TreeSoft CSV

CSV was written by Paul Rowntree, based on information and sample source codes provided by Vectric and other ‘gadgeteers’. Any flaws in this package were introduced by me. I believe it works well, but no guarantees are given for its use in any application. CSV is designed to work with Aspire 4.0 (or greater) and VCP 7.0 (or greater). It will not work with previous versions of either program. Although Rowntree retains copyright, you are free to use the components in any way you wish, including personal and commercial applications.

If you really like CSV or if it has helped you in a significant manner, please consider supporting CSV’s continued development via the author’s website, PaulRowntree.weebly.com. Check this site for updated versions of **CSV** .

Change Log

July 17 2013	- Start from origin made an option, toggled by NoGoingHome=true at top of file
March 28 2013	- Added auto-detection of CSV reference plane (top vs bottom) - Depth scaling and rotation about the anchor point added
Feb 25 2013	- Adapted to Vectric scripting conventions for Aspire 4/VCP7 - Eliminated TreeSoft File Selector hack, replaced by script FileDialog()
Feb 20 2013	- Written, knowing that the gadgets and file selector mechanism will need to be modified for final release.

Disclaimer

By downloading, installing and using this program you are accepting full responsibility for any and all consequences. CNC machinery is potentially dangerous, and the user is 100% responsible for ensuring that the output of **CSV** is safe to use on any CNC equipment, and that it will have the desired effects.

As always with CNC equipment, think many times before running code, and doing air cuts is often a good idea with new files. Verify the Z limits of the loaded files before cutting to ensure that you are not going to destroy your table top, spindle, or both.

Work and Play safely.

What is CSV ?

CSV is a set of tools that can be used to load 2D vectors and 3D toolpaths into Vectric’s Aspire and V-Carve Pro programs. CSV files have ‘Comma Separated Values’ that can be created and edited with any standard text editor, Vectric Post-Processors, or using simple Lua scripts; do not to use a word processor like Word as it will inject extra characters that can’t be read properly. Lua is the scripting language for Vectric programs. CSV files created using the included PostProcessors can describe very complicated 3D toolpaths created by someone else. We normally think of VCP as being a 2D CAD/CAM package. While it is true that VCP cannot be used to prepare and edit 3D toolpaths, it ***can*** output cutting paths based on imported 3D paths. The **CSV** package lets you do the importing, and gives you some ways to create these files too.

The gadgets will work in Aspire and VCP in exactly the same way.

It is sometimes easier/faster/wiser to use math instead of a mouse.

Making CSV files

The tools include Post Processors that produce 3D CSV files instead of g-code from Aspire, V-Carve Pro and PhotoVCarve. Each line of the output from the post-processors is a set of x,y,z coordinates that describe where the tip of the tool is moved to. All other information (e.g., initializations, feed rates, …) is removed. These CSV files can then be loaded into Aspire or VCP as a 3D toolpath, and they will precisely reproduce the original design if you use the same cutting tool as the original toolpath was laid out for. This allows Aspire users to share 3D designs with VCP users. **Users are expected to respect copyright principles when sharing such information. If you didn't create the content, or have the permission from the person who did, you probably do not have the right to distribute the original content, or anything based on it. When in doubt, ask, and respect other people's rights.**

It is fairly simple to use a Lua script to create CSV files, and you can design complex 2D geometric shapes (drawing vectors) or 3D toolpaths that can be imported into either Aspire or VCP. Examples are included in the package; they were created using the LuaForWindows package, which is a free download from

http://code.google.com/p/luaforwindows/

Running SciTE and creating these CSV files from Lua scripts does not involve Aspire or VCP. You will have to learn how to program to use whatever language you pick; you can start by studying and modifying the examples included in the **CSV** package.

Whichever way the 3D toolpaths were created, you should know if Z=0 refers to the top of the material or to the bottom, and the cutting tool that the toolpath was designed for. The gadget will guess the Z=0 Reference based on the data values, but you can override this; it will shift the CSV values up/down if required.

Using CSV Files

CSV to 2D Vectors imports vectors from CSV files, and installs them into the current project. Each line must contain at least 2 numbers, separated by a comma. Any extra information is ignored. Multiple vectors can be imported from a single file if there is at least one line that cannot be read as (x,y) data. For example, a blank text line, or text line(s) with leading symbols like ';' or '*' (or anything else that doesn't look like numbers or commas). Once imported, these vectors can be scaled, rotated, saved and toolpathed like any other 2D vector created within the Vectric program (or imported as EPS files).

In many/most cases the 2D vectors that you import will have complicated layouts, and may be open or closed. Simple shapes are probably more easily prepared inside Aspire or VCP. Even if they are closed, many imported vectors cannot be directly V-carved because the lines cross-over themselves, etc. The Vase_closed.csv and Vase_Open.csv files included in the CSV package are examples of vectors that cannot be V-Carved. All is not lost! Read on to see how to get around this problem.

