Fusion Electronics – A Library starter aid and a 'How to' help sheet

I Introduction

Fusion Electronics (FE) is a module within Fusion that allows the design and production of printed circuit boards (PCBs). It is available to hobbyist users (with some slight restrictions on use) and unrestricted to licenced users. Having this module within Fusion has many features and advantages. For instance, after completion of the PCB design, a custom designed enclosure can be created in the Fusion 3D modelling section that exactly accommodates the PCB geometry. Alternately the reverse might be true where the enclosure is defining the PCB geometry. Once a PCB is completed FE gives a data export facility that delivers all the necessary files for third party manufacture of the finished PCB.

The components that will populate the PCB will have representative images that will appear on the circuit diagram, footprints to show the pad geometry needed on the PCB and a 3D model of the part. This information is contained within a library that can contain one or more components and each component can have each of these constituent parts or variants of these. Components not immediately available within the FE libraries can be downloaded from third party web resources or can be custom created by the user with the Fusion Electronic and Fusion Modelling resources.

This integration of the PCB design module within the overall Fusion software package is a very powerful tool for the electro mechanical engineering world.

II Where to find an initial overview of Fusion Electronics ?

I would recommend two useful YouTube resources. The first one is a seven part series produced by Autodesk presented by Jorge Garcia and the second one is a three part offering by Will Donaldson. There is a lot to absorb from these two presentations and you will almost certainly have to view them a number of times.

Autodesk

https://www.youtube.com/playlist?list=PLmA_xUT-8UlL80Xm8Gxz98YNum3I9GInr

Will Donaldson

https://www.youtube.com/watch?v=NITJZfhjppI&t=1s

III My initial experience with Fusion Electronics

After watching both these video series I was able to design a PCB and export the necessary files from FE to allow me to mill and drill the copper traces on my Tormach CNC machine. I used a program called FlatCAM to convert the exported FE manufacturing files into code that the CNC could interpret. The FlatCAM process will not be covered in this document but some relevant articles and experiences are documented on my blog.

The qualifier to my initial success producing a working PCB was that all the parts I needed for the PCB were available as ready to use library parts.

As my designs became more complex this was no longer the case and I had to delve into library creation and get a deeper understanding of the processes and concepts involved.

I posted a plea for a better understanding on the Autodesk Fusion Electronic forum and received assistance from Jorge Garcia at Autodesk and a Fusion Electronic user called Chuck Todd. You will find their responses as Annexes to this document. What follows is my best understanding of the process of creating library parts using the knowledge accumulated from these various sources.

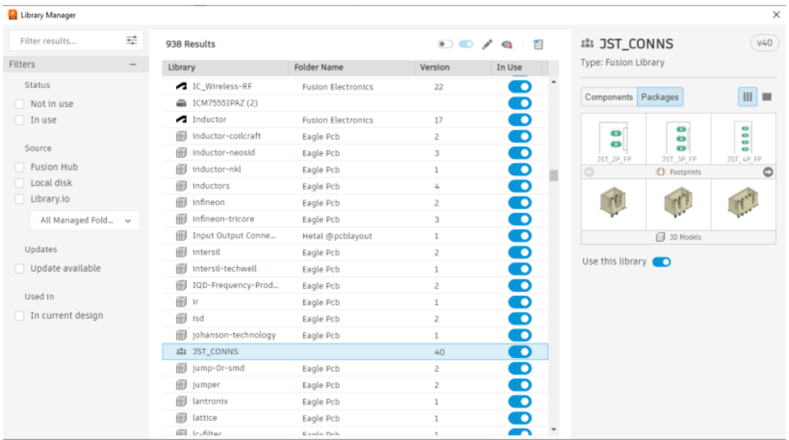
IV Fusion Electronics Libraries

Components are stored in a Fusion Electronics libraries. A library is a container. A library is stored in the cloud and is only available to the Fusion Hub to which you are a member.

One thing that totally passed me by was that the title of the library can be anything because it is just a container file title. Some of the tutorial videos name the library with the same name as the component. This suggests that the library and the component are one and the same and this added to my confusion. In practice the library title can be a generic name (Caps, Resistors, FETs etc) and you can have multiple components of perhaps a similar function created and stored within such a single library. You could also store a mix of different types of components that are perhaps your regular 'go to' parts when starting a new design. Library contents are totally flexible in this respect.

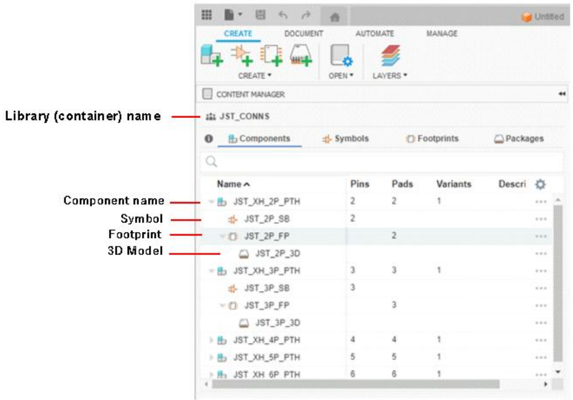
V Example Library

I use a lot of JST HE connectors. I have created a library called JST_CONNS and I have 2,3,4,5,6 pin versions of these JST connectors stored in this library. When I search and find my JST library in Library Manager, I get the view as shown below. On the right hand side area I see small images of each component in the library. The Packages tab shows the footprint and the 3D model and the Components tab shows the symbol for each item. These images are not 'clickable' but are simply there for viewing the library contents so you know you are looking in the right place for your needs.



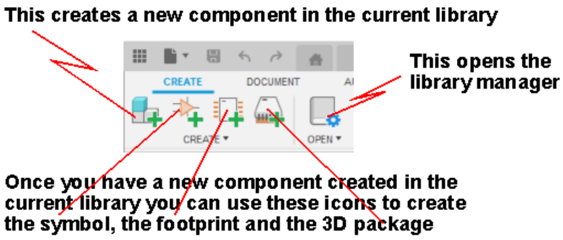
While in this view within Library Manager, if you highlight and right click on the JST_C0NNS highlighted blue line and select Edit, you are taken to the following screen. This shows the components in that library and the tree listing of their constituent parts. I have labelled the important constituent parts of each connector version

.



If I needed to, I can now add a new version of a JST connector to this library. This will involve creating a new symbol, new footprint and a new 3D model. To keep my sanity, I suffix each part of the component constituents with SB, FP and 3D but you can use whatever method of naming that helps you get the desired result. The names are not important, just how the named parts ultimately mesh together. If you want to change any of these visible names then a right click offers a Rename function.

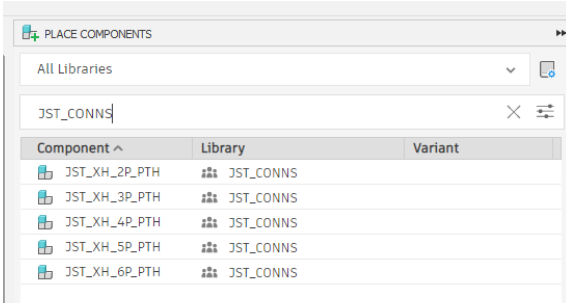
To add a new JST connector to the JST_CONNS library I need to make sure I am in the JST library and to then click the new 'component' icon.



The new component is created using the above icons in sequence. The icons are only used for creating the component and its constituent parts NOT creating the library which is done via the File menu. This is an important differentiation. In this example we do not need to create a new library container file as we are adding to an existing library and working within it.

The first step is to create a new symbol that will appear on any circuit diagram where you want to use this new part. The second icon creates the pad layout for the part when designed into the PCB. We could stop at that point as we do not need to add a 3D package if it is not essential to the PCB design we are working on. However by adding a 3D package you will get a more accurate 3D rendition of the final PCB assembly. This might be critical in making sure the PCB design fits precisely in the enclosure you have in mind. Without a 3D package, Fusion automatically adds a placeholder image.

Having created a new component in a library and be in the process of creating a new circuit diagram, the new part will appear in the Place Component search window. All the components in any library are listed in this window regardless of which library they are contained within. Here is refined window view when I do a search on *JST*. It shows the components that match this search criteria and the middle column shows their container library.



The above is perhaps a daunting introduction and uses menus and terms that you might not yet fully understand. The overview does highlight that there is some thinking to do before jumping in and creating randomly named components. You need to think where you want to have them stored so they are logically in the right place for ease of finding and editing. Components of a like generic use or style could go in a common library while unique use components might just share the same name for the library and the component. (Which can be confusing).

The above description is an overview of the end aim in creating libraries and an example result.

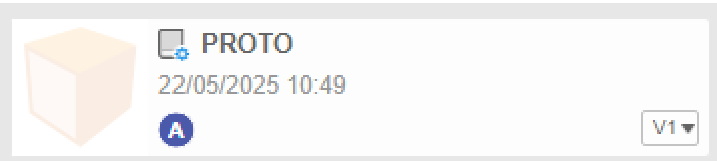
VI Creating a Library and Components.

I will now go into more detail on the process. ***I need to state that this is my understanding of the process and may not accord with Autodesk's views of how things should be done or referred to.***

I am going to create a new library to store the prototyping parts that I use on a regular basis. The first step is to create the library container file. Let's call it PROTO. Remember that this is a container file. I am giving it a name that I will associate and remember in order to help me to quickly find the parts I know are stored in it and which I use most regularly when starting a new design. Follow these steps : -

Stage 1 – Create the Library Title

Go to the Fusion File drop down menu and select New Electronics Library and press the Save icon. The Save dialogue box will pop up. You should check that you are saving in the Libraries folder. If you haven't got a Libraries folder (properly called a Project) you need to create one in your hub root level. Once in the Libraries folder, name the new library (PROTO) and click on Save. PROTO will not yet replace the 'New Library' title text. I am not sure why this takes a while to happen, but could be that it needs you to begin to start the component creation. You will have a new library file showing in the Data Panel (if on display).

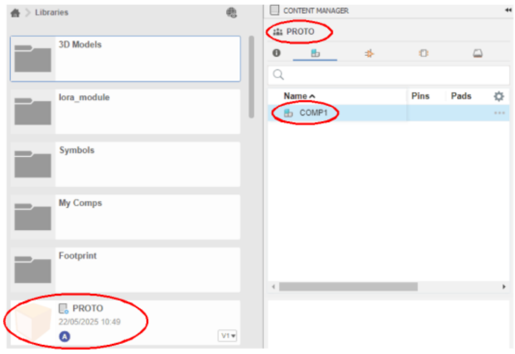


Take notice of the small icon used to represent a library in the data panel. It doesn't stand out in the same way that a PCB project icon does. (Perhaps an Arduino IDE style row of books would be more appropriate ?)

It is now a matter of stepping through the creation steps.

Stage 2 – Create the Component name

Click on the Add Component icon and give a name to the first component to be created. Call it COMP1. This will be in bold and initially will have an * next to it until you press Save. (The * will always appear to remind you if something is not saved). Notice that PROTO has now appeared as the library name.



