SE-EFI

Engine Electronic Fuel Injection

Tuning Guide

V1.8

HAE

Copyright HAE LLC

COPY RIGHTS HAE

ALL RIGHTS RESERVED

Note: this tuning guide will be under continuous improvements. Contact us for the latest version.

Note: only use this Tuning Guide after you finished the installation of the EFI Kit, by following HAE Installation Manual.

Copyright HAE LLC

Quick Answers 3
Chapter 1 System Requirements 6
1.1 Laptop tuning software HAECAL 6
1.2 Smart Phone tuning software HAE DroidCAL 6
1.3 Load the correct files by using HAECAL 6
Chapter 2 Engine Start 8
2.1 Start issue trouble shooting 8
2.2 Global fuel tuning 12
2.3 Start fuel factor 13
2.4 After-Start and Warm-up fuel factor 15
Chapter 3 Engine idle tuning 24
3.1 Target Idle Speed 24
3.2 Idle speed adjustment via Mechanical methods 25
3.3 Electronic Idle Air Control – IAC motor 33
3.4 Idle Controls via ignition timing 45
3.5 Idle fuel too rich because of minimum injector pulse width 48
Chapter 4 Open-loop and closed-loop fuel control 53
4.1 Open Loop Fuel Control 53
4.2 Closed Loop fuel control 54
Chapter 5 VE table and Load table tuning 59
Chapter 6 Ignition Timing Tuning 75
Chapter 7 Advanced Tuning 83
7.1 Transient fuel 83
7.2 Decel-Fuel-Cut-Off 89
7.3 Driver’s desired lambda 90
7.4 Advance Calibration for engine timing 93
7.5 Dual injectors advanced tuning 106

7.6 Altitude tuning 108
Chapter 8 Sensors and actuators 109
8.1 Sensors' characteristics calibration 109
8.1.1 Injector characteristics 109
8.1.2 Engine displacement 110
8.1.3 MAP sensor 111
8.1.4 IAT sensor 112
8.1.5 ECT sensor 114
8.1.6 TPS sensor 115
8.2 Fault and Heal 118
8.2.1 TPS sensor diagnostic 119
8.2.2 MAP sensor diagnostic 127
8.2.3 Temperature sensor (IAT & ECT) diagnostic 131
8.2.4 NB O2 sensor diagnostic 131
8.2.5 VRS signal diagnostic 132
8.2.6 Battery diagnostic 132
8.2.7 Actuator Diagnosis 132
Chapter 9 Advance functions of EFI 133
9.1 Fuel Dial Function 133
9.2 Traction Control function 135
9.3 Oil pump control tuning 138
Chapter 10 Auto-tuning with ALM 139
Chapter 11 On-the-fly Calibration Data 157
Chapter 12 about HAE software tools and ECU communications 159
12.1 Check A2l or CAL which should be matching ECU 159
12.2 Burn to ECU successfully 162
12.3 load configuration in variables list 162
.....

Quick Answers

Q: Is it easy to tune HAE EFI?

This is depending on how much you know EFI and how much time you want to spend on learning. First of all, HAE EFI is coming from an automotive industry professional engine control system. It is more sophisticated than most of the aftermarket EFI systems. Though we tried to make the tuning software simple and user friendly, it still requires some software skills and practice to get familiar with. We recommend the below:

- If you have done EFI conversions and used other EFI tuning software before, and if you are not shy to learn some new software user interface; or
- if you have a strong engineering background, like electrical engineering or computer science, you have a good base to learn our software; or
- if you are college students, and the purpose of the project is to learn and practice anyway;

You should be able to use our tuning software to tune the EFI.

For most weekend racers, hobbyists or similar users, who are mechanical inclined, but shy of computer skills, we recommend you to just log data with our EFI, send data to us, and let us do the tuning for you.

You can always try to tune it by yourself. This is no reason you cannot learn the software. It is only matter of your willingness and time.

HAE tech support team provides tuning help via Emails, not on the phone, because 1) we need data logs via Email first, 2) phone calls are often filled with basic questions on EFI and computer skills which are supposed to learn via reading manuals.

Note: HAE EFI is meant to provide a complete EFI kit that includes almost all parts for you to convert a carburetor engine. It is not a Plug and Play, aka PNP, system. Tuning is required more or less. The benefit of HAE EFI is that you get a complete kit, instead of an ECU with a harness. Plus you get tuning support from HAE. With others, you may be left un-answered, or alone.

Q: I finished the EFI installation; my engine does not start, why?

Go back to your installation manual, in the last chapter, there are detailed procedures for you to trouble-shoot the starting problem. If still cannot find answers there, you can jump to chapter 2 of this manual for further trouble shooting on engine start issues. The EFI installation manual can be downloaded here: http://www.uavenginesystem.com/download/

Q: How do I log data and send the data files?

Assuming you have installed the HAECAL (default directory should be C:\HAECAL or D:\HAECAL). Otherwise, install the HAECAL to your laptop. HAECAL is coming in the CD with the EFI kit or you can download it from our website. Details on how to use HAECAL can be found in our HAECAL manual, also downloadable.

1) Run HAECAL (load the correct A2L and CAL file);
2) Key on, and key on only;

3) Go to menu -> run -> connect;

4) Go to menu -> run -> start measuring and start recording

5) Do your test, or start the engine;

6) When done the test, go to menu -> run -> stop recording

7) go to menu -> run -> Play back;

8) A window pops-up, and click "open", it will take you to where the logged files are. All files are in a folder called “**C:\HAECAL\record**", assuming C:\HAECAL is where you installed the HAECAL.

All files are recorded in sub-folder names as “dd-mm-yyyy-hh-mm-ss”.

9) You need to copy all 3 logged files and send to us. (Every time there are 3 CSV files logged); you can even zip the whole "record" folder and send to us, we can help you to analyze it.

Q: How do I change a calibration?

Assume you want to change the value of the calibration variable "VAL_xxxxx" to "1234":

1) Key-ON, run HAECAL with the correct A2L/CAL files, connect it to ECU;

2) In HAECAL menu, go to: "Advanced" -> "add advanced calibrations";

3) Pop up a window, type in "VAL_xxxxx", select it, and click "add to", and then click "OK".

4) Back to HAECAL window, "VAL_xxxxx" should be displayed in a small window. Double click it and change its value to "1234". Hit "Enter";

5) Go to menu; "run" -> "burn to ECU".

7) Download.......completed.

8) Start the engine, see the effect.

Chapter 1 System Requirements

1.1 Laptop tuning software HAECAL

HAE tuning software HAECAL can be installed and run in the Microsoft Windows XP, Vista, Win7 or Win8.1.

Note: during the installation of the HAECAL, it is automatically set as the compatibility mode for Windows XP.

1.2 Smart Phone tuning software HAE DroidCAL

DroidCAL is the version of HAECAL that can be run on an Android based smart phones and tablets.

Note: On your phone, you can directly search “HAE” in “Google Play Store” and it is free to download.

Note: HAECAL and DroidCAL are free and you can always download them at our website:
http://www.uavenginesystem.com/download/

1.3 Load the correct files by using HAECAL

S19 file: this is a Motorola format microprocessor executable file;

A2L file: this is a description file that contains the ECU info for HAECAL;

CAL file: this is a calibration data file that contains parameters users can tune. CAL file is one part of the S19 file.

Note: most customers don't need the S19 file; unless an ECU software update is necessary;

It is enough to have the A2L file and CAL file to run HAECAL and tune your engines.

If you have not got A2L file and CAL file in your CD from EFI kit or the software package via Email, please contact us:

info@ uavenginesystem.com

Often the user will need to load different A2L file and CAL file than the default ones coming with the HAECAL. For example, a software update or new calibration release will give you newer A2L file and CAL file.

To load the new A2L file and CAL file, in HAECAL, go to "**File->Open**", and then go to the right folder and select the files.

For more details, please download the HAECAL software Manual.

http://www.uavenginesystem.com/download/

Chapter 2 Engine Start

2.1 Start issue trouble shooting

After install the EFI system, the engine doesn't start, why?

There are some questions need to be answered first:

1 .**which signal is used as the pick-up signal**?

Is it the stock pickup (coil), or a Hall Effect Sensor? Or is it from the Kill witch wire?

If it is a stock pickup (coil) signal, connect the orange wire labeled as CKP from the ECU harness to the pick-up signal of your stock CDI ignition. The other wire of the pickup sensor usually is grounded in your stock system. Connect the green GND wire from the EFI harness to the pickup ground.

If it is a hall sensor, please check: is the hall sensor installed correctly (see Chapter 1)? You need find out what polarity the magnet has. Usually our Hall sensor is triggered by the S polarity magnet; and the distance from the hall sensor to the magnet should be in the range from 2mm to 5mm,.

If it is the stock kill switch wire, you need connect the orange CKP wire to the kill wire, and connect the green wire from the ECU harness to the chassis or the ground of the stock system. If possible, use a scope to check the kill wire signal is clean, meaning, only one pulse per revolution.

Make sure your engine block is connected to the 12V negative, so all grounds are common!

2. **The pickup is correct, but the engine still doesn't start, why?**

Do you have the MAP sensor connected to the intake manifold for 4 stroke engines? The Map sensor needs to measure the vacuum for 4 stroke engines to start.

For 2 stroke engines, MAP sensor can be hanged in the air and measure the ambient pressure only.

3. **Does the fuel pump turn on for a few seconds when powering on the ECU?**

If the fuel pump does not turn on after the ECU power is on, then check your wire connection. Or if the fuel pump does not turn on when the engine is spinning, you need to check the CKP wire connection.

4**. Is there a RPM reading by ECU?**

Start to use HAECAL to read the RPM. Connect your laptop to the ECU. If there is no RPM reading when the engine is spinning; that means something wrong with the CKP signal, or the MAP sensor signal for 4 stroke engines.

5. **Is there any fuel injection when cranking?**

1) un-screw and pull out the injector,

2) Seal the hole of injector mount;

3) Hold the injector in air, point it to a safe direction, and use a cloth to block fuel jet!

4) crank the engine;

If there is no injection, please check the CKP wire connection, and/or MAP sensor connection

Have you connected the engine chassis to the negative of the battery?

Have you connect the green wire labeled GND to the negative of the battery or to the chassis?

All grounds must be shared with the negative terminal of the 12V battery.

6**. No spark?**

Do you use the stock ignition system or use HAE ECU to control the ignition?

If stock ignition system, check your stock CDI, coil, and spark plug.

If use HAE ECU to control the ignition, please check the installation of HAE CDI and Coil, and make sure the high voltage cable is fully plugged to the spark plug. Please see the installation manual for details.

If you are using a multi-tooth trigger wheel, and an HAE VRS sensor, please double check the installation, and the setting of the trigger wheel in the HAECAL software. (Details are in the later chapter).

7. No fuel flow?

When the fuel pump is running, (with key on), do you see fuel flowing in the fuel line? Are there bubbles in the fuel hose? Is the pressure regulator installed correctly?

Can you see the fuel flowing back to the tank in the return line after a couple of times of key-cycles?

One way to check fuel supply system is to unplug the high pressure fuel line to the fuel injector, point it to a fuel bottle, and turn on the power; you should see fuel sprout out of the high pressure line.

8. Is the fuel pressure enough?

This is a frequent issue mostly because of the low battery, or incorrect installation of the fuel supply system. A lot of installation problems end up as the low fuel pressure. The best way to trouble shooting the fuel supply system is to install a fuel pressure gauge, which can be bought in any auto parts store.

When the power is on, the fuel pressure should reach 3 bar or 43 psi normally. If the pressure is significantly low, then check your battery voltage, or fuel supply system.

9. Is the idle air enough?

You may not have enough idle air, to start the engine. Try crack-opening the throttle, like 2%-5% throttle, and then crank the engine. Does it fire?

10. Is there enough starting fuel?

The start fuel may not be enough, give a quick enrichment by using “VAL_fFIApp” (see Section 2.3 for details) ?

11. Log data and send us.

“I have checked all the above items, and I think they are all normal, but I still can’t start the engine”.

Please log some data with HAECAL and send us; and we will help you to trouble shoot. Note, when logging data for start issue, follow these steps: Key on → connect HAECAL to ECU → start recording → crank engine → engine not started → stop recording → find the data log files → send by email.

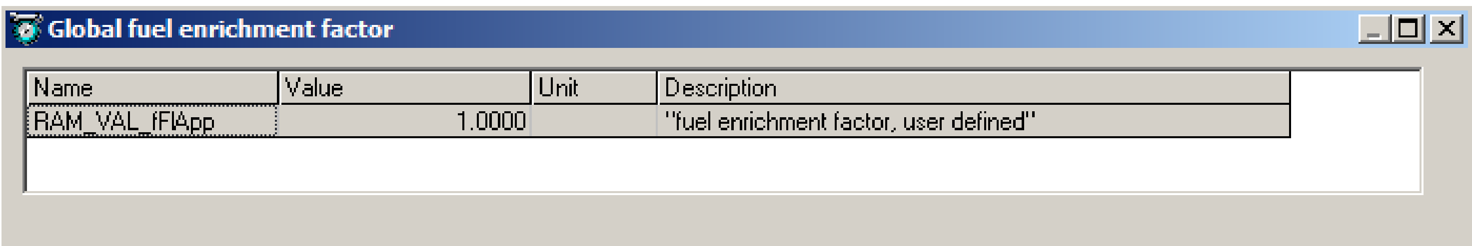
Some engines have 2 pulses per revolution, and some engines have some unique pulses like negative pulses only. All these need to be identified to have the correct CKP signal. One way is to log data with HAECAL and send to us.

It is also helpful to send us some pictures of your EFI installation, especially on how you connect the loose end wires.

2.2 Global fuel tuning

VAL_fFlApp (RAM_VAL_fFlApp) - "global fuel enrichment factor, user defined"

In HAECAL:
Menu → Calibrations → Fuel system → Global fuel enrichment factor →
VAL_fFlApp = 1.0 (default).



It's a global enrichment factor multiplied on the base fuel, meaning, if you change it to 1.5, you'll get 1.5 times of the fuel everywhere. And it applies to all operating conditions (start, warm-up, and steady-state, transient). It has a range of 0-4.0.

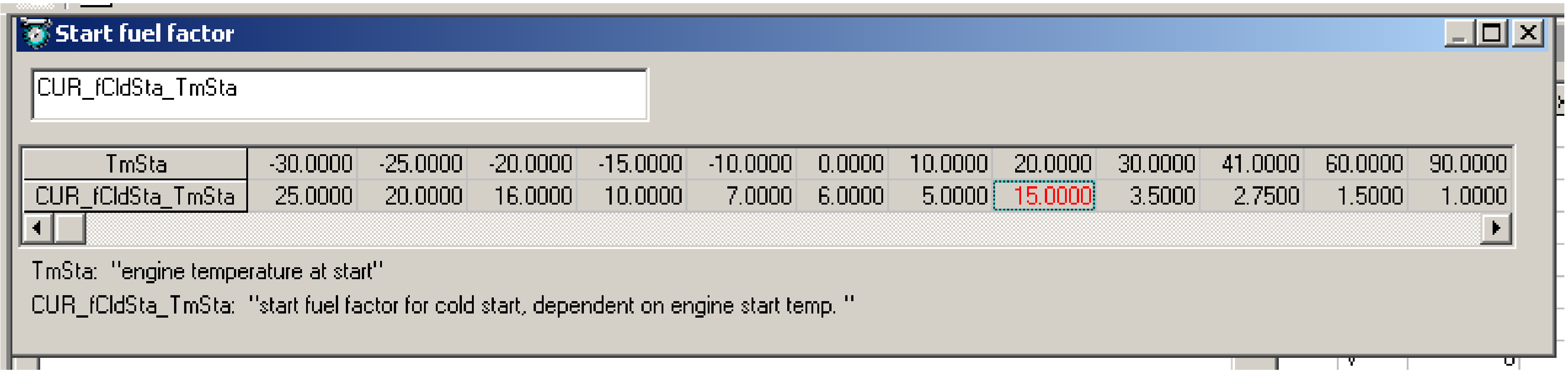
This factor is only supposed to be used temporarily. It's kind of "quick and dirty" fix just for you to fire the engine up and not to stall. The change should be removed once you know the system better, and tune the engine with appropriate parameters.

Certainly you can en-lean the fuel global-wise, by setting it to smaller than 1.0.

2.3 Start fuel factor

CUR_fCldSta_TmSta - "start fuel factor for cold start, dependent on engine start temperature."

In HAECAL:
Menu → calibrations → fuel system → start fuel factor →



This is probably the most important tuning parameter for engine start.

1) Start fuel can NOT be self-tuned by the ECU. Self-tuning is only possible in the "close-loop" control, which is only active after the engine is fully warmed up. ECU cannot learn the engine during the start.
2) Every engine can be different. The start fuel could be different from one to the other. Especially if you have modified the intake manifold or added an adaptor for the throttle, or simply because your engine is different than ours, the base calibration engine, which is a 125cc GY6 engine. The initial wall wetting and the amount of lost fuel during start could be significantly different from engine to engine.

3) Good news, we tried to calibrate the start fuel to be a little rich, and to cover more engines, and to make your initial fire-up successful. So most likely, you can start your engine after a few tries. Yet, you could end up tune your start fuel by your-self. Because this part can only be done via trials.

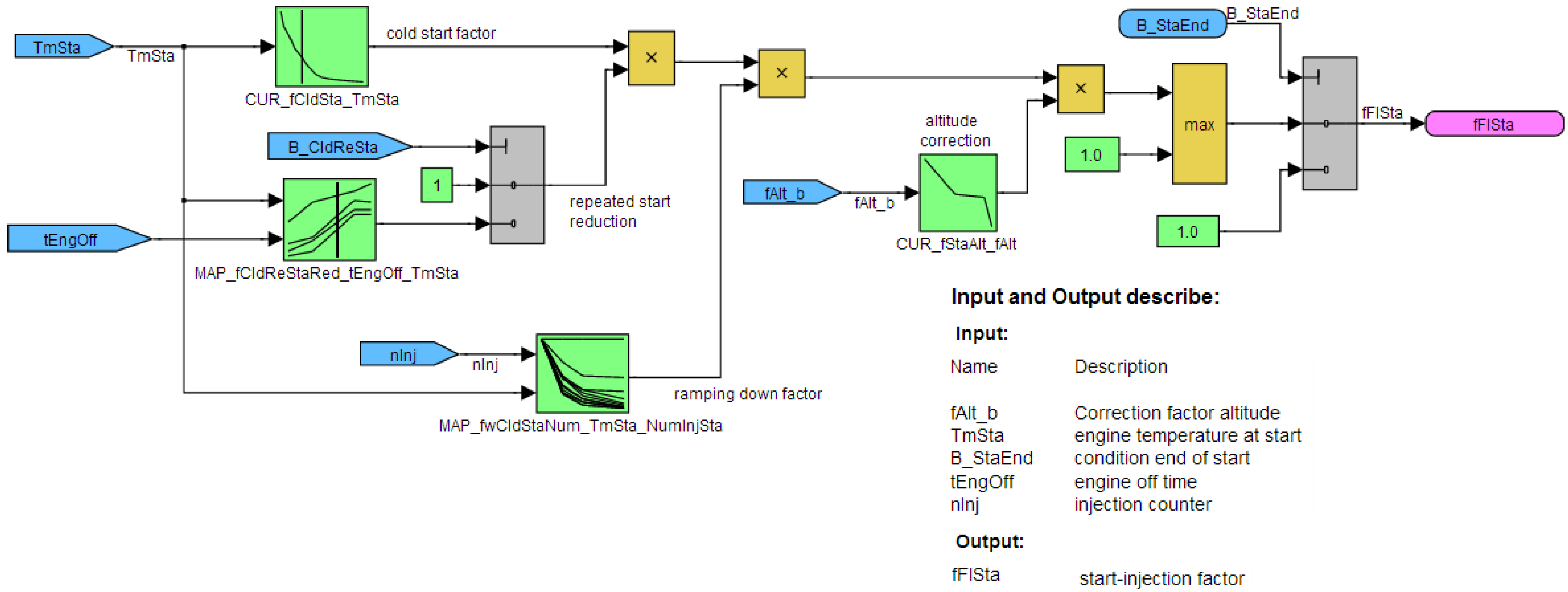
4) Tuning tips: read your engine temperature before start, "Tm", in HAECAL, locates the closest break point of "TmSta" in the table, and change the associated start fuel factor value. Try to start the engine, after a few trials; you identify the best value for that temperature. Then you can apply the similar changes to the neighbor points.

5) Start fuel only applies during start. Once the start of engine is ended, indicated by that the engine speed (N) is greater than 1000rpm; the start fuel will be inactive. The after-start fuel and warm-up fuel will take over.

Example: Early in the morning, Key on, connect your laptop, read the Tm (or ECT), in HAECAL, say, it's 9 deg C. Then find the cell in the table where the break point is the closest, 10C in this case; and the existing value is 5.0; meaning, 5 times enrichment is the pre-set.
Most likely you are starting a little leaner at that temperature; so you would increase the value to 5.5; and then "burn to ECU".
Then you crank the engine again; see if the starting is better. If not, you may have to wait for next day morning, and increase it to 6.0; and try it again.
You may end up over-enriching the start fuel, which causes black smokes, and strong smell, and even engine flooding. In that case, you may go to the other direction, reducing the enrichment factor.

The challenge is that there is no direct measure which tells you whether it's too rich or too lean. It's really based on your feeling and experiments.

Start control strategy:



Note: If your are in the high altitude, there is alo altitude correct factor CUR_fStaAlt_fAlt, need be turnd to get a easy start.

2.4 After-Start and Warm-up fuel factor

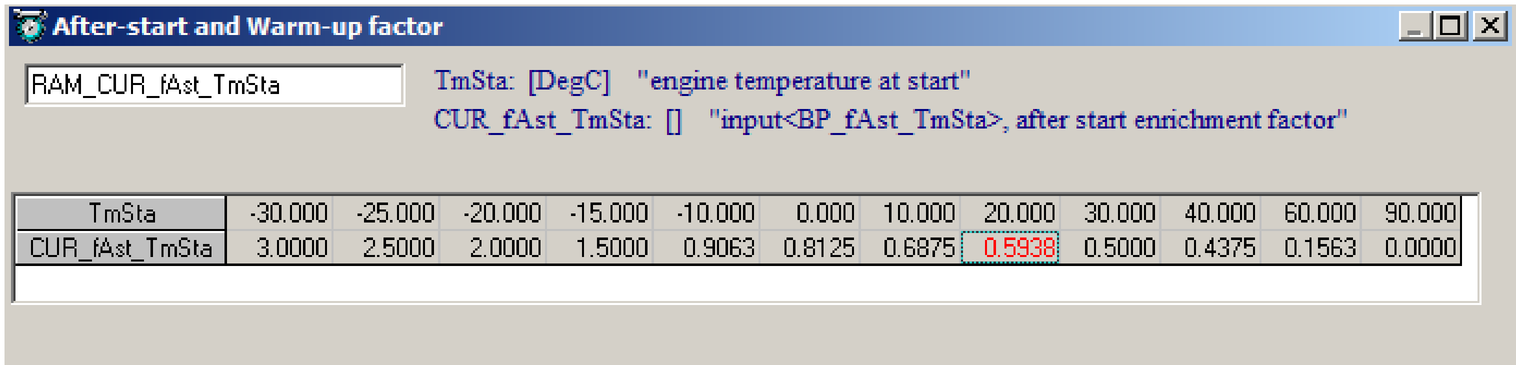
CUR_fAst_TmSta - "after-start fuel enrichment factor, dependent on engine start temp."

The output of this Curve will be added on top of 1.0 as the after-start factor.

It will be ramping down to 1.0 with the engine running time increased after start.

In HAECAL:

Menu → calibrations → fuel system → after start and warm-up factor →



TmSta	-30.000	-25.000	-20.000	-15.000	-10.000	0.000	10.000	20.000	30.000	40.000	60.000	90.000
CUR_fAst_TmSta	3.0000	2.5000	2.0000	1.5000	0.9063	0.8125	0.6875	0.5938	0.5000	0.4375	0.1563	0.0000

Note: **After-start factor is actually = 1 + look-up table value.**

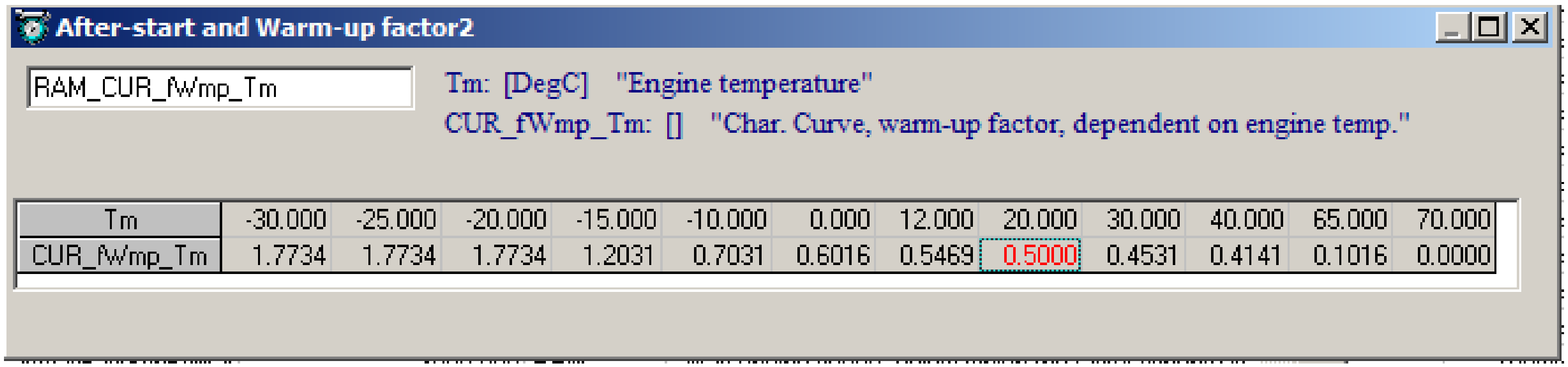
For example, if you set the table value as 0.6; the after-start factor is = 1.6

Why? It is easy for software implementation.

CUR_fWmp_Tm - "Warm-up fuel enrichment factor, dependent on engine temp."

