University of California, Santa Cruz
Computer Engineering Department

PCB EDA Tutorial
Laboratory 2 Addendum 1
Working With Allegro Footprints

EE174, Intro to EDA Tools
S.C. Petersen

1. INTRODUCTION AND OBJECTIVES

This section continues the earlier discussion begun in Lab-1, section 5, where Allegro's Package editor was used to manage existing footprints and for mapping specific footprints back into Capture for later use in Allegro. Unfortunately, actually creating or modifying footprints required a deeper understanding than one possessed at that time, even though the logical point in the design flow for doing this is usually during the schematic capture phase when symbols are being tied to their PCB footprints. Thus, having completed at least one board, it should now be possible to grasp the fundamentals of making footprints. We will create two, one with vias and one without.

Creating footprints is essentially an exercise in properly translating dimensioned mechanical drawings. Knowing how to read mechanical drawings (aka "prints"), whether 3-view, isometric or orthographic, makes translating such drawings into a well-designed footprint relative easy. Such necessary drawings are usually found at the end of a component's datasheet, where several different ways of dimensioning parts can be encountered. It is worthwhile to be familiar with these standards before beginning.

DECIMAL INCH STANDARD – Many domestic manufacturers use this standard, where an inch is divided into 0.001 inch increments (also known as a "mil" or thousandth of an inch), so there are 1000 mils per inch. This is a very easy and sane way to guage sizes, since the "decimal" part is easily read from a mechnical scale or micrometer.

METRIC STANDARD – By contrast, most foreign manufacturers use this standard, where SI units are used and generally expressed in millimeters. Parts dimensioned in mm usually require some numeric conversion to decimal inches. A few conversions, accurate to 3 significant figures, that you should memorize are:

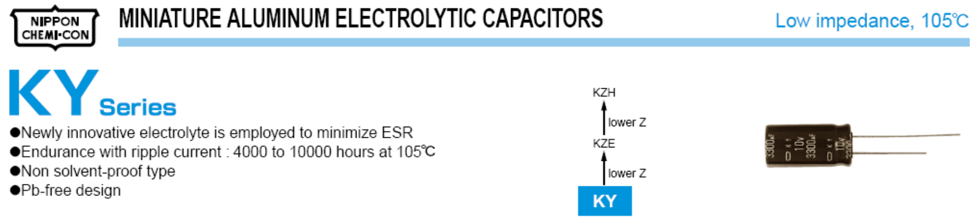
- $25.4\ mm = 1.00\ inch$ or $1000\ mils$, hence it follows by inspection that:
- $2.54\ mm = 0.100\ inch$ or $100\ mils$
- $(2.54mm/2) = 1.27\ mm = 50\ mils$

Many components are created with 50 or 100 mil pin or wire spacings, so a quick look at a set of metric dimensions will instantly reveal this. Parts with true millimeter dimensions, like 2.00 mm must be converted to decimal inches with a calculator.

2. CREATING A THROUGH-HOLE FOOTPRINT – In this example, we will create a new footprint for a 1uF/ 50V aluminum electrolytic radial-leaded capacitor. Actually, a suitable footprint can be found in the *pcb_lib_Orcad* **TM_CYCLND** library, needing at most a change in drill size. But we will create it from scratch to show how things work. This capacitor requires two plated through-hole padstacks for physically mounting and electrically connecting to it.

The first step is to have or make a mechanical drawing for the part. Yes, believe it or not, we sometimes must resort to measuring a specimen. This is not as accurate as a real drawing provided by a manufacturer, but often works just as well. Having a linear dial micrometer accurate to 0.001 inch is quite useful for this, especially for determining appropriate drill sizes from measured wire diameters.

An excerpt from the datasheet for the capacitor is shown below. The component datasheet was downloaded from Digikey while I was finding and ordering this part related to one of the final projects. It can be found in **…lab4_projects\audioamplifier\datasheets\ElectrolyticCap.pdf,** since it's used particularly in the audio amp project (other projects too), where it's cited with different values (and different possible sizes, each requiring their own footprint).



NIPPON CHEMI-CON

MINIATURE ALUMINUM ELECTROLYTIC CAPACITORS

Low impedance, 105℃

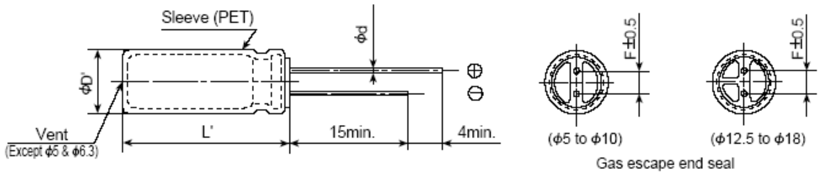
KY Series

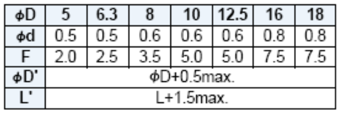
- ●Newly innovative electrolyte is employed to minimize ESR
- ●Endurance with ripple current : 4000 to 10000 hours at 105℃
- ●Non solvent-proof type
- ●Pb-free design

The screen shot following is a two-dimensional (2-D) view, *i.e.* two views of the usual three possible: a left-side view (on the left) and bottom view (on the right). Classic 2-D views are left, right and top or bottom views. Be careful to distinguish between top and bottom views; here it doesn't matter since the capacitor's radial-leaded wires are symmetrical about the axis of revolution. Note the shorter lead is negative however.

◆DIMENSIONS [mm]

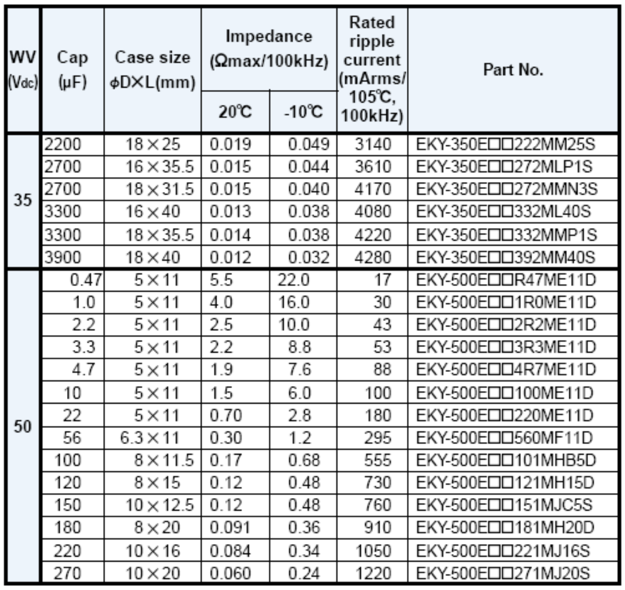
●Terminal Code : E





ϕD	5	6.3	8	10	12.5	16	18
ϕd	0.5	0.5	0.6	0.6	0.6	0.8	0.8
F	2.0	2.5	3.5	5.0	5.0	7.5	7.5
ϕD'	ϕD+0.5max.						
L'	L+1.5max.						

◆STANDARD RATINGS



WV (Vdc)	Cap (μF)	Case size ϕD×L(mm)	Impedance (Ωmax/100kHz)		Rated ripple current (mArms/ 105℃, 100kHz)	Part No.
			20℃	-10℃		
35	2200	18×25	0.019	0.049	3140	EKY-350E□□222MM25S
	2700	16×35.5	0.015	0.044	3610	EKY-350E□□272MLP1S
	2700	18×31.5	0.015	0.040	4170	EKY-350E□□272MMN3S
	3300	16×40	0.013	0.038	4080	EKY-350E□□332ML40S
	3300	18×35.5	0.014	0.038	4220	EKY-350E□□332MMP1S
	3900	18×40	0.012	0.032	4280	EKY-350E□□392MM40S
50	0.47	5×11	5.5	22.0	17	EKY-500E□□R47ME11D
	1.0	5×11	4.0	16.0	30	EKY-500E□□1R0ME11D
	2.2	5×11	2.5	10.0	43	EKY-500E□□2R2ME11D
	3.3	5×11	2.2	8.8	53	EKY-500E□□3R3ME11D
	4.7	5×11	1.9	7.6	88	EKY-500E□□4R7ME11D
	10	5×11	1.5	6.0	100	EKY-500E□□100ME11D
	22	5×11	0.70	2.8	180	EKY-500E□□220ME11D
	56	6.3×11	0.30	1.2	295	EKY-500E□□560MF11D
	100	8×11.5	0.17	0.68	555	EKY-500E□□101MHB5D
	120	8×15	0.12	0.48	730	EKY-500E□□121MH15D
	150	10×12.5	0.12	0.48	760	EKY-500E□□151MJC5S
	180	8×20	0.091	0.36	910	EKY-500E□□181MH20D
	220	10×16	0.084	0.34	1050	EKY-500E□□221MJ16S
	270	10×20	0.060	0.24	1220	EKY-500E□□271MJ20S

Towards the end of the datasheet we find the wanted 1uF/50V dimensions; an excerpt is shown. We note the case size, ϕDXL = 5 X 11 [mm]. Hence our dimensions are:

Width, $\phi \mathrm{D} = 5\ mm = 196.85\ inch \approx 200\ mils$

Lead Spacing, $\mathrm{F} \pm \phi \mathrm{d}\ mm = 2.0 \pm 0.5 \approx 100\ mils$.