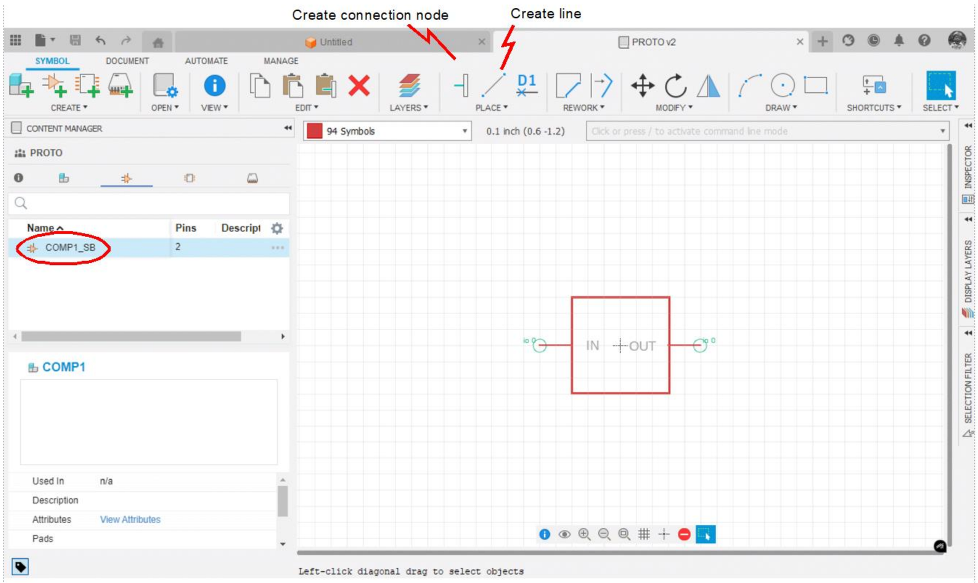
Stage 3 – Create a Symbol for the Component

Next click on the Add Symbol icon and name the new symbol COMP1_SB. This will bring up the creation window where you can draw the symbol as you want it to appear on the circuit diagram. For this example, create a simple box with two connection nodes. Note there are various options with the node creation tool. These can change the style, length etc. Similarly the Line tool has various settings. I will not go into these options as they are well covered by the videos mentioned earlier in the test.

Always centre new symbols on the cross hairs as this will be where you 'grab it' when creating circuit diagrams.

Right click on the circular connection nodes and re-name them IN and OUT.

Click on Save. This will appear as a User Save rather than a new Save dialogue. The asterix will now disappear against COMP1_SB.



If needed use the SET_NAME_VALUE ulp to automatically insert the name and value placeholders on their correct layers. You will need to repeat this same ulp when creating the footprint as they use different layers,

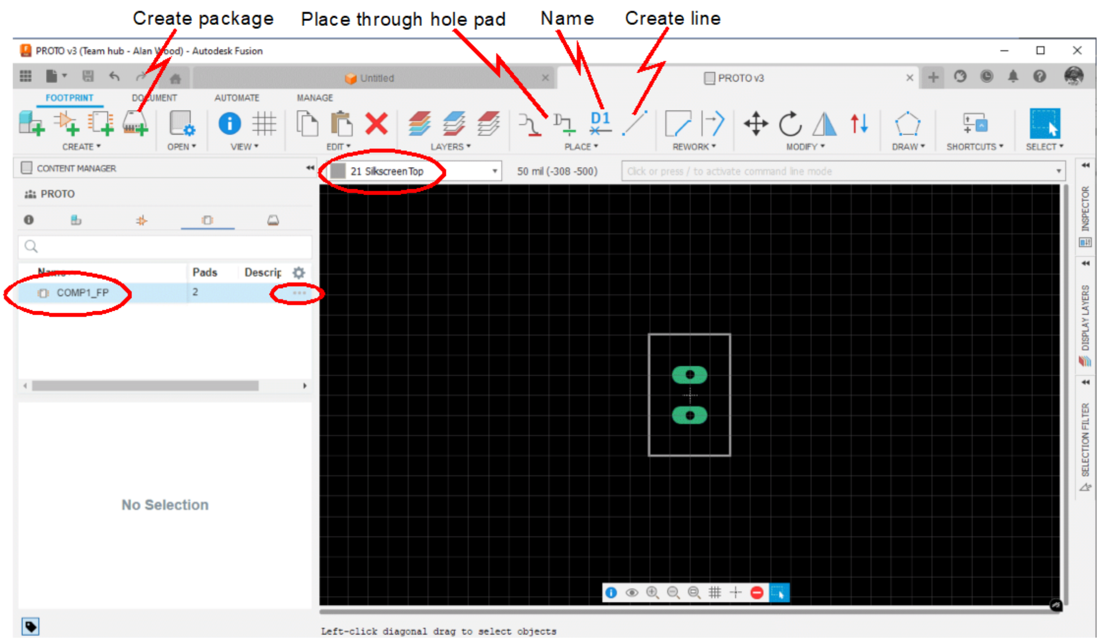
Warning : When creating symbols, do not vary the creation grid from the default values (0.1"/12.7mm) or your circuit connections will not mesh with the node handles on the symbol.

Stage 4 - Create a PCB footprint for the Component

Next click on the Add Footprint icon and name it COMP1_FP. Again see the asterix is present until the Save is performed.

Place two through hole pads (PTH) symmetrically either side of the cross hairs. These are the two pads that will appear on the PCB layout for connecting to COMP1. The image below shows the through hole pad icon. The icon to the left of this PTH icon is the SMD equivalent. Selecting the PTH tool will bring up options. Once the PTH pads are placed, use Name (or right click) on the pads to rename the pads to something more appropriate. Rename the two pads IN and OUT. The Name text will not be obvious unless you really zoom in very close on the pads. Draw a box around the pads. This box will show on the PCB as the silkscreen. Notice that the line drawing tool changes the layer to match this.

Save once again as a User Save only.



Notice how with each step taken so far, the blue highlighted name in the left hand box only shows the name of the part currently being created (i.e. there is no inter part relationship apparent or tree structure as yet).

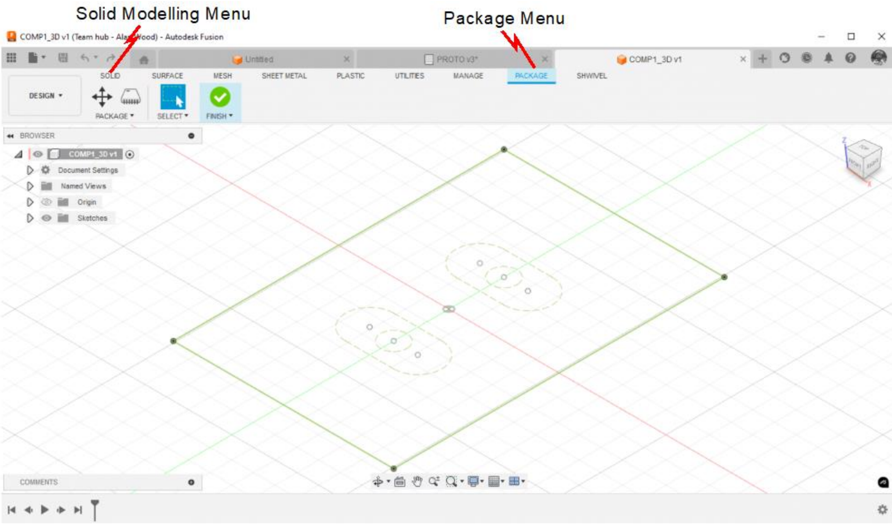
Stage 5 – Create a 3D representation of the Component

We are now going to create a 3D representation of our component. Logically you would expect the next step to be click on the Create Package icon as shown above. If you do this you are taken to the 3D modelling screen and are presented with a Package Generator sub window. This is an excellent tool to create standard packages and is explained very well in the two videos I mention earlier.

To create a specific custom 3D model click on the end of the COMP1_FP blue highlighted line (see above) where there are three little dots as shown above. Click on the dots and select Create New 3D Model from the options.

This will bring up a 3D modelling screen with a sketch centred on X0,Y0,Z0 representing the Footprint you have already created (COMP1_FP).

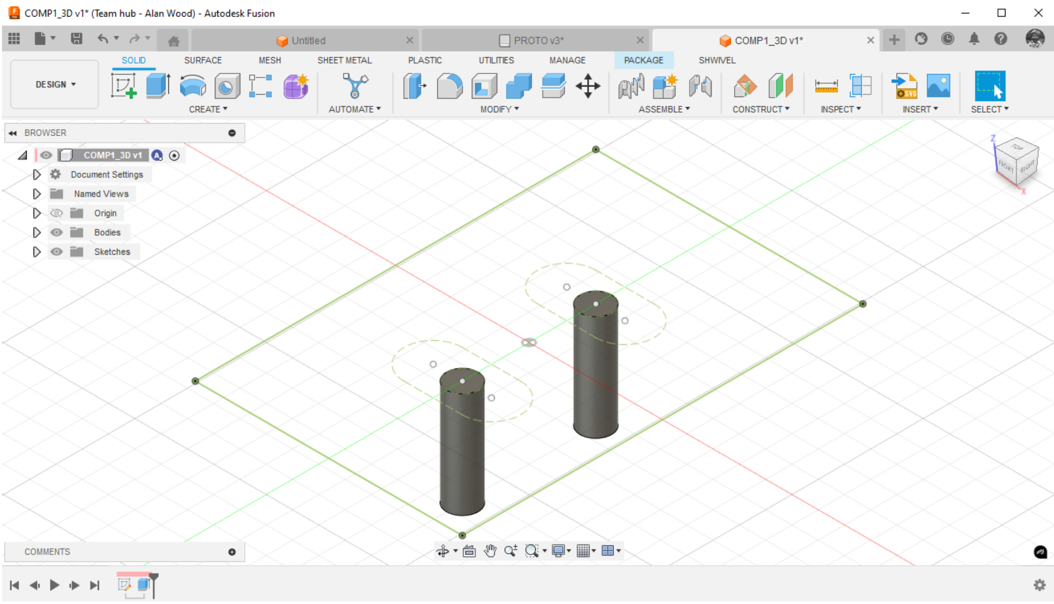
You will be immediately requested to Save (a naming Save not a User Save) so enter COMP1_3D and then click Save.



