

PCB Layout Tutorial – Eagle CAD

1.0 Introduction

An important part of embedded electronics development is the design and creation of printed circuit boards, or PCBs. PCBs are used to conveniently and compactly connect electrical components to one another, and serve as the basis on which nearly all modern integrated electronics are built. This tutorial is intended to show a student how to design a printed circuit board for manufacture.

EagleCAD is a freely available design suite used for the design and layout of printed circuit boards. At the time of this writing, EagleCAD serves as the primary tool used by senior design students for circuit board design. The most recent version of EagleCAD available at the time of this writing is EagleCAD v7.1, although much of what is detailed in this tutorial is applicable to previous and future versions of EagleCAD.

This tutorial will center around the design and layout of a switching power regulator application circuit. The datasheet [1] for the switching regulator IC and other relevant datasheets are packaged with this tutorial in the “resources” folder.

A number of excellent additional resources exist for students needing additional clarification or assistance. These resources are detailed in section 9, “Further Reading”, at the end of this tutorial.

2.0 Obtaining EagleCAD

EagleCAD is available for free for download online here: <http://www.cadsoftusa.com/download-eagle/?language=en>. A freeware version is available for hobbyists which contains some limitations (only 2 layers, noncommercial use only, a single schematic sheet, and a maximum board size of 4 x 3.2 inches). Professional licenses with additional features can be purchased if desired.

Once EagleCAD has been downloaded and installed, open the application. The Eagle Control Panel window will appear, similar to that shown in figure 1.

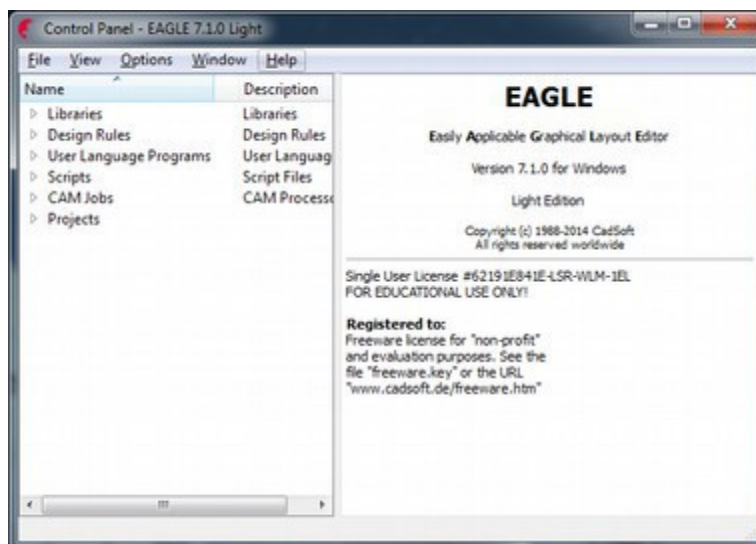


Fig. 1. Eagle Control Panel

3.0 Acquiring Part Models

Prior to a PCB layout being created, Eagle must first have an electrical schematic which details the interconnections between the various circuit elements. The electrical schematic for the switching regulator application circuit is given in figure 2, below (taken from the switching regulator IC datasheet [1]):

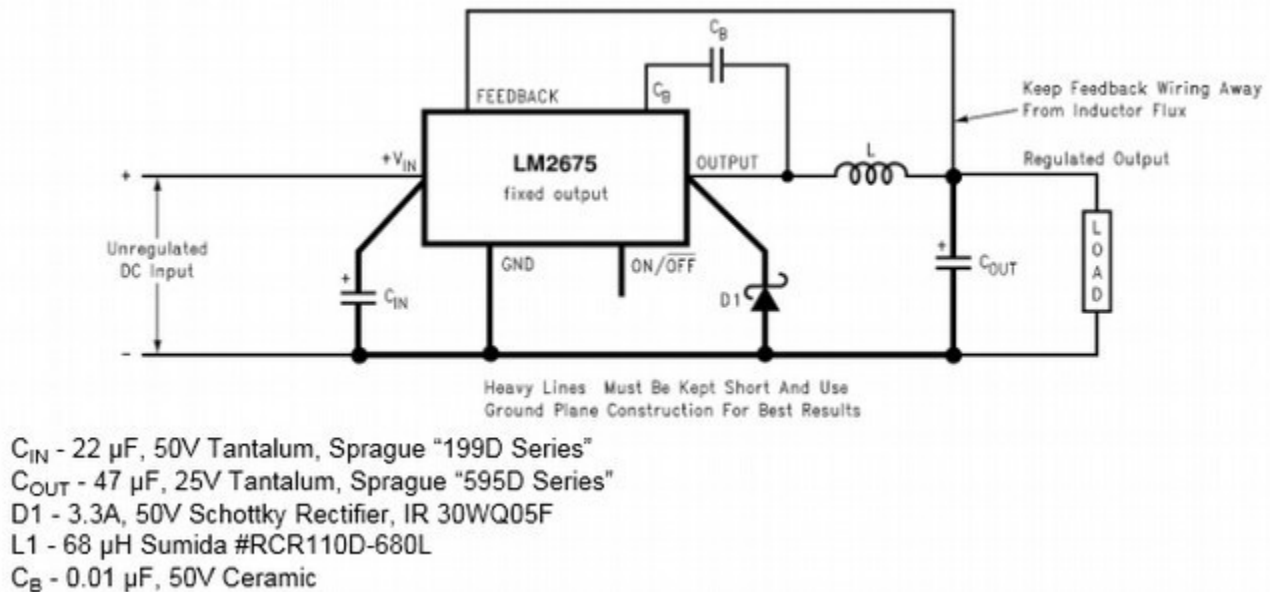



Fig. 2. LM2675 Application Circuit Schematic

In the Eagle Control Panel, select **File** → **New** → **Project**. An empty folder will now appear in the Projects hierarchy. Feel free to name the project whatever you wish (for the purposes of this tutorial, "EagleTutorial" will be used).

A new schematic file must now be created. Right click on the Eagle project folder and select **New** → **Schematic**. A blank schematic window will appear. Save the resultant schematic file (for the purposes of the tutorial, "PowerSupply.sch" was used).

A good practice to follow when creating electrical schematics is to include a documentation frame. Frames allow one to package their schematics neatly, and assist others in locating specific sections of the schematic for commenting or review. We will now add a simple landscape-orientation A4 frame.

Select the **Add**  button from the left toolbar (alternatively, **Edit** → **Add**) to enter the Eagle parts selector, shown in figure 3 below:

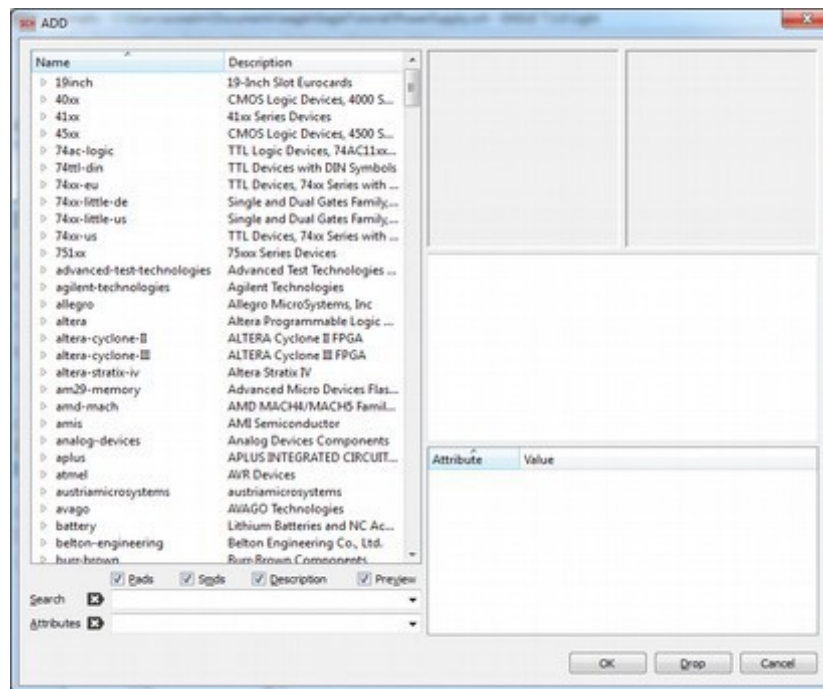


Fig. 3. Eagle Parts Selector

In the search box within the parts selector, type “frame”. The parts selector should now show a handful of documentation frames within the default Eagle library “frames”. Select the part “FRAME_A_L” and click OK. That part will now appear within the schematic window. Place it so that the lower left corner lines up with a pair of dotted crosshairs indicating the origin point. Your schematic should now resemble figure 4, below.

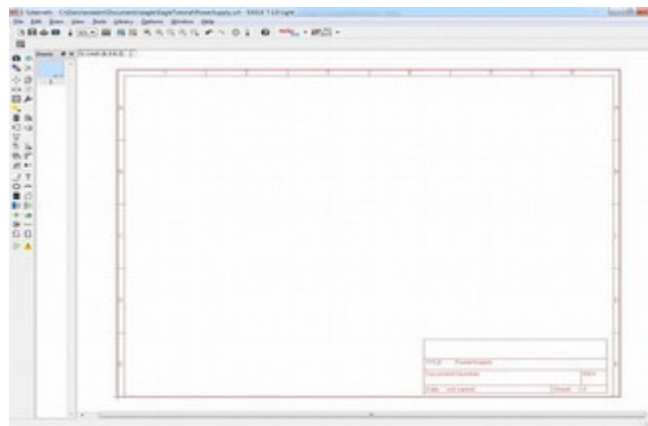


Fig. 4. Schematic w/ Added Frame

With the frame instantiated it's time to add the parts necessary for the switching regulator. These parts can be obtained from the library “ECE477_Resources.lbr”, which is available here: https://engineering.purdue.edu/ece477/Resources/eagle/lbr/ECE477_Resources.lbr. Download this file and unzip it to your Eagle Project directory. Once you have done so, you must add the library to Eagle's active library lists. In the schematic window, select **Library** → **Use...**, an explorer window should open

up. Navigate to your Eagle Project directory, select “ECE477_Resources.lbr”, and click **Open**. The new library and its parts are now available for use in your Eagle schematic.

In some circumstances, you may wish for frequent, convenient access to the libraries you've created or acquired which are not included in Eagle by default. To obtain this access, open the Eagle Control Panel. In the Control Panel, select **Options** → **Directories...** In the paths dialog window that appears, append your desired search path to the “Libraries” box (append an additional search path by adding a semicolon, followed by your desired search path).

With the library included in your project you can now instantiate some of the parts necessary for the switching power supply. Click the Add button. Now navigate to the newly available ECE477_Resources library. The library should contain 3 parts: the 30WQ05F Schottky diode and two capacitors, the 595D tantalum parts called for in the application circuit (at the time of this writing, the datasheet references a 199D part; use a second 595D part as this is a typo). Select each of these parts in turn and place them in your schematic.

