



This software along with its functions is a form of art. There are small prerequisites in the knowledge of mechanics, electronics, minor physics, a fair dose of algebra (calculus for knock), and a close familiarization with one's own automobile to make a great tuner. Knowledge in data-logging in some way, shape, or form **MUST** be used to perform just about any tune. It is one of the only ways to see and log your tuning effects, besides going to a Dyno-Meter facility. **DO NOT GO BY FEEL!** Give yourself time to familiarize yourself with the Data-Logging software, and practice utilizing the software features and functions in **safe conditions**. The inter-state or desolate back-road is the **WORST** place to familiarize yourself with the tuning software. In fact, most gains will probably be made under the guise of a true tuner at a Dyno-Meter facility. Still the ability to achieve personal automotive power gains plus the acquired skill over time, might lead me to you for a tune one day. But till then, proceed with caution because you **WILL** void any warranty, and have no one liable for any damage but yourself for any and all tuning mistakes, possibly "killing" your ECU or causing mechanical chaos. This is called a disclaimer and is far from trying to keep you from ECU tuning, although a **MINIMAL** amount of knowledge will be needed to tune your vehicle **CORRECTLY**.

This software was not designed to fix problems out of the scope of performance modifications. The necessity to make sure the automobile being tuned is in the **GREATEST of MAINTENANCE CONDITIONS** is not to be under-stated. There are excellent methods via OBD II diagnostics and Hanes to get a higher mileage/kilometer vehicle back to standards before the thresholds of the engine can be met (some OBD II diagnostics are included with ECU Flash/data-logging software). The principal is to start with a good *Base* and work from there.

At the time of printing this "Manual Project" was referenced using only Mitsubishi Vehicles. Subaru is a territory I have **YET** to approach. I intend to mess around with my Father 05 FXT after the warranty has run out. Until then be careful of the use of this manual for Subaru vehicles.

INFO - Green
CAUTION - Blue
DANGER - Red

Install *ECU Flash* to desired PC (Laptop), when finished plug in the *Tactrix* cable in UNPLUGGED TO VEHICLE to install drivers. This makes things less complicated when using the program or possibly trouble-shooting vehicles that are stubborn to flashing, not having to deal with driver issues also.



Tactrix 1.3 Mitsubishi Cable

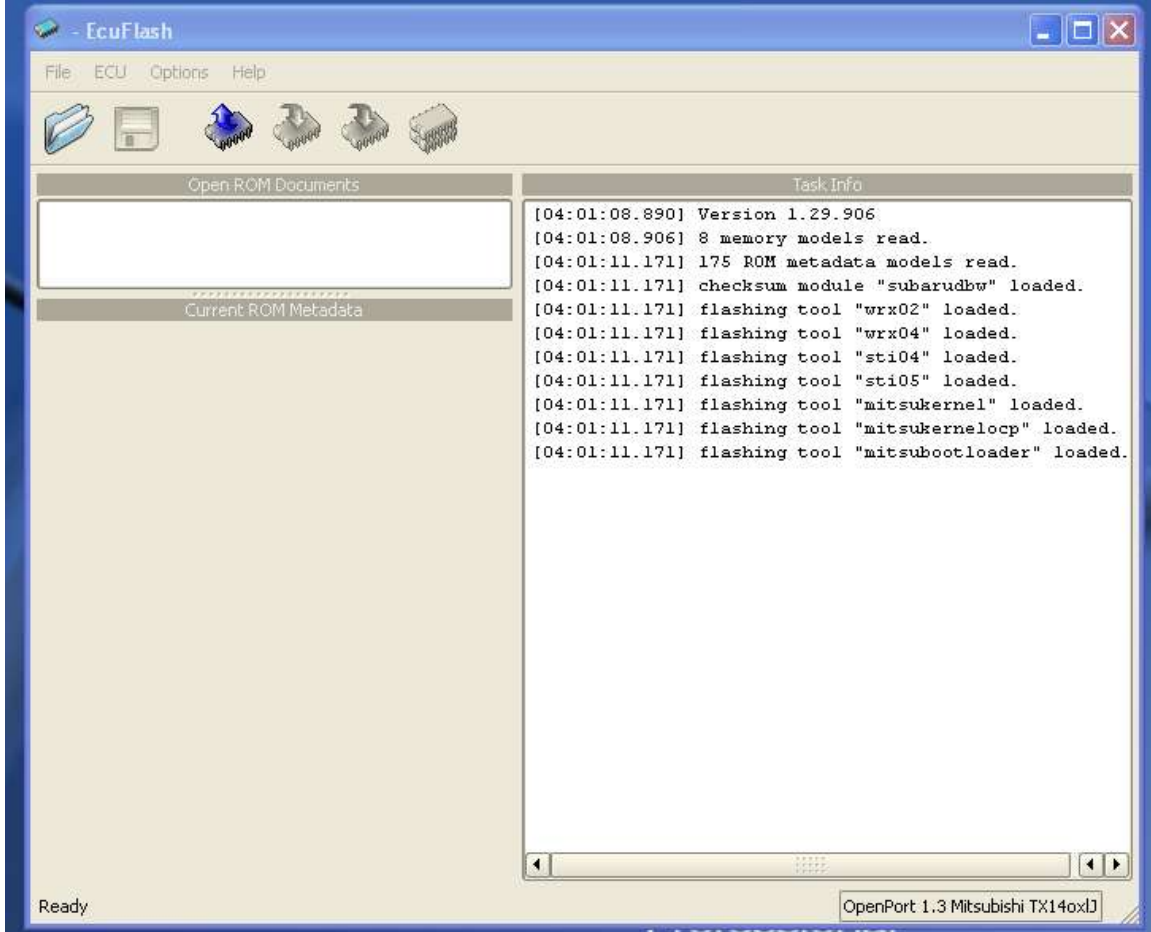
Plug in the *Tactrix* cable into the OBD II connector of your vehicle. Also the white flash initiation block should be plugged in unless you are just data-logging (data-logging does not require initialization of the ECU). Plug in the USB portion to your Laptop or other PC device and wait for device recognition. **Although it does not really matter on sequence, just make sure it is SECURE!**

Also make sure the cable is in the way of operating the vehicle in a safe manner (like not putting the laptop on the dashboard).



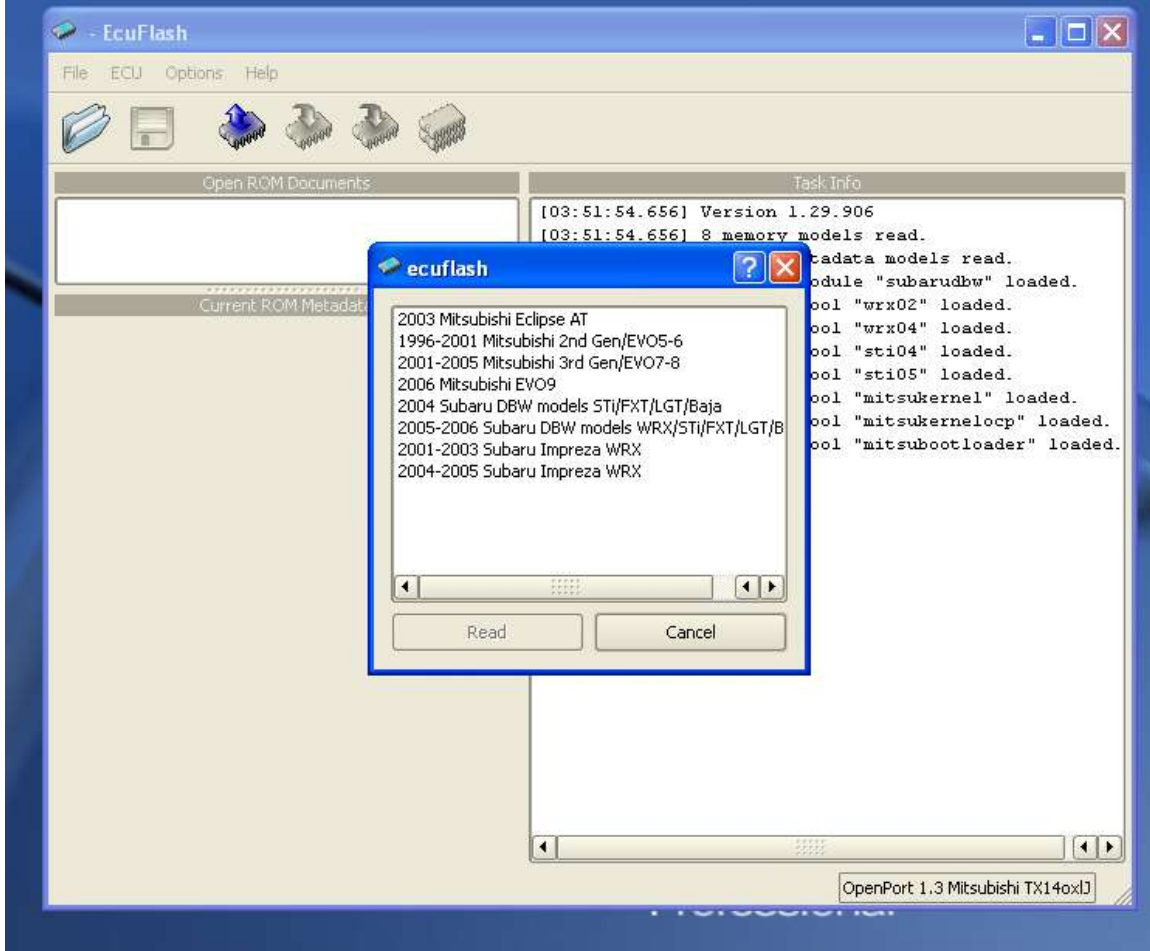
Mitsubishi Style OBD II / Reflash Block

The most common 1st step in learning to use your tuning software/hardware is to pull a ROM from the automobile being tuned. So let's familiarize ourselves with the software by using it a simple and safe 1st run. Below is an example of the *ECU Flash browser*, with the hardware recognized by the "OpenPort 1.3 Mitsubishi TX14oxlJ" in the lower right-hand corner of the browser. (OpenPort may differ depending on model cable).



Open Port 1.3 connected and ready.

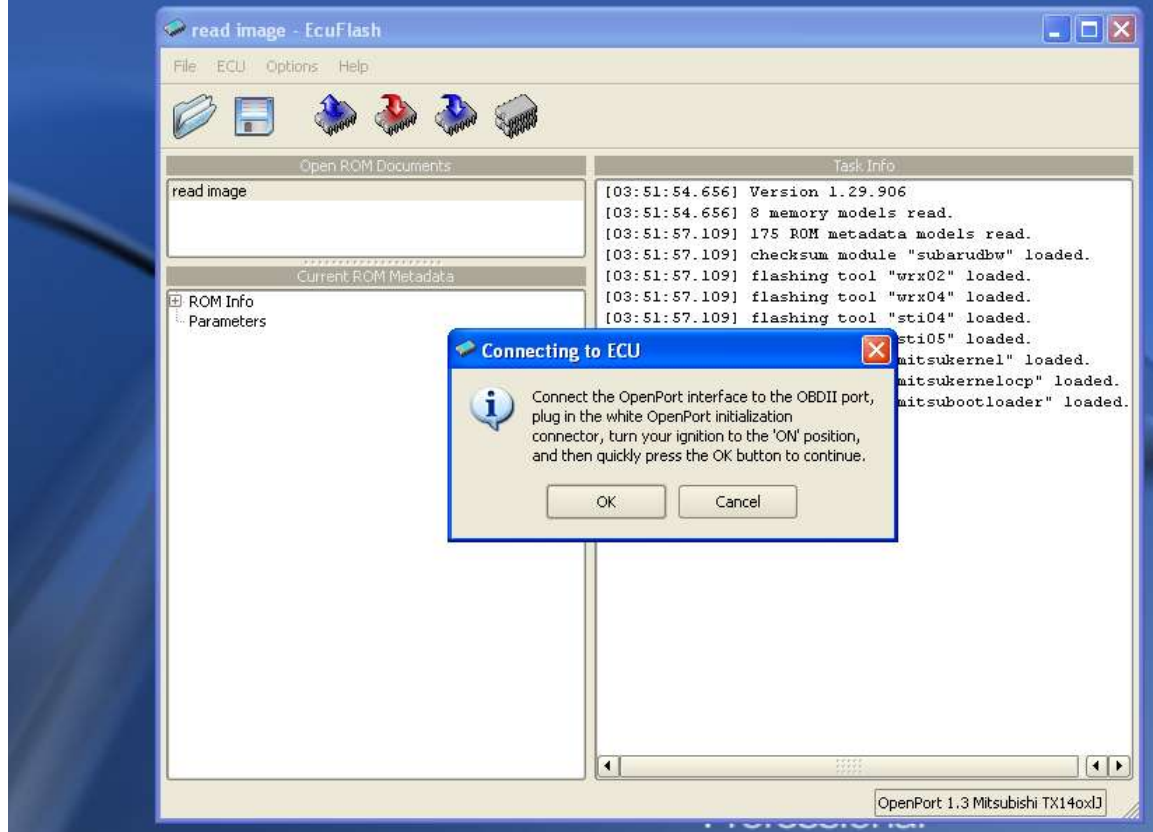
With the ignition in the [ignition off/engine off] position, go to the first icon **or** in the *ECU Flash* browser options, *ECU* and select, *Read from ECU*. A panel will come up and ask of which vehicle you would like to read. With Mitsubishi's it is a little tricky if it is an automobile not directly listed in the panel. You will have to choose the closest year and model to get a link to your vehicles ECU.



Choose your Year and Model. (Actually a 03 Galant – uses Eclipse)

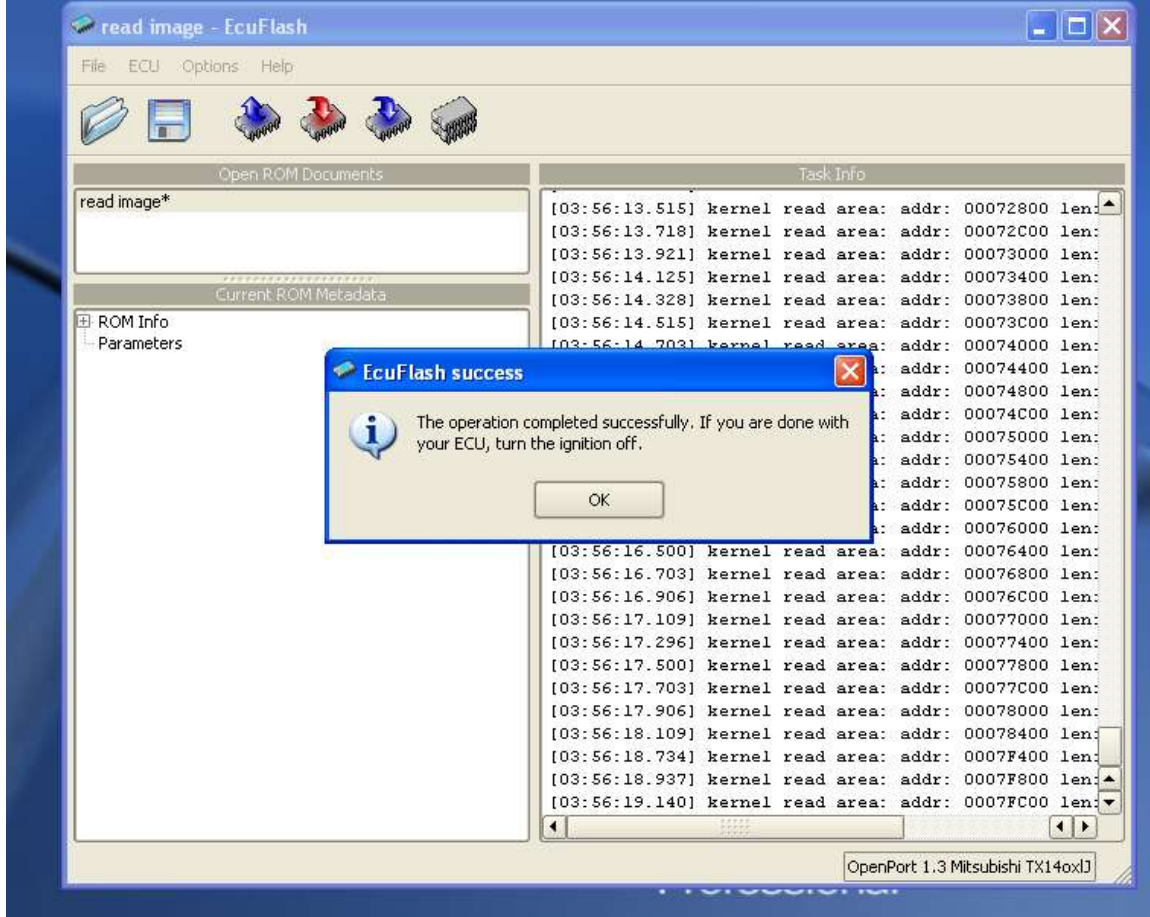
What is being done is very much a conversation between *ECU Flash* and the ECU, but to first have a “good” conversation there needs to be the correct acknowledgement between our talking pair. The kernel (that you are choosing) is the software needed so the two may correctly communicate in a common language.

After choosing a reasonable year/model read vehicle, you will be asked to either 1. (Mitsubishi) Make sure vehicle’s ignition is in the ignition on/engine off position and press *OK* to continue. 2. (Subaru) Turn ignition to, ignition on/engine off and **QUICKLY** press *OK* to continue.

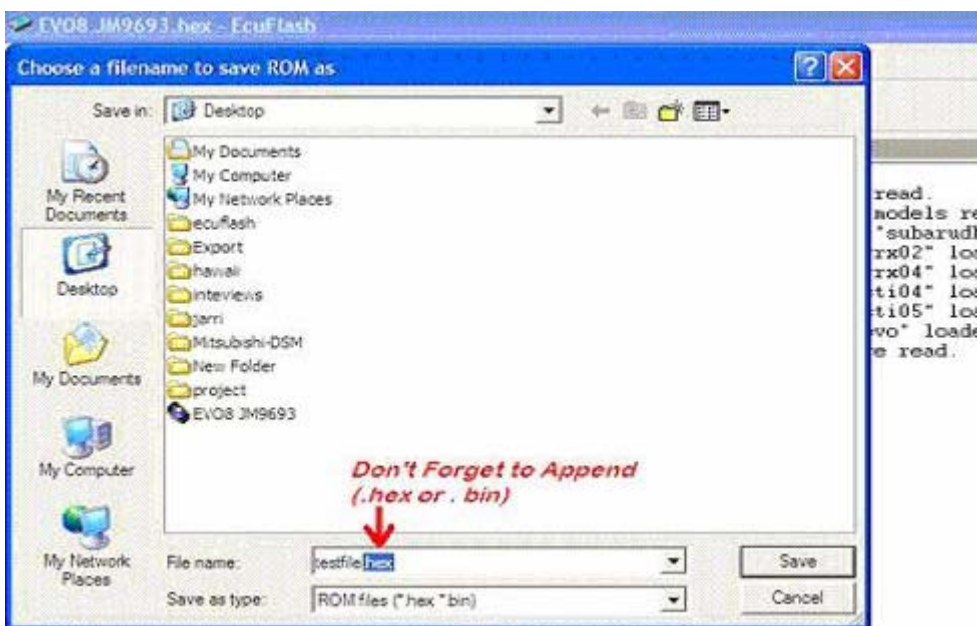


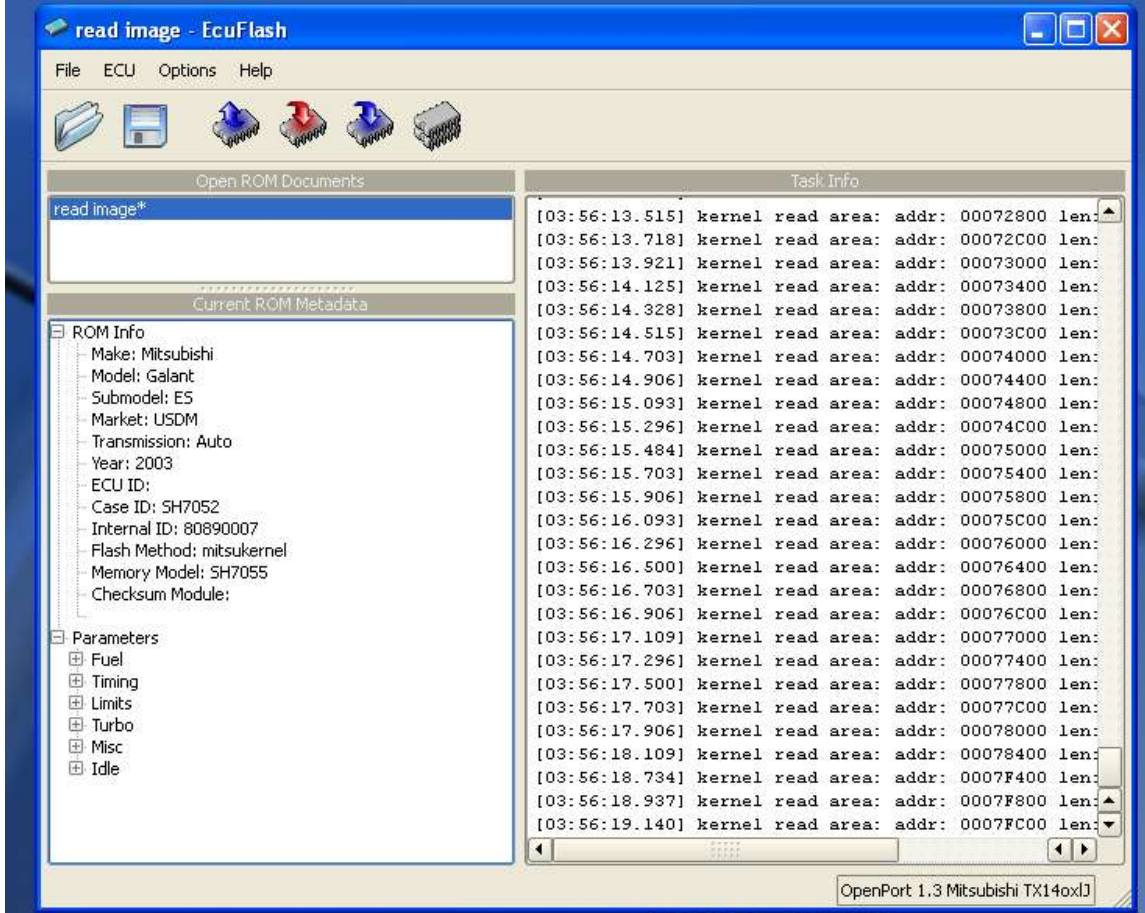
Pre-Initialization sequence.

This is when you just sit back and wait patiently until the entire ROM is read from the ECU. The program will scroll the memory sectors that have been read (in the Task Info window of *ECU Flash* browser) and when finished, ECU Flash will “bing” and you may now turn the *ignition to off* and remove your interface cables. You may also hear the fuel pump or relays “click”, this is completely normal.