Height,

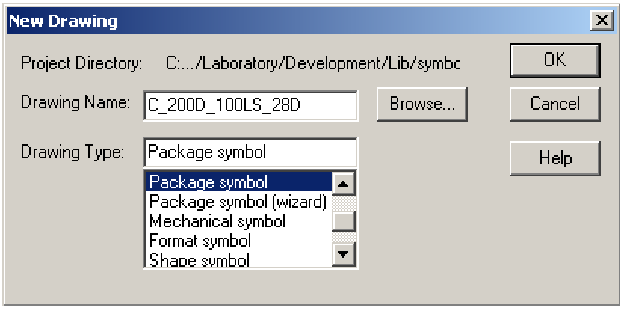
$$\mathrm{L} + 1.5\max = 11.5 = \frac{11.5\ mm}{25.4\ \frac{mm}{inch}} = 453\ mils \approx 0.5\ inch$$

Wire diameter =

$$\phi \mathrm{D} = \frac{0.5\ mm}{25.4\ \frac{mm}{inch}} = 19.685\ mils \approx 20\ mils$$

Adding 3 *mils* for plating allowance gives us 23 *mils* minimum; the neareast APC drill size above this is 28 *mils*, so we will use a 28 *mil* drill.

Launch Allegro PCB Designer and start the new footprint by choosing **File** ⟶ **New** (or use the icon)**.** Specify *Package Symbol* in the *Drawing Type* list. This changes Allegro to "Symbol mode", rather than "Board mode" you used in the previous lab. Anytime we create a new symbol, as we are here, or otherwise open any symbol drawing file (having extension *.dra), Allegro changes to **symbol mode**. Remember, this mode is tailored for use with *footprints* (Cadence calls them "packages") so it necessarily lacks some of the features usually available when working with boards, notably only *Etch edit*, *General edit* and *Symbol edit* modes will be active. Since you should already be familiar with Allegro in Board mode, these will obvious by their absence. Especially note that symbol mode works with only a subset of the classes, namely, those most directly applicable to individual footprint symbols.



New Drawing

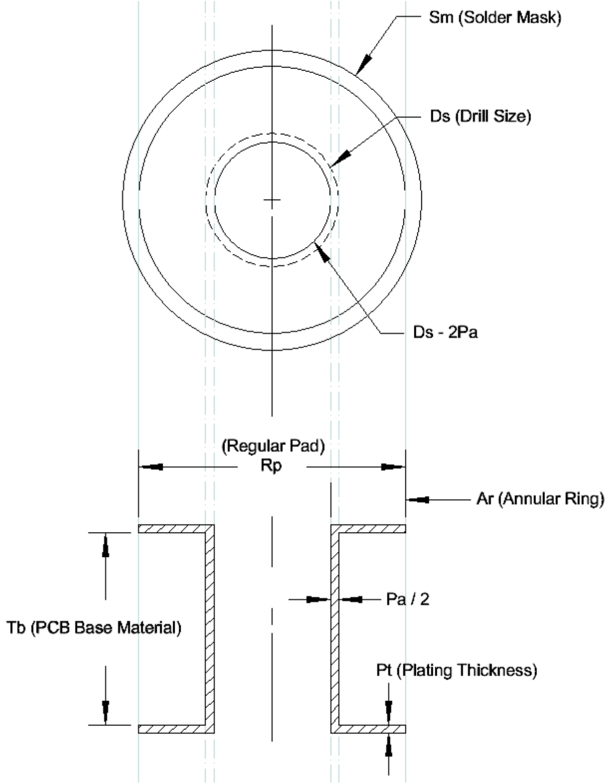
Project Directory: C:.../Laboratory/Development/Lib/symbc

Drawing Name: C_200D_100LS_28D

Drawing Type: Package symbol

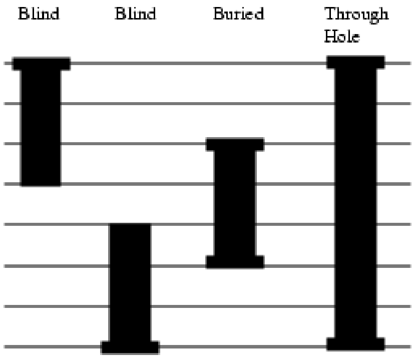
We must also specify a name for our new footprint. Naming footprints is really something of an art. Perusing Allegro's and the older imported Oracd libraries reveals that most are generic; *i.e.*, they apply to a large class of components having the same physical footprint. We will follow that approach here and avoid naming it too specifically, like "1uF_50V_Cap". The name should, however, fully encapsulate the various mechanical parameters noted above. Hence, it needs to refer to a capacitor with a diameter of 200 mils, lead spacing of 100 mils and 28 mil drills; so we'll enter "C_200D_100LS_28D" for the name. Later, when you're looking for a suitable footprint, the name will go a long way to defining what the package symbol applies to. After pressing *OK*, Allegro will enter symbol mode. Also, be sure the Project Directory is your working symbols folder (browse to it if necessary) and check *Change Directory*. Import parameter file "symbol_mode.prm" to set Allegro's default colors and text sizes when working in this mode. These parameters will be saved with the new part. You can change them later if you like. Be sure to set the *Extents* in the *Design Parameter Editor* to something small, like 2000 by 2000 mils; you should also be working in mils with 1 decimal place accuracy, and make the non-etch grid 50 mils.

Making Padstacks – The first step is to select or create the necessary padstacks. Since this is a polarized capacitor having two mounting wires, it obviously needs two 28 mil padstacks, but one will be square, the other round to distinguish plus and minus. Create both padstacks now, saving them in your …Handset\lib\padstacks folder. Doing this requires that we open the *Padstack Designer* briefly discussed in the last lab. The best way to proceed is to choose a padstack that's close to what is wanted, place it on the new footprint, and then modify it. Before doing that, study the simple 2-view orthogonal drawing showing most of the basic padstack dimensional variables. Briefly, these are defined as follows:



- **Ds** = Drilled hole size, 28 mils in our case.
- **Pa** = Plating allowance *diameter*. Amount of additional copper plated inside the raw drilled hole; typically this is taken to be 3 mils. Since this is a diameter, the *thickness* is 1.5 mils or Pa / 2 as shown.
- **Pt** = the original board plating thickness *before drilling*. This can be calculated from the board's area plating density, typically given as 2 or 3 oz. copper per square inch.
- **Ar** = Top or bottom-side annular ring width, $A_r = \dfrac{R_p - D_s + P_a}{2}$, typically 10 mils or more.
- **Rp** = Regular pad diameter (top and bottom). This dimension can apply to inner etch layers (not shown) as well. This can be other shapes besides round (square, oblong etc.).
- **Sm** = Soldermask *antipad* diameter. Typically at least 5 mils larger than the regular pad diameter it surrounds.
- **Tb** = Unplated raw PCB base material, typically 60 mils for FR4.

Padstacks are automatically classified by Allegro depending on what layers they may potentially connect:



The figure shown[1] summarizes the first three of the four classic padstack types:

1. **Blind vias** are only partially drilled, but start on the top or bottom-sides (hence "blind" on one side).

