**Workflow**

Typically, when designing a printed circuit board (PCB) you will follow a similar workflow every time. This consists of

1. Adding or creating all necessary component libraries
2. Drawing the design schematic
3. Creating the board layout
4. Optionally exporting the board layout to a 3D CAD model

The following sections cover the basics and standards for each of these.

**Libraries**

Libraries are the slightly irritating backbone of PCB design. They do all the backend work of connecting schematic designs, board layouts, and 3D models together. While you can usually get away without having to make your own libraries, it is good to know how to edit them and create your own when online resources come up short.

The first step when dealing with libraries, regardless of if you are going to download or build your own is to create a ‘Libraries’ project in your Fusion 360 team. This is as simple as selecting ‘New Project’ and naming it. This allows you to keep everything in a central place and generally makes it easier to navigate to the libraries themselves when you go to add components to a schematic. You may also find it helpful to pin this project to the top.

A screenshot of a computer

Description automatically generated

*Importing or Using Preexisting Libraries*

What most people find easiest is to simply copy premade libraries off the web and add them to the library folder. This is the simplest, plug-and-play solution. Many websites exist to download part libraries, some are free, and some aren’t. The website I have had the most luck with is SnapEDA (<https://www.snapeda.com/home/>) and so this is how I will explain the import process, however, really any website you use will follow the same process.

1. Find the part you need.  
   This is one of the trickiest parts of the whole process. When you go to the search bar and type in the name of your component, you are likely to get tens if not hundreds of library options.

A screenshot of a computer

Description automatically generated

From here you want to filter the search by the package type you are using, which can be found in the datasheet. This way you don’t buy the board and realize your component doesn’t match (yes this is speaking from experience).

A screenshot of a box

Description automatically generated

After filtering there will probably still be a few options left. The next thing to check is the symbols on the far right. These indicate whether the part library has the different schematic and footprint files. The terms symbol and schematic are used interchangeably, these refer to how the part will look on the actual schematic and where the pin connection points are located. Similarly, board layout, footprint, and land pattern are also used interchangeably and refer to how the copper is laid out on the physical board.

A screenshot of a computer

Description automatically generated A close up of a chip

Description automatically generated

A useful final check before selecting the one to download is ensuring the part name exactly matches the part you are looking for. If there are still multiple options, I have also found it useful to go with the option that has a datasheet available, which allows you to make sure they have referenced the same one as you.

1. Download the prebuilt library.

To download the library, simply click into the part and select ‘Download Symbol and Footprint.’ From here many options will pop up, at the moment Fusion 360 supports at least ‘EAGLE’ and ‘Autodesk Fusion 360’ formats, although there’s really no reason not to use the Fusion 360 format since it’s available.

A screenshot of a computer

Description automatically generated

1. Import the library into Fusion 360.

Once you have the library files downloaded from your website of choice, the final step is to import it into Fusion 360. Migrate into your ‘Libraries’ folder and simply upload it through the ‘Upload’ button on the top left.

A screenshot of a computer

Description automatically generated

Lastly, if you want to move the library files, symbol, footprint, and/or 3D model, into an existing library, first open the existing library, select ‘New Symbol’, ‘New Footprint’, etc., and choose ‘Import…’. This will bring up a dialog window showing every available library which you can scroll through until you find the desired library, and then select the file to copy it in. You may, however, need to create a new ‘Component’ using this method, which is explained below. After this you can freely delete the old library if you so desire.

A screenshot of a computer

Description automatically generated

*Creating or Editing Libraries*

If the specific component you’re using isn’t available online, you want to change some formatting or measurements, or you’re more of a do-it-yourself kind of person, then you’re going to need to edit your own libraries. Creating and editing libraries isn’t difficult, but it does take some getting used to. If you simply want to edit an existing library, I recommend just glancing through the following steps until you find what you need. They are pretty tailored for beginners in the process of creating their first library and can be a bit extraneous.

The first thing you will want to consider is if there is a particular format, or guidelines you need to follow. Many companies, clubs, and people have their own fonts, spacing, and styles they like to use to keep everything consistent throughout. Indeed, even if you are working on your own, it can still be helpful to create some personal guidelines so it’s easier to understand your schematics and boards if you come back after a while.

