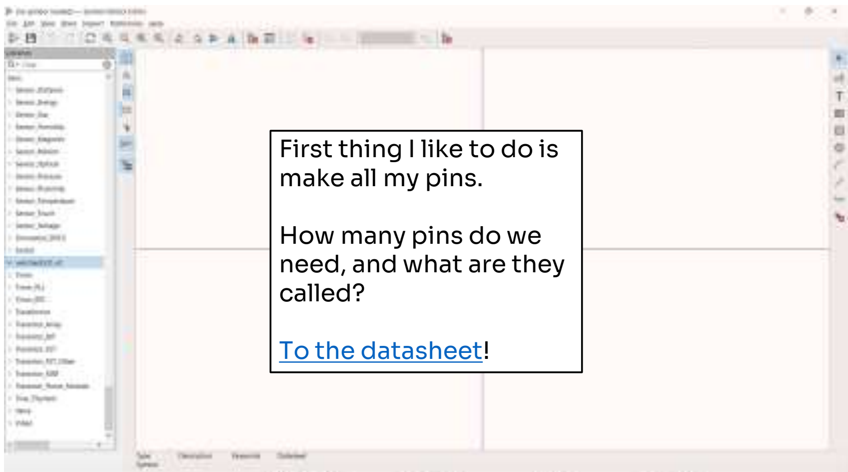


As usual, grid spacing and units are defined on the left. Check the grid by right-clicking. It should read 50 mil or 1.27 mm, or else KiCAD will not be able to attach wires to it and you’ll be sad later.

I usually select mils for my units because I’m used to thinking of hole sizes and trace widths in mils, and because I’m used to imperial units. But you can readily use metric units as well. KiCAD is based on metric values, after all.

Note that mils and millimeters are not the same. Mils, which are also known as thous, are a thousandth of an inch, or 0.001 inches. Millimeters are, of course, a thousandth of a meter. One millimeter is about forty mils.

Creating a symbol for the IC

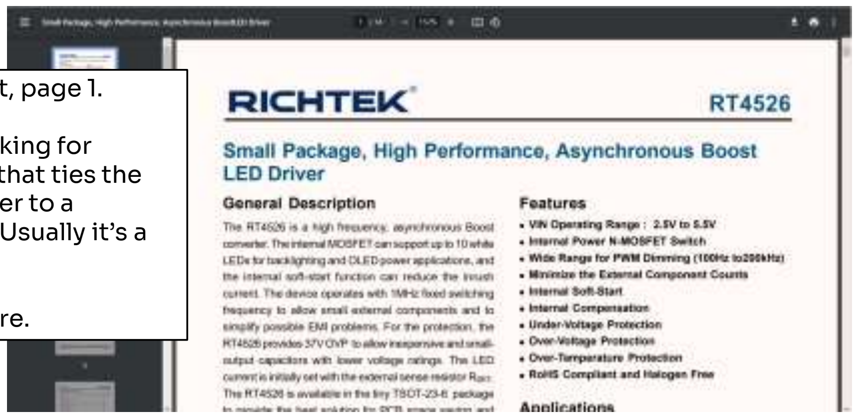


First thing I like to do is make all my pins.

How many pins do we need, and what are they called?

To the datasheet!

Creating a symbol for the IC



RICHTEK RT4526

Small Package, High Performance, Asynchronous Boost LED Driver

General Description

The RT4526 is a high frequency, asynchronous Boost converter. The internal MOSFET can support up to 10 white LEDs for backlighting and OLED power applications, and the internal soft-start function can reduce the inrush current. The device operates with 1MHz fixed switching frequency to allow small external components and to simplify possible EMI problems. For the protection, the RT4526 provides 37V OVP to allow inexpensive and small-output capacitors with lower voltage ratings. The LED current is initially set with the external sense resistor R_{SET}. The RT4526 is available in the tiny TSOT-23-6 package

Features

- VIN Operating Range : 2.5V to 5.5V
- Internal Power N-MOSFET Switch
- Wide Range for PWM Dimming (100Hz to200kHz)
- Minimize the External Component Counts
- Internal Soft-Start
- Internal Compensation
- Under-Voltage Protection
- Over-Voltage Protection
- Over-Temperature Protection
- RoHS Compliant and Halogen Free

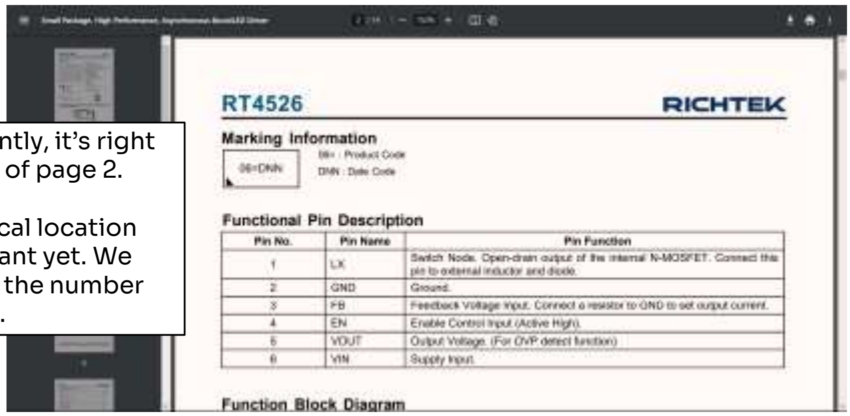
Applications

Datasheet, page 1.

We're looking for anything that ties the pin number to a function. Usually it's a table.

So not here.

Creating a symbol for the IC



RT4526 RICHTEK

Marking Information

0E=DNN

0E= : Product Code

DNN : Date Code

Functional Pin Description

Pin No.	Pin Name	Pin Function
1	LX	Switch Node. Open-drain output of the internal N-MOSFET. Connect this pin to external inductor and diode.
2	GND	Ground.
3	FB	Feedback Voltage Input. Connect a resistor to GND to set output current.
4	EN	Enable Control Input (Active High).
5	VOUT	Output Voltage. (For OVP detect function)
6	VIN	Supply Input.