In HAECAL:

Menu → calibrations → fuel system → after start and warm-up factor →



After-start and Warm-up factor2

RAM_CUR_fWmp_Tm

Tm: [DegC] "Engine temperature"
CUR_fWmp_Tm: [] "Char. Curve, warm-up factor, dependent on engine temp."

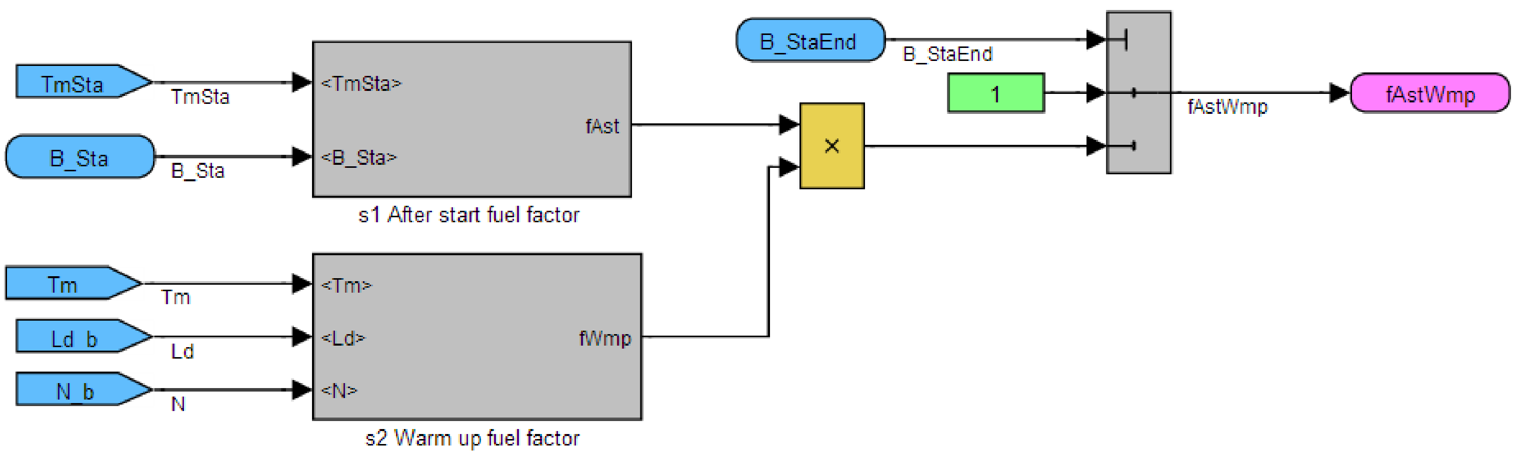
Tm	-30.000	-25.000	-20.000	-15.000	-10.000	0.000	12.000	20.000	30.000	40.000	65.000	70.000
CUR_fWmp_Tm	1.7734	1.7734	1.7734	1.2031	0.7031	0.6016	0.5469	0.5000	0.4531	0.4141	0.1016	0.0000

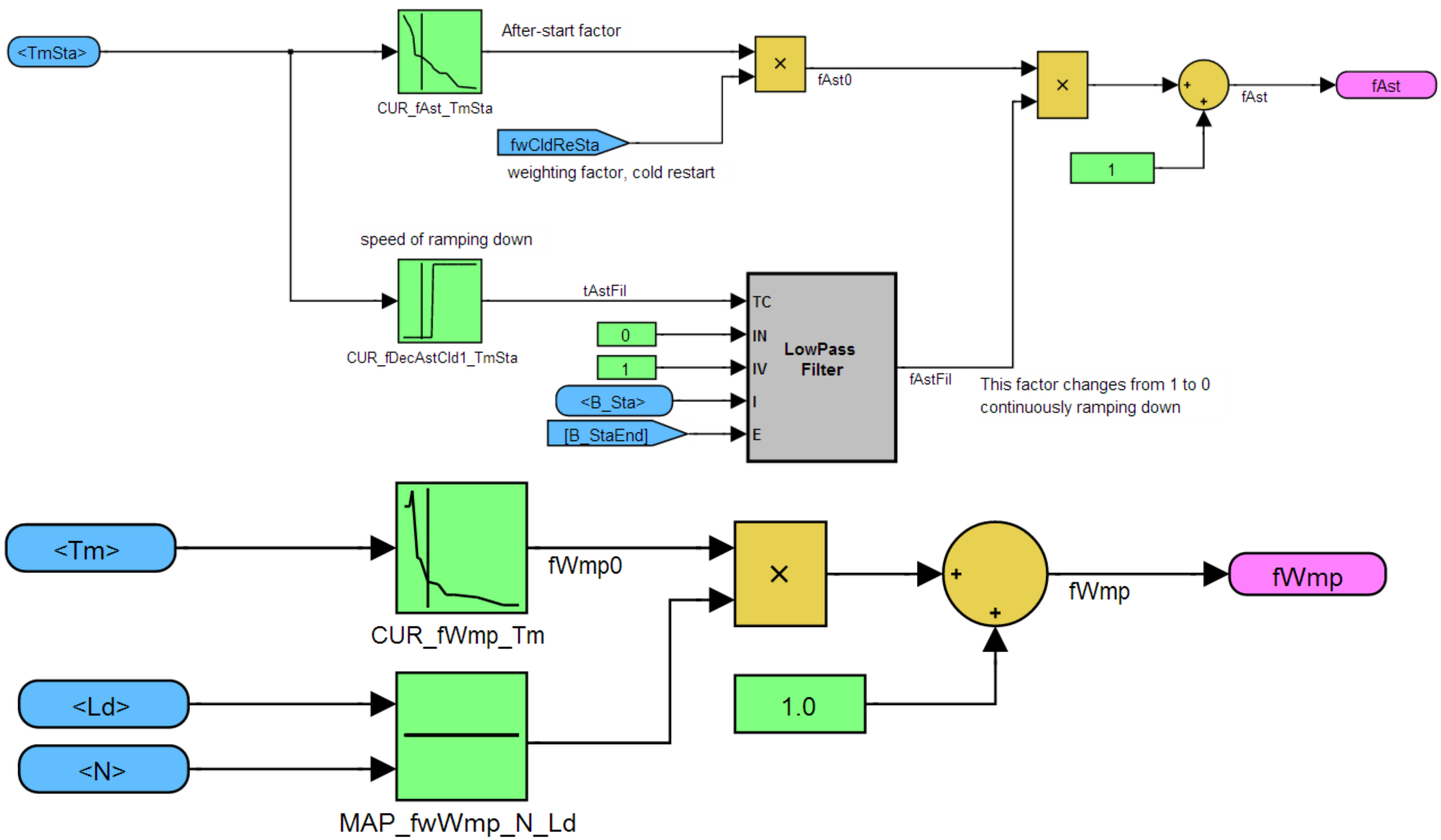
Same: warm-up factor is actually = 1 + look-up table value.

For example, if you set the table value as 0.6; the warm-up factor is = 1.6

Why? It is easy for software implementation.

Start control strategy:





Note: The finial factor is fAstWamp=fAst * fWmp, so you need tune the both variables after start.

Q: What is the difference between after-start fuel and warm-up fuel?

After-start fuel is carrying over the start-fuel enrichment, and quickly ramping down to 1.0. It is dependent on the engine start temperature only.

Warm-up fuel is dynamically adjusting the fuel dependent on how fast the engine temperature warms up. It also takes into account of the impact of engine load and speed.

Q: Tips for after-start and warm-up fuel

Too complicated? Good news! You do NOT have to tune these 2 factors in most cases, because:

1) They are normalized for most engines, close to good for most engines.
2) They are designed for emission reduction purposes. For after-market conversions, pre-set data are often good enough.
3) Exceptions #1: if you notice the engine clearly fires for the start (during first couple of revolutions), and the RPM rises quickly to more than 800rpm, but then stalls immediately; this tells you that the after-start fuel is not enough. You will need to increase the after-start fuel factor.

4) Exception #2: if you notice that the engine starts, and idles at relatively good rpm, like 1800rpm, but then slowly it dies with engine warms up. This tells you that you may need to enrich the warm-up fuel to keep engine running after the start-up and after-start fuel factors ramping out.

Q: Engine starts, and idling, and can stay running, but kind of rough. What's next?

As long as you can start the engine and let the engine running and not stalling. You are fine for now. Engine will warm itself up, and quickly the warm-up fuel factor will ramp down to 1.0 (neutral). Next we will make sure the engine can be run in open-loop mode comparatively stable.

Note: once engine can stay running, then wait for engine warmed up, and tune the base fuel mapping first (VE table, Load table, see later chapters). Don't do fine-tuning on the start fuel factor, after-start, and warm-up fuel factors before you have a good base fuel mapping, because these factors are nothing but a multiplicative factors on the base fuel.

More on Start, After-Start, Warm-up process:

See below the 2 graphs of start and warm-up process from a real scooter engine:

Descriptions:

1) The engine starts at 30C degree; and idles up to 70C degree (where the warm-up process is ended);
2) Start fuel factor starts at 3.2 (fFlSta), for example; and drops to 1 as soon as the RPM > 1000rpm (start ended); this only takes a few seconds;
3) After-start fuel factor starts to take over when start is ended (fFlSta =1); and for example, it starts at 1.23; and quickly ramps down to 1; and this takes about 1 minute (or 60s); This factor is dependent on the engine start temperature only (30C);
4) Warm-up fuel factor happens simultaneously with the after-start factor, but it ramps down very slowly; it starts at 1.4; and takes about 6 minutes (360s) to ramp down to 1, when ECT is 70C deg. This factor depends on real-time engine temperature (changing from 30C to 70C).
5) The close-loop fuel only happens 3 minutes after start, where the O2S signal starts to oscillate.

6) Overall fuel enrichment factor, fPreCtl, is the results of all 3 factors, start factor, after-start factor, warm-up factor:

   fPreCtl = fFlSta * VAL_FlApp (during start)

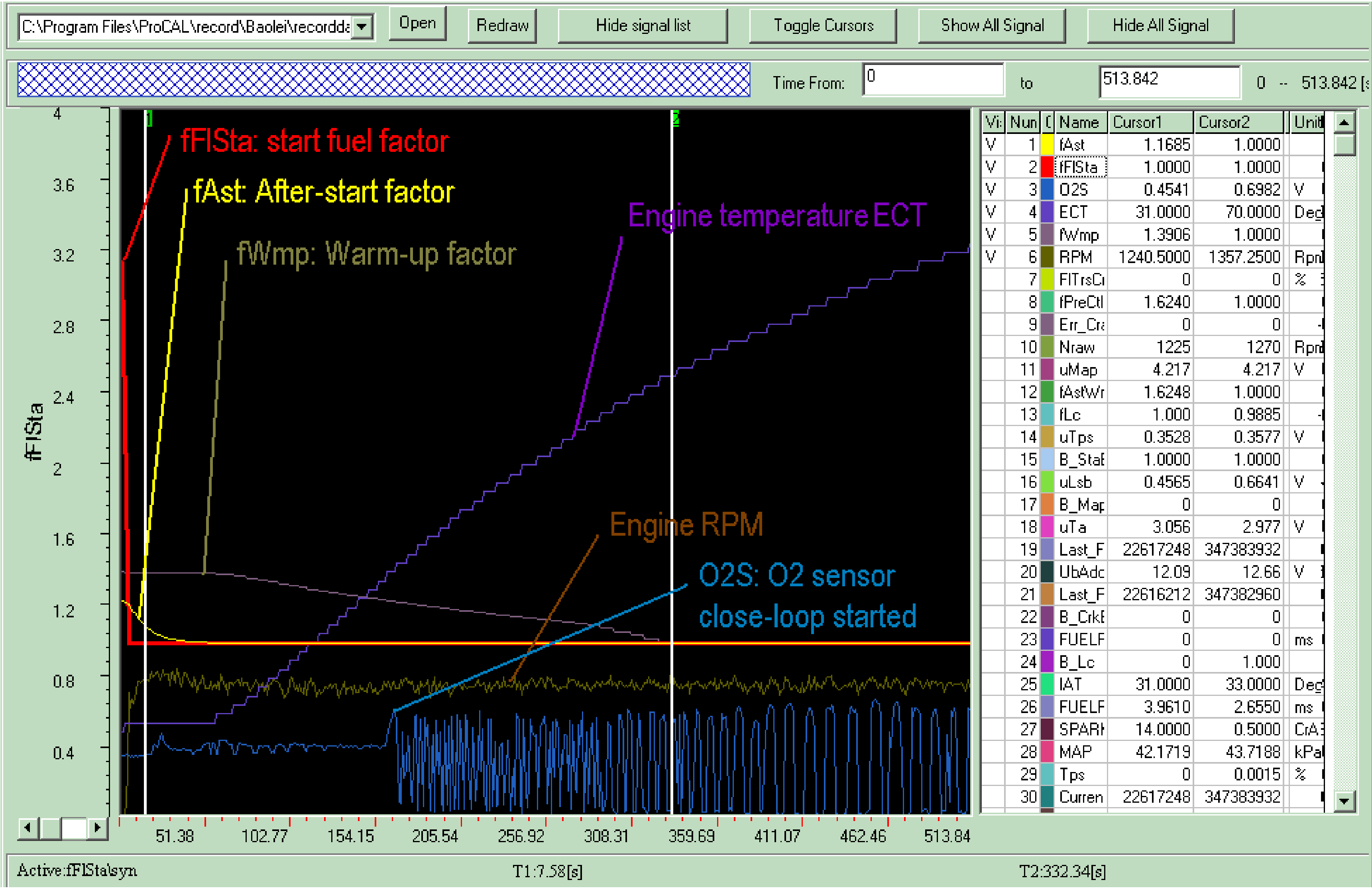
   fPreCtl = fAst * fWmp* VAL_FlApp (during after-start, warm-up)

   Note: **During start, fAst and fWmp are not used; meaning they are equivalent to 1; VAL_FlApp is the global enrichment factor.**

   fPreCtl is the overall enrichment factor.

   It is the green line in the second graph.

   Note: **In your logged data, you can see this variable plotted. It tells you how your starting-warm-up fuel is controlled.**


C:\Program Files\ProCAL\record\Baolei\recordd
Open
Redraw
Hide signal list
Toggle Cursors
Show All Signal
Hide All Signal
Time From:
to
fFISta: start fuel factor
fAst: After-start factor
fWmp: Warm-up factor
Engine temperature ECT
Engine RPM
O2S: O2 sensor
close-loop started
fFISta
Active:fFISta\syn
T1:7.58[s]
T2:332.34[s]


C:\Program Files\ProCAL\record\Baolei\recordda
Open
Redraw
Hide signal list
Toggle Cursors
Show All Signal
Hide All Signal
Time From: 2.146 to 25
0 -- 513.842
fFlSta : start factor
fPreCtl = fFlSta * fAst * fWmp
fAst: After start factor
fWmp: warm-up factor
RPM
fPreCtl
4 3.6 3.2 2.8 2.4 2 1.6 1.2 0.8 0.4
2.15 4.43 6.72 9.00 11.29 13.57 15.86 18.14 20.43 22.71
Vi Num Name Cursor1 Cursor2 Units
V 1 fAst 1.2344 1.1099
V 2 fFlSta 3.1445 1.0000
V 3 O2S 0.4541 0.4541 V
V 4 ECT 31.0000 31.0000 DegC
V 5 fWmp 1.3906 1.3906
V 6 RPM 464.0000 1304.2500 Rpm
V 7 fPreCtl 3.1445 1.5449
8 fLc 1.000 1.000
9 FlTrsCr 0 0 %
10 Err_Cra 0 0
11 Nraw 464.0 1327 Rpm
12 uMap 3.177 3.129 V
13 uLsb 0.4565 0.4553 V
14 fAsfWr 1.0000 1.5457
15 UbAdc 10.18 12.09 V
16 B_Stat 0 1.0000
17 Last_F 18178843 27541973
18 B_Map 0 0
19 uPot 4.970 4.964 V
20 Last_F 18180399 27542925
21 uTps 0.3467 0.3540 V
22 uTa 3.059 3.059 V
23 Last_F 18069050 27497416
24 Tps 0 0 %
25 B_Lc 0 0
26 IAT 31.0000 31.0000 DegC
27 FUELF 11.1970 3.6910 ms
28 SPARI 9.0000 15.0000 CrA
29 FUELF 0 0 ms
30 MAP 57.0547 42.2461 kPa
Active:fPreCtl\100ms
T1:3.21[s]
T2:12.49[s]

Chapter 3 Engine idle tuning

3.1 Target Idle Speed

If your engine's idle RPM is not stable, or too high, or too low, it may be because of the idle air, so you need to adjust it as the below methods.

Certainly, you must make sure the whole air intake system is air-tight first!

Past experience shows that during initial installations, some users forgot to make sure the air-tight of the system, and end up high "idle" RPM, like 4000rpm; and which actually leads to RPM oscillations! Because our ECU is trying to cut the fuel to lower downs the idle RPM.

So if you see an oscillating idle RPM from 4000rpm to 1000rpm that tells you that you should check your intake air system, and find out air leakage!

Note: The idle speed with a cold engine could be lower than the warmed-up engine, especially if you do not use the EFI to control the ignition system.

If your engine idle speed is already high after warmed up, do not make the idle air too high in the cold.

If you notice that you have to hold the throttle to open a little to start the engine and keep the engine running, and as soon as you let the throttle go, the engine stalls. This means you may have too less idle air. This could be the case especially for the big engines, where more idle air is needed than for the small engines.

If you do not have our ECU-controlled CDI to control the ignition angles, you will have comparatively low idle speed at cold start and higher idle speed after warm-up. This is normal.

How do you have a constant idle RPM, regardless of a cold or hot engine? You need to have our ECU-controlled ignition system.

See the chapter of "Ignition Angle Tuning".

3.2 Idle speed adjustment via Mechanical methods

Single cylinder idle air screw adjustment

Our 28mm throttle body comes with an idle screw; if your idle speed is too low, or too high, you can adjust the idle screw to get more or less idle air.

Adjust the idle screw:

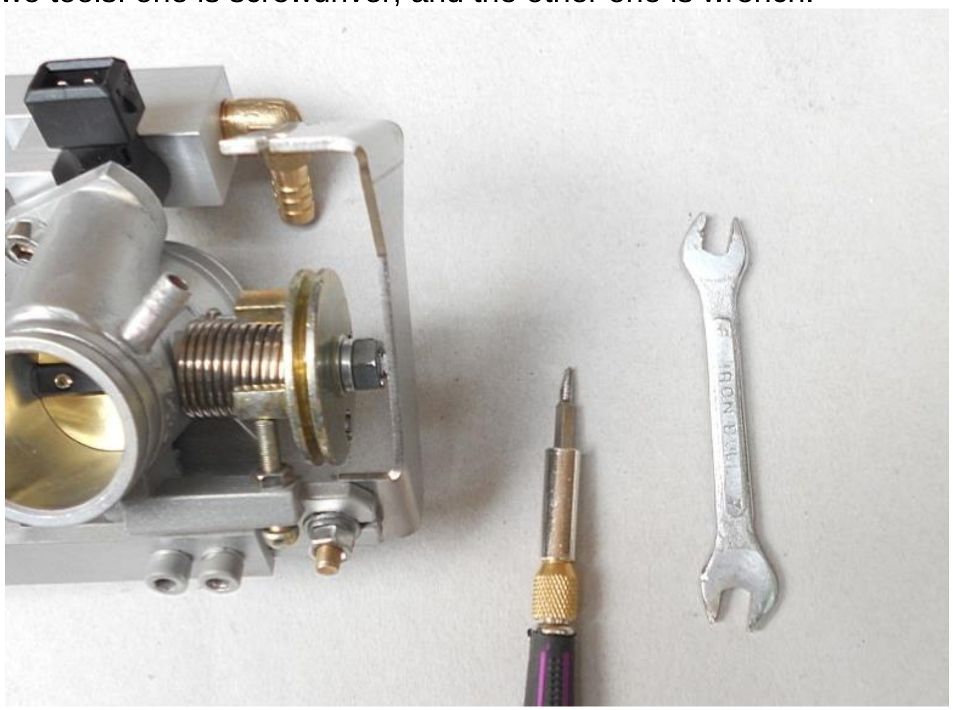
Further mechanical idle Air adjustment

If by only adjusting the idle air screws can't meet your idle air requirements, such as the idle RPM is still too low, you need adjust the mechanical stop position of the throttle valve (this is not preferred, unless you have no other options). The factory setting of the throttle plate is usually optimized. You need to be careful on this mechanical adjustment.

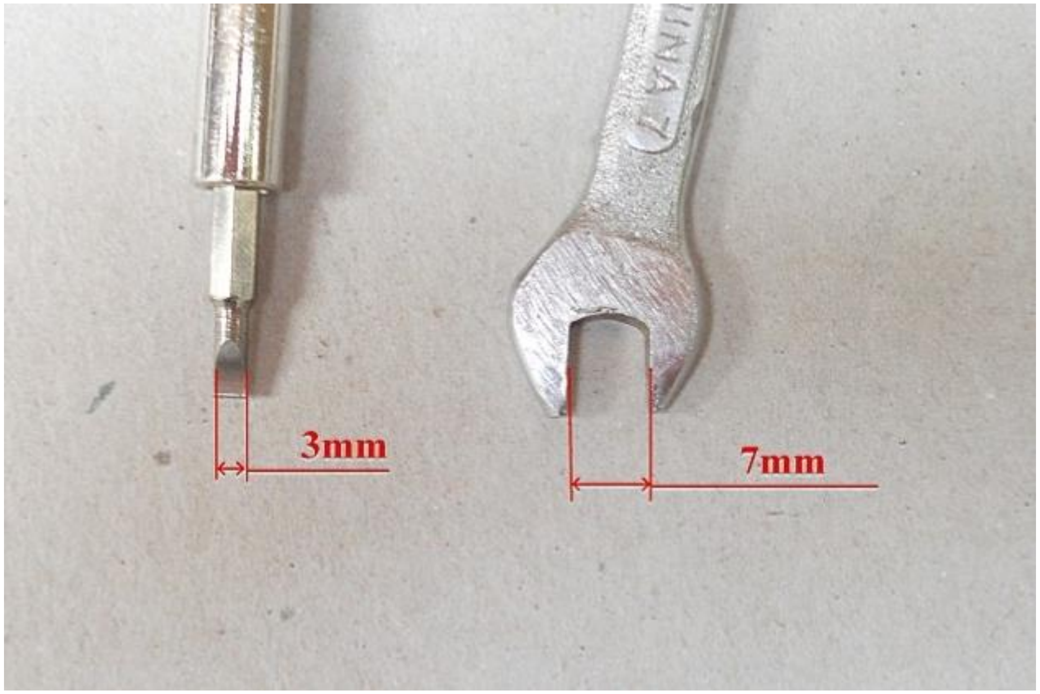
Basically the idea is to open the throttle plate a little tiny bit, so there is more idle leaking air.

The steps are below:

(1) Please prepare two tools: one is screwdriver, and the other one is wrench.

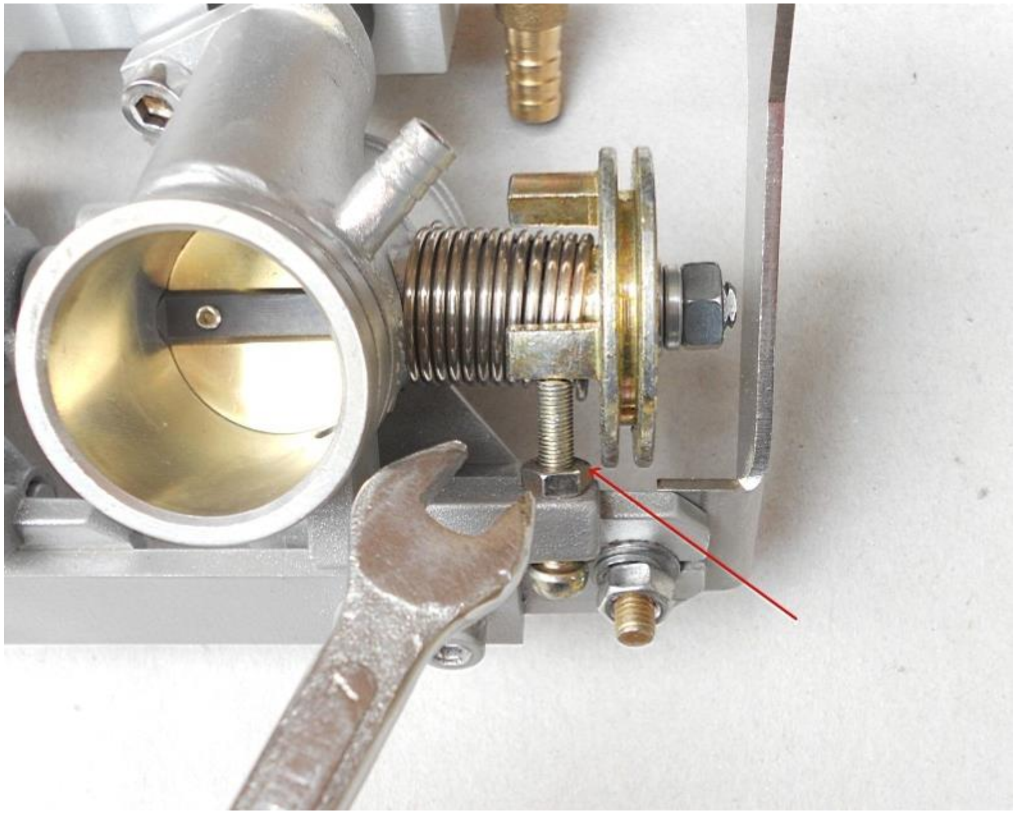


For the size of the each tool, you can see in the below picture.