Save a copy of the ROM by selecting either, *diskette icon* or *file* then **save ROM** (or more preferably **save ROM as** on the browser options). If you just used **save ROM** option, you must make sure you add the extension ".hex" or ".bin" to the file name to be able to read the ROM image later.

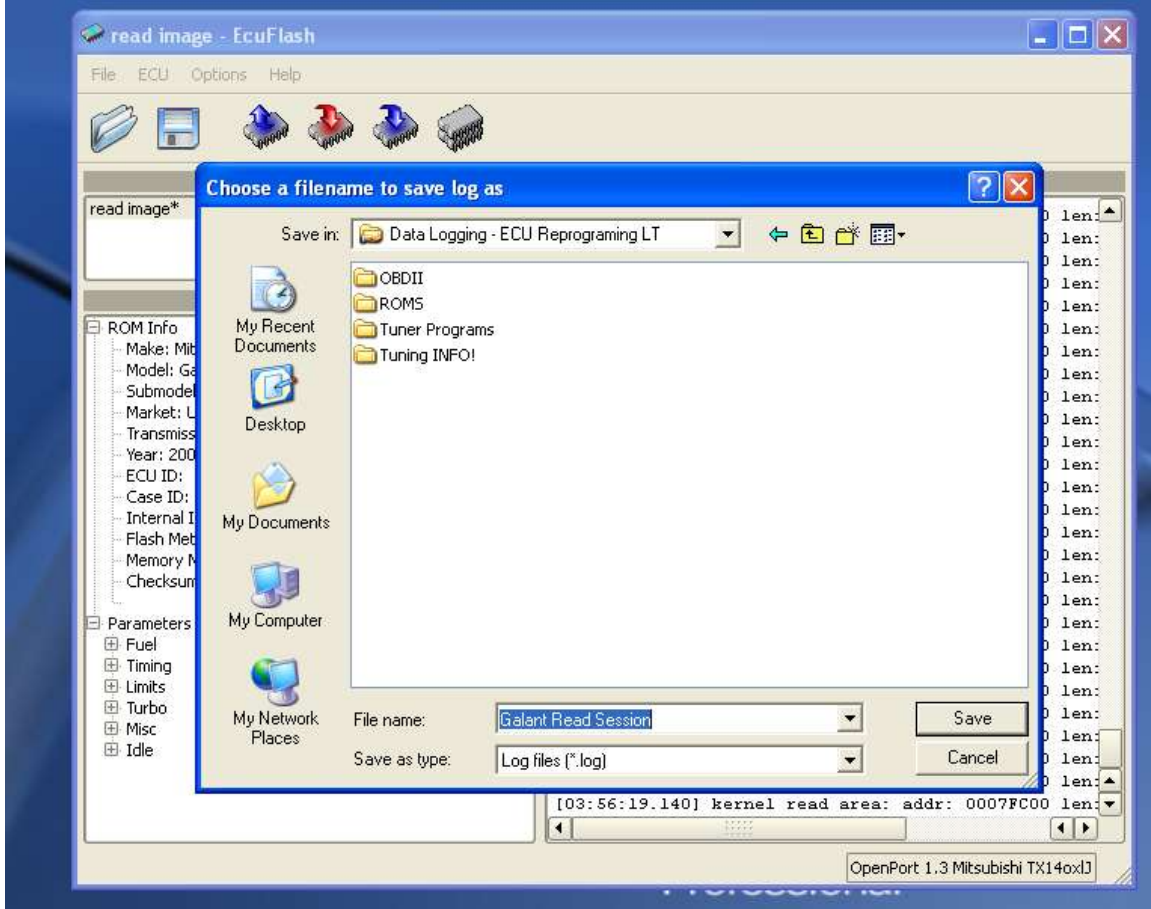




Log Screen after ROM pull.

After completing a "session" (or any time you plug into your vehicle) it is wise to save a copy of the **Task Info** Browser. This takes an entire log of all communications between PC and the vehicle; this is very useful in trouble-shooting failed flashes and other anomalies that might pop up.

Create a personal file for these logs, and save them for later reference.



Save session log.

Reading maps, scales, and single data values.

There are three different types of data tables that are used for ECU Flash, which are labeled in dimensions: 1D, 2D, and 3D (1 dimensional, 2 dimensional and 3 dimensional). This is in respect to how many variables are taken into consideration in the judgment of the particular map, scale or value. One of the dimensions or variables in all of the *Parameters* of a vehicles ROM may be manipulated or tuned (XML – Definitions are to be discussed later). There are also single data values that are usually giving proportions of air, fuel, load, boost/vacuum, temperature, speed, or electricity to a particular system of the vehicle (plus many others for EVAP system and knock controller).

For -1 Dimensional maps,
MAF size = grams of air/second -1D

For - 2 Dimensional maps,
MAF Scaling = Load – 1D, g/s – 2D

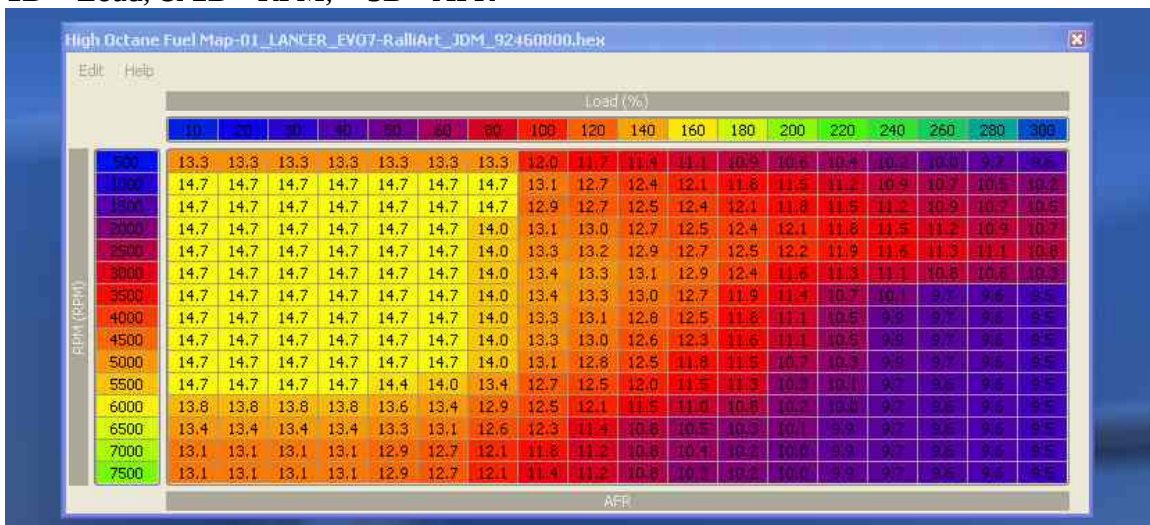


1 Dimension -1D



2 Dimensional – 2D

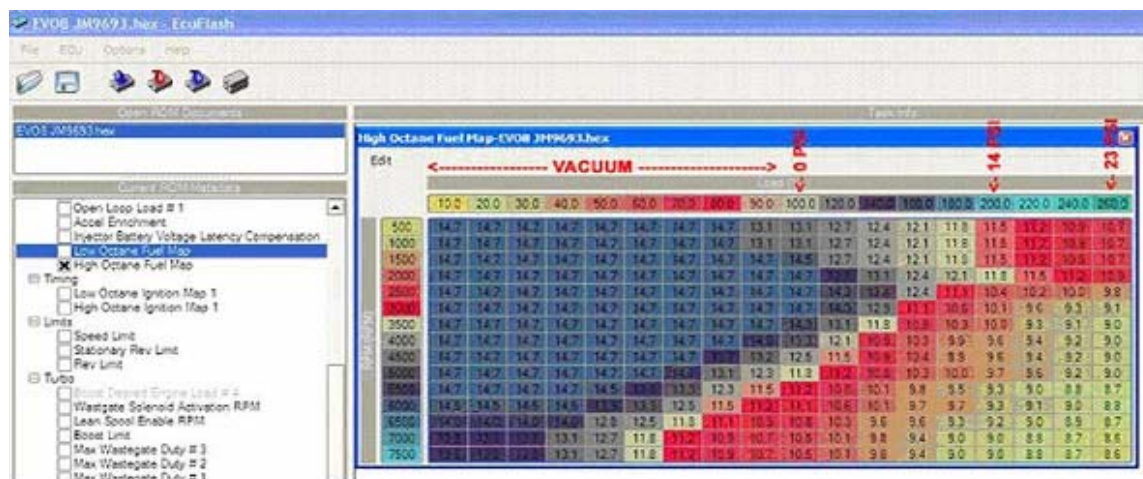
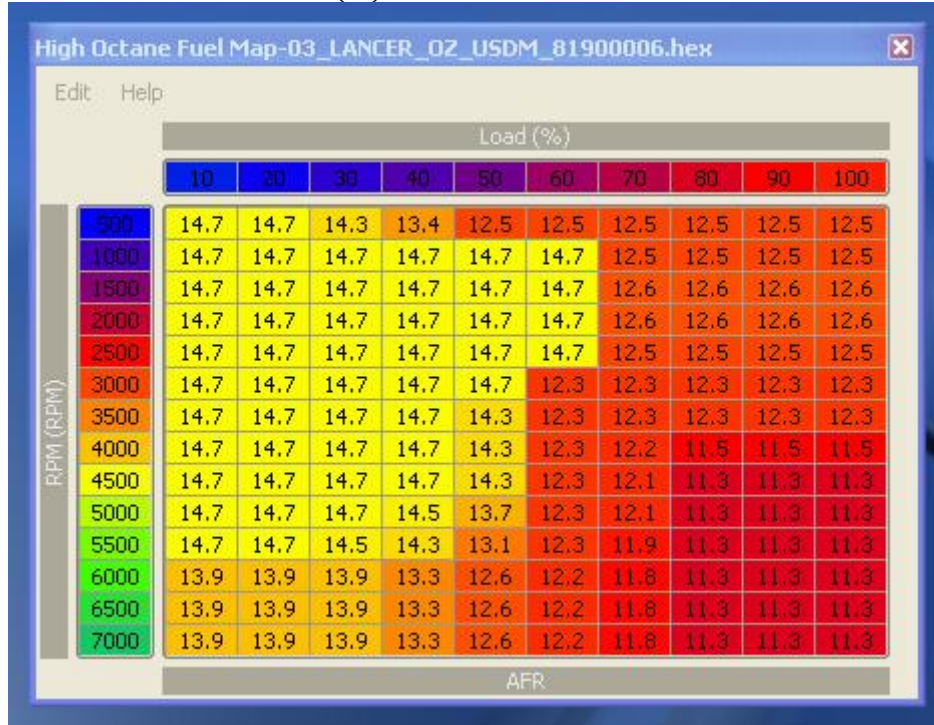
For; 3 Dimensional maps,
1D – Load, & 2D - RPM, = 3D - AFR



3 Dimensional – 3D

The ROM image itself is genius in that it carries all the data information to particular controller system (no matter if it be 1 or 3 given variables within its data) and this information is used to give the ECU the logic it needs to operate within given ranges of input and output devices. Along came some altruist Programmers/Mathematicians, who were able to decipher the codes into scales, tables, and ranges (along with many strings of program data) that can be understood by a “minimally” mechanically-inclined individual. With the help of *ECU Flash*, tuners can keep up to date on data manipulation, based on the modifications. Knowing all this information can also help the tuner be more aware of all the input/output of their/customers vehicle in general, also very helpful in pointing out the next “modification” that can/should be installed.

Notice how certain Load (%) and certain RPM = AFR values on this fuel map.



Load going over 100% is an extrapolation of the amount of load being produced by increasing the atmospheric pressure. Air has weight and is measured as psi or BAR. It is quantified in the metric system in kilopascals (kpa). 100 kpa is 0 vacuum, 200 kpa or 1 bar of boost is ~14.7 psi. Anything below 100 kpa is vacuum.

Load is measured by the amount of air that is passing through the mass air flow sensor (MAF) of the vehicle. One might think that it would be more advantageous to use throttle position, but this does not take in to consideration the environmental conditions like air pressure, temperature, or moisture. Just because the throttle is at 100%, does not mean there is 100% engine duty. Revs are a gradual build and need a graduated reference to keep the most accurate AFR.

Some important sensors and their functions.

MAF – Mass Air Flow – The MAF consists of a conglomeration of many different sensors that when put together in an electrical equation, is “fairly” accurate. The next section of this guide focuses on just the principals of MAF Sensor principals. This is a rather important part to any DSM/Subaru auto for proper fuel burn. They are;

Barometric Pressure Sensor (integral part of the MAF) – This sensor measures atmospheric pressure, density and for elevation.

Intake Air Temperature Sensor (integral part of the MAF) – Measures air temperature buy using a thermoresistor, giving an electrical measurement for the ECU to judge its functions. Note: as temperatures increase, the resistance decreases. Temperatures decrease, resistance increases.

O2 Sensor (Oxygen Sensor) – Used to measure the amount of Oxygen in the exhaust using a galvanic battery that varies in volts depending on O2 concentrations. O2 sensors allow the ECU to help adjust AFR.

Volume Air Flow Sensor (integral part of the MAF) – Measures the volume of air entering the engine by measuring the sound of the air in Hertz, as it passes through a set of tuned tubular or hexagonal channels. The air speed and air sound are relative measurements (scientifically) allowing it to be quantified as Hz *or* Grams/Second simultaneously.

Camshaft Position Sensor (CMP) – The CMP produces a signal that the PCM uses to identify the number 1 cylinder and to time the firing sequence of the fuel injectors.

Crankshaft Position Sensor (CKP) – The CKP sensor produces a signal for the ECU to determine crank positioning (in degrees) and load.

Engine Coolant Temperature Sensor – Is a thermoresistor that sends a voltage signal to the ECM. This helps the ECU to determine a number of ranges, like idle, air/fuel ratio, timing and EGR control.

Fuel Level Sensor – Integrated into some vehicles, responsible for the “low fuel” light.

Fuel Temperature Sensor – Used for the EVAP system not really dealing with tuning.

Fuel Tank Differential Pressure Sensor - This is the sensor that caused a DTC if you did not get the “click,click” sound from your fuel cap and inadvertently left the fuel tank at ambient pressure.

Knock Sensor - This is an acoustical/electronic measuring devise used to detect pre-ignition or a range of mis-fire related detonation. I will put in a little section just on this topic and the theory related to engine knock calculation.

Flapper Door –

This style of air metering uses a spring-loaded, hinged door to estimate the amount of air entering the engine. When air flows through the meter body, it has to force the door open, and the amount of travel corresponds to the volume of air flowing towards the motor. Once the system has determined the volume of the air entering the motor, it needs to solve for the mass flow, which can be done with two additional sensors: temperature and pressure. Both of these sensors measure right next to the flapper door, as it is crucial that they match the properties of the air at the point where the flapper measured the volume flow. This system works, but is probably the least effective of all of the ones in this article. It is fairly restrictive due to the small cross-sectional area of the door, and it also has a limited measurement range, so modified vehicles can run into problems with it.

Karman Hz MAF - When air (or any fluid) flows through a pipe, it has a tendency to form vortices behind any uniform object it comes into contact with. These vortices are like mini "whirlpools" of air, and the rate at which they form corresponds directly to the speed of the air (along with the pressure and temperature). A Karman MAF has several pipes passing through it, and one of those pipes contains an ultrasonic sensor which measures the frequency of the ongoing vortices. The other pipes are simply bypass pipes of known area, such that the ratio of the air volume going through the main section to the total air volume passing through the MAF is known. The MAF also takes pressure and temperature readings, and sends all three (Hz, Temp, Press) to the ECU. The ECU then does several calculations, and comes up with the following information:

Velocity: The velocity can be calculated, using known Karman vortex formulae.

Area: The total cross-sectional area of the MAF is known, as well as the ratio of unmetered area to metered area. With these two facts in mind, the ECU can find the volume flow (speed times cross-section area equals volume flow).

Temperature, Pressure: With the temperature and pressure inputs, the ECU can calculate the mass flow of the air, since it knows the volume already, using basic laws of physics ($PV=nRT$).

The Karman MAF is fairly accurate, but it has a few shortcomings. First of all, because of the separate passages and screening, it can be a bit restrictive, although not nearly to the extent of a flapper-door MAF. Second, there is a limit where the airflow through the MAF becomes turbulent, and at that point the ultrasonic sensor loses its capability to discern between Karman vortices and turbulence, and metering becomes erratic enough to make the vehicle almost un-drivable.

Hotwire MAF - The principles behind a hotwire MAF are very simple, and also very consistent, which makes it one of the better ways to count air. Basically, there is a single wire that passes through the middle of the MAF, which is nothing more than a pipe with an airfoil in the middle of it to support the wire. That wire is heated to a certain temperature, which is verified by some thermoresistor (temperature sensitive resistors) attached to auxiliary wires. An analog circuit measures how much current is required to sustain the desired temperature, and as the number of air molecules coming through the meter increases, so does heat transfer, and the necessary current to keep the wire warm increases. This change is directly proportional to the number of molecules passing through the MAF, or the mass-flow of the intake charge. Because of the way the meter

works, there is no need for secondary pressure or temperature sensors, since pressure and temperature changes are compensated for by the primary measurement system.

Hotwire MAF's are very free-flowing, allowing more air with less restriction than anything besides speed density systems. They automatically compensate for ANY and all changes to the vehicle and any changes to the air coming into the car.

Speed Density - Speed density, also known (incorrectly, perhaps) as "MAP" uses the state of the air entering the motor to figure out how much of it there is.

Using principles similar to those used in the Karman Hz MAF, if you know a bunch of information about the air going into the cylinders, you can figure out how much of it you have. You need the following information:

Pressure - Pressure, measured by the MAP (manifold absolute pressure) sensor, shows what the pressure of the air is within the intake manifold, which is the closest you can come to representing the pressure in the cylinder without measuring it directly (which opens another can of worms, I won't go there). Pressure is one of the key components to knowing the air mass, and it's also a fairly good indicator of load. It is not, however, the only information you need.

Temperature - Temperature, like pressure, is measured in the intake manifold (or right before it). You need to know the temperature of air, as its density changes as the temp changes.

Volume - Volume measurement is one of the most difficult parts of a speed-density system. While the cylinders have a constant volume (and that is where we want to find the mass), the volume of the air entering the cylinders is not a constant. This phenomenon is described by the term "volumetric efficiency" (or "VE"). What happens is, as the piston speeds through its motion, the volume of the chamber may not necessarily be filled totally by fresh charge air. This can be because of exhaust gas staying in the chamber, the air losing density as it flows past the valves, or other things.

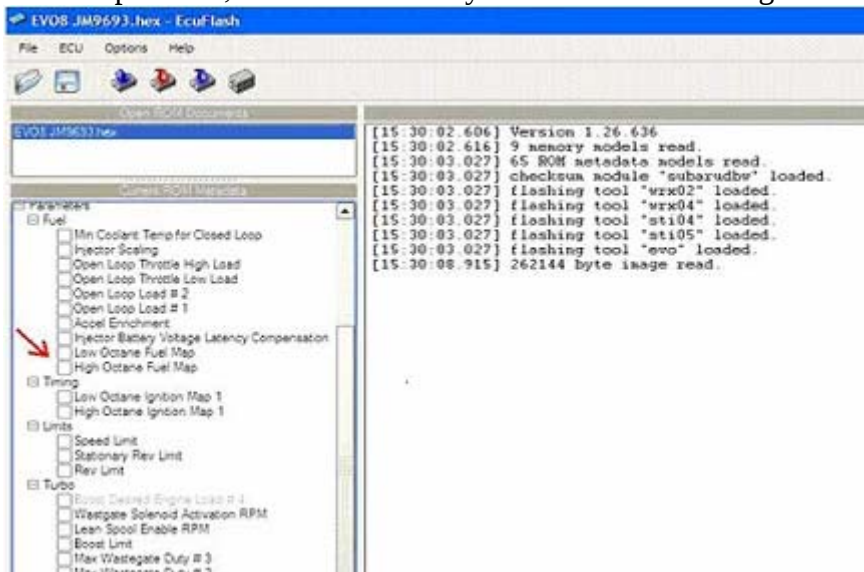
What you have to do, in order to run a speed density system, is you need to create a table of VE compared to engine load (MAP) and engine speed (RPM). Then, the ECU can look up the value for wherever it is at any given time.

Engine Speed - This is pretty simple. If the engine is moving faster, each volume (each cylinder) is filling more per time unit. That means more airflow (for a constant VE).

With this information, the ECU runs some calculations, and you get air mass.

Changing Data Values –

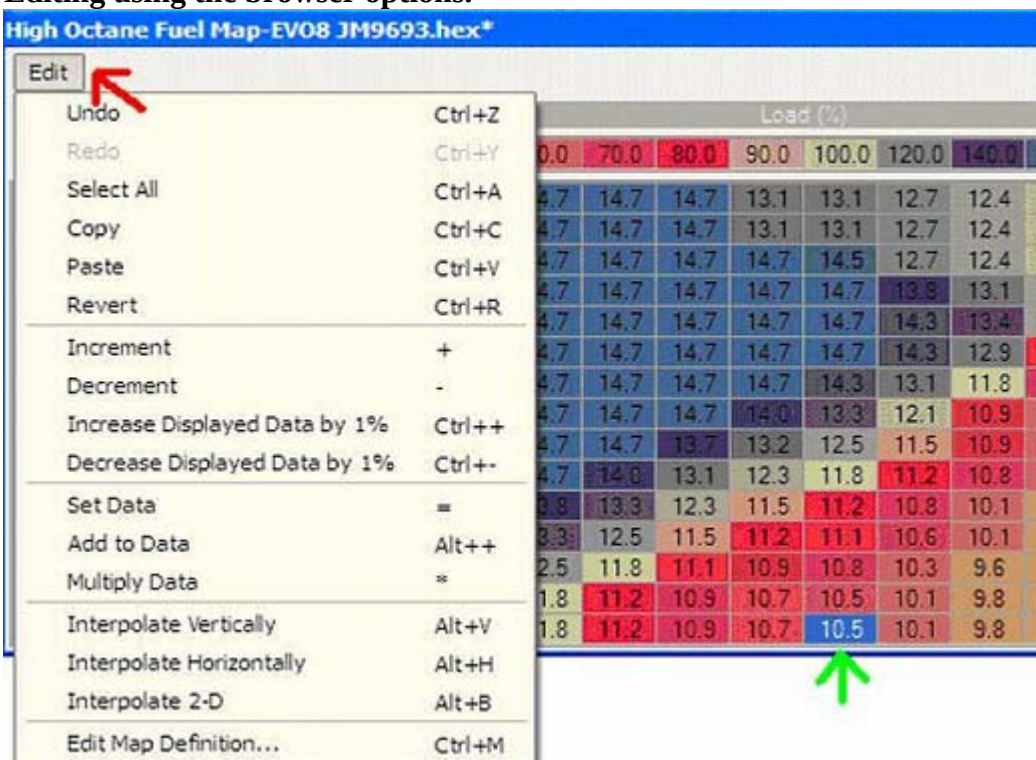
With map loaded, select the feature you would like to change.

