EAGLECAD TUTORIAL

ECE 445
University of Illinois at Urbana Champaign
Department of Electrical and Computer Engineering

Instructions

Read through the tutorial and learn what EagleCAD is used for. Then, construct the attached circuit into both a schematic and board file. This Eagle assignment should be sent to your TA in the form of a .sch and .brd file.

Overview

Schematics are the core of all electrical engineering. They represent a universal list of symbols and notations that can be followed by anyone in order to present some sort of circuitry. In fact, one can even think of some as a sort of language that is used to communicate our electrical ideas. Just like a language, we can present an idea in a clear and specific manner, or we can obfuscate it through a dense and complex layout of thoughts. The goal of this tutorial is to help you learn tips and methods to be in the former camp.

While schematics are nice from an analytical standpoint, real circuits don't often follow the same shapes or layout. As you have most likely seen in previous classes, even trying to prototype on a breadboard often has one moving parts around in order to get the proper connections. Op-amps, for example, aren't sold in triangular packages, they often come in square or rectangular ones, with the pins alternating sides. Resistors aren't zig-zagging lines, they usually come as little brown peanut shapes on a metal wire. In this class, you are going to learn how to design and solder to a **printed circuit board**, or PCB.

EagleCAD is one of many available Electronic Design Automation (EDA) software packages. All EDA software allows us to easily design and transition between schematics and circuit layout. EagleCAD has a large amount of popularity with hobbyists, and due to the nature of the projects in this course, you will most likely find that to be of use. You can download the freeware version of EagleCAD here: http://www.cadsoftusa.com/download-eagle/. The lab computers have the full professional versions installed. **WARNING: DIFFERENT EAGLE VERSIONS MAY NOT BE COMPATIBLE.** This applies both forward and backward. We advise sticking to one version for the entire semester.

Schematic Capture

Figure 1 shows the schematic we're designing. We have some DC power in, here 10V, going through a linear regulator to provide a lower voltage. Then we have a simple LED with a current limiting resistor. This is a fairly high level schematic, as the TPS799 is just presented a block with no actually surrounding circuitry.

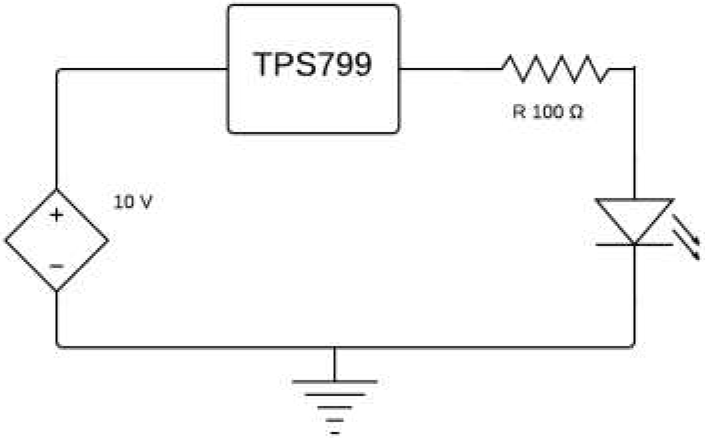
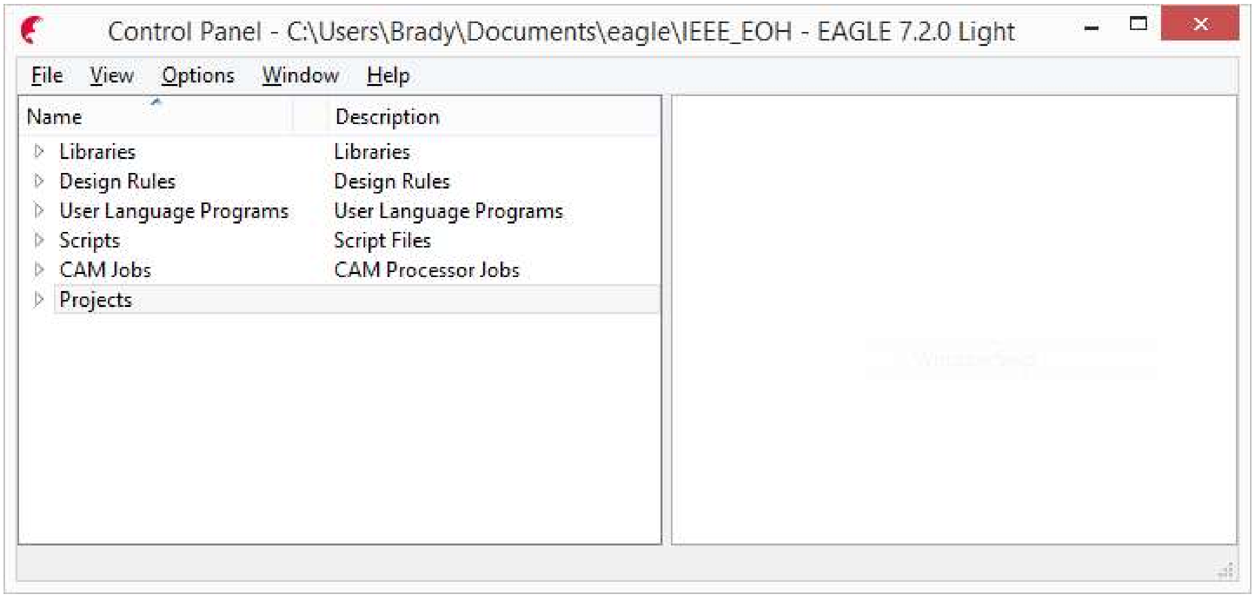


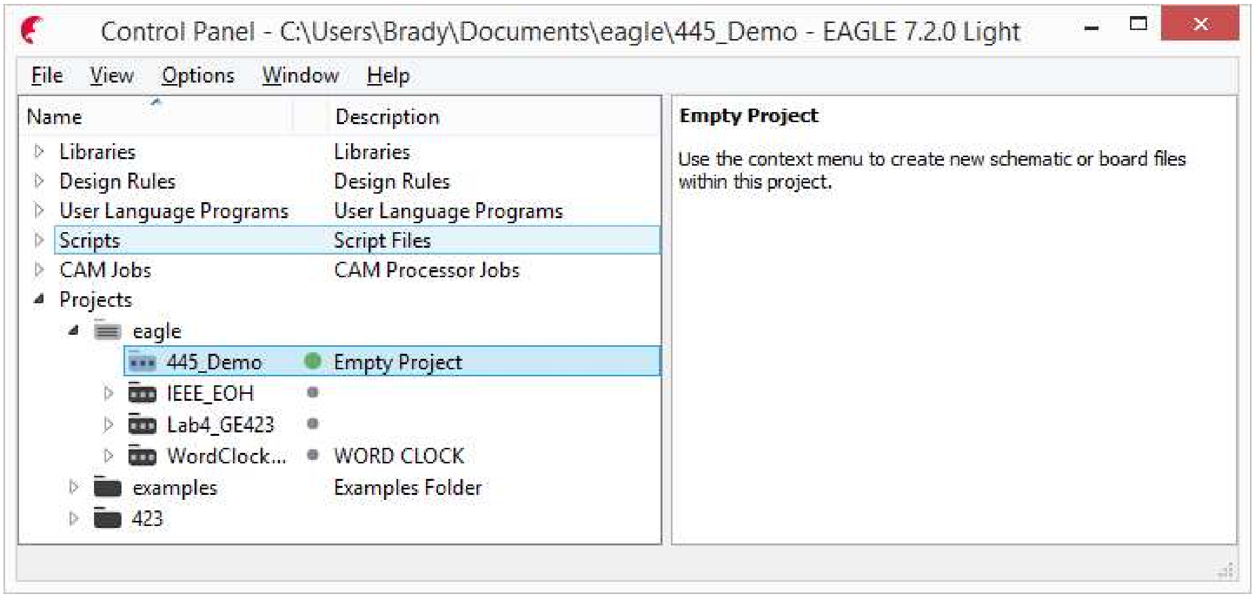
Figure 1. LED driver circuit

Let's begin - open EagleCAD via whatever OS you are currently on. This should open the Project Viewer screen, shown below:


Control Panel - C:\Users\Brady\Documents\eagle\IEEE_EOH - EAGLE 7.2.0 Light
File
View
Options
Window
Help

Name	Description
Libraries	Libraries
Design Rules	Design Rules
User Language Programs	User Language Programs
Scripts	Script Files
CAM Jobs	CAM Processor Jobs
Projects	

From here, create a new project by going to **File -> New -> Project**. Name it whatever you like, I've chosen "445_Demo". Make sure the green dot is visible next to the project – this indicates it is the active project.