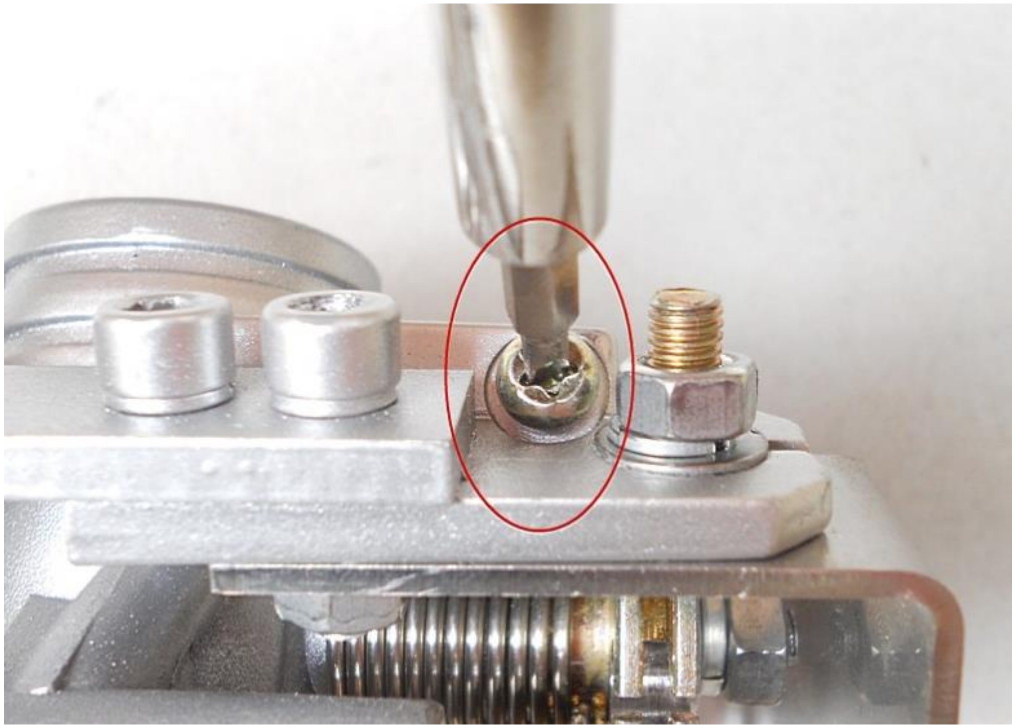


(2) Begin to adjust now, please see the below pictures.

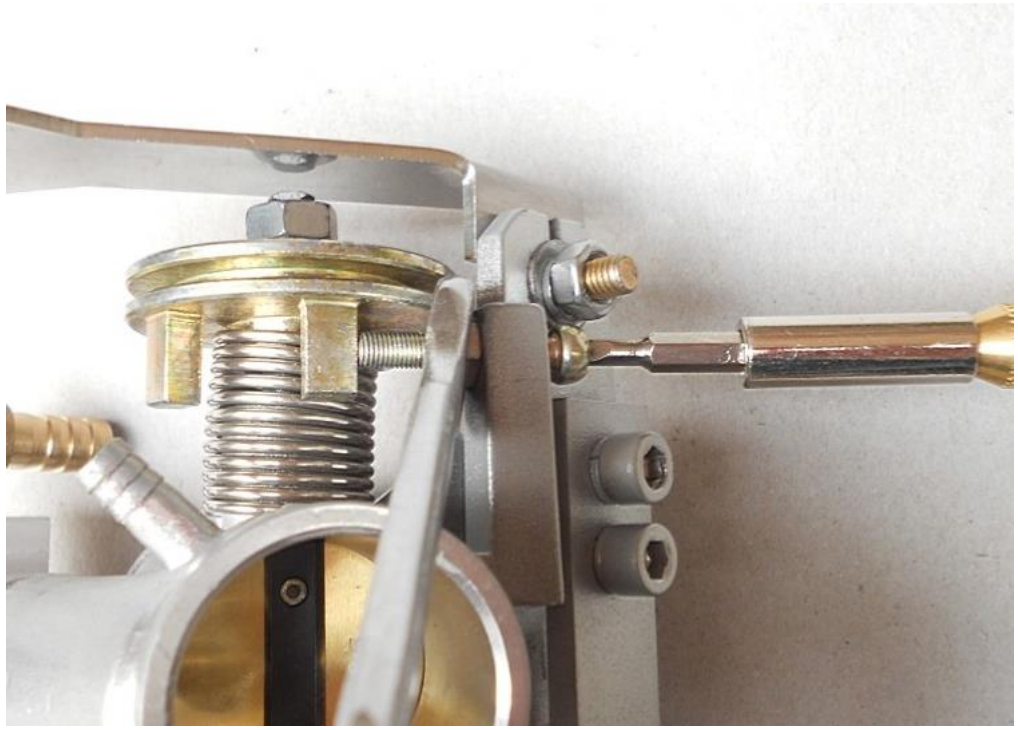
First, loose the nut



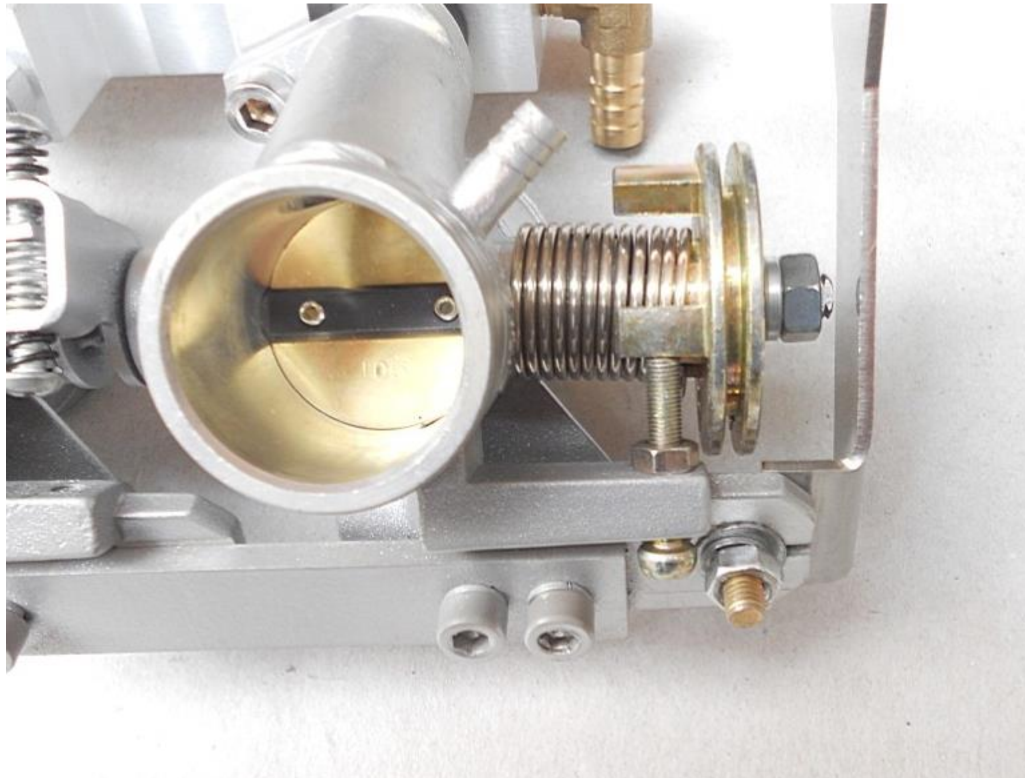
Then, use the screwdriver to turn the screw; if you want more idle air, you need turn the screw in clockwise, and if you want less idle air, you need turn the screw in counter-clockwise.



Last, when you are done adjusting the screw, please tight the nut by using the 2 tools.



Finished



3.3 Electronic Idle Air Control – IAC motor

Our 34mm size and bigger throttle bodies all come with an Idle Air Control (IAC) motor, which is a stepper motor, with 4 wires.

With these throttle bodies, you can set the target idle speed, and ECU will control the stepper motor to reach that speed.

Q: How to tune the idle RPM with an IAC motor?

If your enigne idle RPM is too high, or too low, you need to adjust the desired stepper motor position (MAP_StepPrePos_Tm_N) according to the actual position “StepPos”; which can be logged via HAECAL..

Note: **The stepper motor position has an inverse logic to the idle RPM. By increasing the stepper motor position values in the table, you are reducing the idle air; and idle RPM will be lower.**

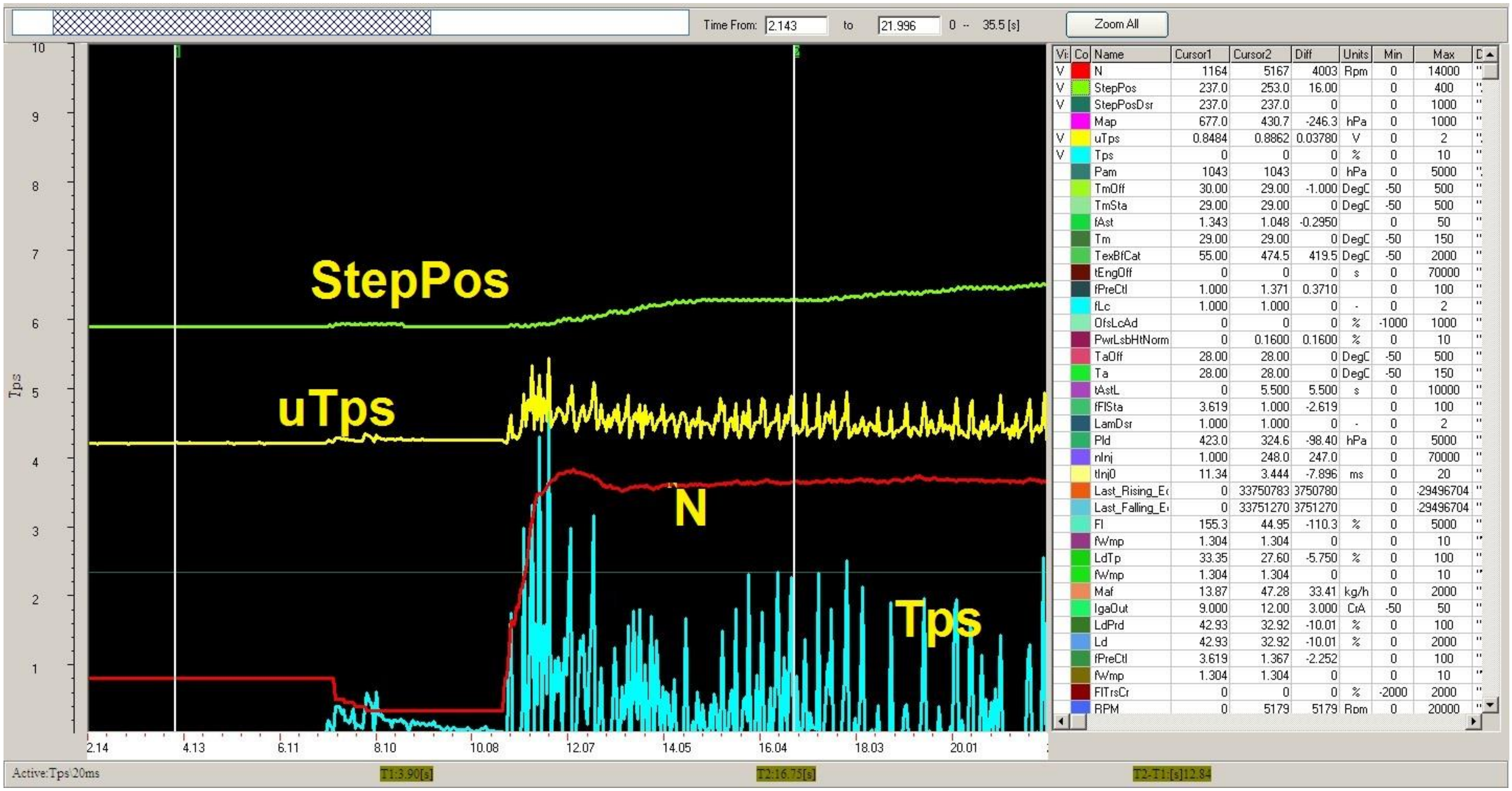
Note: **If your engine idle speed stays higher than your target speed, and no matter what you do with an IAC system, you may have air leak in your intake air system.**

The stepper motor comes with the throttle bodies are designed for certain range of engine displacements (engine cc sizes). If you use these throttle bodies (34mm or 42mm or 50mm or 55mm) on too big engines or too small engines, the idle motor may reach its physical max/min limits, and the idle air cannot be adjusted anymore by the IAC. You could also end up with too high or too low idle RPMs.

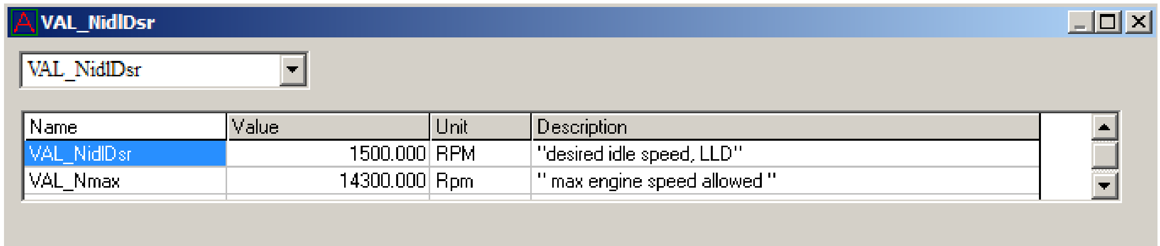
Note: **IAC based idle controls only work if your TPS sensor reads 0%; meaning throttle body fully closed. Sometime your throttle cable is stuck and throttle plate is not fully closed.**

If your engine idle RPM is not stable, or stays high for a long time, and it still can't go down to the desired RPM, then check your TPS signal. Maybe the TPS sensor signal is too noisy. Or the value of TPS is not 0 when engine runs in idle. TPS signal must have a stable value at 0 when engine is running in idle. If this is not the case, you better check the TPS sensor installation and also the throttle cable returning spring, etc.; make sure mechanically the TPS position is fixed at 0 when in idle. Also check the TPS connector and wire connections, make sure they are secured.

Below is a plot of noisy TPS signal example.



How to set the target idle speed in HAECAL?


VAL_NidlDsr

VAL_NidlDsr

Name	Value	Unit	Description
VAL_NidlDsr	1500.000	RPM	"desired idle speed, LLD"
VAL_Nmax	14300.000	Rpm	" max engine speed allowed "

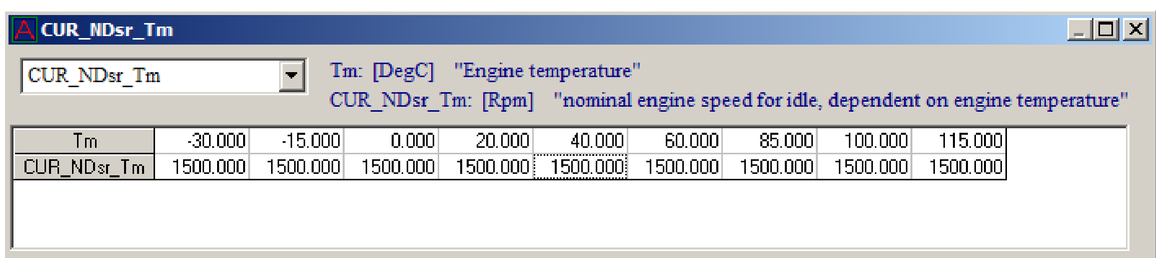
VAL_NidlDsr - "Desired idle speed"

In HAECAL:

Menu → Advanced, add this variable.

By default, it is set as 1500 RPM, if your engine has a different idle RPM, you need to change it.

For newer software version, the desired idle speed is dependent on the engine temperature. You will need to calibrate the CURVE: **CUR_NidlDsr_Tm**


CUR_NDsr_Tm

CUR_NDsr_Tm

Tm: [DegC] "Engine temperature"

CUR_NDsr_Tm: [Rpm] "nominal engine speed for idle, dependent on engine temperature"

Tm	-30.000	-15.000	0.000	20.000	40.000	60.000	85.000	100.000	115.000
CUR_NDsr_Tm	1500.000	1500.000	1500.000	1500.000	1500.000	1500.000	1500.000	1500.000	1500.000

In HAECAL:

Menu → Advanced, add this variable.

Our idle speed controls via IAC motor is a close-loop control. It reads the actually engine RPM and compares it to the target idle speed, and adjust the idle air correspondingly, and maintain the target the idle speed.

Note: The idle speed certainly cannot be set too high or too low. For example, if your engine idle RPM (OEM setting) is 1500 RPM, and you change it to 3000RPM, it will not work, and you may have an oscillating RPM.

If you set the idle RPM too low, like 1000rpm, it may end up stall frequently.

How to calibrate the stepper motor position to get a desired idle RPM? And how to fix the issue after the engine started, like, the idle RPM is too high or too low?

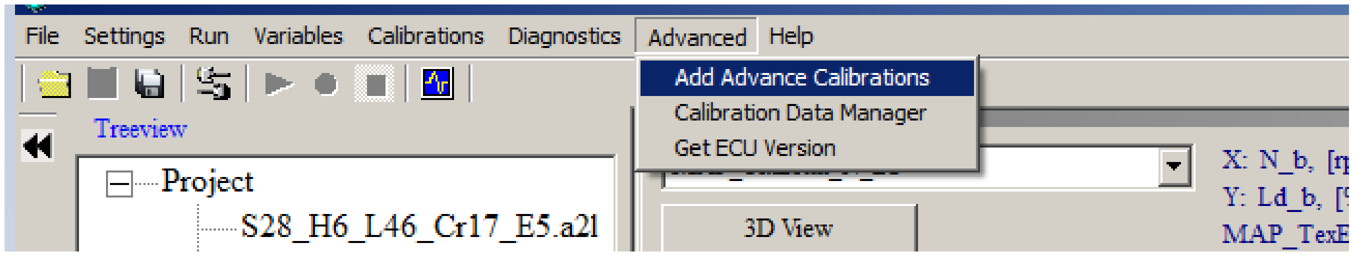
You need to tune the stepper motor position, MAP_StepPrePos_Tm_N, according to the following method.

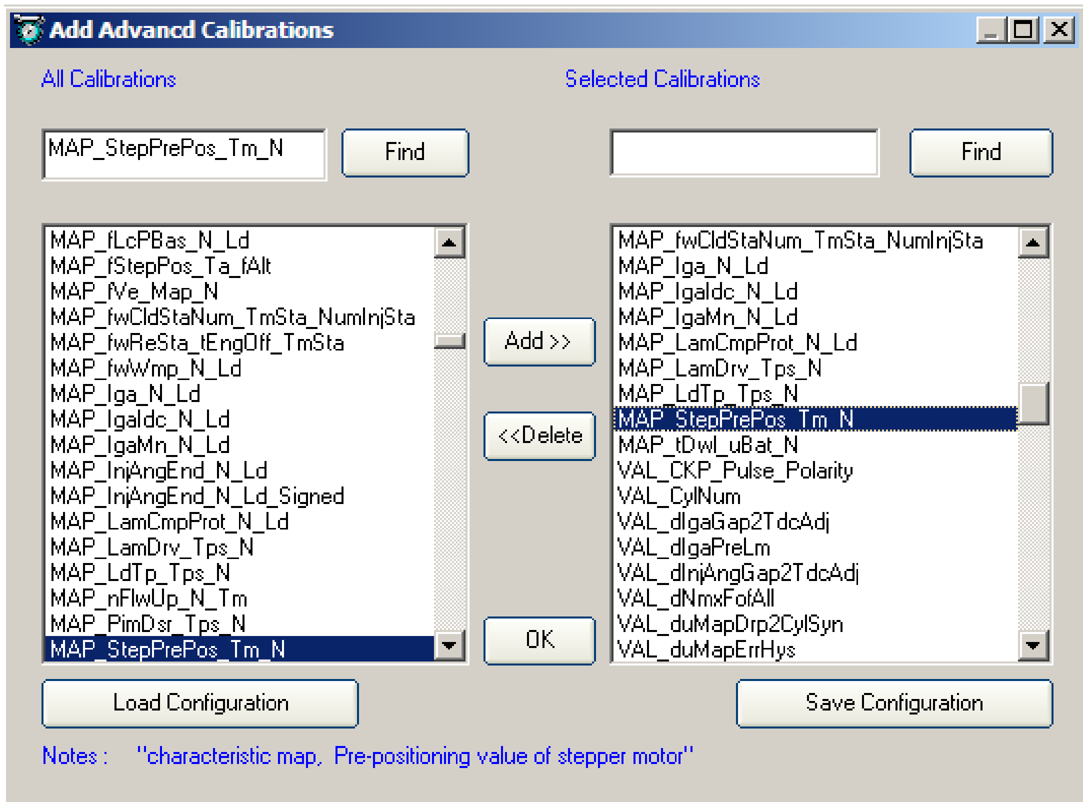
Note: the default stepper motor position is intentionally over- tuned, meaning, and more idle air than necessary. This is to make the initial start easier. If the idle RPM is too high after start, you need increase the stepper position value to reduce the idle air;

If the idle RPM is too low after start, you need reduce the stepper position value to get more idle air.

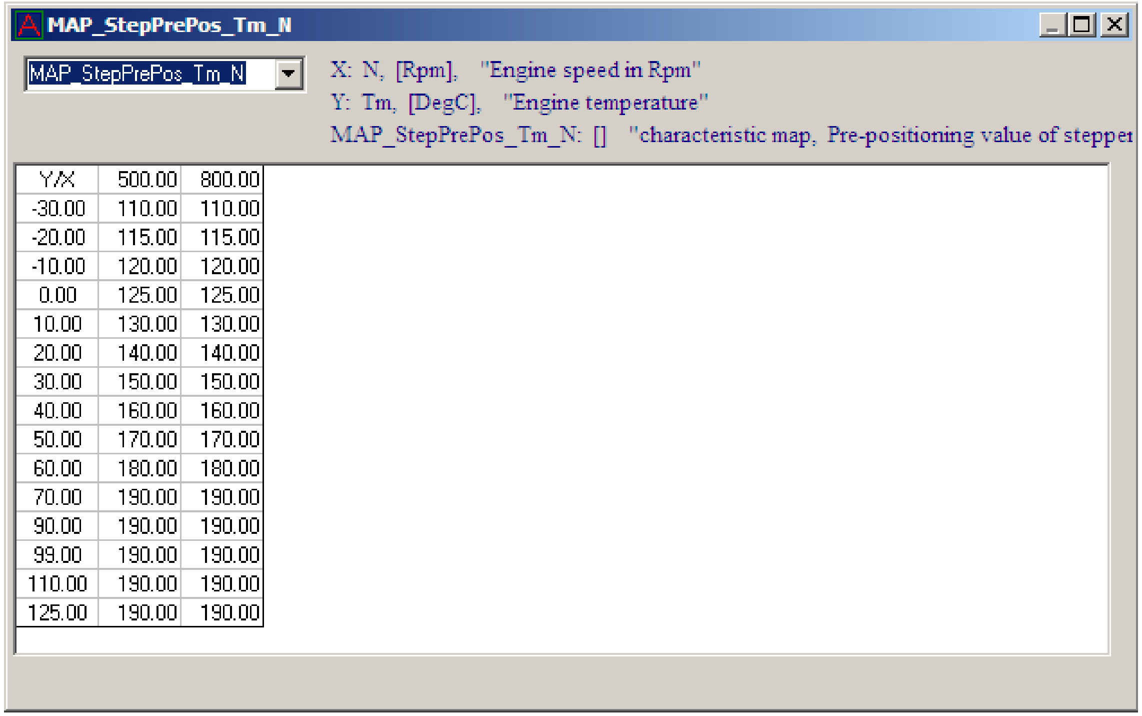
Adjust idle motor desired position; you need to do some advanced tuning, follow the below steps.