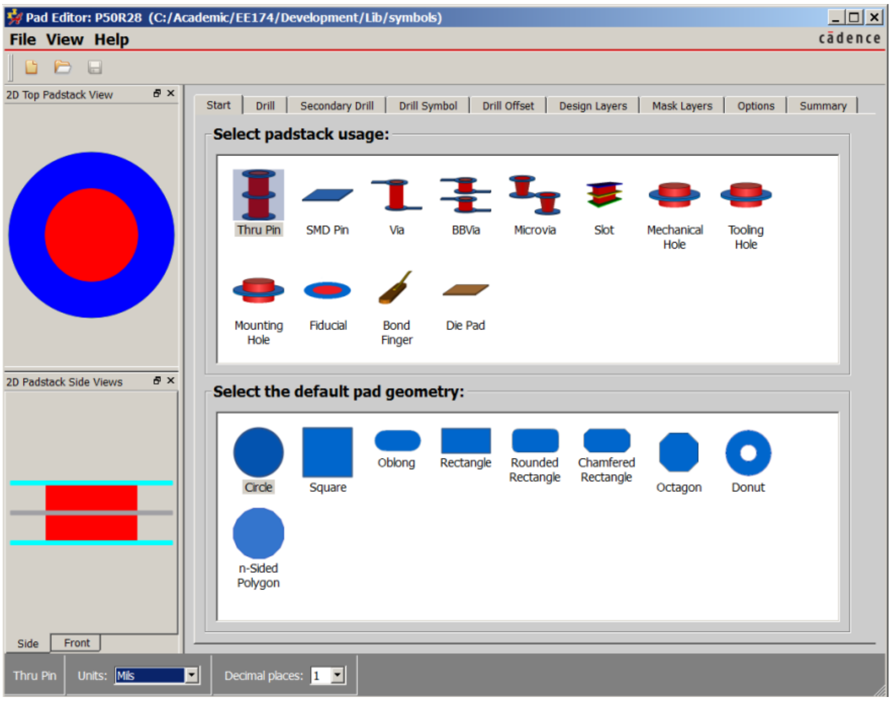
2. **Buried vias** connect only inner layers and are "blind" to from both sides.

3. **Through-hole** padstacks are drilled through the entire stackup.

4. **Single** padstacks have one pad (may be any shape) with no drilled hole. This is for surface-mount (SM) footprints.

Allegro also allows B/B ("blind and buried) vias to be specified as *Microvias*. These are holes, typically under 6 mils created with lasers for specific purposes, like fanning out high-density ball grid arrays. Moreover, up to 16 etch layers can be defined in a library padstack.

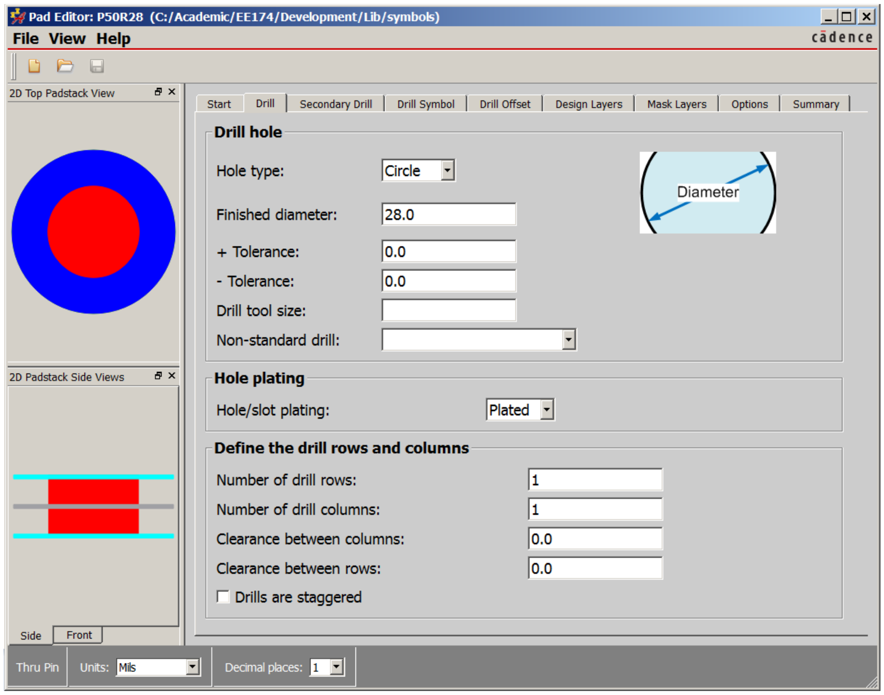
The library padstack we want to create now for this generic capacitor part is clearly a simple 2-layer through-hole type having equivalent regular pads and solder-mask diameters on both sides. Beginning with the drill size of 28 mils, the annular ring should be at least 10 mils. Therefore, sizing making the regular pads on each side 50 mils, results in $A_r = 12.5$ mils after plating or 11 before plating. The soldermask antipads (voided areas of enamel) should have 5 mil annular rings around the regular pads, making $S_m = 60$ mils by inspection. Choose menu sequence **Tools ⟶ Padstack ⟶ Modify Library Padstack** and choose any suitable padstack from Allegro's library. If you want to start with one of your own, then select **Modify Design Padstack** (if available). This gets you into the *Pad Editor*. Necessary entries to define a new round padstack using positive artwork are summarized below. PSB 17.2 has reorganized the way padstacks are defined and edited organizing parameters among nine group tabs:



For this part we will only be using the *Start*, *Drill Symbol*, *Design Layers* and *Mask Layers* options; look over the other tabs if you like, but we'll just leave them with their default settings. Select the *Start* tab. This allows us to specify the padstack type as "Thru Pin" and geometry as a "Circle". Notice the status line at the bottom of the form. It will reflect the chosen "usage". Units should be "mils" with 1 decimal place.

[1] Copied from Allegro PCB Editor User Guide: Defining and Developing Libraries, Ch-2, Library Padstacks.

Note that we're not specifiying any particular figure or character, since this would normally be defined locally after the padstack has been used in a particular design; you did this in the last lab. Select the *Drill* tab and specify the hole type, finished diameter and hole plating parameters as shown. Note that leaving the "Drill tool size" blank causes Cadence to use the number entered in the "Finished diameter" as the actual drill size.



Drill hole

Hole type: Circle

Finished diameter: 28.0

+ Tolerance: 0.0

- Tolerance: 0.0

Drill tool size:

Non-standard drill:

Hole plating

Hole/slot plating: Plated

Define the drill rows and columns

Number of drill rows: 1

Number of drill columns: 1

Clearance between columns: 0.0

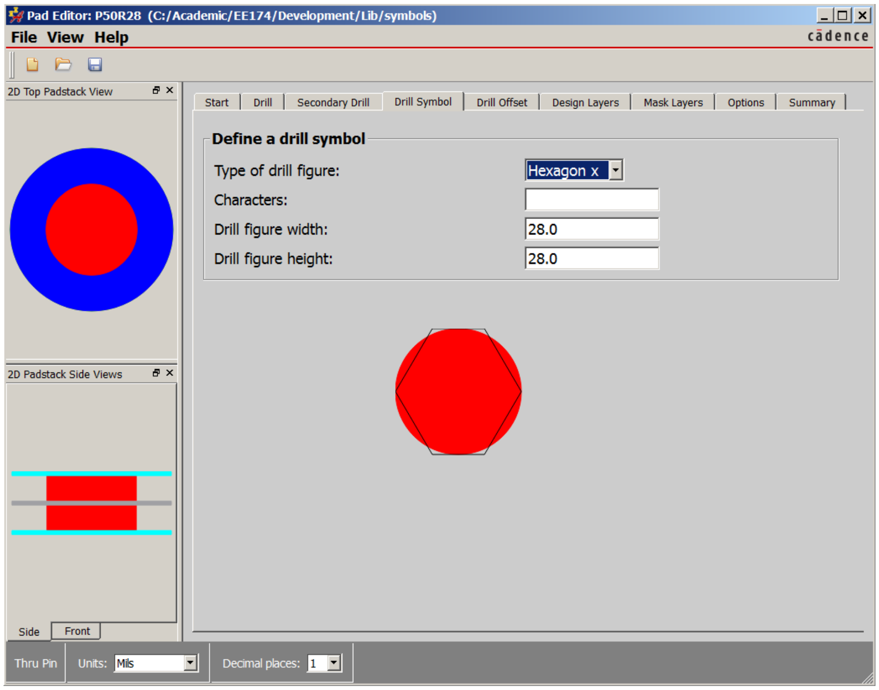
Clearance between rows: 0.0

☐ Drills are staggered

Units: Mils

Decimal places: 1

Select the *Drill Symbol* tab and set the "Type of drill figure" – that used in the drill legend – as "Hexagon x":