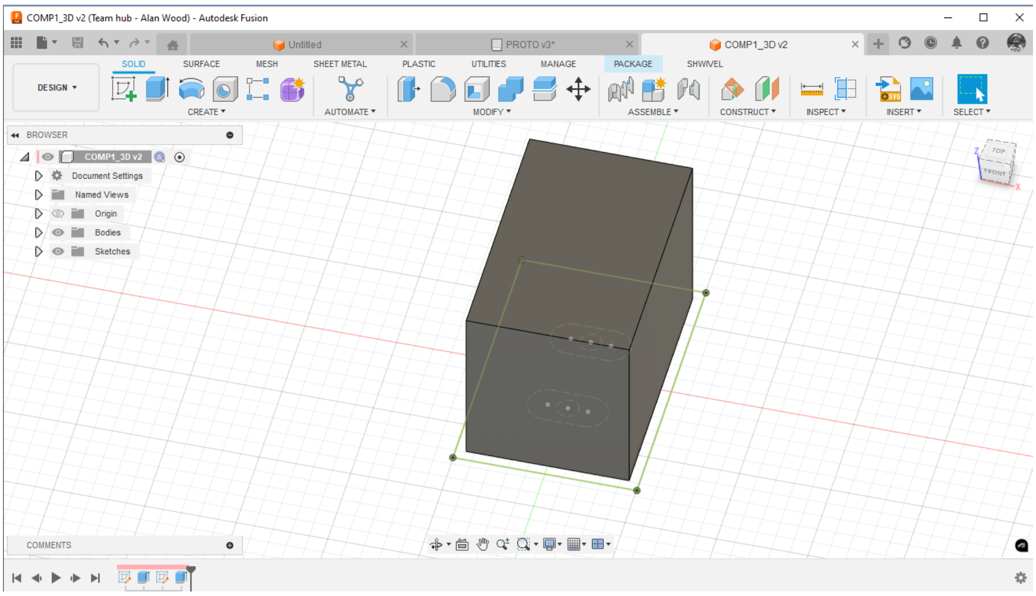
Go to the Solid Modelling Menu to enter the modelling module. The screen will redraw but now with the familiar 3D modelling icons showing across the top line. The faint image of the footprint COMP_FP will remain centred on X0,Y0,Z0.

I like to draw representative connecting pins or wires for the component first. Enter Sketch mode, select the X,Y plane and draw two 0.6mm diameter circles. Use the Concentric tool to move and locate these correctly on the sketch image holes. (You can enter metric dimensions even though you maybe are working in an Imperial work sheet by adding 'mm' after the dimension).

Extrude these two circles downwards by -2mm to represent the pins of the device. Pins of this length will protrude sufficiently to pass through a 1.6mm PCB thick board and give a representative image on the 3D rendition.



Next sketch a centred box on X0 Y0. This should closely match any dimensional information you have from the manufacturer of the component. As our part is fictional just make it similar to the existing sketch of the silkscreen outline. Extrude upwards for +5mm. Make sure this is a Join operation so the pins and box are one item. Save this which will be a User Save.



Note the model will faintly show the original sketch in the background. I prefer to switch off all the sketches at this point and User Save once again. If you don't switch them off, I find they distract when later viewing the PCB rendition.

Go back to the Package tab and click on Finish.

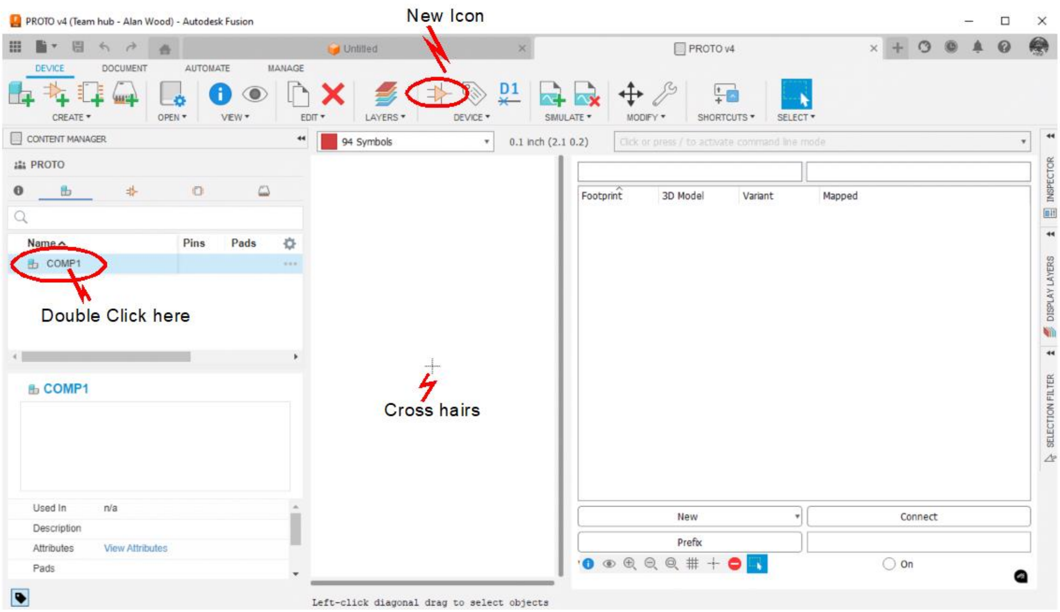
This will take you back into the PCB Library section where the footprint screen will still be showing but COMP1_3D will be listed in the left hand sub screen. - Save again.

Stage 6 – Bringing it all together

You should now have each part of the component showing under each of the creation icons and the name of the Library name will be PROTO.

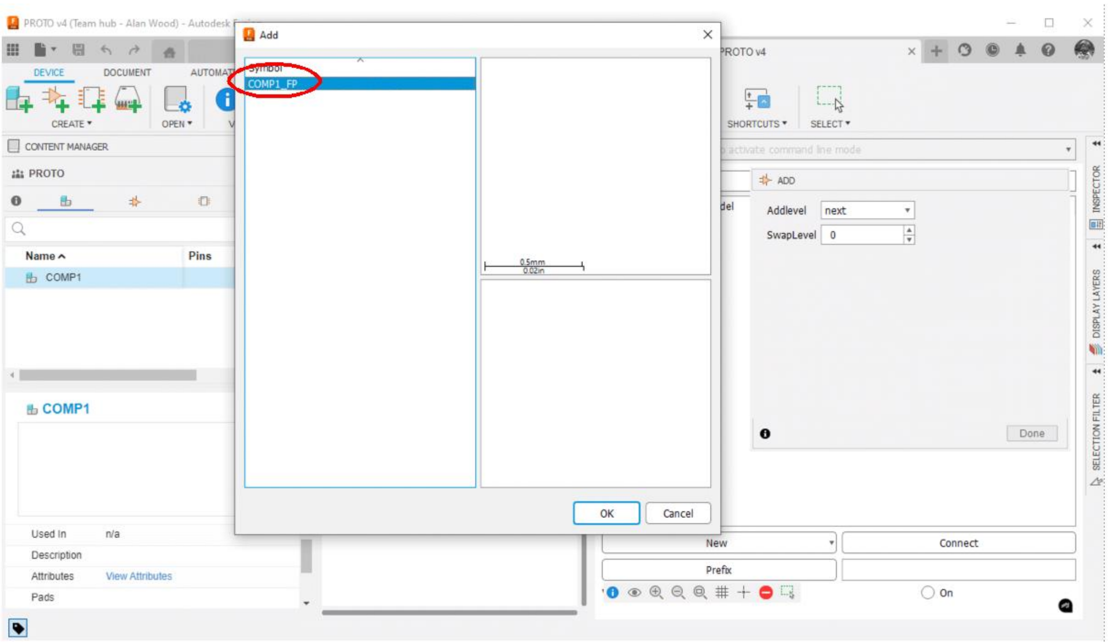
Click on the left hand icon that represents the top level to the component and you will see COMP1 showing. At this moment in time the component (COMP1) is not connected to the SB, FP or 3D parts.

Double Click on the blue line showing COMP1 in the left hand window. A multi area window will appear (these might need resizing) and a new icon will have appeared in the top icon line that looks like a bigger version of the Create New Symbol icon.

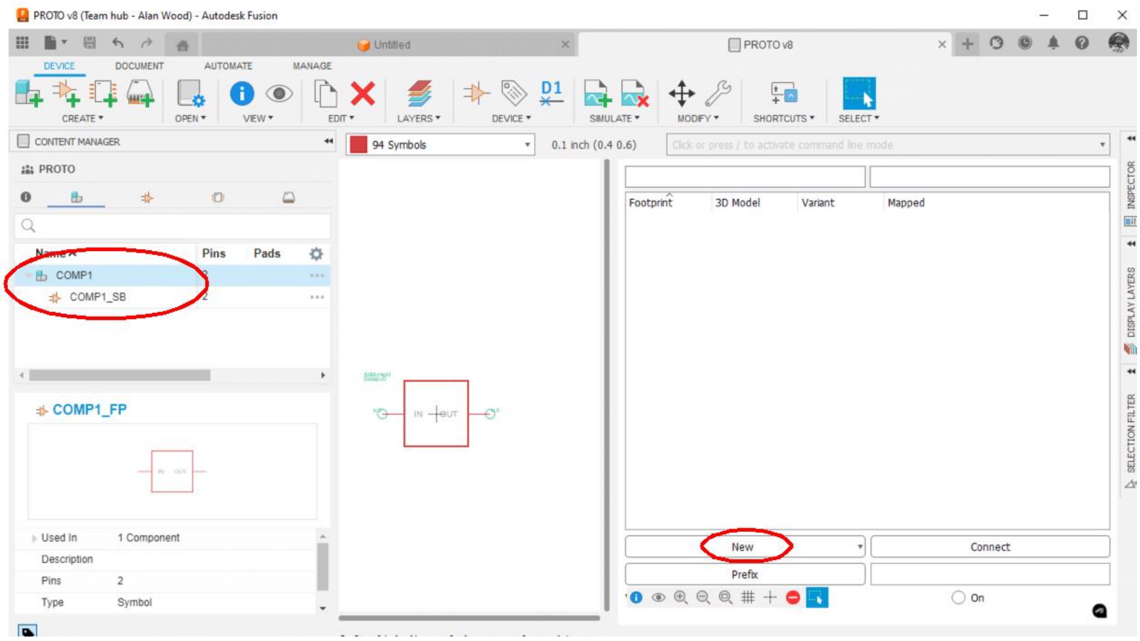


Stretch and pull the various box borders to make the centre box large enough so you can see its central cross hairs.

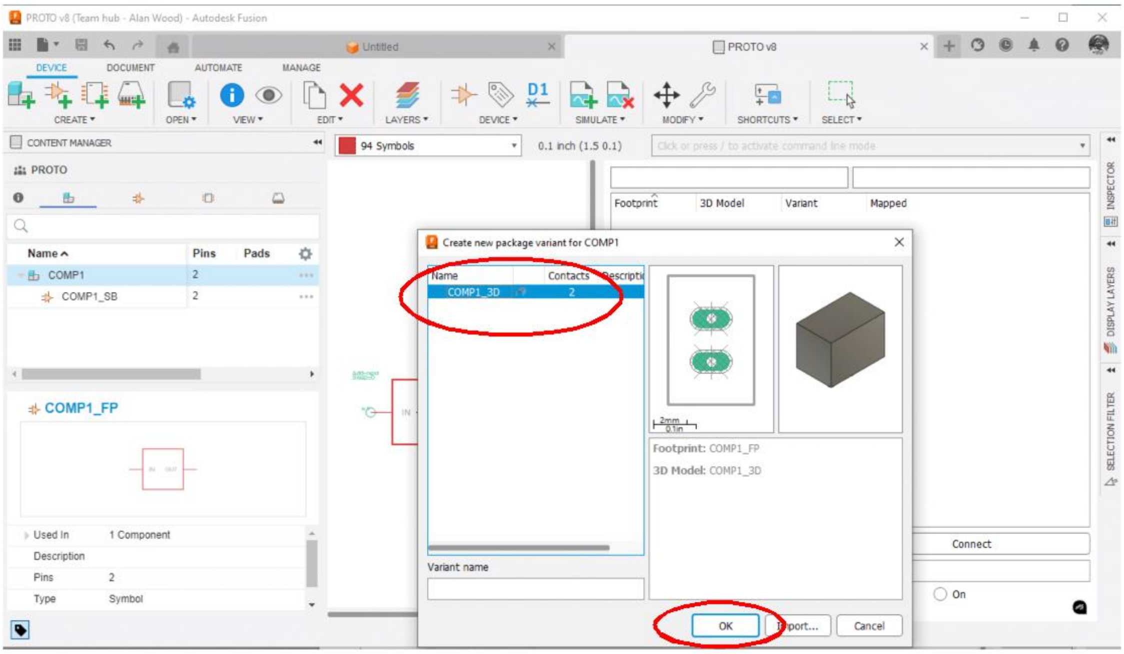
Click on the newly appeared icon whereupon another sub window (Add) will open. This is shown below and will display your symbol. Click on COMP1_FP in the ADD window and OK.