(1) Add advance calibration: MAP_StepPrePos_Tm_N


File
Settings
Run
Variables
Calibrations
Diagnostics
Advanced
Help
Add Advance Calibrations
Calibration Data Manager
Get ECU Version
Treeview
Project
S28_H6_L46_Cr17_E5.a2l
3D View

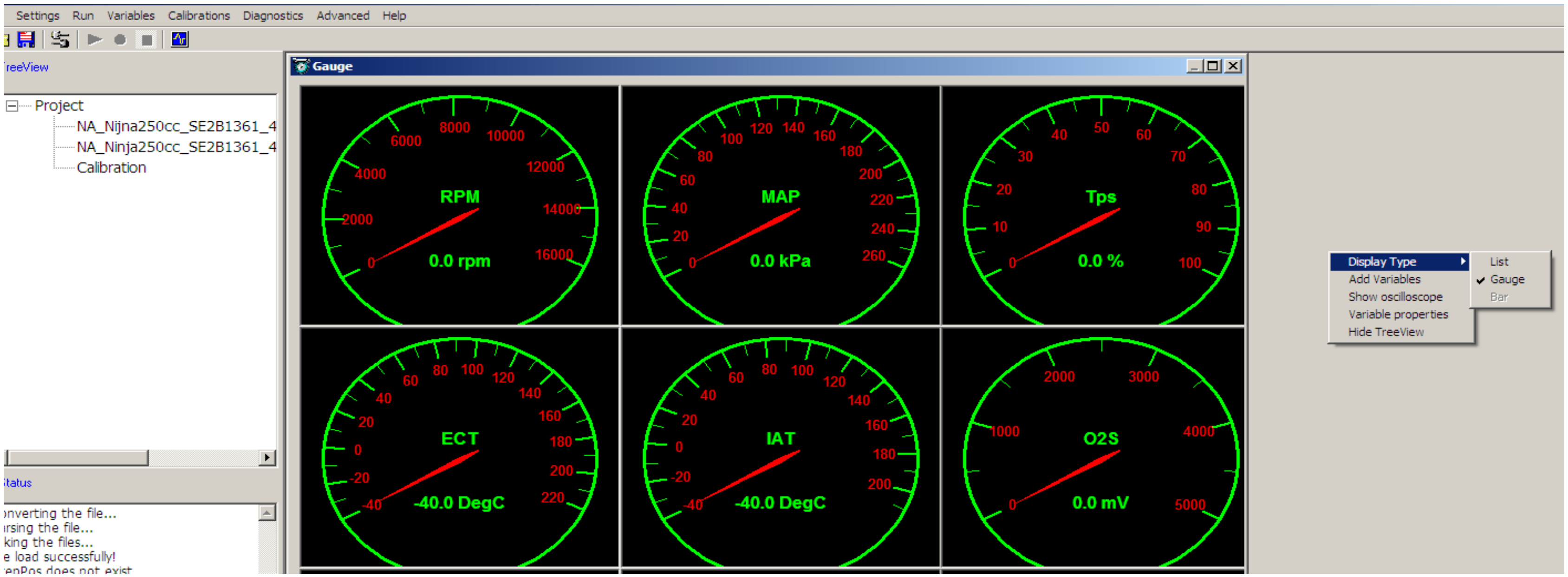

Add Advancd Calibrations
All Calibrations
Selected Calibrations
MAP_StepPrePos_Tm_N
Find
Find
MAP_fLcPBas_N_Ld
MAP_fStepPos_Ta_fAlt
MAP_fVe_Map_N
MAP_fwCldStaNum_TmSta_NumInjSta
MAP_fwReSta_tEngOff_TmSta
MAP_fwWmp_N_Ld
MAP_Iga_N_Ld
MAP_Igaldc_N_Ld
MAP_IgaMn_N_Ld
MAP_InjAngEnd_N_Ld
MAP_InjAngEnd_N_Ld_Signed
MAP_LamCmpProt_N_Ld
MAP_LamDrv_Tps_N
MAP_LdTp_Tps_N
MAP_nFlwUp_N_Tm
MAP_PimDsr_Tps_N
MAP_StepPrePos_Tm_N
Add >>
<<Delete
OK
MAP_fwCldStaNum_TmSta_NumInjSta
MAP_Iga_N_Ld
MAP_Igaldc_N_Ld
MAP_IgaMn_N_Ld
MAP_LamCmpProt_N_Ld
MAP_LamDrv_Tps_N
MAP_LdTp_Tps_N
MAP_StepPrePos_Tm_N
MAP_tDwl_uBat_N
VAL_CKP_Pulse_Polarity
VAL_CylNum
VAL_dIgaGap2TdcAdj
VAL_dIgaPreLm
VAL_dInjAngGap2TdcAdj
VAL_dNmxFofAll
VAL_duMapDrp2CylSyn
VAL_duMapErrHys
Load Configuration
Save Configuration
Notes : "characteristic map, Pre-positioning value of stepper motor"

Find the variable: MAP_StepPrePos_Tm_N on the left side, then click “Add>>”. The variable appears on the right window. Click "OK"

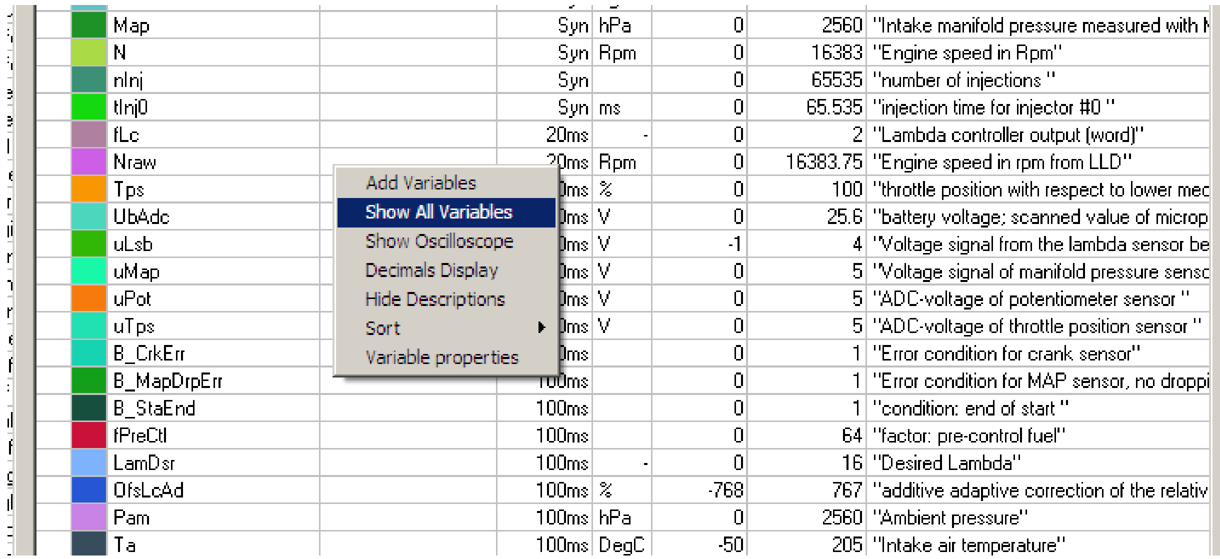


Y/X	500.00	800.00
-30.00	110.00	110.00
-20.00	115.00	115.00
-10.00	120.00	120.00
0.00	125.00	125.00
10.00	130.00	130.00
20.00	140.00	140.00
30.00	150.00	150.00
40.00	160.00	160.00
50.00	170.00	170.00
60.00	180.00	180.00
70.00	190.00	190.00
90.00	190.00	190.00
99.00	190.00	190.00
110.00	190.00	190.00
125.00	190.00	190.00

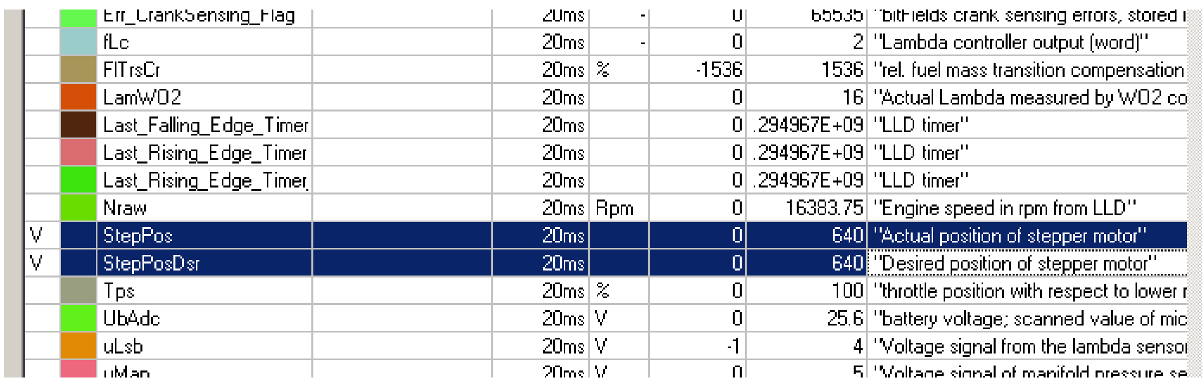
(2) Choose "list": In the interface Right Click, then click "Display type->List"



In the list view, right-click, to Click "Show All Variables"


Map Syn hPa 0 2560 "Intake manifold pressure measured with I
N Syn Rpm 0 16383 "Engine speed in Rpm"
nInj Syn 0 65535 "number of injections "
tInj0 Syn ms 0 65.535 "injection time for injector #0 "
fLc 20ms - 0 2 "Lambda controller output (word)"
Nraw 20ms Rpm 0 16383.75 "Engine speed in rpm from LLD"
Tps % 0 100 "throttle position with respect to lower mec
UbAdc V 0 25.6 "battery voltage; scanned value of microp
uLsb V -1 4 "Voltage signal from the lambda sensor be
uMap V 0 5 "Voltage signal of manifold pressure sensc
uPot V 0 5 "ADC-voltage of potentiometer sensor "
uTps V 0 5 "ADC-voltage of throttle position sensor "
B_CrkErr 0 1 "Error condition for crank sensor"
B_MapDrpErr 100ms 0 1 "Error condition for MAP sensor, no droppi
B_StaEnd 100ms 0 1 "condition: end of start "
fPreCtl 100ms 0 64 "factor: pre-control fuel"
LamDsr 100ms - 0 16 "Desired Lambda"
OfsLcAd 100ms % -768 767 "additive adaptive correction of the relativ
Pam 100ms hPa 0 2560 "Ambient pressure"
Ta 100ms DegC -50 205 "Intake air temperature"
Add Variables
Show All Variables
Show Oscilloscope
Decimals Display
Hide Descriptions
Sort
Variable properties

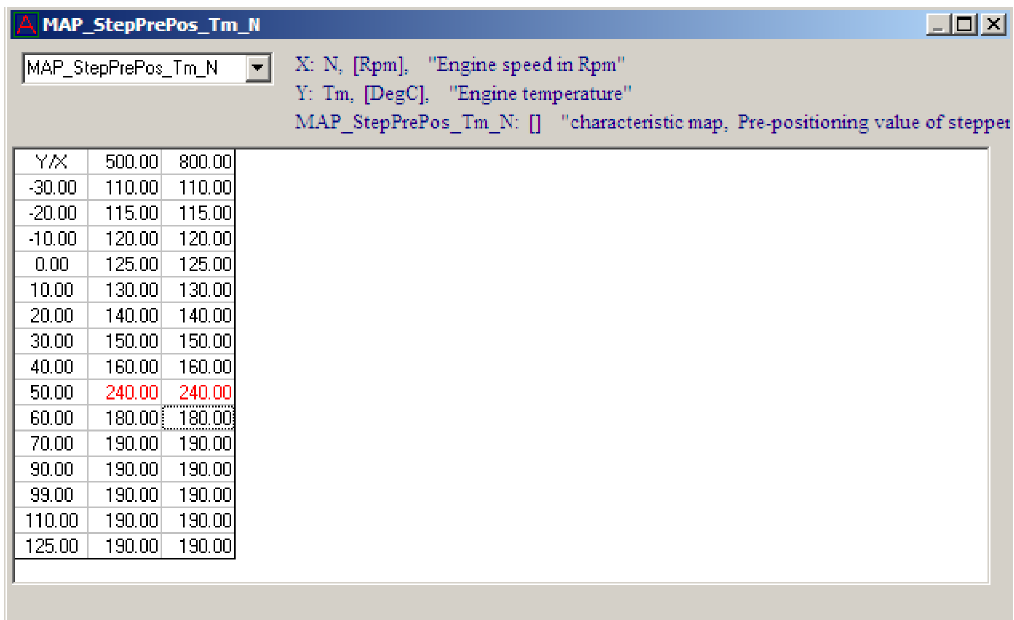
Then


fLc 20ms - 0 2 "Lambda controller output (word)"
FlTrsCr 20ms % -1536 1536 "rel. fuel mass transition compensation
LamWO2 20ms 0 16 "Actual Lambda measured by WO2 co
Last_Falling_Edge_Timer 20ms 0 .294967E+09 "LLD timer"
Last_Rising_Edge_Timer 20ms 0 .294967E+09 "LLD timer"
Last_Rising_Edge_Timer 20ms 0 .294967E+09 "LLD timer"
Nraw 20ms Rpm 0 16383.75 "Engine speed in rpm from LLD"
V StepPos 20ms 0 640 "Actual position of stepper motor"
V StepPosDsr 20ms 0 640 "Desired position of stepper motor"
Tps 20ms % 0 100 "throttle position with respect to lower r
UbAdc 20ms V 0 25.6 "battery voltage; scanned value of mic
uLsb 20ms V -1 4 "Voltage signal from the lambda senso

Note: **The value of "StepPosDsr" is the desired value from the table of "MAP_StepPrePos_Tm_N", based on the engine temperature "Tm". "StepPos" is the actually stepper motor position when engine is running at idle with the desired RPM. You want to tune the "StepPosDsr" as close as possible to "StepPos" at certain engine temperatures.**

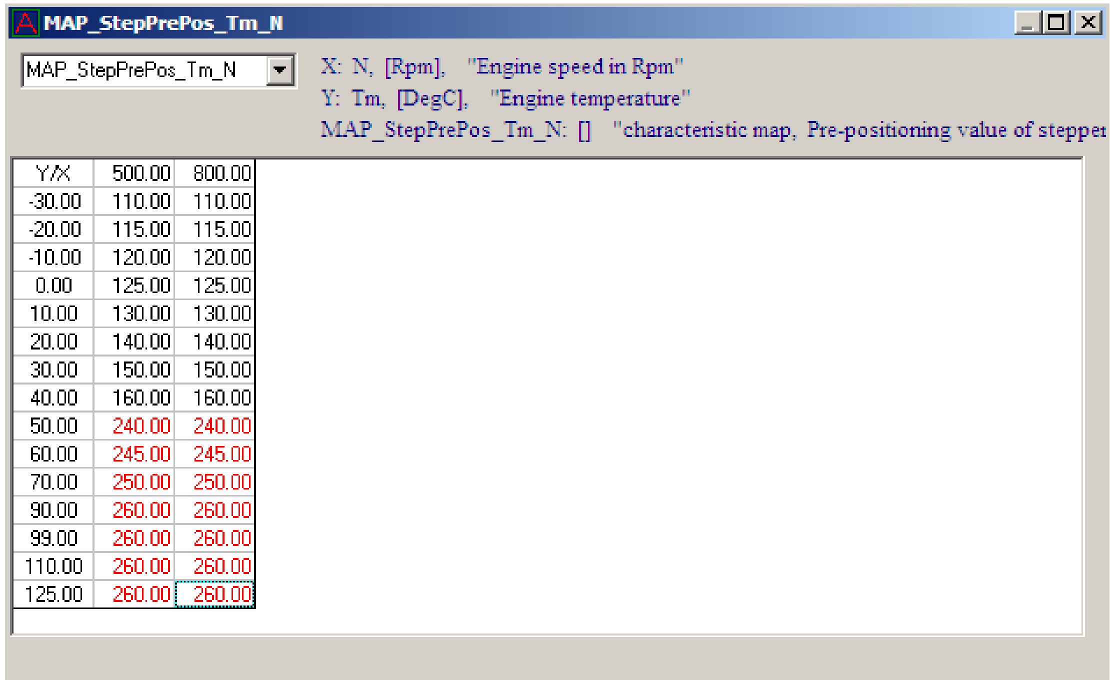
For example:

If engine temperature is 50 C, and the "StepPosDsr" value is 170 from the table, when engine is at idle with the desired RPM; the "StepPos" value is 290. So you can change the cell value to 240 at 50 C (the value is smaller than 290, the difference is about 50 steps, to have some room, for better engine start).



Y/X	500.00	800.00
-30.00	110.00	110.00
-20.00	115.00	115.00
-10.00	120.00	120.00
0.00	125.00	125.00
10.00	130.00	130.00
20.00	140.00	140.00
30.00	150.00	150.00
40.00	160.00	160.00
50.00	240.00	240.00
60.00	180.00	180.00
70.00	190.00	190.00
90.00	190.00	190.00
99.00	190.00	190.00
110.00	190.00	190.00
125.00	190.00	190.00

You can also change the value in the table when "Tm" is higher than 50 degree



MAP_StepPrePos_Tm_N

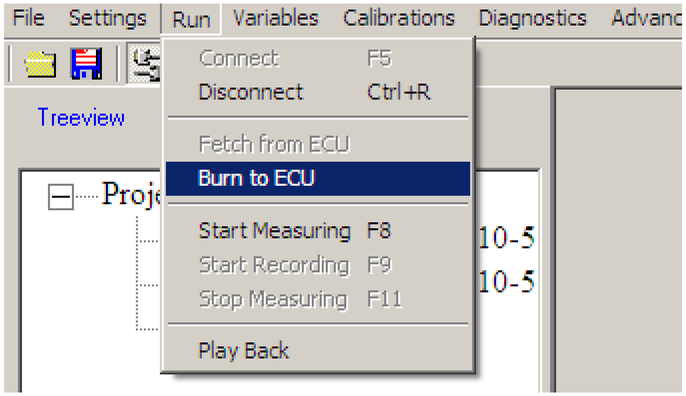
MAP_StepPrePos_Tm_N

X: N, [Rpm], "Engine speed in Rpm"
Y: Tm, [DegC], "Engine temperature"
MAP_StepPrePos_Tm_N: [] "characteristic map, Pre-positioning value of stepper

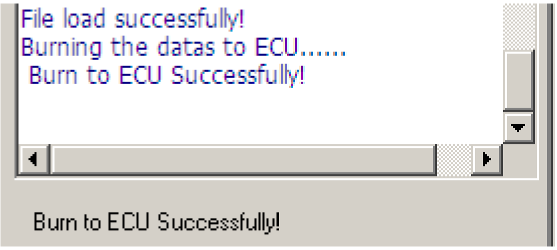
Y/X	500.00	800.00
-30.00	110.00	110.00
-20.00	115.00	115.00
-10.00	120.00	120.00
0.00	125.00	125.00
10.00	130.00	130.00
20.00	140.00	140.00
30.00	150.00	150.00
40.00	160.00	160.00
50.00	240.00	240.00
60.00	245.00	245.00
70.00	250.00	250.00
90.00	260.00	260.00
99.00	260.00	260.00
110.00	260.00	260.00
125.00	260.00	260.00

Note: **Increase the cell values in the table; you will get lower RPM. This means the bigger the stepper motor position, the less idle air.**

(3) After you change the value, then "Burn to ECU"



If the file downloaded successfully, lower left corner will pop up message "Burn to ECU Successfully



Note: Usually, the value of stepper motor position (StepPos) is 390, this means, the stepper motor has went to the largest position, if the idle RPM is still high, please check whether it has air leak, or the stepper motor connector is not connected well, or something wrong with the stepper motor.
The way to check stepper motor is take out it from throttle body, place a side, start to run engine, to see the stepper motor whether is working, and when key-off the ECU, the stepper motor will go to the smallest position and then go to the default position, if the stepper motor doesn't work, it means, the stepper motor is damaged or something wrong with the ECU.

3.4 Idle Controls via ignition timing

Note: If you do not have our ECU-controlled ignition, or if your engine is not recommended for ECU-controlled CDI (2-stroke engines, for example). Or our ECU does not support ignition controls for your engine; skip the rest of this part (ignition controls).

For throttle bodies (28mm for example) that do not have an IAC motor. The idle speed can be controlled by the ECU via ignition angle adjustment. This is only to adjust the small idle RPM deviation.

The mechanical idle air screw must be adjusted to a good position that your idle RPM is a little higher than you desired when the engine is warmed up.

Then the idle ignition angle can be retarded, and reduce the idle RPM at fully warmed up. At cold engine, idle ignition angle is likely at the optimum advanced angle, so that the idle RPM can be at your desired RPM even with a cold engine. For example, your idle ignition angle will be 9 degree CrA positive when it's cold. And it could be **-7** degree CrA, a negative value, when it is fully warmed up.

Idle speed controls via ignition angle controls (only possible if you have ECU-controlled ignition system).

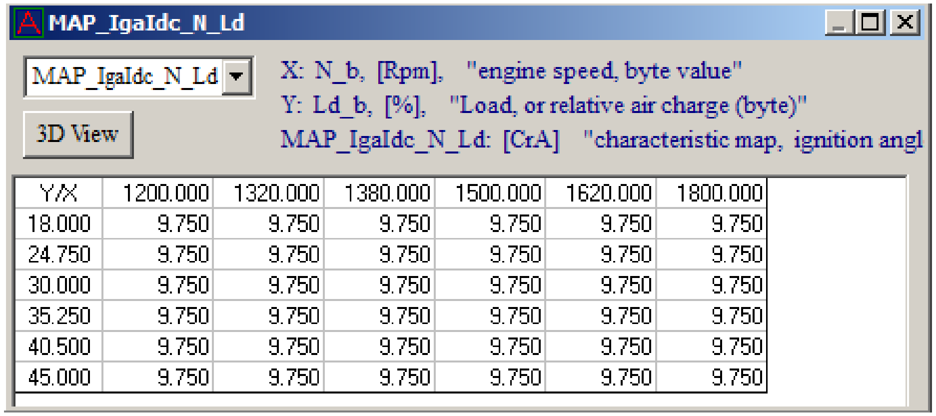
Why do you need the ECU-controlled CDI for idle speed control?

After you have a stable running engine with the fuel control only, now it's time for you to add our CDI box. Because it will help to lower down the idle RPM after engine warm-up. The strategy is to have enough idle air to start in cold and stable idle in cold first. But that amount of air is a little more than necessary for the warm engine. (That's why you get a high idle rpm at hot but a low idle rpm at cold.)

Then the strategy is to retard the spark advance to lower down the idle rpm once the engine is warmed up. It reduces the engine torque because of later sparks. (This also indirectly helps to reduce the emissions).
Furthermore, this gives some torque reserve for possible high idle requests (for example, to prevent undercharging).

And this also helps the launch, because once you open the throttle, the spark advance can be immediately goes to the optimum, and you get big torque increase!

Pre-defined idle ignition angle map MAP_IgaIdc_N_Ld : when engine runs in idle, (throttle body is fully closed).



Y/X	1200.000	1320.000	1380.000	1500.000	1620.000	1800.000
18.000	9.750	9.750	9.750	9.750	9.750	9.750
24.750	9.750	9.750	9.750	9.750	9.750	9.750
30.000	9.750	9.750	9.750	9.750	9.750	9.750
35.250	9.750	9.750	9.750	9.750	9.750	9.750
40.500	9.750	9.750	9.750	9.750	9.750	9.750
45.000	9.750	9.750	9.750	9.750	9.750	9.750

There are also two correct ignition factors based on battery voltage and engine temperature.
CUR_dIgaIdc_Tm, CUR_dIgaIdc_Ub

CUR_dIgaIdc_Ub

CUR_dIgaIdc_Ub

Ub_b: [V] "Battery voltage, byte value"
CUR_dIgaIdc_Ub: [CrA] "Characteristic Curve, delta ignition angle of battery voltage effect"

Ub_b	8.000	9.000	10.000	10.500	11.000	11.500	12.400	13.000	14.000	15.000
CUR_dIgaIdc_Ub	15.000	15.000	15.000	15.000	15.000	15.000	21.000	0.000	0.000	0.000

CUR_dIgaIdc_Tm

CUR_dIgaIdc_Tm

Tm: [DegC] "Engine temperature"
CUR_dIgaIdc_Tm: [CrA] "Characteristic Curve, delta ignition angle of engine temperature effect"

Tm	-30.000	-25.000	-15.000	-10.000	0.000	10.000	20.000	45.000	70.000	90.000
CUR_dIgaIdc_Tm	30.000	24.750	24.750	24.750	24.750	9.750	5.250	0.000	0.000	0.000

The pre-defined idle ignition output angle = MAP_IgaIdc_N_Ld + CUR_dIgaIdc_Tm + CUR_dIgaIdc_Ub
And the close loop ignition angle control for idle speed control is based on this pre-defined ignition angle.

EFI system without an IAC motor, and without ignition control

If the idle air is too much or too less, it leads the Engine idle RPM too high or too low.

Because ECU has no way to control the idle RPM, in this case, you may have to start with a high idle RPM and later on, to fine tune the throttle mechanical stop position or idle air screw position to get a compromise of idle RPM.

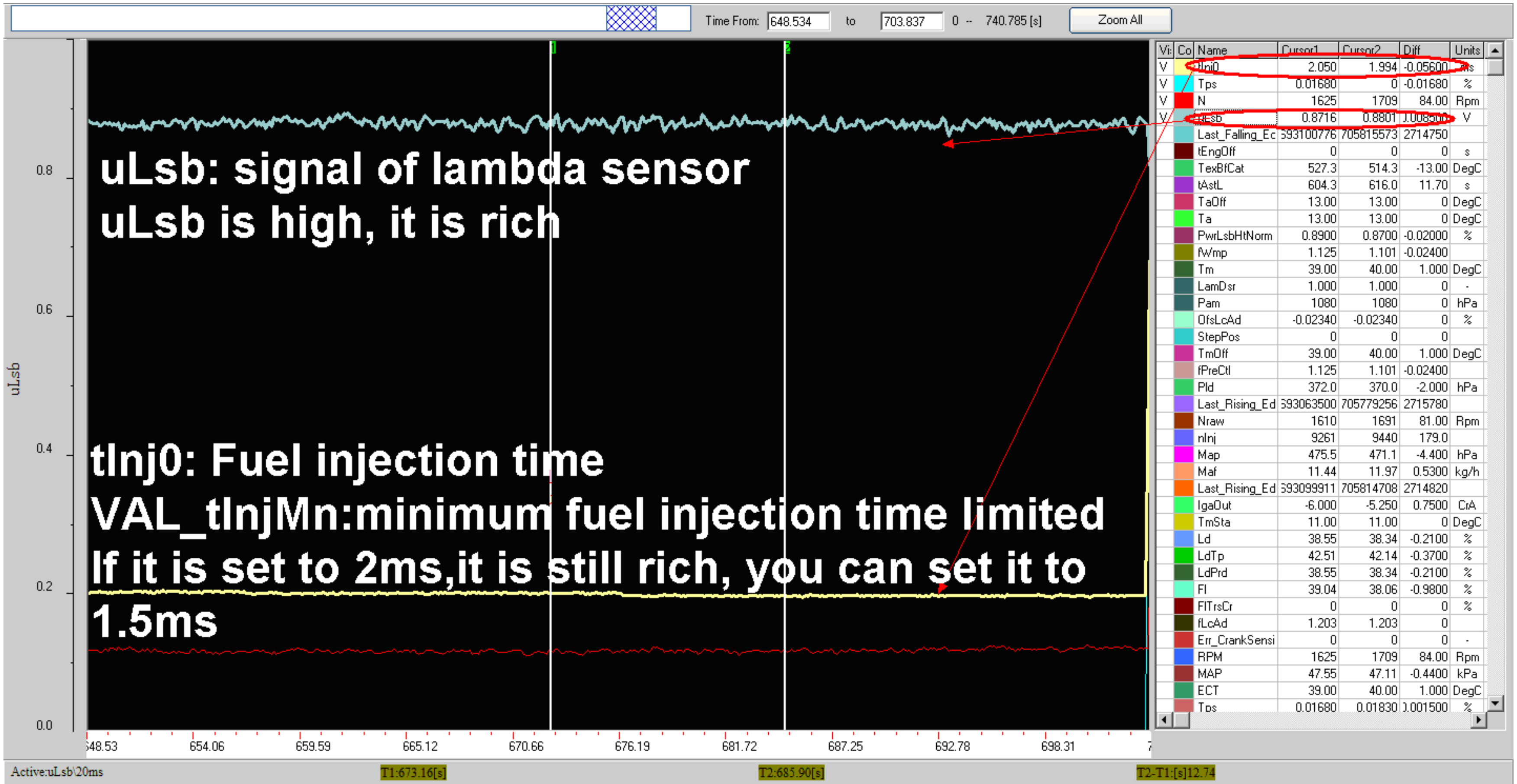
If the engine idle RPM is super high, like 3000rpm, ECU will cut the fuel, and the RPM may go up and down.