These guidelines can take many forms, though some common things to consider are:

* It’s often recommended, or required, to indicate component polarity in some way.
* Do you want to outline components so they are easily visible on the physical board?
* What font and text size will be most readable, yet small enough to not take up much space?
* Should the component be centered on the land pattern or does the origin fit better elsewhere?
* How much information is too much on the schematic?

Once you have found the format you need to follow, or are ready to build your own, it’s time to start creating a library.

1. Assuming you have a dedicated library project already, migrate into it and go to File -> New Electronics Library.

A screenshot of a computer

Description automatically generated

This will open up a new library window, but in order for it to actually show up in your Libraries folder, you will need to save it. When saving libraries, I recommend saving, and using, them in a way that reuses as many footprints and symbols as possible. By this I mean, many integrated circuits use similar footprints to each other such as SOT-23, TSSOP-28, WSON, etc., of which almost no other components use. Similarly, there is no reason to recreate the same capacitor symbol more than once. For these reasons, many people find it most straightforward to create libraries for similar components, such as ICs, resistors, capacitors, etc.

1. You can create symbols, footprints, and 3D models in pretty much any order. However, since the design process typically flows first through datasheets, I find I end up following the path of footprint -> 3D model -> symbol, since the footprints and 3D models are often given at the bottom of the datasheet. Something to note is that 3D models are only required if you want to export your board later as a CAD file, otherwise they serve no practical purpose. To begin creating any part of a library first select one of these four buttons for ‘New …’. I will begin by explaining the process of creating a footprint.

A screenshot of a computer

Description automatically generated

1. Creating a footprint is pretty much as simple as copying the dimensions from the datasheet and doing a little math so your origin is in the correct place. The first step is going to be creating the surface mount (SMD) or through-hole (PHT) pads, which you can access through the buttons shown below.

A yellow and black symbol with blue and red text

Description automatically generated with medium confidence

Punch in the dimensions and select the shape you want to use. Something I find can help with actually placing the pads is to find the spacing between them on the datasheet and change the grid size to match. This way you don’t need to do as much manual math. To change the grid size, select the ‘Grid Settings’ button.

A screenshot of a computer

Description automatically generated

If you do need to edit any of the dimensions or values for anything, either right click on it and select ‘Properties’, or left click to select it and open the Inspector, which in my case is on the right of my screen. The inspector can also be used to edit multiple things at once.

A screenshot of a computer

Description automatically generated

One last thing you may run into is needing custom sized through-hole pads for obscure shaped components. The roundabout way of doing this in Fusion 360 is to first create an outline for the area you want drilled out, with a 0-thickness line on the ‘Milling’ layer. This specifies the inside of the pad. Then add an actual through-hole pad on top of it which is used to specify what sized drill bit to use when drilling it out. In general, set it to the width of the hole. Lastly, using the ‘Top’ and ‘Bottom’, and ‘t/’bStop’ layers, add polygons representing where copper should be placed for the pad, and respectively where the silkscreen should not overlap. Appendix A describes how to draw polygons. An example of what this looked like for a Micro USB footprint I made can be seen below. It looks a little odd, but Fusion 360 recognizes what is happening when you export the board for ordering.

A screen shot of a game

Description automatically generated

After the pads are all placed, the last important thing is to name each of them. This can be accomplished by selecting the Name icon next to the ‘… Pad’ buttons we used earlier. When naming them, consider following the numbering or naming scheme laid out on the datasheet so you’re less likely to mess things up later. If there are multiple pads for the same signal, for instance ground, you can use the @ symbol and numbers starting from 1 to link them together.

Next, it’s really up to you and the format you’re following if you want to add anything else. Some things I have found helpful are to add an outline that either goes around the pads or shows the physical shape of the device, so it’s easier to spot on the board. Also adding text with the ‘>NAME’ identifier will cause it to be replaced by the component number when added to the board. An important note when adding text and lines is that when changing the layers to use different colors or functions, only the t/bPlace and t/bNames layers will actually show up on the physical board unless you change some settings. Similarly, the t and b reference that layer showing up on the top or bottom of the board respectively. Layers like t/bKeepout and t/b/vRestrict can also be used to stop components from being placed too close to each other or traces from running through that part of the board respectively.

A screenshot of a computer

Description automatically generated

1. After creating a footprint, if you want to link it to a 3D model, first find it in the Content Manager, and select ‘Create New 3D Model.’