Function Block Diagram

Conveniently, it's right at the top of page 2.

The physical location isn't relevant yet. We only want the number and name.

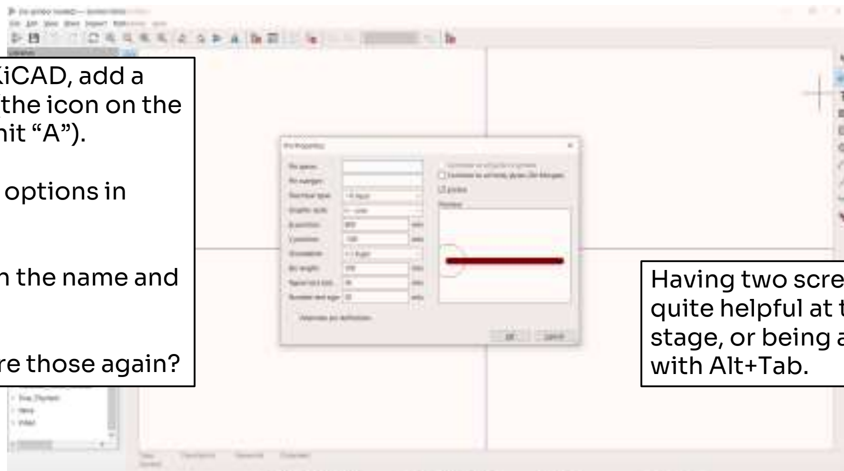
Creating a symbol for the IC

Back to KiCAD, add a new pin (the icon on the right, or hit “A”).

Bunch of options in here.

Start with the name and number.

What were those again?



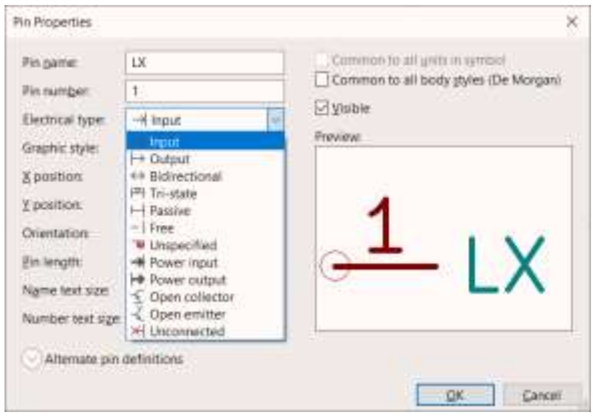
Having two screens is quite helpful at this stage, or being a wizard with Alt+Tab.

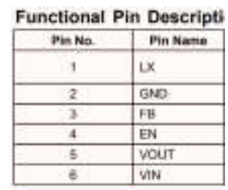
Creating a symbol for the IC

Start with pin 1.

By default, it's set as an "input". But is it?

Back to the pinout table!





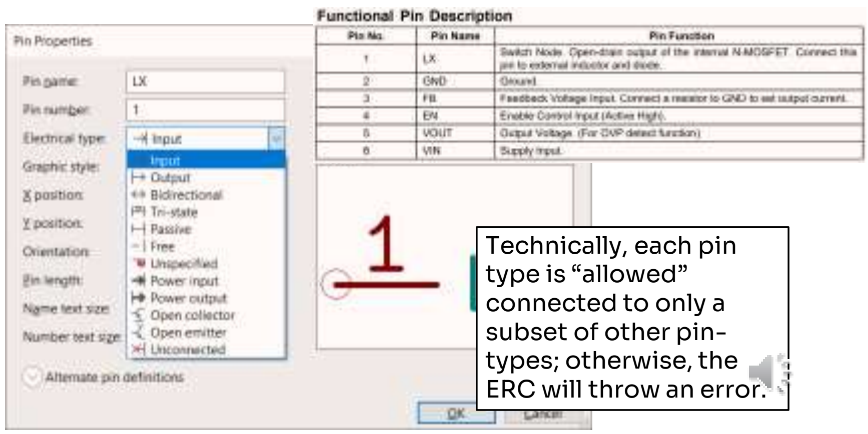
Functional Pin Descripti

Pin No.	Pin Name
1	LX
2	GND
3	FB
4	EN
5	VOUT
6	VIN

Creating a symbol for the IC

If you're not sure what it is, you can either set it to "Bidirectional" or "Unspecified" or "Passive".

The type won't cause the design to fail, but it might cause you headaches later with the ERC.

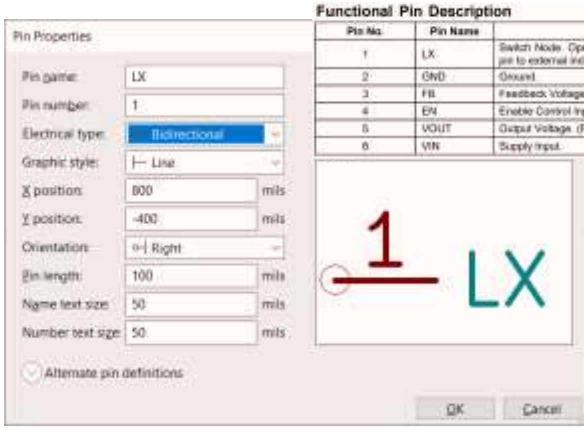


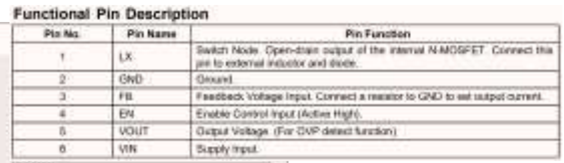
Technically, each pin type is "allowed" connected to only a subset of other pin-types; otherwise, the ERC will throw an error.

Creating a symbol for the IC

The rest of these are graphical choices, so we'll adjust them as needed later.

Click OK.

