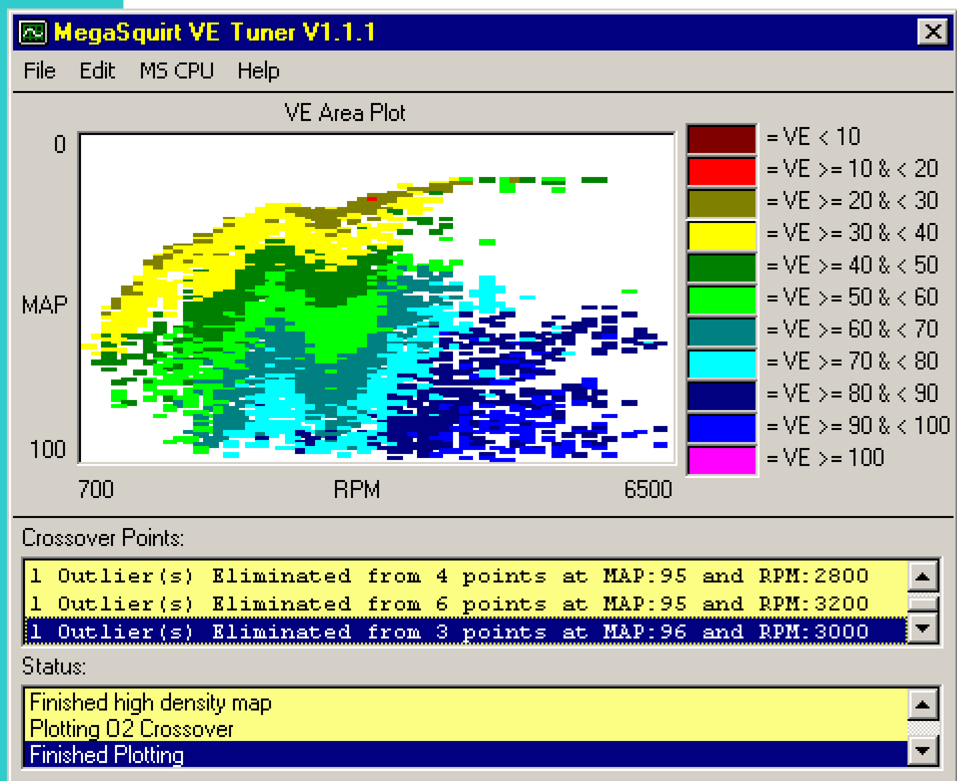
MSTweak3000 User Manual



How to set up and use MSTweak3000 to adjust and optimize the VE table of your MegaSquirt fuel injection controller.

This manual explains how to setup and use MSTweak3000 to fine tune and optimize the volumetric efficiency table in the MegaSquirt fuel injection controller. This software and manual are experimental, the author of these works assumes no responsibility for its use or consequences of its use whether correct or not. The author also does not guarantee the correctness or accuracy of these works. And remember; always backup your old settings!

For more information you can contact the author either by e-mail at mailto:biggranger@tds.net or through the MegaSquirt message list at http://groups.yahoo.com/group/megasquirt/

Manual Version 1.1
MSTweak3000 Version 1.1.33
Date 05.21.03

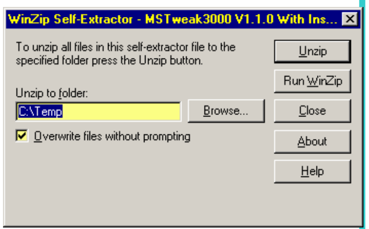
Index

Index

Starting up MSTweak3000 1

Installing MSTweak3000

MSTweak3000 comes 2 ways: 1 the complete program with examples and the latest documentation, or the executable alone. It is highly recommended that the complete program with the installer be loaded and installed the first time to make sure your computer has all the latest support files and latest documentation. The complete version comes as a self-extracting executable, just double click the file once downloaded and it will prompt you to unzip it into a temporary directory (C:\Temp). Press the <Unzip> button and the files will be placed in the directory specified. Using the Windows Explorer find the folder where the files were unzipped to and double click the Setup.exe file.



If you're upgrading to a new version of MSTweak3000 is not necessary to uninstall the current version, just download the executable and place it in the folder where the existing MSTweak3000.exe is replacing the old file.

Setting Up MSTweak3000

Before doing anything with MSTweak3000 for the first time it is necessary to adjust some settings that will be specific to your particular MegaSquirt installation. Start up MSTweak3000 and press <Edit> and <Settings> on the menu bar. This will bring up the settings menu; this menu is broken into 6 groups.