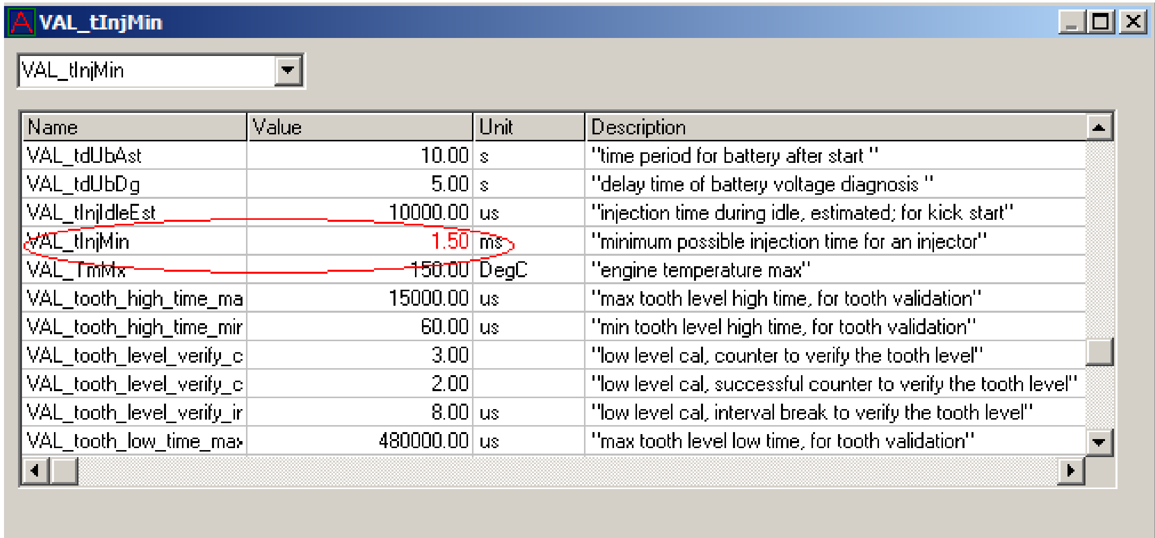
3.5 Idle fuel too rich because of minimum injector pulse width

Q: How and why to adjust fuel injection time in idle?

No matter how much you have changed the base fuel mapping for idle, (via VE table or TPS Load table), the engine still runs too rich. It seems no effect even you put the 0 in the table. It may be because the minimum fuel injection pulse width is limited. You need to change that limit lower, VAL_tInjMin.
This calibration basically defines the minimum pulse width that can open the injector. This is the physical limit of an injector, usually in 1ms to 2ms ranges.
Go to "menu->Advanced->Add Advanced Calibrations” to add the VAL_tInjMin. And decrease this value according to the actual engine running. This value by default is 2ms; or 2.2ms including the battery voltage compensation. The reason to have a 2ms min pulse width is because the injector has a better flow rate when it is bigger than 2ms. For many engines, the min pulse width could be smaller. Usually you can set it to 1.5ms or 1.0ms as the extreme minimum. Anything smaller than 1ms, you may not have fuel injection at all.

For example:


uLsb: signal of lambda sensor
uLsb is high, it is rich
tInj0: Fuel injection time
VAL_tInjMn:minimum fuel injection time limited
If it is set to 2ms,it is still rich, you can set it to 1.5ms
Time From: 648.534 to 703.837 0 -- 740.785 [s]
Zoom All
uLsb
Active:uLsb\20ms
T1:673.16[s]
T2:685.90[s]
T2-T1:[s]12.74



VAL_tInjMin

VAL_tInjMin

Name	Value	Unit	Description
VAL_tdUbAst	10.00	s	"time period for battery after start "
VAL_tdUbDg	5.00	s	"delay time of battery voltage diagnosis "
VAL_tInjIdleEst	10000.00	us	"injection time during idle, estimated; for kick start"
VAL_tInjMin	1.50	ms	"minimum possible injection time for an injector"
VAL_TmMx	150.00	DegC	"engine temperature max"
VAL_tooth_high_time_ma	15000.00	us	"max tooth level high time, for tooth validation"
VAL_tooth_high_time_mir	60.00	us	"min tooth level high time, for tooth validation"
VAL_tooth_level_verify_c	3.00		"low level cal, counter to verify the tooth level"
VAL_tooth_level_verify_c	2.00		"low level cal, successful counter to verify the tooth level"
VAL_tooth_level_verify_ir	8.00	us	"low level cal, interval break to verify the tooth level"
VAL_tooth_low_time_max	480000.00	us	"max tooth level low time, for tooth validation"

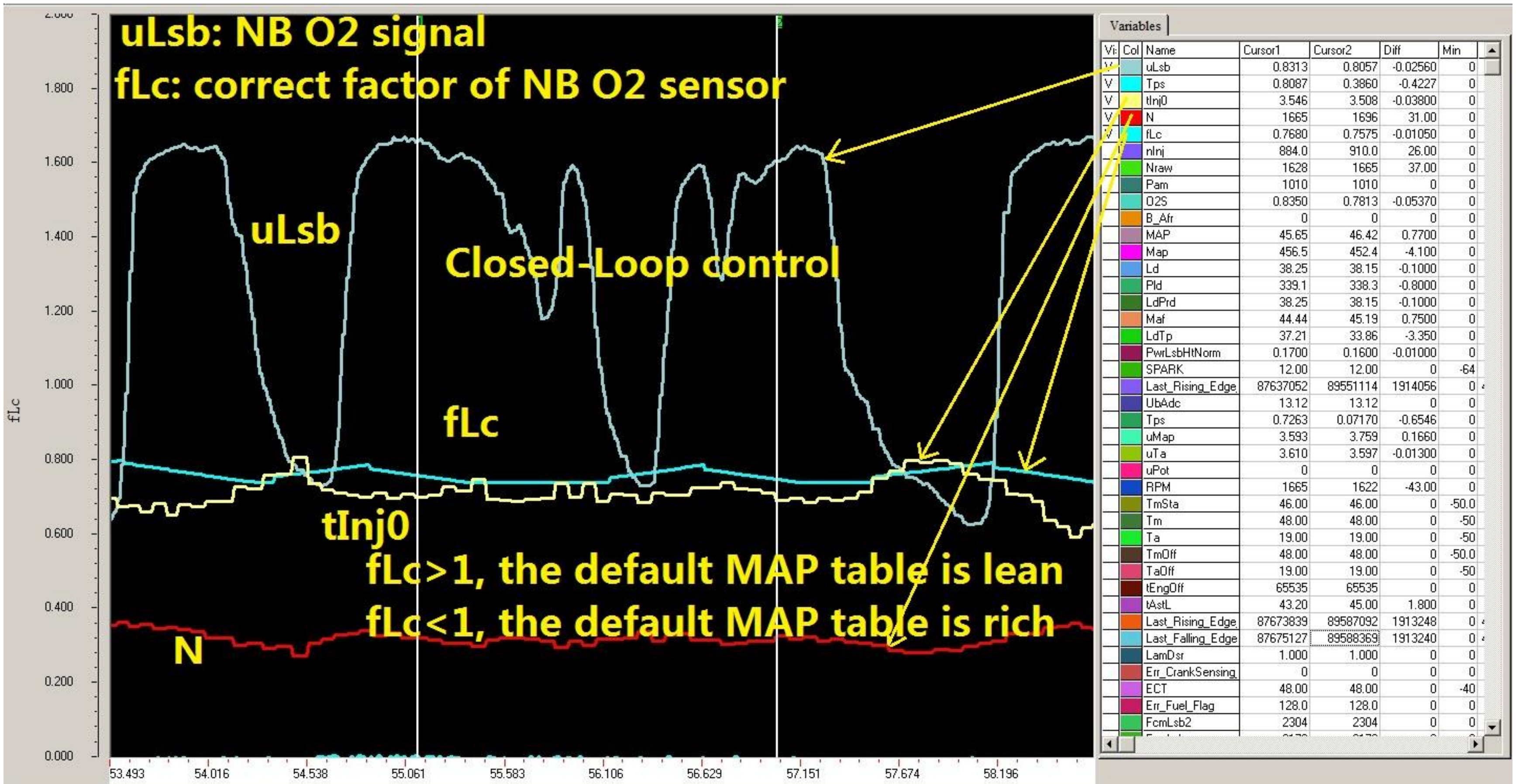
Q: How do I know whether it is rich or lean?

You can use the following method to know whether the engine runs too rich or too lean with a NB O2 sensor or a wideband ALM. Also you may know it by smelling the exhaust gas when it is too rich.

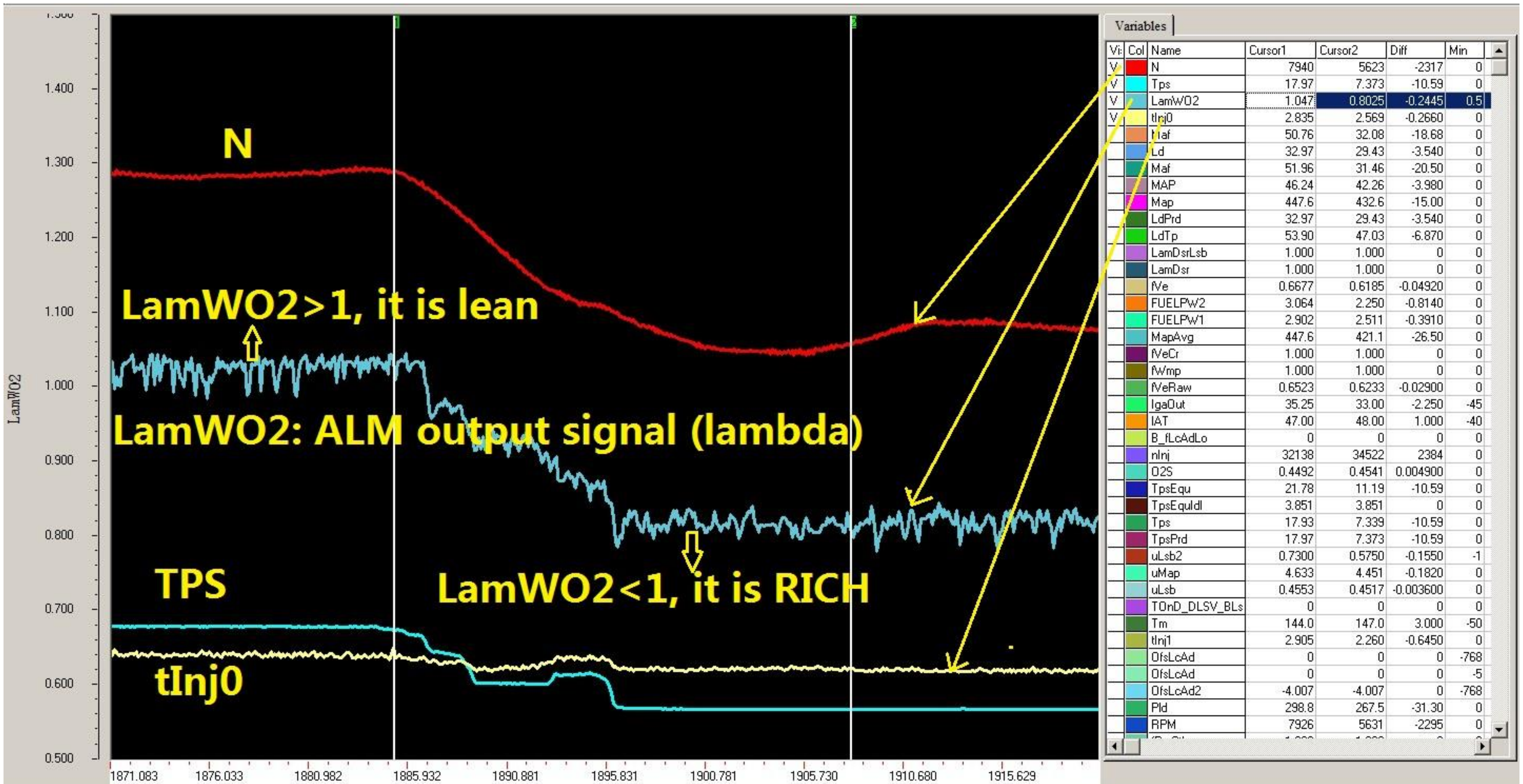
(1) With a NB O2 sensor. You can read the voltage of the NB O2 sensor in the HAECAL gauge, O2S; to know whether it is rich or lean.

uLsb: signal of lambda sensor, in the List of variables.

If it is higher than 0.45V, the engine runs rich. If it is lower than 0.45V, it runs lean.


uLsb: NB O2 signal
fLc: correct factor of NB O2 sensor
uLsb
Closed-Loop control
fLc
tInj0
fLc>1, the default MAP table is lean
fLc<1, the default MAP table is rich
N
Variables
fLc

(2) Read the AFR by using an ALM gauge, which you can buy from HAE. You can also read the real time lambda (LamWO2) in HAECAL, for which you need connect the ALM to ECU through the EFI harness. With the ALM, the advantage is that you can read / log the real-time AFR or lambda value, and even have HAE' auto-tuning feature.



Chapter 4 Open-loop and closed-loop fuel control

4.1 Open Loop Fuel Control

What is open loop fuel control, why do we have it?

Open loop fuel control is to run the engine without installing the O2 sensor. The reason we need to run it is a precautious step to protect the O2 sensor; to have a stable engine running before plugging in the O2 sensor.

A good practice is to start and run the engine in the open loop mode, especially if your engine is very different than our base calibration engines (like a scooter GY6 150cc, which is the mostly tested engine). Or you have modified your engine significantly, like a big bore kit, a high cam, or a bigger intake manifold. All these difference will cause the big deviations in the air charge model pre-loaded in the ECU. Close-loop fuel control and self-learning is only supposed to correct the small errors from engine to engines. The big difference cannot be learnt by the ECU and could lead to the oscillations, or unstable running.

Without a comparatively stable engine running in open-loop, the exhaust from the engine could be erratically rich or lean or have fuel flooding, and random moistures in the exhaust, etc. These could damage the O2 sensor before you even have a chance to run close loop controls.

In open loop mode, ECU is reading the MAP sensor signal, TPS signal, and temperature signals to calculate the fuel, and control the fuel comparatively precisely.

In open loop mode, you can drive the vehicle around, and tip-in, tip-out, and you can do almost everything, except that the ECU does not really know whether you are running rich or lean. And the ECU cannot self-learn the engine for variations. The self-fine-tuning of the fuel is not happening.

Even in open loop mode, if tuned well, an EFI engine can run much better than a carb engine, NO QUESTION.

Many OEMs have open loop EFI engines certified with EPA / CARB emission regulations. The key is the tuning.

What if I cannot get a stable engine running in the open loop mode?

If you "smell" it lean, a quick and dirty fix is to enrich the fuel with the global factor: "**VAL_fFlApp**". And once you have a relative stable run, you can detune this factor by using other parameters.

If your idle speed is too low, and you cannot maintain the stable idle RPM, go to Chapter 3 (idle air adjustment).

For open loop tuning, you can find the details in Chapter 5. But you need to know what a close loop control system is first.

4.2 Closed Loop fuel control

Q: How to tune the close-loop fuel?

A: You do NOT need to tune it. You only need to make sure it's happening. ECU will automatically tune it (with the O2 sensor).

Q: How do I know it's in close-loop controls?

First, you need to know a little bit of the O2 sensor.

The O2 sensor indicates lean or rich by a voltage signal.

When the O2 sensor is not active, the voltage should be about 450mV (or 0.45V);

When the O2 sensor indicates **rich**, the voltage should be **>** 450mV;

When the O2 sensor indicates **lean**, the voltage should be **<** 450mV;

That's right! It's a narrow band O2 sensor. The wideband O2 sensor is too expensive to be included in the base kit. We do have a wideband kit, ALM, for your tuning purpose.

When you see the O2 sensor voltage oscillating up and down around 450mV, that means the close-loop control is active.

Q: How do I see the O2 sensor voltage?

Easy way: in HAECAL, among a bunch of green/black gauges, there is an O2S gauge indicates the voltage.

Better way: You need to log the data and play it back to see what's going on.

In Data Analyzer, check the signal "uLsb", which is "voltage of lambda sensor (O2)".

Q: My O2 sensor voltage is always 450mV (even after warm-up, 3 minutes later), what's wrong?

Possible reason #1: you don't have it connected;

Possible reason #2: the sensor is broken, or the wire is broken;

Read the diagnostic DTC in HAECAL.

Q: My O2 sensor voltage is always greater than 450mV, why?

You are running rich all the time.
Case 1: for many engines, during idle, it has to run a little rich, just because the injector size is not small enough. You are running in the minimum pulse width (calibrated as 2ms). This is OK.

Case 2: you are running rich at all operating conditions, idle, part-load, WOT, etc. This could be caused by: you have enriched too much or you have some wrong calibrations. Check your enrichment factors VAL_FlApp; you fuel injector size, your engine displacement, etc, in HAECAL.

Case 3: The base engine calibration is way off for your engine. The ECU self-tuning is only capable of fine-tuning. You need to do some advanced tuning. Read the later chapters.
Case 4: The sensor is broken, or the wire is shorted.

Q: My O2 sensor voltage is always smaller than 450mV, why?

You are running lean all the time. It may happen if you have fuel supply issues. For example, your fuel pump is having air bubbles; or the fuel pressure is not enough; or the pressure regulator is malfunctioning; or the battery voltage is too low, etc.
You have wrong calibrations. Check you fuel injector size, your engine displacement, etc.

The base engine calibration is way-off for your engine. The ECU self-tuning is only capable of fine-tuning. You need to do some advanced tuning. Read the later chapters.

The sensor is broken, or the wire is shorted.

Q: How long does the ECU need to do the self-tuning?

ECU self-tuning is only possible if you have a stable engine running in active close-loop controls. (meaning, engine has warmed up, O2 sensor is working, system is fault free, open-loop control is in the ball-park of the stoic AFR; close-loop is active, etc) .

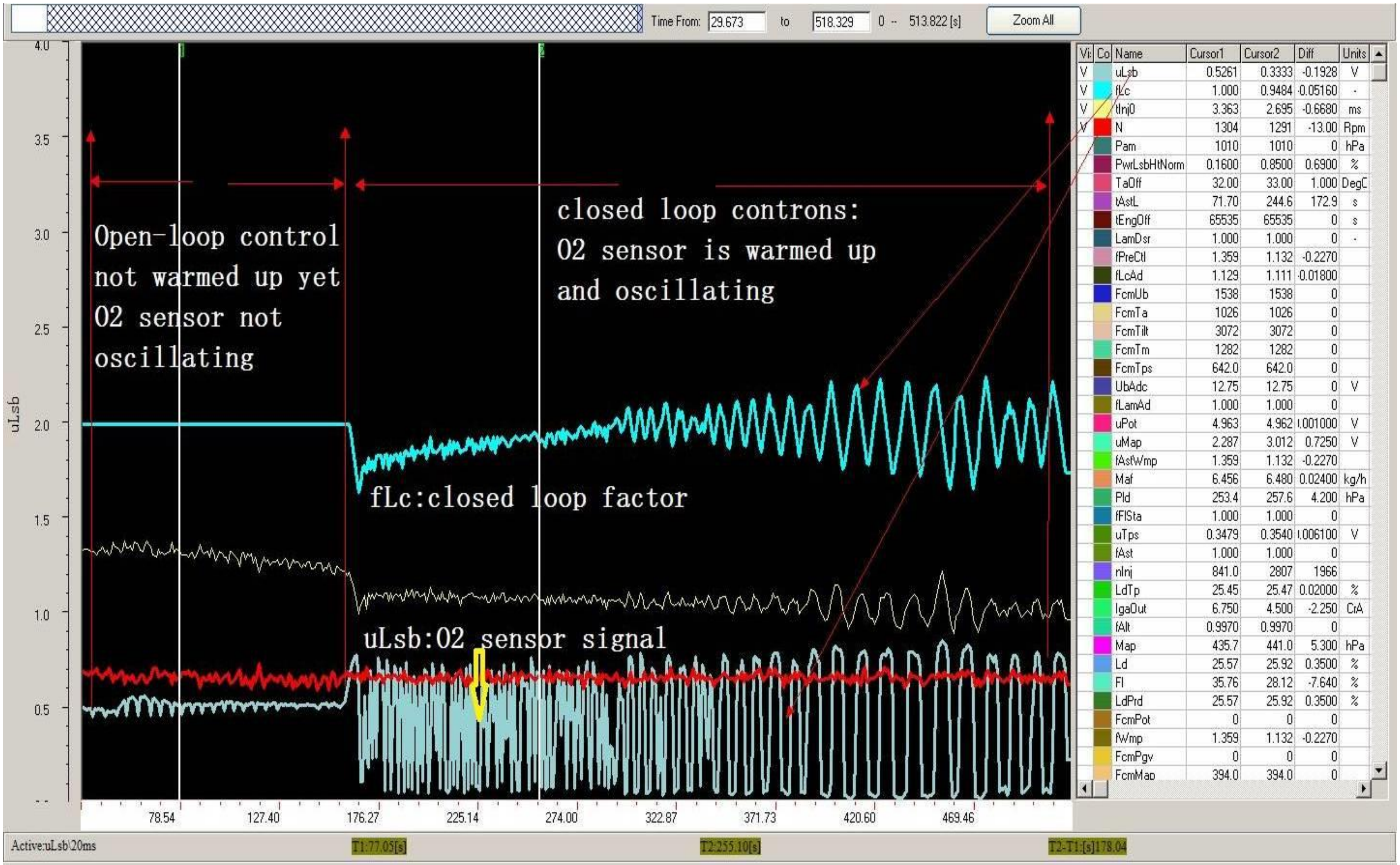
It does not take long for the self-tuning. It takes a few minutes for every different operating condition. The key is to run the engine in a steady-state and traverse though the common operating conditions (idle, part-load, high-load, WOT, etc.)

There are 2 key signals for you to check the self-tuning progress: "fLcAd" and "OfsLcAd". When you see they are slowly changing, the self-tuning is in progress; when they settle down at some non-1 value, the self-tuning is kind of done.

Note: **This self-tuning is based on the Narrow Band O2. It only adjusts small errors. It only works if the base fuel mapping is good.**

Note: **This self-tuning is a life-long continuous process. It changes very slow, and takes only effect of long term factors**

Note: **During start and warm-up process, the engine is running in open-loop mode; the self-tuning only starts in the close-loop mode after the O2 sensor is warmed up (see the graph below):**


Time From: 29.673 to 518.329 0 -- 513.822 [s]
Zoom All
Open-loop control
not warmed up yet
02 sensor not
oscillating
closed loop controns:
02 sensor is warmed up
and oscillating
fLc:closed loop factor
uLsb:02 sensor signal
uLsb
78.54
127.40
176.27
225.14
274.00
322.87
371.73
420.60
469.46
Active:uLsb\20ms
T1:77.05[s]
T2:255.10[s]
T2-T1:[s]178.04

Chapter 5 VE table and Load table tuning

Note: **EFI is an advanced technology compare to the carbs. It requires a thorough understanding of the electronics and software strategy to be able to tune the system.**

Please educate yourself on the basics of engines and electronic fuel injections. There is tons of info on the internet, and all you need to do is find appropriate levels of materials and read them.

The first concept you need to understand is the "Volumetric Efficiency".

Warning: some users tried to tune the volumetric efficiency or throttle based fuel mappings, without fully understanding the EFI system, and ended up with the worse-running engine.

Volumetric Efficiency – VE table

Volumetric efficiency is probably the most important characteristics of an engine.
It determines how efficient the engine is sucking the air into the cylinder, and therefore, how much torque it can generate, given the certain spark advance and air-fuel ratio.
It is the fundamental calibration of the engine tuning.

By definition, volumetric efficiency is the fresh air mass in the combustion chamber divided by the total air mass in it. Basically there is always some residual combustion gas trapped in the combustion chamber at the end of the engine exhaust stroke. The total air mass in the cylinder is the sum of the fresh air and the residual exhaust gas. To know exactly how much fuel you need to inject and to mix with the air in the combustion chamber, you have to know how much fresh air is in there instead of the total air mass, and that is why you have to know the volumetric efficiency.

With a MAP sensor based system, you can measure the pressure in the intake manifold, and use that as the pressure in the combustion chamber, and together with the intake air temperature, and engine displacement, you can calculate the air mass in the combustion chamber. This is called by the auto industry as “**Speed Density**" method. Following the ideal gas law:

$$m = PV / RT$$

m - Air mass

P - Pressure

V - Volume or engine displacement

T - Air temperature

R - Gas constant

Once you know the manifold air pressure, intake air temperature, and the volumetric efficiency, you can calculate the fresh air mass in the cylinder and therefore the fuel quantity that needs to be injected.

Unfortunately, volumetric efficiency is not able to be measured directly with any sensor. You'd have to calibrate it one operating-point (RPM vs MAP reading) at a time.

The best way to tune volumetric efficiency is to put your engine on a dyno and use an accurate wideband air-fuel-ratio (lambda) meter to measure the mixture, and back-ward calculate the volumetric efficiency.

However, knowing most people don't have access to a dyno, we can propose a coarse way to tune the volumetric efficiency.

If you have a wideband controller, you can run the engine with the wideband controller together, or ride the bike in a steady state (comparatively constant throttles and RPMs); at a certain MAP signal and RPM, you can tweak the volumetric efficiency table until you get the stoic AFR read from your wideband controller.

Make sure you have an accurate and reliable wideband controller, or you could end up with wasting your time.

If you don't even have a wideband controller, you can use an even coarser way to tune it. Use the narrow band O2 sensor, which comes with the kit, and read the O2S gauge, lean or rich, at certain MAP signal and RPM, tweak the volumetric efficiency until O2S gauge indicates an oscillating voltages, up and down around the 450mV.

Note: **NB O2 based tuning is only possible if you have everything running normally and in close-loop control.**

LOAD based system

To know more about the advance tuning with our EFI kit, you need to understand what the "LOAD" is.

"LOAD" by definition is the actual air mass charged into the cylinder divided by the ideal air mass that could be filled into the full cylinder.

What is the ideal air mass? When you have your cylinder fully filled with the fresh air at sea level (barometric pressure = 1 bar), and at the temperature of zero degree C (air temperature = 32 F), the mass of the air in the cylinder is the ideal air mass. In other words, if your engine displacement is 150cc, the idea air mass is the 150cc fresh air mass at standard conditions (1 bar, 0 degree C).

"LOAD" is a relative value, in percentage, unit-less.
"LOAD" is not the throttle position, because you can have different air mass in the cylinder at the same throttle position.