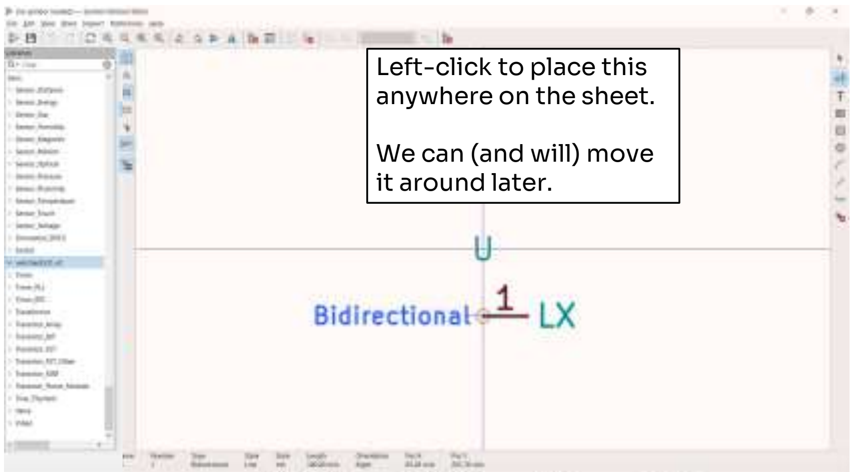




Functional Pin Description

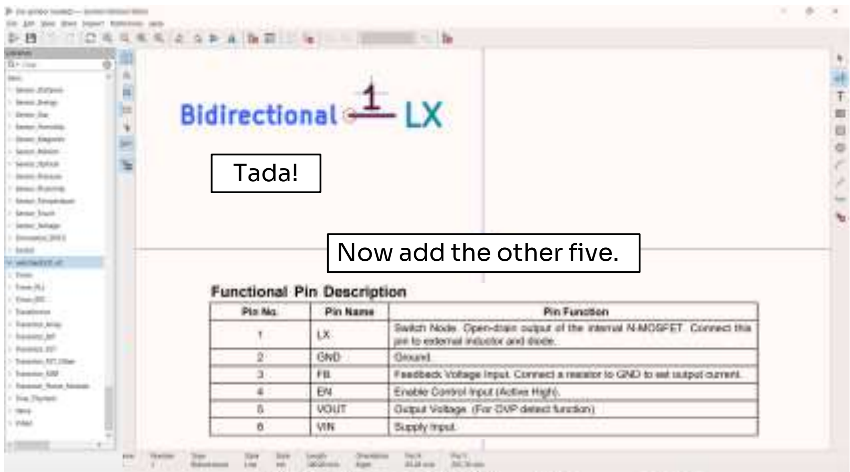
Pin No.	Pin Name	Pin Function
1	LX	Switch Node. Open-drain output of the internal N-MOSFET. Connect this pin to external inductor and diode.
2	GND	Ground.
3	FB	Feedback Voltage Input. Connect a resistor to GND to set output current.
4	EN	Enable Control Input (Active High).
5	VOUT	Output Voltage. (For OVP detect function)
6	VIN	Supply Input.

Creating a symbol for the IC



The pin type is in blue there.

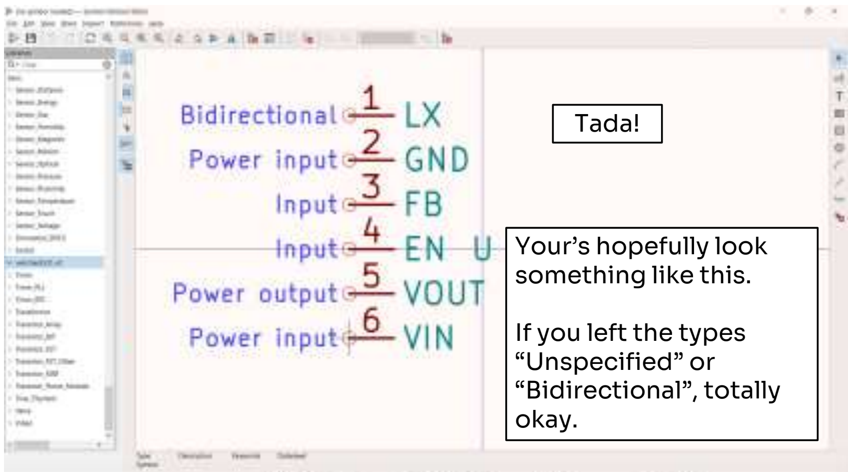
Creating a symbol for the IC



Functional Pin Description

Pin No.	Pin Name	Pin Function
1	LX	Switch Node. Open-drain output of the internal N-MOSFET. Connect this pin to external inductor and diode.
2	GND	Ground.
3	FB	Feedback Voltage Input. Connect a resistor to GND to set output current.
4	EN	Enable Control Input (Active High).
5	VOUT	Output Voltage. (For OVP detect function)
6	VIN	Supply Input.

Creating a symbol for the IC

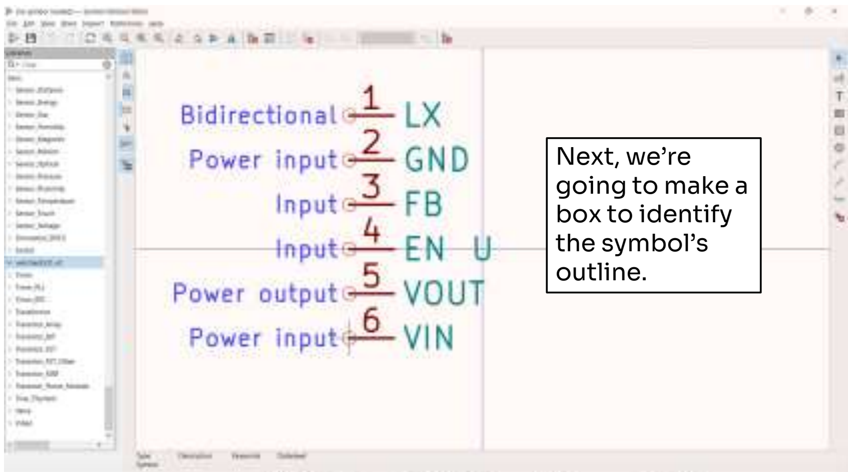


Tada!