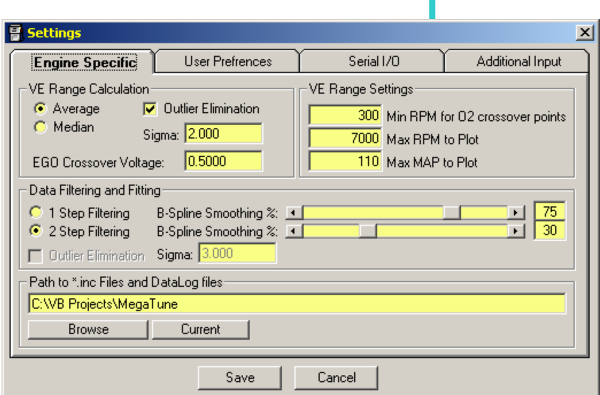
1) VE Range Calculation. This controls what is done when the datalog is first read in, and what voltage the oxygen sensor outputs at a 14.7:1 AFR. When reading in a large datalog it is possible to have many overlapping data points, when this happens MSTweak3000 needs to know how to handle those points, either take the average of the overlapping points or find the mean of the points (average is recommended). It is also possible to have flyer points in the datalog eliminated, is can be done by checking the outlier elimination box and specifying a sigma value (2 or 3 is recommended).
2) VE Range Settings. These are the settings MSTweak3000 uses for plotting extents and where O2 crossover points are calculated. Min RPM for crossover points should be set the same at EGO active above in the enrichments screen of MegaTune. Max RPM to Plot is the redline of the motor, and Max MAP to Plot is the highest MAP value your engine is capable of generating (about 100 for NA, and 200+ for Turbo).
3) MegaSquirt I/O. This does nothing yet
4) Units. Set up units to something a little more familiar.
5) Engine Load Type. Settings for Speed Density or Alpha-N.
6) Data Filtering and Fitting. This is for post processing the raw data (crossover points). There are 2 settings, 1 step filtering creates a B-Spline curve through all the available MAP points at specific RPM's, 2 step filtering includes the 1 step process then it creates a B-Spline across the available RPM points at specific MAP values. B-Spline smoothing is a value of how smooth the spline should be; 0 percent is not smooth at all, and all calculated points are calculated linearly between data points, 100% is the smoothest curve possible through the points (see illustrations). It is recommended that Step 2 be about half of step 1's value.

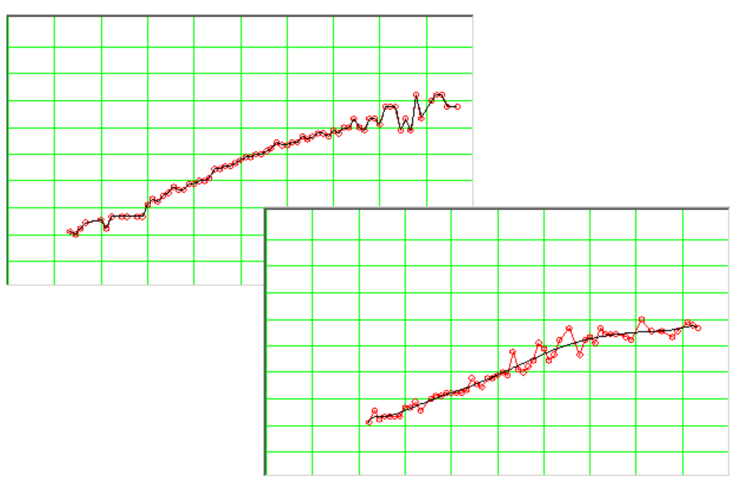
? 14.7:1 AFR.

This is the optimum Air Fuel Ratio (in pounds each of air vs. fuel) for maximum power and efficiency. Standard oxygen sensors report this at 0.5 Volts, the more advanced wide band sensor reports this at 2.50 Volts.

? Sigma?!?!