CSV to 3D Toolpaths will read in (x,y,z) data into Aspire or VCP, and create a fully functioning 3D toolpath that can be previewed and turned into gcode instructions for cutting. These cannot be edited **after** importing, but they can be rescaled, repositioned and rotated **during** importing : if an existing 2D vector was selected when the 3D data was imported, the imported data is scaled so that it fits the rectangular bounding box of the pre-selected vectors exactly. You can also type in precise values. The overall depth of the toolpath can also be scaled, and the choice of Z=0 for the toolpaths adapted to the choice for the current document. Adding 3D toolpaths to VCP, and making use of the Vectric preview capabilities, was the initial motivation for creating the **CSV** package.

CSV gadgets can import ~ 1 MB of information per second; most imports will be very fast. The vectors and toolpaths that are loaded are continuous, unless there are breaks in the CSV file that force the gadget to start another vector or toolpath. This makes it easier to create closed toolpaths, and to edit the vector shapes using standard polyline editing tools.

Unpacking CSV

There are no registry entries associated with the **CSV** package, so you can delete the files at any time with no lingering ‘junk’. The zip file can be opened with Windows, WinZip, Z-zip, or any other standard unzipping program. Extract the files to a convenient directory, then move the individual components in the directories shown below. When Aspire or VCP runs it will find the gadgets and show them in the Gadgets menu lists.

Windows Path	Contents
Aspire and VCP User Gadget directories, created during Aspire or VCP installation **C:\Users\Public\Public Documents\Vectric Files\Gadgets\Aspire Vx.y\** Or **C:\Users\Public\Public Documents\Vectric Files\Gadgets\VCarve Pro x.y\**	**CSV to 2D Vectors.lua** **CSV to 3D Toolpaths.lua** **CSV to 3D Toolpaths.htm** If you want, you can create a subdirectory (‘**CSV**’) and placed these Lua files inside it. This unclutters the Gadgets menu of Aspire and VCP.
Aspire and VCP Post Processor directories (if needed) Find these by going to the Aspire or VCP’s File → Open Application Folder. PostP will be a sub folder	**TreeSoft CSV (inch).pp** **TreeSoft CSV (mm).pp**
PhotoVCarve Post-Processor directories (if needed) **c:\Program Files\PhotoVCarve\PostP**	**TreeSoft CSV_PVC (inch).pp** **TreeSoft CSV_PVC (mm).pp**
Anywhere you want	The remaining files are sample CSV data files and Lua scripts. You can put them wherever you like

Using CSV Components

PostProcessors

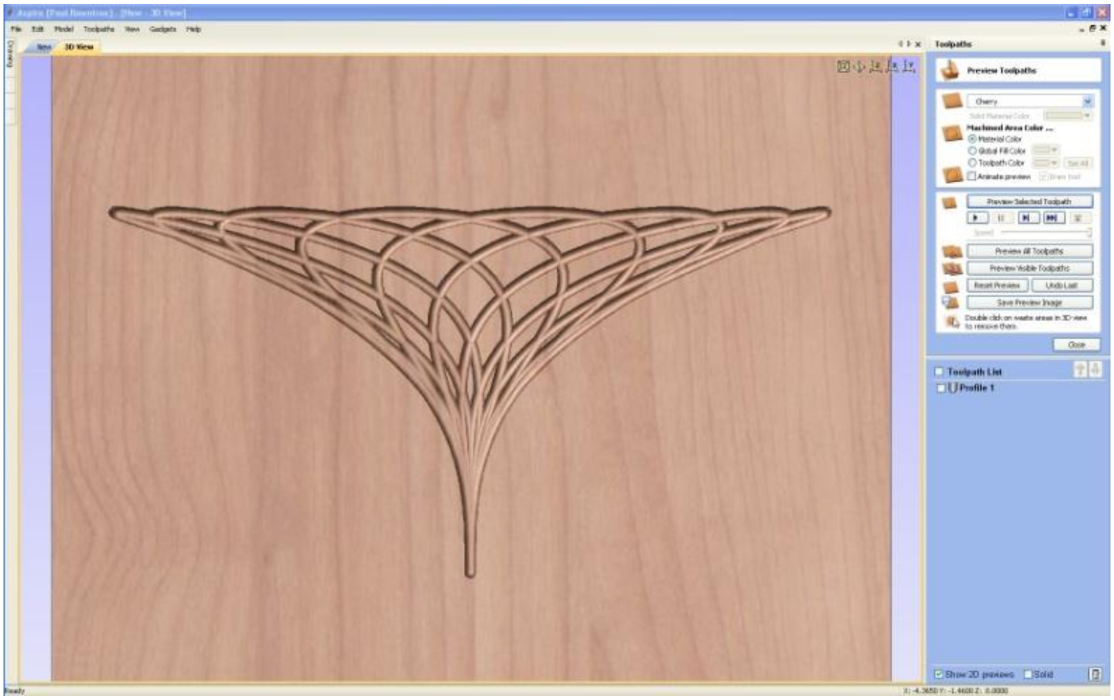
The CSV post-processors work as the other gcode generators provided by Vectric do; select CSV (inch) or CSV (mm) from the PostProcessor drop down list, and save the toolpaths. You can store multiple toolpaths to the same CSV file and they will be loaded in correctly as multiple paths into Aspire or VCP. Normally, a CSV file would contain all toolpaths for a single cutting tool, arranged in order of cutting.