Your’s hopefully look something like this.

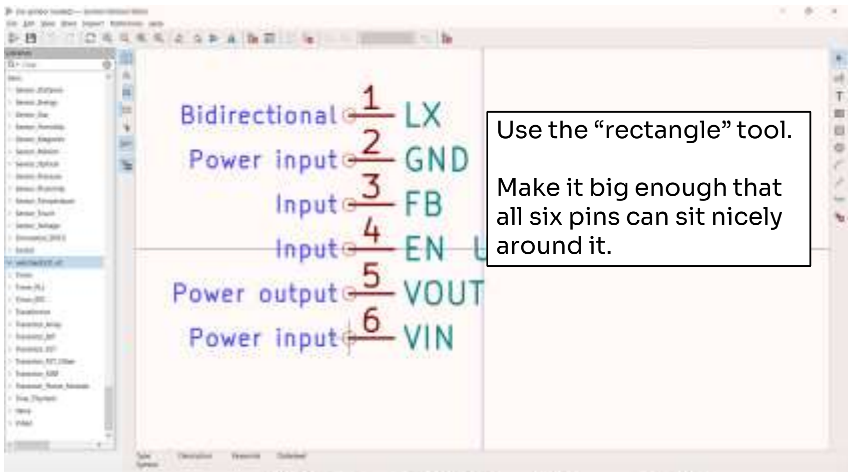
If you left the types “Unspecified” or “Bidirectional”, totally okay.

Creating a symbol for the IC



Next, we're going to make a box to identify the symbol's outline.

Creating a symbol for the IC



Use the "rectangle" tool.

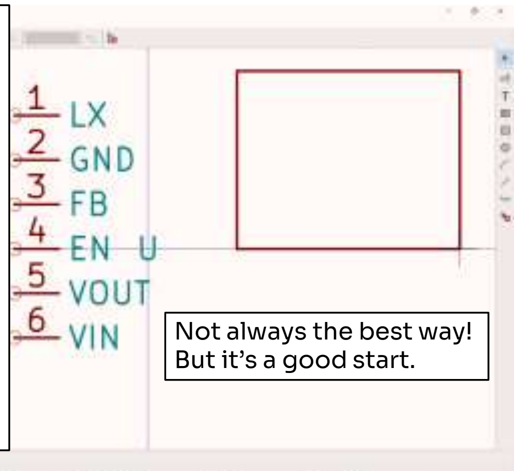
Make it big enough that all six pins can sit nicely around it.

Creating a symbol for the IC

Mine looks like this.

For height, I conceptually split the pins in half and make the box tall enough for half the pins spaced two grids apart plus two grids above and below. In this case, that's 3 pins x 2 grids per pin + 2 extra top grids + 2 extra bottom grids = 10 grids.

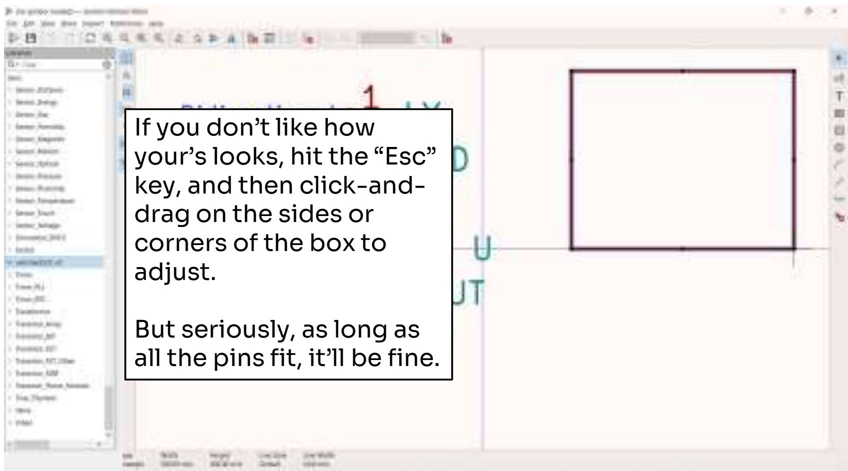
For width, I count the grides for the widest label, double it, and add two. "VOUT" is the widest at 4 grids, so that's 10 grids wide.



Not always the best way! But it's a good start.

This is the algorithm by which I spec'd my box's size, but I abandoned that size basically in the next few slides, so you're safe to ignore that.

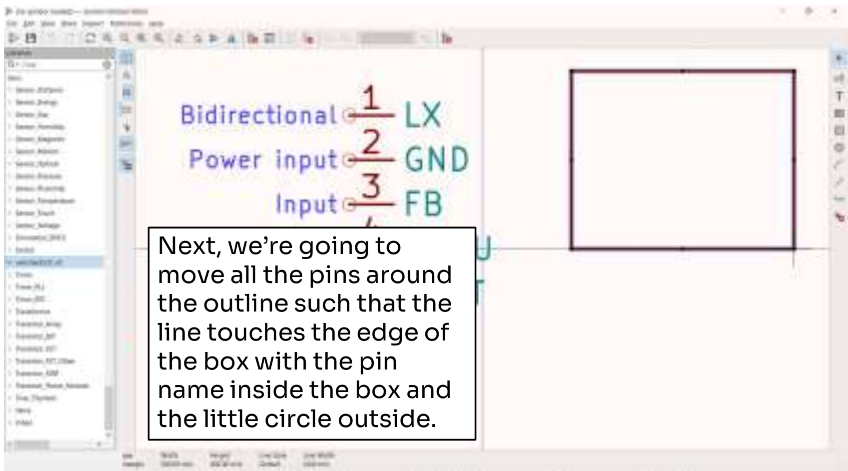
Creating a symbol for the IC



If you don't like how your's looks, hit the "Esc" key, and then click-and-drag on the sides or corners of the box to adjust.