Define a drill symbol

Type of drill figure: Hexagon x

Characters:

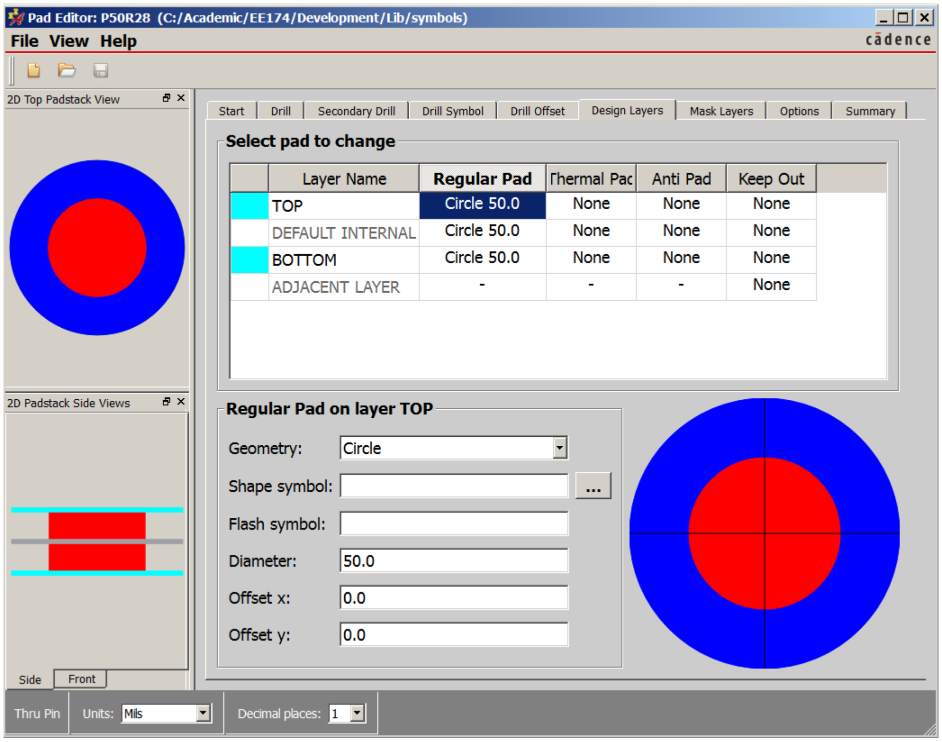
Drill figure width: 28.0

Drill figure height: 28.0

Units: Mils

Decimal places: 1

Select the *Design Layers* tab and edit the layers as shown. These are the etch layers consisting of top, bottom and any additional inner layers that may be used in a design having more than a 2-layer stackup. Edit each layer by first selecting it with the mouse in the *Regular Pad* column; this will populate the fields at the bottom of the form with that layer's current settings. All three pads should be specified as 50 mils. The internal default is often made slightly larger to allow for layer misregistration during the manufacturing process; typically this is set 10 mils larger. For your part, we'll just use 50 mils.



Pad Editor: P50R28 (C:/Academic/EE174/Development/Lib/symbols)

Select pad to change

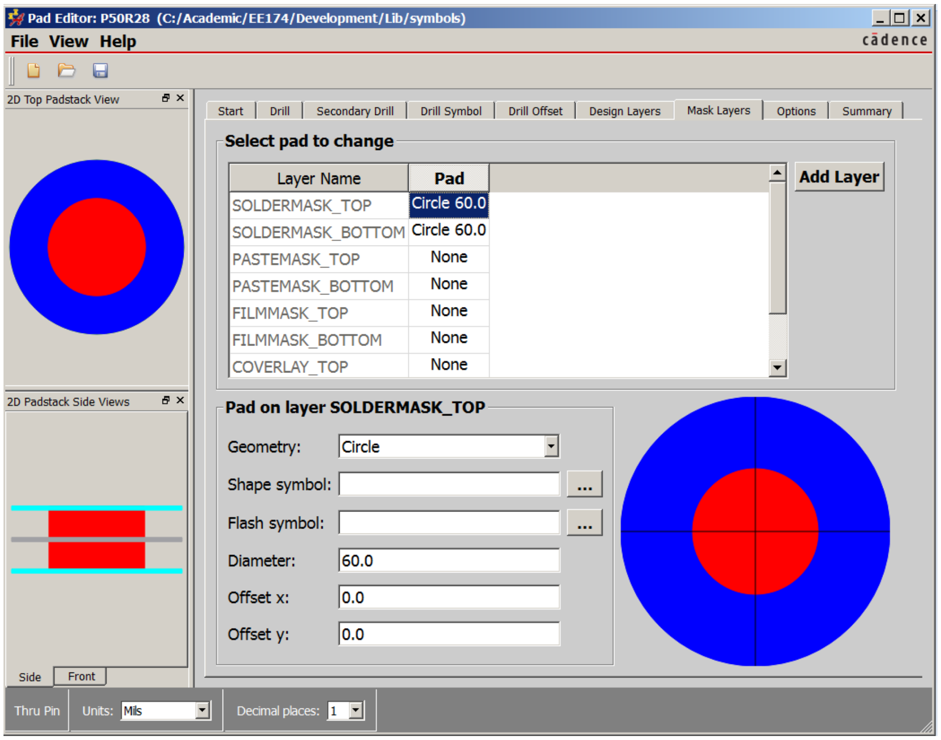
	Layer Name	Regular Pad	Thermal Pac	Anti Pad	Keep Out
🟦	TOP	Circle 50.0	None	None	None
	DEFAULT INTERNAL	Circle 50.0	None	None	None
🟦	BOTTOM	Circle 50.0	None	None	None
	ADJACENT LAYER	-	-	-	None

Regular Pad on layer TOP

Geometry: Circle
Shape symbol:
Flash symbol:
Diameter: 50.0
Offset x: 0.0
Offset y: 0.0

Thru Pin Units: Mils Decimal places: 1

Note the film-mask layers are not shown but they should be the same as those for the pastmask layers. The two columns: *Thermal Padf* and *Anti Pad* are only useful with *negative artwork* layers. An in-depth discussion of negative artwork is beyond the scope of these labs. Suffice to say that for positive artwork, Allegro uses autovoiding with dynamic shapes to create thermal reliefs for you. When negative layers are defined, copper pours are *assumed* to already exist and necessary thermal reliefs and voids are *not automatically* created for you; rather, they are defined by entries from these two latter columns. Since our stackup only specifies *positive artwork* layers, we won't pursue this topic further.



Pad Editor: P50R28 (C:/Academic/EE174/Development/Lib/symbols)

Select pad to change

Layer Name	Pad
SOLDERMASK_TOP	Circle 60.0
SOLDERMASK_BOTTOM	Circle 60.0
PASTEMASK_TOP	None
PASTEMASK_BOTTOM	None
FILMMASK_TOP	None
FILMMASK_BOTTOM	None
COVERLAY_TOP	None

Pad on layer SOLDERMASK_TOP

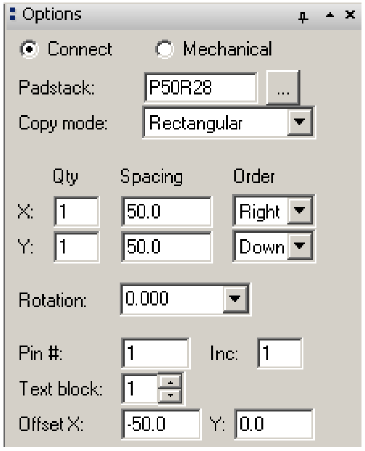
Geometry: Circle
Shape symbol:
Flash symbol:
Diameter: 60.0
Offset x: 0.0
Offset y: 0.0

Thru Pin Units: Mils Decimal places: 1

Finally, select the *Mask Layers* as shown and define the top and bottom soldermask layers. Remember, these are applied as a single layer of enamel paint (typically green) having openings of the type and size specified in the "Pad" column. Making these 10 mils larger than the interior copper pad leaves 5 mils of open space (an annular ring) before the edge of the enamel coating. As you already know, this provides a means to apply solder to these pads.