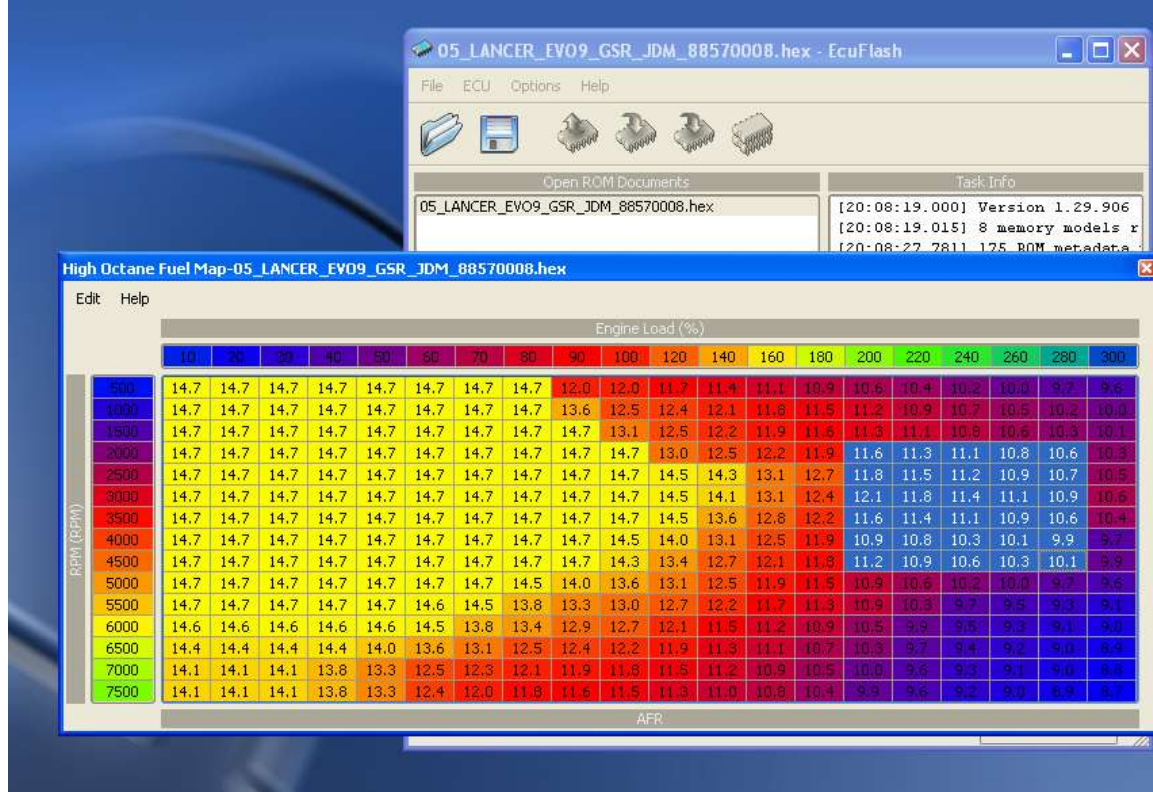


Then X or “check” the particular box. Using the *Edit* in *ECU Flash* browser options, you may change the cells one at a time by selecting the particular cell and using the browser option to manipulate data.

More conveniently, you may also highlight a group of cells you would like to change by left-clicking in one of the corner-most cells and holding left button while dragging to the opposing corner, then releasing then using (+) or (–) on the keyboard to manipulate the data in the shaded area.

Editing using the browser options.





Highlighting and manipulating using the + & - symbols.

!KEEP IN MIND DURING TUNING!

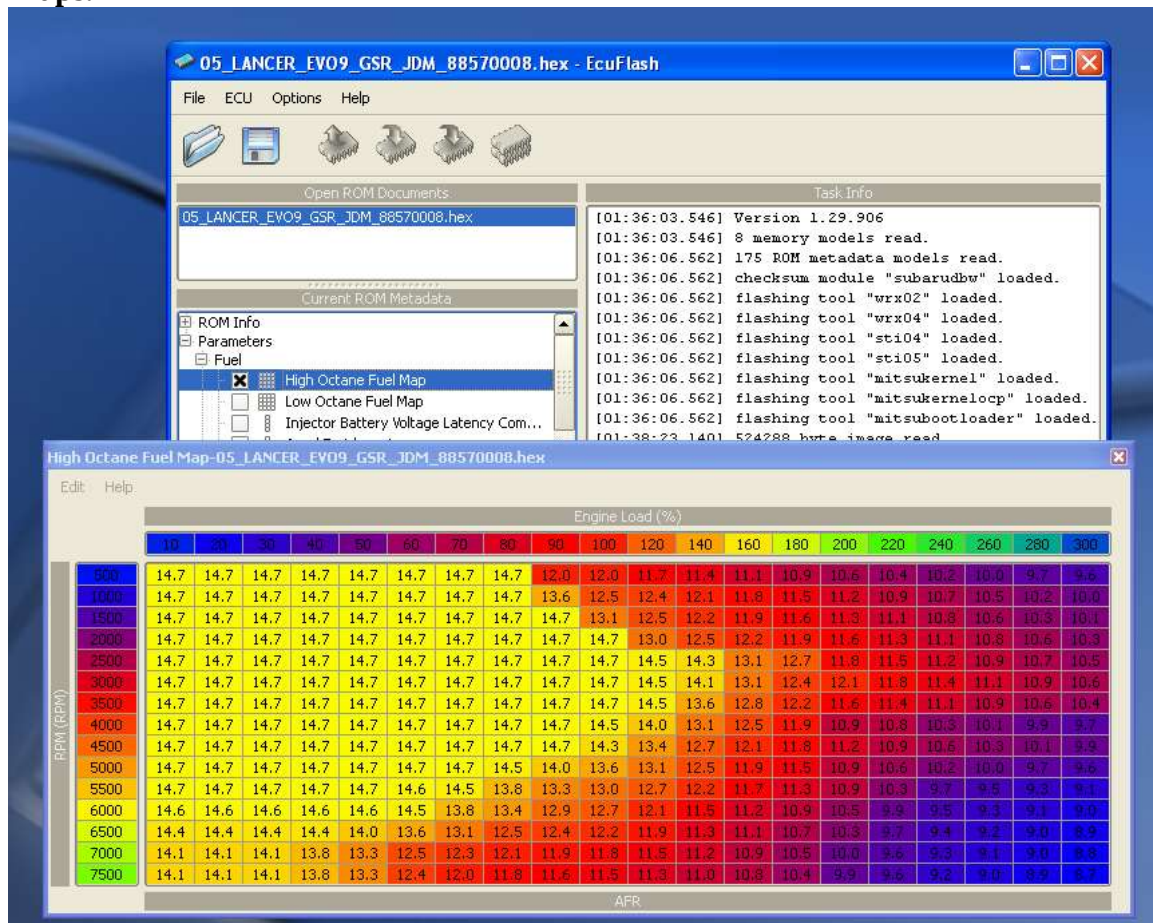
All tuning is relative to the parts! If you are tuning a turbo or high output engine, some calculations of changes will need to be made or very particular actions should be taken **simultaneously** to correct certain auto ailments. There are rare occasions that you will tune a car by adjusting JUST the Fuel map or JUST an Ignition tweak. A proper balance is needed so that all the maps follow their functions and that one function is not doing all the duty, or is cancelled out by another changed setting. Do not rely on just maps you feel comfortable with and neglect equal if not more important adjustments because they are foreign to you, especially when compensating for a Fuel Injector/Pump/Regulator upgrade.

Write to ECU – Will flash selected ROM image to ECU.

Test Write to ECU – Will do a general ECU check on the Flash ROM area and check that the ROM image that is anticipated to be flashed has all of the *ECU specific* data. This does not mean that a botched manipulation can not be flashed and is a rather general connection and hardware specific check.

Compare to ECU – Will do a comparison of the ROM image selected to the vehicle the *Tactrix* cable is connected to.

High and Low Octane Fuel Maps.



The **High/Low Octane Fuel Maps** are tables used to control fuel distribution by trying to keep fuel consumption at a minimal (or maximum depending on your driving style). On one hand the vehicle is trying to maintain a stoichiometric *AFR* (air to fuel ratio of 14.7 also referred to as *lean*) fuel burn or to have complete fuel consumption for better emissions. But on the other hand a vehicle has the ability to use extra fuel to achieve maximum reliability of its performance nature by cooling the combustion chamber, lowering ignition pre-fire/knock, and helping overall by avoiding detonation, especially turbocharged vehicles (also considered *rich* or *enriched*). If knock is detected, the car might reference (depending on ECU tolerances) the Low Octane Fuel/Ignition Maps lowering engine output.

There are 2 points of reference or constants that can be accurately monitored to figure *AFR*; they are **Load** and **RPM** (revolutions per minute).

Load is measured by the amount of air that is passing through the Mass Air Flow sensor (MAF) of the vehicle. One might think that it would be more advantageous to use throttle position, but this does not take in to consideration the environmental conditions like air pressure, temperature, or moisture. Plus just because the throttle is at 100% does not mean there is 100% engine duty. Revs are a gradual build and need a graduated reference to keep the most accurate *AFR*.

RPM is measured by the crankshaft position sensor (CKP) using a magnetic pick-up.

The actual “map” is referred to as cells. The cells contain the measurement of air in the *AFR* (air/fuel ratio). This makes sense because the way the MAF measures load, to give an accurate outcome from the 2 other variables (*Load* and *RPM*).

The **AFR** or *Air/fuel ratio*, of an engine is determined as the weight ratio of the air entering the engine in relation to the amount of fuel being mixed with the air that creates a combustible mixture. The stoichiometric AFR is 14.7 to 1, (14.7 grams of oxygen to 1 gram of fuel). Stoichiometric means *that a ratio of 14.7 grams of oxygen to 1 gram of fuel, when burned, will theoretically result in complete combustion*. Stoichiometric isn't the only AFR that supports combustion. Most engines can "run" (I did not say if it would run horribly or not) with *rich* AFR's of about 8 to 1, (more fuel) up to *lean* AFR's of about 15 to 1, (less fuel).

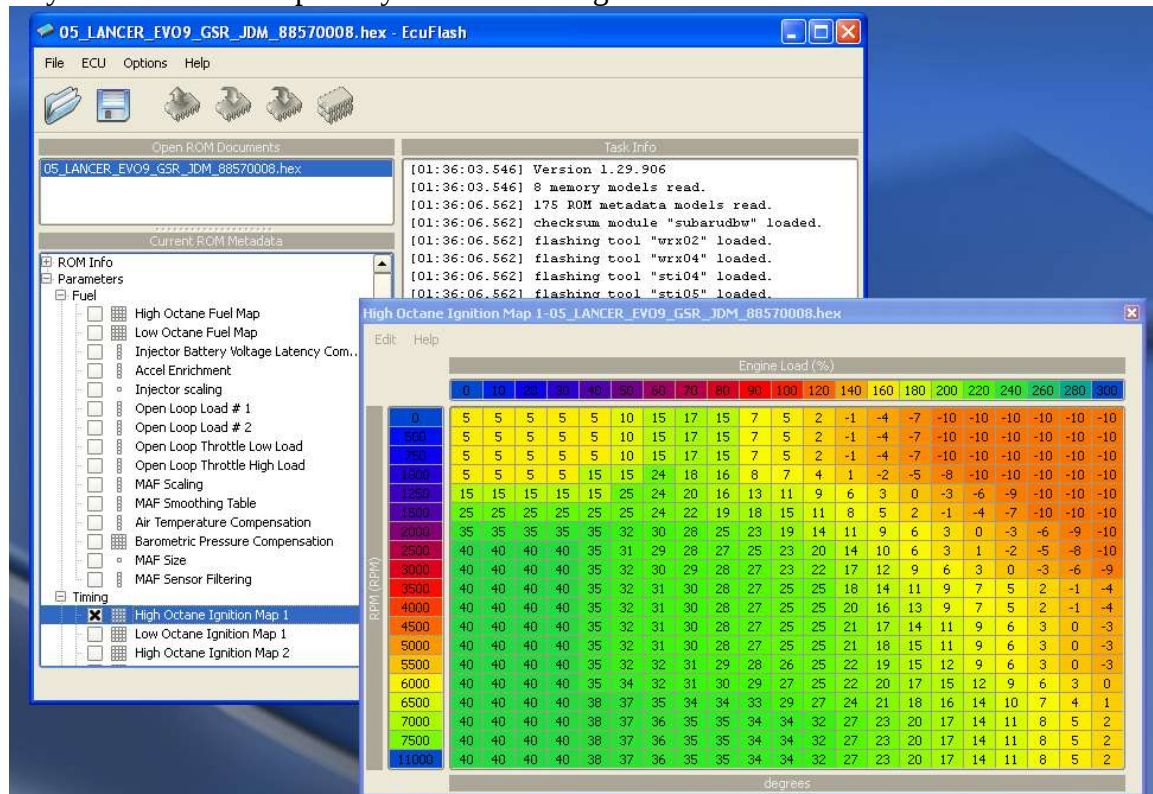
The **High Octane Fuel Map** is the map is the most likely to be changed while tuning. The computer references the *High Octane Fuel Map* under open loop (no o2 sensor feedback) conditions, and "references" the table during closed loop. While in Closed Loop, the vehicle is looking for the best emissions friendly AFR and will wander off the High Octane Fuel mapping (and will be discussed later in *Open/Closed Loop Systems*)

NOTE! I have the 4G64 and plan to turbo-charge using JUST Open ECU and my factory ECU for engine management. A major issue for a 4G64T is the lean condition of the factory mapping in the higher RPM and Load scales. Just a quick reference to any EVO will show what differences there are between AFR maps of a turbo-charged vehicle and an NA vehicle. BEWARE, just don't copy over an EVO map, there is a lot more involved in a 4G64T tune.

PM me through TGC, Open ECU, or OS ECU ROMs sites if you want more info on tuning a 4G64T.

AGAIN – To increase fuel (or enrich), lower the value.
To decrease fuel (or lean), increase the value.

The **Low Octane Fuel Map** is the same table as the *High Octane*, although it has a richer fuel table to compensate for knock. Avoiding Low Octane Fuels is probably the easiest way to avoid knock especially in a turbo-charged vehicles.



High Octane Ignition Map

The **High/Low Octane Ignition Map** is measured in a similar way as AFR is, although the table or cells have different meanings. The cells represent the state of advance or retard in the ignition form TDC (Top Dead Center or 0 degrees). The **Load** and **RPM** are also given extrapolated figures for both load and RPM. Again, load and RPM are constants while the cells represent the actual ignition advance or retard.

The **High Octane Ignition Map** is the map most likely to be changed while tuning. The computer references the *High Octane Ignition Map* under open loop (no O2 sensor feedback) conditions. While in Closed Loop, the vehicle is looking for the best emissions friendly *timing advancement* and will wander off the High Octane Ignition mapping (and will be discussed later in *Open/Closed Loop Systems*).

Low Octane Ignition Map is sought to for a less advanced profile to avoid presents of knock (octane and mechanical failure enacted). The figures in the cells are measured in + (advance) or – (retard) Degrees TDC.

Barometric Pressure Compensation Map.

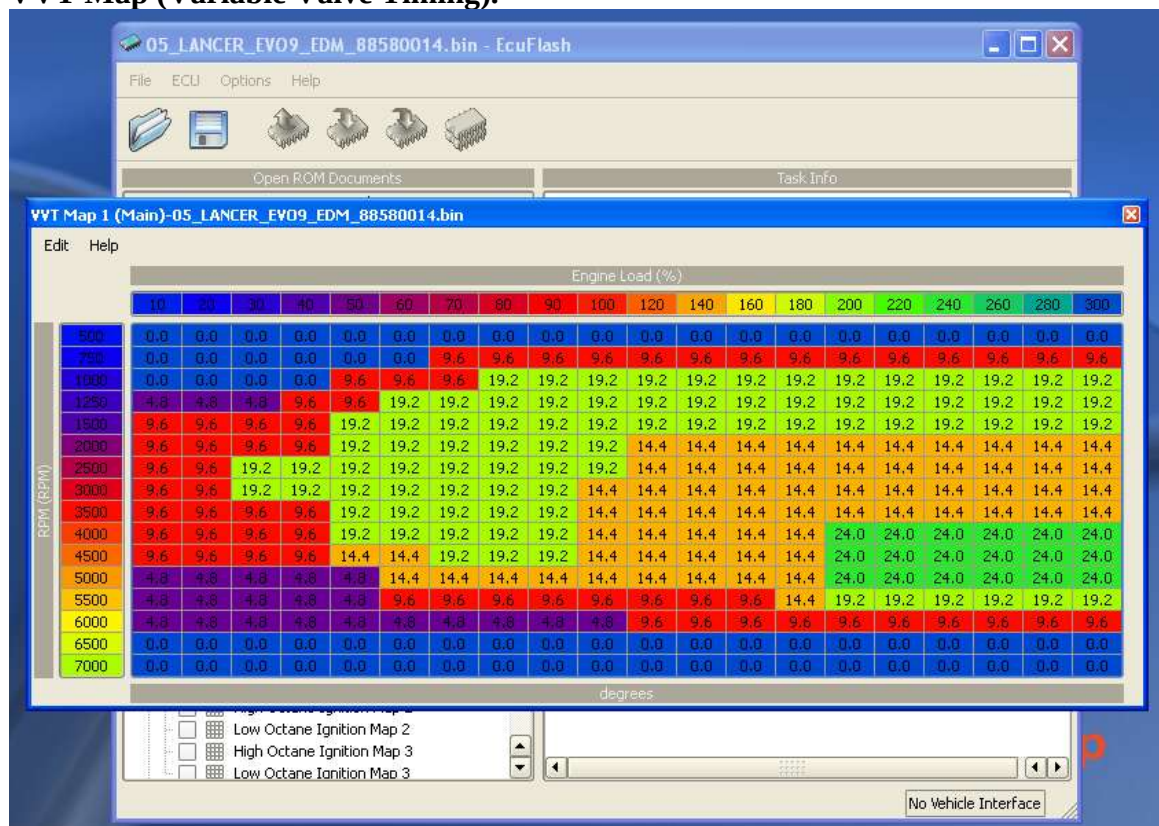
The screenshot shows the EcuFlash software interface. The main window displays the 'Barometric Pressure Compensation' map. The map is a table with RPM on the y-axis (0 to 8000) and Atmosphere Pressure on the x-axis (21.2 to 51.2). The map is currently set to 'bar' units. The software interface also shows a list of parameters on the left and a task log on the right.

	21.2	31.2	41.2	51.2
0	1.39	1.37	1.36	1.34
1000	1.40	1.37	1.36	1.34
2000	1.41	1.37	1.33	1.32
3000	1.40	1.37	1.32	1.31
4000	1.42	1.37	1.33	1.32
5000	1.42	1.37	1.33	1.32
6000	1.41	1.37	1.32	1.31
7000	1.40	1.37	1.32	1.30
8000	1.37	1.37	1.37	1.37

The **Barometric Pressure Compensation** map is used to keep the cars fuel AFR is properly maintained by measuring air pressure and density. Adjusting this map has no keen advantage over performance, and is usually adjusted to geographical location based on elevation (USDM/JDM/EDM).

[illegible]

VVT Map (Variable Valve Timing).



The **VVT (Variable Valve Timing) Map** is used to adjust the duration in degrees, that the intake valves are left in the “open” position to allow more air to enter the compression chamber at higher RPMs. This map can help maximize turbo output by allowing more air to enter the cylinder while a non-VVT would be stuck with just a single cam profile, and “limited” in potential (although most EVO/WRX’s have pushed the limits of my adrenalin, and the “limited” is used loosely in the world of these vehicles).

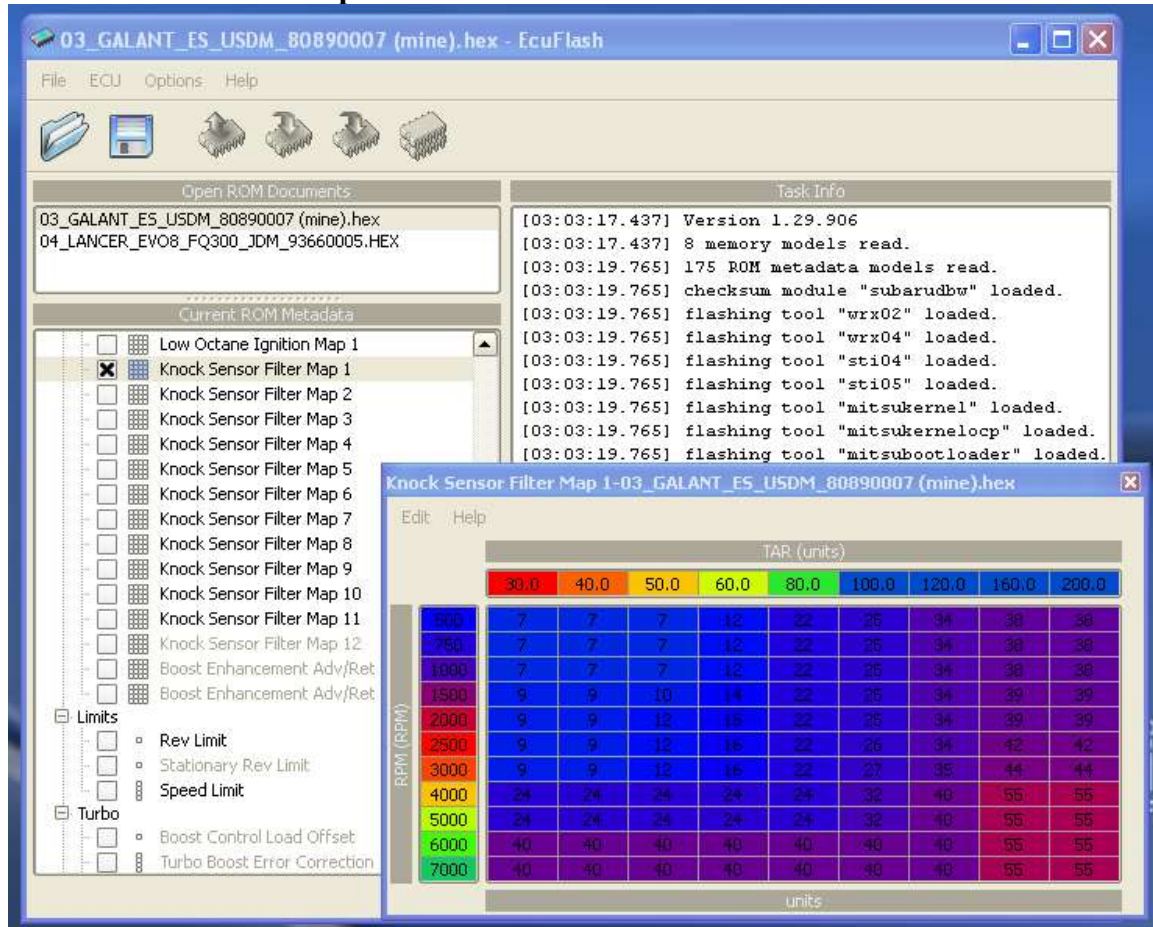
Idle Stepper Look-up Table.