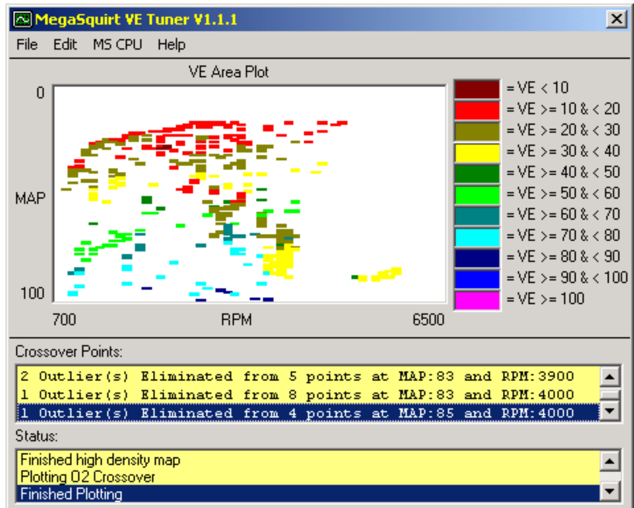
Sigma or standard deviation is the sum of the squares of the delta from average of a dataset. In our case if a particular datalog bin has the values 30,31,35,32,60 we can safely say 60 is an error the sigma of the dataset is 12.6 so anything exceeding +- 12.6 from the average is deleted.





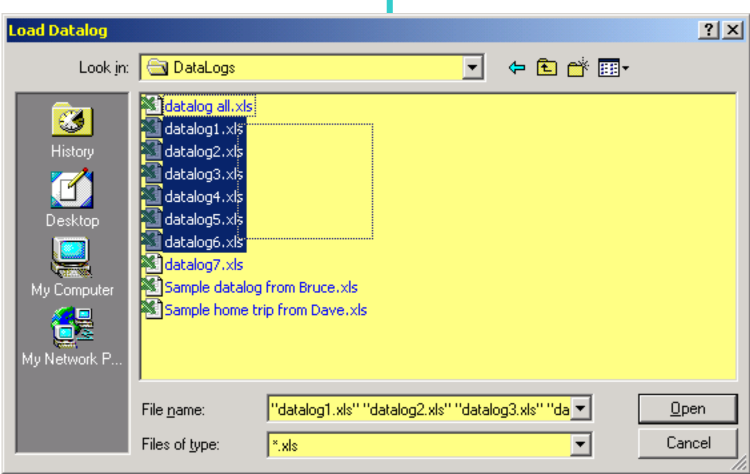
The Main Screen

The main screen is the first window that pops up when MSTweak3000 starts up. From this window datalogs can be imported, saved, and loaded. O2 crossover points can be saved as an excel file, plotted on the screen or printed. And most importantly the modify VE map window is accessed from here. In the future MSTweak3000 will be able to connect directly to the MegaSquirt controller. The plot window displays the calculated engine VE value at specific MAP and RPM points based on the oxygen sensor output, and applies a color value on them based on the VE value.



Loading a Datalog File

Loading a datalog file into MSTweak3000 is pretty straightforward, but getting a good datalog is a different story. To learn more about how to get a good datalog see the section of this manual titled "How to get a good datalog". To load a datalog go to <File> and click <Open Datalog>, this open the open dialog window. It is possible to open several datalogs at one time by holding the control key and selecting multiple datalogs, this will append all the datalogs into one big raw data map. It is also possible to just click <Open Datalog> again and import another datalog, when doing this a prompt appears on the screen asking whether the datalog should be appended to the current raw data map or a new map be created.



When to datalog is loaded the O2 crossover points are found, and the engines VE at that MAP and RPM point is calculated. If outlier elimination is selected then the outlier points will then be deleted.