When you're finished, choose the Designer's menu sequence **File⟶Save As…** and save it as "P50R28.pad" in your padstacks' folder. This padstack will now show up in the library list of those available along with Allegro's *and* any padstacks that happen to already be present in your working directory (this should be your symbols' folder). Later, after the padstack has been placed (instanced), it will then appear as a *design* padstack, *i.e.* one that is part of your current design work.

Go ahead now and instance or place this first padstack as pin-2 in the new footprint (the convention is square pins are always pin-1; we'll create and place that next). Set the non-etch grid to 50 mils and choose **Layout ⟶ Pins** (or use the icon) to change the context. Then use the Options' panel to select the new padstack and place it anywhere you like. Note the fields in this context are very powerful. Array's of pins can quickly be created with specific geometries. If you were creating a part that had multiple pins, you could also place them manually one-by-one and have the pin numbers imcrement automatically. It's worth the time to experiment with these settings to get familiar with what the fields do and perceive how they can best be used. One critical point that needs to be here is the distinction between *Connect* padstacks and *Mechanical* padstacks. The former are treated as electrical objects having pin numbers associated with nets, while the latter are not. The 125 mil mounting holes you placed at the beginning of the last lab are an example. It is possible to change pin types after a padstack appears in a design.



● Connect ○ Mechanical

Padstack: P50R28 ...

Copy mode: Rectangular

	Qty	Spacing	Order
X:	1	50.0	Right
Y:	1	50.0	Down

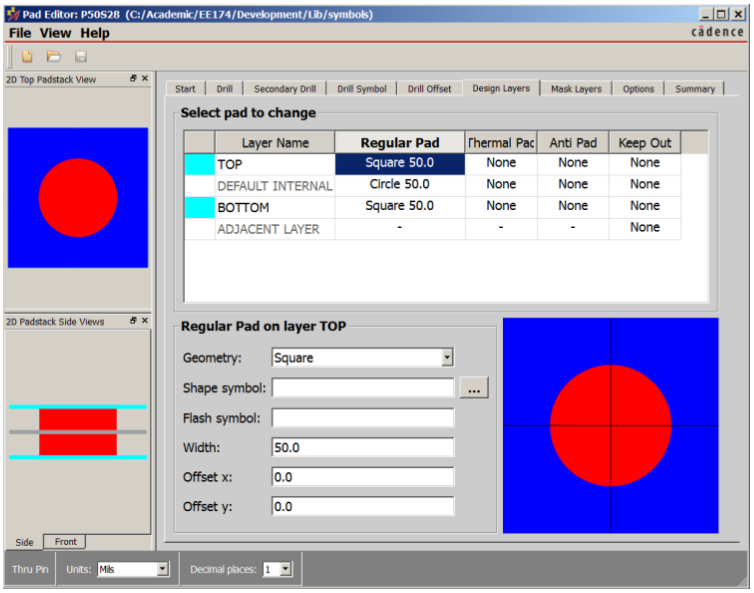
Rotation: 0.000

Pin #: 1 Inc: 1

Text block: 1

Offset X: -50.0 Y: 0.0

Now create another padstack having the same drill except make the pads square instead of round. The easiest way is to modify the one you just made, and save it as "P50S28.pad". From the *Start* tab change the default pad geometry to square, and from the *Design Layers* and *Mask Layers* tabs specify the pad geometry to now be square as shown following below. Note that since the new padstack is now part of the footprint, you can choose **Tools ⟶ Padstack ⟶ Modify Design Padstack** rather than the library; this will shorten the selection list considerably.



Select pad to change

	Layer Name	Regular Pad	Thermal Pac	Anti Pad	Keep Out
🟦	TOP	Square 50.0	None	None	None
	DEFAULT INTERNAL	Circle 50.0	None	None	None
🟦	BOTTOM	Square 50.0	None	None	None
	ADJACENT LAYER	-	-	-	None

Regular Pad on layer TOP

Geometry: Square

Shape symbol:

Flash symbol:

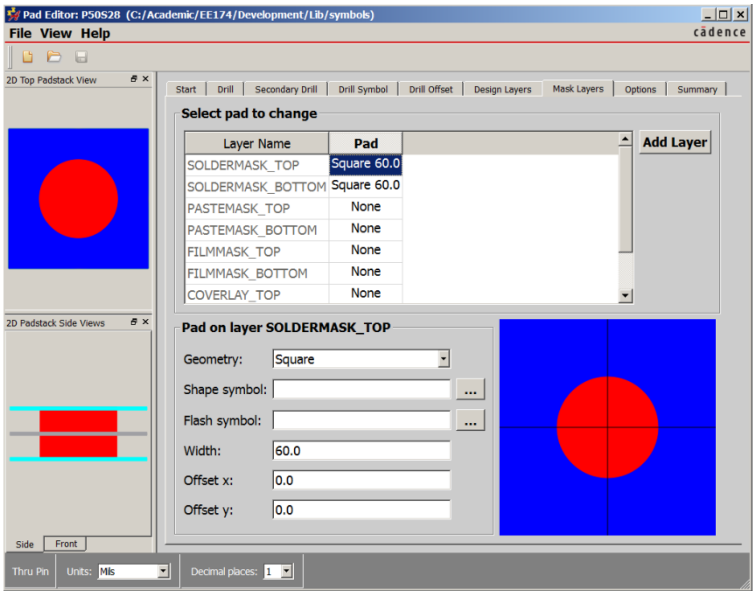
Width: 50.0

Offset x: 0.0

Offset y: 0.0

Units: Mils

Decimal places: 1



Select pad to change

Layer Name	Pad
SOLDERMASK_TOP	Square 60.0
SOLDERMASK_BOTTOM	Square 60.0
PASTEMASK_TOP	None
PASTEMASK_BOTTOM	None
FILMMASK_TOP	None
FILMMASK_BOTTOM	None
COVERLAY_TOP	None

Pad on layer SOLDERMASK_TOP

Geometry: Square

Shape symbol:

Flash symbol:

Width: 60.0

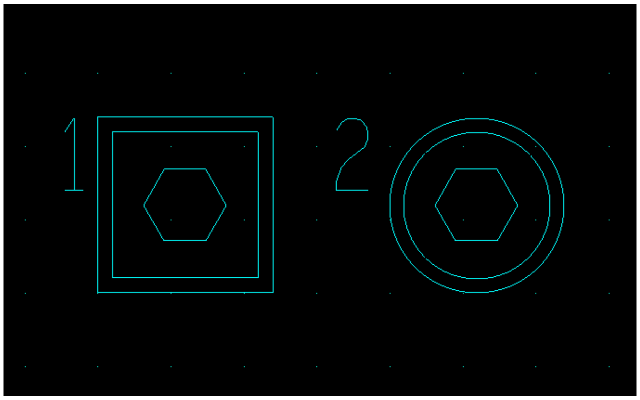
Offset x: 0.0

Offset y: 0.0

Units: Mils

Decimal places: 1

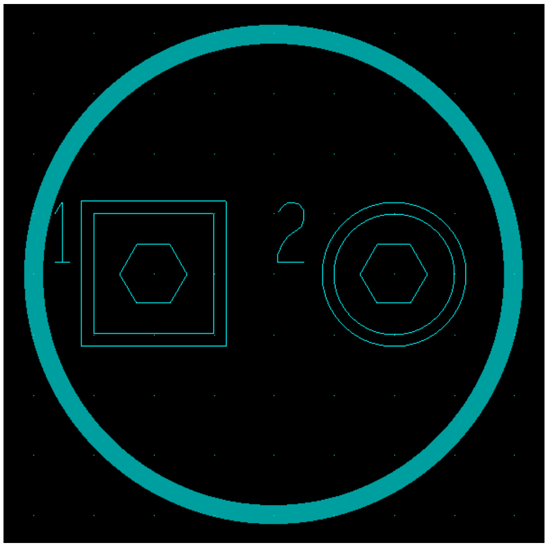
After saving the new square padstack, place it as pin-1 100 mils from the round pin (take advantage of *Relative XY mode* to measure out the 100 mils for the next pin placement). Your two new padstacks should look similar to that shown (I moved the text around a little after placement).



We now need to make two line drawings as follows. You should already know how these are used from the last lab:

- **Silk-screen drawing** of that will appear on the finished PCB, and placed on the Package Geometry / SST layer. This will be a 200 mil diameter line drawing with 8 mil line width.