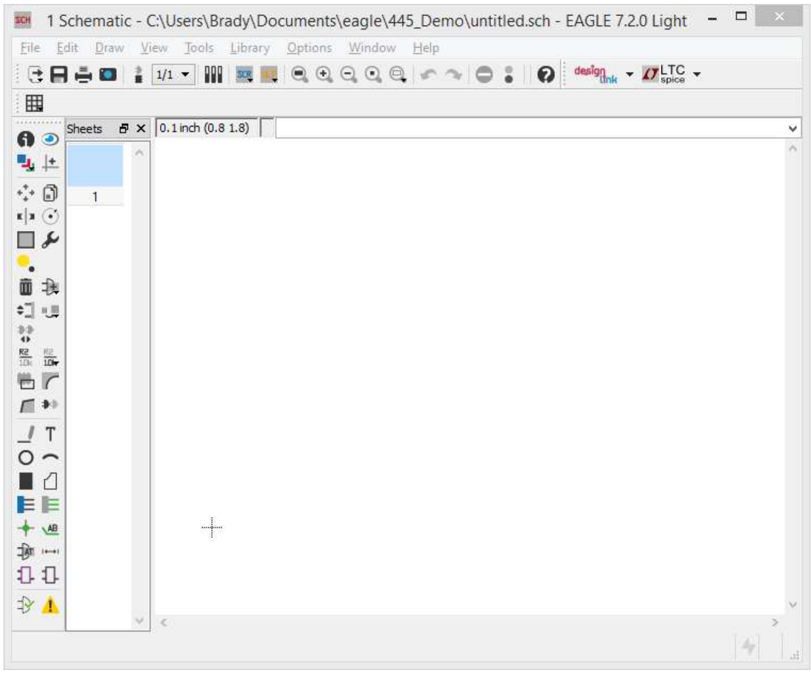

Control Panel - C:\Users\Brady\Documents\eagle\445_Demo - EAGLE 7.2.0 Light
File
View
Options
Window
Help

Name		Description
Libraries		Libraries
Design Rules		Design Rules
User Language Programs		User Language Programs
Scripts		Script Files
CAM Jobs		CAM Processor Jobs
Projects		
eagle		
445_Demo	●	Empty Project
IEEE_EOH	●	
Lab4_GE423	●	
WordClock...	●	WORD CLOCK
examples		Examples Folder
423		

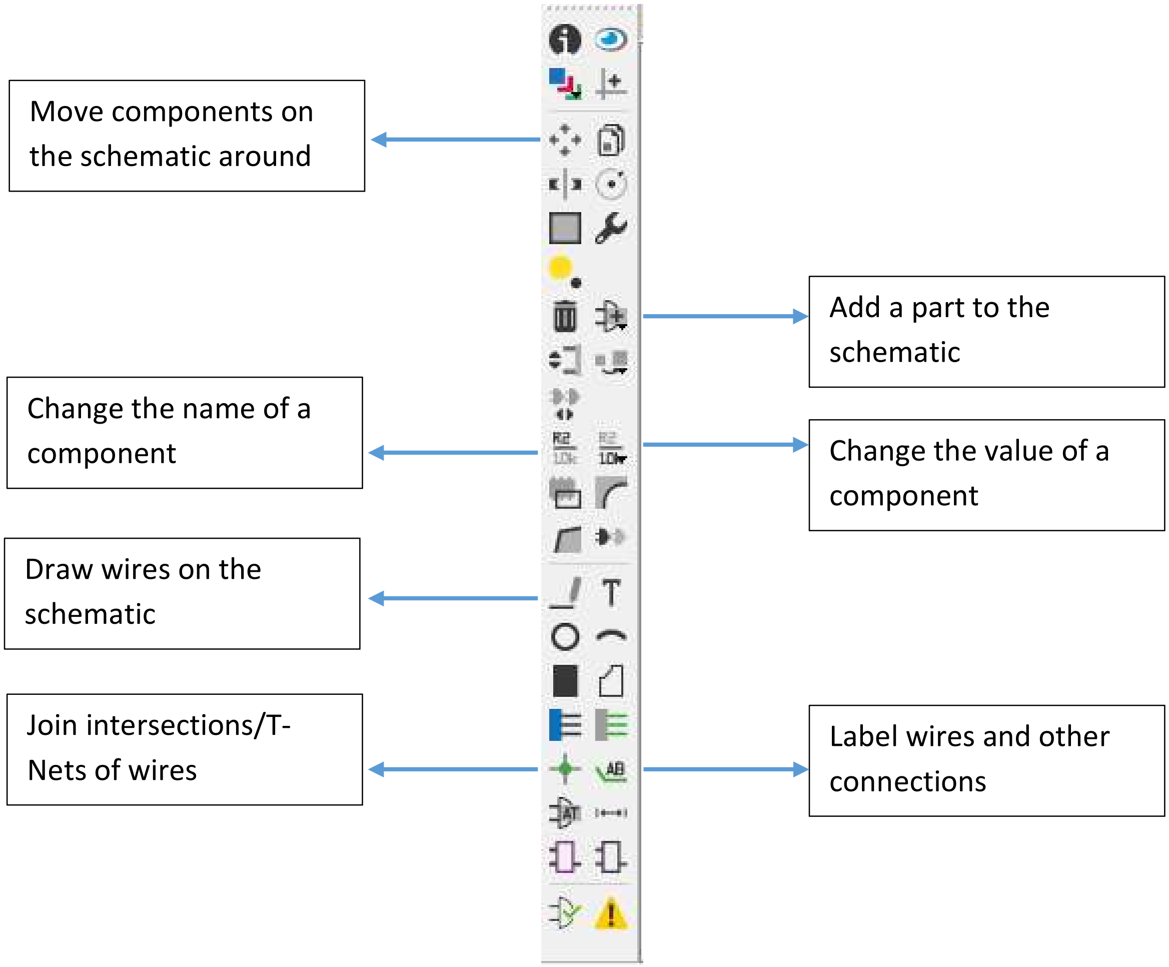
Empty Project

Use the context menu to create new schematic or board files within this project.

Right click on the project, and select **New -> Schematic**. The following window should pop up, which represents our Schematic Capture window.



While all the buttons available on the left have some use case, we'll really only be using a few of them for 90% of the work.



Move components on the schematic around

Add a part to the schematic

Change the name of a component

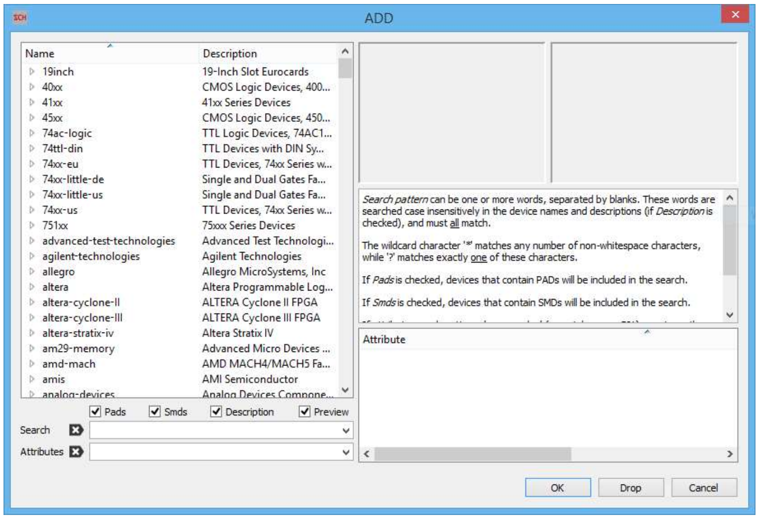
Change the value of a component

Draw wires on the schematic

Join intersections/T-Nets of wires

Label wires and other connections

Let’s get started with the simple resistor and LED half of the circuit. First we’ll add the resistor and capacitor. Press the add part window, and you’ll see a list of libraries appear.



Name	Description
19inch	19-Inch Slot Eurocards
40xx	CMOS Logic Devices, 400...
41xx	41xx Series Devices
45xx	CMOS Logic Devices, 450...
74ac-logic	TTL Logic Devices, 74AC1...
74ttl-din	TTL Devices with DIN Sy...
74xx-eu	TTL Devices, 74xx Series w...
74xx-little-de	Single and Dual Gates Fa...
74xx-little-us	Single and Dual Gates Fa...
74xx-us	TTL Devices, 74xx Series w...
751xx	75xxx Series Devices
advanced-test-technologies	Advanced Test Technologi...
agilent-technologies	Agilent Technologies
allegro	Allegro MicroSystems, Inc
altera	Altera Programmable Log...
altera-cyclone-II	ALTERA Cyclone II FPGA
altera-cyclone-III	ALTERA Cyclone III FPGA
altera-stratix-iv	Altera Stratix IV
am29-memory	Advanced Micro Devices ...
amd-mach	AMD MACH4/MACH5 Fa...
amis	AMI Semiconductor
analog-devices	Analog Devices Compone...