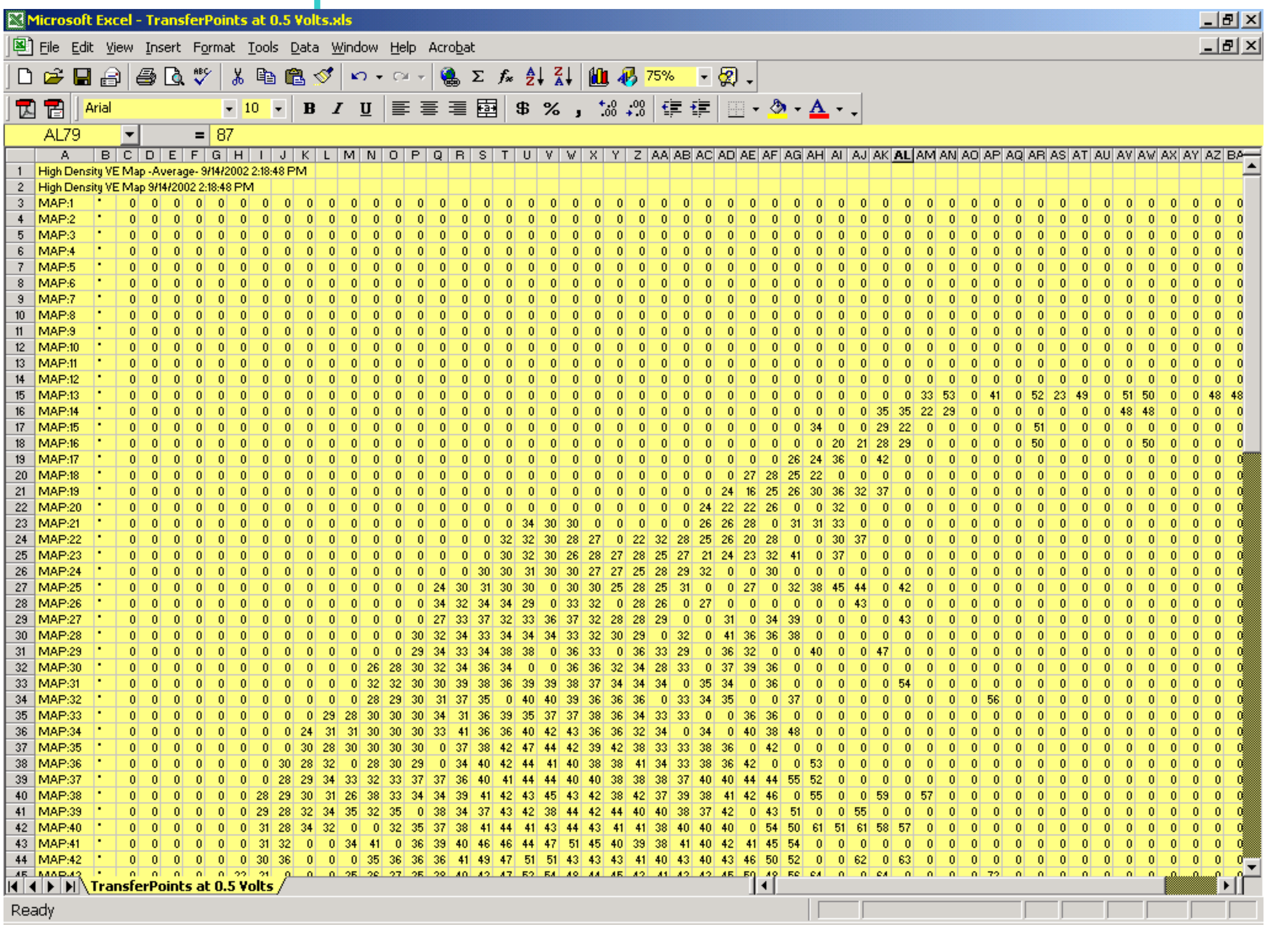
Loading the datalog process can be time consuming so there is the option to <Save Raw Data Map>, this will save all the crossover points and the calculated VE values in a smaller file then can then be reopened later much faster.

Once the datalog has been brought in the resulting points will be plotted on the screen, going to <File> and pressing <Print Raw Data Map> can also print out this plot.

To send the

Saving the Crossover points

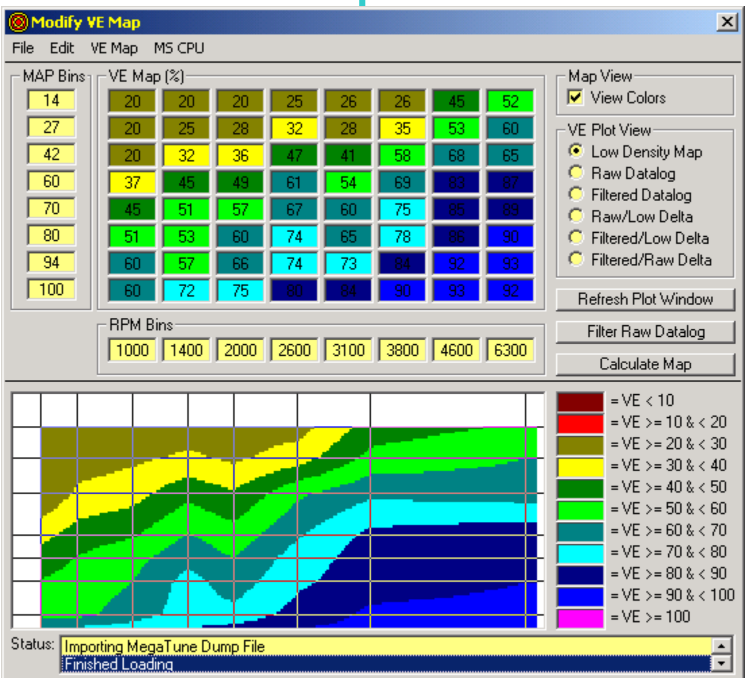
To see exactly what MSTweak3000 did with your datalog is you can go to <File> and press <Save Crossover Points>. This will create an Excel file with the raw data map and the crossover points.