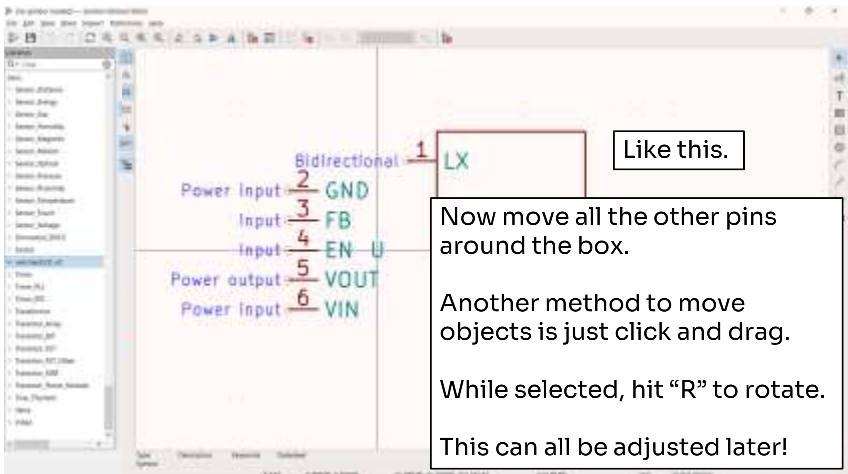
But seriously, as long as all the pins fit, it'll be fine.

Creating a symbol for the IC



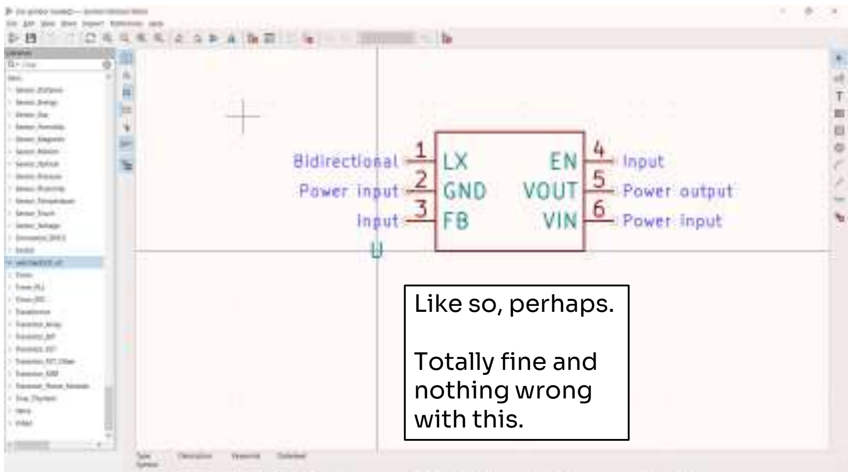
Next, we're going to move all the pins around the outline such that the line touches the edge of the box with the pin name inside the box and the little circle outside.

Creating a symbol for the IC



You might consider pausing the video here while you place your pins before seeing what I did.

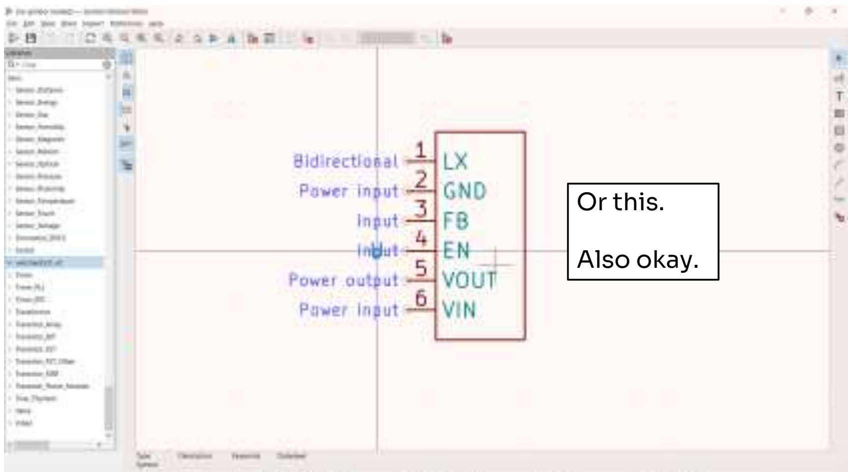
Creating a symbol for the IC



Like so, perhaps.

Totally fine and nothing wrong with this.

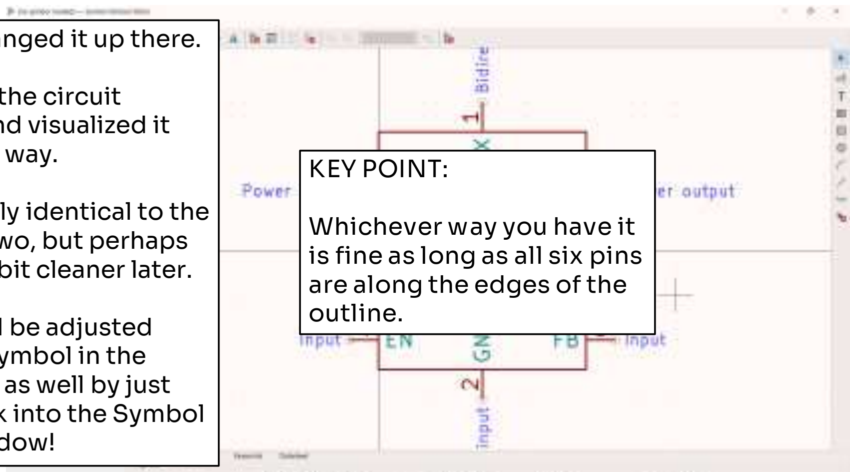
Creating a symbol for the IC



Or this.

Also okay.

Creating a symbol for the IC



Totally changed it up there.

Looked at the circuit drawing and visualized it better this way.

Functionally identical to the previous two, but perhaps will look a bit cleaner later.