☑ Pads ☑ Smds ☑ Description ☑ Preview

Search

Attributes

Search pattern can be one or more words, separated by blanks. These words are searched case insensitively in the device names and descriptions (if *Description* is checked), and must all match.

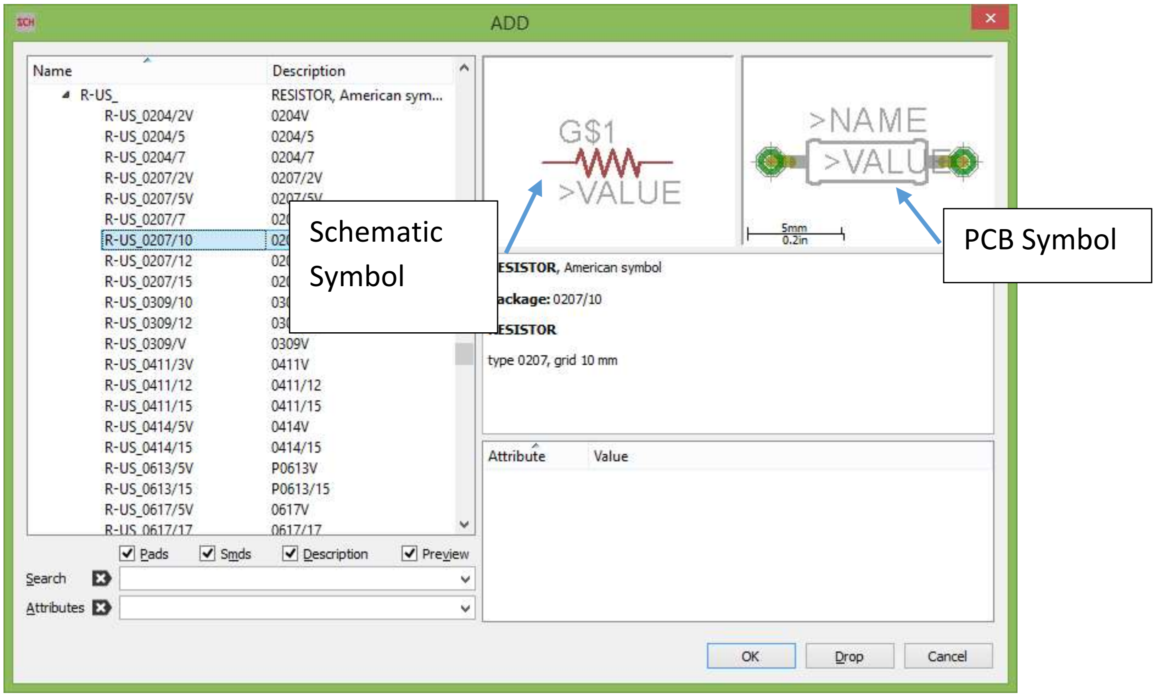
The wildcard character '*' matches any number of non-whitespace characters, while '?' matches exactly one of these characters.

If *Pads* is checked, devices that contain PADs will be included in the search.

If *Smds* is checked, devices that contain SMDs will be included in the search.

These libraries are supplied from a variety of sources: CadSoft (the creators of EagleCAD), numerous semiconductor companies, and even just regular hobbyists. Since the files are very simple to generate for new components (just simple SVG drawings), you can find a library for nearly any part. The downside is that the organization leaves a bit to be desired, but luckily the search feature is simple enough to use. For now, we'll start off using standard libraries.

Scroll down to the library called **rcl** – this is your standard passives library. It contains all common and uncommon for the basic resistor, capacitor, and inductor parts. Opening it up, it'll have several sub folders. Since this is America, open up **R-US**. Everything contained in that sub list is a resistor, but with a different package.



Name	Description
R-US_	RESISTOR, American sym...
R-US_0204/2V	0204V
R-US_0204/5	0204/5
R-US_0204/7	0204/7
R-US_0207/2V	0207/2V
R-US_0207/5V	0207/5V
R-US_0207/7	020
R-US_0207/10	020
R-US_0207/12	020
R-US_0207/15	020
R-US_0309/10	030
R-US_0309/12	030
R-US_0309/V	0309V
R-US_0411/3V	0411V
R-US_0411/12	0411/12
R-US_0411/15	0411/15
R-US_0414/5V	0414V
R-US_0414/15	0414/15
R-US_0613/5V	P0613V
R-US_0613/15	P0613/15
R-US_0617/5V	0617V
R-US_0617/17	0617/17

☑ Pads ☑ Smds ☑ Description ☑ Preview

Search

Attributes

ESISTOR, American symbol

ackage: 0207/10

ESISTOR

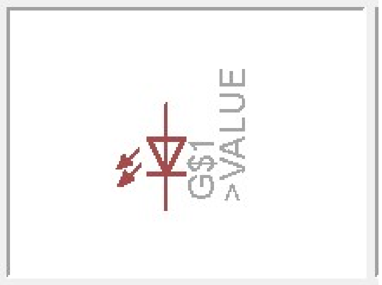
type 0207, grid 10 mm

Attribute	Value

Schematic Symbol

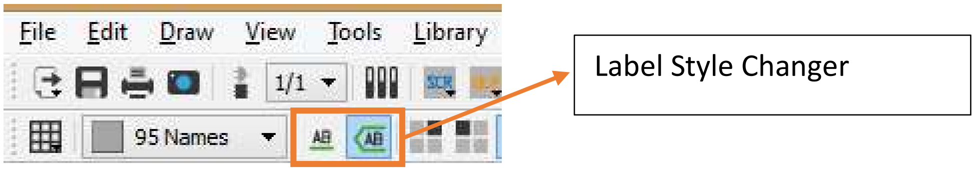
PCB Symbol

Choose any through-hole resistor you want. You can tell a resistor is "through-hole" by looking at the PCB symbol. If it has green circles, it's a through-hole part as those indicate vias in EagleCAD. If it has red squares instead (bottom half of list), those represent pads, so it is a surface mount part. Double-click to add it to the schematic and drop it anywhere. Do the same for the LED. I recommend looking under **led -> LED**, but feel free to use any singular two-terminal LED you find. If you're unsure, make sure the schematic symbol looks like:

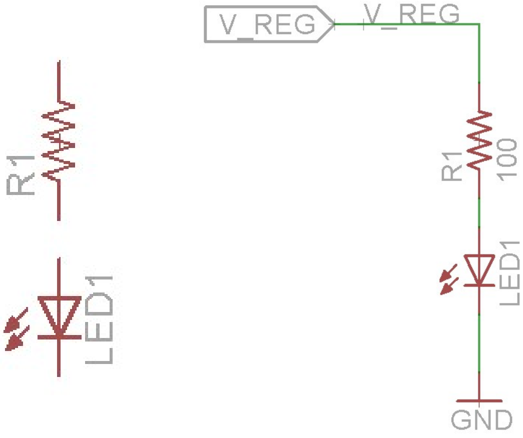


Your schematic should now look similar to the figure on the left below. If it doesn't quite line up, grab the Move Tool and use it. Right click when using it to rotate components. Now that we have a few basic components down, let's start wiring some things up. Use the Line Tool to draw a connection between the components. Also, go back and add a ground to the schematic. You can find the ground symbol under the **supply1** or **supply2** libraries. If you look, you'll see that ground doesn't have a board view – which makes sense, ground is an electrical reference voltage, not a tangible, physical *thing*.

Let's also define a value for the resistor while we're at it. Use the Value Tool and select the resistor. Give it a reasonable value, somewhere between 100 to 300 Ohms. Next, we'll add a connection to the topside of the resistor. Draw a wire coming out, but leave it unconnected. Use the Name Tool to give it an appropriate name – V_REG, V_OUT, REG_OUT, etc. You'll notice that doesn't really do anything by default, we need to also show the label as well. Use the Label Tool and click on the connection, then place the label somewhere nearby. You can change the style of your label via the buttons on the top bar. Your schematic should know look similar to the picture on the right below (I've shown both labeling styles).