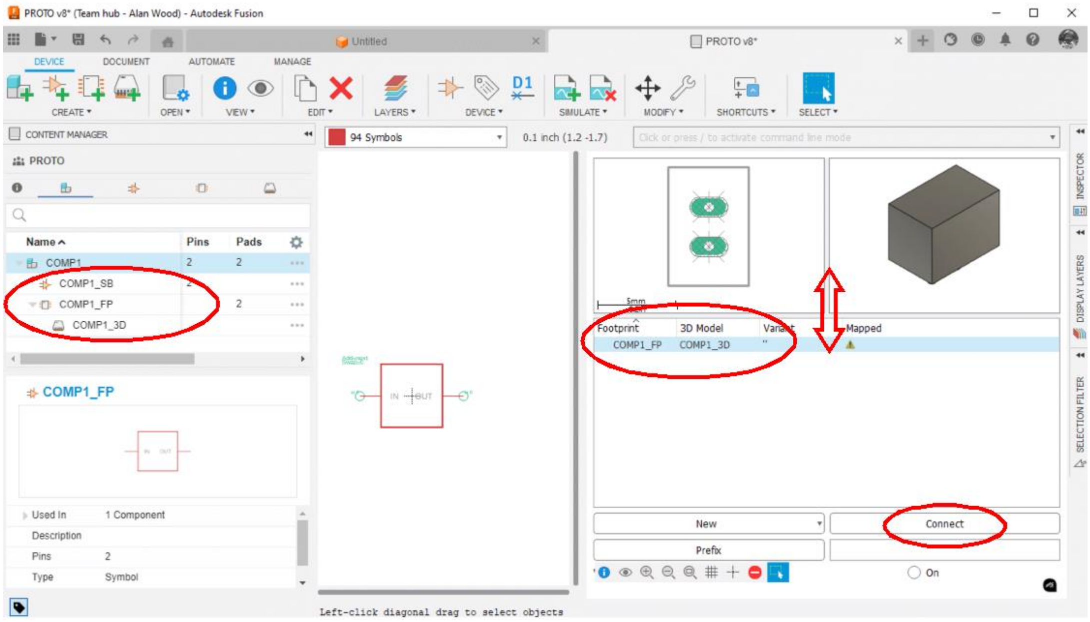
You will now have your symbol 'stuck' to your mouse cursor and you need to move it to match the cross hair on your symbol with the cross hairs in the left hand box. Click when correct. Notice that the COMP1_SB file has now appeared below the COMP1 in an indented manner on the left window. Your component tree is beginning to take shape.



Click on New in the bottom window. Select Local Package from the two options. The Create New Package Variant window will appear where you can select the model you have created within the blue highlighted area and then click OK.



OK this and they will be pasted next to the symbol on the subsequent screen as shown below. You will probably have to drag down the box divider line to see these more easily.



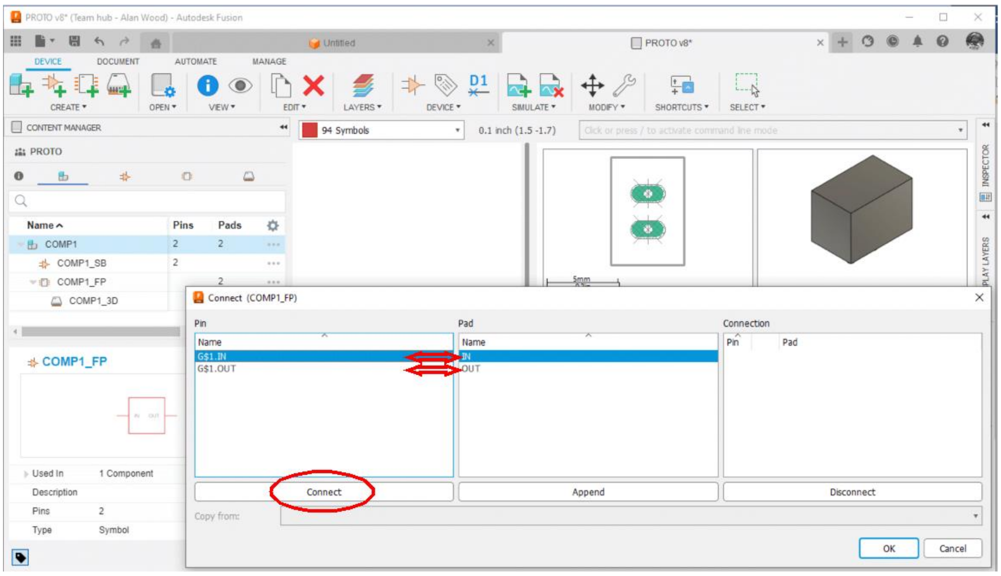
Notice how the COMP1_FP has appeared in the left hand window and if you click on the associated side arrow to expand it out you will see the COMP1_3D is also present but indented.

The Prefix button allows you to set what the value is prefixed with when used in an design. For instance, it is normal for resistors to be prefixed with R etc. Once defined click on the ON radio button to enable it. Note that if you use a third party modelled component they might have used a different prefix to the one you have done leading to a mixed prefix labelling of the same type of parts.

At this stage all parts are collected together in the library container but they do not realise how they relate to each other pin to pin.

Stage 7 – Mapping the Pin and Pad connections

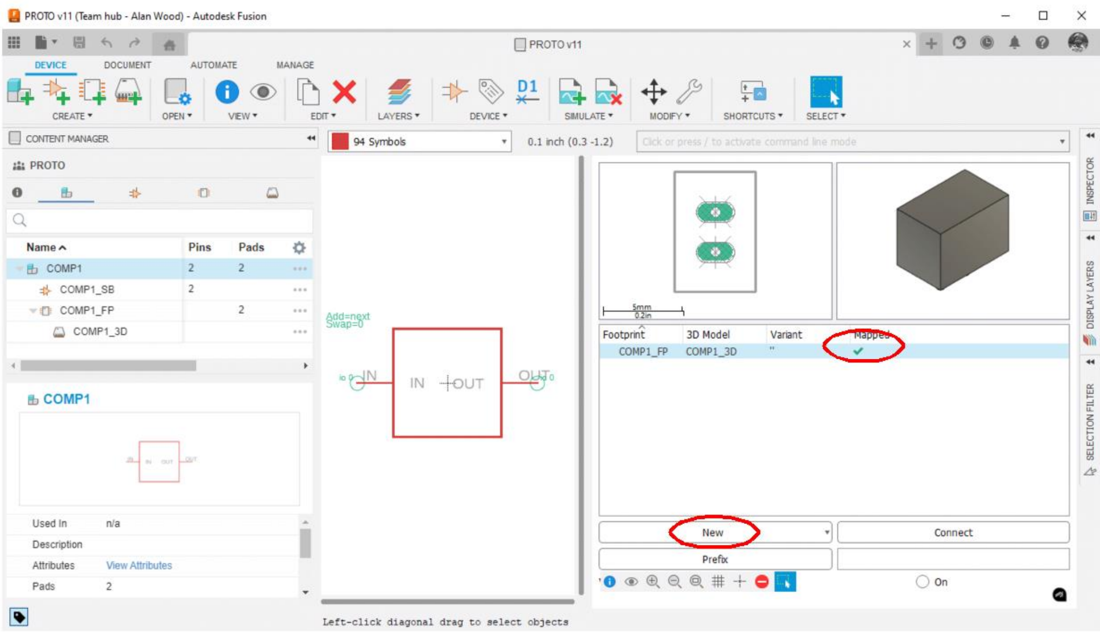
Click on Connect to go to the mapping screen.



This is a simple process of linking pin names to pad names. Go down the list in the Pin window and click on the associated Pad name and then click Connect. The pairings will then move across into the Connection window. The example that has been created for COMP1 will only have IN and OUT to be linked. Once this is done press OK. A tick will appear in the Mapped column as shown below.

Click on SAVE on the main menu.

The library part is completed and ready for use in future designs.



You can expand the view of what the component file contains using the indenting arrows in the left hand box. You can right click and Rename any of these should you wish.

If you need to edit any aspect you can right click edit on the blue line under the footprint and 3D images on the right hand side.

That is the basics of creating a new component in a new library. As ever there is more depth you can go to but this is better covered in various YouTube videos.

Here are some one liner prompts : -

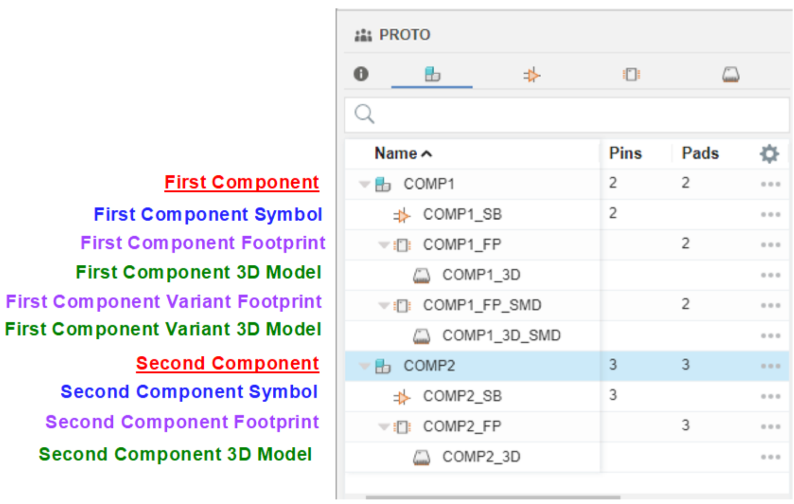
1 Create a new library container using the File menu and Save
2 Add a new Component, name and Save
3 Add a new Symbol, name and create the geometry and Save
4 Add a new Footprint, name and create the geometry and Save
5 Use the three dots on the Footprint line to Create a new 3D part
6 Move between the Package and Solid work areas to create a new model that matches the Footprint and Save in both work areas.
7 (Unclick sketches from the 3D model and Save).
8 Double click on the top level component name and click on the large icon of a new symbol that has appeared in the top line of icons.
9 Go to the Add sub-window and select the file containing the symbol
10 Centre the symbol on the cross hairs in the left hand window and Save.
11 Click on New to go to the Add Local Package sub-window.
12 Select the New Package variant and OK
13 Click on Connect and map the Symbol nodes to the Footprint pins
14 Click on Prefix to set the letter you want to appear before a part number (e.g. R, C, etc) and then enable this with the tick box.
15 Save

VII Closing comments

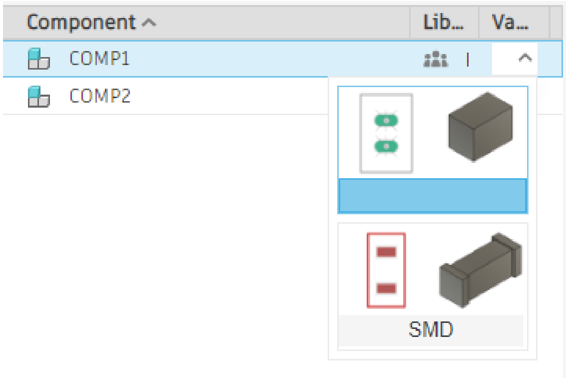
A library has been created called PROTO and at the moment it only contains one component COMP1. No doubt further parts will be regularly needed for prototyping experiments. By following the above steps a COMP2 part can be created with a similar tree structure. It would appear in the left hand listing either as single line or as an expanded tree structure.