> *** IMPORTANT: If these 3D toolpaths require roughing out processes, make sure that you create these as CSV files as well, and import them into Aspire/VCP. When preparing the actual cut files, use cutting tools similar to those used to layout the originals, and the roughing must be executed before the finish passes. Ensure that toolpaths that should overlap in the final project do so perfectly; this may be tricky if you are rescaling them during the importing process (see below). The toolpaths for roughing passes do not follow the same coordinates as finishing paths unless the tools are the same size (unlikely.

CSV to 2D Vectors

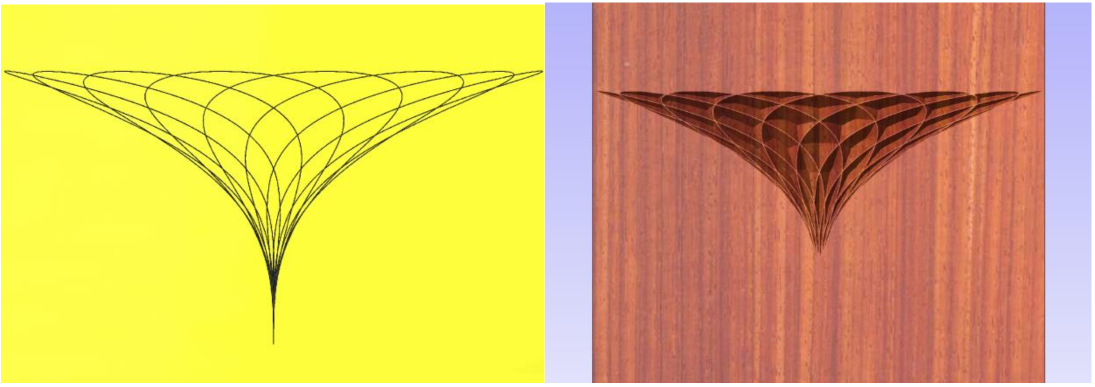
Importing 2D CSV files is easy, and the created vectors are placed on the material with no changes or scaling. The **CSV to 2D Vectors** gadget will prompt for a filename, and you navigate to the CSV file you want to work with. Normally, CSV files will have the extension ‘csv’ or ‘CSV’, but you can choose any file you want. The gadget will then load it up if it has two or more comma-separated values per line using the first two as x and y, respectively, and give some information about the vectors it found.

The following 2D pattern was imported from the Vase_closed.csv file (which was created by the Vase.lua script and then rotated by 90 degrees. Both the Lua script and the CSV file are in the CSV package. It was toolpathed with a ball-nosed bit. For vectors that can be easily described by formulas, this is a simple way to get it done, and to tweak the design until it looks like what you want.



Imported vectors (like Vase) may not be suitable V-carving, although it can be profiled as shown above. If you really do want to V-carve out the spaces between the lines to give it the ‘chip carved’ look, this approach works for most vectors:

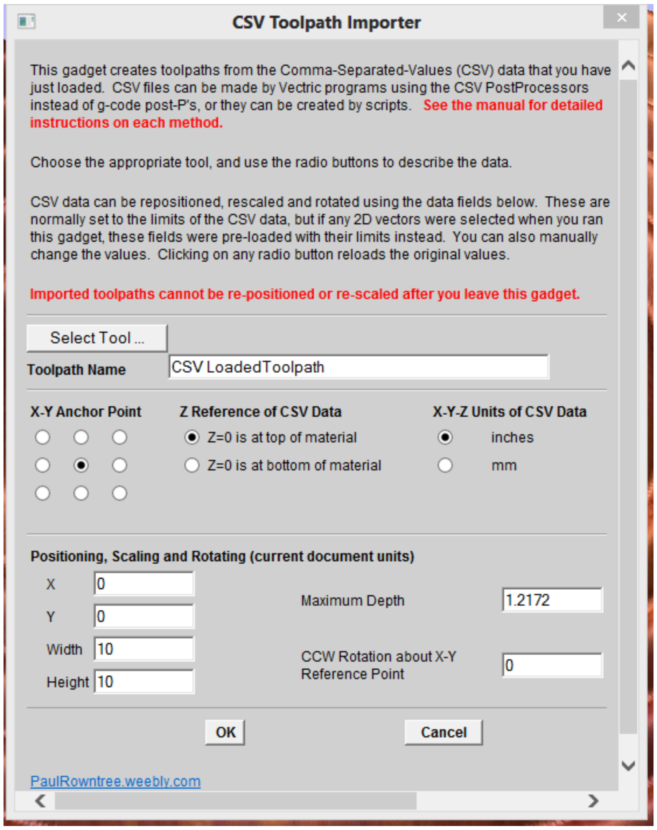
1) Profile the vector with a 90 degree V-bit, and set the toolpath depth to be ~0.01” (~0.25mm). Set the material to be yellow and the toolpath colour to be black, and use the highest review quality possible. Make the material as large as possible on the screen but still showing all of the design; view from along the Z axis. You should see a set of black lines on the yellow background. Reducing the depth may give better resolution, but the lines must be continuous in the preview bitmap.
2) Save the preview screen as a bitmap, then restore the material to what you really want to use.
3) Hide the original vectors, and import the bitmap (from step 2) back into the drawing and design area. For best results make it as large as possible.
4) Use the bitmap trace functions of Aspire/VCP to create lines on sides of your black lines. **Erase or hide the outermost vector that surrounds the entire shape and any nuisance vectors**. Check to see if any additional cleanup is required, perhaps on the bitmap itself.
5) Group, select, scale and position the remaining set of closed vectors, then use them in a V-carving toolpath.