Our switching power supply will be used in conjunction with other circuits and designs, thus our design should be given pins to serve as inputs and outputs to the system. For this, we will be using something called a “pin header” or simply “header” for short. Headers provide solder points on boards for connectors, among other things. A widely used header is the 0.1” header, in which the holes for pins are spaced 0.1” apart. Using the **Add** command, locate the part PINHD-1X2, located in the default library pinhead. Instantiate 2 2-terminal headers, one for each set of terminals in the application circuit of figure 2.

Part “C_B” in the application circuit calls for a 0.01μF 50V ceramic capacitor of no specified package type. For the purposes of the tutorial a capacitor utilizing a standard 1206 package shall be used. Eagle includes such parts in its default libraries; it is left as an exercise to the reader to locate a suitable part. (Hint: the standard Eagle library “rcl” is a go-to destination for passive components such as resistors, inductors, and capacitors).

With the current parts instantiated, your schematic view should resemble that shown in figure 5, below (zoomed in for clarity).

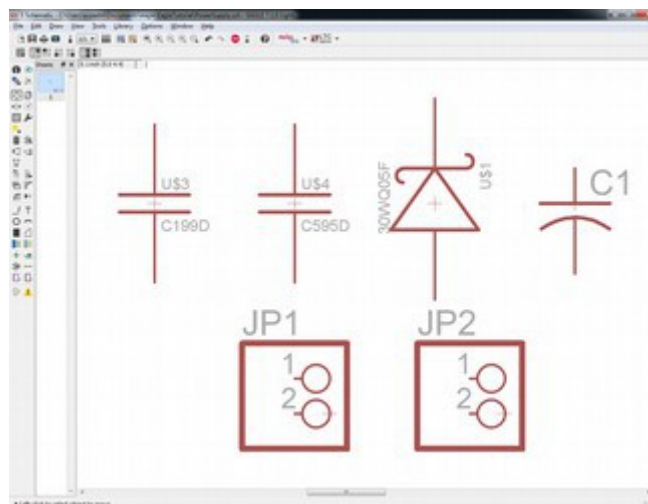


Fig. 5. Schematic with Some Parts Added

Referencing the application circuit of figure 2, we see that the electrical schematic is still missing the LM2675 IC and the inductor. These parts must be created, which is the topic of the next section.

4.0 Custom Part Creation

In this section we will be creating the LM2675 and 68 μ H inductor parts in Eagle for use in our schematic and PCB layout. Eagle requires that parts be maintained in libraries, so to begin we will create one. Navigate to the Eagle Control Panel and select **File** → **New** → **Library**. A window will appear similar to that shown in figure 6 below. Save the library in your EagleTutorial project directory as “ECE477_Tutorial.lbr”.

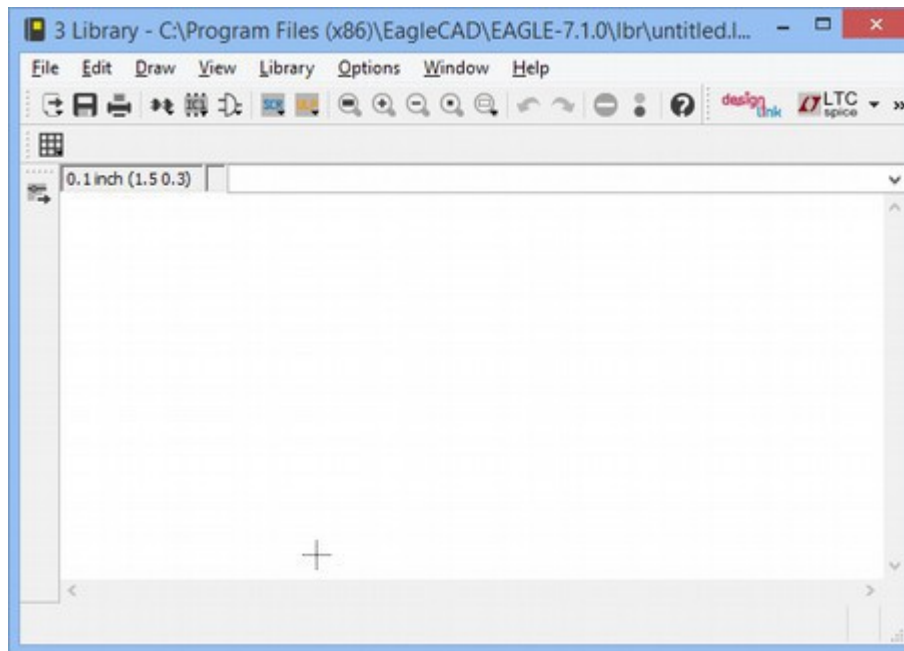



Fig. 6. EagleCAD New Library Window

Eagle parts consist of 3 separate entities: a schematic symbol (how something appears in the schematic view of Eagle), a package (a PCB footprint; how a given part appears in the layout view of Eagle), and a device (an object that describes the interaction between the symbol and the package and what is directly instantiated when a part is added in Eagle). We will begin by creating the schematic symbol for the LM2675. Click the **Symbol**  button in the top toolbar to bring up the library symbol manager. In the box titled **New:**, enter “LM2675” then press **Okay**. Click yes in the text box that pops up. The Eagle schematic symbol editor will appear.

For the purposes of the tutorial, the schematic symbol will resemble the physical part. A diagram of the LM2675 and its associated pinout are shown in figure 7, below:

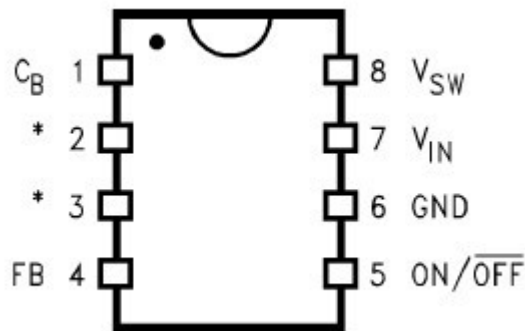






Fig. 7 LM2675 Pinout

Select the **Line**  tool; notice that the toolbar immediately above the schematic area changes context based on which tool you select. Click in the symbol area and drag your mouse around, noticing how the created line reacts to your mouse. Click again to drop a point and establish a line segment. At any time you can undo actions with Ctrl + Z or exit the current instance of line mode by pressing the Esc key. Using the line tool, create a small 6x5 square box. You can move and/or resize your lines at any time using the **Move**  command and points and/or line segments can be deleted using the **Delete**  command.

Once the box has been created, pins must be added to the symbol. Pins are used to establish electrical connections between parts. To instantiate pins within your design, click on the **Pin**  icon to bring up the pin tool. Of particular note, the context-sensitive toolbar for the Pin tool provides the ability to switch between various pin types, including clocking pins, inverted pins, and various pin lengths. Similar to the pinout in figure 7, place your initial pin in the upper left corner, placing 3 pins below it. Right click your mouse and see how the pin rotates 90 degrees. This is one of many useful Eagle shortcuts that works across all the various tools in Eagle (schematic, layout, part creation, etc.). With your pins now rotated 180 degrees from their original orientation, place 4 pins in ascending order on the right side of the box. Your schematic symbol should now resemble that of figure 8.

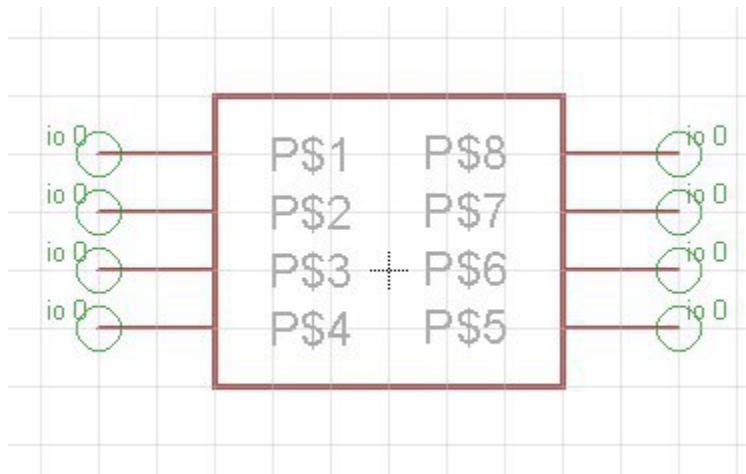




Fig. 8. LM2675 Symbol with Pins

With the pins instantiated they should now be given names which correspond to the pinout in figure 7.

Select the **Name**  tool from the lefthand toolbar and click on each pin to assign it a name. For pins 2 and 3, name them “NC@1” and “NC@2”. Eagle does not allow 2 pins to have identical names, however the @number portions of the names are invisible in schematic mode, allowing you to create a symbol in which it appears as though 2 pins have been given the same name. For the purposes of the tutorial, pin 6 has been named “EN” instead of “ON/OFF” in order to create a cleaner and more compact schematic symbol.

There are 2 important pieces of data that most schematic symbols should have which allow reading schematics to be substantially easier. From the lefthand toolbar select the **Text**  tool; in the box for text type “>NAME”. Before placing the text on the schematic symbol, click the first drop down menu on the context-sensitive toolbar and change the selection to “95 Names”. This changes the active layer to the Names layer, where component name-related information is stored; the color of the text will change to the color specified for the layer (gray, by default). Notice that when attempting to place the text above the schematic symbol, the text snaps to the design grid, causing the bottom of the text to coincide with the top of the box and making it difficult to read clearly. To get around this, press and hold the Alt key. This enables the alternative design grid, or alt grid, a design grid which is finer by default to allow for precision placement of design elements in Eagle. Place the text above the schematic symbol, so that there is a small amount of space between the text label and the box drawn earlier.

Click the **Text** tool again; this time, enter “>VALUE” when prompted for text. Change the active layer once again, this time to “96 Values” and place the resulting text below the schematic symbol. With this done, your schematic symbol is complete, and should resemble figure 9, below.

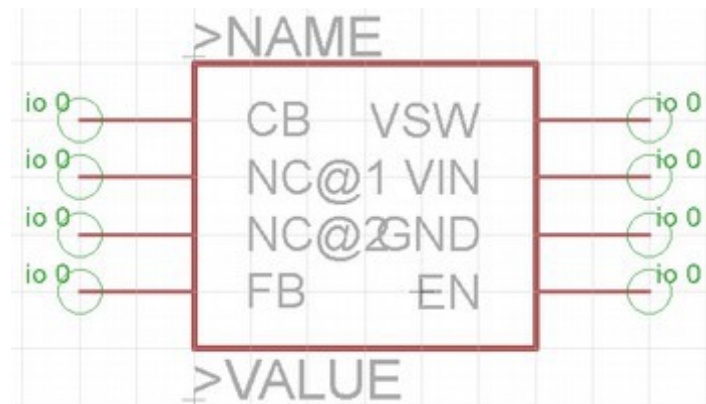




Fig. 9. Completed LM2675 Schematic Symbol

Before the LM2675 can be used in our Eagle project, it must have an associated footprint with it. At the time of this writing, EagleCAD enjoys widespread support and popularity, and as such, many well-made manufacturer footprints exist for a wide variety of component packages. If a suitable footprint can be found, it can be used in our library without needing to be created from scratch, saving us valuable time.