A screenshot of a computer

Description automatically generated

This will open up the original Fusion 360 modeling software with an extra layer showing your current footprint. From here you can either model the part on your own following the dimensions on the datasheet, or, if it’s a common shape, use the nifty ‘Package Generator.’

A screenshot of a computer

Description automatically generated

This will open a menu that has generators for all the common package types. All you need to do is select the one that most closely resembles yours, put in all the dimensions from the datasheet, and it will automatically generate and align the model to the best of its ability.

A computer screen shot of a black rectangular object

Description automatically generated

When saving your 3D models, I suggest adding them to a folder titled 3D Models within your Libraries project, or else it can get a bit cluttered. Additionally, you may find it convenient to put the name of the library at the start of the name for the 3D model to keep it neatly organized. Lastly, since we added the package through the footprint, they are automatically linked. If you have a 3D model already, you can use the ‘Attach Copy of Existing 3D Model’ option to link them.

1. The third part of the library trio is creating the symbol. Your guidelines likely list common symbols to use or else where to find them. If not, there are a few different approaches you can take. One would be to draw the circuit schematic symbols where you can, such as passive elements, op-amps, transistors, etc., or you can just use boxes to represent everything and identifier tags to specify what it is. I tend to use a mix of both where the large ICs are just blocks with pins, but the small ICs with internal op-amps, passives, and transistors, I usually draw out.

Regardless, start by sketching out what you want the symbol to look like, keeping in mind how many pins you will need to fit. This can be accomplished with the ‘Line’ tool. Keep in mind, the ‘Grid Settings’ have been moved to the bottom for the symbols, because usually the default settings are best.

A screenshot of a computer

Description automatically generated

Next, use the ‘Pin’ tool next to the ‘Line’ one and add the correct number of pins. Similar to the pads, name these pins so that you account for all of them. You can then go into the pin’s properties and change the ‘Visible’ setting if you don’t want to see the name, the pad it will be connected to, or both. Additionally, the pins can have different ‘Directions’ which are displayed. I only know of three options that actually do something and those are:

* + Not connected (nc) – This puts a visible x on the pin
  + Passive (pas) – This only used for passive components
  + Supply (sup) – This is reserved for supply symbols which are discussed in the Extra Considerations section

As far as pin placement goes, you don’t have to exactly match the pads, and I would usually err on the side of readability and usability over putting the pins in the same location as they physically appear on the package.

A diagram of a circuit

Description automatically generated

The last thing you’ll want to consider for the symbols is adding different identifiers. First, adding the text ‘>NAME’ on the ‘Names’ layer will link it to the same component number shown on the board. Secondly, adding ‘>VALUE’ to the ‘Values’ layer will cause the text to be replaced with the actual component name. Lastly, you can add anything in full caps after a > symbol on the ‘Info’ layer to specify additional information, or attributes, you want displayed. These are actually declared in the next step. An example of my layout for a resistor is shown below along with its implementation in an actual schematic.

A diagram of a graph

Description automatically generated with medium confidence A diagram of a circuit

Description automatically generated with medium confidence

1. The last part of building a library is to fully link everything together into a component. Components are what you actually add to a schematic, and they hold the physical data of the part while linking them to the footprints added to the board. When you create a component, there are two ways of going about naming them. The first option is to create a generic component, say, ‘Resistor’, which houses all the different types of resistors you will use. Otherwise, you can create a new component for each variant. The tradeoff is it’s easier to search component variants when they are separate, but it’s more compact and structured otherwise. I will be explaining how to create generic components because they are more complicated.

The first step after creating a component is to add a symbol through the ‘Add’ tool. You are also going to want to set a prefix in the bottom right corner. These are inserted before the component number when added to a schematic for easier understanding. Appendix B contains a list of the most common prefixes and their use cases.

A screenshot of a computer

Description automatically generated

From here you can choose a symbol from the library to use for the component. Next, under ‘New’ in the bottom right corner, add a package from the library using the ‘Add local package’ option. This will only allow you to add packages with a number of pads equal or greater to the number of pins on the symbol.

A screenshot of a computer

Description automatically generated

This will add the footprint, and package if you created one, to a list of ones the component can use. The next step is to double click the footprint name to map the pins to the pads. You can map multiple pads to the same pins, but all pins must be mapped before the component can be used. This is an important step to double check against the datasheet because getting it wrong can ruin an entire board.