CSV to 3D Toolpaths

***** If needed, create CSV files of roughing toolpaths as well, and import them into Aspire/VCP. When preparing the actual cut files, run the roughing paths first.**

Importing 3D toolpaths is simple, but since toolpaths cannot be edited like vectors, adjustments have to be done during the importing, not after. **For this example, make sure there are no selected vectors on the project.** This gadget will prompt for the CSV file just like the **CSV to 2D Vectors** gadget did. Once you pick the file, it will take a few seconds to load up the data, and then a dialog box will appear that lets you control the importing process. The following is for importing “Sombrero_TOP.csv”, which is included in the CSV package. This file has X and Y ranging from -5 to +5, Z goes from -1.22 to about 0. There is no unit information in any CSV file.



CSV Toolpath Importer

This gadget creates toolpaths from the Comma-Separated-Values (CSV) data that you have just loaded. CSV files can be made by Vectric programs using the CSV PostProcessors instead of g-code post-P's, or they can be created by scripts. **See the manual for detailed instructions on each method.**

Choose the appropriate tool, and use the radio buttons to describe the data.

CSV data can be repositioned, rescaled and rotated using the data fields below. These are normally set to the limits of the CSV data, but if any 2D vectors were selected when you ran this gadget, these fields were pre-loaded with their limits instead. You can also manually change the values. Clicking on any radio button reloads the original values.

Imported toolpaths cannot be re-positioned or re-scaled after you leave this gadget.

Select Tool ...

Toolpath Name CSV LoadedToolpath

X-Y Anchor Point

○ ○ ○
○ ● ○
○ ○ ○

Z Reference of CSV Data

● Z=0 is at top of material
○ Z=0 is at bottom of material

X-Y-Z Units of CSV Data

● inches
○ mm

Positioning, Scaling and Rotating (current document units)

X 0
Y 0
Width 10
Height 10
Maximum Depth 1.2172
CCW Rotation about X-Y Reference Point 0

OK Cancel

PaulRowntree.weebly.com

The dialog works best from top-to-bottom; clicking radio buttons may change values presented below, and so you may lose your installed values if you work in a different order.

1) The text gives a short version of these instructions.

2) Use the **"Select Tool"** button to pick a cutting tool from the Vectric Tool Database. The dialog box will keep coming back until you pick one or press Cancel (and lose the imported data). The identifier of the tool will show to the right of the button. You will probably want to use a tool similar to what was intended when the toolpath was created unless you are scaling the X-Y dimensions by a lot.

3) The **"Toolpath Name"** is the name that will show up on the Toolpath palette. The default is "CSV Loaded Toolpath", but you can change it. If blank, it will return to the default. Projects can have multiple toolpaths with the same name.

4) The **"X-Y Anchor Point"** is one of nine possible positions on the imported shapes' bounding box. The default is the centre. The "X-Y Anchor Point" coordinates are shown in the "X" and "Y" fields below (0.0 for both X and Y for the centre of Sombrero_TOP.csv), and these fields update when you change the choice of anchor point. Click around the nine choices to see how this works.

5) **"Z Reference"** is where the CSV file data is referenced on the material, which can be the same or different from the current document's reference. If there are more negative Z values than positive in the CSV file, the gadget assumes that Z=0 refers to the top of the material in the CSV file. Change this if the gadget guessed wrong.

6) **“X-Y-Z Units”** has the choice of inches or mm. By default this is set to the current document’s units, but you can change this if necessary. Changing the units will change the values shown in the next section of the dialog box, which are always in the current document’s units.

7) The **“X”** and **“Y”** values show where the “X-Y Anchor Point” will land on the current document. You can manually enter values to reposition the toolpath. For example, if the document has (0,0) in the bottom left corner of the material, directly importing the Sombrero_TOP toolpath will have only one quadrant (+,+) visible. If you choose the bottom left corner to be the anchor point, set “X=0.0” and “Y=0.0”, then the pattern will be imported with the lower-left corner of the toolpath aligned to the lower-left corner of the material.