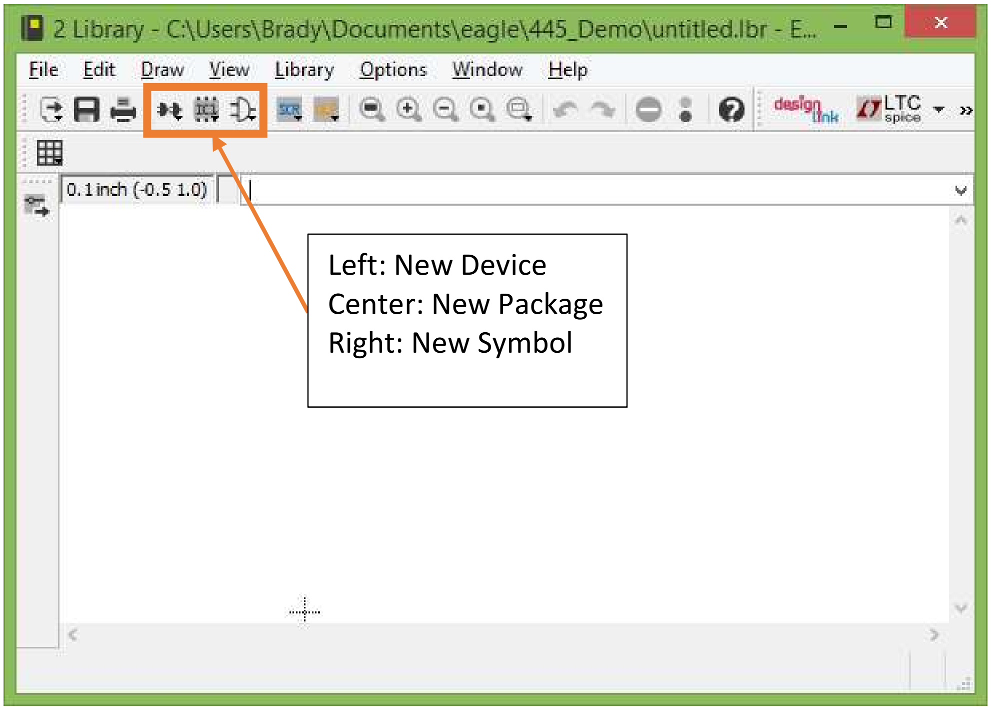


Label Style Changer



Creating a New Part

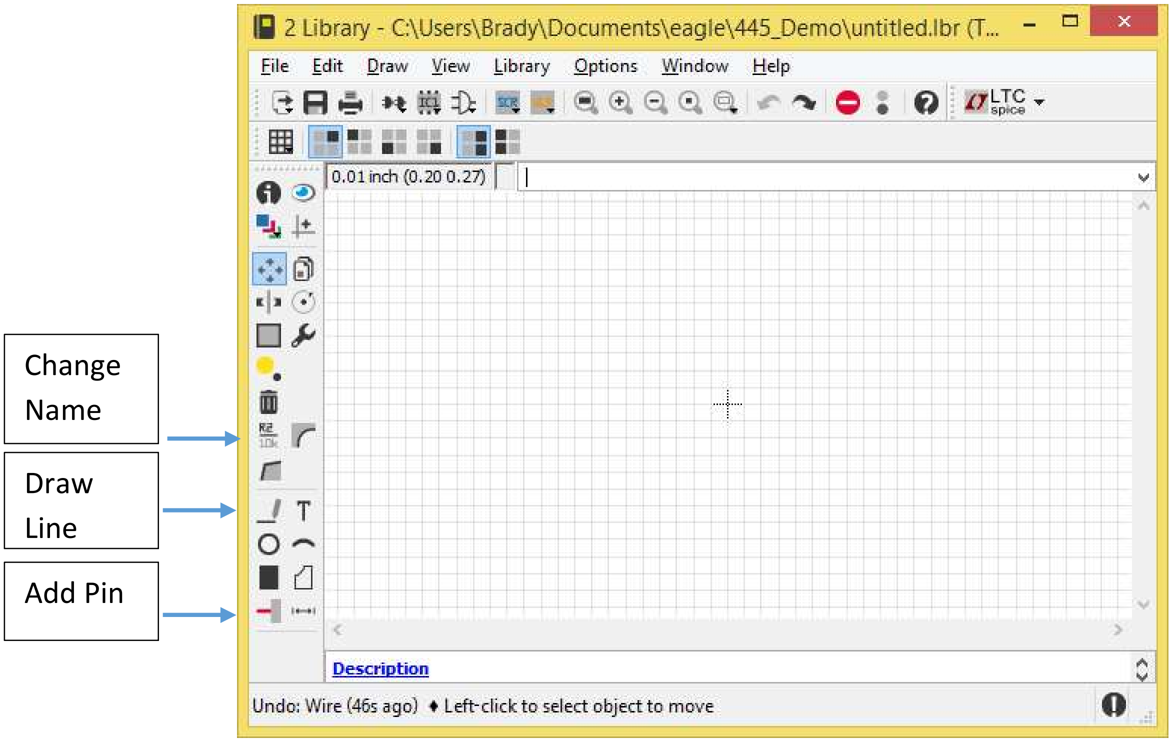
The next step is to add in the TPS799 regulator. You can search in the libraries, but it's not there (part deliberately chosen this way). You can find the datasheet here: http://www.ti.com/lit/ds/symlink/tps799.pdf. For the simplicity of the assignment, we're going to use the "Fixed Voltage Version" on the bottom of the first page. Now the first thing to do is to create a new library in Eagle for this part. Go to the project manager window, select your project, and right click **New -> Library**.



Left: New Device
Center: New Package
Right: New Symbol

Now is a good time to talk about how EagleCAD stores parts. Like you saw when adding parts, components generally have both a schematic and board view. The schematic, or ***symbol*** view, is the agreed upon electrical symbol, and the board, or ***package*** view, is the physical layout. Some items – such as supply lines (e.g. +5V, GND, VSS) – only have a symbol view, where items like vias, traces, and wells only have a board view. When creating a new component, we generate the symbol and package views asynchronously, then combine them together with a ***device*** view. **Note that EagleCAD is not a simulation software of any kind,** we can call a part a resistor and give it three inputs and one output, or a capacitor with only one pin. EagleCAD doesn't care about what the part is electrically doing, only physically.

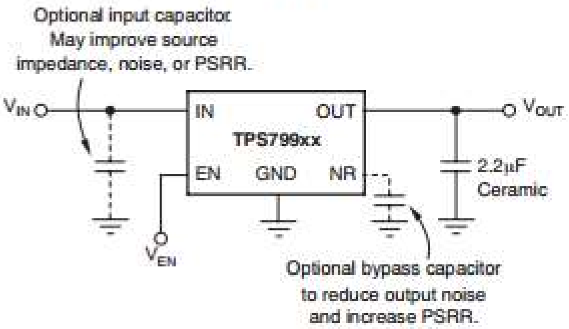
Let's start with making a new symbol. In the pop-up, give your part a name, presumably something along the lines of "TPS799". Press **Sym** to verify we're creating a symbol, the create it.



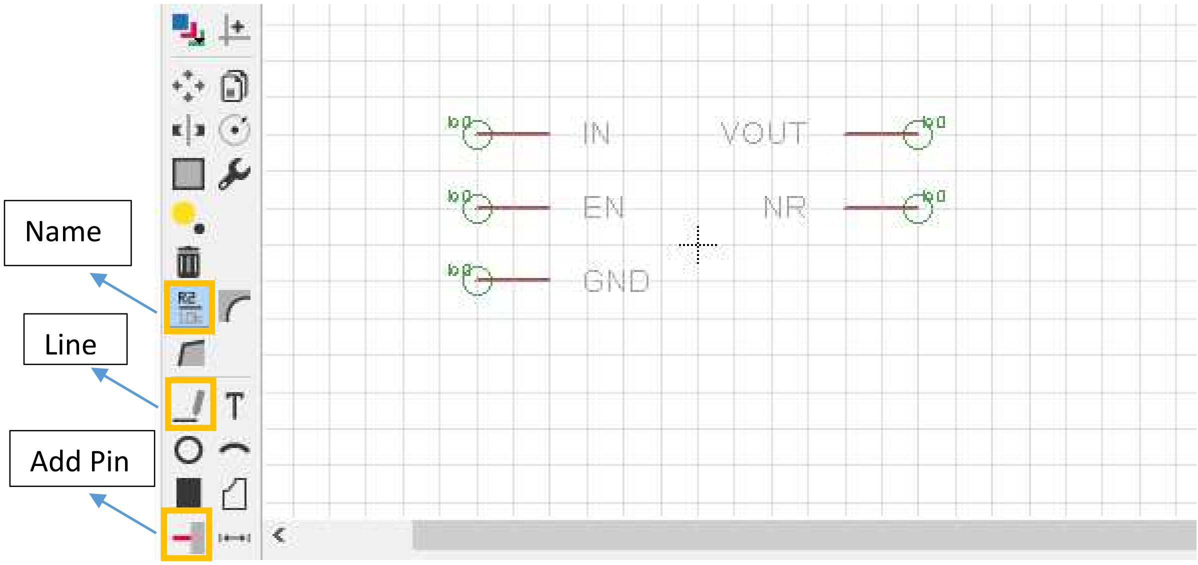
Undo: Wire (46s ago) ♦ Left-click to select object to move

We now have the symbol editor view open. The part we are trying to create is the Fixed Voltage version shown on the bottom of the first page of the datasheet, shown below for reference. For simplicity's sake, let's try and copy the TI's schematic layout as much as we can.