A screenshot of a computer

Description automatically generated

For generic components you can add ‘Variant’ names to the footprints, which will show up as a drop down when adding components. You can change the name of the variant by right clicking on it and selecting ‘Rename.’ These serve to allow one symbol to map to multiple different package sizes. For instance, one resistor symbol, and component, can be used for any size of surface mount resistor. An example where I did this is shown below.

A screenshot of a computer program

Description automatically generated

The last layer to this is that each variant can have multiple attribute sets. This was a feature of Eagle which was migrated into Fusion 360 but has yet to be promoted thus far. To access a variant’s attribute sets type ‘technology’ into the command line in the top right with the variant selected. This brings up an interface where you can add new attribute sets to the variant by either typing them in, or selecting ones used by other variants.

A screenshot of a computer

Description automatically generated

The practical use case for this much nesting is to consider a generic resistor component. All resistors share the same symbol. However, you can have different footprints for different types or sizes of resistors. Lastly, each footprint will have different resistance values you can use with it. For example, I’m adding a 1608 sized 10k resistor to the schematic. This way we don’t have to make a separate component for each size and value of resistor we have.

This brings us to the final part of components which are attributes. You can access attributes through the ‘Attribute’ button.

A screenshot of a computer

Description automatically generated

Attributes let us define the values of the different identifiers we created for the symbol. In the interface that appears you should see a list of all the attribute sets you added to the selected variant. From here, select the ‘New’ button to add the identifier names used in the symbol, e.g. ‘VALUE’, ‘VOLTAGE’, etc. Note that if the ‘VALUE’ identifier isn’t specified, it will be replaced by the component name. Lastly, double click in one of the empty slots to set the identifier values for the given attribute set. The variable or constant selection under the value field sets whether the value is the same or different for all attribute sets.

A screenshot of a computer

Description automatically generated

*Extra Considerations*

There are a few niche things that are helpful to know about libraries but aren’t common enough to include them in the official section. One of these is creating just symbols which don’t link to something on the board.

To create symbols which will only live on the schematic for either organization or clarity, first draw the symbol itself. If this symbol is a voltage reference, add a pin to it using the supply (sup) direction. This internally links all traces connected to one of these pins together and renames them to the name of the pin. If this symbol is not a voltage reference, I recommend not using any pins.

A screenshot of a computer

Description automatically generated

Next, create a component and instead of adding a package along with the symbol, define a new attribute called ‘\_EXTERNAL\_’. This specifies to Fusion 360 that the symbol does not reference any board land pattern or package.

A screenshot of a computer

Description automatically generated

*Fusion 360 Quirks*

Fusion Teams

**Schematic**

Assuming you have all your libraries in place or are using your company’s / club’s library base, the first step to creating a PCB is the schematic. Because this tutorial is about creating a PCB, I will assume you know the circuit you are building and simply need to convert it to a physical board. In this case building the schematic should be as simple as digitizing the circuit you already have.

*Creating a Schematic and Block Diagram*

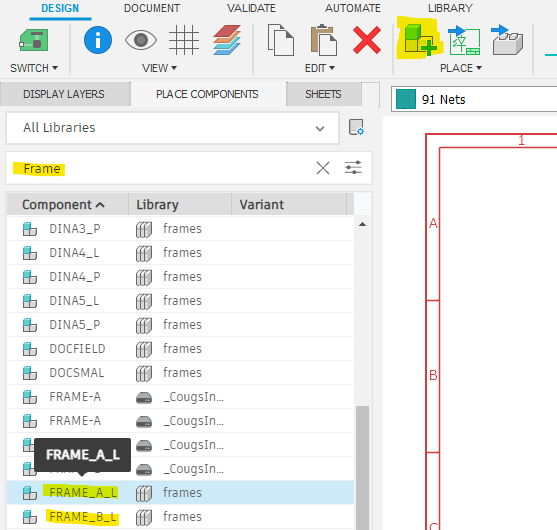
To create a schematic, either enter the default untitled Electronics Design tab or go to the File icon -> ‘New Electronics Design’. Both of these will take you to the same blank tab with only four options. Select the New Schematic icon to get started. The way these electronic designs files work is they automatically link all schematics and PCB layouts together, meaning you can create the PCB file now or later and it will still automatically fill with the necessary components. Similarly, referencing a different schematic or PCB document will just add those parts to the project.