Why making it so complicated? Because it is the only way, the RIGHT way, to control the engine.

"LOAD" tells the ECU how much fuel is exactly needed for the desired air-fuel-ratio. Because you can only calculate the fuel quantity, if you know how much air is in the cylinder.

"LOAD" based system makes the tuning process universal for all engines! It does not matter how big or how small your engine is.

"LOAD" based system makes the EFI system modularized. The subsystems can be tuned independently (namely air sub-system, fuel-subsystem, ignition-subsystem, etc.)

You may say: "Hey, I can just use the throttle position and RPM, to map the fuel pulse width on the dyno".

While, you will end up re-tune all your mapping for some small things that you overlooked, or any small changes of the engine. You will have to start the tuning for a new engine from the scratch every time.

LOAD based tuning makes most of our tuning data “portable” from one engine to another, with minimum changes. LOAD based tuning is the only professional way to tune.

TPS based LOAD mapping

Once you know what the "LOAD" is, you can tune the TPS based load mapping.

Our system do not let you map the fuel pulse width directly out of the TPS / RPM table, because it is too coarse, and it is affected by too many factors (temperatures, altitude, speed, AFR, etc.). It may seem easier to map the engine in the coarse way, but it actually costs more time and efforts later on. You'll pay for that "short-cut" sooner or later.

Our system let you map the "LOAD" out of TPS/RPM table, and then the load is used everywhere else as the base inputs (fuel, ignition, lambda, etc). Why? Because LOAD is the most representative physical variable for air charge in the cylinder. Throttle position is not even proportional to the air mass. It has a non-linear relation to the air mass (if you know some math☹).

LOAD is normalized against the air temperature, and altitude, and pressure. This means, you base LOAD mapping is not affected by the altitude, temperature, density, etc. Those factors will be added in later. This is the fundamental of "model-based-design" , a modern method used by Ford and GM, etc. engineers.

The way to calibrate the LOAD mapping is similar to the "volumetric efficiency" table. The best way is to use an engine dyno. If not, use a wideband controller, and if not, use the narrow band O2 sensor to do "estimations".

At certain throttle position, and RPM, tweak your LOAD output, until you have a good AFR.

This is called "**Alpha-N**" model by the auto industry.

Blended Speed-Density and Alpha-N methods

So, "Speed-Density", or "Alpha-N", which method is better for engine controls? There have been too many debates on this. Just google it.

But the truth is: neither is perfect. You need both. You need blend them together to have the best engine controls, especially for high RPM, sporty motorcycle engines.

As common sense, MAP sensor is a direct measure of air mass into the cylinder, while TPS sensor is an indirect measure of the air mass. It seems MAP sensor based "speed-density" is better than TPS based "Alpha-N" method for air charge detection. This is true at low RPM and small loads when the MAP sensor signal has enough resolutions and the air mass calculated based on the instant manifold pressure is more accurate than the estimated by TPS position. TPS based air mass estimation has very poor resolution at low throttle opening. A small throttle position change at low throttle can cause big air mass change.

But as you may have noticed, small engines' manifold pressure is changing so dynamically, there is no "stable, constant" MAP signal; even you hold the throttle position unchanged. This becomes worse when you have large throttle opening, and you are running more than 6000 RPM. MAP sensor signal becomes un-stable and even not usable. For motorcycle sporty engines, at large throttle opening, which is usually accompanied by high RPMs, the air flow in the manifold is so fast, that the pressure change cannot be detected by MAP sensor any more. The result: you open the throttle more, but your MAP sensor gives you the same pressure reading, which is certainly wrong, because you have more air flow into the cylinder. In this situation, you must use "Alpha-N" method to calculate the air charge.

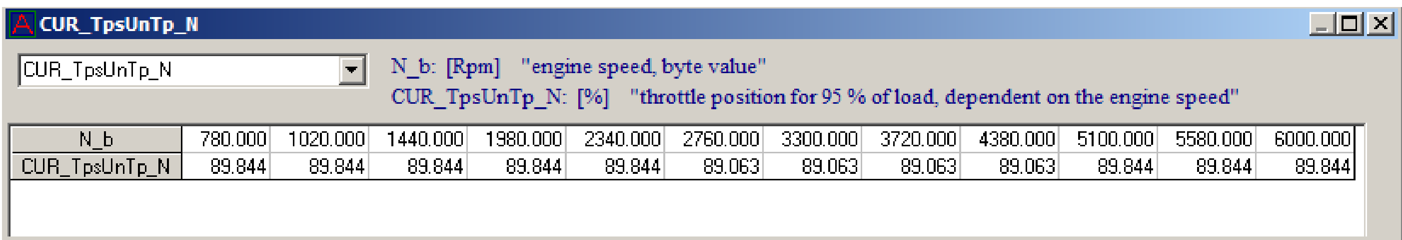
So, in short, you should use "Speed-density" for low RPM, low loads; but use "Alpha-N" method for high RPM, high loads. This is exactly how our system works.

We have a TPS characteristic curve that is based on RPM to split the fuel mapping between the "speed-density" and "alpha-N" methods. This curve is the threshold of TPS, that below it, you use "speed-density", and above it, you use "alpha-N". The curve is dependent on RPM.

You can find this curve in the advanced calibrations in HAECAL (this calibration is only available for certain users):

HAECAL → advanced → add advanced calibrations; Add:

CUR_TpsUnTp_N "Characteristic curve, TPS threshold, air flow is un-throttled and/or pressure change is insensitive."



N_b	780.000	1020.000	1440.000	1980.000	2340.000	2760.000	3300.000	3720.000	4380.000	5100.000	5580.000	6000.000
CUR_TpsUnTp_N	89.844	89.844	89.844	89.844	89.844	89.063	89.063	89.063	89.063	89.844	89.844	89.844

Too complicated? I don't want to tune both tables, How about just one table?

Of course, you can just use one method: "Speed-Density", or "Alpha-N". Only you don't have as much accuracy as the blended method. But most users do not really need that much of accuracy. Using one method is often good enough.

If you engine max RPM is less than **8000RPM**, "speed-density" method often gives good enough controls without using "alpha-N". Actually most of our base engine calibrations use the "speed-density" over the whole load / speed range. This means you only need to tune the VE table.

Some other users prefer to using TPS based fuel mapping. And use MAP sensor only for Baro pressure reading (therefore adjust fuel for different altitude). This is especially true for 2-stroke engine tuning. As you know, 2-stroke engine does not have much meaningful manifold pressure to be used as air charge detection.

In this case, you only need to tune the "TPS based Load Mapping" table.

The disadvantage of "Alpha-N" method is that it does not adapt to the small changes in the intake air systems. Say, a small air leak will cause your idle AFR deviated.

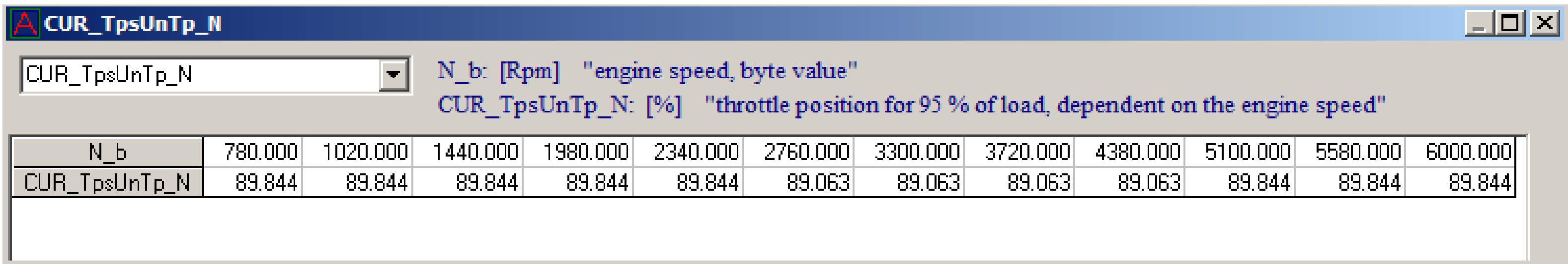
For a sporty, high RPM engine, like Kawasaki Ninja 250r, which has a RPM range of 1500rpm - 13000rpm, the blended "speed-density" and "alpha-N" is a must to have a good control over the wide range.

Specific examples:

Ok, you have read and known the above information, let's try to tune it.

If your engine is running rich or lean in different throttle positions, you need do some advanced calibrations. There are two fuel mapping tables, one is "RAM_MAP_fVe_Map_N" which is based on pressure signal "Map" and RPM, the other table is “RAM_MAP_LdTp_Tps_N" which is based on Tps and RPM.

For **four-stroke** engines, the default fuel mapping usually is volumetric efficiency table (MAP_fVe_Map_N), at least for all low RPM and mid-range RPM.   For high RPM and WOT, it is based on "MAP_LdTp_Tps_N". The split between the 2 tables is defined in "CUR_TpsUnTp_N" :



N_b	780.000	1020.000	1440.000	1980.000	2340.000	2760.000	3300.000	3720.000	4380.000	5100.000	5580.000	6000.000
CUR_TpsUnTp_N	89.844	89.844	89.844	89.844	89.844	89.063	89.063	89.063	89.063	89.844	89.844	89.844

If the throttle position (Tps) is bigger than the value of the table, it will be changed to use the Alpha/N model, otherwise it will use VE table.

Example:

(1) You can use NB O2 sensor voltage in the closed loop control to know whether it is rich or lean.

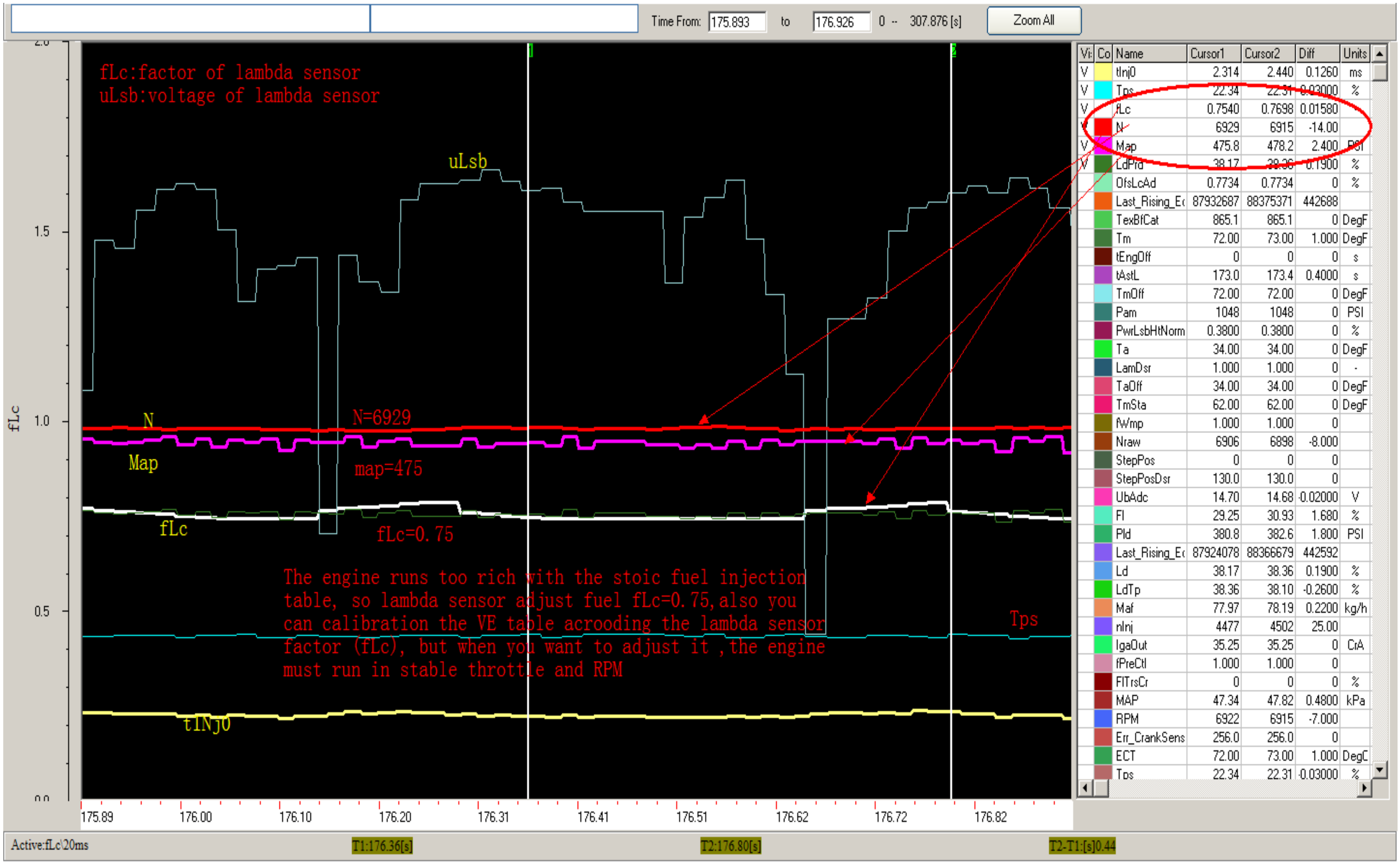

Time From: 175.893 to 176.926 0 -- 307.876 [s]
Zoom All
fLc:factor of lambda sensor
uLsb:voltage of lambda sensor
uLsb
N
N=6929
Map
map=475
fLc
fLc=0.75
The engine runs too rich with the stoic fuel injection table, so lambda sensor adjust fuel fLc=0.75,also you can calibration the VE table acrooding the lambda sensor factor (fLc), but when you want to adjust it ,the engine must run in stable throttle and RPM
Tps
tINj0
fLc
Active:fLc\20ms
T1:176.36[s]
T2:176.80[s]
T2-T1:[s]0.44

Figure 1

Note: **fLc is the factor of lambda control, when closed-loop is active. It automatically changes based on the O2 sensor feedback. If it is smaller than1 for a long time, it means the default VE table value is too bigger; you need change it to a smaller one to reduce the fuel quantity.**

(2) Using the wideband ALM, you can read the AFR or lambda directly, and also log the real-time lambda (lamWO2) in HAECAL.

For details on how to integrate the wideband ALM to the ECU and enable the ALM based auto-tuning, please download our document of "ALM-ECU integration Manual" here:
www.uavenginesystem.com/download/

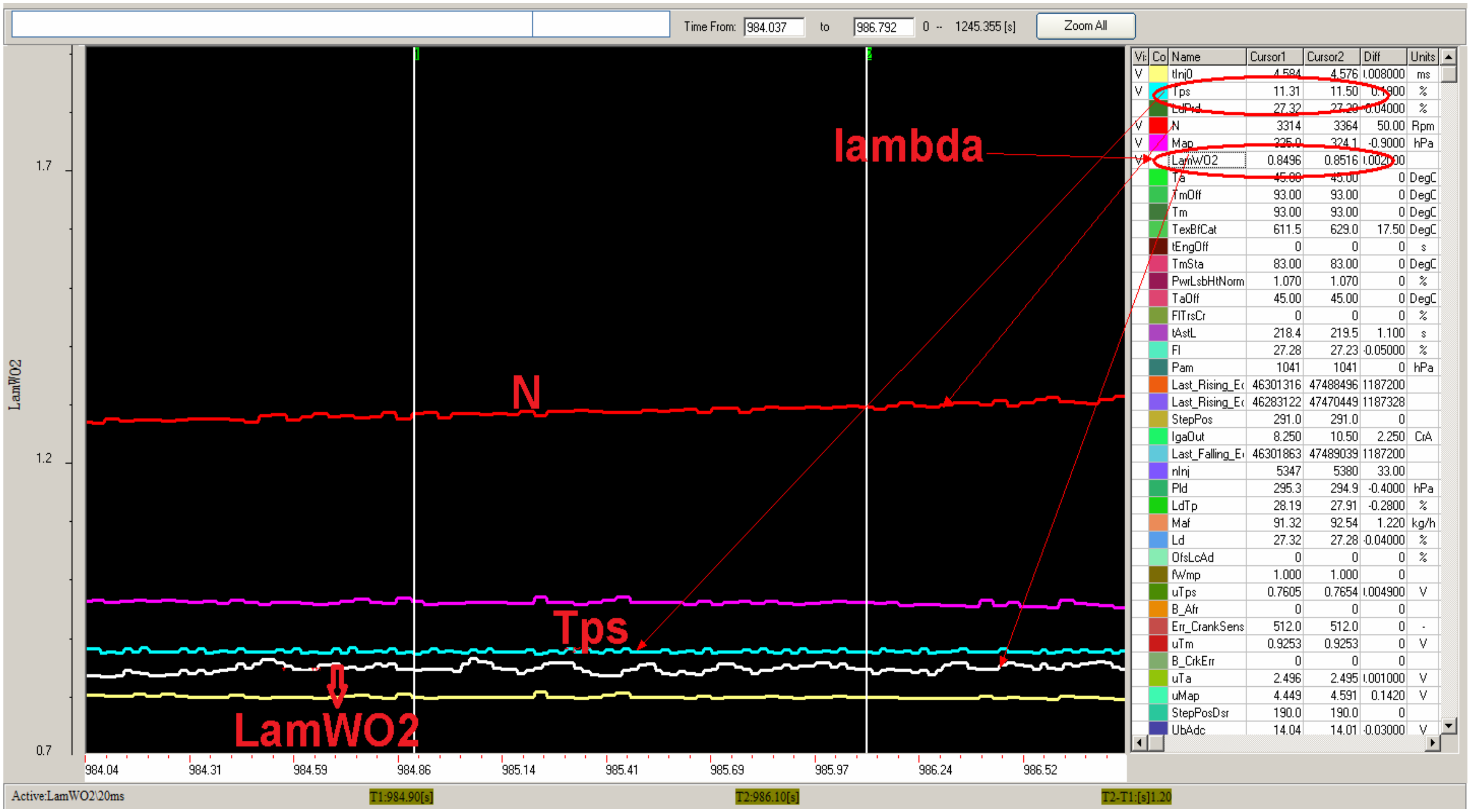
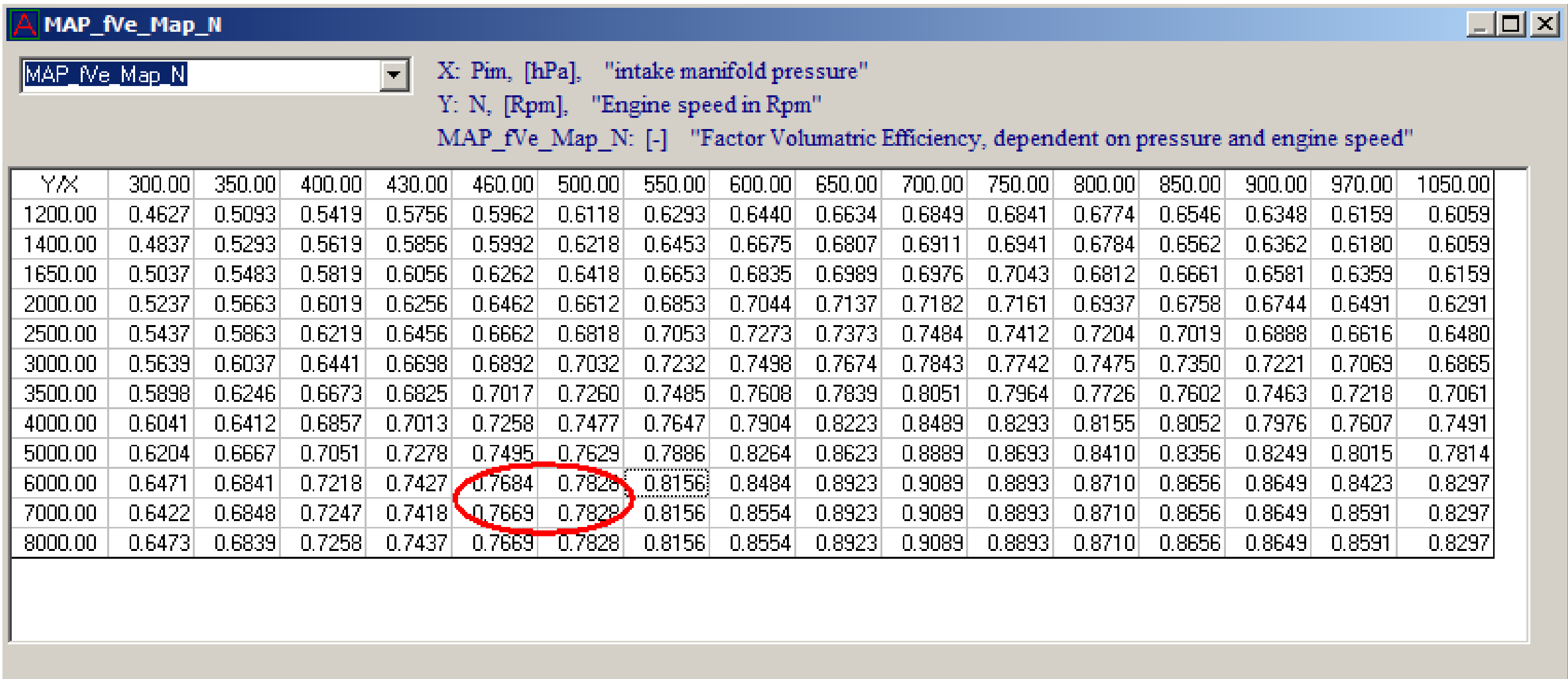


Figure 2

You can change the VE table value based on "Map" and "N", and adjust MAP_LdTp_Tps_N based on "Tps" and "N". You can read the value of "Map" and "N" in HAECAL (Go to menu->Variables->Display Type->List).

NOTE: If you want to tune the value of VE table or Alpha/N table, the engine must be warmed up, there is no enrichment factor, and you set it to ECO mode with the NB O2 sensor.

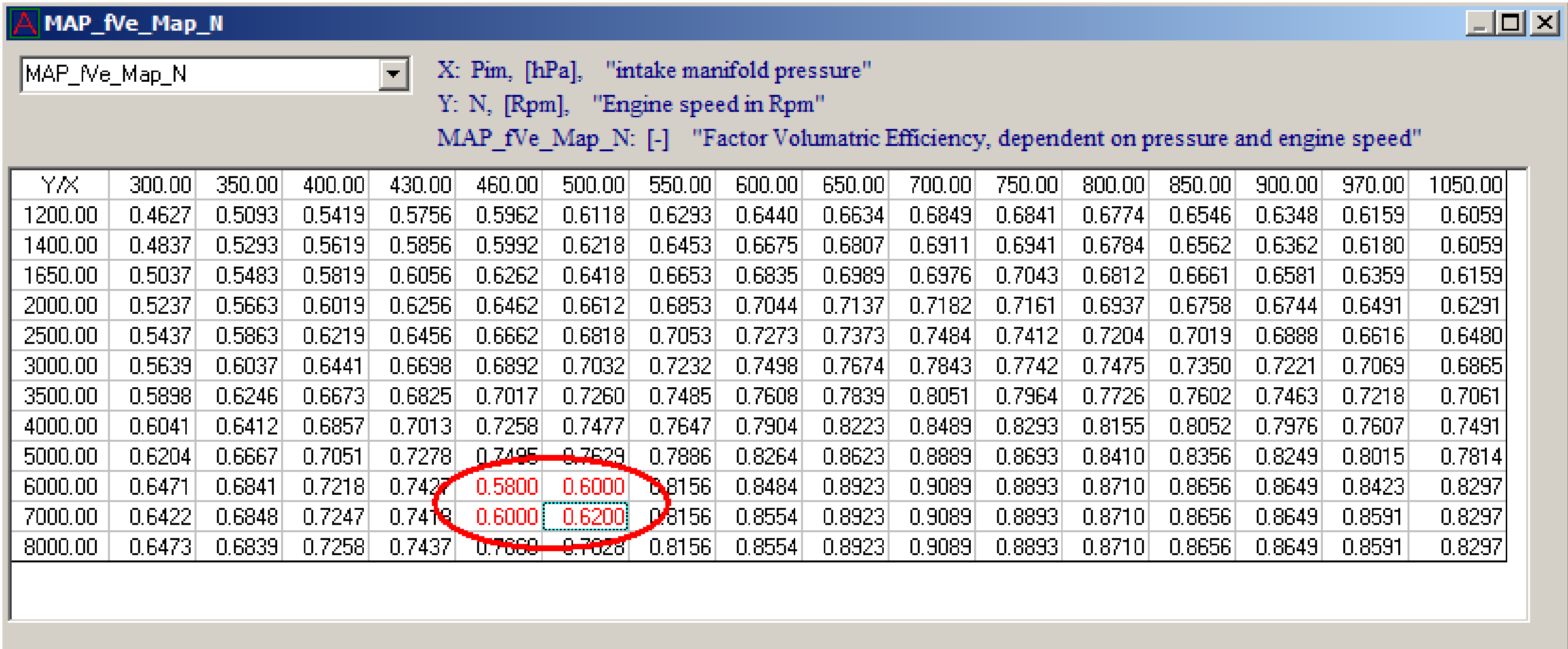
Try to calibrate one cell value now, N=6929, Map=475, fLc (Or lamWo2) =0.75



Y/X	300.00	350.00	400.00	430.00	460.00	500.00	550.00	600.00	650.00	700.00	750.00	800.00	850.00	900.00	970.00	1050.00
1200.00	0.4627	0.5093	0.5419	0.5756	0.5962	0.6118	0.6293	0.6440	0.6634	0.6849	0.6841	0.6774	0.6546	0.6348	0.6159	0.6059
1400.00	0.4837	0.5293	0.5619	0.5856	0.5992	0.6218	0.6453	0.6675	0.6807	0.6911	0.6941	0.6784	0.6562	0.6362	0.6180	0.6059
1650.00	0.5037	0.5483	0.5819	0.6056	0.6262	0.6418	0.6653	0.6835	0.6989	0.6976	0.7043	0.6812	0.6661	0.6581	0.6359	0.6159
2000.00	0.5237	0.5663	0.6019	0.6256	0.6462	0.6612	0.6853	0.7044	0.7137	0.7182	0.7161	0.6937	0.6758	0.6744	0.6491	0.6291
2500.00	0.5437	0.5863	0.6219	0.6456	0.6662	0.6818	0.7053	0.7273	0.7373	0.7484	0.7412	0.7204	0.7019	0.6888	0.6616	0.6480
3000.00	0.5639	0.6037	0.6441	0.6698	0.6892	0.7032	0.7232	0.7498	0.7674	0.7843	0.7742	0.7475	0.7350	0.7221	0.7069	0.6865
3500.00	0.5898	0.6246	0.6673	0.6825	0.7017	0.7260	0.7485	0.7608	0.7839	0.8051	0.7964	0.7726	0.7602	0.7463	0.7218	0.7061
4000.00	0.6041	0.6412	0.6857	0.7013	0.7258	0.7477	0.7647	0.7904	0.8223	0.8489	0.8293	0.8155	0.8052	0.7976	0.7607	0.7491
5000.00	0.6204	0.6667	0.7051	0.7278	0.7495	0.7629	0.7886	0.8264	0.8623	0.8889	0.8693	0.8410	0.8356	0.8249	0.8015	0.7814
6000.00	0.6471	0.6841	0.7218	0.7427	0.7684	0.7828	0.8156	0.8484	0.8923	0.9089	0.8893	0.8710	0.8656	0.8649	0.8423	0.8297
7000.00	0.6422	0.6848	0.7247	0.7418	0.7669	0.7828	0.8156	0.8554	0.8923	0.9089	0.8893	0.8710	0.8656	0.8649	0.8591	0.8297
8000.00	0.6473	0.6839	0.7258	0.7437	0.7669	0.7828	0.8156	0.8554	0.8923	0.9089	0.8893	0.8710	0.8656	0.8649	0.8591	0.8297

To calculate the fuel injection quantity within the red circular area, for example, at 6929 RPM and 475 Map, because it is rich, and needs to reduce the value of this area to get less fuel quantity, according to the O2 sensor factor; you can multiply the value of "fLc" or "LamWO2" to the cell value and hit "ENTER". See below:

Note: the VE table and LOAD table can be calibrated "on-the-fly", meaning, it takes effect immediately.



