

# Electronic design

## Advanced Ultimate Box-Maker

---

*With Parametric Design using OpenSCAD*



# Advanced Ultimate Box-Maker

---

## Table of contents

Introduction.....	0
Originally box designs, used libraries, license.....	0
User Part of the “Advanced Ultimate Box-Maker” .....	0
Installation.....	0
Features.....	0
Summary of Existing Features which the original version and the improved version have .....	0
Added more features, list of the most important ones .....	1
Handling the program.....	1
Sections and values .....	1
Explanation of the most important values.....	2
Explanation of program-parts .....	0
Connection between the box-program and the lib “NopSCADlib”.....	0
Fansize and screw-distance .....	0
Feet for PCB.....	0
Additional holes .....	1
Box fixation tabs .....	1
Long holes.....	1
Square Holes with a Recess .....	1
Case-Feet.....	2

## Introduction

For my projects in electronics I often need an appropriate housing.

Commercial products have the disadvantage that they don't fit exactly and cause a lot of work for manual drilling, cutting, adapting, etc.

I needed a software solution for creating an individual box using a 3D-printer.

- The software must have mainly a parametric design so that the properties of the box can be easily changed.
- The software must have a GUI for changing the parameters so that the program code does not need to be changed.
- Ideally the software allows to save the parameters in individual files.

The software with the less disadvantages I could find, was OpenSCAD.

It is relatively easy to learn and has a so called customizer for changing and saving the parameters. But it has also a lot of restrictions to deal with. Despite this OpenSCAD was the first choice. My normally used CAD-Software "Fusion-360" is nice to use but has only a bit of a customizer.

For that reason, the decision was using OpenSCAD and I searched around which programs exists to solve my problem.

I found the "Ultimate Box Maker" with its forks and put me on to extend the features of these programs with help of some excellent libraries.

The result is this "Advanced Ultimate Box Maker".

### Originally box designs, used libraries, license

- Programs are under license CC BY-NC 3.0 License and GNU General Public License 3
- The original design was by "Ultimate Box Maker" from Heartman [www.thingiverse.com/thing:1264391](http://www.thingiverse.com/thing:1264391) and <http://heartygfx.blogspot.com>
- The great improvement to a PCB-based construction was the "Customizable revised Ultimate Box Maker" from jbebel [www.thingiverse.com/thing:2938921](http://www.thingiverse.com/thing:2938921) and <https://github.com/jbebel/Ultimate-Box-Maker>.
- I was inspired for the fan-part by the "Customizable Fan Grill Cover" from mightynozzle [www.thingiverse.com/thing:2802474](http://www.thingiverse.com/thing:2802474)
- Library "BOSL2" from <https://github.com/revarbat/BOSL2/>
- Library "NopSCADlib" from <https://github.com/nophead/NopSCADlib/>

# User Part of the “Advanced Ultimate Box-Maker”

Description of the features, the installation and how to use the software

## Installation

You need the following parts

- OpenSCAD, from <https://openscad.org/downloads.html>. download and install the exe-file.
- The library “BOSL2” from <https://github.com/revarbat/BOSL2/> unpack the zip-file into the appropriate lib-directory (My Documents\OpenSCAD\libraries)
- The library “NopSCADlib” from <https://github.com/nophead/NopSCADlib> unpack the zip-file into the appropriate lib-directory (My Documents\OpenSCAD\libraries)
- The actually “Advanced ultimate Box Maker” from <https://github.com/m-oster/Avanced-Ultimate-Box-Maker> Unpack the zip-file into the directory My Documents\OpenSCAD so that all files are in the sub-directory \Advanced-Ultimate-Box-Maker

To check the installation, run the main-file “Advanced-Ultimate-Box-Maker-main.scad”, and press F5 to create the box.

## Features

**Summary of Existing Features which the original version and the improved version have**

- Editable back- and front panel
- Calculation of the box size from the PCB size and margin parameters
- Horizontal or vertical text alignment arguments
- Variable panel and tab thickness separate from the box
- Square corners (no filet or chamfer)
- Parameterize all dimensions, such that the box is infinitely scalable. Doubling every parameter will double the box in all dimensions.
- Vents go through the floor and are scaled to fit the size of the filet.
- For larger filets, scale interior filets to make exterior and interior filets concentric.
- Decorations are optional
- Box fixation tabs can individually be placed on the top or bottom shell

## User part of the “Advanced Ultimate Box-Maker”

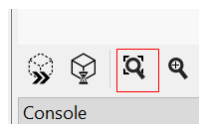
### Added more features, list of the most important ones

- Allows to integrate the electronic parts of the NopSCAD-lib
- Choice of different case feet
- Choice of different threaded inserts for fixation tabs
- Choice of different kind of PCB-feet (inserts, pillars, spacers)
- Holes with a recess for front- or backpanel
- Longholes for front- or backpanel
- Customizable fan-guard for the backpanel
- Usual fan sizes are integrated and the correct screw-distance is autom. choosen.
- Ledges on both upper edges of the box shells for better stabilization
- Allows to create extra holes in the bottom shell
- Choose predefined font for paneltext
- Frequent remarks added in the code and a lot of debug outputs

With this advanced version the number of program lines has grown from round about 900 to more than 1800 lines.