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

Once you’ve created a schematic file, before adding all the components, you’ll want to add a frame to each sheet. This is mostly for printing it later but can also be helpful for keeping everything together. To add a frame, either select your company or club’s default one, or navigate to the Place Components tab and type ‘Frame’ into the search field. This tab is typically located on the left side of the screen but can also be toggled with the Place Component icon in the DESIGN -> Place menu.



Frames sizes are defined by letters which reference their physical size. ‘A’ refers to a standard 8 ½” by 11” piece of paper, while successive letters can be though of as doubling the size of the previous. To add the frame double click it and place it on the sheet. I find it easiest to line up the frame’s origin (+ symbol) with the origin of the sheet, then all the coordinates will reference the bottom corner of the sheet.

A line drawing of a rectangle

Description automatically generated

The last step before adding components is to create a block diagram on the first sheet. While this isn’t necessary and has no effect on the schematic or board itself, it can be helpful to yourself, others, or even your future self in understanding what the circuit does and/or what sheets to reference for each part.

A block diagram is pretty self-explanatory, make it look like any other block diagram you would make. Some things that might be helpful are adding the connection protocols used, page numbers for what sheet the block is implemented on, and staying general enough that the block describes the function, not necessarily the exact part. An example of a block diagram I made recently for a personal project can be seen below. Note that the signature block is not part of the diagram I just put it on this sheet.

A diagram of a machine

Description automatically generated

*Adding and Connecting Components*

To add components first create a new sheet (Sheets -> Right Click the empty space -> New) and add the frame. Then navigate to the Place Components tab and start adding in everything for your circuit. Some ways to speed up this process would be filtering for the library you want from the drop down. You can also copy and paste selections.

One thing to be aware of while adding components is that while you can select variants from a drop-down menu on the part itself, but if you added attribute sets to an individual variant, say different resistor values, you need to select the part and navigate to Attributes to set it, before double clicking to add the part in.

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

After adding in the components you need for your circuit schematic, start connecting everything using the DESIGN -> Connect -> Net icon. Nets will create a noticeable dot at three- or four-way connections while continuing straight through each other when they aren’t connected.

A screenshot of a computer program

Description automatically generated

One last thing to be aware of is if you are connecting between sheets or within the same sheet without a physical connection, you have to name both nets the same name. The Name icon is located right next to the Net one and is achieved by clicking on the net and setting the name. Sometimes you may get a notification letting you know you are merging two nets through a name, which you can just ignore or select Yes. Some people also find it helpful to add tags that call out where on a sheet the net is connecting to. Typically, these tags specify the direction of signal flow and the (Sheet, Letter, Number) designation it connects to, where the letter and number refer to the respective section designated on the frame. An example of this can be seen below where the top net connects to the E4 section of sheet 2, though in this image the component origins make it a bit hard to read. Naming nets can also be helpful when you want to understand the connection during the board layout phase.

A diagram of a micro sd connector

Description automatically generated

*Useful Tips and Tools*

Making schematics can sometimes get messy so below are some tips and tools I’ve found helpful.

* You can freeform select by changing the selection type under DESIGN -> Select, to ‘Group with Polygon’. Now you just need to click in a shape around parts to select exactly what you need.

A diagram of a circuit

Description automatically generated

* Adding lines around parts of the schematic and adding descriptors can help with understanding and connect the block diagram to the circuit. This can all be achieved through the DOCUMENT menu.
* Adding text labeled ‘Design Note:’ or ‘CAD Note:’ can help you remember where the inspiration for a circuit, or equation for the component value came from or specifics for the board layout.
* If you update a library, you can use the LIBRARY menu to refresh the schematic to match.

**Board layout**

After creating the schematic, either switch to the board layout or create a new one. The physical board layout is where things can get a little tricky. However, in general you can just follow the recommended layout on the IC documentations.

*Initial Steps*

Before setting up all the components, it is best to define the required dimensions of the board. This is accomplished by drawing them on the ‘Dimension’ layer with zero width lines. If you don’t have any required dimensions, just keep everything compact and set the dimensions at the end to a small value that still holds everything to reduce the cost. Holes can also be created using the DESIGN -> Place -> Hole icon but they may throw an error if placed too close to an edge.

The first step for laying out the board is usually following the ‘Recommended Layout’ section of the datasheet. If the required dimensions of the board are very tight you may need to slightly adjust these to make them more compact. In general, though, a good start is ensuring the decoupling capacitors are close to their respective pins and traces are able to be kept short.