If a part has the same symbol on the circuit diagram but is now a SMD version it will have a different footprint and a different 3D package. This is created by highlighting the symbol of interest and then creating a new footprint and 3D model to match. Once created follow the same process as detailed above and at the Add stage you will see the new variant available to select and the original one greyed out.

The image below shows how this might look on the library structure.



Just as a cross check here is the view when selecting a component from the PROTO library with the drop arrows revealing the two variants of COMP1.



There will be a large re-use/commonality of constituent parts that make up library contents. The effort in creating these base constituents is therefore well rewarded in the long term.

To summarise : -

1 A library can contain any number of components and their names do not have any bearing on the name of the library.
2 Each library part will have a different name and have an associated symbol, footprint and 3D model although the 3D model is an optional constituent. If not needed it gets replaced by a placeholder image.
3 The constituent parts of a component are represented in a tree structure.
4 Once the library has been created the process of creating the symbol, footprint and 3D model is a repetitive process and can be repeated to create either new components or variants of components within the library. The additional components or variants will show in the tree view as new lines that expand out to show the constituent parts.
5 If a component uses the same symbol as another component and the footprint or 3D package are different then a Variant can be created that sits in the Component tree below the symbol that it is common to both the original and variant.
6 Once a component is brought into a design it becomes independent of the original library. All of the data it needs is stored in the design. You can delete or otherwise mangle the original library and the design would be unaffected. In short there will be no retrospective back linkage problems.
7 There is currently no easy way to merge existing libraries in FE. You have to manually import the components between libraries and then delete the duplicates.
8 An interesting effect – a fixed resistor has a single symbol. All types of fixed resistors are therefore variants. If you like a particular resistor from a third party library and you decide to import it into your local library … you will get all the variants, not just the one you liked.

VIII Some useful basic hints in using Fusion Electronics

Having got a better understanding (hopefully) of the library process we can now proceed to creating a PCB design and not be restricted to only being able to use readily available components.

The following listing is my memory jogging list of things I have discovered and use regularly when designing PCBs. These are in no particular order and the list is likely to grow with each new issue of this document. The list is sorted into a generic set of headings but these are not hard and fast.

Screen and layout related Issues

1 When FE is first booted it is overpowering. **Minimise the sub menus** so you get a better working area for inputting the circuit diagram.

2 **Change background colours** using Prefs. This is particularly useful when you first go to the PCB tracking screen and all the circuit parts sit outside the PCB black area and are hard to see.

3 Use the **X and Y coordinates** readout to accurately get things in the right place. These always show the position of the mouse pointer.

4 Use the **'Lock' check box** in the Properties sub menu to keep critical parts from moving accidentally (connectors, control pots, mounting holes etc).

5 Edit **the board outline** by clicking on the white line that borders the PCB black image. Use the coordinates to set the correct shape by doing 'to and from' geometry.

6 **Moving around the screen** gets frustrating particularly if you have a small monitor. Use F6 to centralise the view and middle mouse button hold down to pan and zoom in and out. My 3D Connexion Space Mouse does not appear to work very usefully in FE but I did manage to program F6 onto the left button.

7 You can't have a big enough viewing screen … PCB layout desktop gets very cluttered with sub menus. Two monitors works a treat.

8 There are lots **of right click contextual menus** to get you quickly where you want to go or do. Keep an eye on this and familiarise what does what where.

9 CTRL + ALT + C brings up the **Command Line** so you can access short cut functions**.**

10

Component and Circuit Diagram Entry

1 Rule #1 is **do not deviate from the 0.1" grid** when placing connection nodes while creating new circuit diagram symbols. The other associated imagery (silkscreen etc) does not matter.

2 Get used to making **sub circuit diagrams** and link them rather than trying to get the whole circuit on one confusing screen.

3 When making **new symbols** for components you will find initially that you make them too large. Practice what looks neat and make your own notes and rules for this but always put the connection nodes on the 0.1" grid.

4 Use the SET_NAME_VALUE ulp script to **automatically enter the Name and Value** placeholders when creating symbols and footprints. You have to use it in each creation activity as these placeholders are displayed on different layers.

5 Set your **preferred component numbering prefix** using the Prefix button in the Connection window. Enable it with the radio button. Note that third party libraries might use a different prefix to the one you prefer which will result in potential confusion on the circuit diagram etc.

6 **Renumber parts** across the circuit using the Automation script (RENUMBER_SHEET.ulp) using Automate/Run/Run Script ULPs. This moves up layers as shown in the dialogue box. After using this script reset the view using Layers Display, Show All.

7 **Press and hold down ALT to go to finer grid steps** when doing tracking or moving comps

8 Library view as mentioned earlier : -
   a. In use means ready to use
   b. Available means available to use but not enabled
   c. Sparkfun is a good goto library for most parts

9 **Searching for a part** in the library screen must use the full part number or use start and stop wildcard (e.g. *abcd*).

10 SamacSys library loader is a **good source of library parts**. The items are downloaded as a ZIP file. Check the ZIP contents and use the Eagle variant for Fusion import..

11 When **creating connector symbols** and footprints I suggest you add some form of orientation index so you know how it will sit on the PCB and not have to suddenly mirror all the connections because it 'came out wrong' when you tried to track it on the PCB.

12

<u>PCB Tracking</u>

1 **To save all parts** in a particular PCB design as a new library use EXPLIB script while in Schematic mode. This is very useful. Not confirmed but I think this would allow a schematic to be created of all the parts you prefer to use just blobbed onto a single sheet with no interconnects and then use EXPLIB to create a 'My_preferred_comps' library.

2 Use **named nets** wherever possible to clean up the PCB layout view and ease confusion when doing layouts

3 To change a component from one side of the board to another use 'Mirror' (not exactly obvious …) you will see the component and associated tracking change colour to suit.

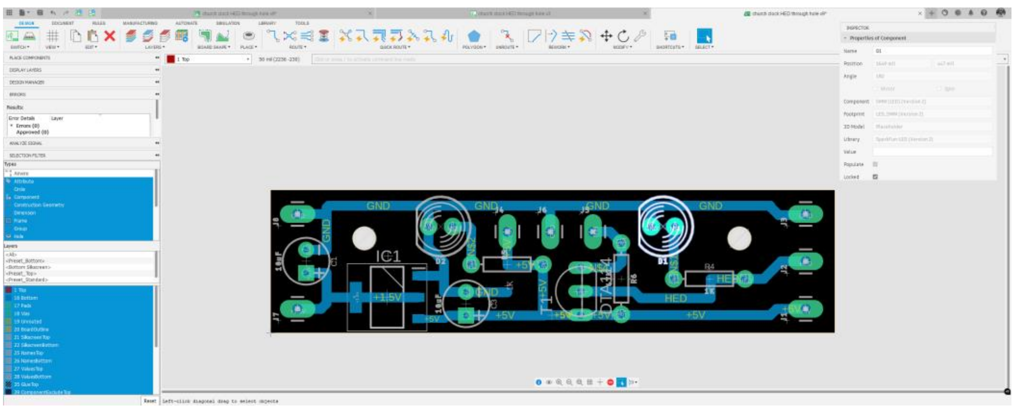
4 To **change sides** when tracking the PCB press and hold the Space Bar while clicking with the mouse where you want a Via to be. I find I have to do this a couple of times to convince myself it has happened. Confirmation will be a change of tracking colour.

5 Keep the **Errors window always viewable** to identify tracking errors. What looks like a good termination to a pad might not be in practice and will leave an Airwire. The 'ping' sound you get on making a connection is useful but can get irritating for others nearby …

6 **Vias with the default settings** I find are too small to allow a wire link when prototyping on a milled PCB using FlatCAM. A global change can enlarge this.

7 When placing pads for items like connectors, the **physical pin sizes** might not match the drill hole in the pad. I use the elongated pad style for most connectors. If you change the drill hole size the pad appears to enlarge or shrink to match the new drill chosen. Note that the weird imperial drill sizes listed for choice are actually conversions from standard metric sizes.

8 The PCB layout screen can be confusing. The zoom out menus do not seem to anchor where you put them or stay minimised from action to action. I try to **minimise as much as possible** and let the Inspector submenu float and minimise it in width and height down to the Locked tick box. This is how I try to keep it with minimum confusion. On the



left hand side I only have Errors and the Selection Filter. This will have Types and Layers showing where all or distinct selections can be made.

Note that with all highlighted blue in Types and Layers this gives full access to all aspects and you can then select one particular line to focus by type. Use Click at the top and bottom with Shift to reselect all or in Layers use the Default Presets. For multiple selections hold CTRL and Click as wanted.

9 **Add additional silkscreen text** on blayer25 or tlayer26. Note that the **Text tool** is on the Document menu (obvious ? …).

10 Press ALT and hold to go to **finer grid steps** when doing tracking or moving comps. You can define your preference for the finer grid size.

11 **Add mounting holes** to a board either through the 3D model module or using the hole command. The Properties info screen will give the hole centre. This applies to any component providing the library component is built on the central cross hairs. See also Add holes as routing lands

12 To **paint (flood fill) a large area of copper** use Polygon and define each geometry change position with a mouse click or highlight the whole board as a rectangle and let it do its own thing. Name the polygon net as appropriate in right click properties. Note if you have 0V or GND tracking as a named net then you can add the polygon to ground by using the same net name. Fusion adds thermal reliefs on all other associated pads. To **change the thickness of the thermal relief tracks** right click on the polygon edge and change the property value

13 **When tracking to a passive part** on a SMD based board try to ensure the track to the SMD pad is less than the pad size and always have the entry position of the track to the pad to be central on the long dimension and symmetrical the same on both pads. Avoid having two tracks to the same pad from different directions. Without care on these aspects of geometry there can be a tenancy for the SMD parts to tombstone on reflow.