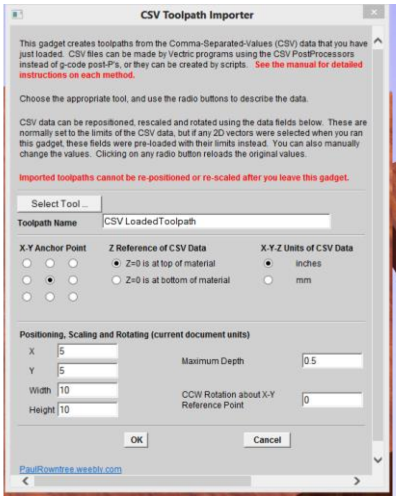
8) The **“Width”** and **“Height”** fields show by default the extents of the CSV data in the X and Y directions, respectively. Changing these values will rescale the imported toolpaths in the X and Y directions, keeping the anchor point fixed.

> Remember that you are scaling toolpaths, not vectors. These toolpaths were (hopefully) constructed with some consideration given to the tool’s diameter, stepover etc. Stretching a toolpath ***along*** raster lines works well, but if you stretch too much ***across*** a rastered toolpath you may see ridges between raster lines. If you shrink a toolpath too much you may be wasting your time with very small intervals between raster lines.

9) **“Maximum Depth”** is the depth of the imported data below the surface of the material, based on the Z values of the CSV data (transformed into the current document’s units if necessary) and the “Z Reference” identified above. Changing this value is possible too.

10) **“CCW Rotation”** allows you to rotate the imported toolpath by this number of counter-clockwise degrees, keeping the “X-Y Anchor Point” fixed. The default is 0.0 (no rotation).

Press **“OK”** to complete the process. A dialog box will tell you how many discrete toolpaths were imported, then you can preview these paths. Here is the result for the Sombrero_TOP imported to fully cover a 10”x10”x2” material, which has the (0,0) origin in the bottom-left corner. The centre of the CSV file is set to be (5,5), and the Width and Height have been kept at the 10”x10” found in the file. The depth has been reduced from 1.2” to 0.5”. The second image shows a shrunken, distorted and rotated version of the same Sombrero_TOP data.



CSV Toolpath Importer

This gadget creates toolpaths from the Comma-Separated-Values (CSV) data that you have just loaded. CSV files can be made by Vectric programs using the CSV PostProcessors instead of g-code post-P’s, or they can be created by scripts. **See the manual for detailed instructions on each method.**

Choose the appropriate tool, and use the radio buttons to describe the data.

CSV data can be repositioned, rescaled and rotated using the data fields below. These are normally set to the limits of the CSV data, but if any 2D vectors were selected when you ran this gadget, these fields were pre-loaded with their limits instead. You can also manually change the values. Clicking on any radio button reloads the original values.

Imported toolpaths cannot be re-positioned or re-scaled after you leave this gadget.

Select Tool ...

Toolpath Name: CSV LoadedToolpath

X-Y Anchor Point
○ ○ ○
○ ● ○
○ ○ ○

Z Reference of CSV Data
● Z=0 is at top of material
○ Z=0 is at bottom of material

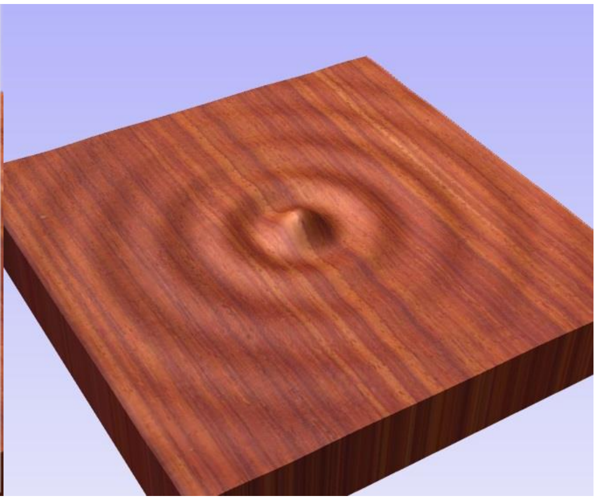
X-Y-Z Units of CSV Data
● inches
○ mm

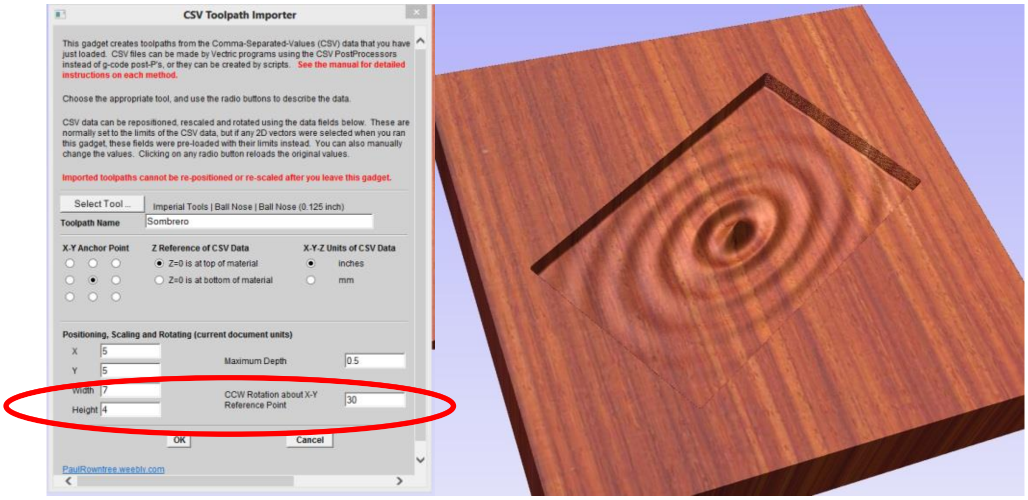
Positioning, Scaling and Rotating (current document units)