The screenshot shows the EcuFlash software interface for a Lancer Evo8. The main window displays the 'Idle Stepper Look-up Table' for the file '04_LANCER_EVO8_FQ300_JDM_93660005.HEX'. The table is a grid where the vertical axis represents RPM (0 to 1010) and the horizontal axis represents Throttle Position (8, 32, 47, 53, 60, 67, 74, 81). The values in the table represent the desired idle speed in RPM for each combination of RPM and Throttle Position. The table is color-coded, with blue for low RPM and throttle, transitioning through green and yellow to red for high RPM and throttle.

		Throttle Position (units)							
		8	32	47	53	60	67	74	81
0	0	0	0	0	0	0	0	0	0
39	18	18	18	18	18	18	18	18	18
78	20	20	20	20	20	20	20	20	20
117	30	30	30	30	30	30	30	30	30
156	40	40	40	40	40	40	40	40	40
195	51	51	51	51	51	51	51	51	51
234	58	58	58	58	58	58	58	58	58
273	62	62	62	62	62	62	62	62	62
312	65	65	65	65	65	65	65	65	65
352	67	68	68	68	68	68	68	68	69
391	70	70	70	70	70	70	71	72	72
430	72	72	73	73	73	73	74	76	76
469	74	75	75	75	75	76	77	79	79
508	76	77	77	77	77	78	79	82	82
547	78	79	79	79	79	81	85	89	89
586	81	81	81	82	83	86	92	104	104
625	83	83	84	85	86	90	99	140	140
664	85	86	87	88	90	95	115	160	160
703	87	88	90	91	93	102	140	160	160
742	89	90	92	94	97	111	160	160	160
781	92	93	95	97	101	133	160	160	160
820	94	95	98	103	111	146	160	160	160
859	96	98	101	110	120	160	160	160	160
898	98	100	108	120	120	160	160	160	160
938	100	105	115	133	140	160	160	160	160
977	104	110	120	160	160	160	160	160	160
1016	136	130	160	160	160	160	160	160	160

The **Idle Stepper Look-up Table** is used to find a reasonable idle speed. This is the **base** table that is used in calculating final idle speed. The **Idle Stepper Look-up Table** is the starting table of idle and when used in conjunction with other variables, the ECU can find a calculated sum and judge what the final Idle RPM should be.

Knock Sensor Filter Map.



Knock sensor filter map has **RPM** (engine speed) and **TAR** as constants (no idea what the acronym TAR stands for).

TAR (Short of understanding its acronym) is a theoretical load corresponding to engine speed and sound intensity/frequency. This is filtering out ordinary combustion (timed explosions), to the presence of detonation/pre-ignition (Knock).

RPM is measured by the crankshaft position sensor (CKP) using a magnetic pick-up.

Knock is a phenomenon that occurs when high temperature and pressure causes the end gas to self ignite (end gas is the final AFR of O₂ + Fuel atomized in a homogenous gas in the cylinder). This causes a very high local pressure and the propagation of pressure waves across the combustion chamber. These pressure waves excite the resonance modes of the cylinder (this is particular to the metallurgy of the cylinder and amounts to an abstract forms of friction and heat). The frequency of the oscillations under knocking conditions depends on engine geometry, and is often in the range of 5 to 10 kHz.

There are several theories about what it is that causes the damage on the engine during knocking conditions. The most accepted is that it is caused by heat transfer. When knocking conditions occur, the piston and the walls of the combustion chamber are exposed to a great deal of additional heat which results in overheating of these parts (piston face, cylinder walls, and the heads valve surface). As a result, the thermal boundary layer at the combustion chamber wall can be destroyed. This causes increased heat transfer which might lead to certain surfaces causing pre-ignition. Substantial knock

can damage the engine and is stressful to the driver and is therefore the most important limitation for any engine. In order to control the knock it is sometimes necessary to regulate away from the most efficient operating point.

The most common way to control knock is to delay the ignition and thus prevent overheating. When changing the ignition angle it must be taken into consideration that this affects not only the knock but also the output torque.

Knock is different for most models. In sum it is a value derived from many different aspects of engine dynamics. Crank angle, precise piston displacement, and volumetric efficiency are the major variables that really bring the final "knock equation" close to home. Most testing is done in laboratory setting, while they run every conceivable (and inconceivable) variable like fuel, ignition, and valve manipulation.

The *knock sensor* system in your car is an advanced way to protect the engine from a possible catastrophic event. The first time I personally ran into knock is when I spun a bearing on a b18 Honda engine, but still left it in a "more reparable" condition than running it home (or garage) on a normal ignition table.

To extrapolate this theory from V-8 tuning, specifically with port polishing and "doming" the valve side of the head (an excellent way of dealing with knock for any vehicle). We always were dealing with removing the causes of knock because there was no knock retard on modified stock cars unless you bought one aftermarket. Therefore, we had to come up with extreme (but rather fundamental) measures to rid us of knock; the most logical is lower the compression.

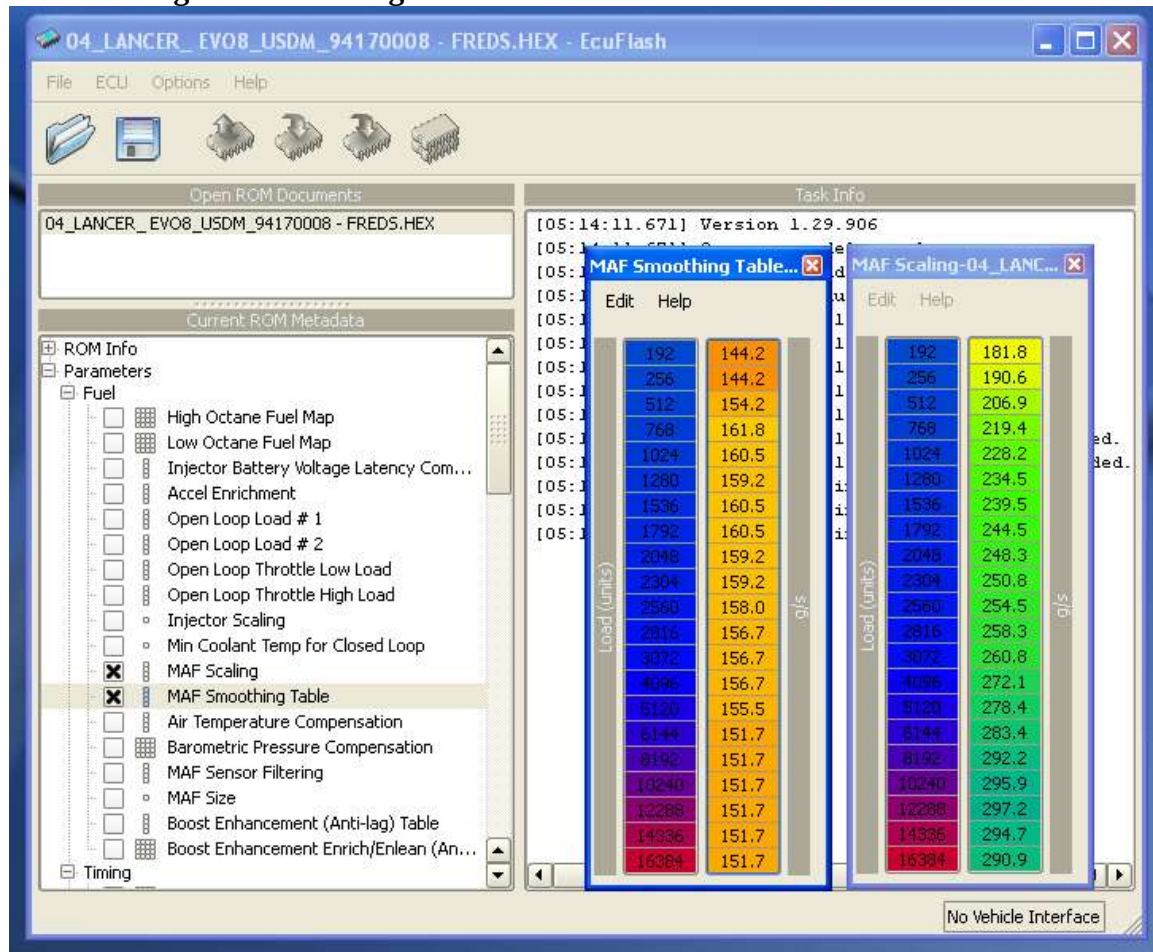
Make sure you are running good quality 93+ Octane fuel before blaming the motor or tuning on any sign on knock activity.

BUT! I would not change your knock sensor filter, unless you are changing the engine to a different displacement, shorter or longer rods, or aftermarket crankshaft that will most likely be close to stock where I would grab the stock knock map of the vehicle it (particularly just the crank) was engineered for.

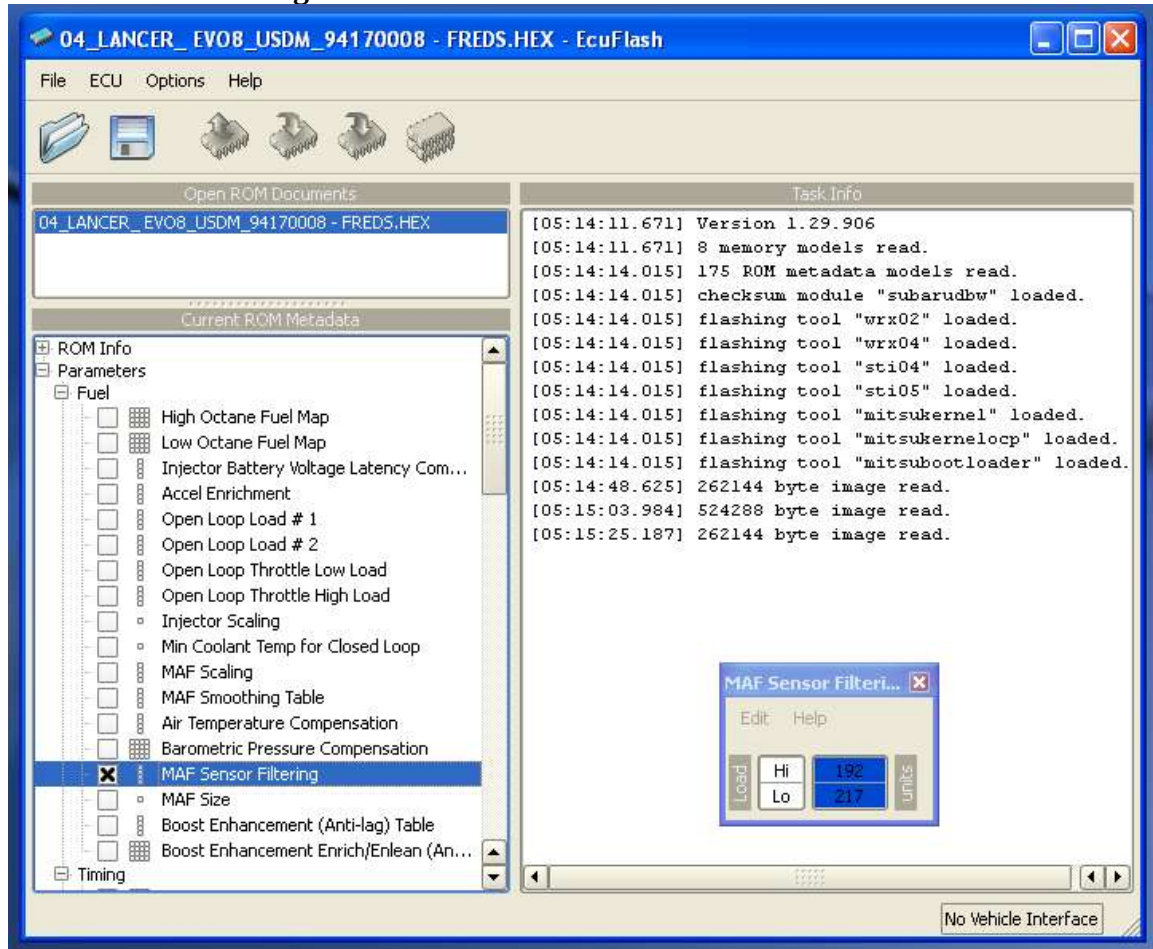
This may seem drawn out, **but** a fundamental knowledge can help you understand your vehicle in ways, so that it may offer up many years (or race seasons) of fun use.

2D Maps –

MAF Scaling and Smoothing tables.



MAF Sensor Filtering.



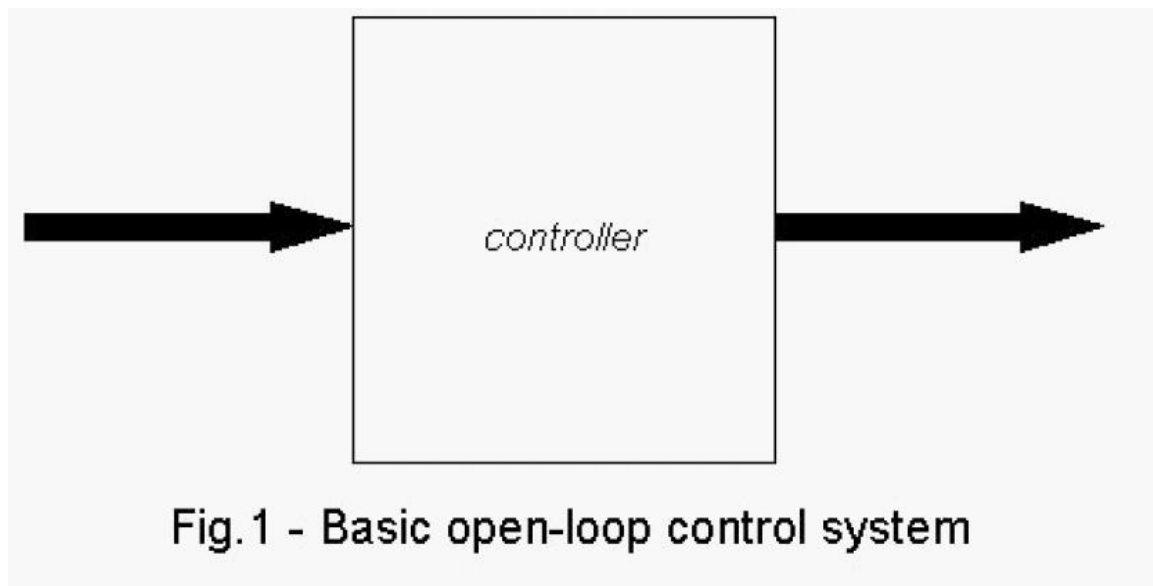
The **MAF Sensor Filtering** table is 2 MAF figures corresponding to a high and low load value. *****

Open/Closed Loop Theory.

All control systems, including the ECUs, can operation in one of two ways: **Open-Loop** mode and **Closed-Loop** mode. These are the terms most frequently misunderstood, so let's state what they mean.

Open-loop Systems -

Open-loop systems apply control signals without regard to the output signals being produced. To model one, take a piece of paper and draw a box on it. On the left side, draw a horizontal arrow leading into the box - this is the input. On the right side, draw a horizontal arrow leading out of the box - this is the output. Voila, an open-loop control system!



The box is the *controller* or *control system*; the part that actually does the “thinking”, if you will. The controller can be anything - an electronic circuit, a mechanical system, a computer, whatever. It takes *input signals* (the leftmost arrow), processes them and generates output signals directing the action. To get a little fancier, you can draw another box on the other end of both arrows to make 3 boxes. The leftmost box represents the *input system*, which provides the original input signals to the controller. The rightmost box is the *controlled system* - in other words, the part that the control system is running.

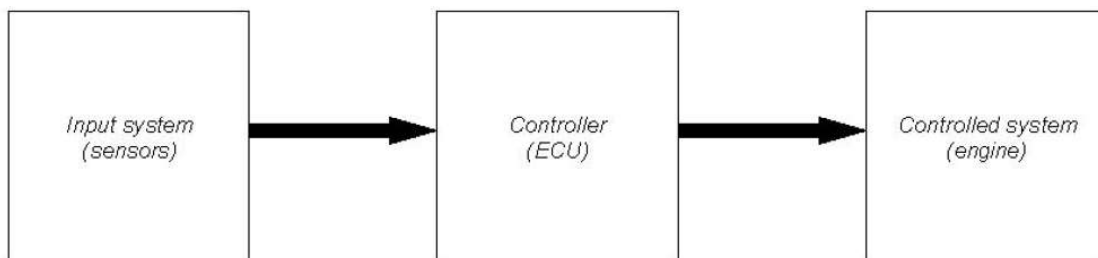


Fig. 2 - Basic DSM open-loop system

If you like, you can imagine the middle box on the page representing the ECU inside a Mitsubishi, or any other car. The input system can be any or all of the input sensors used

by the ECU, including airflow, throttle position, air temperature, barometric pressure, etc. The output signals consist of control outputs to fire the injectors, provide spark to the cylinders, and so on, which (naturally) run to the engine. As you might expect, *open-loop* control systems are generally not very useful, because the controller does not have any way of knowing how the controlled system is behaving.

As a simple example, let's make the input a temperature sensor, and the output a set of heating elements. For any given temperature, the heater controller 'knows' to make a certain amount of heat, and it will direct the heating elements accordingly.

The system will work fine when it is put together and probably for some time thereafter. Let's just say, however, that something goes a little bit wrong with the heating elements - one breaks, or they just wear out a bit, so the heater is now putting out less heat than it should.

In this example, the air temperature is the same, so the inputs are unchanged. The control system is still producing the correct signals to run the heater at that air temperature. The heater could have lots of extra heating capacity left, which could easily compensate for the temperature drop. The problem is that the heater is not behaving like it should, and the control system has no way of knowing this.

There are some examples where open-loop control is fine. If all that is being controlled is an indicator light, or an LCD display, maybe it doesn't matter if they work because a person will notice the fault, and correct it. (Systems that are easily predictable, in other words). This isn't always the case, however. Where open-loop control fails us, the solution to this is the *closed-loop* system.

Closed Loop System -

Closed-loop systems add an extra element to allow the controller to detect changes in the controlled system, and compensate accordingly. To show this capability, you need to draw one more arrow on your diagram. This arrow begins at the top of the existing right-hand box (the 'engine'), and runs to the top of the existing middle box (the 'ECU').

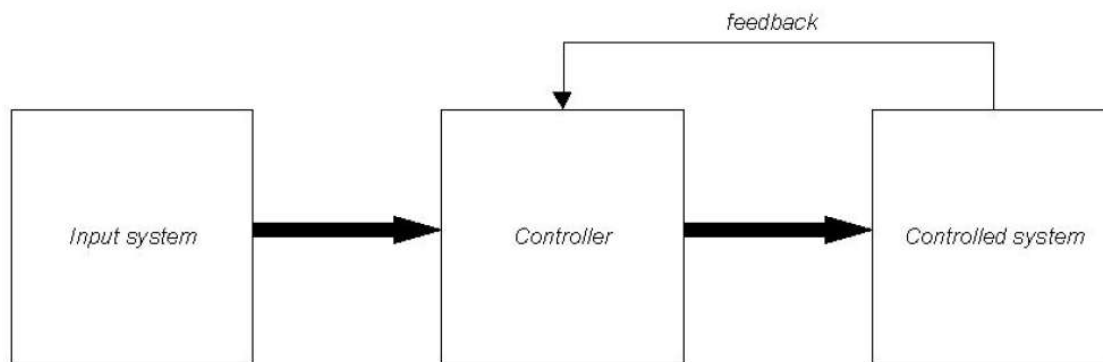


Fig. 3 - Basic closed-loop system

This extra arrow represents a quantity known as **feedback**. This means just what it says: some part of the operation of the controlled system is measured and 'fed back' to the controlling system. Feedback represents very useful information, because now the control system can apply a signal to the controlled system *and measure the results of that signal*. Going back to the simple heater example, we already have an input system (the temperature sensor), a controller, and a controlled system (the heater).

To introduce feedback into the system, we could add a temperature sensor inside the heater itself, capable of measuring the actual heat output of the unit. Prior to this, the control system would measure the air temperature and direct the heater to generate a certain output. With the addition of the extra heater sensor, the control system can actually check to see if the heater is producing the correct amount of heat, and react accordingly.

So, as the heater wears and output drops, the controller will 'notice' the drop in output and can direct the heater to produce more heat to compensate. The heating system will continue to work correctly until the heater breaks, or becomes so worn that it can't compensate for the output losses any more. Once this happens, though, a 'smart' controller can always sound an alarm or send an error message of some kind. So, closed-loop operation allows the controller to compensate for external variations that may affect the operation of the controlled system. Since very few parts in the real world are absolutely precise, closed-loop operation is a necessity in virtually all control systems.

Relating the closed-loop system to the ECUs, the newly-added feedback arrow represents the output of various sensors which are placed in the engine to monitor its operation. The primary sensor of concern is the oxygen sensor, a device which measured oxygen in the exhaust gases produced from the engine. Others include the RPM sensor and crankshaft angle sensor, which are also crucial elements in engine control.

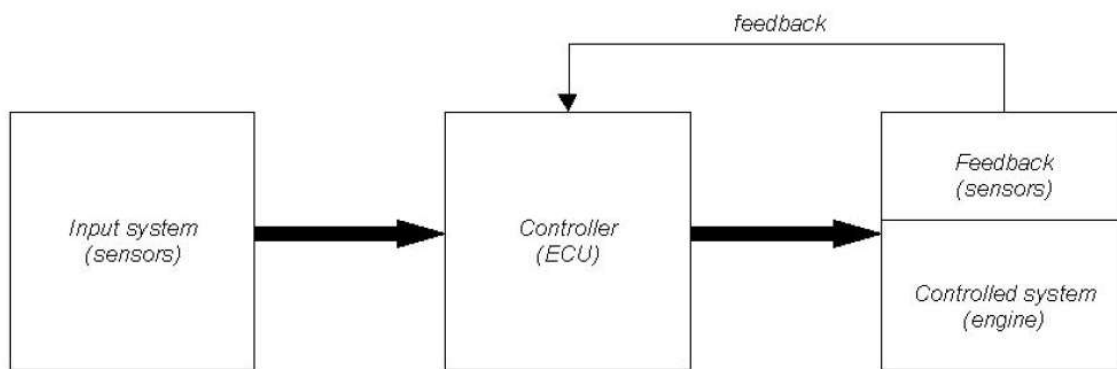


Fig. 4 - DSM closed-loop system

When people talk about open or closed-loop operation on ECUs, they are invariably talking about the specific function of fuel delivery. The ECU has other functions as well, but the most prominent task is the requirement to inject the correct amount of fuel into the engine, based on the amount of air entering the cylinders. Too little fuel and the car won't run - too much and the car will not burn it all, leading to waste and poor fuel economy.