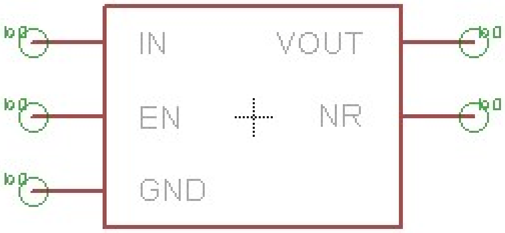
Typical Application Circuit
Fixed Voltage Versions



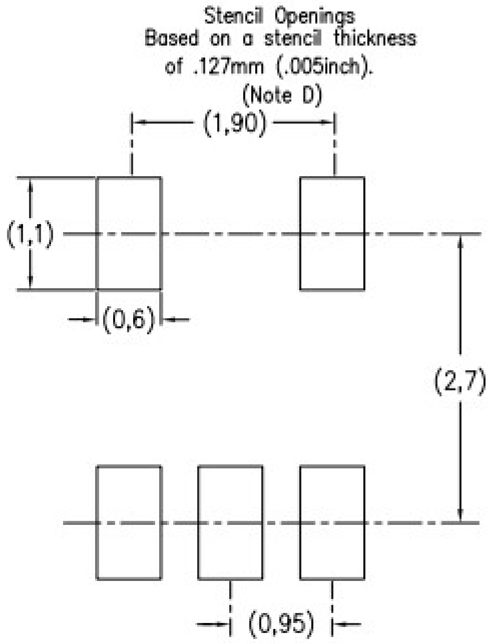
Begin by adding in some pins. For most schematics, people find left to right the easiest to read (like a book), so we'll put the ground pin on the side rather than the bottom. Add 5 pins using the Add Pin Tool, then rename them using the Name Tool. The green circle on the pin is the I/O point, so we'll want those facing outwards. Again, we tend to put inputs on the left and outputs on the right. This makes for a logical reading as we decipher a schematic.



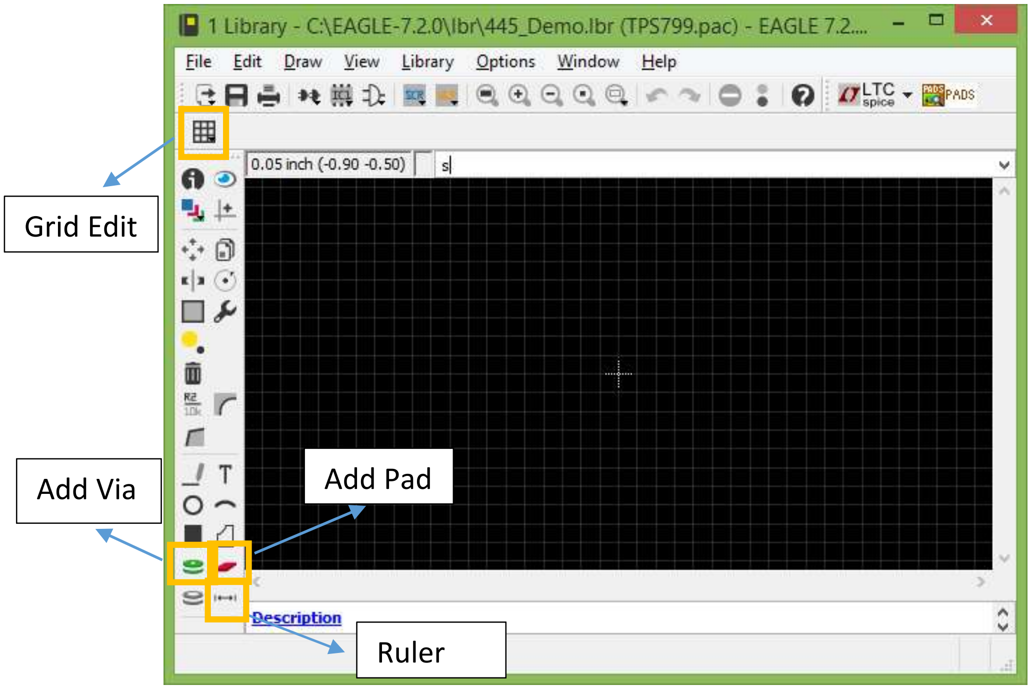
Finish up with the Line Tool to box it off. Try to center the part as much as possible to make your life easier when adding it and moving it around. Save the symbol (as well as the library if you haven't already).



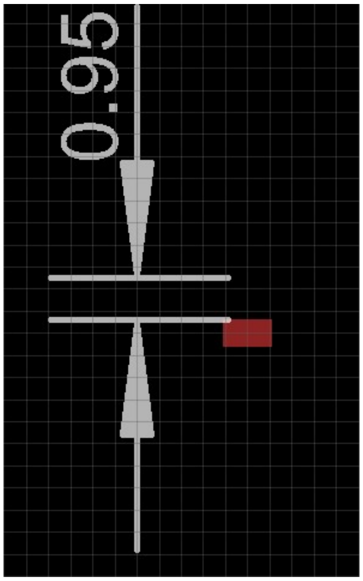
Now we'll handle the package. There are several ways to do this, but we'll do it the most manual way for now, just in case we ever get a new part that's not in Eagle at all. The part has several different packages, but we'll use the biggest one in this case, since if we were to actually create the board, it would be the easiest to use. Packages come in many different types and names, often redundantly so as you'll find out. We'll be using the *DDC* package, also known as *Plastic Small Outline*. While they give many views, we're most interested in the footprint – this is how much space we should leave on the board for the part to sit in. Dimensions shown below in millimeters (from the datasheet).



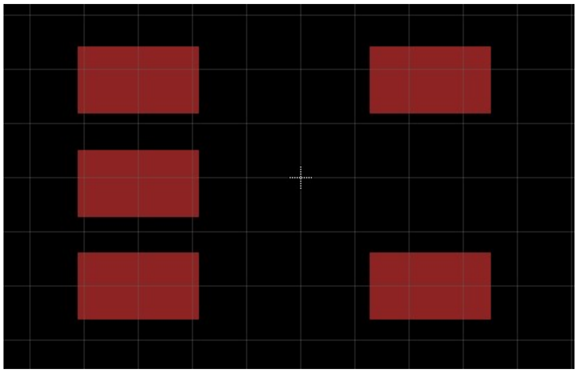
Let’s Hit **New Package**, **PAC**, and name it “TPS799”or something similar. The following window should open. Once again, very similar to what we have seen before, but with two critical new options. The Add Via Tool allows us to add a hole to the footprint that is used for leads or pins. The Add Pad Tool allows us to add a pad for any surface mount part. Lastly, we’ll need to use the Grid Tool to make sure that the minimum grid spacing lines up with the smallest significant figure we have – here 0.05mm.



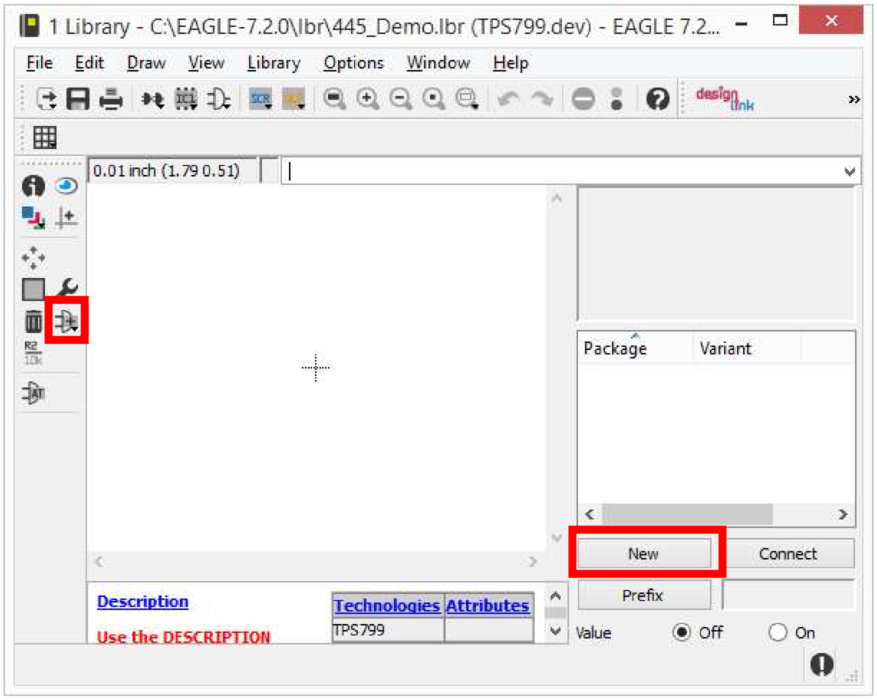
I suggest using a 0.5mm grid, with a 0.05mm alt setting. Add a pad, and place it relatively near the center. Right click on it, **Properties**, and change **Smd Size** to “1.1 x 0.6”. This automatically sizes the pad to the correct dimensions – if it’s off by 90 degrees, rotate it. Now use the Ruler to go and measure out the appropriate distance for the next pad. Remember to hold ALT as you do so – otherwise we’ll be on the main grid dimension, not the finer one. I’ve attached a picture of what the output should like below. Once that is done, copy and paste the previous pad and place it along the same line.