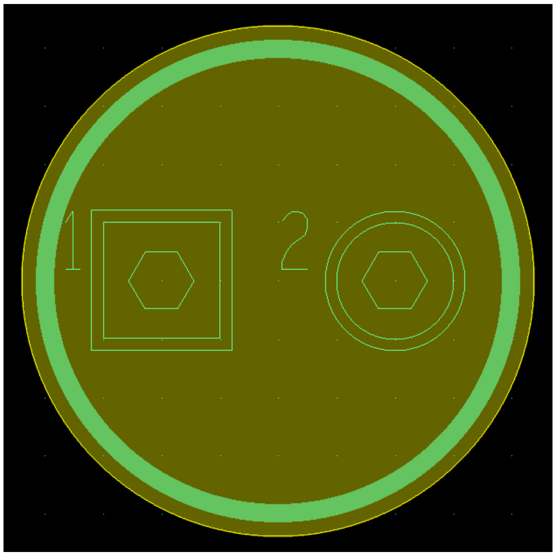
- **Assembly top drawing** of the component's outline on the Package Geometry / AST layer. For this part, we'll make the AST drawing the same as the SST. Generally, it doesn't have to, since it never actually appears on the physical PCB.



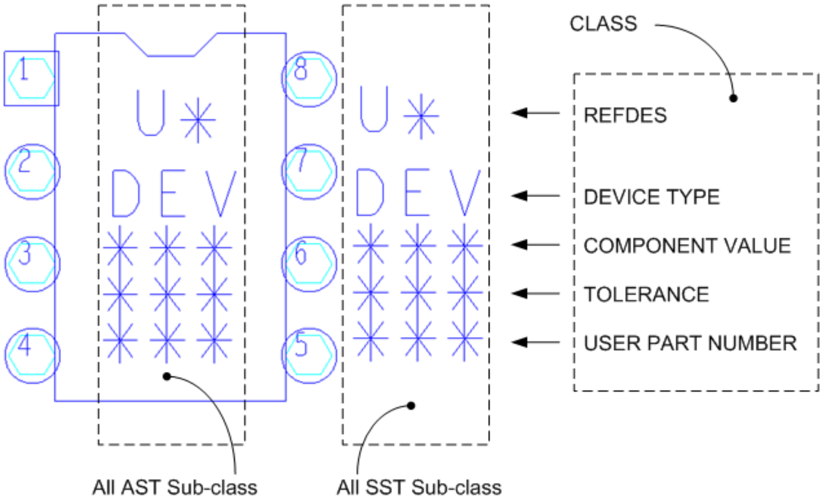
Draw both of these using the *Circle* tool: follow **Add → Circle**. Set the *Line width* to 8 and the *Line font* to Solid in the Options' panel. Copy the circle from between layers or draw it again.. Note that Allegro's footprint's are 0 mil lines, but this just means we have to manually change them after placing them, so it's better to define them the way we want now. Generally, SST draws should be greater than 5 mils or the ink won't show properly; 8 mils is a good nominal size to use, but the AST thickness is up to you.

The result is shown to the left.

Adding the Placement Outline – Every part must have a filled shape representing the part's physical placement boundary placement boundary. This provides the DRC and potential autorouter a means to identify what board area is occupied by the component. These boundaries can be defined for either the top or bottom or both, depending on physical characteristics of the component to prevent overlap of adjacent components (which will also have their own place outlines). We'll place only a single shape on the Package Geometry / Place_Bound_Top layer. Since the part's diameter is 200 mils, a 220 mil diameter round boundry about footprint should be sufficient. Do this using a filled round shape. If you're still not sure how to do this, first set the class / sub-class and then choose **Shape → Circular** (or use the icon); set the *Shape Grid* in the Options' panel for 10 mils and start at the very center of the part moving the mouse out 110 mils. Be sure to use Relative XY mode! It should like the graphic shot below.

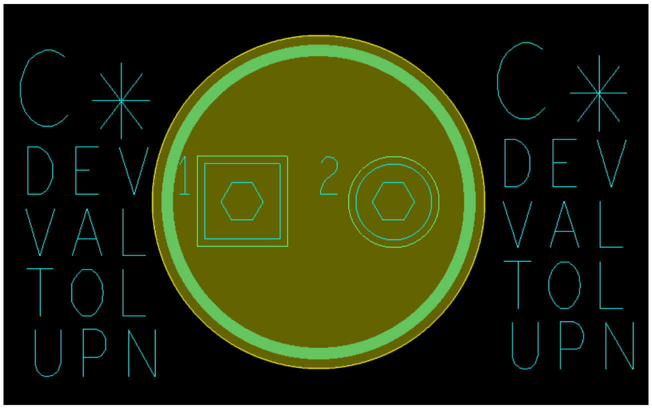


The last step is to add mandatory and any optional text.[2] The graphic below nicely summarizes the usual text that normally appears in most of Allegro library symbol parts (ref.: dip8_3.dra). Study this carefully and be sure you really understand it.

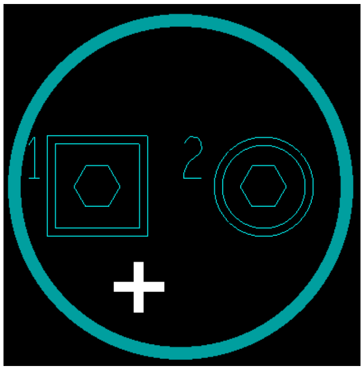


Allegro requires there be at least one instance of text in the REFDES class; the others are optional. And it doesn't really matter what is actually entered, since it serves merely as a place holder indicating a field; the string substitution is based only on the class. Thus, we will enter strings having more meaning than just asterisks. Before proceeding, note the assembly-top fields are inside the part outline, while the silk-screen fields are outside. This makes sense, since the SST is mean to actually appear on a PCB, while the AST isn't. Enter text fields now, arranging them about the part, based on the class / sub-classes noted above and text strings indicated as follows:

• Refdes	**C***
• Device Type	**DEV**
• Component Value	**VAL**
• Tolerance	**TOL**
• User Part Number	**UPN**



Make the reference designators text size 3 and all the rest 2. Your text should look something like that shown. The rightmost column is all the silk-screen fields; those on the left are assembly-top.

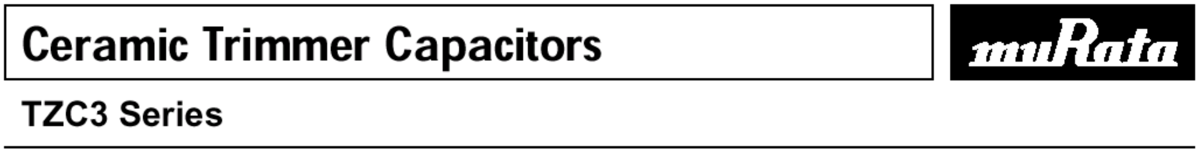


As a final touch, identify the the positive pin by placing "+" signs near pin-1 on both the Package Geometry SST and AST layers. Create these as two orthogonal line segments 30 mils long having 6 mil widths. Choose **Add ⟶ Line** (or use the icon shown)**.** Since lines follow the non-etch grid setting, change it to 5 mils for this operation so you can draw a nice looking "+" sign.ith the non-etch grid still to 5 mils, re-position the segments the way you would like it to appear when Allegro loads the footprint. When finished, save the new footprint.

[2] For a general discussion, refer to Allegro PCB Editor User Guide: Defining and Developing Libraries, Working with Symbols, pgs. 29-31.

Defining the Drawing Origin – Every footprint must have a reference point to be used whenever it is selected for placement or for moving after being placed. When creating a new part from scratch, this *drawing origin* is usually nowhere near the part and must be moved to a relevant point somewhere on or nearby. Whenever a footprint is selected for moving, Allegro will instantly relocate the part so its drawing origin is now at the mouse point's position when it was first selected. Thus, it behooves us to know where this point is beforehand, and so avoid the annoying side-effect of selected parts jumping about on the screen. Generally, the center of pin-1 is most often used for this purpose, since we also can usually quickly find this point by inspection. The drawing origin can easily be moved by following **Setup → Change Drawing Origin**. *Be sure the point you want coincides with the grid point used to place pin-1* to ensure proper part alignment with the placement grid used when the part is actually placed or later moved about.