X: 5
Y: 5
Width: 10
Height: 10
Maximum Depth: 0.5
CCW Rotation about X-Y Reference Point: 0

OK | Cancel

PaulRowntree.weebly.com





CSV Toolpath Importer

This gadget creates toolpaths from the Comma-Separated-Values (CSV) data that you have just loaded. CSV files can be made by Vectric programs using the CSV PostProcessors instead of g-code post-P's, or they can be created by scripts. See the manual for detailed instructions on each method.

Choose the appropriate tool, and use the radio buttons to describe the data.

CSV data can be repositioned, rescaled and rotated using the data fields below. These are normally set to the limits of the CSV data, but if any 2D vectors were selected when you ran this gadget, these fields were pre-loaded with their limits instead. You can also manually change the values. Clicking on any radio button reloads the original values.

Imported toolpaths cannot be re-positioned or re-scaled after you leave this gadget.

Select Tool ... Imperial Tools | Ball Nose | Ball Nose (0.125 inch)

Toolpath Name: Sombrero

X-Y Anchor Point
○ ○ ○
○ ● ○
○ ○ ○

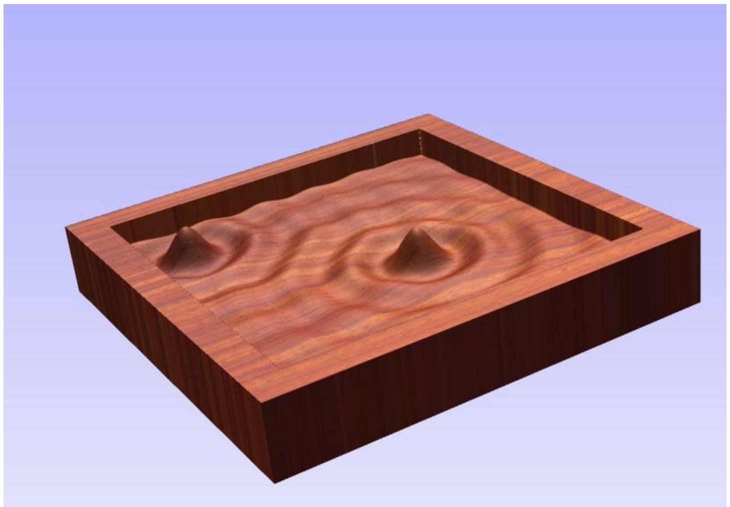
Z Reference of CSV Data
● Z=0 is at top of material
○ Z=0 is at bottom of material

X-Y-Z Units of CSV Data
● inches
○ mm

Positioning, Scaling and Rotating (current document units)

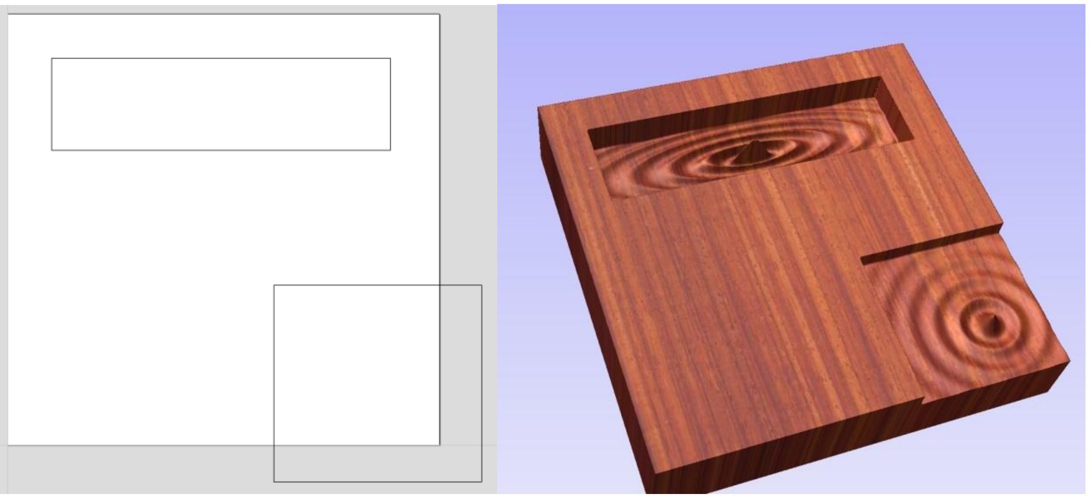
X: 5
Y: 5
Width: 7
Height: 4
Maximum Depth: 0.5
CCW Rotation about X-Y Reference Point: 30

Finally, the TwoSombreros.csv files show interesting interference effects.