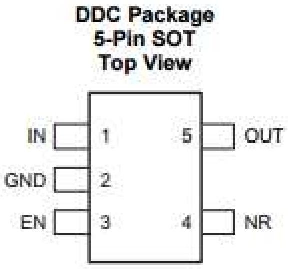
Now this is clearly going to get pretty painful if we want to do this for every pad. Fortunately for us, we can get around this with a little math. Delete the ruler drawing for now, we won’t need it. Add another new pad via copy paste, just putting it anywhere. Right click, select **Properties** on the first pad, and note its **Position**. Now do the same to the newly added pad, but just move it to the appropriate point – either 0.95mm up, or 2.7mm out. Absolute position doesn’t mean anything here – only relative. Once that’s done, go back and name the pads! Any pattern works: [1 2 3 4 5], [TopLeft, MidLeft, BotLeft, TopRight, BotRight], [NW W SW SE NE], as long as you can tell which is which. Final image should look like something below. Don’t forget to verify your layout with a ruler – never hurts to double-check.

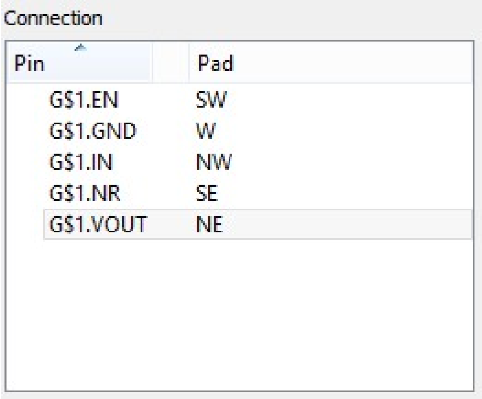


Save it, then go back to the top and open a **New Device**. Once again, let’s call it TPS799. The following window will show up. This should be the easiest of the previous steps – all we do is connect the dots! The button highlighted on the left shows the Add Symbol Tool, and the highlighted new button is the Add Package Tool. There should only be one of each in this library, go ahead and add the schematic and package we created.



Now hit **Connect**, and let’s *connect* the dots (pardon the pun). You can find the order in the datasheet, generally under a section called “Pin Mapping” or “Pin Configuration”. Go through, select the appropriate Pin Name and Pad Name, and then hit Connect. Proceed for all five options, and you should end up with a configuration shown below.

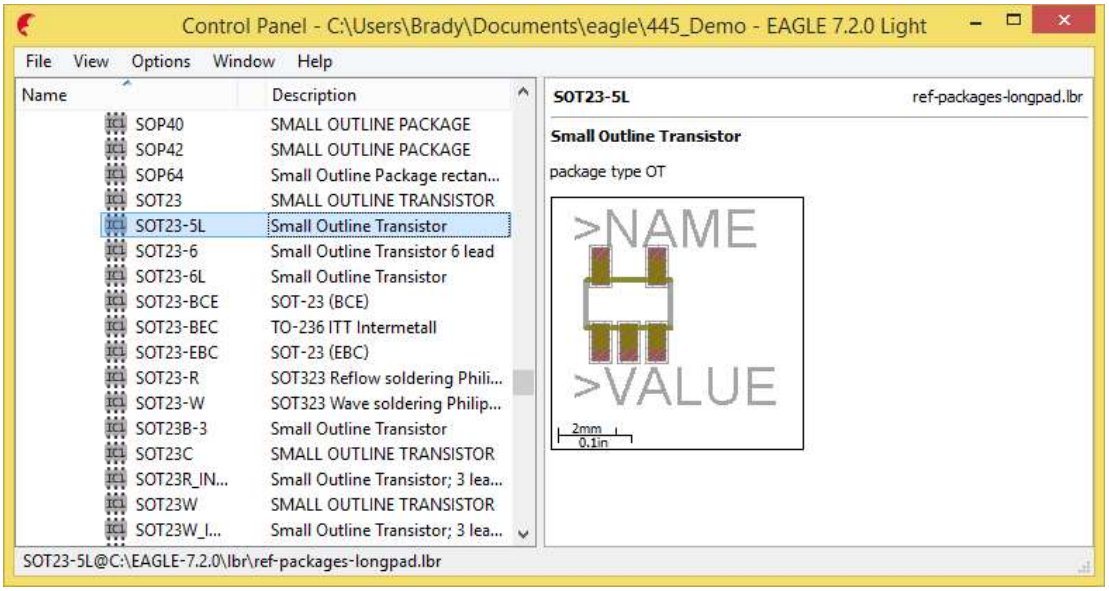



Connection

Pin	Pad
G$1.EN	SW
G$1.GND	W
G$1.IN	NW
G$1.NR	SE
G$1.VOUT	NE

Finish it off and we're (almost) good to go! Before we finish this, let's go ahead and learn one more way of doing this. The large majority of parts you'll be making will probably be IC's. Nearly all IC's come in common packages, so there's no sense in re-inventing the wheel when it comes to drawing packages. EagleCAD stores all its reference IC footprints in **ref-packages.lbr**, or if you aren't comfortable with your soldering skill **ref-packages-longpad.lbr**. The only difference is the longpad variant has, no surprise, larger pads in order to give more soldering area. If you have limited experience soldering, or are using part with a small pitch, it's recommended to use the longpad variant.

The tricky part about this is that you have to know that many IC footprints are the exact same physically, but they have plenty of name variants. For example, the part we have here is given to us as DDC or R-PDSO-G5, but is found in ref-packages as SOT23-5L. While you have your old library open, go back to the project viewer. Go to **Libraries** -> **ref-packages-longpad.lbr -> SOT23-5L**. Right click, **Copy to Library**. This will copy a version of the footprint to your library, so when you go to make a new symbol, you can see you now have two footprints to choose from – the one we designed, and the one given to us.



Control Panel - C:\Users\Brady\Documents\eagle\445_Demo - EAGLE 7.2.0 Light

Name	Description
SOP40	SMALL OUTLINE PACKAGE
SOP42	SMALL OUTLINE PACKAGE
SOP64	Small Outline Package rectan...
SOT23	SMALL OUTLINE TRANSISTOR
SOT23-5L	Small Outline Transistor
SOT23-6	Small Outline Transistor 6 lead
SOT23-6L	Small Outline Transistor
SOT23-BCE	SOT-23 (BCE)
SOT23-BEC	TO-236 ITT Intermetall
SOT23-EBC	SOT-23 (EBC)
SOT23-R	SOT323 Reflow soldering Phili...
SOT23-W	SOT323 Wave soldering Philip...
SOT23B-3	Small Outline Transistor
SOT23C	SMALL OUTLINE TRANSISTOR
SOT23R_IN...	Small Outline Transistor; 3 lea...
SOT23W	SMALL OUTLINE TRANSISTOR
SOT23W_I...	Small Outline Transistor; 3 lea...