The datasheet for the LM2675 being used in this tutorial indicates that the part is manufactured by Texas Instruments and is available in PDIP, SOIC, and WSON packages. For the purposes of this tutorial, we will search for an appropriate SOIC-8 package to repurpose for our own uses. Bring up the

Eagle Control Panel, and in the file viewer, select **Libraries** → **lbr** → **ref-packages.lbr**. This is an Eagle default library which contains a variety of footprints for various package types. Expanding this library reveals a list of devices and packages contained within the library. Devices are indicated by a symbol  resembling an AND gate while packages are indicated by a symbol  resembling a 6-pin IC. Navigating through the library, search for a package titled “SO8”. This package is an 8-pin SOIC (small outline integrated circuit package). Right click on the package and select **Copy to Library**. The package has now been copied to the currently open Eagle Library (in this case, ECE477_Tutorial.lbr).

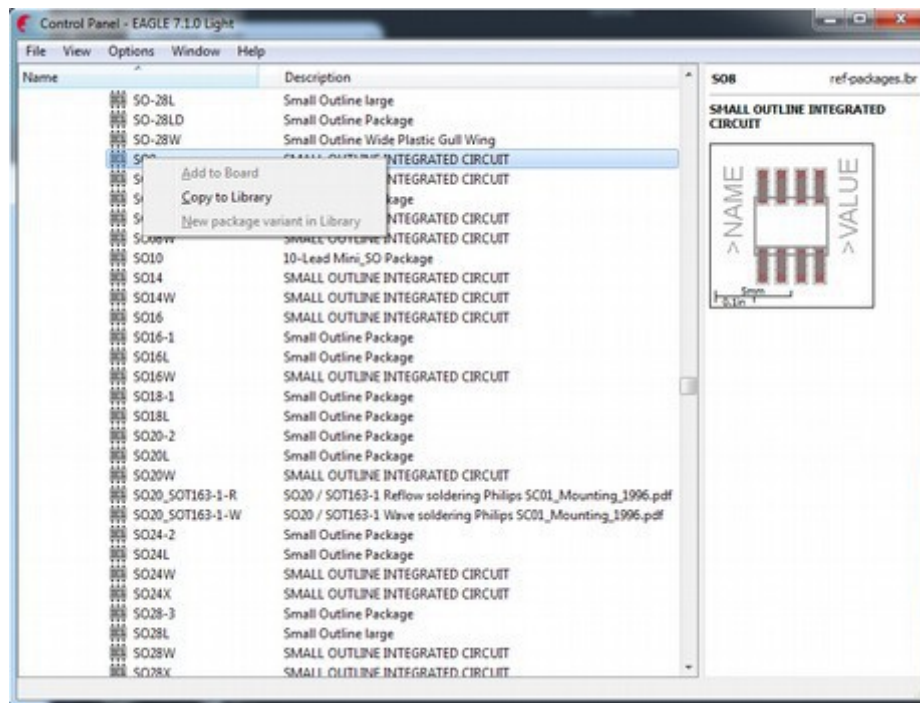



Fig. 10 Copying a Package to an Eagle Library

There is one word of warning when borrowing default Eagle packages. Many packages, especially those from manufacturer-specific libraries, are designed for industrial assembly processes, which include such things as reflow soldering and robotic SMD placement tools. When designing a prototype that is to be assembled by hand, some of the tolerances of these parts may be less than ideal, and the packages may require tweaking to be more forgiving to hand soldering. For this reason, always print out a 1:1 copy of PCB designs and place components you intend to use on the printout to ensure that the packages in your design match and are compatible with the actual, physical parts.

The last step for creating the LM2675 device is to associate the symbol and the package together. In the Library view of Eagle, select the **Device**  icon to bring up the Device Manager for the ECE477_Tutorial library. In the **New** box, enter “LM2675” and hit OK (when asked if Eagle should create the new device, click “Yes”). The Eagle Device Editor will open; it should resemble figure 11, below.

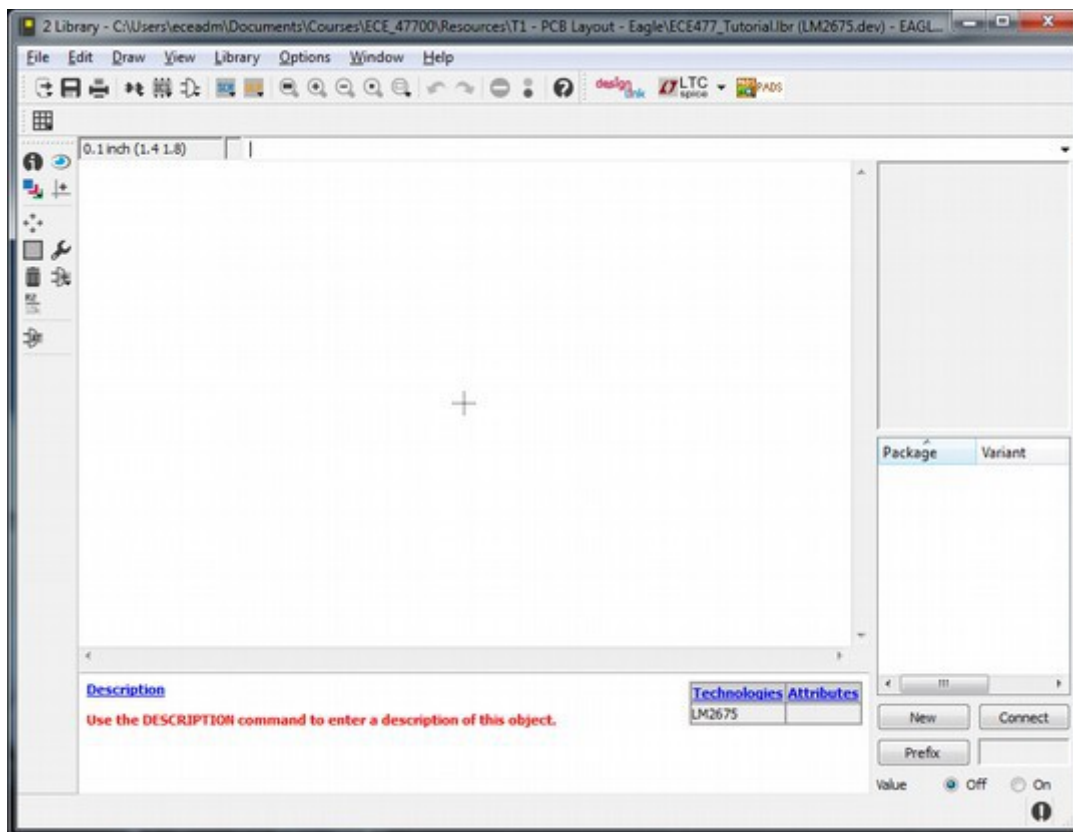


Fig. 11. Eagle Device Editor

Click the **Add** button from the lefthand toolbar to select and instantiate the LM2675 symbol previously created into the device editor. In the lower right corner is the packages window. Click the **New** button to create a new package variant of the LM2675 device. In the window that pops up, select the SO8 package. Enter “SOIC8” in the **Variant name** field, then press OK. Now, press the **Connect** box to bring up the device connections menu; it should appear similar to figure 12 below.

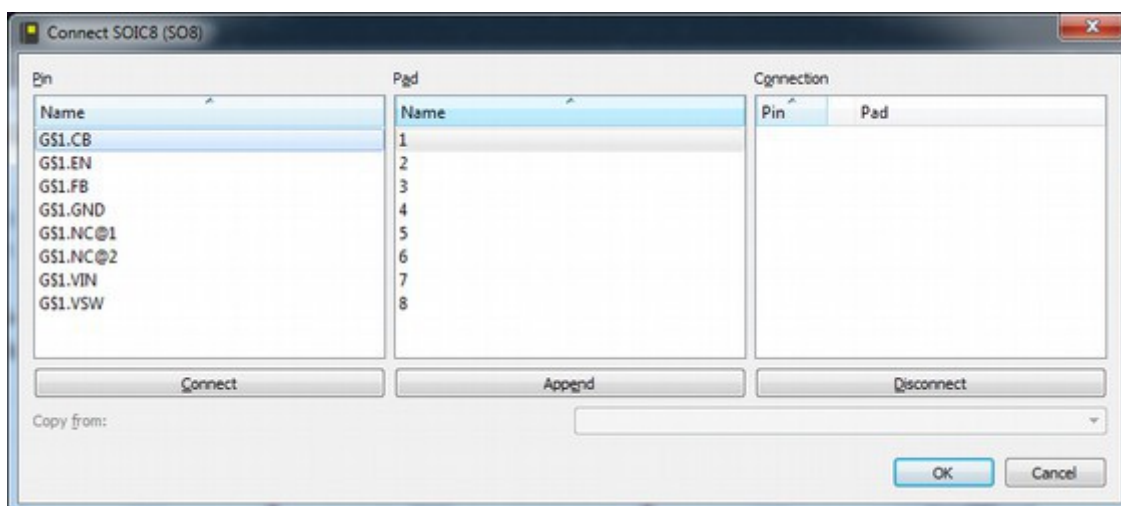


Fig. 12. Device Connections Menu

Using the LM2675 pinout of figure 7 as a guide, define all of the electrical connections for the LM2675 device. Select a schematic pin from the leftmost window and a package pad from the center window, then press **Connect** to associate the pin to the pad at the device level. When all of the electrical connections have been defined, press OK. Your LM2675 part is now ready for use within Eagle.

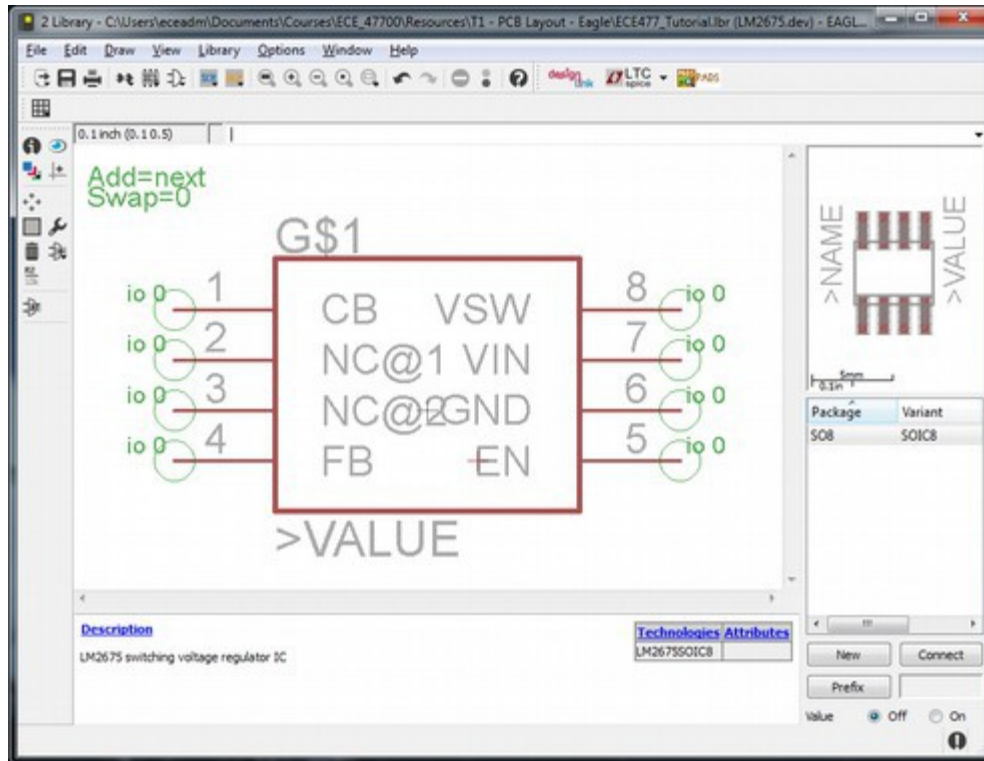



Fig. 13 Finished LM2675 Device

One final part, the 68 μ H inductor, must be created in order to develop the switching power supply design in Eagle. Click the symbol button in the ECE477_Tutorial library and create a new schematic symbol; call this symbol “L”. In the schematic editor, create a 2-terminal symbol resembling the common inductor symbol. This is left as an exercise to the reader (hint: the **Arc** command in the lefthand toolbar is particularly useful for this project). Once the pins are instantiated, use the **Change**  command to set the “Visible” attribute of both pins to “none”. When finished, your symbol should resemble that shown in figure 14.