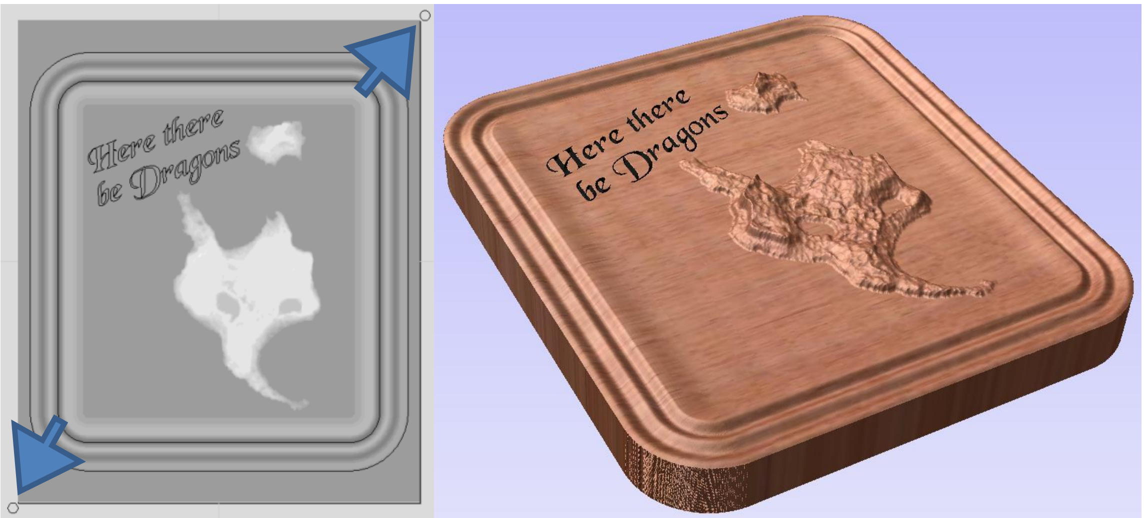


Finding and typing in coordinates can be a chore, especially if you want to accurately align the imported toolpaths with some existing features of the Aspire or VCP project. If you have selected one or more vectors on the drawing surface, the gadget will use the rectangular bounding box of the selected vector(s) to set the default X,Y, Width and Height instead of those found in the CSV file. If you are using vectors to set the position and size of the toolpath, then the units of the CSV data are irrelevant; changing units will only change the Maximum Depth parameter. Here is an image of two rectangles on the drawing surface; they were selected one at a time, then the **CSV to 3D Toolpaths** gadget was run for each one (different Depths) to produce the preview.

Having a box that outlines the imported region is also useful for working around the pattern at the design stage.

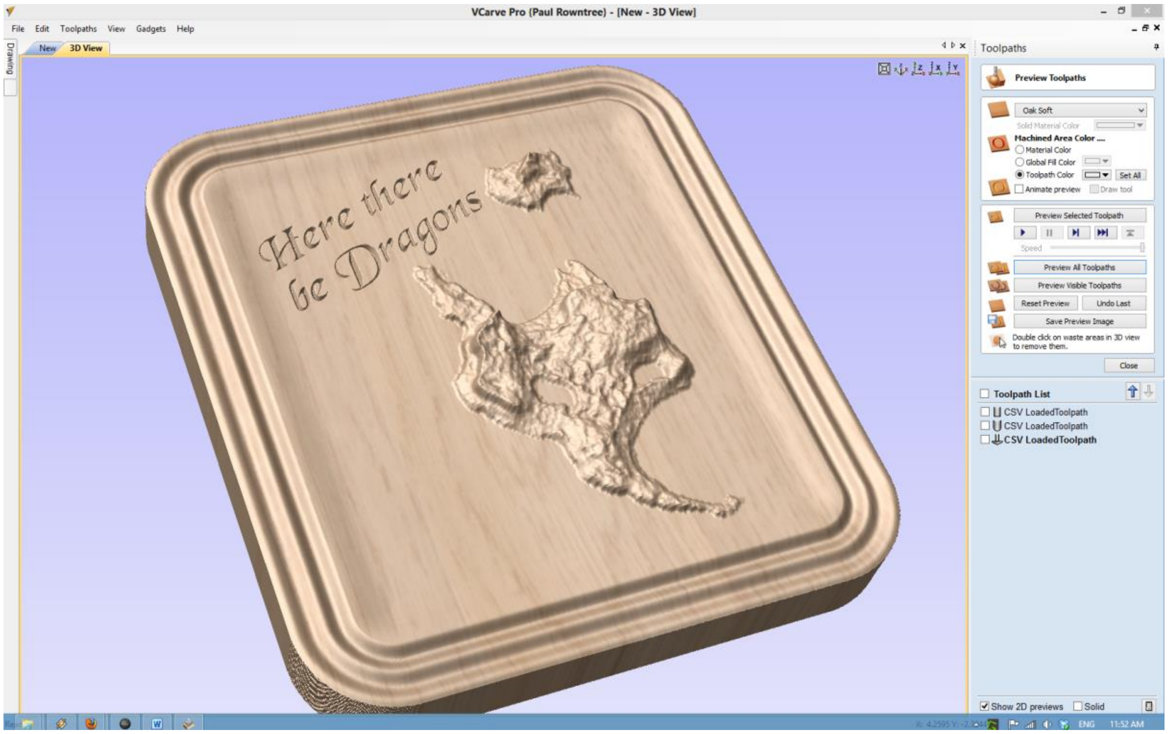


Here is an example of how to align toolpaths from different CSV files. The Aspire project had a topographical STL of Christian Island in Georgian Bay (created using BigTopo), a rippled frame (created using curves from the RippleToolkit) and some text. These two images are screenshots of Aspire showing the normal toolpath previews.