The ECU has several input sensors that help it determine the correct amount of fuel. The most widely recognized is the *mass air flow sensor (MAS or MAF)*. The MAS consists of the air volume sensor, the intake air temperature sensor, and the barometric pressure sensor. These three sensors form a single component located directly behind the air filter, and it provides the ECU with the information required to calculate air mass. [Why mass? Well, the ideal ratio of air to fuel depends on the number of molecules of each type of substance. Knowing the volume doesn't get you anywhere, since the number of air molecules in a given volume changes with temperature and pressure. If you know air volume, temperature and pressure, though, you can get air mass, which in turn can be used to determine the correct amount of fuel.]

So the input signals consist of the air volume, temperature and pressure from the MAS cluster. Feedback is provided by the oxygen sensor, located in the cars downpipe. Why oxygen? Measuring the unburned oxygen in the exhaust is a practical way of figuring out

how much fuel was really burned in the engine. Very little oxygen in the exhaust indicates that too much fuel was added to the air/fuel mixture. This is known as a *rich* condition. On the other hand, lots of oxygen in the exhaust indicates that there is too much air in the mixture, a condition known as a *lean* mixture. If the mixture is too rich, the engine is eating more fuel than it can use, which is a waste of gasoline. Also, unburned gas and soot from the incomplete burn will be expelled from the engine, coating exhaust parts and leading to increased air pollution. When the mixture is too lean, though, the mix will burn much hotter than normal, leading to the possibility of literally melting engine components. Bad bad bad.

With the MAS providing the input and the oxygen sensor the feedback, our fuel control system can run in closed-loop mode. Note I said “can”, not “will”. Just because the feedback sensor is present doesn't mean the ECU has to use it. More on this later.

Before we can continue a little background on the oxygen sensor is needed. The sensor output is inversely proportional to oxygen - the less oxygen, the higher the sensor output. Also, the sensor output does not increase steadily with decreasing oxygen. Instead, the sensor can be thought of as an 'oxygen thermostat', which tends to change a great deal at a certain oxygen level.

Not by coincidence, the oxygen sensor is designed for a large change in output right around the oxygen level that results from the ideal 14.7:1 air/fuel ratio. Add a little bit of air, and the signal goes way down, while a little bit more fuel makes the signal go way up. Outside this 'magic' range, the oxygen sensor is roughly linear, although it is difficult to predict the actual output for any given air/fuel ratio because of variations in the sensors themselves.

Under normal operating conditions, the engine is running at a near-constant speed with the throttle in a near-constant position. This can be at idle, or while cruising down the street or highway. Under these conditions, the engine is doing a steady amount of work, well within its power capabilities. The ECU, therefore, is programmed to maximize fuel economy and minimize emissions. To this end, it uses closed-loop operation to try to optimize the amount of fuel being delivered to each cylinder.

So what does this mean??

Well, the ideal air/fuel mixture is 14.7:1. From the MAS information, the ECU knows the mass of air entering each cylinder, and can determine the correct amount of fuel to deliver. It translates these into injector open and close times to control the fuel delivery. Good enough? Well, not really. Variations in injectors, fuel pressure, and the sensors themselves (all of them) can lead to significant inaccuracies in this system. So the injector open time that the ECU has figured out is really not much more than an educated guess. So what happens? Well, the ECU will fire the injectors, and then check the oxygen sensor output to see if the fuel delivered was too much, or too little. As described before, the oxygen sensor output changes sharply at the special 14.7:1 point. If the ECU is a little off in one direction or the other, the oxygen sensor feedback signal changes a lot, letting the ECU know that it missed the correct fuel amount. Knowing this information, the ECU can change the next set of injector times to be a little closer to ideal. If too much fuel was delivered, the times are shortened; too little, and the injectors are fired a bit longer. The ECU will check the oxygen sensor output again after this round, readjust the injector times, and repeat over and over again. In this manner, the ECU is continually adjusting the fuel quantity to try to get it just right, and hold that oxygen sensor signal at the magic point of 500 mV.

Owners of air/fuel gauges will know, however, that the ECU never gets it 'just right'. In fact, the oxygen sensor signal changes so much around that perfect 14.7:1 point that the

ECU cannot ever hit the 500 mV point and hold it steady. Instead, the ECU keeps hitting around that point, never quite getting it right.

In more detail, what happens is the ECU is a little off one way (too rich by a bit), and the oxygen sensor signal shoots up. The next round, the ECU trims the fuel down a little to fix it. This time, the ECU is off a little the other way (a tad too lean), and the oxygen sensor signal shoots down. The ECU trims the fuel up a bit in response, and keeps missing in this fashion all the time. This is what leads to the 'bouncy' oxygen sensor signal output that all A/F gauge users are familiar with. Far from being a sign of malfunction, this up-and-down signal behavior is perfectly normal and, in fact, is an excellent sign of a healthy oxygen sensor.

To sum up, closed-loop operation means that the ECU uses the MAS information to determine an initial amount of fuel to deliver. The oxygen sensor signal provides feedback to help correct this fuel amount for variations in the individual sensors and engine. This leads to better fuel economy and emissions, as there is very little wasted fuel. Closed-loop operation occurs when the engine is running at a near-constant speed and a near-constant throttle position.

Open Loop Load Tables 1 & 2.

The screenshot shows the EcuFlash software interface. The main window title is '04_LANCER_EVO8_USDM_94170008 - FRED5.HEX - EcuFlash'. The left sidebar has a tree view with 'ROM Info' and 'Parameters'. Under 'Parameters', 'Fuel' is expanded, and 'Open Loop Load # 1' and 'Open Loop Load # 2' are checked. The right pane shows 'Task Info' with a log of operations. Two smaller windows, 'Open Loop Load # 1-0...' and 'Open Loop Load # 2-0...', are overlaid, showing tables of RPM vs. Load (%).

RPM (RPM)	Load (%)
800	86.9
1000	86.9
1500	88.1
2000	88.1
2500	88.1
3000	88.1
3500	88.8
4000	85.6
4500	84.4
5000	84.4
5500	84.4
6000	71.2
6500	66.2
7000	61.9
7500	52.5

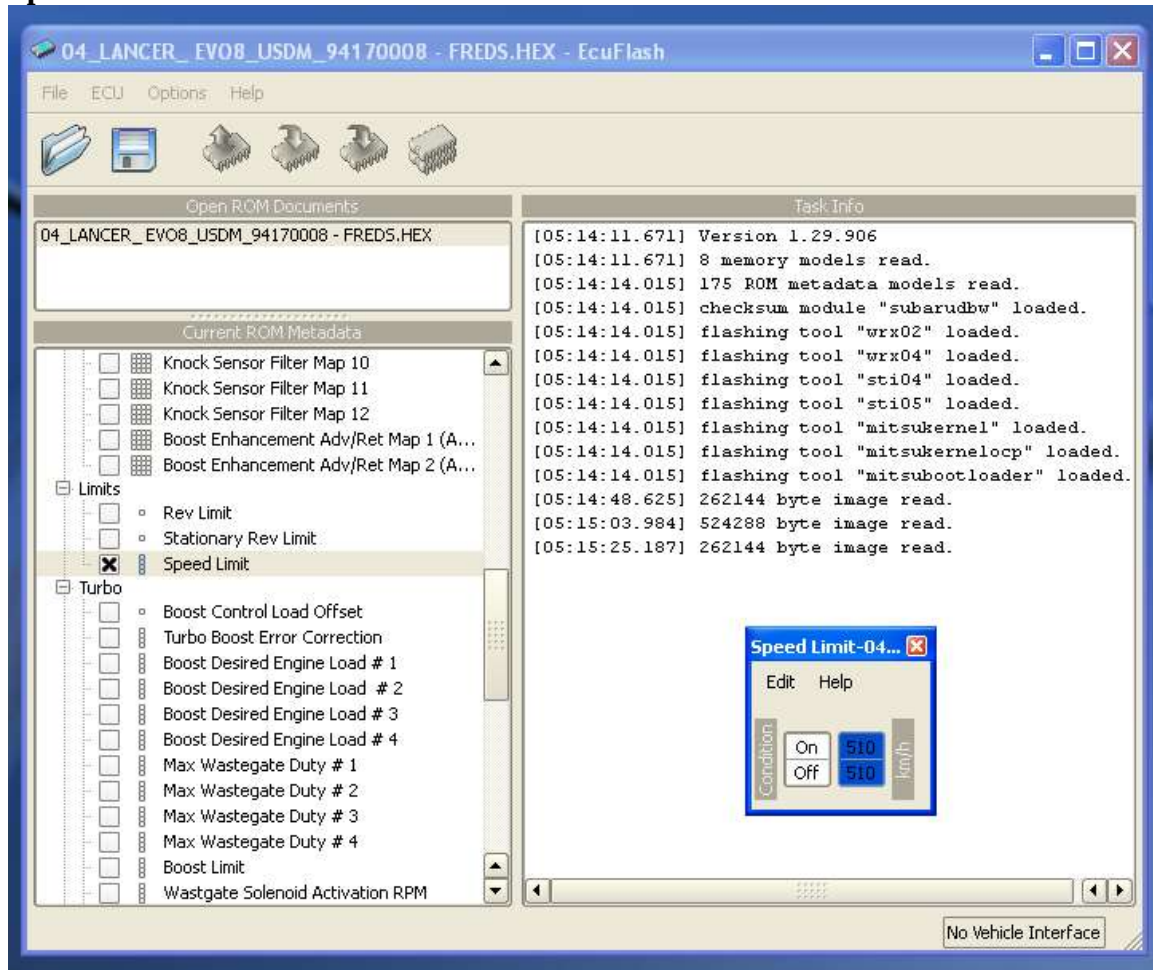
RPM (RPM)	Load (%)
800	86.9
1000	95.0
1500	95.0
2000	95.0
2500	100.0
3000	97.5
3500	95.0
4000	95.0
4500	84.4
5000	84.4
5500	84.4
6000	71.2
6500	66.2
7000	61.9
7500	52.5

The **Open Loop Load** Tables correspond to the amount of throttle angle needed to bypass or “disregard” the *After Catalyst* O2 (#1 Map) sensor then the *Pre-Catalyst* O2 sensor (#2 Map), usually located on the exhaust manifold. Reducing their feedback allows a richer AFR, therefore a little extra power on each step. But keep in mind your Fuel consumption is going to go up.

[illegible]

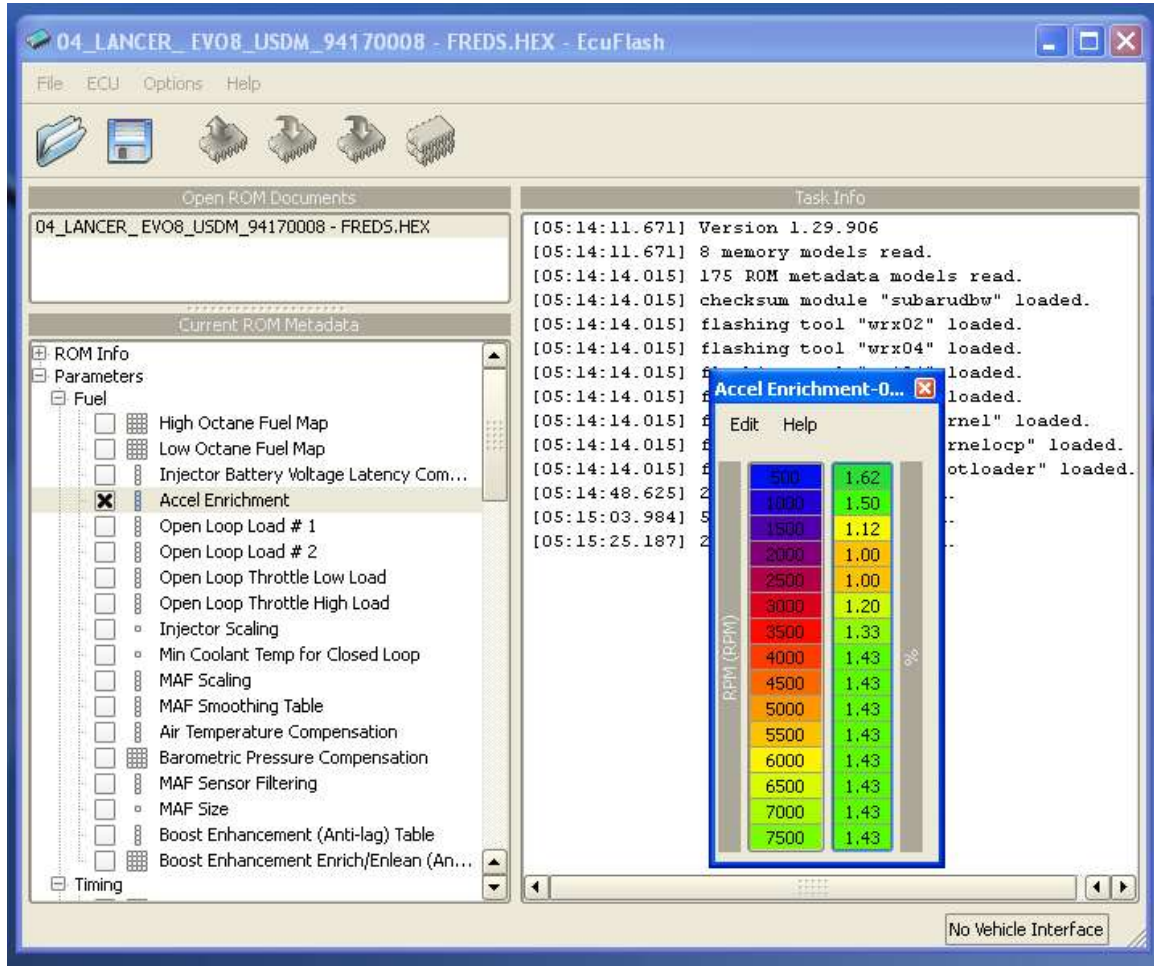
The ***Open Loop Throttle High/Low Load***, are relatively the same as above, and give more a second source for a more accurate feedback from the vehicle based on the drivers desired load. This gives the sense that there is less of a “controller” feedback from you wandering on the accelerator, like say 5:00 traffic (up and down). If the *Open Loop Load* Tables were changed, they might also need to be reflected in changes made to the *Open Loop Throttle Load* and vice-versa.

Speed Limiter.



The **Speed Limit** Table is as the name implies, although measured in kmh. The **On**, is when the **Speed Limiter** engages. The **Off**, is when the **Speed Limiter** disengages and will allow throttle response. If you wish to turn them off set them **BOTH** to 510.

Acceleration Enrichment Table.



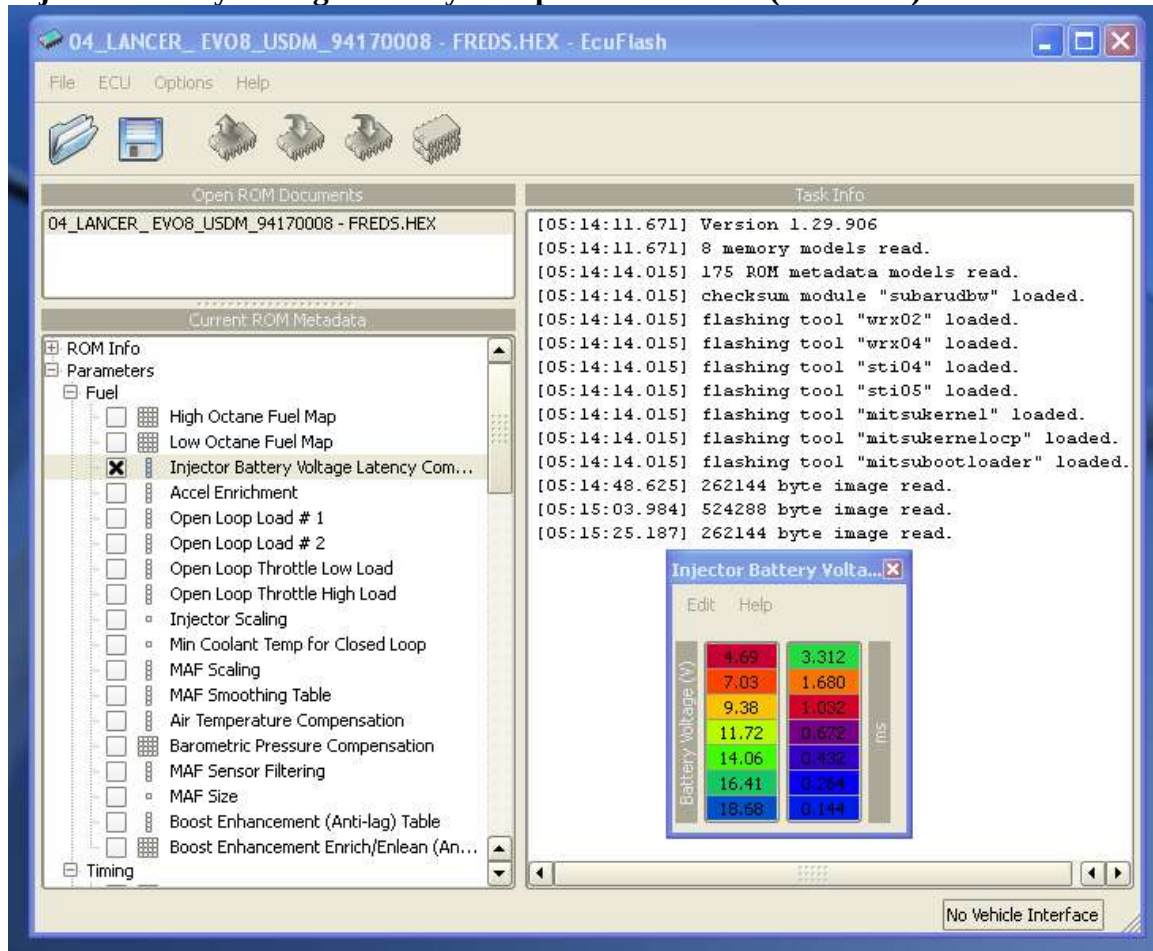
The screenshot shows the EcuFlash software interface. The main window displays the 'Current ROM Metadata' section with a tree view on the left. The 'Fuel' section is expanded, and 'Accel Enrichment' is selected. A dialog box titled 'Accel Enrichment-0...' is open, showing a table of enrichment values. The table has two columns: 'RPM (RPM)' and '%'. The values are as follows:

RPM (RPM)	%
500	1.62
1000	1.50
1500	1.12
2000	1.00
2500	1.00
3000	1.20
3500	1.33
4000	1.43
4500	1.43
5000	1.43
5500	1.43
6000	1.43
6500	1.43
7000	1.43
7500	1.43

The background window shows the 'Task Info' section with a log of events, including 'Version 1.29.906', '8 memory models read.', '175 ROM metadata models read.', and various modules being loaded.

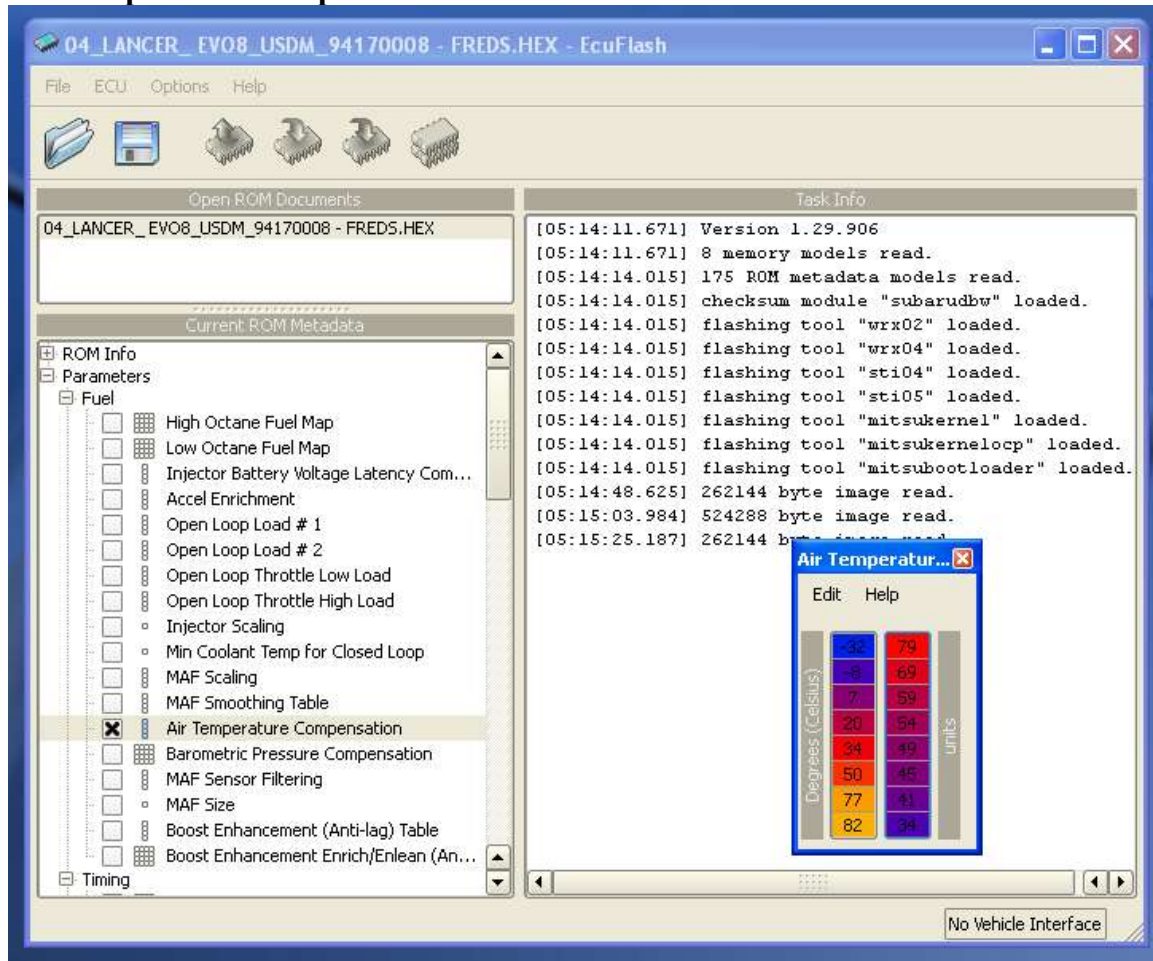
The **Accelerator Enrichment** Table is used as an actual % increase to the AFR **during acceleration**, based on RPM. Changes in these values can affect the responsiveness of an acceleration based on engine speed. Some vehicles have only a short range compared to other like models; this is because the vehicle is likely to have an automatic transmission.