SOT23-5L ref-packages-longpad.lbr

Small Outline Transistor

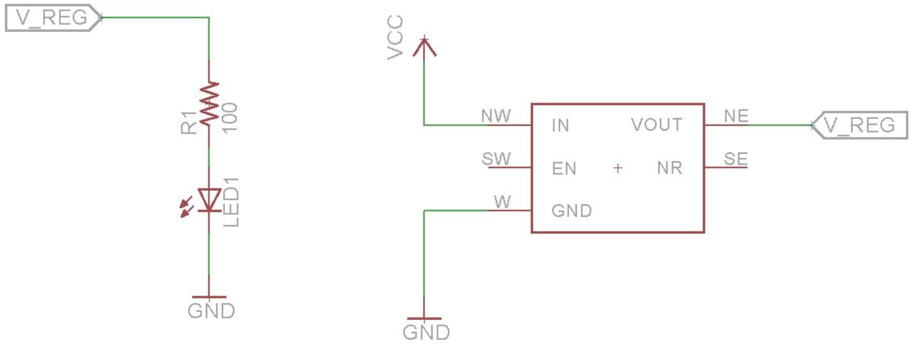
package type OT

SOT23-5L@C:\EAGLE-7.2.0\lbr\ref-packages-longpad.lbr

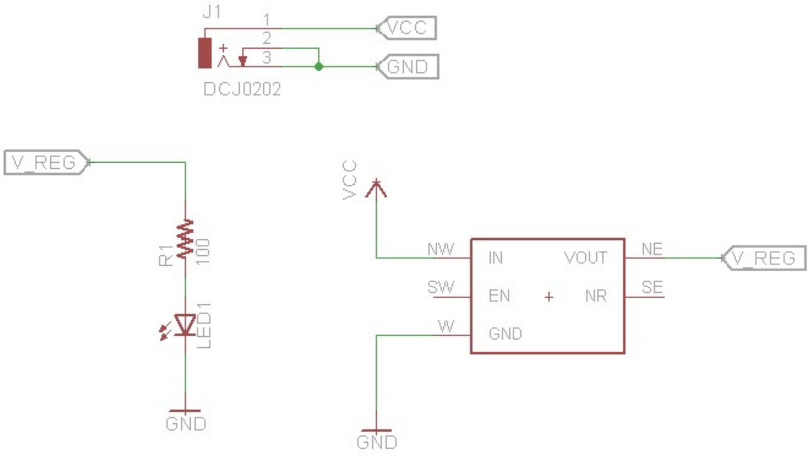
The choice is yours on which method to use. Both have their pros and cons, and unfortunately unless you get lucky with the footprint, take a little time. Regardless, now that we have our part created, it's time to add it to the schematic and move on to the PCB layout.

Adding our New Part

Let's quickly add our new TPS799 to the schematic and add some basic connections. To use our part, go to the top toolbar and hit **Library -> Use**. The library you just created should be in the default folder, open it and the parts in it will be available to us. Add it in and wire it up. Below just a quick view for the purpose of the tutorial – **don't forget to finish the full schematic for the regulator later**.



Now we need to add some way to connect to the outside world. While it's nice to having VCC and GND listed on the schematic, they can't just magically appear there. We need to add connectors of some sort to interface to a power supply, a breadboard, wall outlet, etc. Add a part, and choose from any of the connectors available. There is easily hundreds, and most are found under the prefix **con-**, or if you just want to use simple breadboard headers, **pinhead**. I'm going to use a DC jack (the barrel jack style a wall-wart has) and hook it up to VCC and GND. Feel free to use whichever jack you like as long as you can get those two levels off of it.

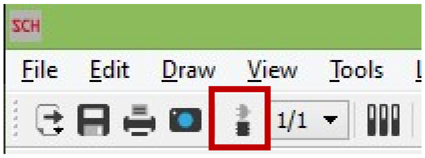


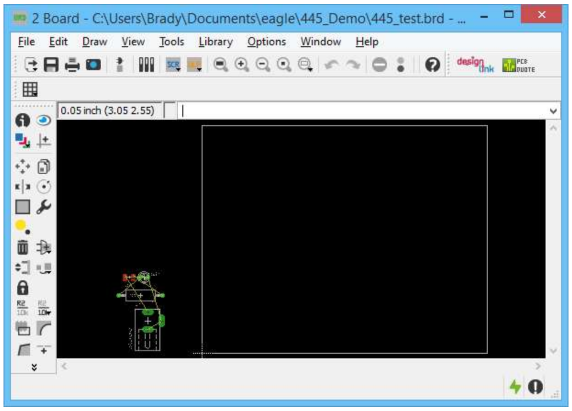
Another important note is the green dot I added on the jack when I crossed two wires. This was done by the Junction Tool, which is located next to the Label Tool. Whenever we cross wires in EagleCAD, by default those are NOT connected. We need to add the green dot to ensure they are indeed connected. It's a nice visual aid, as well as a way to confirm that we do indeed want to connect wires when we cross them. We can check for this and other small problems with the **ERC Tool,** Electrical Rule Check. It's the bottom left most button, press it and see what errors you get. Feel free to ignore any irrelevant ones and deal with the important ones. **Again**, **finish the full schematic for the assignment**, but for now let's learn the PCB side of it.

Making the PCB

PCB design is just a big puzzle. If you've ever played or enjoyed one of the "connect the dots" or "Flow" style games, you'll probably enjoy this part. All we try to do is match our schematic by connecting the pads and vias, with a general goal of minimizing area. Each board consists of several layers to "route" (or connect) on. The free version of Eagle allows you to use two layers, the paid version on the EWS machines allows four, and most professional boards can have anywhere from six to sixteen!

The first step is to switch from schematic view to PCB view. Press the highlighted button in the toolbar to create the board view. You'll be presented with the screen below. What EagleCAD has done is save the netlist (or nodelist) from the schematic, and connected all the parts that need to be connected with yellow wires. The white square is our "work area", but we can make it any size or shape we prefer.

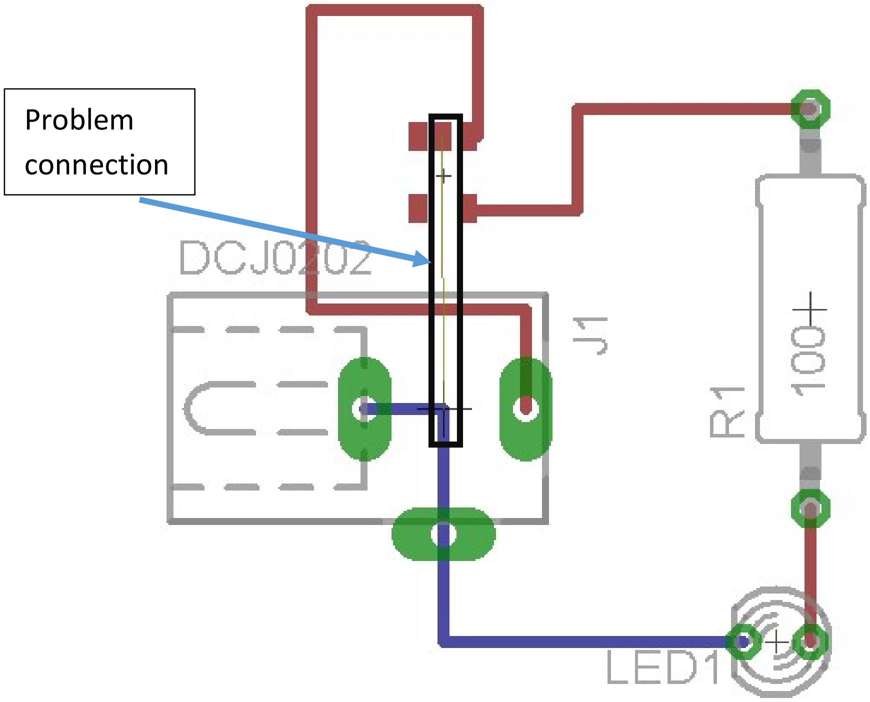



2 Board - C:\Users\Brady\Documents\eagle\445_Demo\445_test.brd - ...
File
Edit
Draw
View
Tools
Library
Options
Window
Help
0.05 inch (3.05 2.55)

Begin by moving some of the parts into the work area. The Route Tool is what we use to make these connections. When you select it, you'll see one of the toolbars show a menu like so 16 Bottom. The dropdown bar chooses what layer we route on, and additionally, each layer is color coded so you can always visually confirm as well.

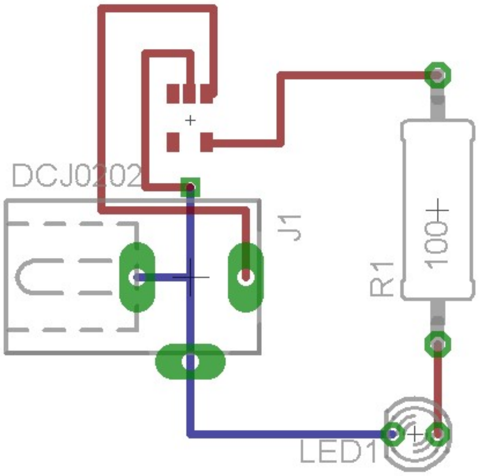
You can only route between nodes, so pick any part with a yellow line and begin. When you click one pad/via, the nodes is needs to connect to light up. A green via represents a hole that goes all the way through the board, so you can connect on any layer. A red or blue pad is created only on its corresponding side, so you will need to draw on the appropriate side. These lines we draw represent copper traces, which is a conductive material. Therefore if you overlap these lines, or accidentally overlap a line with a pad/via that you're not supposed to, you'll be creating an electrical short. Watch out for that, and only connect what's supposed to!