14 A lot of the standard library parts I use have a **small text font** which doesn't present well. Using the Selection filter choose Attribute, click on the text and right click for properties to see all aspects of the font. I usually change it from Vector to Proportional which looks cleaner. Silkscreen is on the 27ValuesTop layer. The Inspector sub menu also gives you access to the properties of the Attribute.

15 **Thermal relief** on ground pins in flood areas can be changed by clicking on the edge of the polygon and change parameters (see also #12).

16 **Right click 'Show' on a layout net** to see where the track meanders around the board.

17 Use **Gerber export** if you want to use the individual manufacturing files (i.e. for FlatCAM) but ODB++ seems more popular with offshore manufacturers.

18 Use sketch.ShowUnderConstrained **to check integrity**

19 You can make **global changes** to specific parameters by selecting just the Attribute in the Inspector such as track or via etc and then highlighting the whole PCB layout. This will fail if the global change causes any aspect of the change to violate the design rules settings. This is particularly so with track width changes. I tend to track with a thin track so the view is not too cluttered then change widths once I have a better feel for the layout. If I have a global change failure, I then reduce the selection window size and select smaller multiple areas of change to narrow down the area of the violation.

20 **Selection filter types** sub menu is useful to just pick up certain layout options otherwise leave all types selected for most activity.

21 There are lots of things to **tweak in Rules** to set more personal defaults for things like track width. Take care !

22 When you have all blue selected in the Selector menu click on an item the Inspector will show you the associated data which includes which layer this Attribute is placed on.

23

3D Modelling

1 Modelling the associated 3D part for a component is an **optional addition**. The board layout will work without this but the 3D rendition of the PCB will just have black placeholder images.

2 There will be repetitive use of the same 3D models for different components so keep them in a **separate folder** under the Libraries location.

3 When adding a 3D model to a footprint you will need to manipulate the 3D model to **match the footprint holes or pads**. Get the part in the correct orientation and then if possible use a Joint command to lock it in place.

4 After creation of a 3D part, switch off all sketches before saving so it does not clutter the 3D rendered assembly view.

5 Save and save and save again constantly when manipulating the model. I have no confidence that the Finished click box also Saves.

ANNEX 1 – Dialogue on Autodesk Fusion Electronic forum

When I put out a plaintive cry for help about the basics on the Autodesk FE forum I received two key responders. The first was from Jorge Garcia at Autodesk who is a key lead on FE.

Below are the ping pong responses I had with Jorge.

Me :-

I get by creating new parts, but quite often wonder how I get a result and where it ends up.

I get that there are libraries and that there are devices.

I have watched all the tutorials by Autodesk and third parties and it is still as clear as mud.

The Autodesk tutorial focusses on steps to do but like all tutorials written by an engineer it misses the simple background stuff.

If you create a library is this similar to folder and if so where does it live ?

Should I see a folder within Library folder that matches the name ?

How do the commercial libraries get multiple devices inside a library ?

Should I be able to do this ?

I know I can create a library from a PCB design that holds all the parts in that design but that isn't quite the same.

I find myself in idle moments wondering if it is just me and I am missing something so simple that would light the bulb and my nightmare would be over.

But it hasn't happened for me yet.

Anyone else out there struggling like me ? Or is there anyone out there that really understands it all and can explain in really simple terms so I can get on with the complicated stuff designing PCBs

Jorge : -

First, remember that Fusion stores everything on the cloud so the comparison to a Folder breaks in a few places but thinking about it as a container is more general and works better.

When you create a library in Fusion from scratch and save it, you are saving it to a project within a Hub in Fusion. By default every user gets 1 Hub, think of it as your sandbox. All of your data stays contained within the hub. You can think of these projects as folders. Personally I recommend, creating a Project called libraries and storing all your libraries in that project that way it's all in a single location and you don't have to think about it any further.

Because a library is a container, it can contain as many components as you want in it. You can use the import button when creating new components/symbols/footprints to re-use assets from other libs. If you have an EAGLE background all of the managed libraries are stored on servers.

Me with **Jorge in bold** : -

1.First of all a general question - top down when logging onto Fusion I see what I regard as being the root folder (under the House icon) which has My First Project, Templates, Libraries etc.   Is this root folder a hub or is My First Project a hub ?
The "root folder" is a hub. Your First Project is a project within said hub.

2. I think you are saying that this is the wrong way to do it and all these individual projects should be listed as projects under the root icon ? (But perhaps it doesn't matter because they are all in my storage allocation on the web with appropriate database linkages ?

It is the wrong approach but it will only bite you when you try to share files with others. Projects have access controls that make it easier to share data in a granular way. Folders within a project have more limited controls so you may find it difficult to only share one design and not everything in the project. This is why we recommend using one project for each design.

*3.In Fusion Electronics the term library file appears to have different meanings. It can be a single component with a *.lbr name or it can be a group of components stored for convenience as generic 'Capacitors' or 'Connectors' or it could be a single component that has many different physical properties (different footprint or package etc). This latter aspect suggests that each of these variants in properties must have a *.lbr file ?*
OK so here is where the root of the issue is. I mentioned in the previous post to think of a library as a container. The .lbr (or .flbr in Fusion) is the only thing we refer to as a library. Now an electronics library can contain a single component. Often when users or third party make a library that contains only one component they give the library the same name as the component. That's what has giving you that first impression. The 'Capacitors' is again just a container but it contains more than one component. A component is made up of symbols, footprints, and optionally a 3D Model. Components are trees within the container that is the library.

4. To create a new component (.lbr) file you appear to have to follow a rigid process of creating a symbol, then a footprint and then a 3D package.   Each of these constituent parts appear to be completely independent and can be named anything you wish to call them.   Once you have followed the process your final action is to name and save what I assume must be a linkage file combining these constituents and this becomes the *.lbr.   Then and only then does the new component exist.*

*Are the constituent parts frozen in their form at this point together with copies of what they 'looked like' at the time of the creation of the component being created. Are they stored away as linked items to the *.lbr?*
The process is not rigid you can make symbols at any time, same with footprints and 3D models. These constituent parts are independent and you link them in a Component to make a completed part. In 3, I mention that Components are trees. They branch with symbols and footprint, the 3D models are linked to the footprint. So for example within a library you can have a component that looks like this.
LM555
-Symbol of LM555
-Footprint1 of LM555
--3dModel of Footprint1
-Footprint 2 of LM555
--3dModel of Footprint2
Components are created within Libraries(.lbr), they are not themselves libraries. In your model the Component is the linkage file. Then contents of the library are independent of any copies that are made in other libraries. So for example changing Symbol of LM555 in library A will not change the same SymbolofLM555 in library B. 3D models are the only exception to this because they are externally referenced.

5.Can you access these constituent parts to re-use them on new components and if you do and edit them in any way does this impact back onto the original component characteristic's ?

You can definitely re-use constituent symbols, footprints and 3D models to make new components. This is done through an import workflow where you can bring existing assets into the library you are actively working on. The copies generated through this process are independent of the original assets. The idea behind this behaviour is to be conservative, you don't want a bad change propagating through possibly hundreds of libraries. Some users have argued that this is wasteful and creates duplication, but we have chosen to err on the side of caution and avoid links.

This was very helpful for me and cleared some of my confusion.

Annex 2 – Chuck Todd's Document via Autodesk Forum

In the course of my dialogue with Jorge I received the following document from a FE user called Chuck Todd. Some of what Chuck has written is similar to the YouTube information already linked above and matches some of Jorge's responses but collectively adding the two documents helped me understand and paint a better picture.

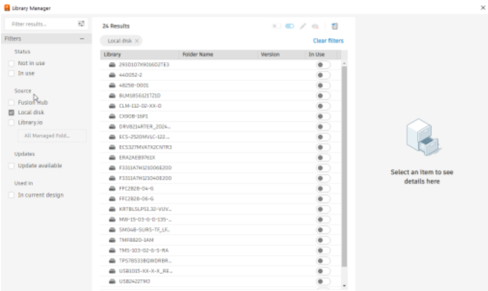
Contents

1. Where are Fusion Electronics Libraries stored?

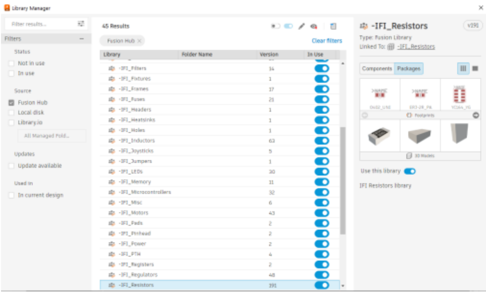
We must start by talking about the different types of libraries. There is a lot to unpack here. I'll try to be as clear as possible.

Types of Libraries:

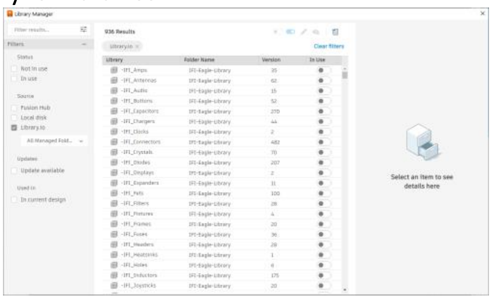
- Local Libraries



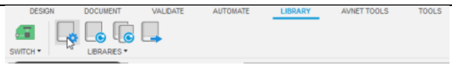
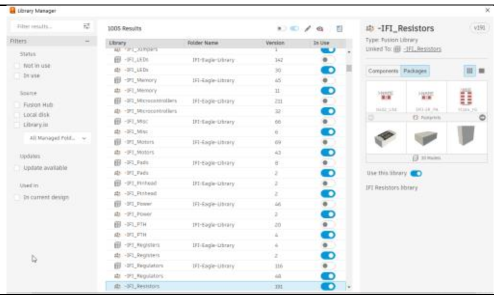
- Fusion Hub Libraries

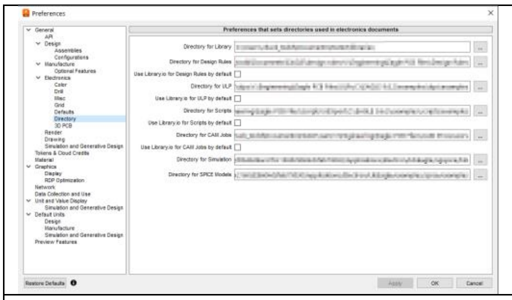


- Library.io Libraries



All these libraries can be accessed through the library manager if Directory paths are set up correctly in the Preferences.

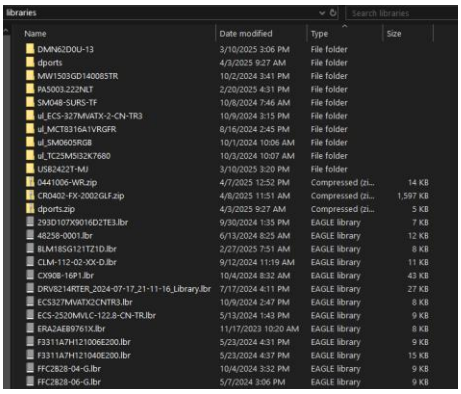
	
Accessing Library Manager	Unfiltered Library Manager Dialog

	
Directory Preferences	

Local Libraries:

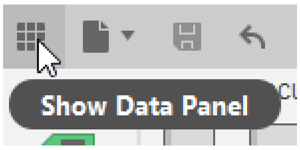
These libraries are stored on your local hard drive. They could contain a single component or multiple. They will most likely be downloaded from a manufacturer's website and have a ".lbr" extension. These libraries can be imported into your Fusion Hub libraries.