Injector Battery Voltage Latency Compensation Table. (Deadtime)



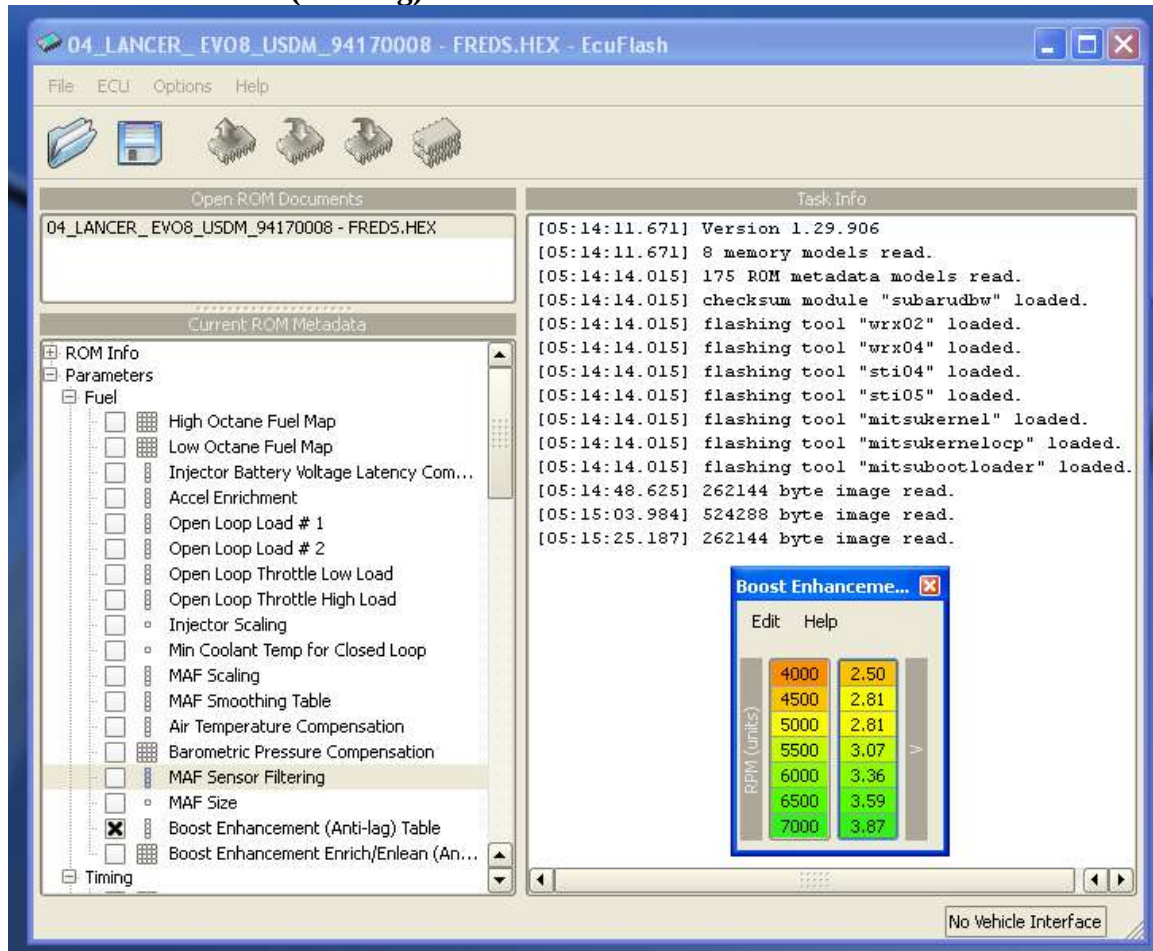
“Deadtime” is also known as the **Battery Voltage Injector Latency Compensation table/map**. The table shows what the **deadtime** of the injector is in milliseconds to different values of **voltage**. The basics are that it takes the injector a certain amount of time, or **deadtime**, to open before it can start spraying fuel. This value is different for differing voltages. The higher the voltage, the shorter the **deadtime**. **Deadtime** has the largest effect on idle or very light loads where the total injector pulse is very small. A small change to deadtime will have a large affect on the fueling. So increasing it will richen up your idle. Under load **deadtime** is only a fraction of the total injector pulse, thus it doesn't have much affect.

Air Temperature Compensation Table.



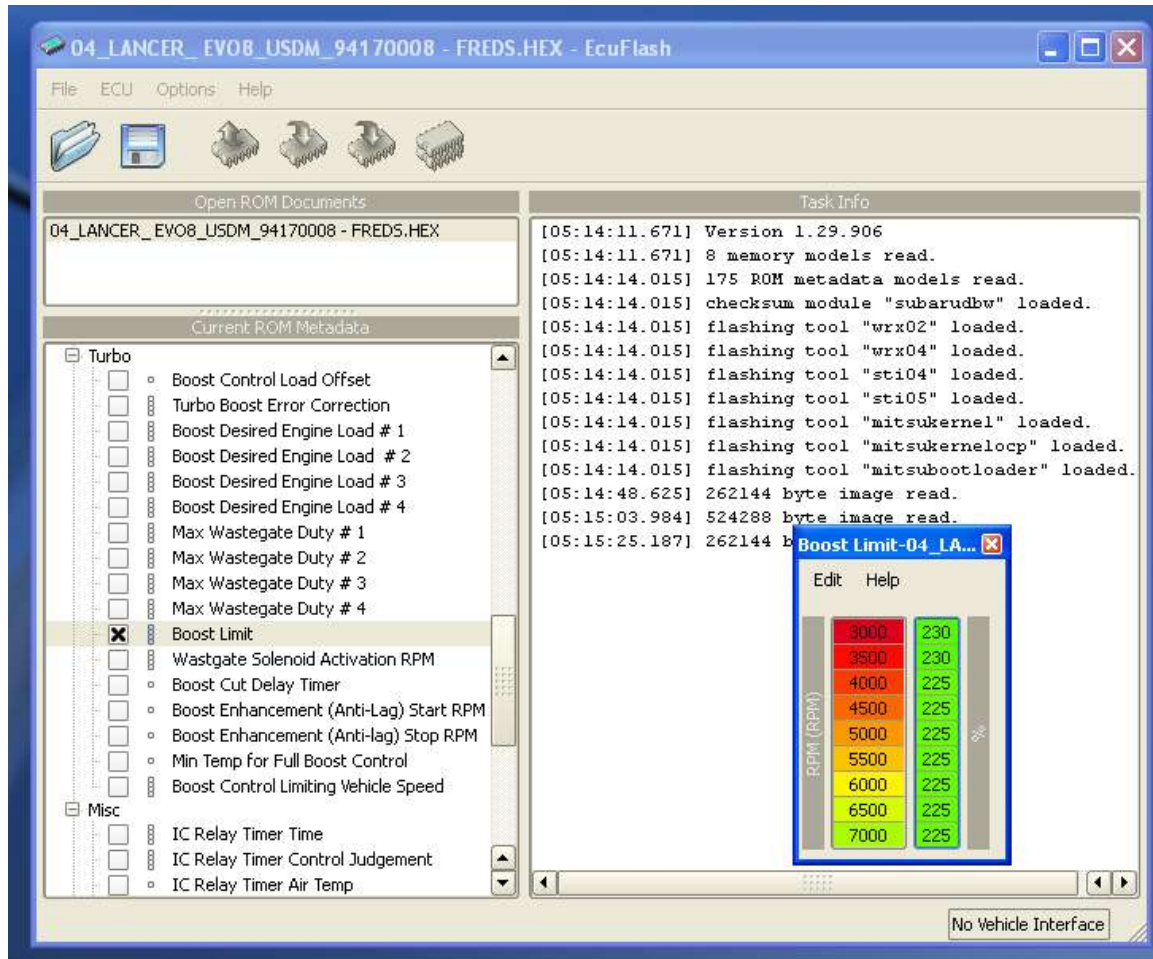
The **Air Temperature Compensation** Table is used to give a proper fuel adjustment based on air temperature. Cold air is denser than hot and needs proper adjusting for a proper stoichiometric AFR.

Boost Enhancement (Anti-lag) Table.



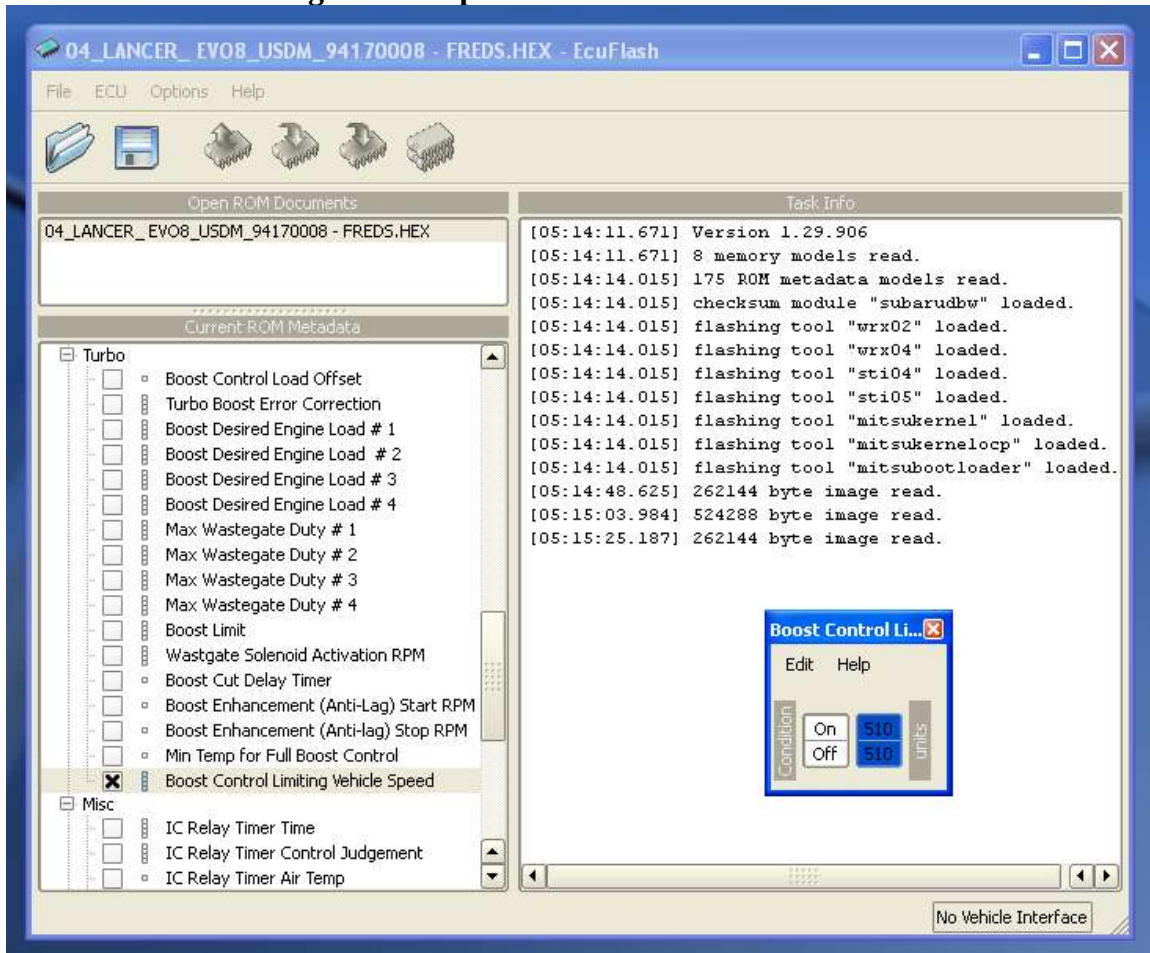
The **Boost Enhancement (Anti-lag)** Table is used to adjust the amount of throttle voltage signal needed (based on engine speed) on when to use the *Boost Enhancement Enrich/Enleanment Map*. Setting the values lower will apply the Boost Enhancement Map at a lesser throttle angle (measured in 1-5 volts).

Boost Limit Table.



The **Boost Limit** Table is representative towards the size of the turbo you may have.

Boost Control Limiting Vehicle Speed Table.



The **Boost Control Limiting Vehicle Speed** Table is much like the *Speed Limiter*, but is boost activated.

Boost Desired Engine Speed Tables 1-4.

The screenshot displays the ECU tuning software interface for a LANCER_EVO8_USDM_94170008. The main window shows the 'Current ROM Metadata' section with various settings. Four smaller windows, titled 'Boost Desired Engine ...', are overlaid, each showing a table of boost values (psi) for different engine speeds (RPM). The tables are color-coded: blue for low boost, green for medium, and red for high boost.

Boost Desired Engine ... (Top Left)

RPM (RPM)	Boost (psi)
500	159.4
1000	159.4
1500	159.4
2000	159.4
2250	159.4
2500	159.4
2750	159.4
3000	159.4
3500	159.4
4000	155.0
4500	150.0
5000	150.0
5500	150.0
6000	140.0
6500	135.0
7000	125.0

Boost Desired Engine ... (Top Right)

RPM (RPM)	Boost (psi)
500	159.4
1000	159.4
1500	159.4
2000	159.4
2250	159.4
2500	159.4
2750	159.4
3000	146.9
3500	146.9
4000	141.2
4500	140.0
5000	140.0
5500	140.0
6000	133.1
6500	126.9
7000	116.9

Boost Desired Engine ... (Bottom Left)

RPM (RPM)	Boost (psi)
500	159.4
1000	159.4
1500	159.4
2000	159.4
2250	159.4
2500	159.4
2750	159.4
3000	146.9
3500	146.9
4000	141.2
4500	140.0
5000	140.0
5500	140.0
6000	133.1
6500	126.9
7000	116.9

Boost Desired Engine ... (Bottom Right)

RPM (RPM)	Boost (psi)
500	159.4
1000	159.4
1500	159.4
2000	159.4
2250	159.4
2500	159.4
2750	159.4
3000	153.8
3500	153.8
4000	148.1
4500	143.1
5000	143.1
5500	143.1
6000	136.2
6500	130.0
7000	121.9

The bottom right window shows a 'No Vehicle Interface' message.

The **Boost Desired Engine Speed** is read as, the boost desired based on engine speed. This is what controls the *Blow Off Valve*, and should be set accordingly. Multiple tables are used in the case that boost error is detected.

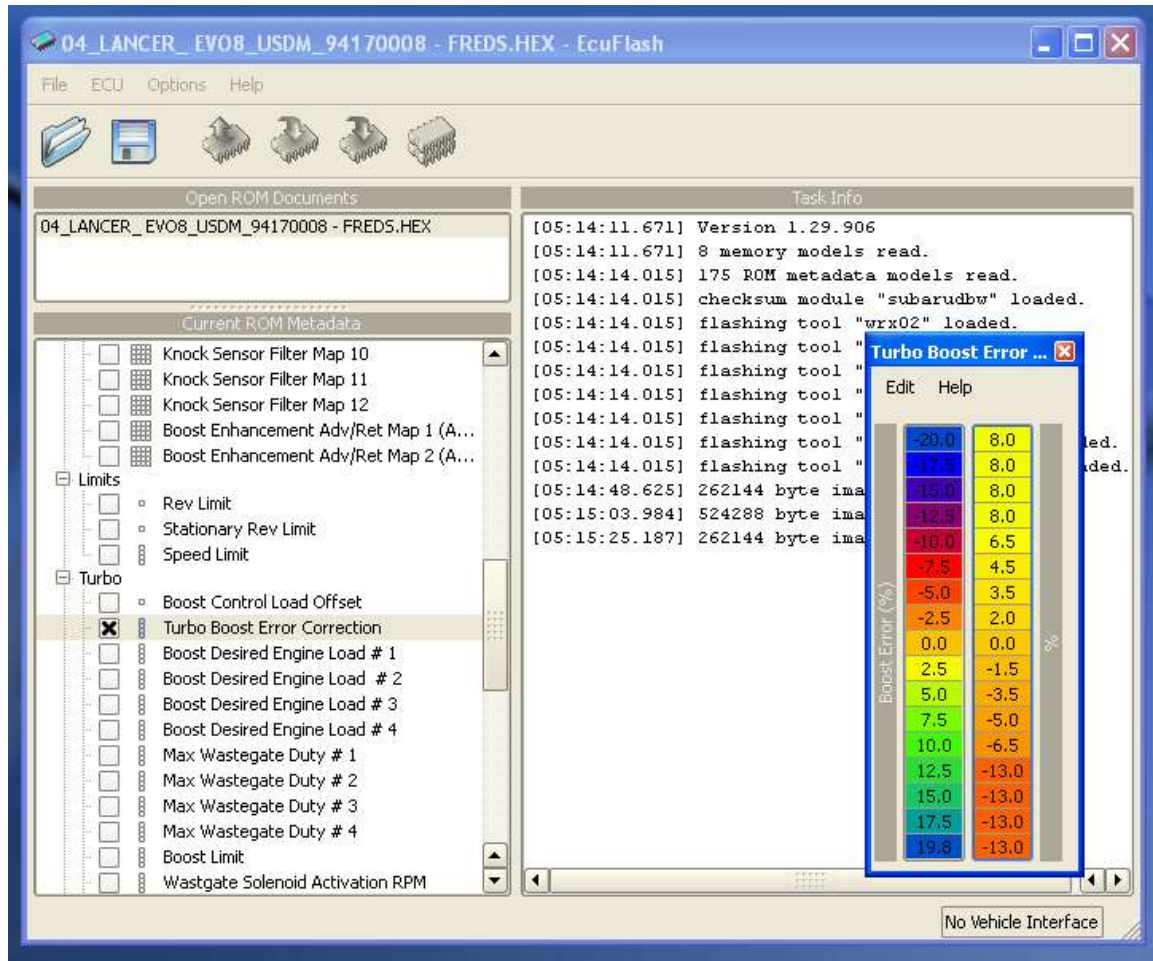
Max Wastegate Duty Tables 1-4.

The screenshot shows a software interface for engine tuning. The main window displays engine metadata and settings for '04_LANCER_EVO8_USDM_94170008 - FRED'S'. Four pop-up windows show the 'Max Wastegate Duty' tables for different engine conditions. Each table has RPM (RPM) on the y-axis and % on the x-axis. The tables show a constant 100.0% duty across all RPM values from 900 to 7000 RPM.

RPM (RPM)	%
900	100.0
1000	100.0
1500	100.0
2000	100.0
2250	100.0
2500	100.0
2750	100.0
3000	100.0
3500	100.0
4000	100.0
4500	100.0
5000	100.0
5500	100.0
6000	100.0
6500	100.0
7000	100.0

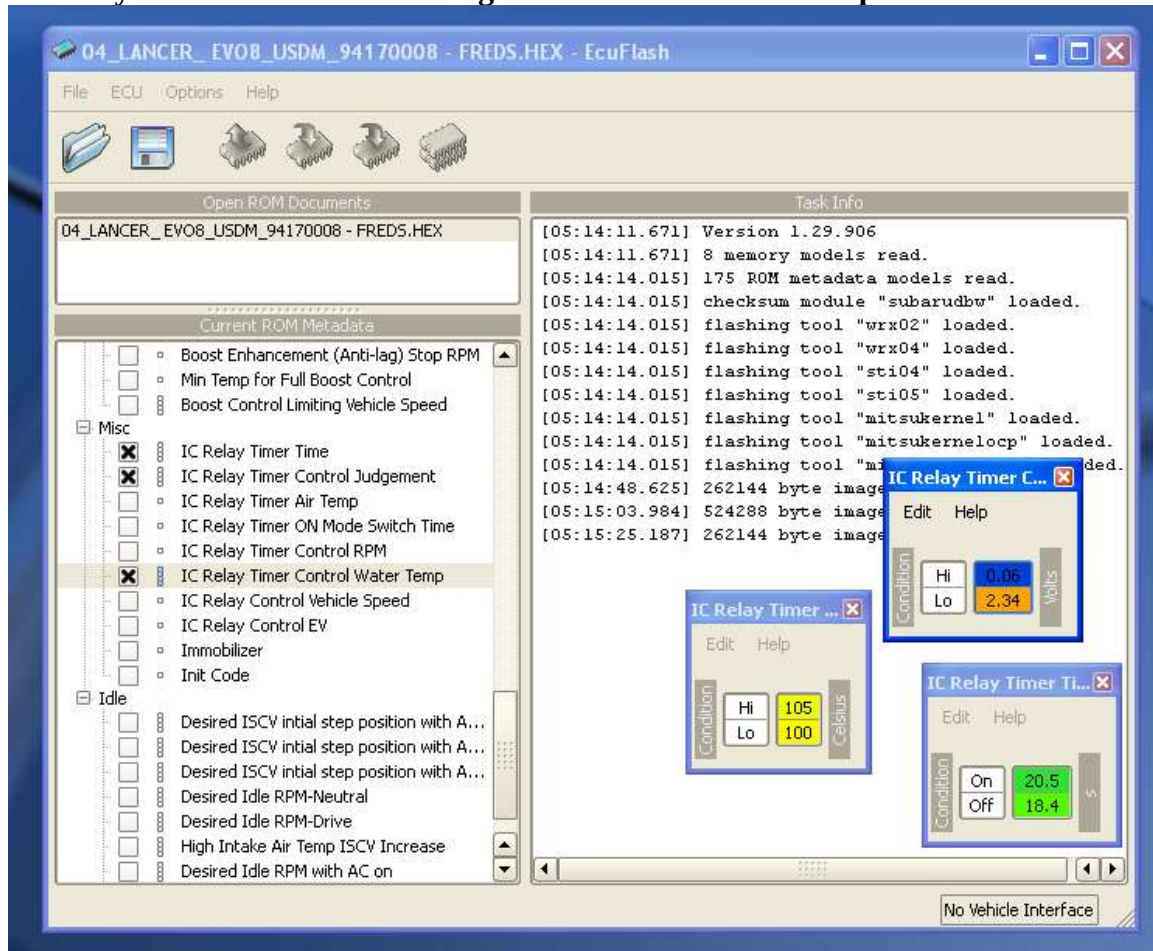
The **Max Wastegate Duty** Table is the amount the *wastegate* will open up so built up exhaust pressure does not cause boost creep. This can be change to raise the turbo spool time and to a certain extent, increase turbo output. Multiple tables are used in the case that boost error is detected, although this particular ROM is set to 100% *Wastegate duty*.

Turbo Boost Error Correction.