This can all be adjusted once the symbol in the schematic as well by just going back into the Symbol Editor window!

KEY POINT:

Whichever way you have it is fine as long as all six pins are along the edges of the outline.

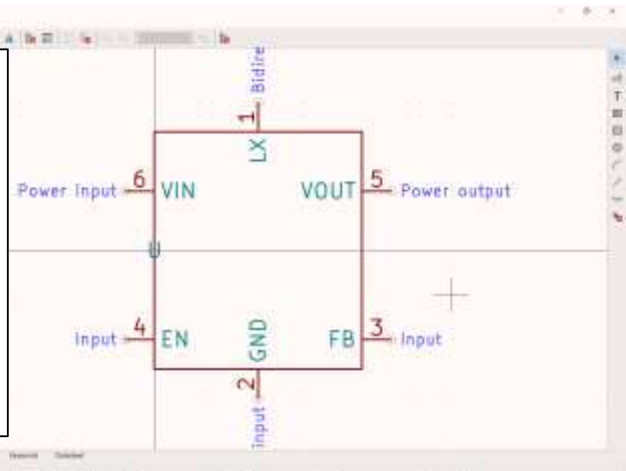
Additionally, when we learn about nets in the next video, you might understand why it doesn't matter how the symbol is laid out so much – pins can connect anywhere without a direct line.

Creating a symbol for the IC

Next, move the reference designator (the “U”) to somewhere that makes sense and won’t be obstructed.

Often, this is outside one of the four corners of the box, but it doesn’t have to be.

The reference number will later appear to the right of the “U”.



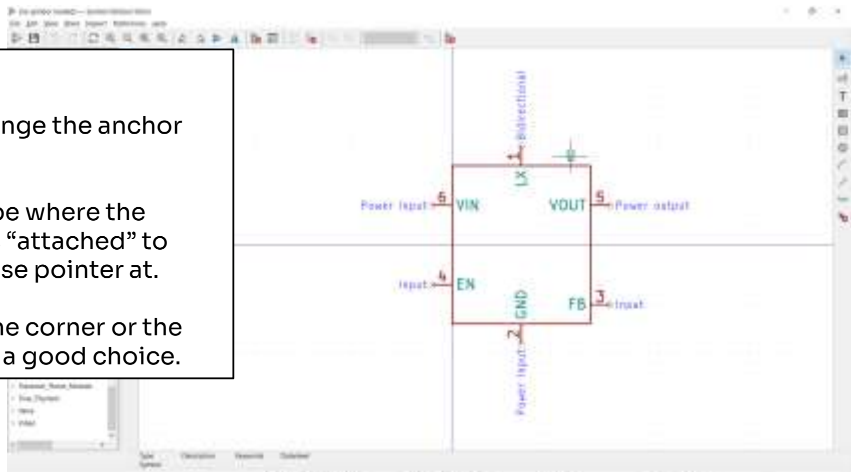
Creating a symbol for the IC

Great.

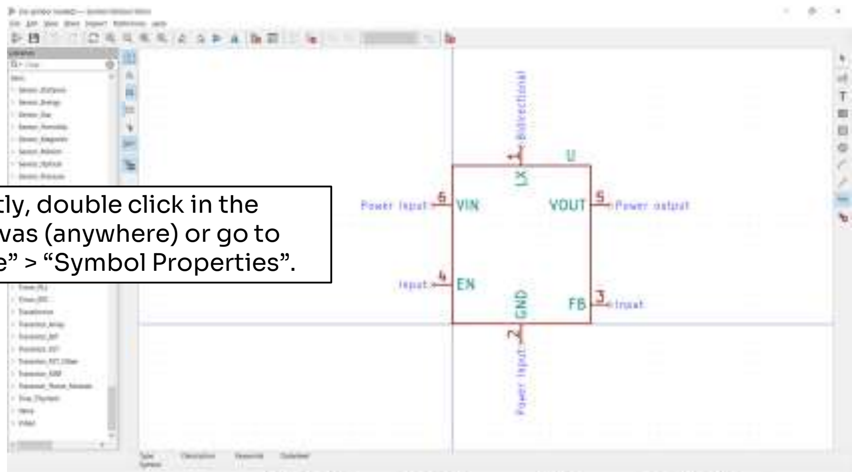
Next, change the anchor point.

This will be where the symbol is “attached” to your mouse pointer at.

Usually the corner or the middle is a good choice.



Creating a symbol for the IC



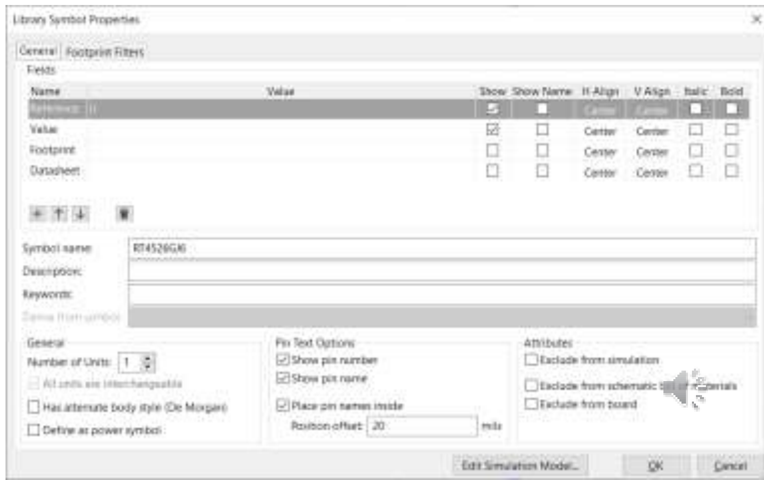
Lastly, double click in the canvas (anywhere) or go to "File" > "Symbol Properties".

Creating a symbol for the IC

These windows often have a lot more options than we need right now, but they do offer a lot of flexibility.