### Handling the program

After starting OpenSCAD, you see the customizer on the right side with the various program values categorized in sections.



Click the center symbol to view the whole box

### Sections and values

Sections and values in the customizer on the right side	
<ul style="list-style-type: none"><li>▶ Box options</li><li>▶ Case Feet</li><li>▶ Box Fixation Tabs</li><li>▶ PCB Options</li><li>▶ PCB Feet</li><li>▶ Frontplate Holes</li><li>▶ Frontplate Text</li><li>▶ Backplate Holes</li><li>▶ Backplate Text</li><li>▶ FAN guard Settings</li><li>▶ STL element to export</li><li>▶ Special Values</li></ul>	These sections include several parametric values to adapt the box to your requirements in an easy way
Same values are on the left side in the editor window	
<pre>/* [Box options] */ // - Wall thickness Thick = 2.8; //[2.0:0.1:5.0]</pre>	<div>▼ Box options</div> <div>Thick</div> <div>- Wall thickness</div> <div>2,8</div>

## User part of the “Advanced Ultimate Box-Maker”

Kind of values in the customizer	
<b>aAddHoles</b> - Additional Holes - [OnOff,xPos,yPos,diam] [[0, 54, 103, 3]]	<b>Arrays</b> , beginning with “a..”: Cannot be changed in customizer. MUST be changed in the editor window on the left side because of the customizers restrictions.
<b>TextOnPCB</b> - Text on PCB PCB	<b>Textbox</b> , sets a text. f.e. a text on top of the PCB. Change in customizer.
<b>Vent</b> - Decorations to ventilation holes Yes	<b>Boolean Value</b> , activates/deactivates an option. Change in customizer.
<b>CaseFeet</b> - Kind of Case Feet Folding Feet small	<b>Selection</b> , selects a value out of a given list.
<b>CutoutMargin</b> - Printer margin around interior cutouts 0,3	<b>Numeric value</b> , can be changed in customizer. Change in customizer.

All values, with exception of the arrays, can be saved in a configuration-file on top of the customizer window. The arrays must be saved with the complete scad-file (i.e. changing of the code)

Customizer

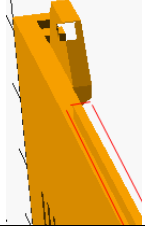
☒ Automatic Preview

only Bottomshell ▼

One member is the minimum for an array, otherwise the program throws an error. If the part is not used, array-members can be deactivated when On/Off is set to “0”.

## Explanation of the most important values

### Section “box options”

<b>PCBDraw</b> - Draw PCB Yes	<b>PCBDraw</b> , draws the PCB inside the box and labels it with the text of “TextOnPCB”. Only one PCB possible.
<b>aAddHoles</b> - Additional Holes - [OnOff,xPos,yPos,diam] [[0, 54, 103, 3]]	Array must be edited in code window. For manual adding holes in the bottom shell for additional PCBs f.e.
<b>PartsDraw</b> - Draw Electronic Parts No	Draws electr. Parts with help of the NopSCADlib. Parts must be defined in “MyComponents.scad”, imported then positioned in Module “ShowElectrParts”.
	<b>Bar recess</b> , can be activated/deactivated. When Thick <2.5mm, recess is autom. deactivated because it cannot be printed.
<b>stextfont</b> - Font general (Windows-Fonts) Calibri	Selection of the font for text on panels. List contains windows-fonts and can be changed in code window. Font names must fit to the name in windows.



### Section “case feet”

<b>CaseFeet</b> - Kind of Case Feet OnlyHoles	Various feet can be chosen, simple hole, rubber feet, foldable feet in 2 sizes. The last both are created as stl-files and can be 3D-printed.
---	---

### Section “Box Fixation Tabs”

<b>tab inserts</b> - Varius kinds to fasten the Shells Screws	Various kinds, from simple holes for screws to threaded inserts
---	---

### Section “PCB Options”

With defining these variables the dimensions of the box are autom. calculated.

The value “TopMargin” defines in result the hight of the box.