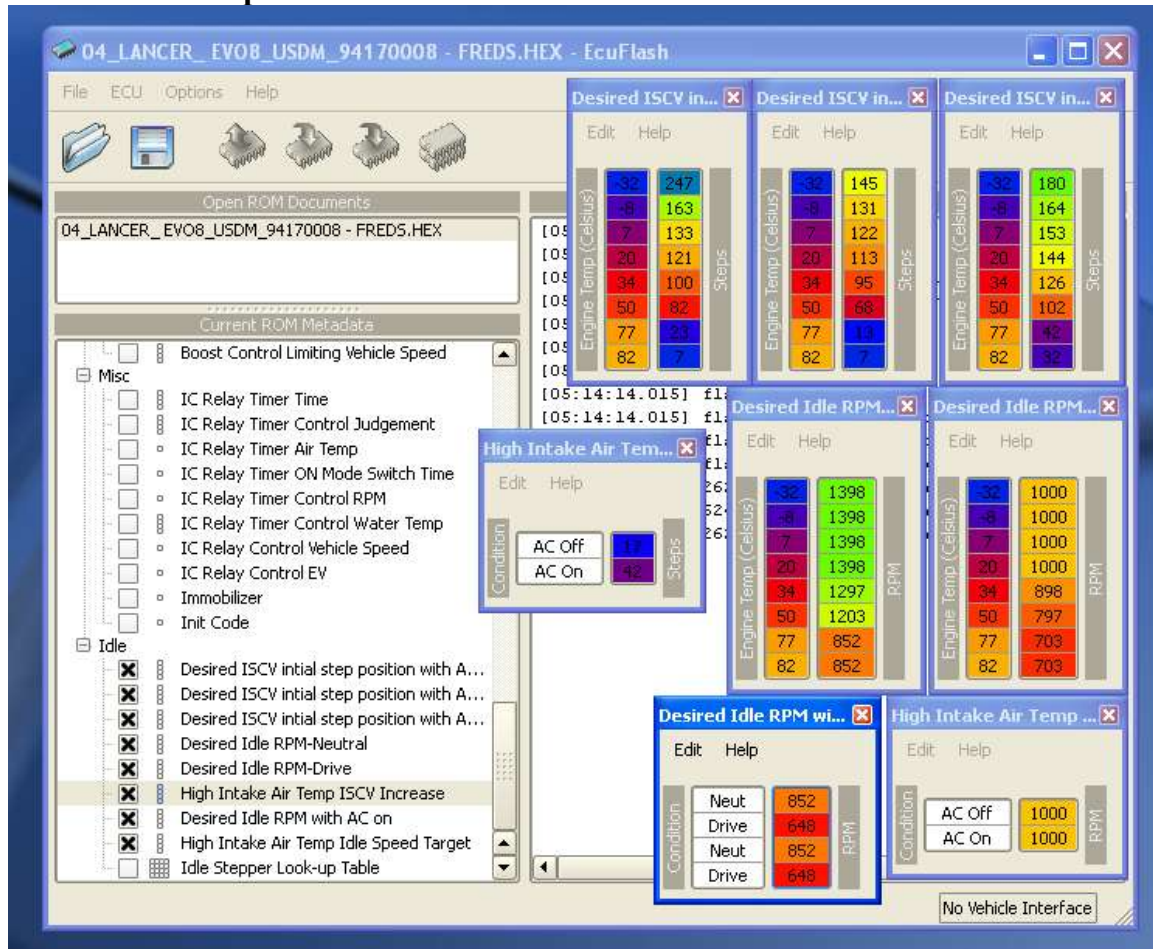
The **Turbo Boost Error Correction** Table is a ratio of % of *Boost Error*, to the % of overall turbo duty (max pressure). The more error recorded by the boost sensor the more compensation is needed.

IC Relay Timer Time/Control Judgement/Control Water Temp Tables.



The **IC Relay Timer** Tables are an advanced set of parameters based on particular models. I imagine that if someone were to come up with a custom engine most likely race modified and the stock engine management will suffice for your needs. Then change the given values. If you need to change the values and do not know what they mean, please take the car to a professional tuner.

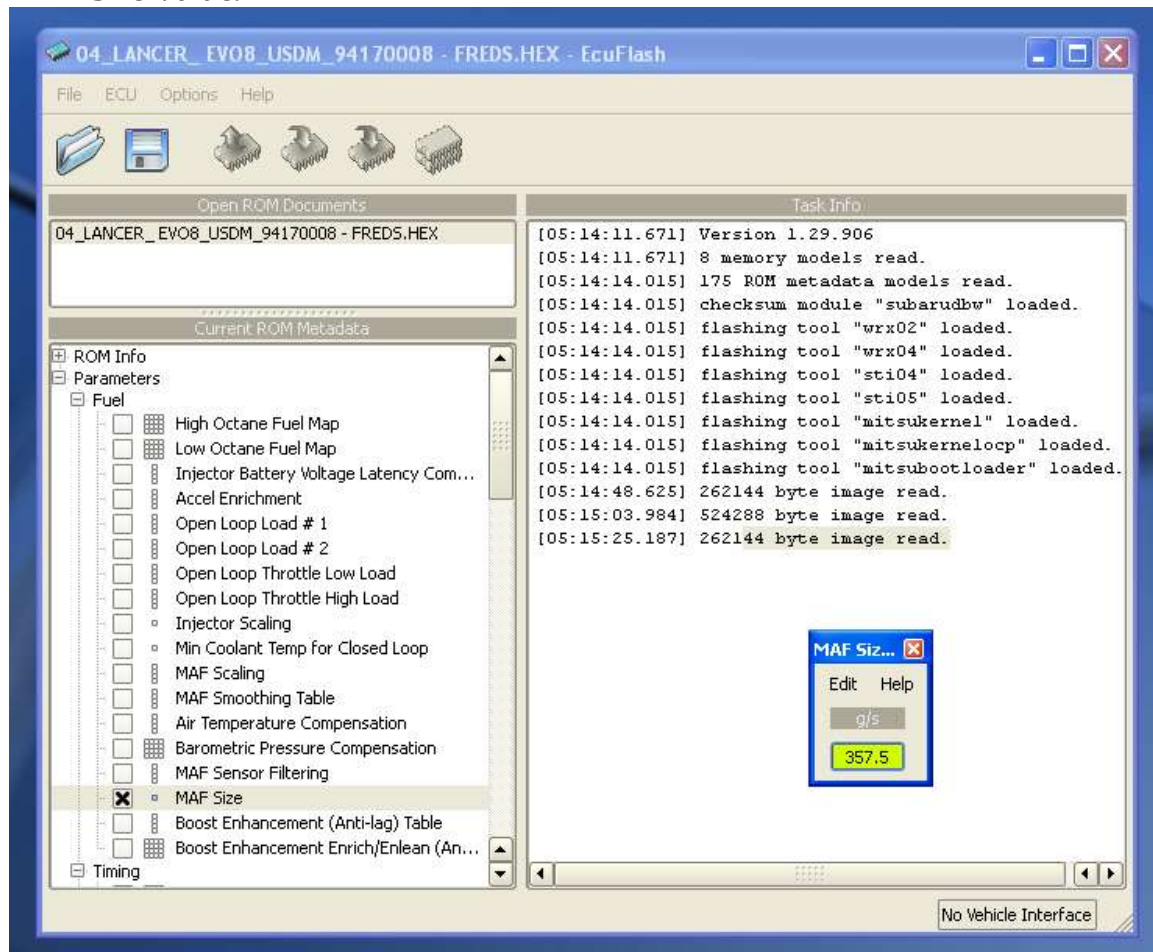
Desired ISCV steps/Desired Idle Tables.



Desired ISCV and Idle Adjustment Tables are used to fine tune a vehicle to the particular options package supplied. Settings are/can be made depending on the amount the engine was tuned but are going to probably be changed to the driver's desired setting. All are self explanatory and be changed only if you know the functions of the table.

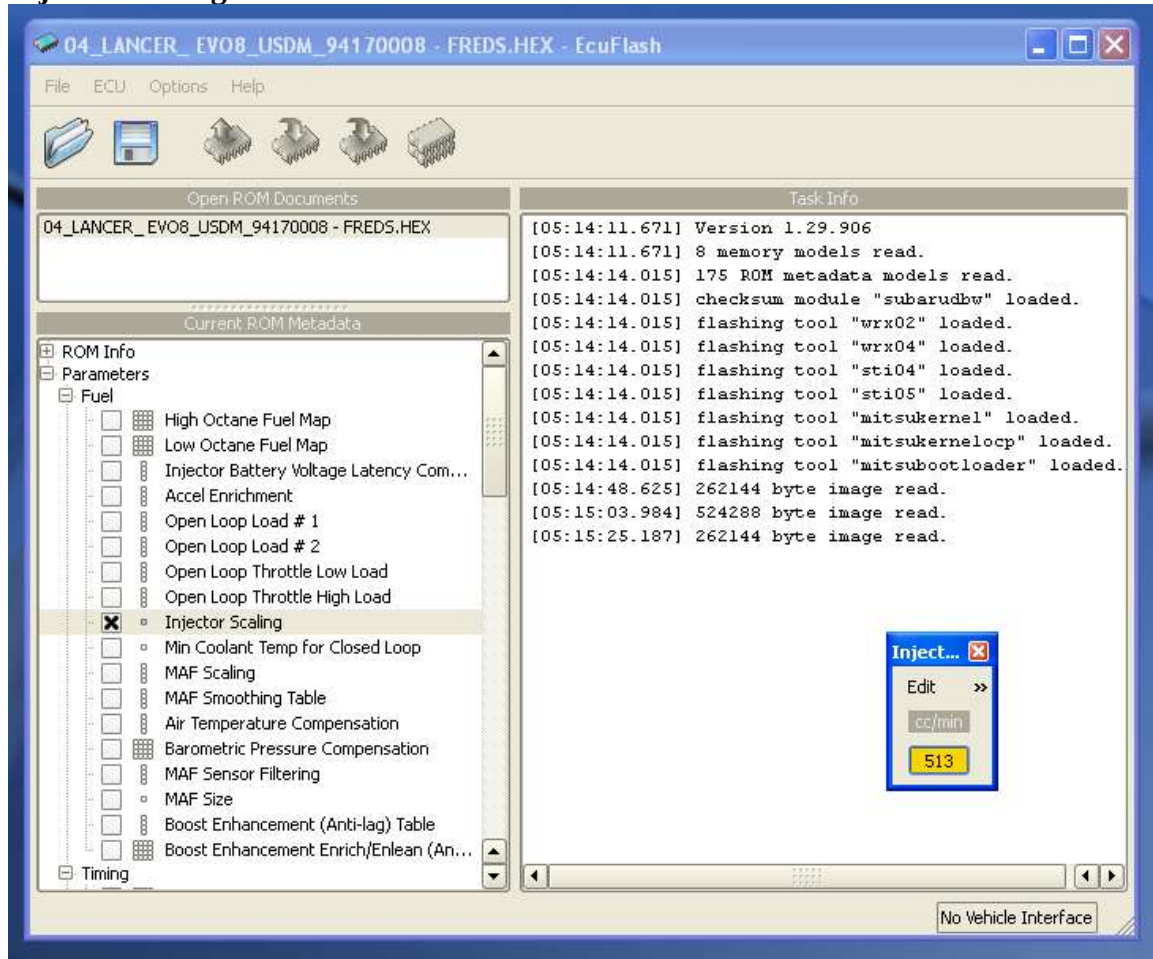
1D Maps –

MAF Size Value.



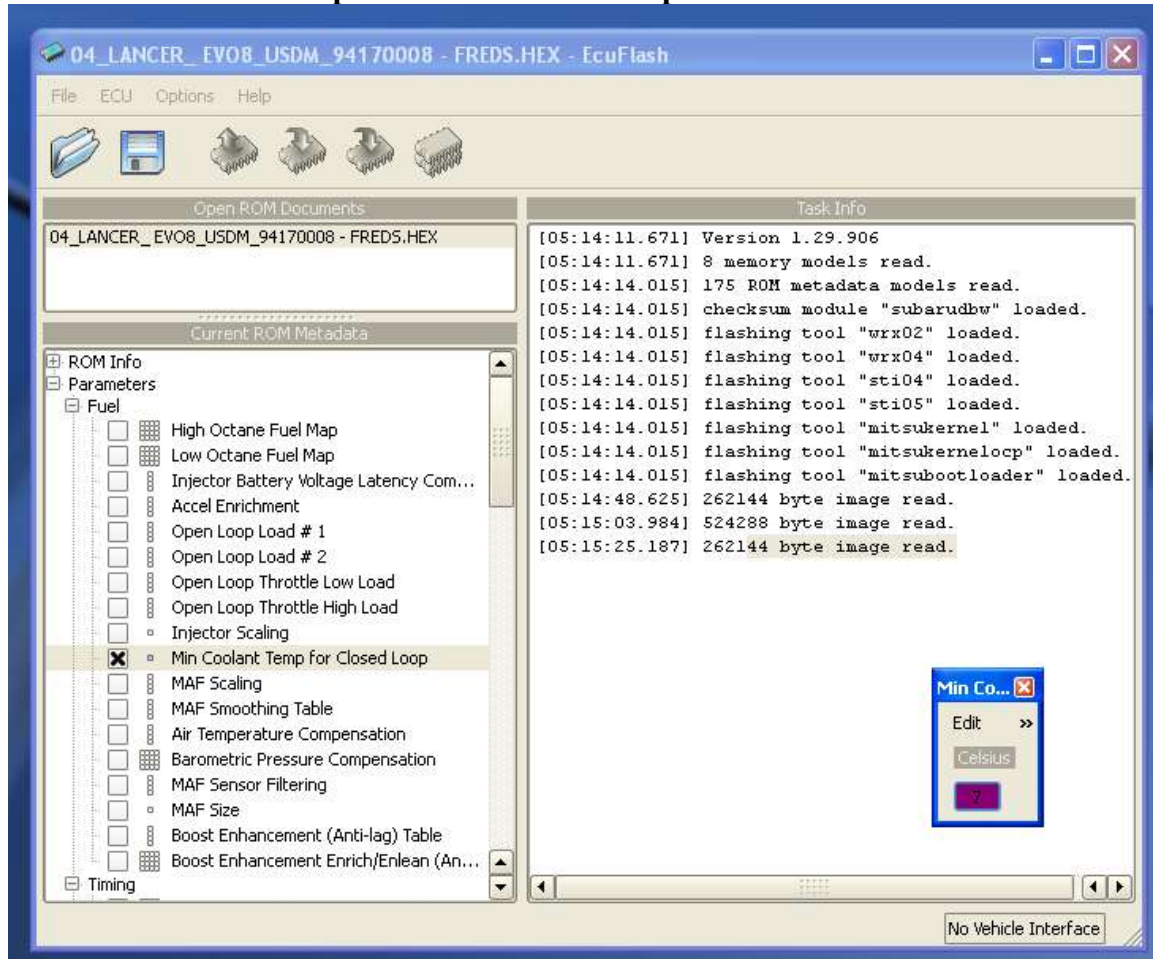
The **MAF Size** data value is the most allowable amount of air measured in *grams/second*. The **MAF size** is also correlated with almost everything fuel related.

Injector Scaling Value.



Injector Scaling data value representing how much fuel that is coming out of the injectors measured in *cubic centimeters/ minuet*. The value should be set to the size of the injectors that are replacing the stock. Sometimes a little tweaking is needed to dial in the correct amount depending on injector manufacturer and fuel system pressure (FPR, Pump) This is another figure that is important to any tune.

Minimum Coolant Temperature for Closed Loop Value.



The *Minimum Temperature for Closed Loop* value is as the name implies.

04_LANCER_EVO8_USDM_94170008 - FRED.S.HEX - EcuFlash

File ECU Options Help

Open ROM Documents

04_LANCER_EVO8_USDM_94170008 - FRED.S.HEX

Current ROM Metadata

- ☐ Boost Desired Engine Load # 4
- ☐ Max Wastegate Duty # 1
- ☐ Max Wastegate Duty # 2
- ☐ Max Wastegate Duty # 3
- ☐ Max Wastegate Duty # 4
- ☐ Boost Limit
- ☐ Wastegate Solenoid Activation RPM
 - ☐ Boost Cut Delay Timer
 - ☐ Boost Enhancement (Anti-Lag) Start RPM
 - ☐ Boost Enhancement (Anti-lag) Stop RPM
 - ☐ Min Temp for Full Boost Control
 - ☐ Boost Control Limiting Vehicle Speed
- ☒ Misc
 - ☐ IC Relay Timer Time
 - ☐ IC Relay Timer Control Judgement
 - ☐ IC Relay Timer Air Temp
 - ☐ IC Relay Timer ON Mode Switch Time
 - ☐ IC Relay Timer Control RPM
 - ☐ IC Relay Timer Control Water Temp
 - ☐ IC Relay Control Vehicle Speed
 - ☐ IC Relay Control EV
 - ☐ Immobilizer

Task Info

```
[05:14:11.671] Version 1.29.906
[05:14:11.671] 8 memory models read.
[05:14:14.015] 175 ROM metadata models read.
[05:14:14.015] checksum module "subarudbw" loaded.
[05:14:14.015] flashing tool "wrx02" loaded.
[05:14:14.015] flashing tool "wrx04" loaded.
[05:14:14.015] flashing tool "sti04" loaded.
[05:14:14.015] flashing tool "sti05" loaded.
[05:14:14.015] flashing tool "mitsukernel" loaded.
[05:14:14.015] flashing tool "mitsukernelocp" loaded.
[05:14:14.015] flashing tool "mitsubootloader" loaded.
[05:14:48.625] 262144 byte image read.
[05:15:03.984] 524288 byte image read.
[05:15:25.187] 262144 byte image read.
```

Boost C... Edit Help

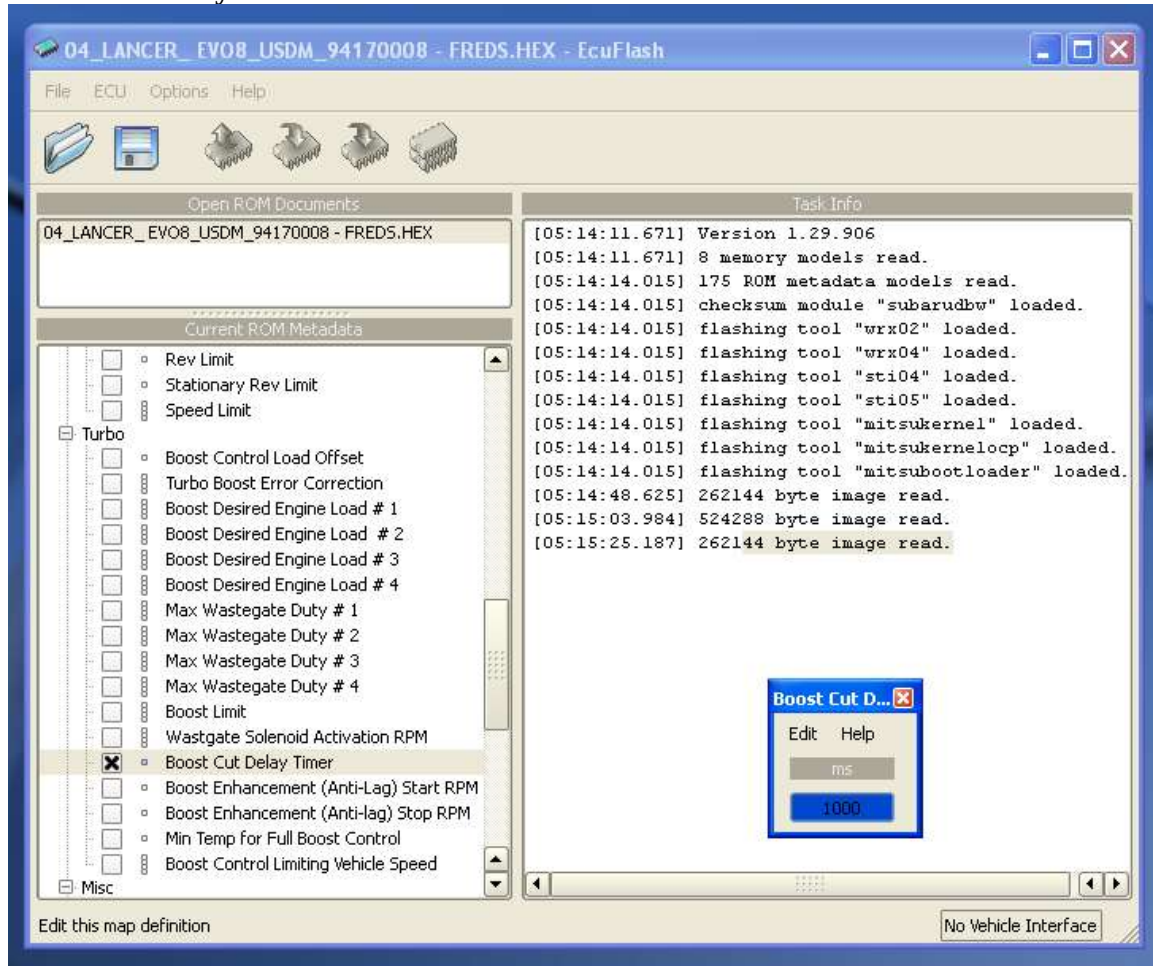
units

60.0

No Vehicle Interface

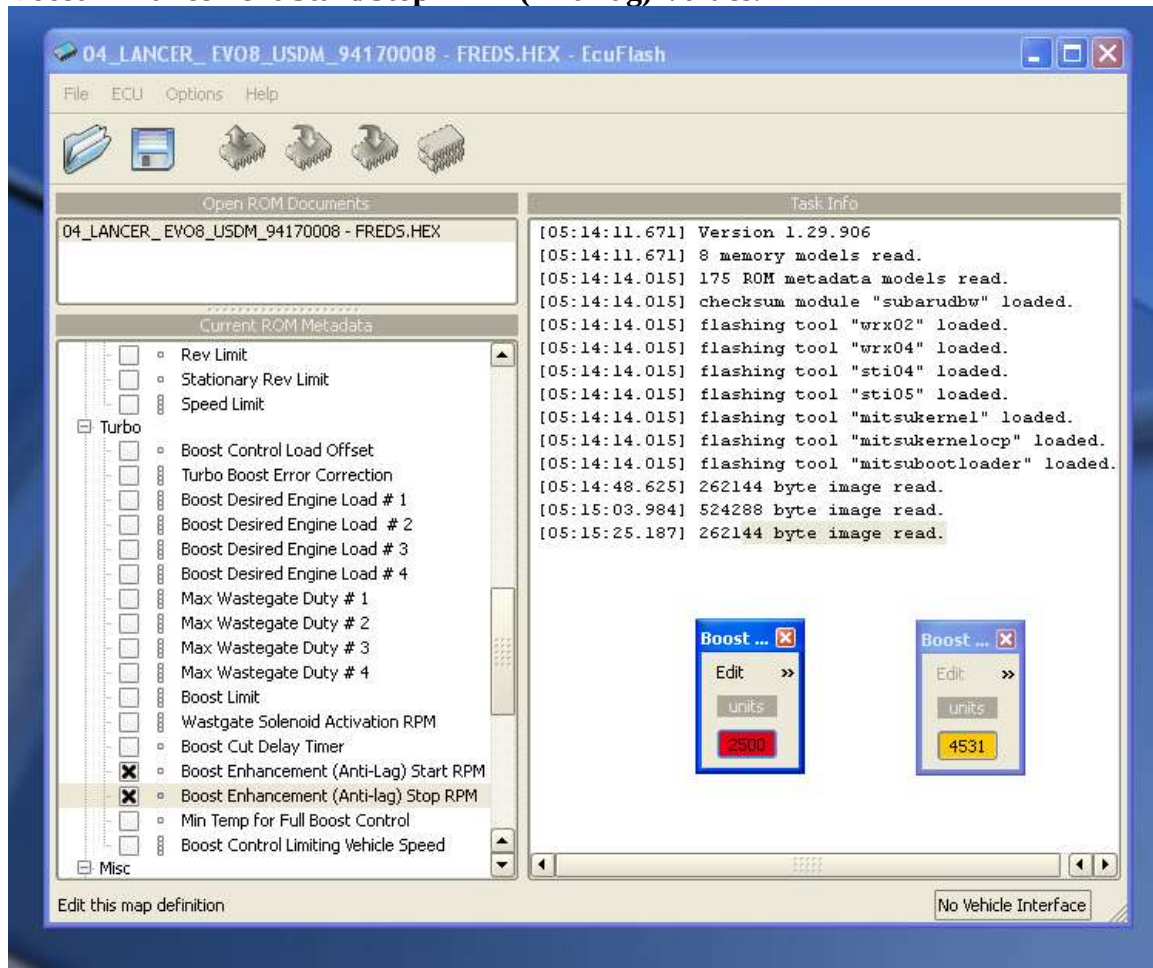
The ***Boost Control Load Offset*** is ??