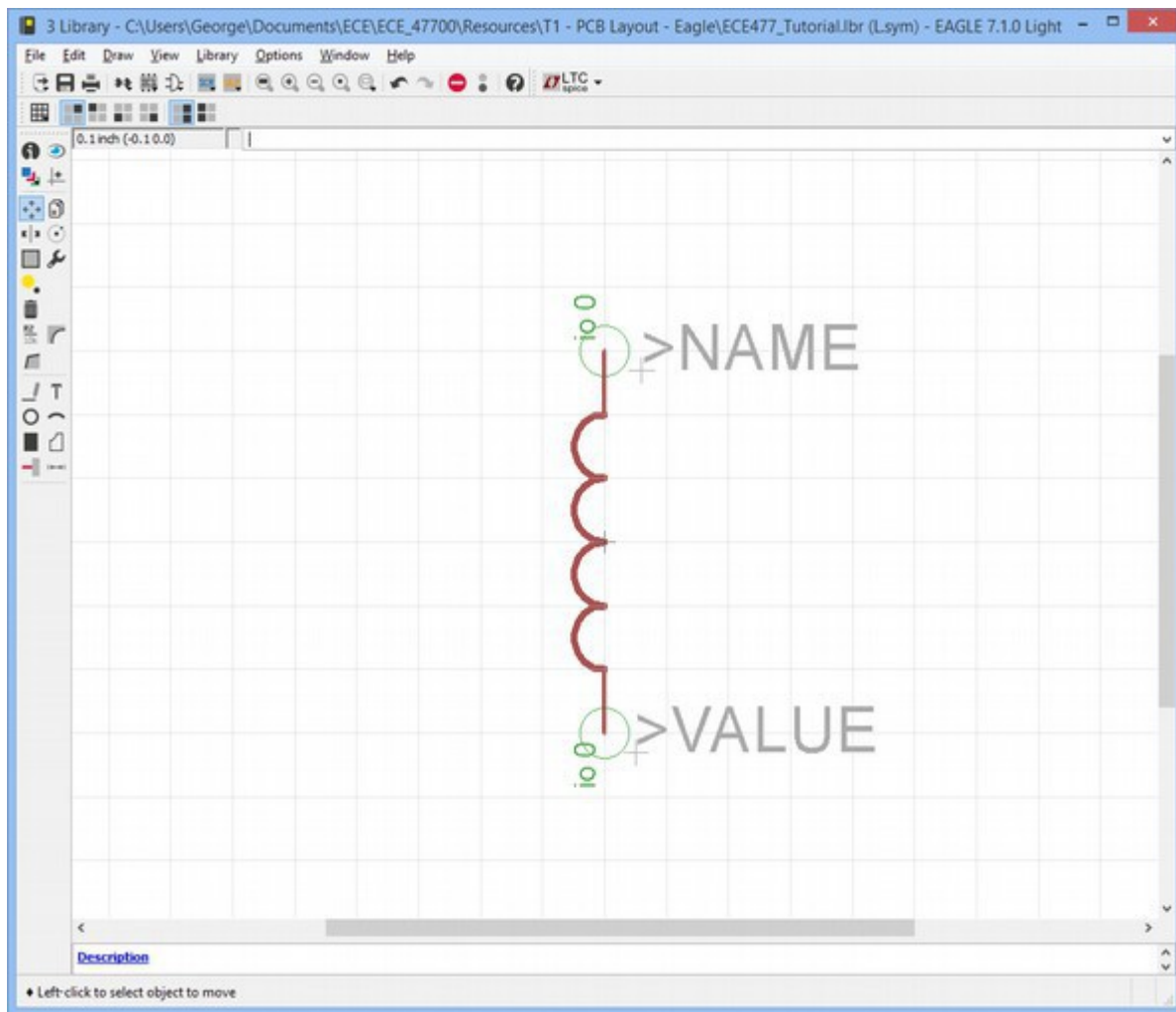


Fig. 14. 68 μ H Inductor Symbol

The inductor called for in this switching regulator does not utilize any industry standard inductor; a custom package must be created in Eagle for the part. Create a new package in the ECE477_Tutorial library called "L_SMD"; the package editor window will open.

The datasheet for the inductors [3] specifies a recommended layout and product dimensions for the part, shown in figure 15, below.

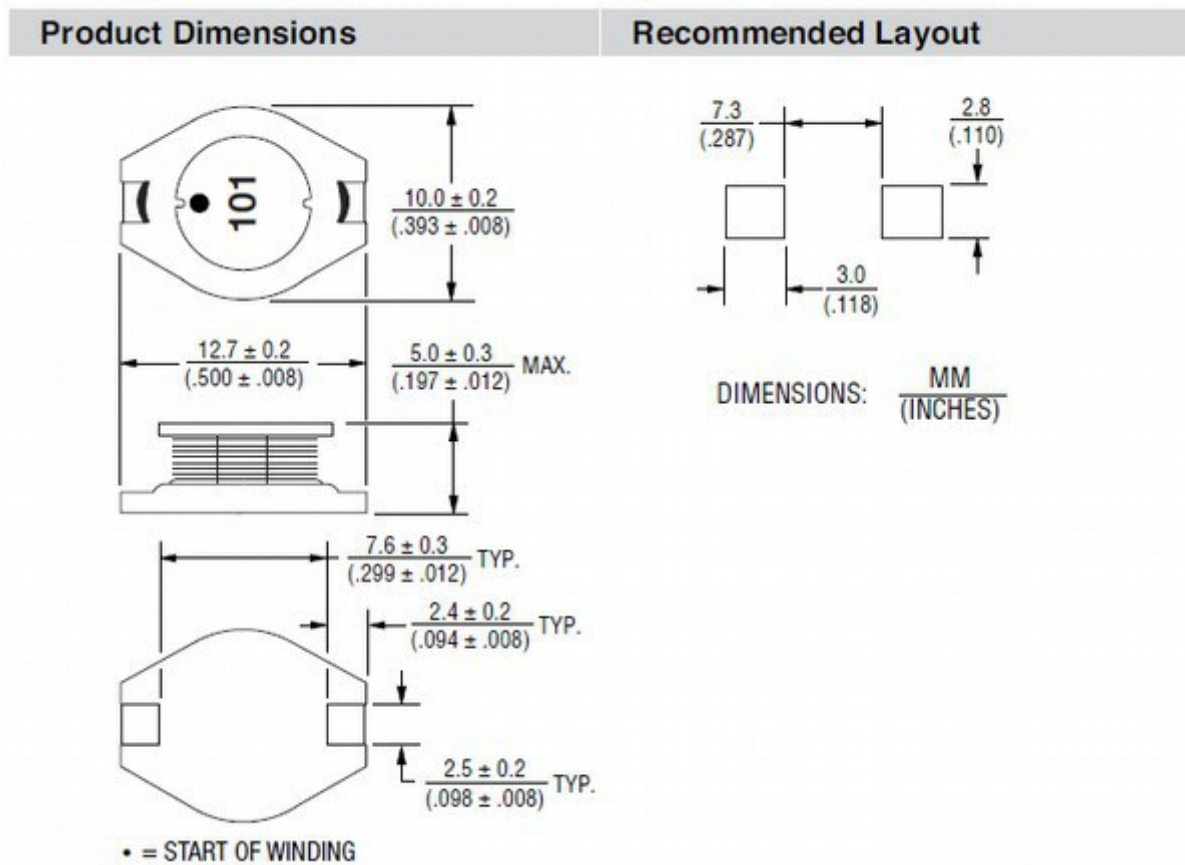




Fig 15. Inductor Package Specifications

Add a pair of SMD pads to the package using the **SMD**  tool from the lefthand toolbar. Specify the size of the pads using the **Size** option in the context-sensitive toolbar above the package editor workspace. Note that the size of the pad is specified in whatever units the main grid is specified (inches, by default). Based on the recommended layout of figure 15, enter a size of “0.118x0.110”. Place the two pads on the layout; don't worry too much about their exact positions, as these will be

adjusted next. With the pads placed, select the **Properties**  tool and click on the pad. By modifying the properties of the pads, we can place them in precise positions on the board which correspond to the recommended layout of figure 15. Move pad P\$1 to a position of (-0.2025, 0) and P\$2 to a position of (0.2025, 0). The pads now match the recommended layout from the datasheet.

In order to finish the package for the SMD inductor, we will now add a few elements to the silkscreen layer of the package. The contents of the silkscreen layer can be printed to the physical printed circuit board and provide values, names, and other useful bits of documentation on the circuit board to assist with assembling and troubleshooting it. For our inductor, we will include a rough outline of the physical part. Using the **Line** tool, create a small octagon around the pads; specify the layer “21 tPlace” for the layer to place lines in (tPlace is a layer reserved by Eagle which is used to place top-layer silkscreen information). As an important note, do not allow silkscreen lines to overlay copper pads, as this can interfere with soldering your device. For the purposes of the tutorial, a small circle was added to the center of the package, resembling the appearance of the inductor. Add “>NAME” and

“>VALUE” tags to your package as was done with symbols previously. Your inductor package is now complete, and should resemble figure 16, below.

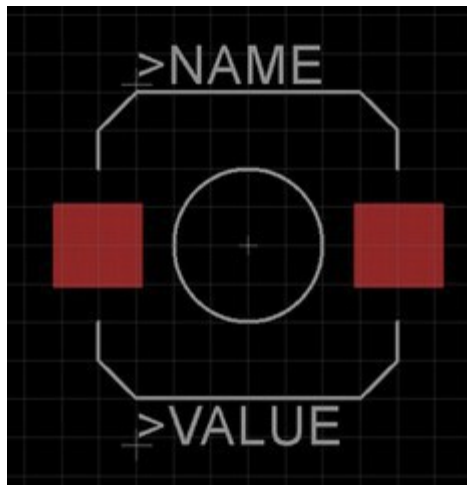


Fig. 16 Completed Inductor Package

A new device should now be created to associate the inductor symbol and package. For the purposes of this tutorial, the device should be given the name “SDR1005”. Connecting the pins and pads of the device is straightforward and is left as an exercise to the reader. Congratulations, your Eagle parts library is now complete. A complete electrical schematic and printed circuit board will now be created in the subsequent sections.