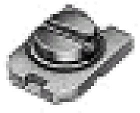
3. CREATING A SURFACE MOUNT FOOTPRINT – For this example, we will create a new footprint for the FM Transmitter's 5-20 pF VCO tuning capacitor. Excerpts from the datasheet for MuRata's TZC3 series RF trimmer capacitors are shown below. This example also assumes you have already learned how to create a footprint with vias in the preceding section 6.2

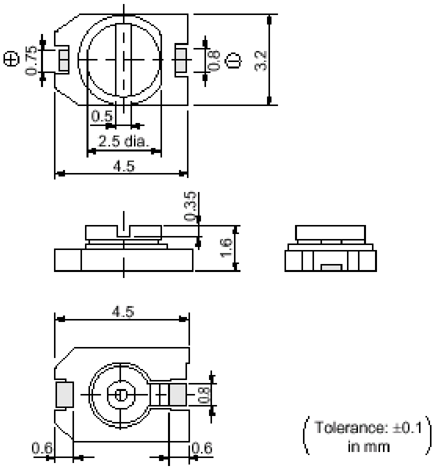

Ceramic Trimmer Capacitors

muRata

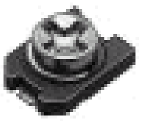
TZC3 Series

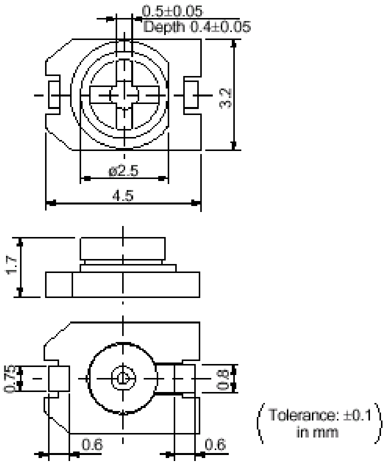
The datasheet was downloaded from Digikey while I was finding and ordering this part for the FM Transmitter project. You can find it in **…lab4_projects\FM Transmitter\datasheets\TZC3 Variable RF caps.pdf.**





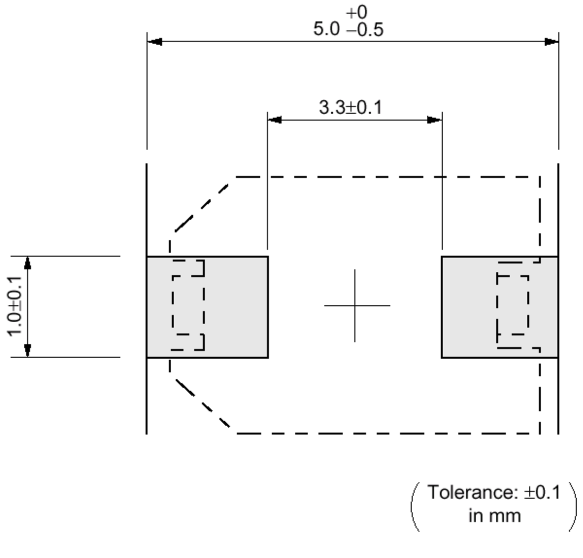
Standard Type





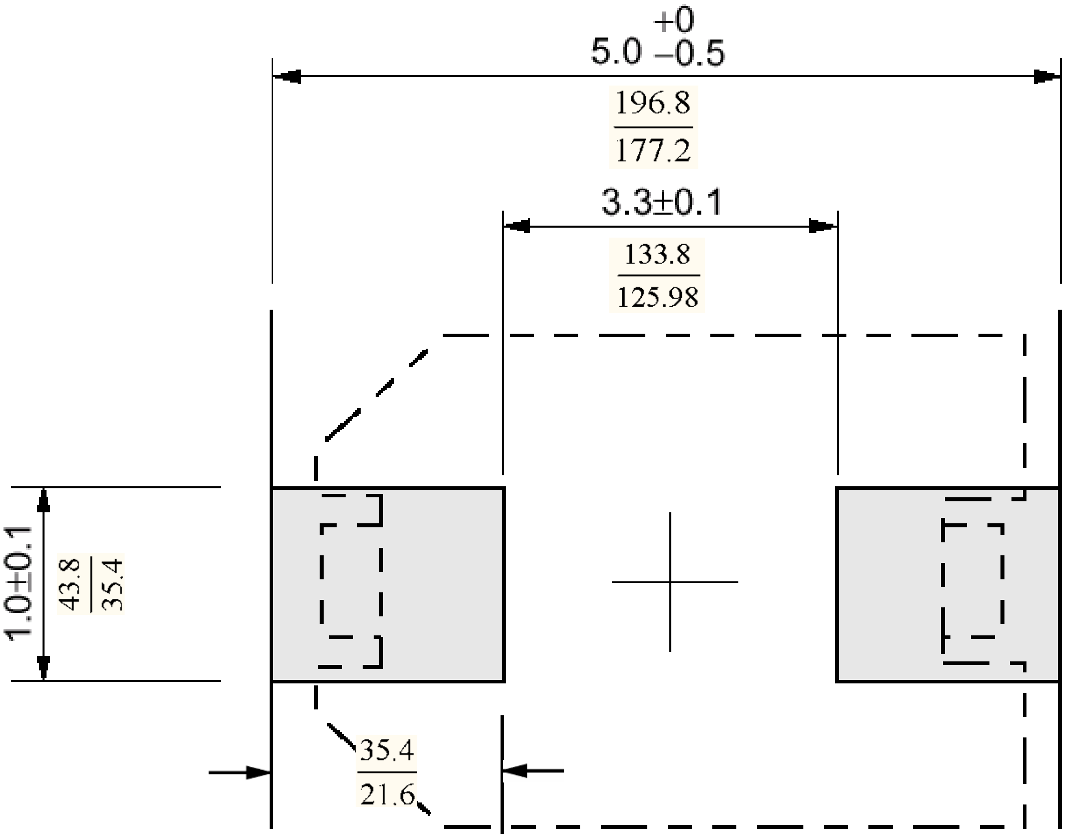
Cross Slot Type

Both the Standard Type and Cross Slot Type are shown to the left. A careful comparison of the recommended PCB footprint found later in the datasheet ("Land Pattern"), reveals that MuRata recommends the same footprint for both as shown below.



I have experience with this component and will change the recommended SMT pad sizes somewhat.

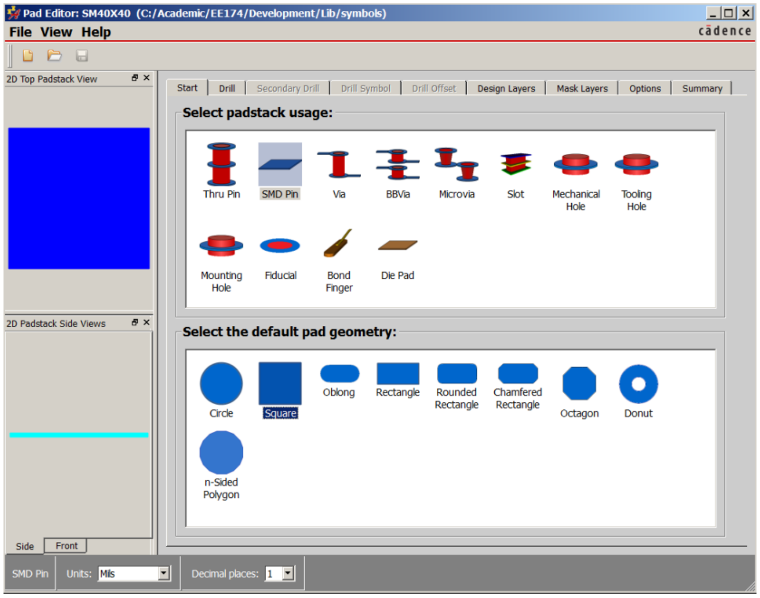
Since I like to work in decimal inches, the first step is to annotate directly on the mechanical drawings what the various toleranced dimensions are in mils. For example, the top dimension shows the pad spacing to be 5.0 mm with a tolerance of +0, -0.5 mm; this just means it should lie between 4.5 and 5.0 mm. Converting this to mils yields 177.2 to 196.8 mils. Similarly for the 3.3 mm dimension, 3.2 to 3.4 mm translates to 126.0 to 133.8 mils. And the width of 0.9 to 1.1 mm translates to 35.4 to 43.8 mils. The missing pad lengths have to be deduced from the other dimensions. Hence, it will lie between 21.6 and 35.4 mils (calculation details are left to the interested student). This technique of making dimension notes directly on the drawing and keeping them in your engineering notebook is very useful.