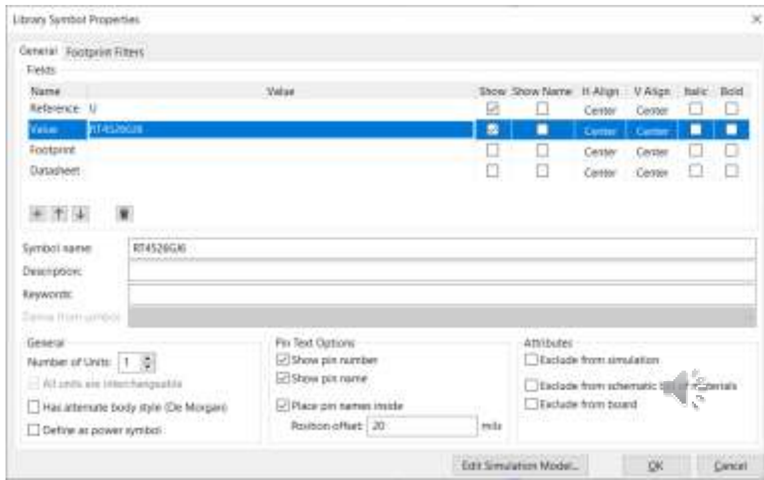
For now, just add the part number (RT4526GJ6) to the "Value" field's value.

Adding a footprint here can be done if there's a single footprint for this part. We don't have a footprint yet, so we won't add it, but later maybe.

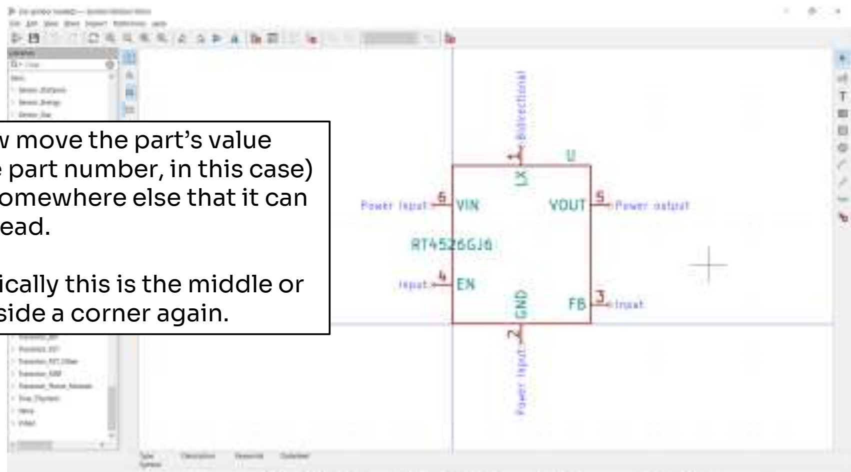


Creating a symbol for the IC

Click OK once your done.



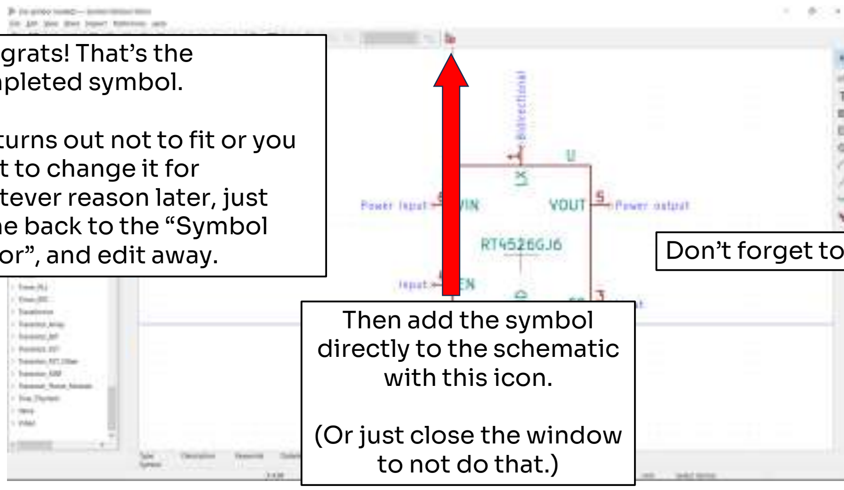
Creating a symbol for the IC



Now move the part’s value (the part number, in this case) to somewhere else that it can be read.

Typically this is the middle or outside a corner again.

Creating a symbol for the IC



Congrats! That's the completed symbol.

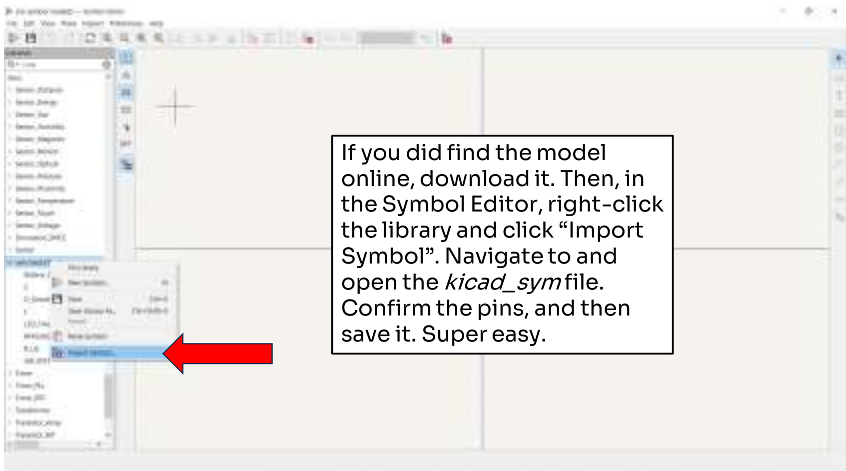
If it turns out not to fit or you want to change it for whatever reason later, just come back to the "Symbol Editor", and edit away.

Don't forget to save!

Then add the symbol directly to the schematic with this icon.

(Or just close the window to not do that.)