5.0 Electrical Schematics

Add the ECE477_Tutorial library created in the previous section to the Eagle library list if it isn't already. Then, instantiate the LM2675 and SDR1005 parts to your schematic. Your schematic should now have all necessary parts, as shown in figure 17.

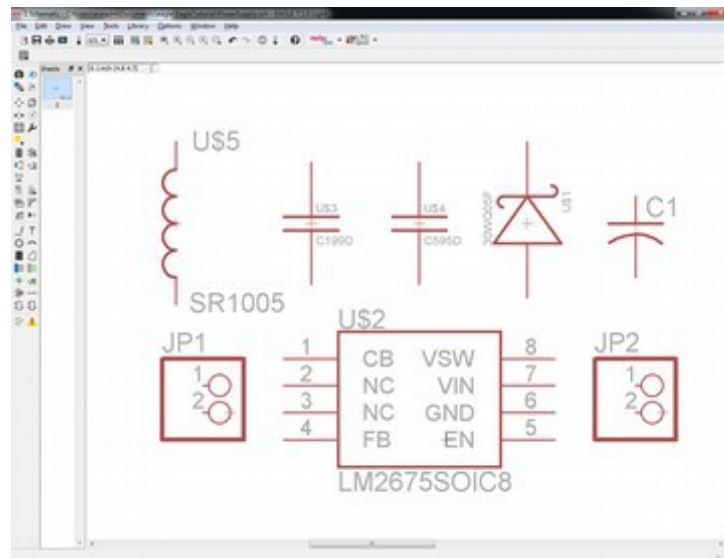








Fig. 17 Power Supply Schematic w/ all Components Instantiated

Using figure 2 as a guide, create the application circuit for your switching power supply. The **Net**  command (known as the **Wire** command in previous versions of Eagle) can be used to define electrical connections within your circuit. Nets/wires snap to the design grid by default; you may have to adjust this to draw your schematic properly. When in Net mode, the context-sensitive toolbar changes to allow the user to specify the bend style for the wires. Right clicking when laying nets allows a user to toggle through the various bending modes. Right angle mode () or one of the two octagonal modes ( or ) are strongly recommended for the purposes of this course.

A word of caution: nets in Eagle must be aligned precisely to the end of pins in order to create an electrical connection. In some situations, the **Junction**  command can be used to formalize electrical connections in Eagle, such as where a net connects to a pin or where nets intersect at three-way or four-way junctions.

When completed, your electrical schematic should resemble figure 18, below. (Note, the **Smash**  command was used to separate the Name/Value information from some of the schematic symbols, allowing them to be repositioned or removed as necessary.)

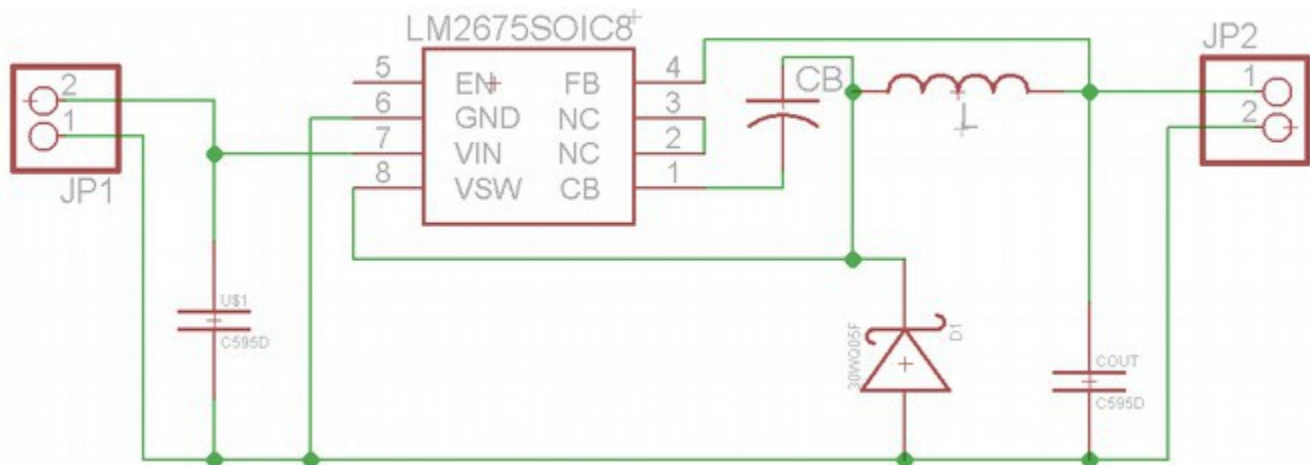



Fig. 18. Completed Switching Power Supply Schematic

6.0 PCB Layouts

From the schematic view, select the **Generate/Switch**  command. This command is used to switch between the electrical schematic and PCB layout Eagle tools; if a schematic or layout for a given circuit does not exist in Eagle, the command will generate one from the other. When asked to create the Eagle PCB layout (.brd) file from the schematic, select 'Yes'. You will be taken to the Eagle PCB Layout tool, shown in figure 19, below.

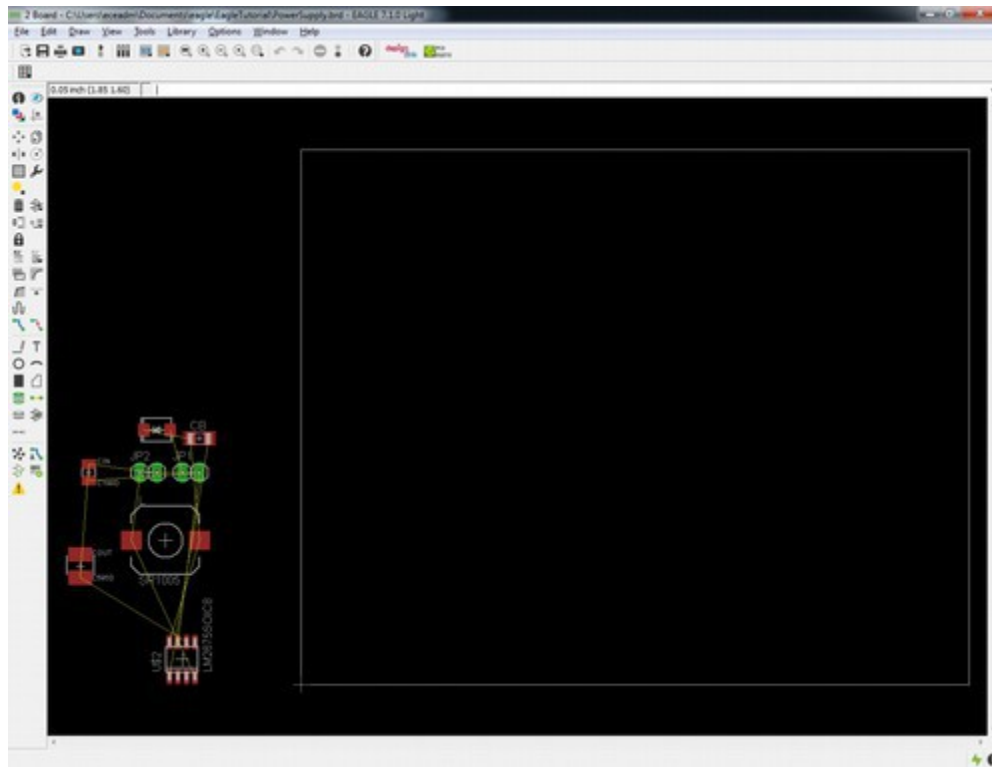



Fig. 19. Eagle PCB Layout Tool – Initial View

When first generated Eagle includes a white rectangle corresponding to the dimensions of the final printed circuit board. In the Freeware version of Eagle this corresponds to the maximum useable board area; no parts or traces may be placed outside of this area. Additionally, the newly generated tool places all component packages in the lower left corner, near the origin point, shown by a small crosshair. The pins and pads on these packages are connected by direct yellow lines; these yellow lines are known as airwires and indicate unrouted electrical connections defined by the electrical schematic.

One of the most important parts of designing a good printed circuit board is proper placement of the various parts. In general, good part placement minimizes board size, limits the amount of intersections between airwires, and places components in logical locations. You can have Eagle recompute airwires and draw shortest airwires by using the **Airwires**  tool (known in previous versions of Eagle as the **Ratsnest** tool). In this particular case, the LM2675 has a manufacturer-defined recommended layout for the application circuit; this is copied from the datasheet and presented in figure 20. Using figure 20 as a reference, place your components in the circuit board; use the jumpers for the input and output specified in the figure.

TYPICAL SURFACE MOUNT PC BOARD LAYOUT, FIXED OUTPUT (4X SIZE)

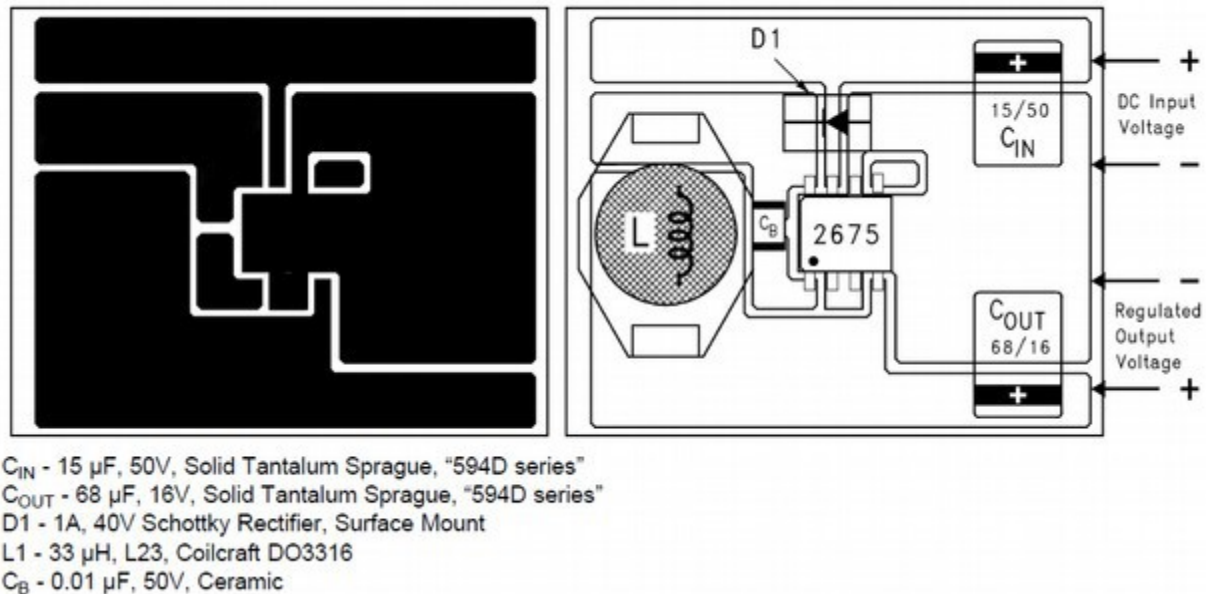




Fig. 20 Manufacturer-Specified PCB Layout

With parts placed in the manufacturer-specified location, we must now create physical connections for the electrical connections defined by the airwires. Select the **Route**  command from the lefthand toolbar. The route command is used to define physical connections in the PCB layout based on the electrical connections defined in the electrical schematic.