The toolpaths are (1) the STL image’s roughing path (0.25” end mill in raster pattern)+profile cutout, (2) the STL image’s finishing pass (0.125” ball-nose in angled raster pattern), and (3) the V-carved text (60 degree bit). The 3D portions were baked into a single component. Each toolpath would have their own X-Y limits; if imported as-is (no shifting, scaling, or rotating) they would reproduce the original design. Unfortunately, scaling, shifting or rotating the toolpaths would change the design because the toolpaths do not share the same bounding box (all adjustments are based on the bounding box), and you cannot adjust these patterns after importing. For a workaround, add two drill holes outside of the desired design (top-right, bottom-left) to each tool’s toolpath list, to make the three bounding boxes identical.

These toolpaths were exported into CSV files using the CSV PostProcessor. Importing it into VCP using **CSV to 3D toolpaths** gives the same pattern, ready to cut, or tweak and cut.



Oak Soft

○ Material Color
○ Global Fill Color
● Toolpath Color   Set All
□ Animate preview   □ Draw tool

Double click on waste areas in 3D view to remove them.

□ Toolpath List
□ CSV LoadedToolpath
□ CSV LoadedToolpath
□ CSV LoadedToolpath

☑ Show 2D previews   □ Solid

Here there be Dragons

Tips for Using CSV

1) **Please do not share, or ask others to share, copyright-protected patterns. If you did not create it, do not share it.**
2) It would be nice sometimes to be able to set a 'Start Depth' **above** the plane of the material, but Aspire/VCP do not allow this (for good reasons : what does a negative depth mean?); making controlled V-Carving in multiple passes is one application. CSV lets you do something better, very easily. Output your V-Carve toolpath via the CSV PostProcessor, then re-import that CSV file as a 3D toolpath. Scale the Maximum depth to say 75% of the original depth, (use the same V-bit!) and you will have an initial roughing pass which does 75% of the cutting, which you then follow with the original V-Carve toolpath to get the full cut. It is probably better because instead of *shifting* the toolpath in Z, it *scales* it; if the tip of the bit was in the material for the

original path it is also in the material for the roughing pass, just not so deeply. More of the V-Carving is done in the roughing pass.

3) It may be more useful to distribute files that are referenced to the top of the material and separate out cutout-profiles to separate toolpaths, so that the end-user is not bound to use the same material thicknesses.
4) If you get surprises, try to look through the CSV files to search for odd lines, gaps between lines that could break up the loaded information in an unexpected way, etc. This being said, the imports work as expected on my machines.
5) The import gadgets do not close any vectors or toolpaths. If you want to close them, ensure that the last line of the x,y or x,y,z data for a vector or toolpath is *exactly* the same as the first line. Alternatively, use the Vectric editing tools to close the shapes.
6) In the "Here there be Dragons" example, alignment was forced by including in each toolpath files drill points in the top right and bottom left corners of the job, and the importer used these to scale the imported toolpaths. If you are given toolpaths that were aligned by the author, but do not have registration marks, you can manually add the same three lines of text to the top of each CSV file that sets points just outside the desired shape, and this will force registration. For example, the Sombrero_TOP.csv file extends from -5.000 to +5.000 in both the X and Y directions, and you may have a roughing toolpath and a finishing toolpath (which will cover the same portion of the material but not the same tool trajectories because of different tool diameters), and perhaps a V-carved text section in the bottom right corner. Adding the following three lines to each file ensures that the (x,y) data in the file extends across exactly the same region. For toolpaths that have Z=0 at the bottom of the material, change the third value to the material thickness+0.01 (e.g. if the material is 2" thick, use 2.01 to avoid cutting). Don't add the comments in red.

-5.001, -5.001, 0.01	fake point at bottom left of toolpath, above material
+5.001, +5.001, 0.01	fake point at top-right of toolpath, above material
;	force a break to the next (real) toolpath

If you use **CSV** to produce something interesting, please consider posting photos on the Vectric forums. If you have any questions, I am frequently on the Vectric and CNCZone forums as PaulRowntree, or you can reach me via the PaulRowntree.weebly.com website that you may have downloaded the **CSV** package from.

And of course, if you really like this or if it has helped you in a significant manner, please consider supporting CSV's continued development via the http:\\PaulRowntree.weebly.com website.

More complicated custom models can be prepared upon request.

Cheers!

PR