Y/X	300.00	350.00	400.00	430.00	460.00	500.00	550.00	600.00	650.00	700.00	750.00	800.00	850.00	900.00	970.00	1050.00
1200.00	0.4627	0.5093	0.5419	0.5756	0.5962	0.6118	0.6293	0.6440	0.6634	0.6849	0.6841	0.6774	0.6546	0.6348	0.6159	0.6059
1400.00	0.4837	0.5293	0.5619	0.5856	0.5992	0.6218	0.6453	0.6675	0.6807	0.6911	0.6941	0.6784	0.6562	0.6362	0.6180	0.6059
1650.00	0.5037	0.5483	0.5819	0.6056	0.6262	0.6418	0.6653	0.6835	0.6989	0.6976	0.7043	0.6812	0.6661	0.6581	0.6359	0.6159
2000.00	0.5237	0.5663	0.6019	0.6256	0.6462	0.6612	0.6853	0.7044	0.7137	0.7182	0.7161	0.6937	0.6758	0.6744	0.6491	0.6291
2500.00	0.5437	0.5863	0.6219	0.6456	0.6662	0.6818	0.7053	0.7273	0.7373	0.7484	0.7412	0.7204	0.7019	0.6888	0.6616	0.6480
3000.00	0.5639	0.6037	0.6441	0.6698	0.6892	0.7032	0.7232	0.7498	0.7674	0.7843	0.7742	0.7475	0.7350	0.7221	0.7069	0.6865
3500.00	0.5898	0.6246	0.6673	0.6825	0.7017	0.7260	0.7485	0.7608	0.7839	0.8051	0.7964	0.7726	0.7602	0.7463	0.7218	0.7061
4000.00	0.6041	0.6412	0.6857	0.7013	0.7258	0.7477	0.7647	0.7904	0.8223	0.8489	0.8293	0.8155	0.8052	0.7976	0.7607	0.7491
5000.00	0.6204	0.6667	0.7051	0.7278	0.7495	0.7629	0.7886	0.8264	0.8623	0.8889	0.8693	0.8410	0.8356	0.8249	0.8015	0.7814
6000.00	0.6471	0.6841	0.7218	0.7427	0.5800	0.6000	0.8156	0.8484	0.8923	0.9089	0.8893	0.8710	0.8656	0.8649	0.8423	0.8297
7000.00	0.6422	0.6848	0.7247	0.7418	0.6000	0.6200	0.8156	0.8554	0.8923	0.9089	0.8893	0.8710	0.8656	0.8649	0.8591	0.8297
8000.00	0.6473	0.6839	0.7258	0.7437	[illegible]	[illegible]	0.8156	0.8554	0.8923	0.9089	0.8893	0.8710	0.8656	0.8649	0.8591	0.8297

You can use the same method to calibrate all other cells of the table, when you have finished the calibration, the table must be manually made smooth and burn it to ECU. It is critical to have a smooth transition from on operating point (one cell) to the next in the VE table, never ever leave a spike in the table (a good way is to export the table to EXCEL and plot 3D surf, and manually smoothen out any spikes). Spikes are no good for engine controls.

You can also use the real-time lambda (ALM output) to calibrate the table. Make sure to use the engine steady-state running signals.

The method to calibrate the MAP_LdTp_Tps_N (Alpha/N model) is same as MAP_fVe_Map_N, only different in the dependency of "Tps" and "N", throttle position and RPM.

Note: **To make it easy on the time-consuming tuning, understand you don't want to tune each cell, we developed the "EXPORT" functions of the tables. You can modify the whole table in Microsoft EXCEL!**

Right click the table, select "export", it will save the table in the CSV format. And you can do the editing in Excel, which most users are very familiar. After you done the editing in EXCEL, you right click the table in HAECAL again, and select "import", select the CSV file you saved, and it will import the whole table back in.

It is so easy to edit the table in EXCEL that it has become a must to use this feature to do the tuning. You can also do the 3D plots, and use EXCEL math functions to tweak the cell values, and smooth out the spikes.

You can calibrate CUR_TpsUnTp_N to choose when to use Alpha/N model.

If you want us to help you to tune the EFI, Please log data at different conditions, and have comparativel steady state driving at different throttle positions.

For two-stroke engines, the default fuel mapping is Alpha/N method, "RAM_MAP_LdTp_Tps_N", so if you want to calibrate the fuel quantity for a 2-stroke engine, you just modify the LOAD table only. The procedure is same as the four-stroke system.

Some notes before tuning the fuel mapping tables:

1) **The engine has finished warm-up, this means it doesn't have other enrichment factor. You can see the value of "fPreCtl", when the fPreCtl is 1, you can go to tune the tables.**
2) **The fuel pressure must be enough and stable, the fuel pressure is about 3Bar (43psi).**
3) **When you tune the value of the fuel tables, you need read the lambda (LamWO2) in the stable condition; stable throttle position and RPM, not use the lambda in accelerate and decelerate to tune the tables.**

Chapter 6 Ignition Timing Tuning

Pickup (sensor) to TDC angle

If you are using our ECU and our CDI to control the ignition timing (spark advance), based on a single pickup (sensor), you must identify the offset angle between the pickup pulse and the engine TDC.

Only by knowing, this offset angle can the ECU control the CDI to fire the ignition at the right timing.

This offset angle is usually about **29** degree crankshaft angle before the TDC, for most CDI ignition systems.

If you don't know this offset, or if you have trouble to figure out this offset angle, you better keep the stock ignition system and use the EFI to control fuel only. This way, you can start the engine, and worry about the ignition system later.

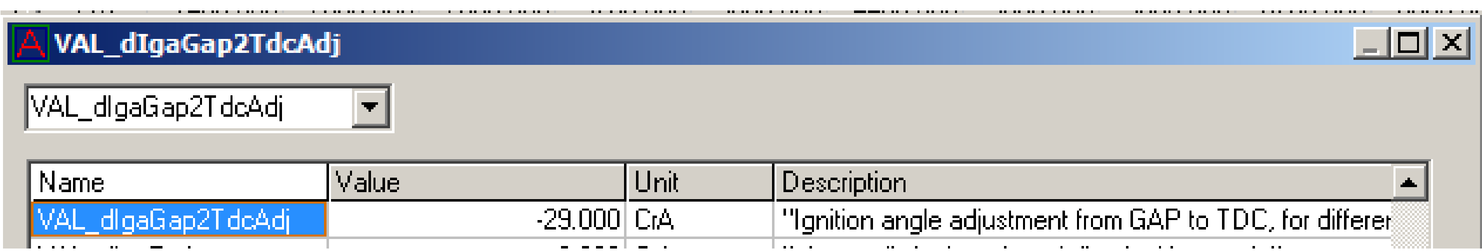
For the information of how to configure a Crank sensor with a multi-tooth trigger wheel, refer to Chapter 2.

How do I set the offset angle in HAECAL?

Please add this advanced calibration variable in HAECAL:

HAECAL → Advanced → Add Advanced Calibrations:

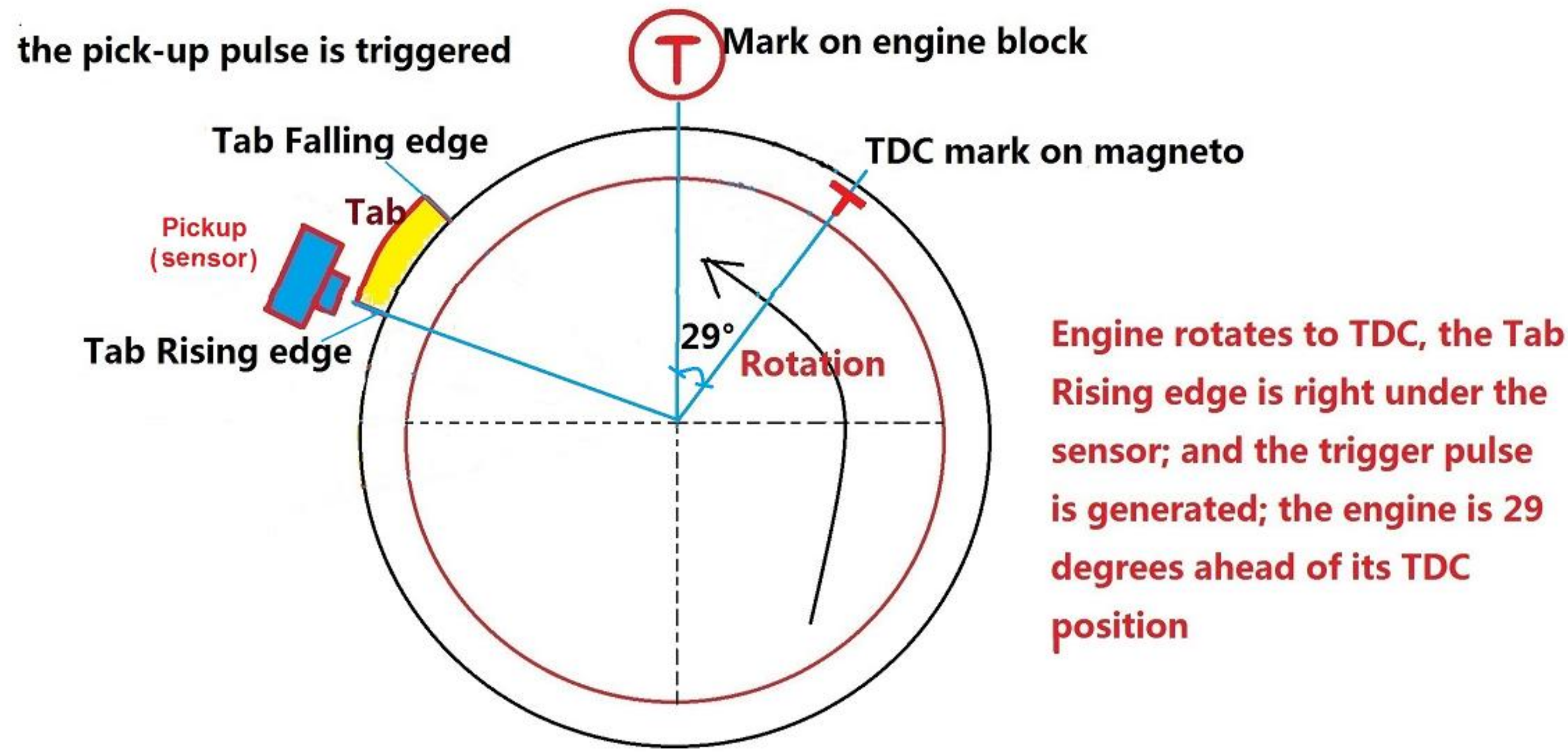
Default: VAL_dIgaGap2TdcAdj = **-29** CrA (crank-shaft angle)

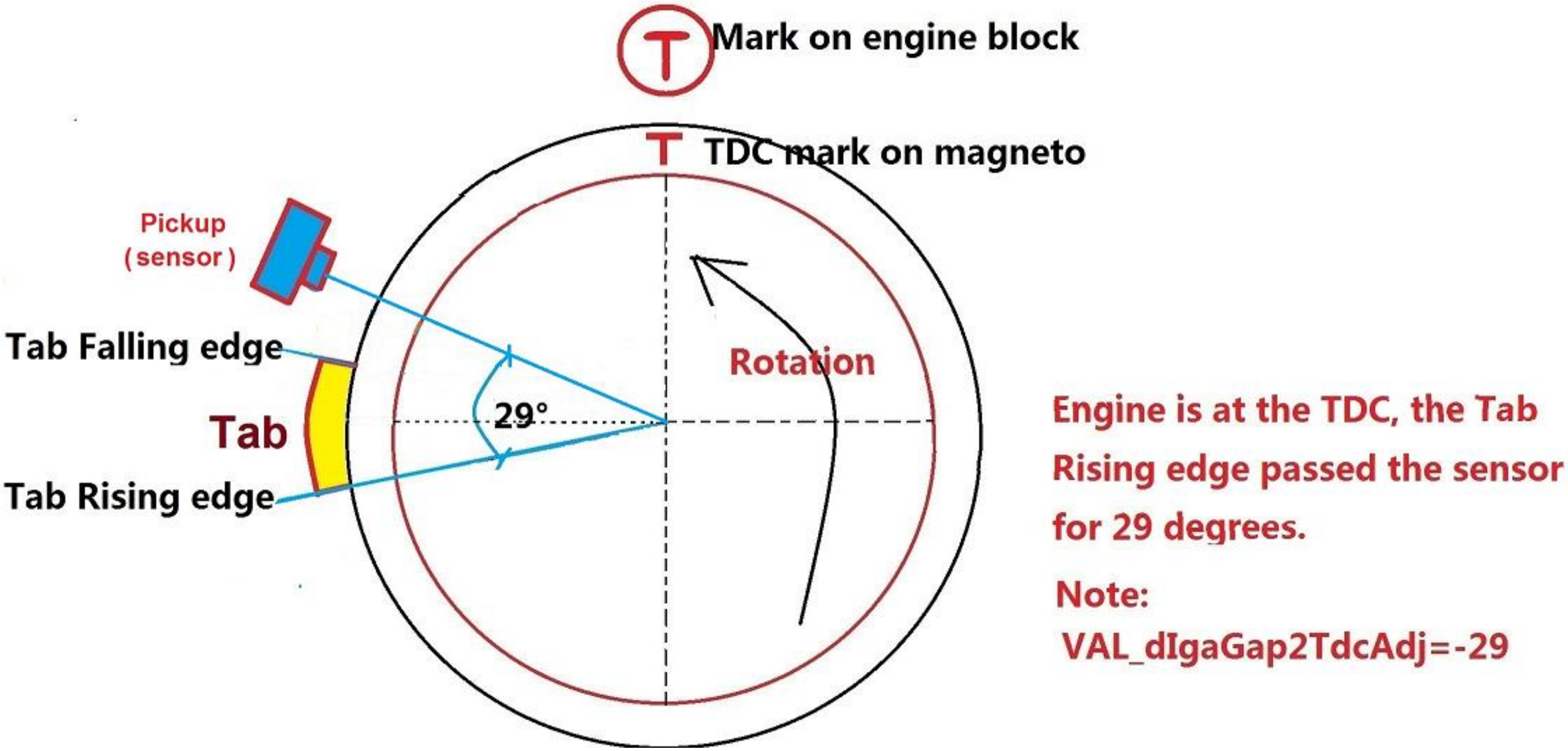


This defines the offset between your trigger pulses to TDC.

The default is -29 CrA. That means 29 degrees in advance to the TDC.

How to find the offset angle:





Note: **It is a negative value. -29 degree of crankshaft angle (CrA); here negative means before TDC.**

This number (-29) is OK for most scooter/bike engines, so you don't have to change.

You can use a timing light to verify whether the ignition angle is close to what should be.

Note: **VAL_dIgaGap2TdcAdj has been changed to the positive value with our newest software.**

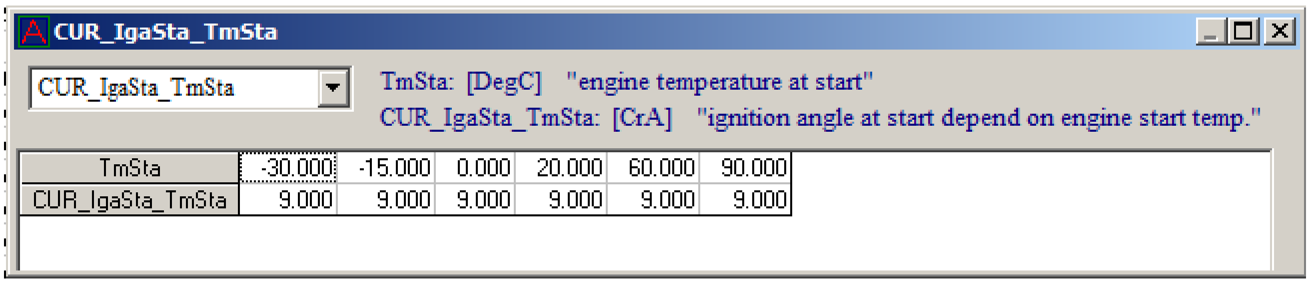
You can find out whether it should be a positive value with your A2L file, such as:"S33_H1_L48_Cr17_AT3.4_E4.a2l" if the number is bigger than "Cr17", for example, "Cr18 or Cr19…." it means it is the newer software; the VAL_dIgaGap2TdcAdj should be positive value.

Start ignition angle

If you use HAE' CDI to control ignition, the default start ignition angle may be not suitable, if it has start ignition issue, you can try to modify the start ignition angle table Ignition; it is based on the engine temperature at start.

Please add this advanced calibration variable in HAECAL:

HAECAL → Advanced → Add Advanced Calibrations:



TmSta	-30.000	-15.000	0.000	20.000	60.000	90.000
CUR_IgaSta_TmSta	9.000	9.000	9.000	9.000	9.000	9.000

Angle in start: CUR_IgaSta_TmSta

When engine go to start, the ignition angle is the value of CUR_IgaSta_TmSta

Basic Ignition Angle mapping

Basic ignition angle is the spark advance that you can run the engine at the max torque (at given LOAD and RPM), without causing the knock.

Actually it is not 100% optimized. There are 2-5 degrees of buffer reserved to protect the engine. You cannot run 100% optimized spark advance unless you have a good knock control systems (with knock sensors, for additional cost, etc).
There is a 16x12 ignition angle table (based on RPM and Load) for you to tune at different operating conditions. There are other tables, to adjust ignition angles dependent on temperatures, AFR, altitude, etc. It is fully programmable ignition control systems, compared to a pre-set factory ignition curve only dependent on the RPM.

The 16x12 basic ignition table is dependent on RPM and Load, and you can tune it to whatever curve, or actually "surf" as you want. Factory CDI timing is usually a fixed curve - more advance at higher RPM. That's it.

ECU controlled ignition angle has much more flexibility: it also takes into account of AFR (rich or lean), temperatures (cold or hot), altitude, etc. You need to read through the strategy book to understand how these factors affect the final ignition angle output.

In HAECAL:

Menu → calibrations → ignition system → basic ignition angle →

Basic ignition angle

RAM_MAP_Iga_N_Ld

X: Ld_b, [%] "Load, or relative air charge (byte)"

Y: N_b, [Rpm] "engine speed, byte value"

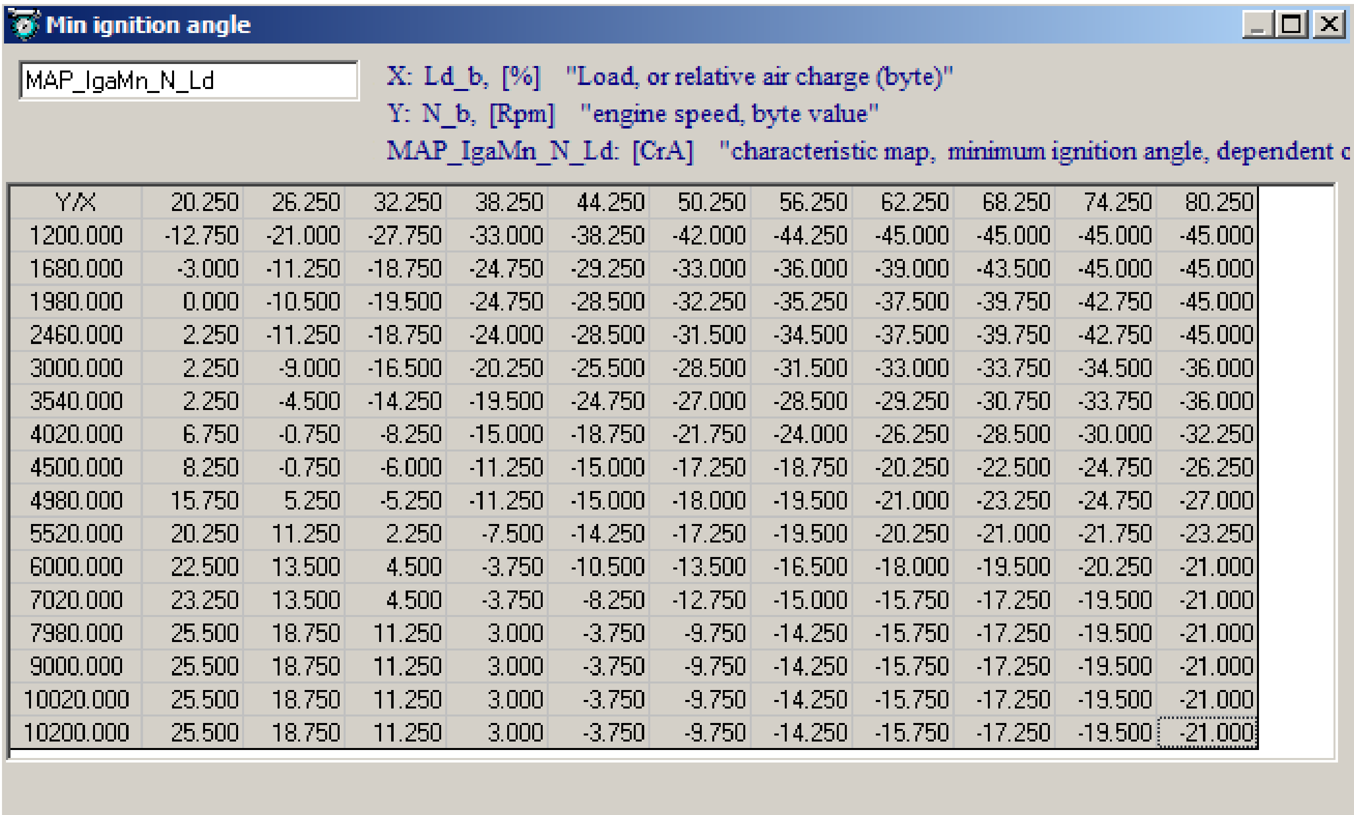
MAP_Iga_N_Ld: [CrA] "characteristic map, basic ignition angle, dependent on engine sp

Y/X	20.250	26.250	32.250	38.250	44.250	50.250	56.250	62.250	68.250	74.250	80.250
1200.000	12.000	12.000	12.000	12.000	12.000	12.000	12.000	12.000	12.000	12.000	12.000
1680.000	15.000	15.000	15.000	15.000	15.000	15.000	15.000	15.000	15.000	15.000	15.000
1980.000	20.250	20.250	20.250	20.250	20.250	20.250	20.250	20.250	20.250	20.250	20.250
2460.000	22.500	22.500	22.500	22.500	22.500	22.500	22.500	22.500	22.500	22.500	22.500
3000.000	24.750	24.750	24.750	24.750	24.750	24.750	24.750	24.750	24.750	24.750	24.750
3540.000	30.000	30.000	30.000	30.000	30.000	30.000	30.000	30.000	30.000	30.000	30.000
4020.000	30.750	30.750	30.750	30.750	30.750	30.750	30.750	30.750	30.750	30.750	30.750
4500.000	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250
4980.000	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250	32.250
5520.000	33.000	33.000	33.000	33.000	33.000	33.000	33.000	33.000	33.000	33.000	33.000
6000.000	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250
7020.000	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250
7980.000	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250
9000.000	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250
10020.000	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250
10200.000	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250	35.250

Note: **The default ignition table is very close to the best ignition angle, so if you want to change it, please adjust it in a small step and range, such as 1 degree or 3 degrees (For example, if the default value is 36, you can change it to 38). It protects the engine from too aggressive ignition angle, which can damage the engine.**

Minimum Ignition Angle

Minimum ignition angle is the latest spark retard angle (refer to TDC) that you can still run the engine without misfiring.



