

## Edge Rounded Cube OpenSCAD Utility

 **Polymath**

[VIEW IN BROWSER](#)

updated 24. 4. 2023 | published 24. 4. 2023

## Summary

Utility routine in OpenSCAD to round any and all edges of a 'cube'

[Hobby & Makers](#) > [Other Ideas](#)

Tags: [rounding](#) [smooth](#)

Needing to add some freeSCAD 'cubes' to a design with different edges radiused to different radii I sidetracked myself into building a reuseable module to radius any number of edges of a cube each to any radius of choice. The module `ercstd.scad` (edgeroundcube) exists in two formats the main one is verbose and includes full descriptions of use and function. If you like to keep files small there is also `erct.scad` (edge rounded cube tokenised) which has had all variables tokenised to single characters and comments and whitespace removed, but still functions identically. There's not much use for an edge rounded cube per-se (except for making dice) but as a component of something complex this can be called with two twelve-element vectors and the call `edgeround(x,y,z,vector1,vector2);`. Both copies of the module have a test routine which can be uncommented to build a fully radiused 'bar of soap' model.

Models pictured are:

- 1) a cube with a single radiused edge.
- 2) a block with all vertical edges radiused.
- 3) a 'smartphone' shape clearly showing the tricuspid artifact at each

corner where radii on meeting edges are not the same.

4) A fettled block with very small edge radiuses which, because they are all the same radius allows for very clean corners

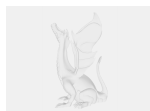
and 5) a 'bar of soap' where all edges are radiused larger and the same allowing again for fully eighth-spherical corners.

Be aware that due to the large number of subtractions made during the build there are often unwanted and invisible artifacts. All models should be tested for mesh accuracy and fixed before attempting to slice.

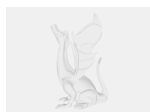
erc5.obj still has 8 open edges, these faults should be susceptible to autocorrection and software is available that can clean up the mesh (but PrusaSlicer doesn't fix model 5).

I needed to be able to reference the 12 edges of a 'cube'. In order to do this I also needed to reference the 8 vertices. I settled on the concept of a one unit cube such that the axis values of each corner could be written down. The corner at the origin thus became 000 and the most remote corner 111, everything else followed from this. After a false start (in which I named the sides after their end vertices, which made the code very difficult to follow) I settled on 1 to 4 for the base plane edges, 5 to 8 for the vertical ones and 9 to 12 for the top plane edges. In order that users could follow this process, not least so that they could edit the two 12 element vectors used, I have included a full exposition in the comments of the verbose version of the scad file. I hope that this helps. Enjoy ...

## Model files



**erct.scad**



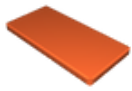
**ercstd.scad**



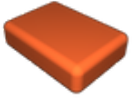
**erc1.obj**



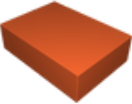
**erc2.obj**



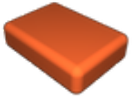
**erc3.obj**



**erc6.obj**



**erc4.obj**



**erc5.obj**

## License



This work is licensed under a  
**Creative Commons (4.0 International License)**

**Attribution—Noncommercial—Share Alike**

- 
- ✗ | Sharing without ATTRIBUTION
  - ✓ | Remix Culture allowed
  - ✗ | Commercial Use
  - ✗ | Free Cultural Works
  - ✗ | Meets Open Definition