***You can also change the setting for pushing things out of the way, airwires***

Additionally, due to the nature of traces being very thin pieces of metal, they can only practically support a specific amount of current. To determine the minimum trace width necessary for each trace based on the amount of current which needs to flow through it, I find it easiest to use an online calculator such as the Advanced Circuits (<https://www.4pcb.com/trace-width-calculator.html>) one. This calculator looks like the following, where you can input many different settings, however the most important ones to pay attention to are inputting the correct current, thickness of the copper on the board, and then reading out the width for what type of layer you’re using. In this case ‘External Layers in Air’ refers to the top and bottom layers. Below is what this typically looks like, where I have shown that the default width set by Fusion 360 (.15mm) can support up to 600mA of current.

A screenshot of a computer

Description automatically generated

A few other considerations to keep in mind while setting up the board layout are using copper pours for high current paths and to help with thermal management. Copper pours are achieved through polygons in Fusion 360, which I have outlined in Appendix A. These can help act like heat sinks to dissipate heat from components, and as a way of reducing trace impedance for high current paths. The following sections explain these in greater detail. For reference, below is an example of a datasheet’s recommended layout and my implementation. As you can see it is similar for the decoupling capacitors but is tailored to my specific application. In this instance I used a ground plane on the bottom layer and didn’t have much current flowing through any trace.

A diagram of a computer

Description automatically generated A computer screen shot of a circuit board

Description automatically generated

*Grounding and Return Current Paths*

Where things get interesting in board layouts is when you have to start considering component grounding and signal return paths. The mindset shift you have to make is realizing that, while on a schematic the ground is simply a voltage reference, in reality ground acts as the return path for every signal and current. These don’t just transmit in one direction, they create a loop, think KCL, there needs to be the same amount of current flowing into and out of a node.

On its face, this seems easy, just draw the equivalent return path for every power and signal trace. And, while this can work for very simple designs, it gets extremely complicated for even mildly complex circuit boards. This is where ground planes come into play.

The premise of ground planes is to create a low impedance return path for every signal and power supply trace on the board. In a sense, this is mimicking the thought process of a ground symbol on a schematic, but it’s not that simple. This brings us to the first interesting topic, analog and digital components.

Analog and digital circuitry have very distinct properties that don’t mesh very well together. Analog circuits typically require high gain, low noise operation, partnered with signal filtering and bias conditions which need stable ground references. Digital circuits on the other hand, consist of thousands if not millions of transistors, constantly switching and producing transients that source and sink current from the decoupling capacitors. What this leads to is analog parts and signals which are very susceptible to noise, and digital signal traces which can appear as high frequency noise and induce similarly fast switching currents in the surrounding ground. For these and many other reasons, it is recommended to keep analog and digital components on opposite sides of the board.

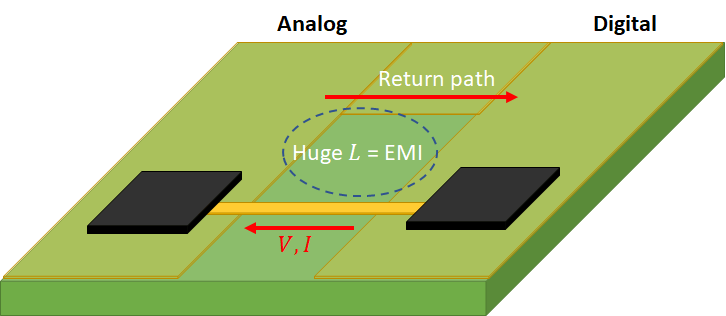
A lot of people think that this means you need a separate ground plane for both, to reduce noise from leaking into the analog components. This actually can work very effectively in some instances, and there is a whole topic about this called ‘Star Grounding,’ but again, for more complicated circuits, it starts to break down. Some careful thinking reveals why this doesn’t work. First of all, the grounds still need to connect somewhere, so they have the same reference voltage. In star grounding this is called the star point.

However, what happens when you need to connect signals between the analog and digital sides? You could use an ADC/DAC. These seem promising because they have pins labeled AGND and DGND. However, these pins refer to the internal grounds of the IC, which do need to be separated due to their close proximity, but they are specified as needing to be tied together outside of the IC to ensure an equivalent voltage reference. So you could put the star point here, assuming you only need one signal trace between the two sides. But what about the power traces? They need a return current path as well.