We will make the pads 40 mils square and space them so their inner vertical edges are 130 mils apart.
Create a new part named “C_RF_TZC3” following following the same procedure as for the through-hole part (see page-2). Set the non-etch grid resolution for 10 mils and, if you haven’t done so already, import parameter file “symbol_mode.prm”.

Again, following the same general procedure as we did earlier, open and modify one of the library surface mount padstacks. When you’re done rename it to refer to a 40 mil square SM pad, “SM40X40” and save it your padstacks folder. Note our naming convention is based on what we did with the earlier part. Since this will be a surface mount padstack, the only layers needing anything on them will be those related to the top layer.

Following are screen-shots summarizing how to do this using the *Padstack Editor*:



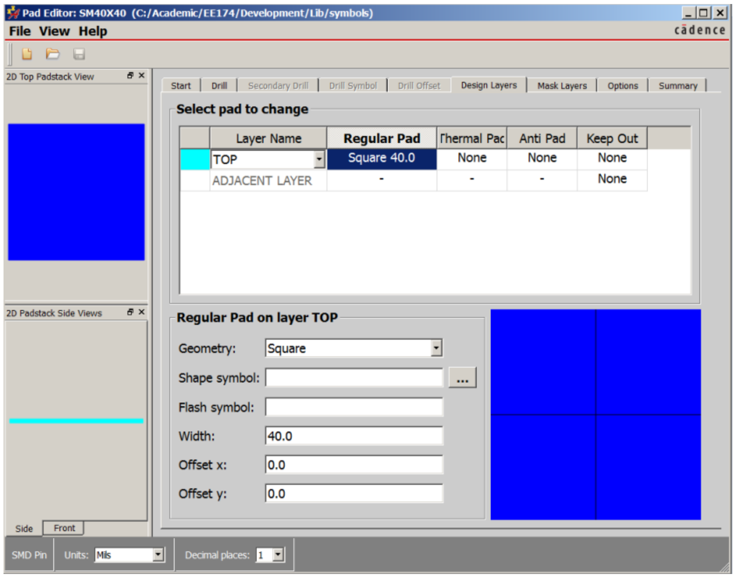
Pad Editor: SM40X40 (C:/Academic/EE174/Development/Lib/symbols)

Select padstack usage: SMD Pin

Select the default pad geometry: Square

SMD Pin Units: Mils Decimal places: 1

The *Start* tab should specify an “SMD Pin” and the geometry as “Square”. Note that since this is a surface-mou nt padstack, three of the drill options tabs are now disabled. (Selecting the lone *Drill* tab will show the “Drill Hole” as *none*.)



Pad Editor: SM40X40 (C:/Academic/EE174/Development/Lib/symbols)

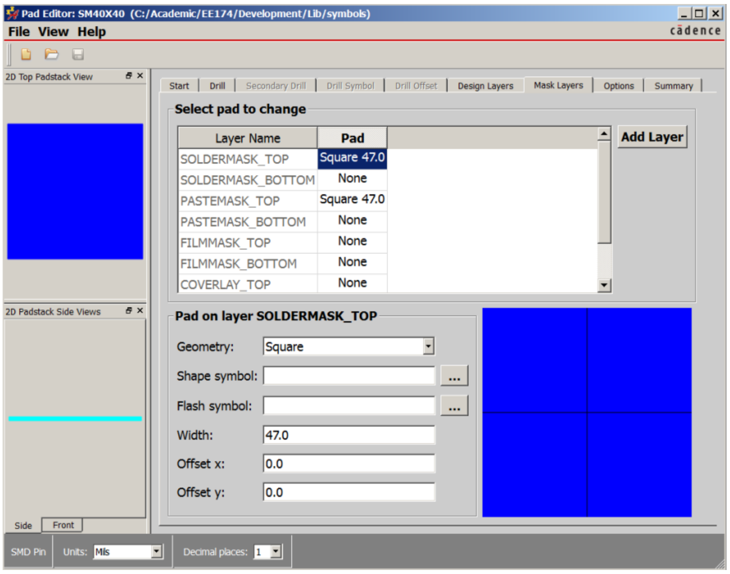
Select pad to change

Layer Name	Regular Pad	Thermal Pac	Anti Pad	Keep Out
TOP	Square 40.0	None	None	None
ADJACENT LAYER	-	-	-	None

Regular Pad on layer TOP

Geometry: Square
Shape symbol:
Flash symbol:
Width: 40.0
Offset x: 0.0
Offset y: 0.0

SMD Pin Units: Mils Decimal places: 1



Pad Editor: SM40X40 (C:/Academic/EE174/Development/Lib/symbols)

Select pad to change

Layer Name	Pad
SOLDERMASK_TOP	Square 47.0
SOLDERMASK_BOTTOM	None
PASTEMASK_TOP	Square 47.0
PASTEMASK_BOTTOM	None
FILMMASK_TOP	None
FILMMASK_BOTTOM	None
COVERLAY_TOP	None

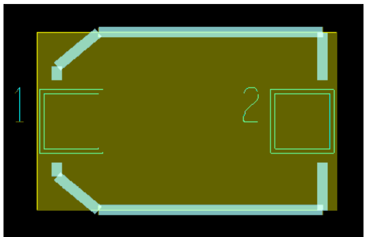
Pad on layer SOLDERMASK_TOP

Geometry: Square
Shape symbol:
Flash symbol:
Width: 47.0
Offset x: 0.0
Offset y: 0.0

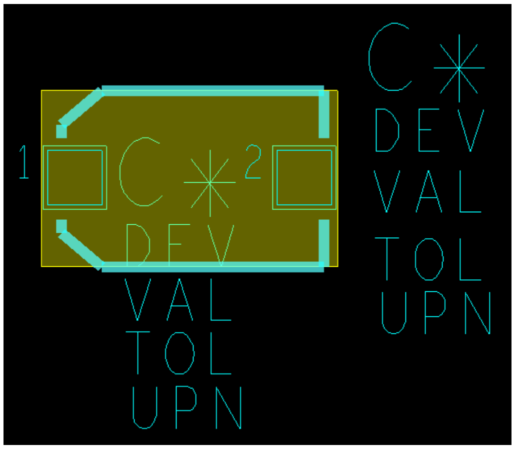
SMD Pin Units: Mils Decimal places: 1

Note the increased decimal resolution over the earlier part. After saving the part, place the two new square SM pads so their inner edges are spaced by 130 mils. This raises the next question: where do we place the second pad if we want it to be spaced 130 mils to the right? Since the pads are each 40 mils wide, placing the second one at 170 mils will give the wanted separation. Go ahead and place both pads now.

Placement Boundary and Outline – You will find these tasks much easier if the non-etch grid is set to 5 mils. First create the placement outline on the Package Geometry / Place_Bound_Top layer as a properly centered 130 by 220 mils (XY) filled rectangular shape. Then, using this as a guide add the part’s outline on the silk-screen and assembly layers. Create an artistic outline on the SST and AST layers depicting the component’s outline. Typically, this need not be very accurate, but it should at least identify the part. Shown below are the results after doing this. Don’t make the SST linework than an 8 mil draw. The AST draw can be any size, since its presence is primarily for internal documentation purposes. The result showing the filled boundary, SST and AST layers is shown below.



Go ahead and add the necessary text fields using the earlier discussion on page 7 as a guide. Note that since this is also a capacitor, the reference designator fields will be the same as for the through-hole part.



The finished part should look something like that shown on the left. Note the SST classes are along the right side, while the AST classes begin inside the part. After finishing, save your new part.

4. CONCLUSION

This concludes the addendum 2.1 to the second laboratory.

Oral Evaluation of your work for grading:

Your grade for this laboratory will be based on showing the instructor compentence using Allegro and understanding things presented and discussed in the lab. Your understanding will be gauged by responses to any of the following topics and questions.You will be asked to answer or expound on several of them.

1. Explain how Allegro uses the text fields on various classes / sub-classes.
2. What is positive and negative artwork?
3. What are the four basic padstack types?
4. What elements are required by Allegro to be on every part symbol?
5. Demonstrate how to use the padstack editor.
6. The Placement outline. What is it and why is it necessary?
7. Explain why specific text fields can be anything we like.