Modifying the VE map 4

The Modify VE map Screen

The modify VE map screen is where all the action is. From here it is possible to tweak in your existing VE map or create a whole new one; based on the calculated VE points from the datalog.



This window allows the user to import and export VE maps through the VEX file format which is supported in the MegaTune software, which will then transfer the new map to the MegaSquirt fuel injection controller.

From this screen it is also possible to read and write the VE table to MegaSquirt via the <MS CPU> menu button, and also available are an array of tools to help manage the VE table in the <VE Map Tools> menu button. These tools allow the user to scale the VE table based on a required fuel PW change, re-calculate injector flow based on changing the fuel pressure, erasing the low density map to start over, and modify the VE entries in the map based on target air fuel ratios for map bins.

What is data filtering and how do I use it?

The data filtering function on the Modify VE Map window is separate from outlier elimination, which is done when first importing a datalog. Outlier elimination will get rid of flyer data points that do not make sense, data filtering will smooth out the data and fill in any empty gaps. In 1-step filtering as the torque curve moves across an RPM band at a given MAP reading, filtering draws a line across the table. So you're calculating different torque curves (going from low RPM's to high RPM's) for every possible load (MAP reading). The 2-step b-spline just does more interpolation from the other axis further smoothing out the data and filling in gaps.

How much filtering to use depends on the quality of the data and the engine being tuned. It is possible to over and under filter data. Obviously torque peaks are different from engine to engine. Stock V8's and V6's have a nice flat torque curve so setting the b-spline filter up to 100% should be safe for that type of engine because it will clean up any of the peaks in the data. But for a properly tuned high revving "cammed" up motor with a tuned intake and exhaust there will be 1 or more torque peaks that come up rather abruptly and a lot of filtering will eliminate good data, so I'd probably go with maybe 40 or 50%.

Ultimately you should compare the filtered data with the unfiltered data (the tools are in MSTweak3000 to see that, VE plot view -> filtered/unfiltered data). If you are chopping the peaks off of the collected data then the filtering is too aggressive.

If you do have large gaps in the collected map try getting data for that part of the map if possible and appending it to the data you have already collected to fill it in and get a better idea of what your map should look like.

How to get a good datalog.

In my testing I've found that setting the O2 +/- limit (in MegaTune) to 100% works great for setting up a really rough map or starting from the baseline VE map that comes preprogrammed in the controller, and if your map is more or less tuned in 50-70% will work too (keep this high though for tuning). The critical settings are O2 step% and ignition events per step. When tuning anything in the lower RPM range (1000-3000 rpm), set step% to 1 and ignition events to 32 (2000 rpm with a V6 = 100 events per second = about 3% change in a second). Then when tuning the higher rpm's with a rough map; step% at 3 and ignition events to 64 (about 3.5 changes per second at 4500 rpm). Once things settle down a bit when the map is tuned better I set step% to 1 and ignition events to 72, which gives the closed, loop control some more stability and allows for better fine-tuning at the higher rpm's.

Here is the formula for figuring out your O2 adjustments per second: ((rpm/120) * cylinders) / ignition events per step. For fine-tuning I like to keep O2 adjustments per second between 3 and 5 for the particular rpm bands I've been tuning. Roughing in maps try to stay between 5 and 10 (depends on how good the O2 sensor is, if it's old go slower).

Once the map is tuned in I set the limit to 5% (it can go higher depending on how questionable the map is) step% to 1 and ignition events to a value that would switch about 4x a second at your average cruising speed.

Outlier Elimination:

As of right now this only applies to importing a datalog. When MSTweak3000 reads in a datalog it's possible that the same RPM and MAP point (in the internal high density map which can be saved to excel) have several entries collected in them. This is a good thing because we're building our VE map out of a lot of data, the more the better. For example let's say at cruising speed 1900 RPM and 40 kPa we've collected 8 entries (although I'd expect several hundred possibly) and these 8 entries are 32, 34, 33, 34, 32, 55, 34, 33. It's obvious that the number I'm looking for is around 33, but I have a flyer in there that will throw off my result. I need a way to get rig of the 55, and statistically I have a perfectly valid tool to do that with, it called outlier elimination, and it's based on a multiplier of the sigma value (the sigma of the above samples is 7.77). So if we eliminate everything that is 2 sigma (2 * 7.77=15.54) anything that is +/- 15.54 / 2 from the average of the entries (36.66) is thrown out. That is how we get rig of flyer values.