Boost Cut Delay Timer Value.



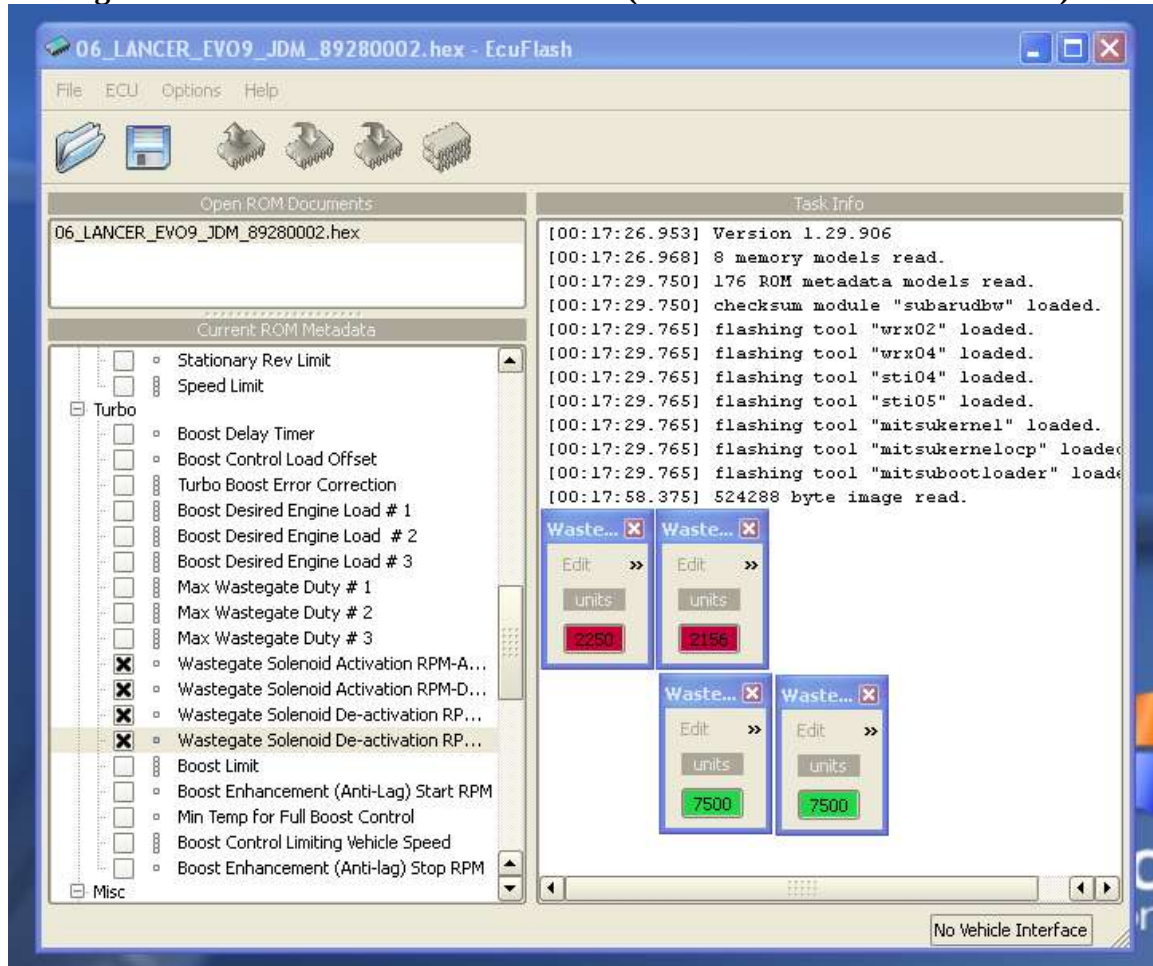
The **Boost Cut Delay Timer** allows a timed delay for the effects of boost cut off. It allows a quick return buffer if the boost cut were onset by a quick drop into the cut off range but a return to acceleration.

Boost Enhancement Start/Stop RPM (Anti-lag) Values.



The **Boost Enhancement Start/Stop RPM (Anti-lag)** Values are as the name applies, measured in RPM. This directly correlates as the main on/off controller to the Boost Enhancement (Anti-Lag) Tables, both 2D and 3D.

Wastegate Solenoid Activation/De-activation (RPM Activation/De-Activation)



The **Wastegate Solenoid Activation/De-Activation (RPM Activation/De-activation)** values are related to *engine speed (RPM)* and the *Wastegate* of the turbo.

Most automotive turbochargers have a **Wastegate**, which allows the use of a smaller turbocharger to reduce lag while preventing it from spinning too quickly at high engine speeds. The wastegate is a valve that allows the exhaust to bypass the turbine blades. If boost pressure gets too high, it could be an indicator that the turbine is spinning too quickly, so the wastegate bypasses some of the exhaust around the turbine blades, allowing the blades to slow down.

The *Wastegate Solenoid Activation/De-Activation* have 2 additional values each, representing *RPM* initiated (or *Load* initiated) values. This may be confusing but here is the sum of it.

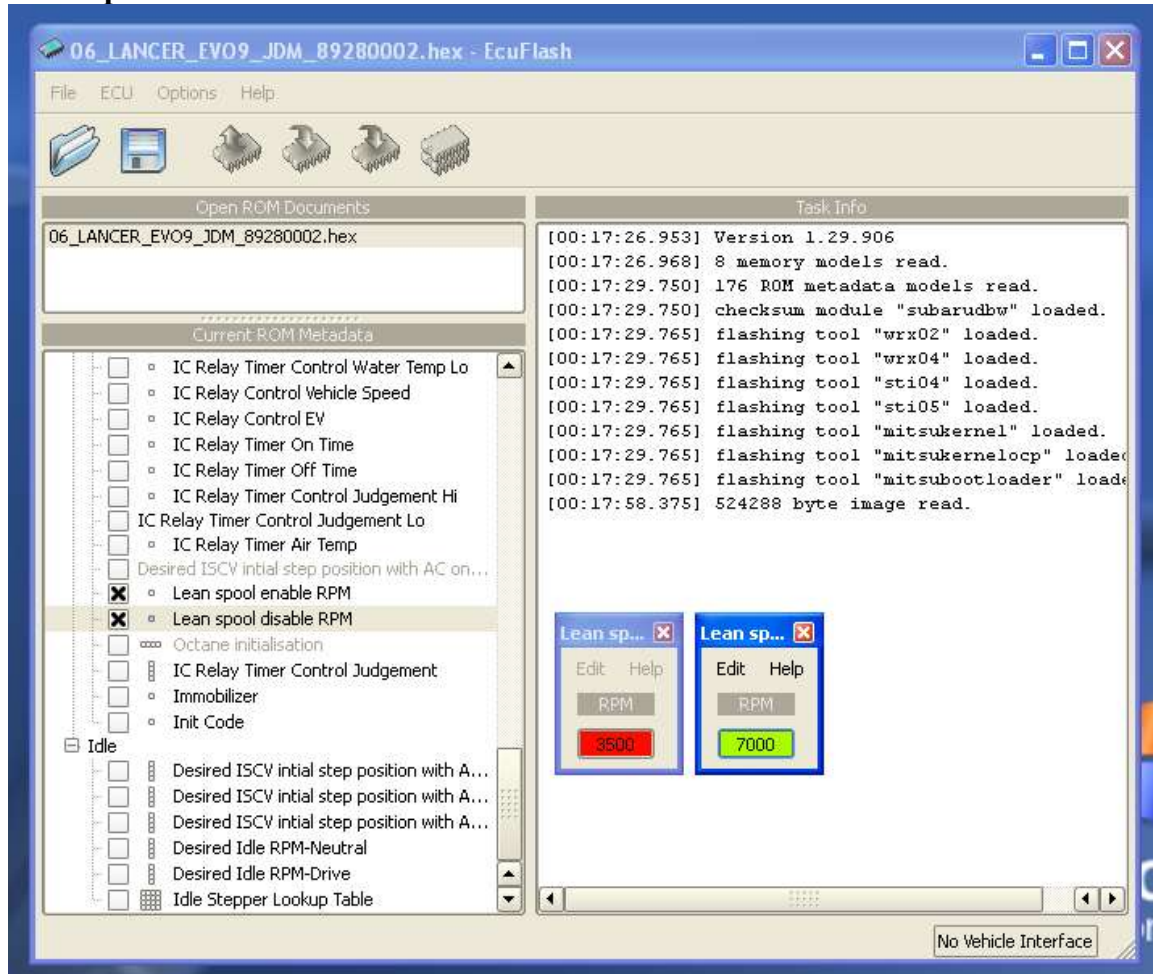
The **Wastegate Solenoid Activation RPM – Activation** is the RPM value related to when the *Wastegate* will initially activate and respond to the other *Wastegate* related tables. The **Wastegate Solenoid Activation RPM – De-Activation** is the minimum value related to when the *Wastegate Solenoid* will “turn off”. This helps with boost lag at low *RPM* by disregarding boost pressure until a *RPM* thresh-hold is met, and then allowing a safe level of boost up the *RPM* band.

The **Wastegate Solenoid De-Activation RPM – Activation** is the RPM value representing when the wastegate will be deactivated and is usually set to somewhere around the Rev Limit.

The **Wastegate Solenoid De-Activation RPM – De-Activation** is the same as above and is reflective of the same principals as above.

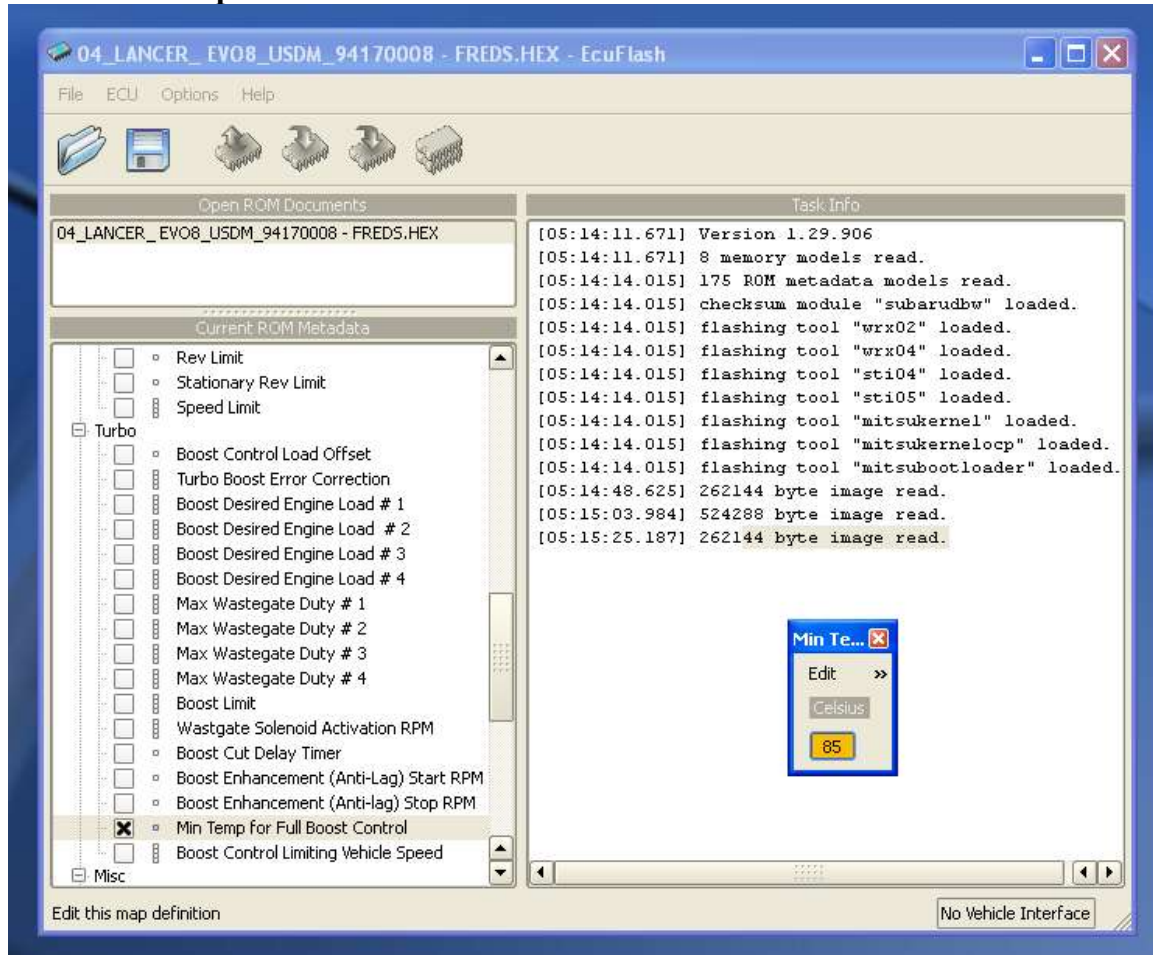
My only thought of to why there is a small offset in some of the *Rev Limits* and the *Wastegate Solenoid De-Activation Tables*, is to give some extra boost at the final 200-300 RPM band (before the Rev Limiter kicks in). Perhaps to keep the spool up on the turbo while an imminent shift is coming, giving extra boost while “pushing” into the next gear. I mean if your about to hit your *Rev Limit*, it is probable you will most likely be going “full load” into the next gear.

Lean Spool Enable/Disable



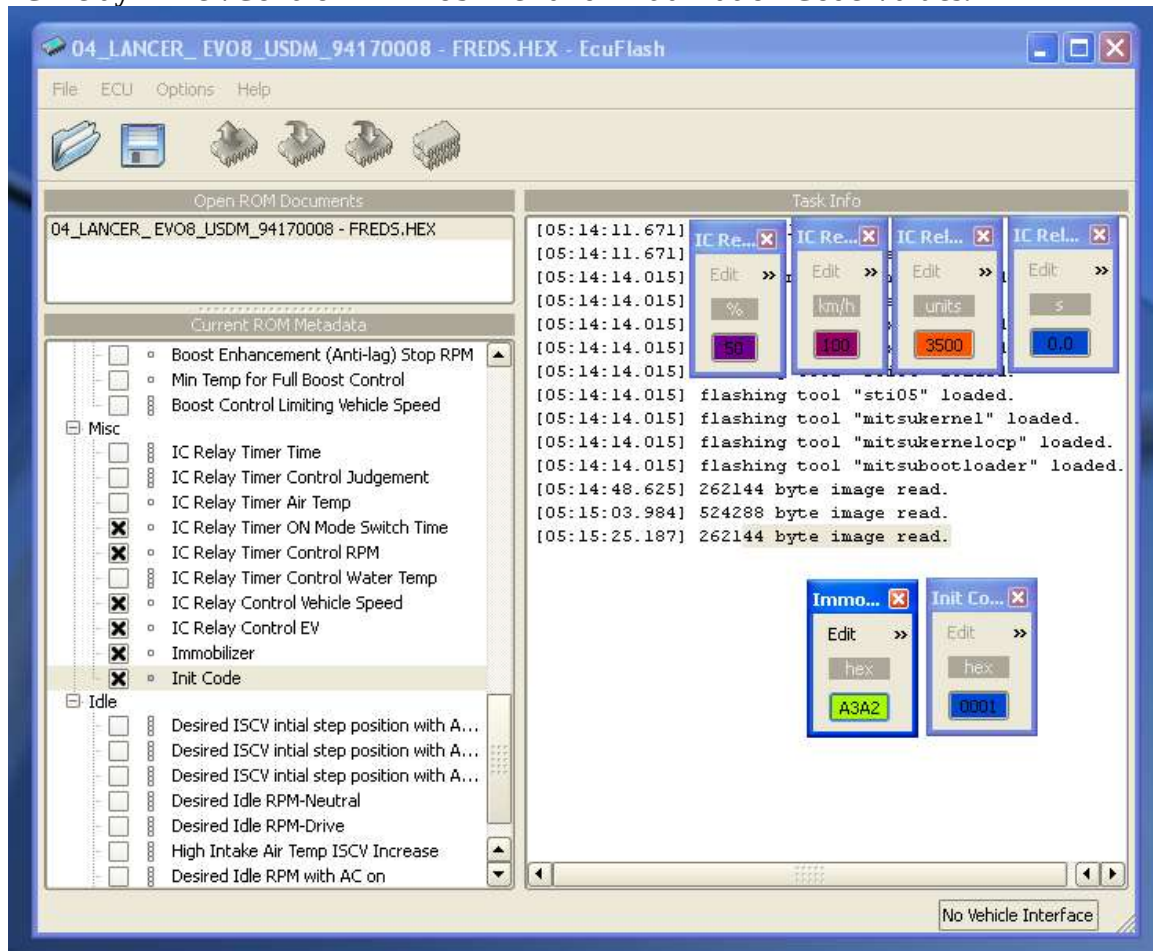
The **Lean Spool Enable/Disable** tables are values representing the level of RPM to the ability to run positive (+) boost under lean conditions. The minimum value is set to a level not to interfere with any of the anti-lag functions. The maximum setting is to prevent too much positive (+) boost while approaching the Rev Limit. This allows the turbo to function under the AFR table which is usually substantially richer at higher RPM and Load. This could come in handy if you are running the full length of the RPM often and need higher boost pressure going into the next gear.

Minimum Temperature for Full Boost Control Value.



The ***Minimum Temperature for Full Boost Control*** Value is the measurement in *Celsius* that allows the use of the most aggressive desired setting to be used.

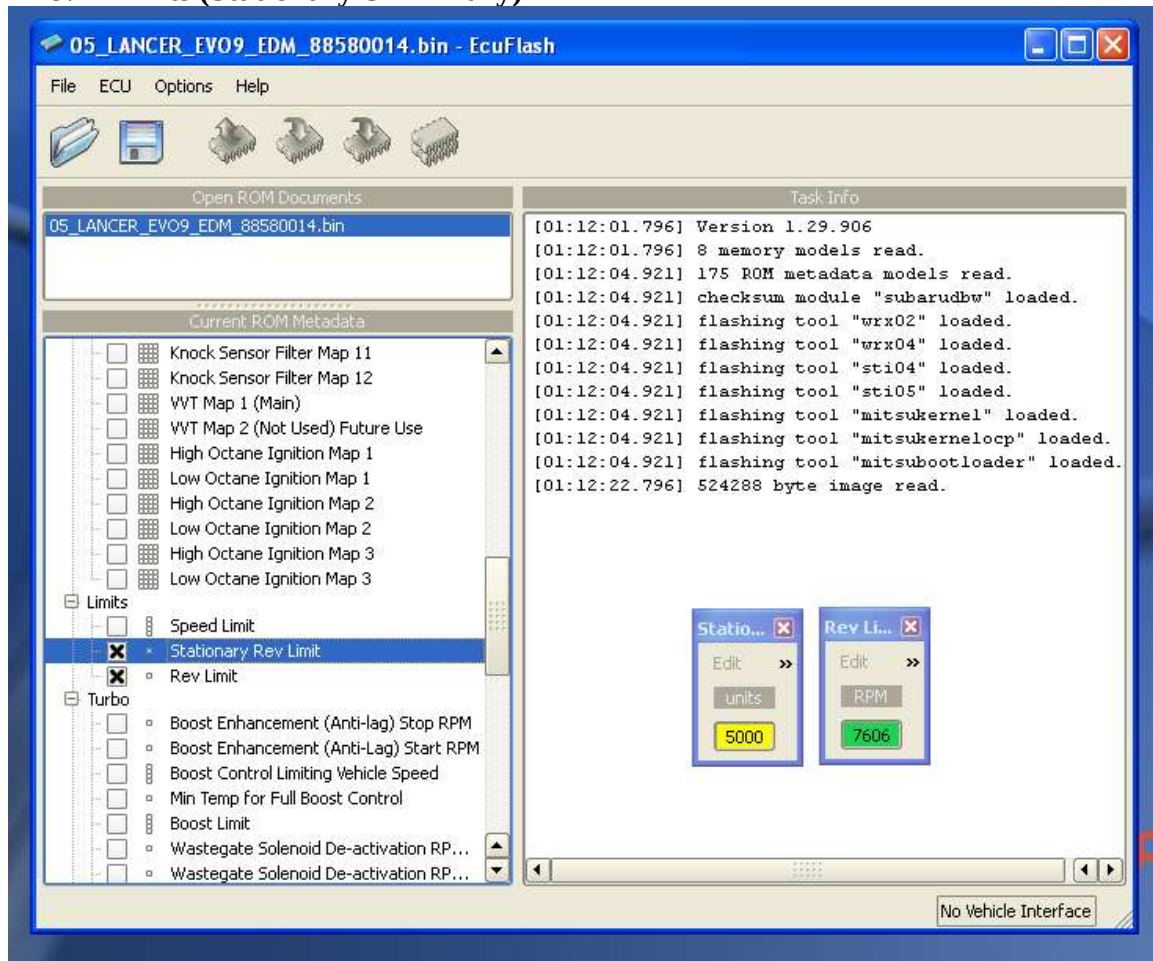
IC Relay Timer/Control – Immobilizer and Initialization Code Values.



The **IC Relay** Tables are used to fine tune a vehicle to the particular options package supplied. Settings are/can be made depending on the amount the engine was tuned but are going to probably be changed to the driver's desired setting. All are self explanatory and be changed only if you know the functions of the table.

The **Immobilizer** is the **HEX** of the Ignition Key Security Pill uses to start and protect the vehicle from being hotwired. This particular car should have changed this value before posting or sharing the ROM image (writing FFFF in its place) and stored/protected for personal use. Although his wife got the car in the divorce, so have fun. This should match for all ROMs flashed to a particular model/option vehicle, and should stay the same as the ORIGINAL ROM IMAGE. Poor alterations or forgetting to fallow through on this might leave the vehicle UNDRIVABLE. Although a simple reflash with the right code should fix it.

“Rev” Limits (Stationary & Primary)



The **Rev Limiter** Value is the actual maximum *RPM* value that is allowed until the engine is limited by the ignition and fuel systems. This Value represents a “vehicle moving” table if a *Stationary Rev Limit* Value is present, depending on the particular model. If no *Stationary Rev Limiter* is on the particular model, then the *Rev Limiter* will engage, even when the vehicle is in *Neutral* or *Park*.

The **Stationary Rev Limiter** Value is the same as above, although “stationary” means “not moving”, so it is the limit of the engines *RPM* in a parked state. To reiterate, this value is depending on the model and year vehicle. When absent from the *Parameters Option*, the *Rev Limiter* will take over in its place.