There are a number of important options that can be specified for the **Route** command. The first box contains a drop-down selection that allows you to specify the layer being routed on. Routing is performed on the copper layers of a circuit board. These are layers 1 (Top) and 16 (Bottom) in basic PCB layouts. In more advanced multi-layer board designs, additional routing layers exist. To the right of the layer selection box is the bending mode selection identical to the bending modes of the **Net** command in the schematic tool. Due to the orthogonal nature of the manufacturer-specified PCB layout, the right angle bending mode shall be used. In general, the two octagonal bending modes are recommended. Right of the bending modes are some routing options, a bend radius selection box, and miter options; these will not be used for the purposes of this tutorial. To the right of the miter options is a dropdown box to specify the width of the laid traces (specified in the current units of the main grid). To the right of the width selection box are selection options for the shape of the annular rings of vias, as well as selection boxes for the width of the annular ring and inner diameter of drilled circuit board vias. While these options will not be adjusted for the purposes of this tutorial, they are nonetheless extremely useful options that will be used often when designing your own custom PCBs.

Using the routing tool, route physical connections to the various parts in the switching regulator design (Hint: You may need to adjust the main grid size or utilize the alt grid in order to line up some parts

correctly. The **Grid**  command is useful for these purposes). Don't worry about making the traces between the parts thick; this will be dealt with in a subsequent section. When finished, your circuit should resemble the circuit shown in figure 21, below (don't worry if labels and other silkscreen elements overlap; these can be cleaned up once the layout is finalized). Use the move command to

resize the box containing the PCB board dimensions to closely match the dimensions of our PCB.

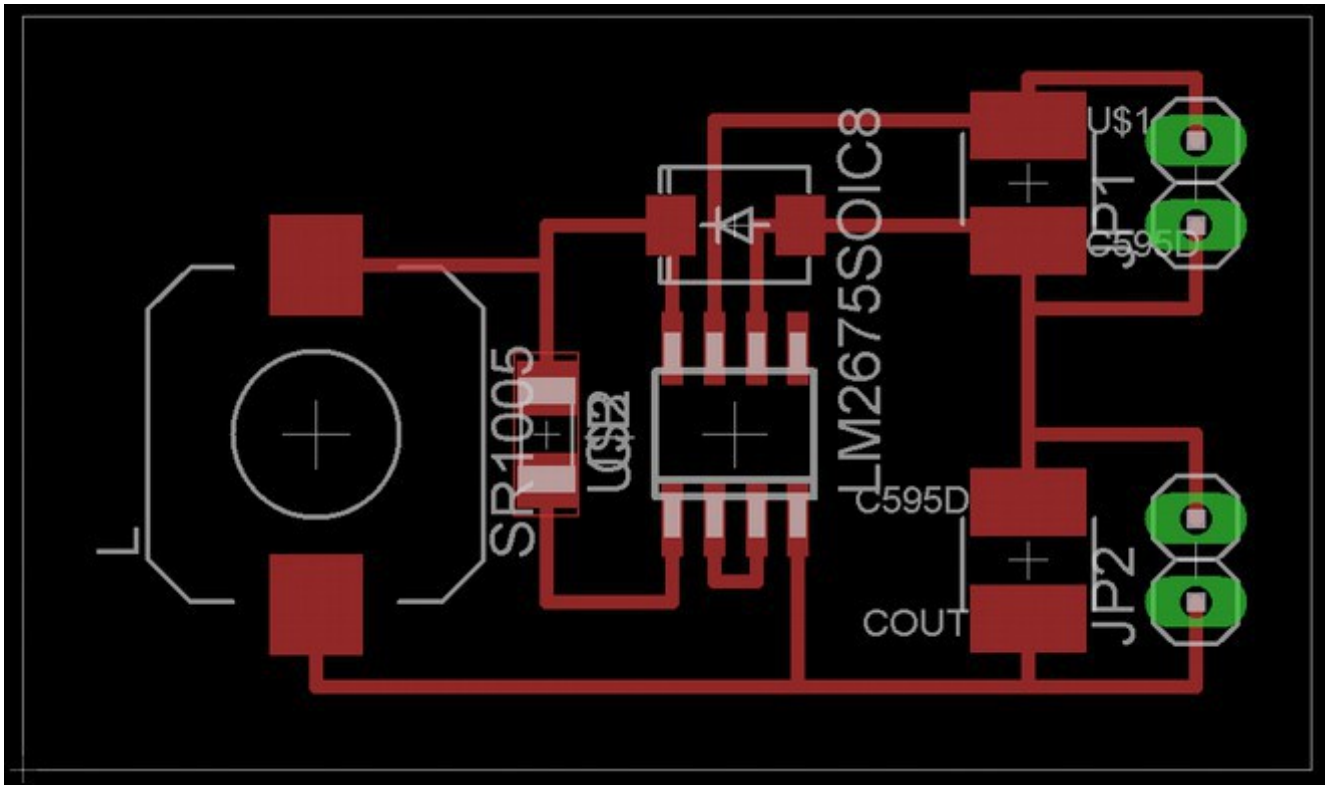



Fig. 21. Preliminary PCB Layout

Although all circuit elements are now placed in their proper locations, we're not done yet! The manufacturer-specified layout requests large copper planes on many of the signal paths. These planes are important to the proper functioning of the analog circuitry within the LM2675, and given that this circuit would be used to supply power to the rest of our design, following the manufacturer's layout is vital. We will be utilizing signal planes, which are introduced below.

A signal plane (also known as a ground pour or a copper fill in some contexts) is a two-dimensional conductive region for propagating an electrical signal. Signal planes are used for a variety of reasons, including convenient routing of signals on a PCB, noise immunity and electrical integrity of signals of interest, thermal elements such as heat sinks, and design aesthetics. To create a signal plane, we will begin by selecting the **Polygon**  tool. Using the polygon tool, draw a rectangle along the top of the board design; make sure that layer 1 (Top) is selected for the polygon. When you have done so your design should resemble fig 22, below.

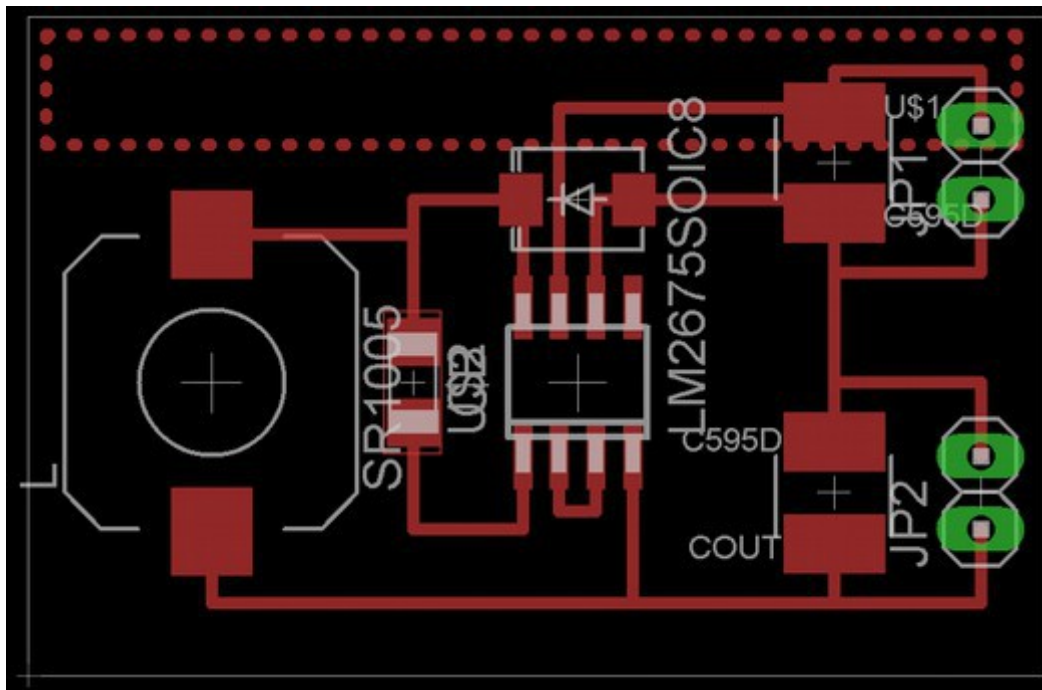


Fig. 22. PCB Layout with Included Signal Plane

The signal plane appears as a dashed line at the top of the box. We can command Eagle to “fill in” this signal plane by using the **Airwire** command which was used earlier to have Eagle compute shortest airwires. Use the airwire command; your design should now resemble figure 23, below:

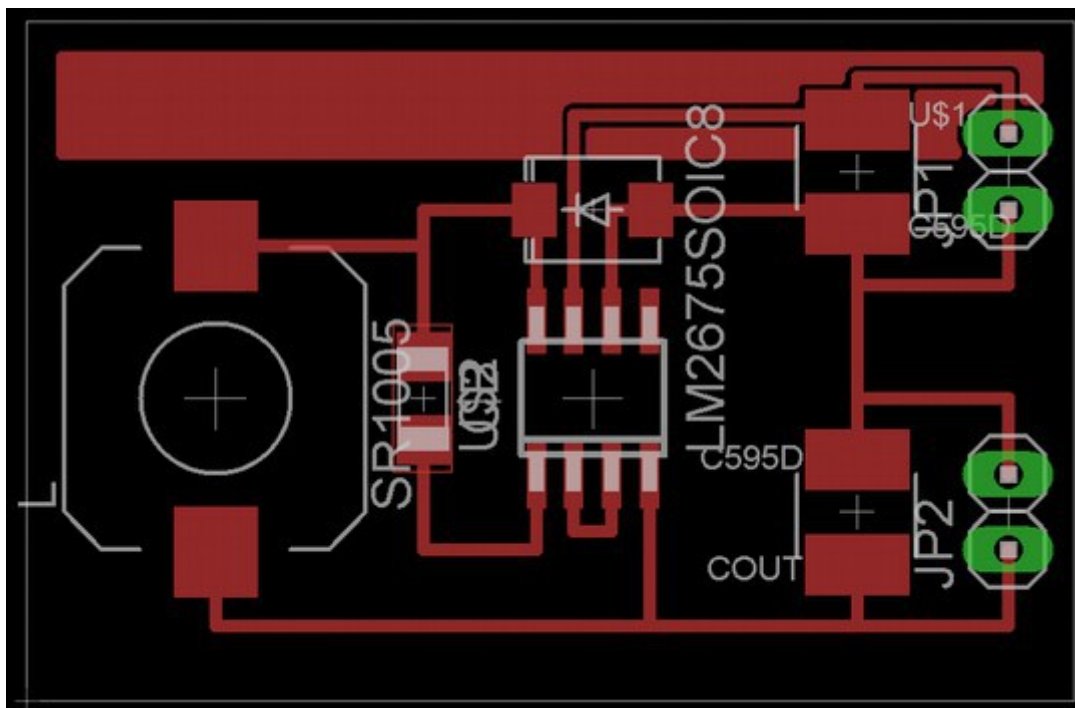


Fig. 23. PCB Layout w/ Incorrect Signal Plane

This signal plane behavior is not correct based on the manufacturer-specified layout of figure 20. The signal plane we created should physically connect to the traces and pads contained within it. In order to fix this behavior to the desired signal plane behavior, we must associate the signal plane to the net we wish to connect it to. Using the **Name** command from the lefthand toolbar, determine the name of the uppermost net on the board. Once the name is known, click on the signal plane and rename it to have the same name as the signal of interest. When done correctly, Eagle will associate the signal plane with the net, filling in the gaps in the signal plane and producing a result similar to figure 24, below.