Where I keep my local libraries is shown below.

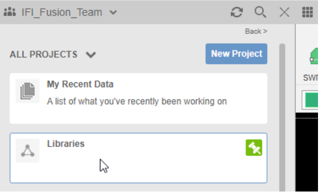


Fusion Hub Libraries:

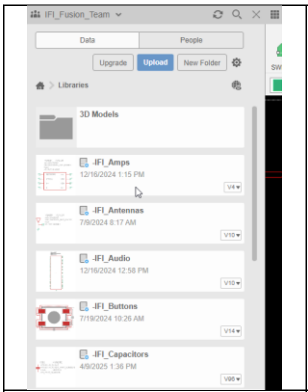
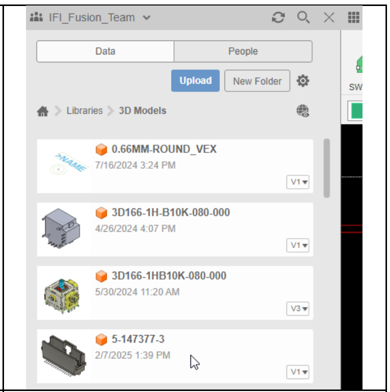
These libraries are found in the Fusion Environment and are stored in the cloud. I created a Project (you could think of it as a folder) just for Fusion Libraries. Access to these libraries is through Data Panel .



You can see that I am a member of the IFI_Fusion_Team Hub located in the upper left corner of the image below. This is a drop down that would show all the Hubs available to you.



Double click to enter the Libraries project.

	
Libraries Project	3D Models Folder

Local libraries can be imported directly into the Libraries project.

(Instead, I would open the Fusion Hub library that the local library would most likely relate to and import it there.)

As you can see, I have a folder for 3D Models in the Libraries project.

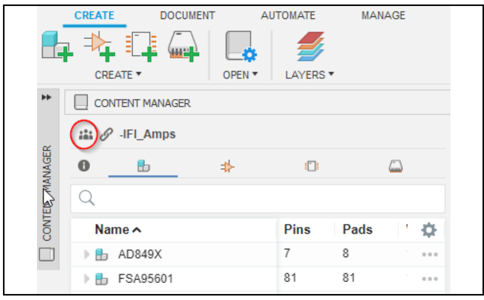
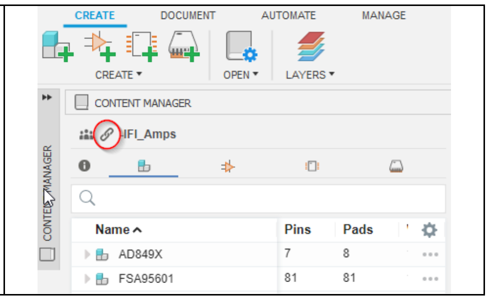
It is recommended that you use Fusion Hub libraries once the Library.io libraries have been imported.

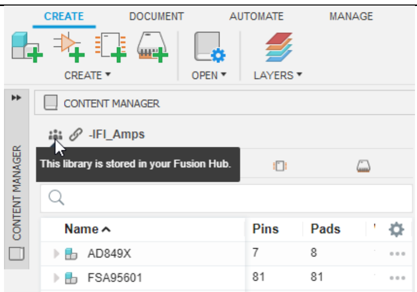
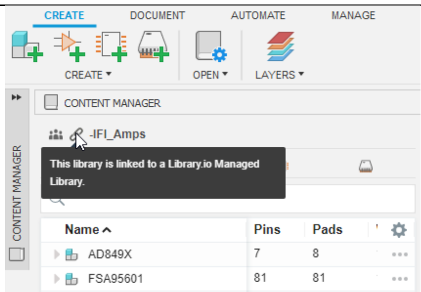
Fusion Hub and Library.io libraries are linked after importing the Library.io libraries. Changes made in the Fusion Hub libraries can be pushed and pulled from the cloud.

Library.io Libraries:

These are legacy Eagle libraries and are stored in the cloud. They can be imported and turned into Fusion Electronics Libraries. Once they are imported into Fusion they become Fusion Electronics libraries.

After a Library.io library is imported into Fusion the two libraries are linked to each other.

Fusion Hub Icon	Linked Icon
	
Stored in Fusion Hub	Linked to Library.io library

If someone were to modify a library.io library and push to the cloud the changes. The changes would be available to pull down to the Fusion Hub library and vise versa.

2. What is a Fusion Electronics Library?

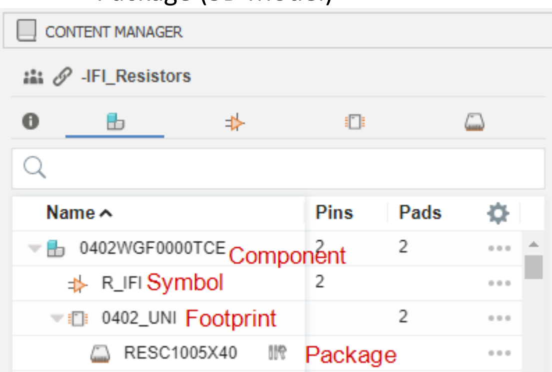
A Fusion electronics library is a container for components that are stored in the cloud and is only relevant to the Fusion Hub you are a member.

3. What is stored in a Fusion Electronics Library?

Components are stored in a Fusion Electronics library.

Each component has 3 pieces.

- Schematic Symbol
- Footprint
- Package (3D Model)



Components:

Components can be found in the content manager and links the Schematic Symbol and Footprint together.

Schematic Symbols:

Schematic Symbol can be found in the content manager and is a representation of the component with all the pins.

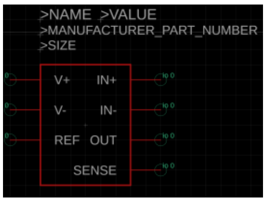
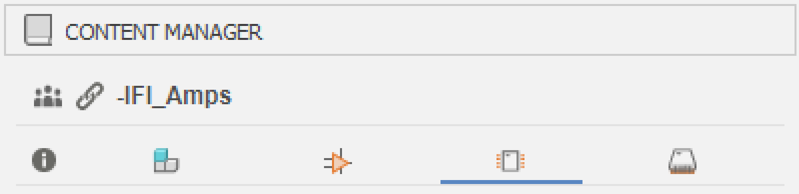


Figure 1: Schematic Symbol

Footprints:



Footprints can be found in the content manager and is made up of pads, outline, etc.

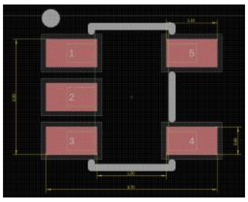
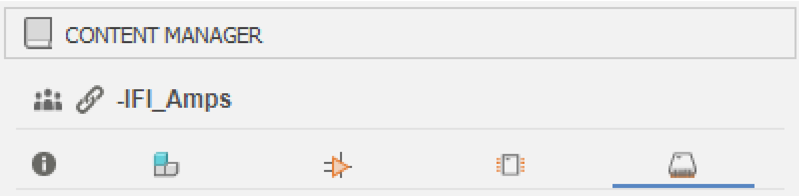


Figure 2: PCB Footprint

Packages:



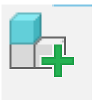
Packages can be found in the content manager and is a 3D representation of the component.

Figure 3: 3D Package

4. How is a schematic symbol and footprint linked?

Asssumptions:

- Schematic symbol is already created.
- PCB footprint is already created.

Create new component  . Give the component a name.

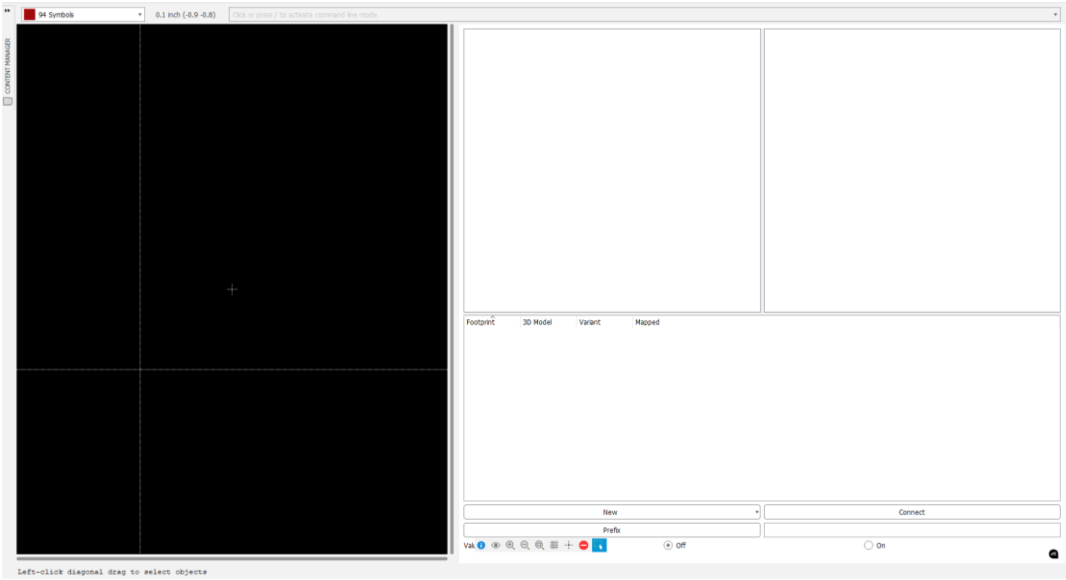
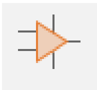
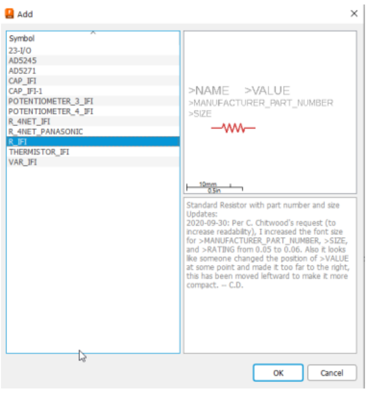


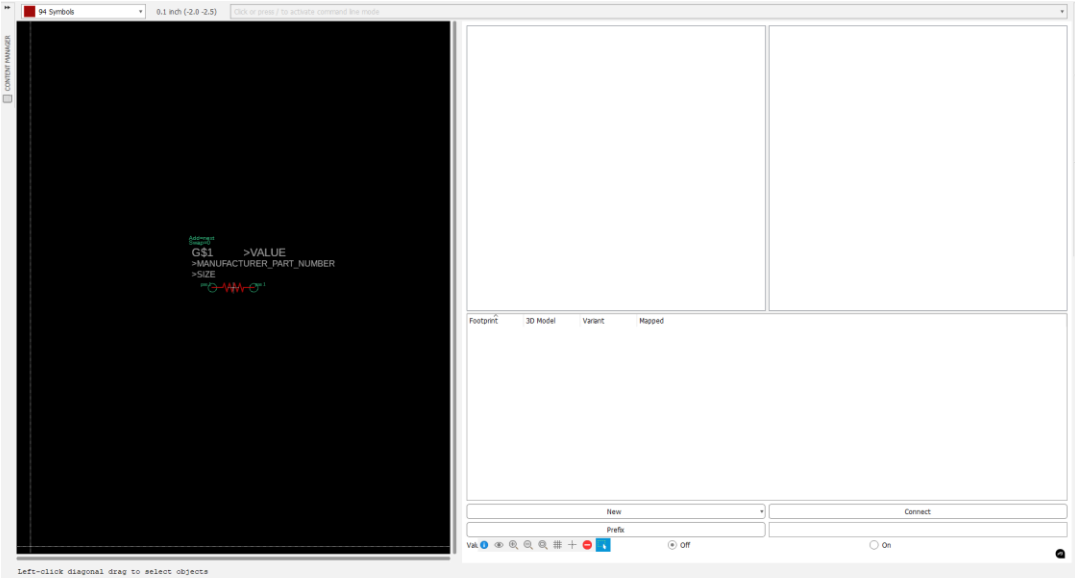
Figure 4: Blank Component

Add a schematic symbol to the component  .