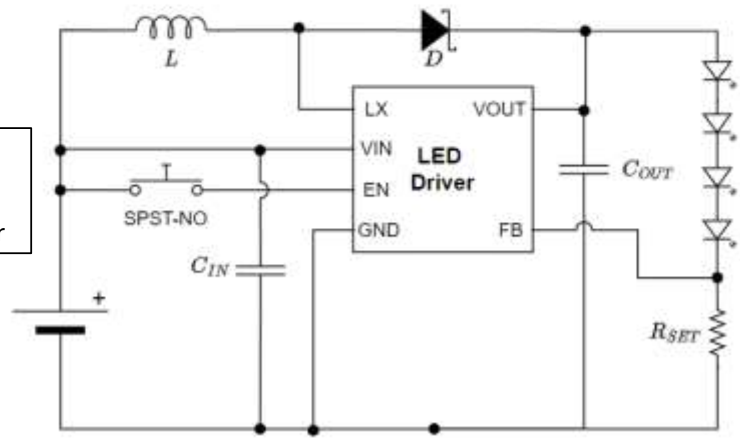
Aside: Importing a model



If you did find the model online, download it. Then, in the Symbol Editor, right-click the library and click “Import Symbol”. Navigate to and open the *kicad_sym* file. Confirm the pins, and then save it. Super easy.

Take a few minutes to arrange your schematic to something like this (minus all the lines).

Select a part:
"M" – Move
"R" – Rotate
"X"/"Y" – Mirror



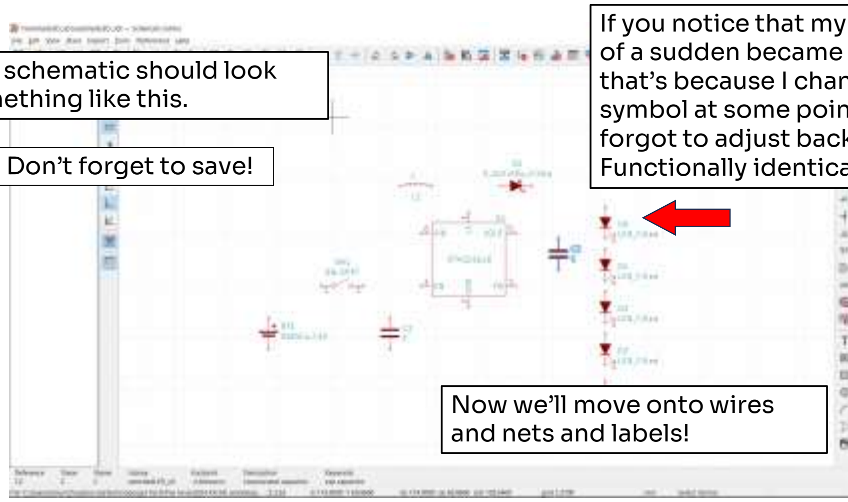
Again, pause the video here and take a few minutes to arrange your schematic. You can arrange it like this if you'd like to, but you don't have to – connections between components can be made without actual lines, as we'll cover in the next video.

Schematic with all the parts

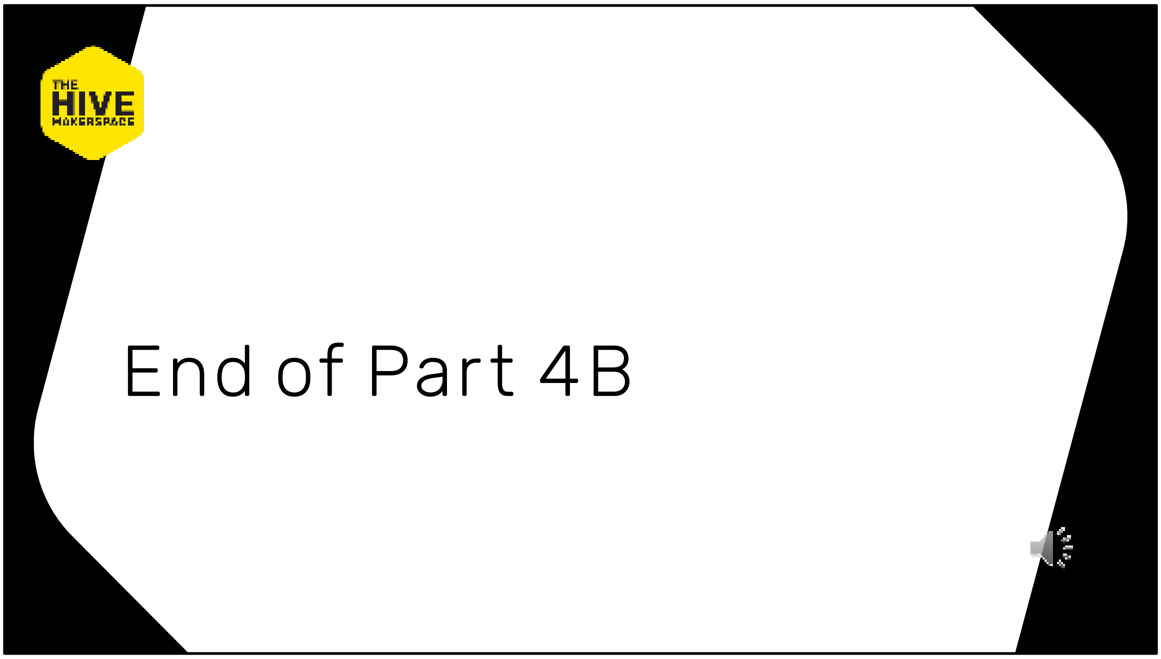
You schematic should look something like this.

Don't forget to save!

If you notice that my LEDs all of a sudden became filled, that's because I changed the symbol at some point and forgot to adjust back. Sorry. Functionally identical though!



Now we'll move onto wires and nets and labels!



And that ends part 4B of this video series in which we covered locating device models online, and making a symbol for our integrated circuit component. A PDF of this video is available as well, linked in the description and hosted on The Hive's Wiki.

In the next video of this series, part 4C, I'll teach you how to connect components together.

See you then.