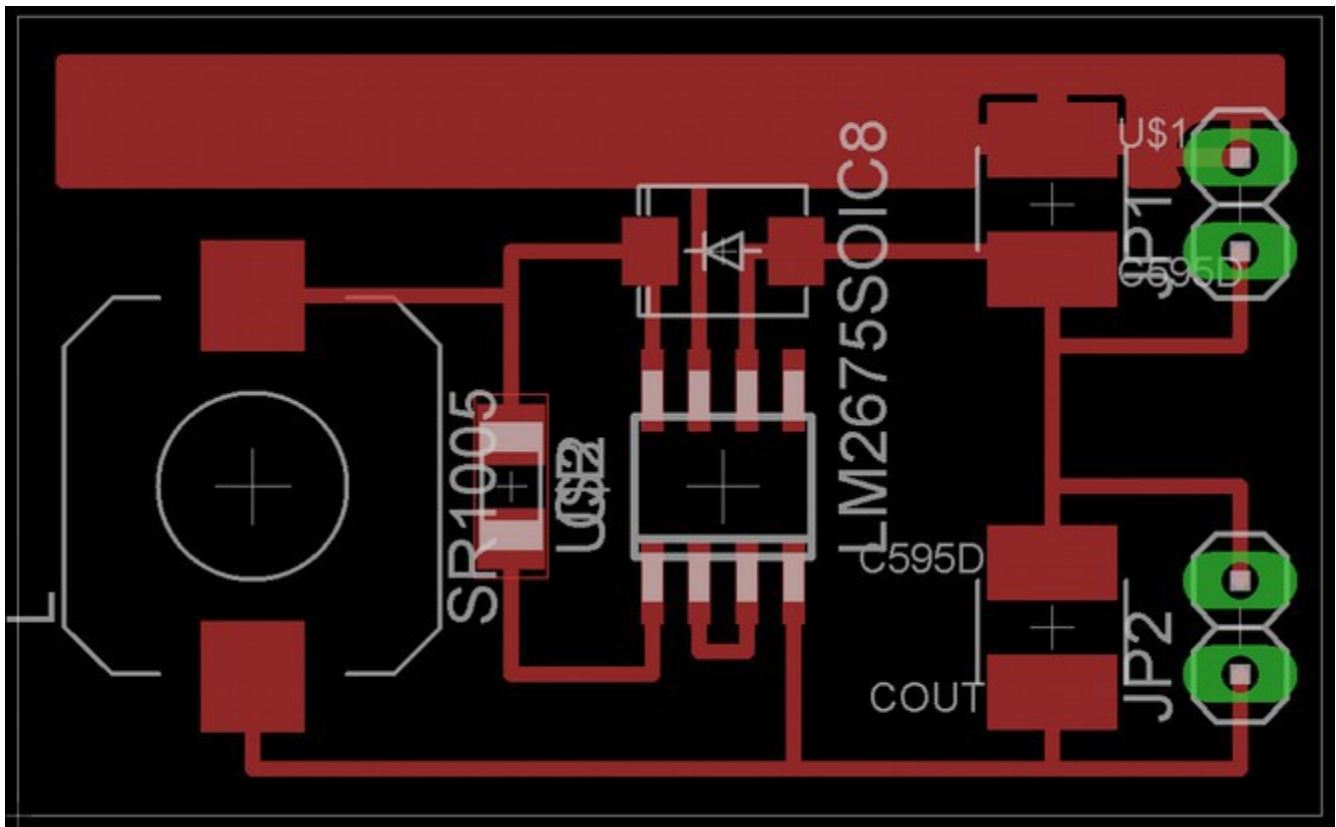


Fig. 24. PCB Layout w/ Corrected Signal Plane

There are 2 additional important properties of signal planes that students will have to use in their PCB designs: rank and isolation. Both properties can be accessed by right-clicking on a signal plane and selecting the **Properties** option from the menu. These are discussed below.

When filling in a signal plane, Eagle naturally maintains a certain distance between the plane and any signal that passes through the plane with a different signal name. The property that specifies this separation distance is known as the “Isolate” property; by default it is set to 0 in every newly-created plane. When a signal plane is filled, Eagle ensures that the plane remains the distance specified by Isolate away from every signal crossing through the plane which doesn't share a signal name with the plane. When filling a plane with an isolate of 0, it appears that Eagle maintains distance from other signals, but don't be fooled! The lack of a specified isolate value can cause the signal plane to “flood” other signals when manufactured by a boardhouse, causing electrical shorts and nightmares for the engineers involved. Therefore, it is good practice to always specify a defined isolate value for all signal planes; 12-16 mils (0.012 – 0.016 inches) is recommended for all ECE477 designs.

Rank is a relatively new feature to the Eagle design suite at the time of this writing, having been introduced in Eagle v6. Rank has to do with the order in which signal planes are filled by Eagle; planes with a lower rank are filled prior to planes with a higher rank. By default, all newly created signal planes in Eagle are assigned a rank of 1. To see where adjusting the rank of a signal plane is necessary, consider figure 25, in which we have 2 signal planes which intersect one another. Plane 1 runs left to right while plane 2 runs top to bottom.

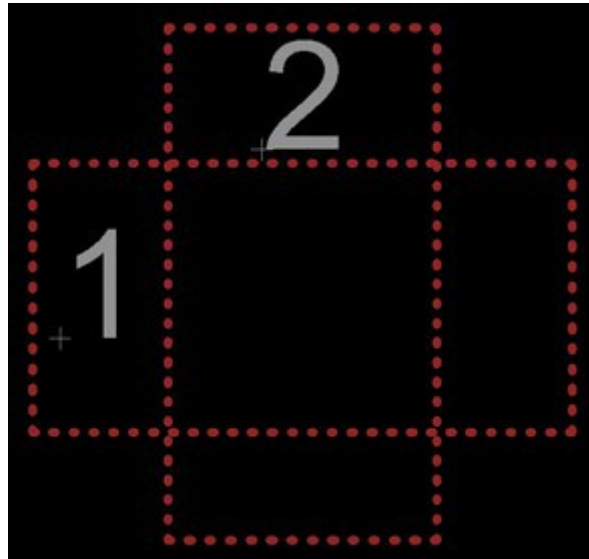


Fig. 25 Intersecting Signal Planes

Running the airwire command, we see that Eagle shorts the 2 planes together when they are given equal rank, which is unacceptable. By adjusting the “Rank” property of the signal planes, however, we can specify which plane is given preference depending on our needs. This is illustrated in figure 26, below.

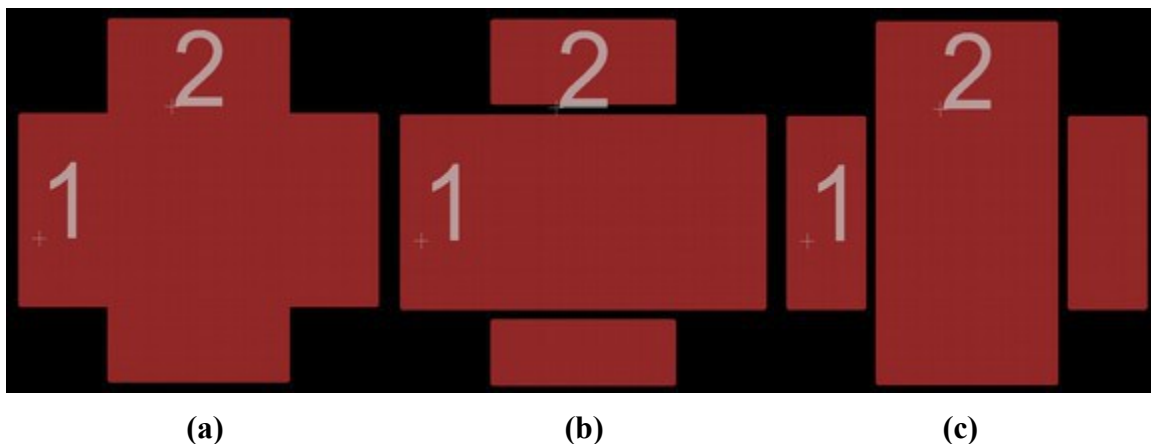


Fig. 26 a) Equal Rank; b) Plane 1 Given Preference; c) Plane 2 Given Preference

With your newly-acquired knowledge of signal planes, add signal planes to your PCB layout to make it resemble the manufacturer-specified layout of figure 20. Don't forget to add the necessary rank and isolation properties to all signal planes. When finished, your design should resemble figure 27, below.

(Note: the square, polygon, and line tools were used to create the signal plane associated with pin “EN”, which the design leaves floating and has no electrical connection).