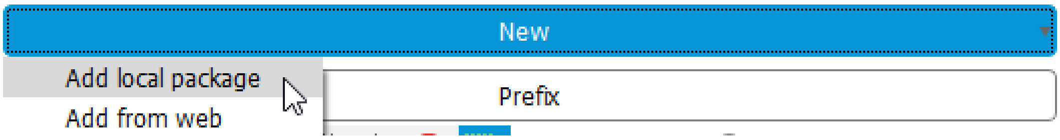


Select a symbol and click ok.

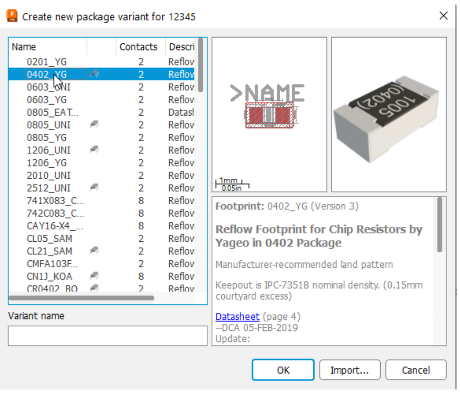
Place the schematic symbol on the origin as shown below.

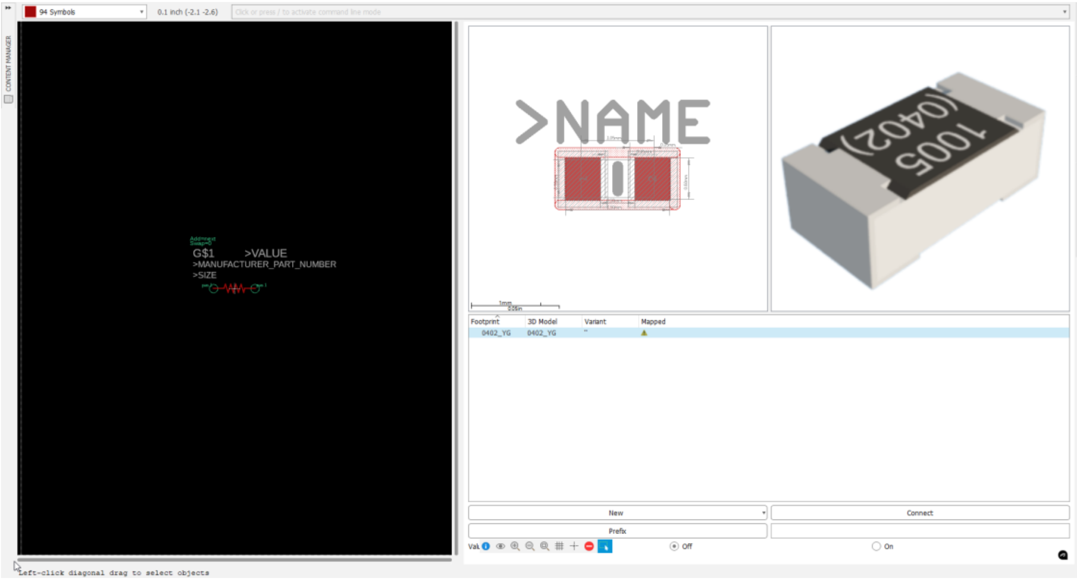


Add a footprint to the component.

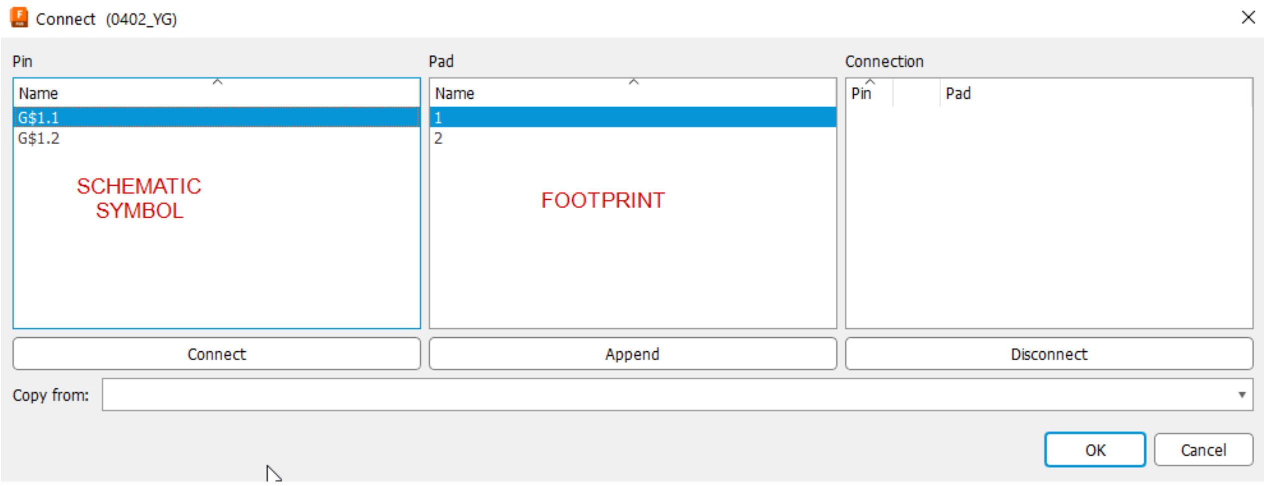


Select the appropriate footprint. Click OK.

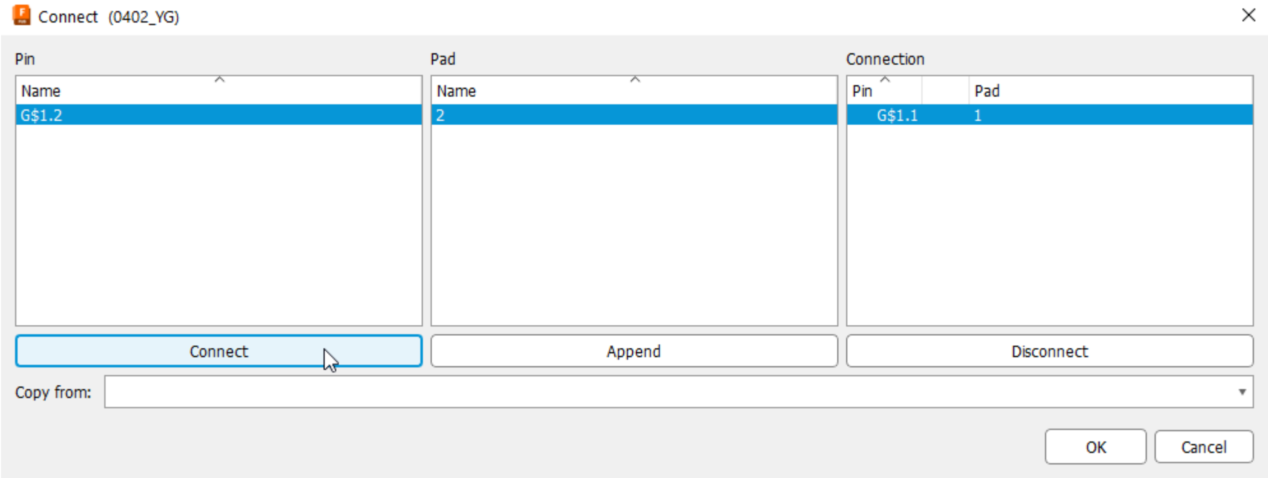




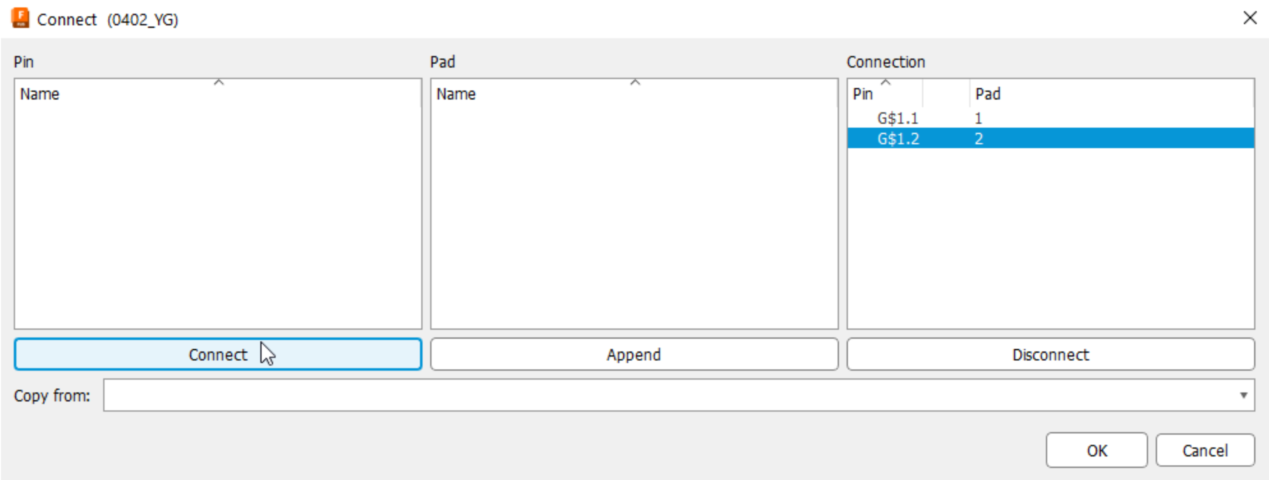
Connect the schematic symbol to the footprint.



Connect schematic symbol pin 1 (G$1.1) to footprint pin 1.



Connect schematic symbol pin 2 (G$1.2) to footprint pin 2.



Click OK. Component is ready to use.