Continue adding parts on and connecting them. Every once in a while, hit the Ratsnet Tool ⁙. This tool redraws the yellow electrical lines to the shortest configuration possible, which is very useful when you have a lot of parts you're moving around. Below you'll see a drawing of mine mostly done, except for one last connection. The problem here is my closest connection is on the bottom, but my pad is on the top.



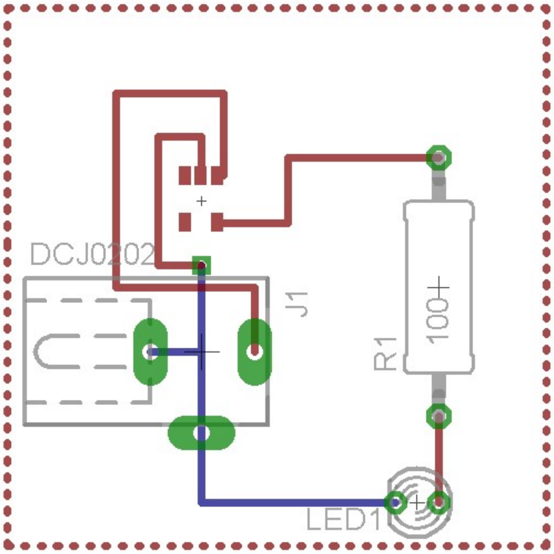
Problem connection

We can easily solve this by adding in a via when we draw the route. For this case, I would start off Route Tool on the top side, selecting the pad. When I get to a not crowded spot, *with the Route Tool still selected*, change the dropdown menu from "Top" to "Bottom" and finish the trace. See that small green square in the middle? EagleCAD automatically placed a via for me to short + connect my two traces.

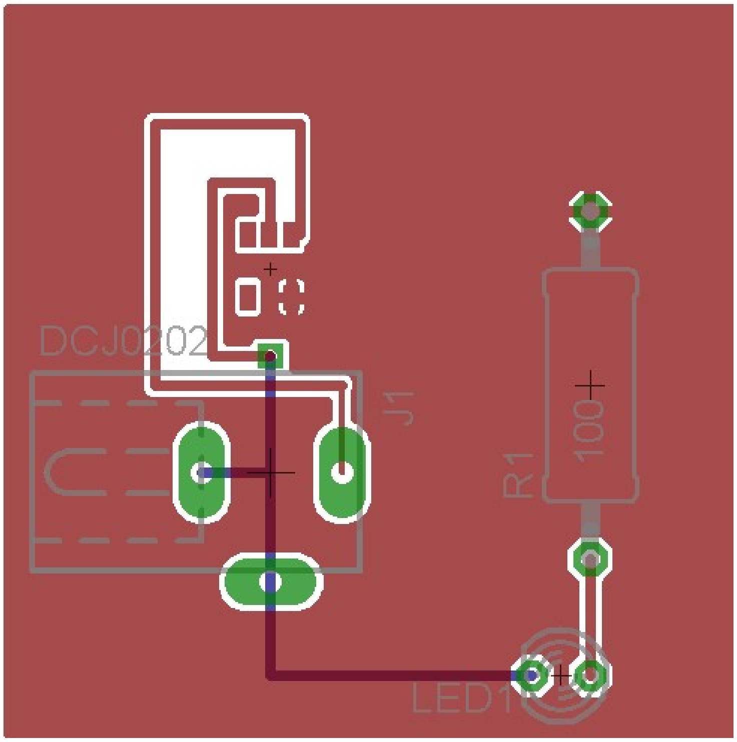


Nice, now it's all wired up. Let's get into a bit more detail about some of the other tools in EagleCAD. For one, you can't delete traces, you rip them using the Rip-Up Tool . You can delete parts and holes and other added components, but not traces. Eagle also allows us to use wells/planes on our PCB. For most manufacturers, anything that isn't a trace is just left alone as raw material, but this often isn't desirable. Remember, all traces are effectively resistors (or you start to go fast, capacitors and resistors) so we want minimize the impact they have on the circuit. By making them as wide and as short as possible, we can reduce their resistance.

To do this, make a box around the PCB using the Shape Tool . Don't overlap at all, you might get some ugly errors if you do, just connect the end right to the start. You should have a nice dotted box around your circuit like so.



Now use the Name Tool to name the box. The name is going to be whatever signal you want to fill that area with, so you have to choose a net name that you assigned in the schematic view! I'll choose V_REG for mine, as it gives the best example for my layout. Hit RatsNest to see the outcome. In my case, the large majority of the board just became entirely red, as all that signal got filled in with the regulator output voltage. Right click on the box (the outline, not the filled in area), and select **Properties**, then under **Polygon,** change **Isolate**. This is the parameter that determines how much separation exists between the fill and the other signals on that level. Be careful about making it too small, as you'll find it harder to solder to, as well as increasing the risk of coupling signals together accidentally! Below you'll see mine with a decent Isolation value.

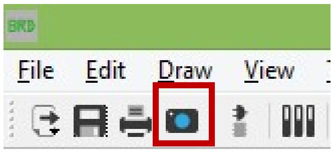


Almost done! Just like in the schematic how we had the Electrical Rule Check, here we have the Design Rule Check . While the ERC looks our circuit over and sees if we have anything wired oddly, the DRC not only tells us if we laid out anything strangely, it also warns us if we did anything that isn't possible to be manufactured. While Eagle may allow you to place parts apart with nanometer, good luck trying to find someone who will actually make that! Most services offer a design rule listing so you can go in and find out exactly how wide to make your traces, or how far apart to spread your parts.

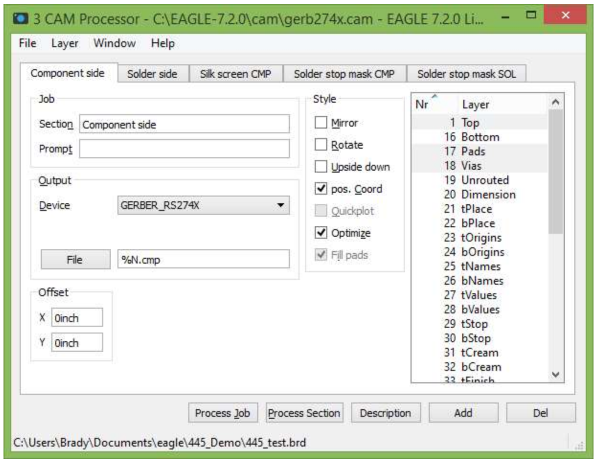
The last tool we haven't really talked about is the AutoRouter . This is for good reason – it tends to suck. PCB layout, and EDA in general, is a huge unsolved problem in terms of optimization and automation. While with digital solutions there is enough reduction that the problem is more manageable, analog layout is still a bit of an art that hasn't been completely mastered. The AutoRouter will most likely find a solution, but it's far from optimal most of the time. Feel free to experiment and see what designs it churns out, but in general it probably won't be of much use.

Creating Gerbers

While not required for the assignment, it's good to learn how to go from the EagleCAD board file to gerber files. The gerber format is a binary vector image filetype that is the common medium for manufacturers. Plenty of free and online gerber viewers exist if you wish to view what is actually being outputted, but it effectively just slices the PCB layer by layer and lists each x by y point as a region.



Select the CAM Processor (highlighted above) to get started. Select **File, Open, Job** and select **gerber274x**. The window should populate with several tabs, each representing a new output file. Each output file is a new "slice" of the board. Pressing Process Job at the bottom will create all of these files in the same directory as your project.



3 CAM Processor - C:\EAGLE-7.2.0\cam\gerb274x.cam - EAGLE 7.2.0 Li...

Job
Section: Component side
Prompt:

Output
Device: GERBER_RS274X
File: %N.cmp

Offset
X: 0inch
Y: 0inch

Style
□ Mirror
□ Rotate
□ Upside down
☑ pos. Coord
□ Quickplot
☑ Optimize
☑ Fill pads

Nr	Layer
1	Top
16	Bottom
17	Pads
18	Vias
19	Unrouted
20	Dimension
21	tPlace
22	bPlace
23	tOrigins
24	bOrigins
25	tNames
26	bNames
27	tValues
28	bValues
29	tStop
30	bStop
31	tCream
32	bCream
33	tFinish

C:\Users\Brady\Documents\eagle\445_Demo\445_test.brd

Why are there so many files for a two layer board? Well you can see what each "slice" actually is in the layer tab on the right. We have our top and bottom solder, as well as soldermask, silkscreen, and a drill file to notate all the holes in the board. Fancier boards require even more layers, with items like glue points and solder stop points.