The console window shows the resulting dimensions of the box.

### Section “PCB Feet”


<b>PCBFeet</b> - Kind of PCB feet? (x4), instead of PCB-Feet with Feet	Various kinds of PCB-feet, from simple holes for screws to holes with threaded inserts. All dimensions are autom. adapted.
Coord. Of the 4 PCB-feet	The console window shows the resulting distances of the PCB-feet to prove the correct values of the PCB-holes.

If you enable “ShowExplanations” in the section ”Special Values”, the console window shows the position of the 4 PCB-feet.

### Section “Frontplate Holes”

The variables within this section define 3 kind of holes.

All these hole-parameters are Arrays and must be changed in the code window, not in the customizer.

aFP_Holes	Defines round holes or longholes There are 2 diams, diam1 and diam2. If both are equal, the result is a round hole. If diam1 > diam2 the result is a longhole If sym=0, the hole is narrower on the right side. If sym=1, the hole is narrow on left and right side
aFP_Shoves	Defines square holes
aFP_SHovesRecess 	Defines a recess of PanelThick minus Recess. If Recess is positiv, its on the outer side. if recess is negativ, its on the inner side.

## User part of the “Advanced Ultimate Box-Maker”

---

### Section “Frontplate Text”

There are 2 kind of Texts: Normal text and round text.

All these hole-parameters are Arrays and must be changed in the code window, not in the customizer.

aFP_RTexts	Defines round text. F.e. useful for labeling Poti-values
aFP_STexts	Defines normal text

**Section “Backplate Holes”**, same as for Frontplate Holes

**Section “Backplate Text”**, same as for Frontplate Text

**Section “Fan-guard settings”**, implemented only for the Backplate

BP_fan_position	Pos. of the fan-guard, can be activated by On/off-value set to “1”
Fan_size_in_mm	Dim. of fans are integrated in an array. When selecting a size, all parameters are calculated autom.
Screw_hole_diameter	Diam. of the screw between M2-M4
Fan_min_border_size	Try out the best fitting values. But the simplest way is not to change them.
Fan_line_size	
Fan_line_space	
Number_of_support_lines	Number of crosshairs of the fan-guard, 4 is a good value.

### Section “STL-element to export”

These Boolean values determine which part should be printed. It makes sense to enable only one part at a time.

With keys F5, then F6, you can render a part for STL-Export.

Export the part with key F7 and save it as STL-file.

TShell	Top shell of the box
BShell	Bottom shell of the box
FPanl	Frontpanel
BPanel	Backpanel
SmallParts	Small assembly-parts which should be printed. It depends what you have chosen, F.E. It can be spacers for the PCB.

### Section “Special values”

ShowExplanations	Self defined echo-outputs in the console-window. They are defined in the “MyExplanations.scad” file. F.E. the position of the 4 PCB-feet
Debug	When “1”, there is a lot of echo output from program-values. For easier assignment, the output has format:: <mod.:> from which module the output comes <description:> name of the variable or its description <value> the value itself of the var



## Explanation of program-parts

The most important parts for better orientation with their line numbers in brackets

### Connection between the box-program and the lib “NopSCADlib”

Its possible to use this library to insert electronic parts in the design of the box for better illustrating and checking the dimensions. These parts are only visible in preview-mode (<F5>) because it makes no sense to create stl-files from that parts.

The use of elements in the NopSCADLib” are all similar structured. On the upper level is a definition-file which presets the dimensions, position and other values for an object.

The second level is a second file which draws this object.

For example the file “Inserts.scad” contains several types of threaded inserts.

This module calls the actual part in another file named ”insert.scad”.

The description files with several types are named in plural, the files for drawing the object are named in singular.

This has the advantage to separate the drawing from the definition of an object.

And this makes possible to use own definition files in an easy way without need to change the library code.

The files “My..” in the program folder are self-created definition files for that purpose.

The import takes place over several layers:

- ❖ The module “ShowElectrParts (L.1715) in the Main-program makes call of a part
  - Module with Electronic part in “MyComponents.scad”
    - there are all electr. parts listed, for each part an own module
    - this layer abstracts the definitions and values to follow on the next layer
      - Call of the drawing part itself in the NopSCADLib/vitamins directory

### Fansize and screw-distance

The size of a fan is predefined in a selection field “fan\_size\_in\_mm” in the customizer area. (L.258)

This field is a list of fans with the dimension. The choice of a fan in the list results in an index-value. F.e. the fan with dim. 40mm results in an index with value “2”.

The according screw-distance of the choosen fan is calculated from line (329) with the array “aPos” and the fan-type.