Now we have an average of 33.1 (33 in MSTweak3000) not 36.66 (37 in MSTweak3000).

A sigma of 2 or 3 is good, the lower the number the more data you will throw out, and too high a value will not discard any bad data.

Average/Median:

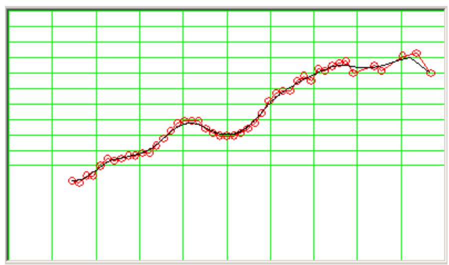
This is the method for calculating a value from our now 7 entries (since we've deleted the invalid 55 value) the average of the 7 entries is 33.1, and the median is 33. Average is probably a more statistically sound method of finding the correct value while median is the number in the middle of a set of numbers; that is, half the numbers have values that are greater than the median, and half have values that are less.

Crossover Voltage:

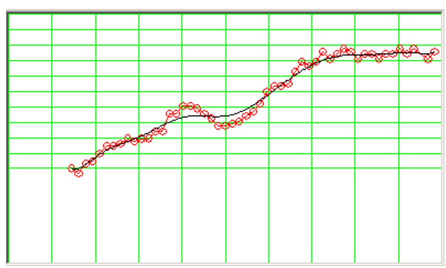
This is the same as EGO Switch point in MegaTune, and should have the same value. 0.5 volts for the narrow band O2 sensor, and 2.5V for the wide band (or whatever voltage you're tuning for). Just a note, this screen is changing soon because of the new wide band implementation routines I'm putting in for air fuel ratio based on engine load.

Spline Fitting:

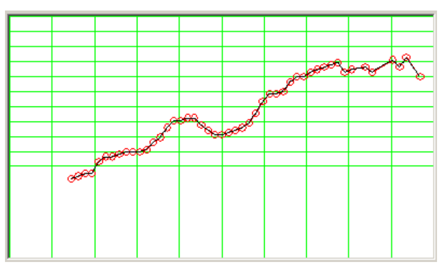
1-Step fitting will fit a curve to data points across the engine load (MAP) as specific RPM points. This should be pretty linear, so the filtering percent can be a larger value (75%), 2-Step fitting includes 1-Step, then fits a curve across the RPM band at certain MAP values. The second step is usually a lower value then the first, because we're now looking at the torque curve. What values should be used for my engine.



Good example of a correctly set spline % (Note this is a second step across the RPM band, so you're looking at the torque curve). See how the black line follows the red circles



An example of too high of spline fitting %. Part of the curve is chopped off.



And too low of fitting, there is no smoothing going on at all.

1 - load your datalog(s) -> explained will in the manual *** Correct ***

2 - load your current VE from a file or direct from MS (?) *** Only from Datalog now **

3 - click on 'Filter Raw Datalog'

3.1 - adjust your RPM bins to line up with the vertex of the slopes of the VE plot. You want the sections in between the RPM bins to be as straight as possible. You also want an RPM bin at a very low RPM (say 400) and one near redline. This leaves 6 bins to move around.

3.2 - adjust your MAP bins in the same, one at the lowest MAP point you see, and one just past the highest. Then equally space the rest out. This will give you a good first off map.

4 - click on 'Calculate Map'

4.1 - this fills some sections of the 8x8 VE map. The sections it didn't fill in will stay "0" these will need to be guessed (an educated guess). You may also want to export the high density map (on the main screen).

4.2 - Now see how well the maps over lap. Press the filtered/low delta radio button. This will display a plot of where the 2 maps are different. Try changing some of the VE values and replot so that most of the plot is green. (Don't worry too much about this the first time through. It may take an iteration or two of datalogging and changing VE bins and values to get the VE table good enough to worry about).

5 - menu 'MS CPU', click on 'Send VE Map'

6 - menu 'MS CPU', click on 'Burn RAM to FLASH'

6.1 - repeat steps 1 through 6 a few times, and you're done!