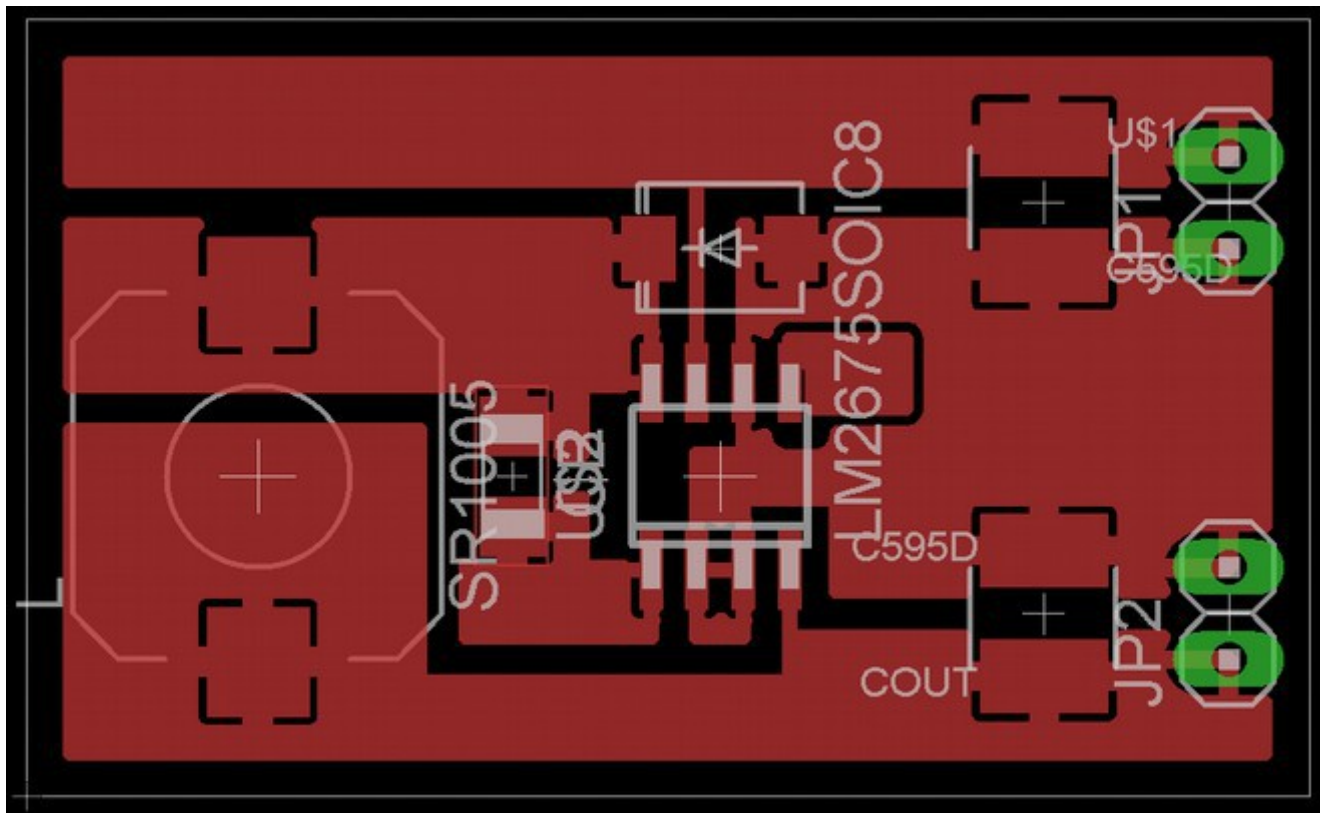



Fig. 27 Completed PCB Layout

Congratulations, your PCB layout is now complete.

7.0 Cleaning Up Designs and Verification

The PCB layout given in figure 27 is electrically complete, but the silkscreen layer could use improvements. Silkscreen labels overlap one another, some run over the top of copper pads, and text is placed on top of areas of the board that will be drilled out for hole connections. In this section, we will make the board presentable for fabrication.

Select the **Smash**  tool from the sidebar, then click on the inductor. When doing so, you will see that the Name and Value labels for the part are given their own crosshairs. They can now be moved, deleted, resized, or otherwise modified without altering the footprint of the inductor. If desired, you can undo the smash operation on a particular part; this can be done by right-clicking on the part and selecting the **Unsmash** option. This will reset the part to its default state.

Smash all of the components on the board, moving labels, changing names to more useful ones where applicable, and getting rid of unnecessary labels. Any labels in the silkscreen layer should be positioned where they don't overlap copper pads, other silkscreen labels, or drilled holes within the PCB. A cleaned up version of the PCB used in the tutorial can be seen in figure 28, below.

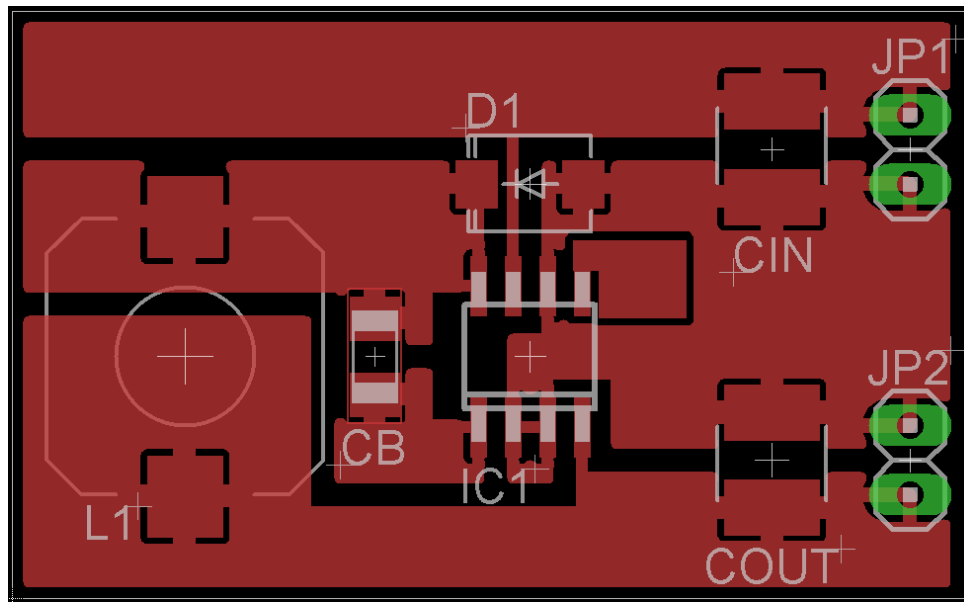



Fig.28. Cleaned up PCB Design


With the board cleaned up, we must now verify the PCB design to check for and correct issues prior to sending the board out for production. The ECE Instructional Labs staff provide useful Eagle files to assist with this process. Download the file “ECE_Senior_Design.dru”, available here:

https://engineering.purdue.edu/ECEIL/Eagle_CAD/Design_Rules/. In the Eagle Layout View, select

the **Design Rule Check**  tool from the lefthand toolbar; a DRC window should appear. This window can be used to specify tolerances for various aspects of the PCB layout for the design. For our purposes, we will load the design file “ECE_Senior_Design.dru” using the **Load...** command. Once the file has been loaded, DRC can be run by selecting the **Check** button. A list of errors can be reviewed and investigated. When running DRC on the tutorial design, an “Overlap” error was thrown on pin 5 of IC1, where the IC pad overlaps with the non-polygon improvised signal plane. This error will not adversely affect the finished PCB, so the error was disregarded.

Once the design has been thoroughly inspected and issues have either been addressed or cleared, continue on to learn how to generate files to send to the board manufacturer for ordering finalized PCB designs.

8.0 Generating Board Files

Printed circuit boards consist of a number of layers and the design process for developing a printed circuit board consists of a number of steps. Most PCB fabrication is performed in large factories known as “boardhouses” or “fabhouses” using automated tools. In order to manufacture PCBs, the tools require files in a specific format, known as Gerber files. Eagle provides a software component known as a CAM Processor which is used to generate Gerber files from PCB layouts. To invoke the processor, select the **CAM Processor**  icon from the top toolbar. The CAM Processor window will appear, similar to figure 29, below.

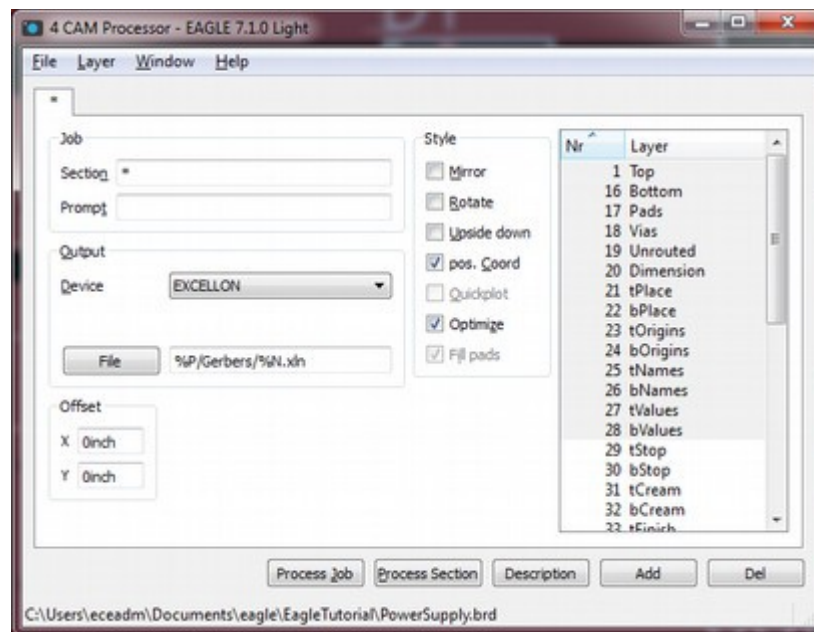


Fig. 29. Eagle CAM Processor

The CAM Processor generates Gerber files according to a list of instructions provided to it known as a “job”. Eagle includes some default jobs that can be run, in addition to jobs that a user is able to create themselves. For the purposes of ECE477, a CAM processor document is provided for students to use for PCB creation purposes. Download the file “senior_design.cam”, which is available here: https://engineering.purdue.edu/ECEIL/Eagle_CAD/CAM_Job_Scripts/. Additionally, create a folder within your Eagle project directory titled “Gerbers”; the CAM script requires this script to be created in order to provide a location for PCB files to be placed.

Within the CAM Processor, select **File** → **Open** → **Job...** and select the CAM file you just downloaded. The CAM Processor should change, adding in a number of tabs for various layers of the PCB that need to be created (Top, Bottom, Solder Mask, etc. etc.). Select the **Process Job** selection in the CAM Processor window. The CAM Processor generates gerber files in the Gerbers directory, which can then be zipped up and sent out to boardhouses for PCB fabrication. More information on that process can be found in the “PCB Submission and Ordering Process” document, available here: <https://engineering.purdue.edu/ece477/Course/Process/process.html>.

9.0 Further Reading

- [1] Cadsoft USA. “Eagle Tutorials”. Available online: <http://www.cadsoftusa.com/training/tutorials/?language=en>
- [2] Sparkfun Electronics. “Using EAGLE: Schematic”. Available online: <https://learn.sparkfun.com/tutorials/using-eagle-schematic>
- [3] Sparkfun Electronics. “Using EAGLE: Board Layout”. Available online: <https://learn.sparkfun.com/tutorials/using-eagle-board-layout>
- [4] Sparkfun Electronics. “Designing PCBs: SMD Footprints”. Available online: <https://engineering.purdue.edu/ece477>

<https://learn.sparkfun.com/tutorials/designing-pcbs-smd-footprints>

[5] Sparkfun Electronics. “Making Custom Footprints in EAGLE”. Available online:

<https://learn.sparkfun.com/tutorials/designing-pcbs-smd-footprints>

[6] Sparkfun Electronics. “Designing PCBs: Advanced SMD”. Available online:

<https://learn.sparkfun.com/tutorials/designing-pcbs-advanced-smd>

[7] Dangerous Prototypes. “HOW-TO: Create Eagle parts with pins that have the same name”.

Available online: <http://dangerousprototypes.com/2012/05/30/how-to-create-eagle-parts-with-pins-that-have-the-same-name/>