### Feet for PCB

The module “Feet” (L.913) is upgraded to make it possible to use spacers, hex-pillars. With help of the resulting index of the choosen feet-type the appropriate part is called within the “nopSCADlib”.

When inserts are choosen, the correct hole-diameter of the pcb-feet is calculated.

## Explanation of program-parts

---

The dimensions of the inserts are defined in the file "MyInserts.scad".

### Additional holes

Within this concept of this program it is only possible to use 1 PCB.

In the array "aAddHoles" (L.111) can be defined more holes to make it possible to implement a 2<sup>nd</sup>. PCB.

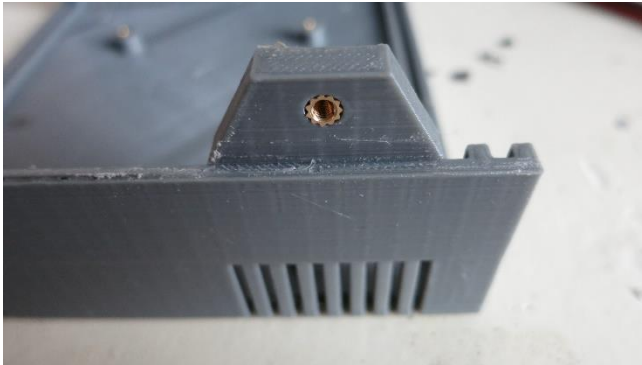
Additional holes can only be defined for the bottom shell not for the top shell.

Otherwise it would be the case that the add. holes in the bottom shell are also in the top shell because the top shell is a mirror the bottom one.

### Box fixation tabs

The fixation tabs "tab\_inserts" (L.139) were upgraded for the use of threaded inserts.

The resulting value for the index is evaluated in (L.348) and presets the correct value for the tab-hole in module "Tabs" (L.720)



### Long holes

There are some parts like powerplugs or powerjacks which need long holes.

The flattening of the hole can be on the right side or on both sides.

For easier coding the arrays aFP\_Holes and aBP\_Holes have been extended by a 2<sup>nd</sup> diameter "diam2" to define the second value for this long hole. For that the last parameter "symmetric" in the array must set to "1".

To create a normal hole, the values for "diam1" and "diam2" must be equal.

"diam1" must be greater than "diam2".

### Square Holes with a Recess

Sometimes is the need of holes with a recess. That can be necessary on the outer side for an usb-connector or on the inner side of a panel for a LED-Display.

The code for a recess is implemented only for a square hole.

The definition of the parameters is in arrays "aFP\_SHolesRecess" and "aBP\_SHolesRecess". (L.219 and L.237).

The programm sequence is similar to that of normal holes.

The modules "FPanelHolesRecess" (L.449) and "BPanelHolesRecess" (L.495)

evaluate the elements of the arrays and pass them on to the module

"SquareHoleRecess" (L.1378) which creates the hole(s).

## Explanation of program-parts

---

### Case-Feet

With the options in the field for case feet it is possible to choose two sizes of feet which can be fold up.

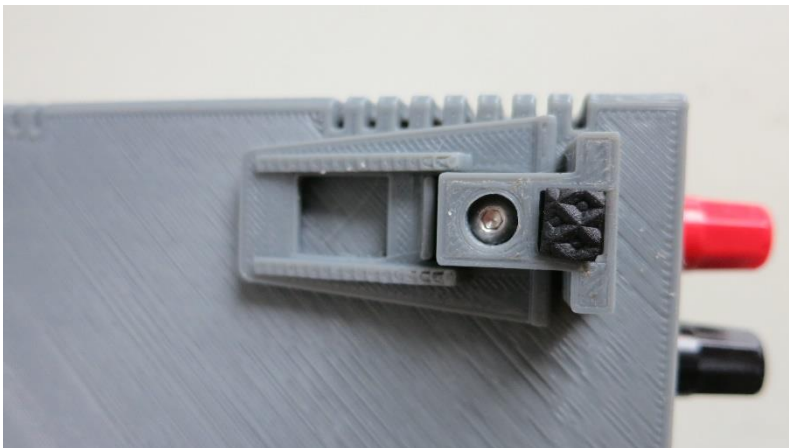


The definition of the feet is in “CaseFeet” (L.119)

The feet consist of 2 parts: the base-part and the arm.

For easier handling the foot has been created in Fusion and can be direct printed using the stl-files in the subdirectory “stls”. The source design file for the feet is in the directory “assemblies”.

To prevent the powersupply form sliding over the ground the some rubber-feet can be added into the recess of the foldable feet.



But it must be self-adhesive rubber material not sponge rubber. That's too soft.