As you can see, while it is possible to create separate ground planes, it leads to a long list of contingencies and only really works on a board-by-board basis. Additionally, beginning at frequencies greater than 1-10kHz the return current will actually flow back along the same path as the trace, just through the ground plane, meaning any signals routed over breaks in a ground plane will lead to capacitive coupling and a lot of electromagnetic interference being generated.

A diagram of a medium frequency

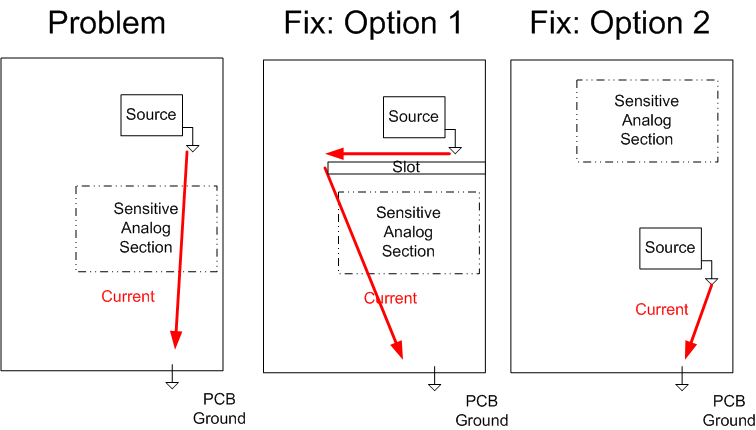
Description automatically generated



So, in general it is easier to use a single ground plane and pay attention to the return current paths, making sure they don’t cross and create interference. For a single ground plane, the general layout recommendation is to have analog components on one side, digital on the other, and then the power supplies in the center on the top or bottom. This keeps all the current paths as separate and possible while allowing connections between the three that don’t interfere with one another.

IMAGE of recommended layout (bad, better, best)

The last consideration with grounding is when it comes to high current traces and return paths. High current can lead to significant voltage drops, even in low impedance ground planes. This means that the area around high current return paths will experience a lifted ground. The reason this matters is that any components needing a ground reference placed along this path will be referencing a voltage higher than other components. One way to control these paths around sensitive components is to introduce a slit in the ground plane to shape the return path. Note that this only works because it is high DC current and there are no high frequency signal traces over the slit.



*DRC*

After finishing the board layout, be sure to check the design rule check (DRC) by typing ‘drc’ into the command line.

A screenshot of a computer

Description automatically generated

This will open the DRC interface which lets you specify the different tolerances that are required. The default settings are typically overkill and will work in the vast majority of cases. Selecting ‘Check’ will then list every airwire that wasn’t routed correctly, any errors that were found, and any overlapping pads. If any of these were intentional, you can select the error and choose ‘Accept’ and it will no longer flag that instance. This can be useful for connectors that have to be close to the edge of a board, or footprints with pads that have to be really close together or overlapping.

IMAGE of DRC and the accept button

Lastly, once everything is valid and you are confident the layout and circuit are correct, typically this involves multiple board reviews from others, it is time to export the board and order it. This is explained, and can be found, in Appendix C.

**Output 3D PCB**

**Appendix A**

**Polygons**

**Appendix B**

**Standard Component Prefixes**

B: Board, Module

BT: Battery

C: Capacitor

CN: Capacitor Network

D: Bridge Rectifier, Diode, Zener

DS: Signaling Dode (LED)

F: Fuse, Resettable Fuse

FB: Ferrite Bead

FD: Fiducial

FL: Filter

H: Hardware

J: Jack, Connector, Combination of Connectors, Row of Headers, Plug

JP: Jumper

K: Relay

L: Inductor (not including ferrite bead), Coil

LOGO: Artwork, Logo

LS: Loudspeaker, Buzzer

M: Motor

MK: Microphone

Q: Transistor (all types)

R: Resistor

RN: Resistor Network

RT: Thermistor

RV: Varistor

SW: Switch, Push Button

T: Transformers

TC: Thermocouple

TP: Test Point

U: IC, Inseparable Assembly

V: Vacuum Tube, Tube

VR: Variable Resistor (potentiometer)

Y: Resonator, Oscillator, Clock Generator

**Appendix C**

**Exporting Gerber Files and Ordering**