Y/X	20.250	26.250	32.250	38.250	44.250	50.250	56.250	62.250	68.250	74.250	80.250
1200.000	-12.750	-21.000	-27.750	-33.000	-38.250	-42.000	-44.250	-45.000	-45.000	-45.000	-45.000
1680.000	-3.000	-11.250	-18.750	-24.750	-29.250	-33.000	-36.000	-39.000	-43.500	-45.000	-45.000
1980.000	0.000	-10.500	-19.500	-24.750	-28.500	-32.250	-35.250	-37.500	-39.750	-42.750	-45.000
2460.000	2.250	-11.250	-18.750	-24.000	-28.500	-31.500	-34.500	-37.500	-39.750	-42.750	-45.000
3000.000	2.250	-9.000	-16.500	-20.250	-25.500	-28.500	-31.500	-33.000	-33.750	-34.500	-36.000
3540.000	2.250	-4.500	-14.250	-19.500	-24.750	-27.000	-28.500	-29.250	-30.750	-33.750	-36.000
4020.000	6.750	-0.750	-8.250	-15.000	-18.750	-21.750	-24.000	-26.250	-28.500	-30.000	-32.250
4500.000	8.250	-0.750	-6.000	-11.250	-15.000	-17.250	-18.750	-20.250	-22.500	-24.750	-26.250
4980.000	15.750	5.250	-5.250	-11.250	-15.000	-18.000	-19.500	-21.000	-23.250	-24.750	-27.000
5520.000	20.250	11.250	2.250	-7.500	-14.250	-17.250	-19.500	-20.250	-21.000	-21.750	-23.250
6000.000	22.500	13.500	4.500	-3.750	-10.500	-13.500	-16.500	-18.000	-19.500	-20.250	-21.000
7020.000	23.250	13.500	4.500	-3.750	-8.250	-12.750	-15.000	-15.750	-17.250	-19.500	-21.000
7980.000	25.500	18.750	11.250	3.000	-3.750	-9.750	-14.250	-15.750	-17.250	-19.500	-21.000
9000.000	25.500	18.750	11.250	3.000	-3.750	-9.750	-14.250	-15.750	-17.250	-19.500	-21.000
10020.000	25.500	18.750	11.250	3.000	-3.750	-9.750	-14.250	-15.750	-17.250	-19.500	-21.000
10200.000	25.500	18.750	11.250	3.000	-3.750	-9.750	-14.250	-15.750	-17.250	-19.500	-21.000

Other Ignition Angle Adjustment

Basic ignition angle will be adjusted according to temperature, altitude, lambda, knock limit, etc. That's probably why sometime you see the actual ignition angle is not the one you filled in the basic ignition able. It can be something in the middle of basic angle and minimum angle.

Chapter 7 Advanced Tuning

7.1 Transient fuel

To improve the performance of the throttle response, you need to change the transient fuel.

When you open the throttle, if you feel the bike bogs or pops or doesn't have much power, it may be too rich or too lean, you can read the AFR/lambda by using the ALM wideband senor. Then do some advance tuning according to the below instructions. Add two calibration variables by using the advanced calibration menu, “CUR_Wf_Ldp" and" VAL_fTrsRedS"

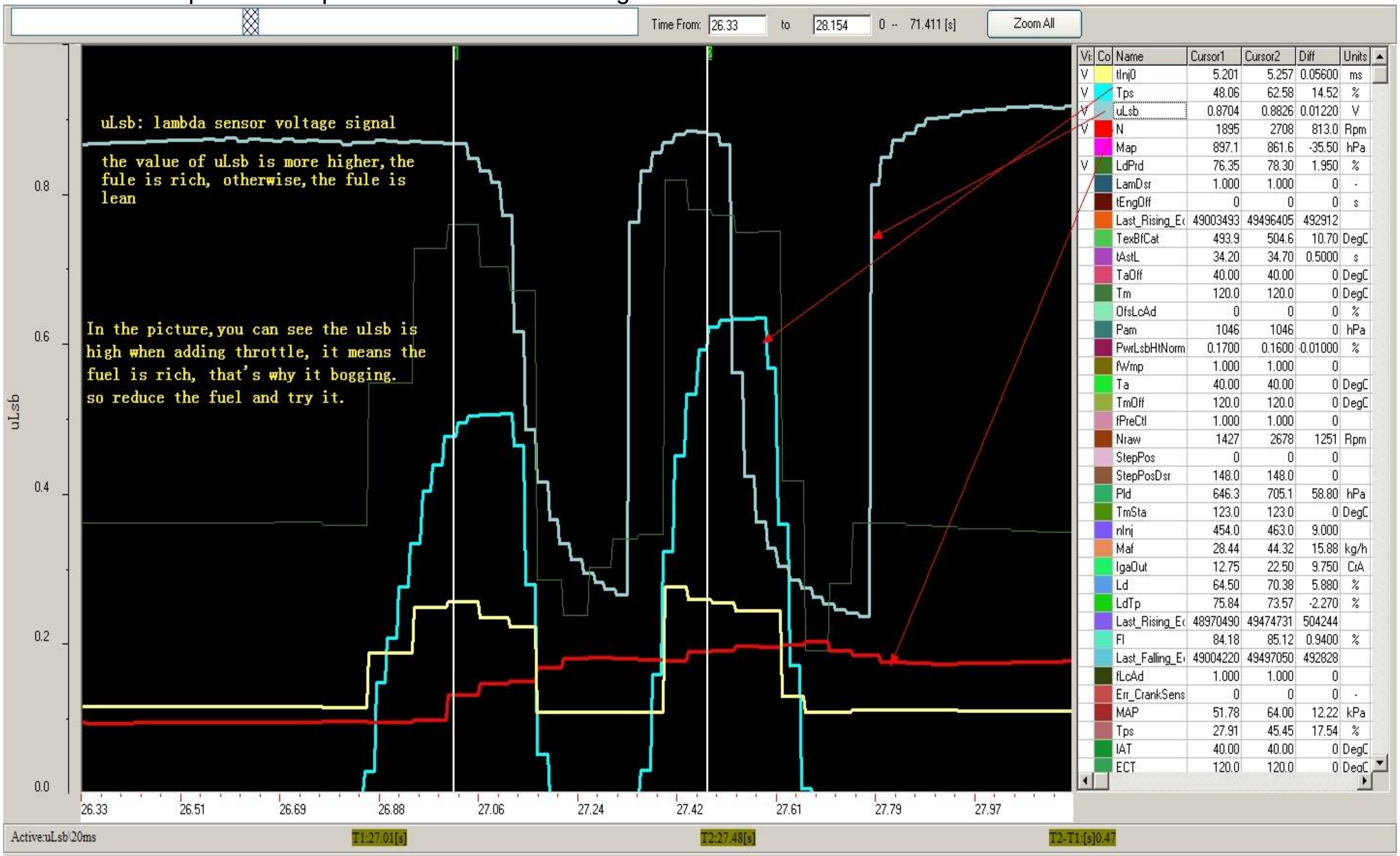
"CUR_Wf_Ldp": the wall film fuel quantity during the transient, depend on the predicted LOAD.

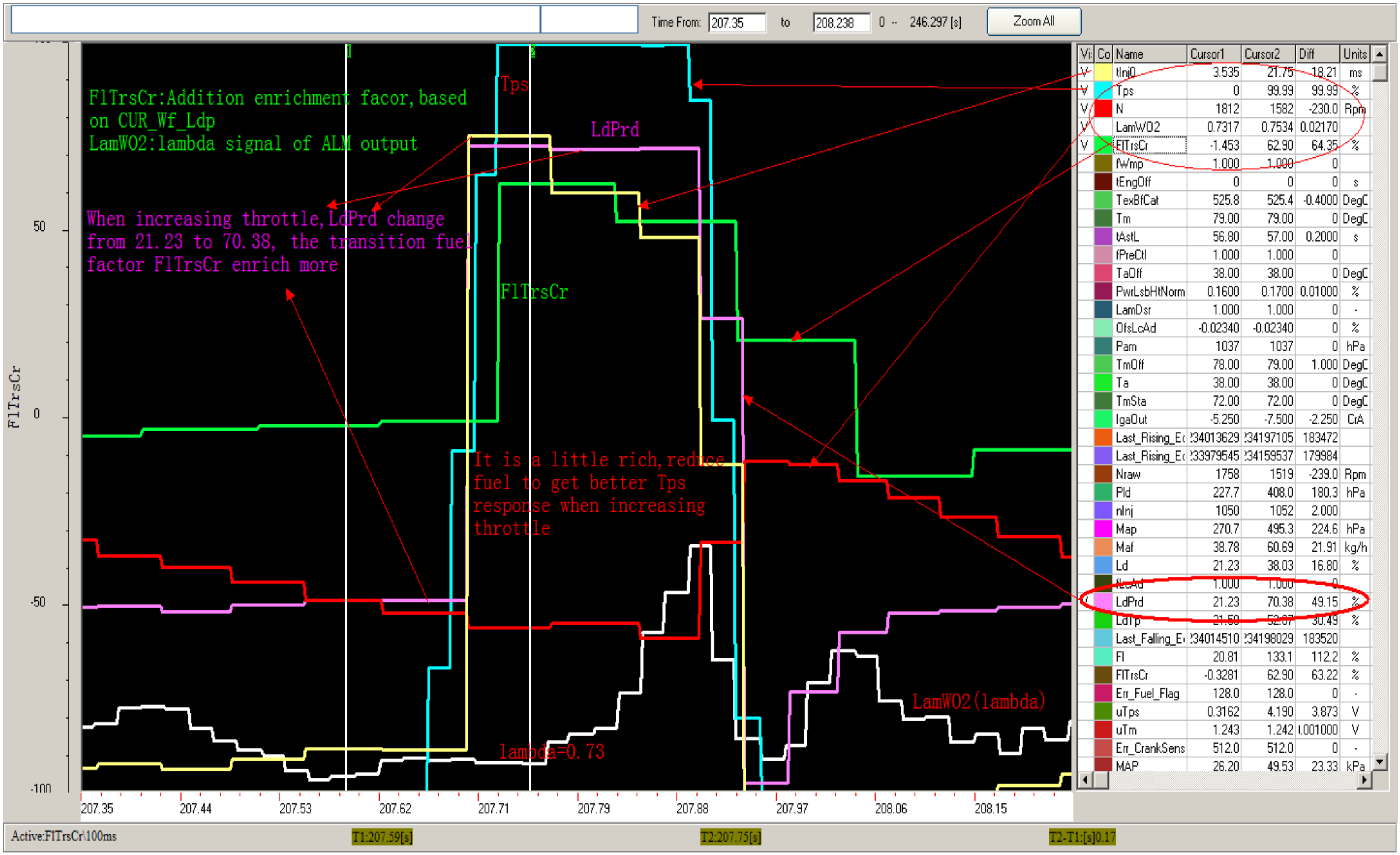
“VAL_fTrsRedS": reduction factor of the short-term transient fuel

Our transient fuel tuning, is based on the Wall film model. Wall film fuel is the fuel stuck on the intake manifold wall or combustion chamber corner, clearance, which is not participating the combustion. Then you open the throttle or close the throttle, the wall film fuel will increase or decrease, and therefore change your desired AFR.
The transient fuel tuning is to model this wall film at different Load conditions, and then compensate the fuel that is caused by the wall film fuel changing.

The wall film fuel is modeled based on the “**Predicted LOAD**”. Predicted load is the engine load of the next combustion cycle. You must use predicted load, not the current load, for the next cycle’s fuel calculation. This is a whole advanced engine control theory, if you are interested.

Below are a couple of examples of transient fuel tuning.




Time From: 207.35 to 208.238 0 -- 246.297 [s]
Zoom All
FlTrsCr:Addition enrichment facor,based
on CUR_Wf_Ldp
LamWO2:lambda signal of ALM output
When increasing throttle,LdPrd change
from 21.23 to 70.38, the transition fuel
factor FlTrsCr enrich more
Tps
LdPrd
FlTrsCr
It is a little rich,reduce
fuel to get better Tps
response when increasing
throttle
lambda=0.73
LamWO2(lambda)
FlTrsCr
Active:FlTrsCr\100ms
T1:207.59[s]
T2:207.75[s]
T2-T1:[s]0.17

You can calibrate the "CUR_Wf_Ldp" value based on the LdPrd, you can read the value of LdPrd in HAECAL.

CUR_Wf_Ldp

CUR_Wf_Ldp

LdPrd: [%] "load predicted for injection time calculation"
CUR_Wf_Ldp: [%] "characteristic curve: fuel wall film, depenedent on Load predicted "

LdPrd	0.00	6.00	12.00	18.00	24.00	30.00	36.00	42.00	48.00	54.00	60.00	66.00	72.00	78.00	84.00	90.00	96.00	102.00
CUR_Wf_Ldp	0.00	4.69	12.00	41.06	65.06	90.00	126.94	159.94	180.94	199.88	216.94	232.88	241.88	249.00	257.06	261.94	276.00	296.06

Selected Variables

Vis	Col	Name	Value	Rate	Unit	Min	Max	Discription
		Fl	56.95	Syn	%	0	3072	"Relative fuel mass"
		IgaOut	35.25	Syn	CrA	-96	95.25	"Ignition angle output value"
		Ld	41.25	Syn	%	0	1536	"Load, or relative air charge"
V		LdPrd	41.25	Syn	%	0	1535.977	"load predicted for injection time calculatic
		LdTp	41.25	Syn	%	0	1535.977	"Load through throttle valve"
		Maf	14.57	Syn	kg/h	0	1572	"Mass air flow"
		Map	2559.96	Syn	hPa	0	2560	"Intake manifold pressure measured with M
V		N	2652.25	Syn	Rpm	0	16383	"Engine speed in Rpm"
		nInj	1288.00	Syn		0	65535	"number of injections "
		tInj0	3.37	Syn	ms	0	65.535	"injection time for injector #0 "
		fLc	1.0000	20ms	-	0	2	"Lambda controller output (word)"
		Map	2559.96	20ms	hPa	0	2560	"Intake manifold pressure measured with M
		Nraw	2650.00	20ms	Rpm	0	16383.75	"Engine speed in rpm from LLD"
V		Tps	12.50	20ms	%	0	100	"throttle position with respect to lower mec
		UbAdc	12.03	20ms	V	0	25.6	"battery voltage; scanned value of microp
		uLsb	0.4541	20ms	V	-1	4	"Voltage signal from the lambda sensor be
		uMap	4.9707	20ms	V	0	5	"Voltage signal of manifold pressure sensc
		uPot	4.9756	20ms	V	0	5	"ADC-voltage of potentiometer sensor "
		uTps	4.9744	20ms	V	0	5	"ADC-voltage of throttle position sensor "

You can change the wall film quantity according to LdPrd before increasing the throttle and after increasing the throttle. The delta of the wall film quantity at different LdPrd is the base transient fuel. If you see lean conditions at the throttle opening, that means you need to add more wall film at high Load conditions (big LdPrd).

You can also try to adjust the value of "VAL_fTrsRedS" to get different effects of the throttle opening conditions.

"CUR_fwLdpMapTps_N" is a weight factor for PN model and Alpha/N model in transient conditions.
Raise the value of CUR_fwLdpMapTps_N will make TPS-based prediction more weighting, when you increase throttle in acceleration condition.

When you begin to calibrate the transient fuel, the basic injection table must be stable, the engine must run normally. And to calibrate it, you need to repeat the test to get a better performance.

Advanced tuning for throttle blips

If the response of throttle blip is not in favor, and you have the right software version from us for this issue, you can calibrate it by yourself, and find a suitable value to improve the response (usually for the big displacement engine).

Injection compensation variables for throttle blips:

VAL_dtInjTpBlip_Max --- max delta injection pulse width in ms when throttle blip happens.

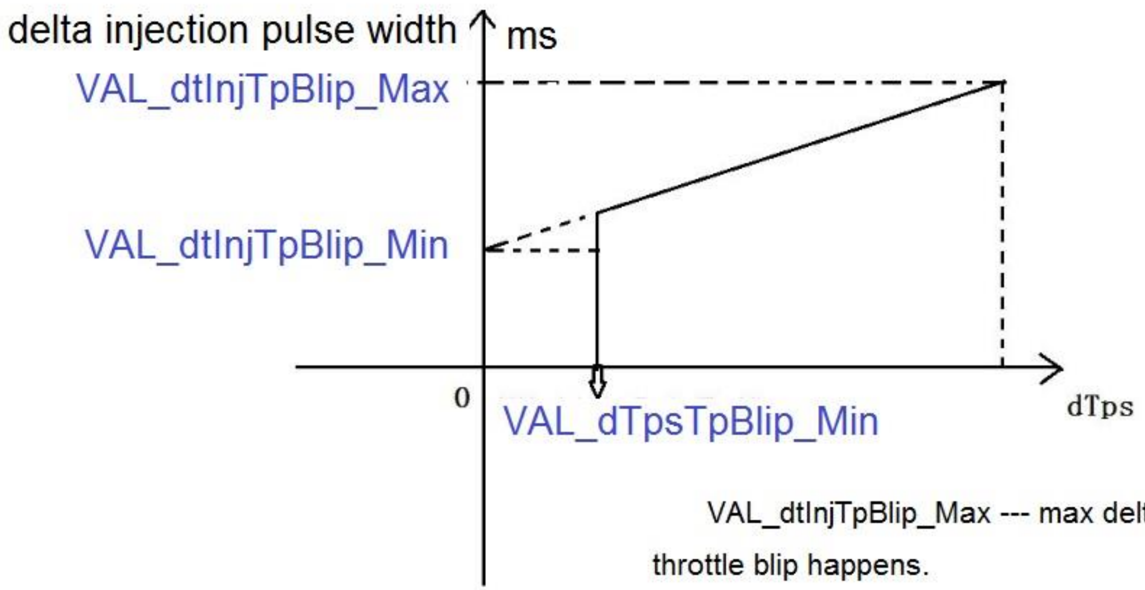
VAL_dtInjTpBlip_Min --- min delta injection pulse width in ms when throttle blip happens.

VAL_dTpsTpBlip_Min --- min delta Tps change in 20ms to recognize the throttle blip.

(If the Tps change is bigger than VAL_dTpsTpBlip_Min, it will enable the function to add fuel.)

VAL_NTpBlip_Max --- max RPM limited when throttle blip

(If the RPM is higher than VAL_NTpBlip_Max when the throttle blips, there is no additional fuel)



VAL_dtInjTpBlip_Max --- max delta injection pulse width in ms when throttle blip happens.

VAL_dtInjTpBlip_Min --- min delta injection pulse width in ms when throttle blip happens.

VAL_dTpsTpBlip_Min --- min delta Tps change in 20ms to recognize the throttle blip.

Dashpot functions for deceleration:

And if you have the software for deceleration to avoid a stall issue with the stepper motor control system, you also can calibrate the below variables.

VAL_McDashpot_DeltaDecrements: the step size of IAC ramp-down, range: 12-24; the bigger the value, the faster the ramp, but more likely to stall.

VAL_McDashpotInitValue: initial position of the IAC motor for dashpot, range 640-320;

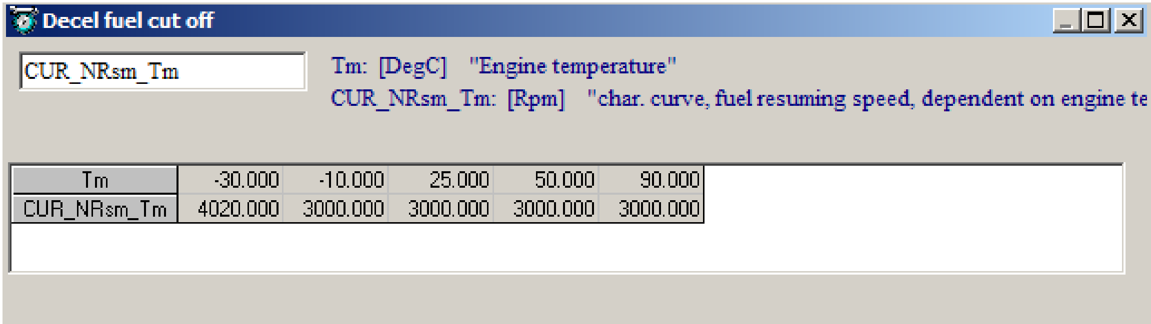
Note: **You can find the variables in HAECAL if you have the correct A2L file, go to menu->Advanced->Add Advanced Calibrations. You need repeat the attempts to tune it.**

7.2 Decel-Fuel-Cut-Off

Our EFI system by default has enabled the decel-fuel-cut-off function. It is RPM dependent, when you release the throttle to idle and RPM is higher than the table value, it will cut the fuel injection.

And you can adjust the table value to decide at what RPM to resume fuel.

Menu->Calibrations->Fuel system->decal-fuel-cut-off



Tm	-30.000	-10.000	25.000	50.000	90.000
CUR_NRsm_Tm	4020.000	3000.000	3000.000	3000.000	3000.000

Note: **So if you want to disable decal-fuel-cut-off, you can adjust the table value to a bigger one. But don't disable the decal-fuel-cut-off, if you are running in a mountain area, and it may damage the engine in downhill driving because a lot of un-burnt fuel getting into the exhaust system.**

7.3 Driver's desired lambda

1) Driver desired lambda is only feasible after you have the system running stably in close loop controls and the ECU has done the self-tuning. That means, the ECU can control the AFR in the vicinity of stoic AFR (14.7 for gasoline). Only after that, ECU can command a meaningful AFR other than the stoic. Otherwise, the driver desired lambda could have a big deviation from the actual lambda value.

2) Driver desired lambda is meant to give the user a way to command the AFR other than Stoic, either for performance purpose (more likely) or fuel saving purpose.

3) Default value of desired lambda is as below table. It is enriched at high end. 1 means stoic AFR (14.7 for gasoline).

4) Desired lambda table can be activated by turning on the "RICH" mode, by switching the "Performance switch" to ON position. By default, the engine is running in "ECO" mode; and the desired lambda table is not used.

Some more comments on RICH mode: Rich mode only works well after ECO mode is tuned.

RICH mode is actually an open loop mode, meaning, ECU will command the "desired Lambda” based on the ECO mode tuning. It assumes you have tuned the base mapping like VE table and TPS Load table close to Stoic AFR at all conditions. It then divides the desired lambda on the top of the base fuel quantity.

In RICH mode, you only tune the “desired lambda table", which should between 0.8 to 1.2 ranges. This is nothing but a dividing factor on top of the base fuel mapping.

For example, you base fuel mapping outputs a fuel quantity of 3ms pulse width, and your desired lambda is 0.9; then your final fuel pulse output is 3 / 0.9 = 3.33ms.

Menu->Calibrations->Driver's demand->desired lambda

Desired lambda

MAP_LamDrv_Tps_N

X: N, [Rpm] "Engine speed in Rpm"
Y: TpsEqu, [%] "throttle position equivalent value, including the idle by-pass air flow"
MAP_LamDrv_Tps_N: [-] "characteristic map, Driver desired lambda, dependent on TPS and N"

Y/X	1400.000	2000.000	2600.000	3200.000	3800.000	4400.000	5000.000	5600.000	6200.000	6800.000	7400.000	8000.000
0.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
3.914	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.986	0.986	0.986	0.986
6.526	1.000	1.000	0.993	0.986	0.986	0.986	0.986	0.986	0.952	0.952	0.952	0.952
9.132	1.000	0.993	0.952	0.952	0.952	0.952	0.952	0.952	0.918	0.918	0.918	0.918
13.048	0.986	0.986	0.918	0.918	0.918	0.918	0.918	0.918	0.918	0.885	0.885	0.885
16.963	0.986	0.986	0.918	0.885	0.885	0.885	0.885	0.885	0.885	0.850	0.850	0.850
20.879	0.885	0.952	0.891	0.885	0.850	0.850	0.850	0.850	0.850	0.850	0.850	0.850
24.792	0.885	0.952	0.891	0.878	0.850	0.850	0.850	0.850	0.850	0.844	0.844	0.844
32.617	0.878	0.885	0.885	0.871	0.850	0.844	0.844	0.844	0.844	0.837	0.837	0.837
39.145	0.878	0.885	0.885	0.871	0.850	0.837	0.837	0.837	0.837	0.837	0.837	0.837
45.671	0.878	0.885	0.885	0.864	0.850	0.837	0.837	0.837	0.837	0.830	0.830	0.830
52.193	0.878	0.885	0.878	0.864	0.850	0.830	0.830	0.830	0.830	0.830	0.830	0.830
58.716	0.878	0.885	0.878	0.864	0.850	0.830	0.830	0.830	0.830	0.830	0.830	0.830
65.242	0.878	0.885	0.878	0.864	0.830	0.830	0.830	0.830	0.830	0.830	0.830	0.830
78.290	0.878	0.885	0.878	0.864	0.830	0.830	0.830	0.830	0.830	0.830	0.830	0.830
99.994	0.878	0.885	0.878	0.864	0.830	0.830	0.830	0.830	0.830	0.830	0.830	0.830

7.4 Advance Calibration for engine timing

If you use HAE ECU to control ignition system, you may do some advanced calibration on the ignition system to get a good performance.

Pickup type

1) One pulse per revolution.

VAL_dIgaGap2TdcAdj: "Ignition angle adjustment from GAP to TDC, GAP is pickup pulse location."

This is the first calibration you need to be sure. It determines the engine TDC position in regarding to the pulse generation.

See Chapter 6, how to calibrate this variable.

You can also adjust the injection timing, based on the pickup pulse. It is the mapping of “MAP_InjAngEnd_N_Ld” MAP_InjAngEnd_N_Ld" is the end of injection angle referring to TDC; it means the angle from injection end to the ignition TDC position.

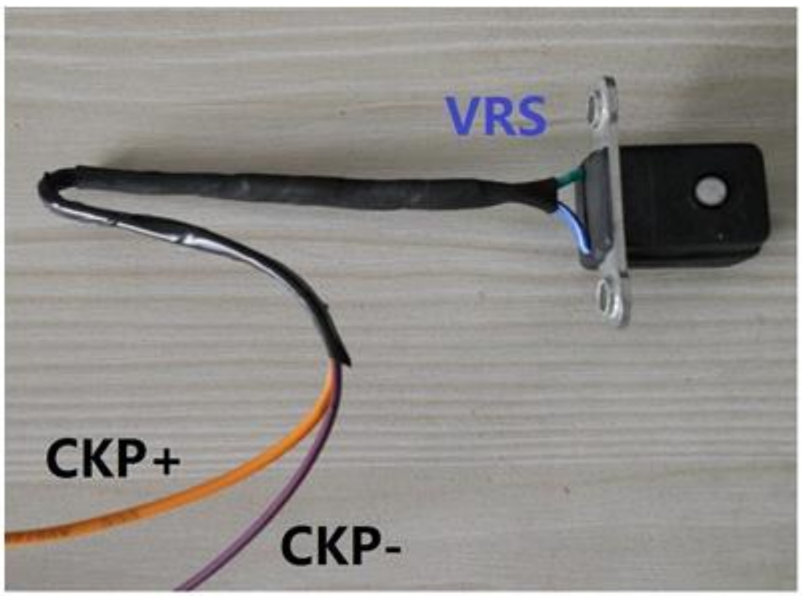
2) How to calibrate the ignition timing and injection timing with a multi-tooth trigger wheel installed on the crankshaft?

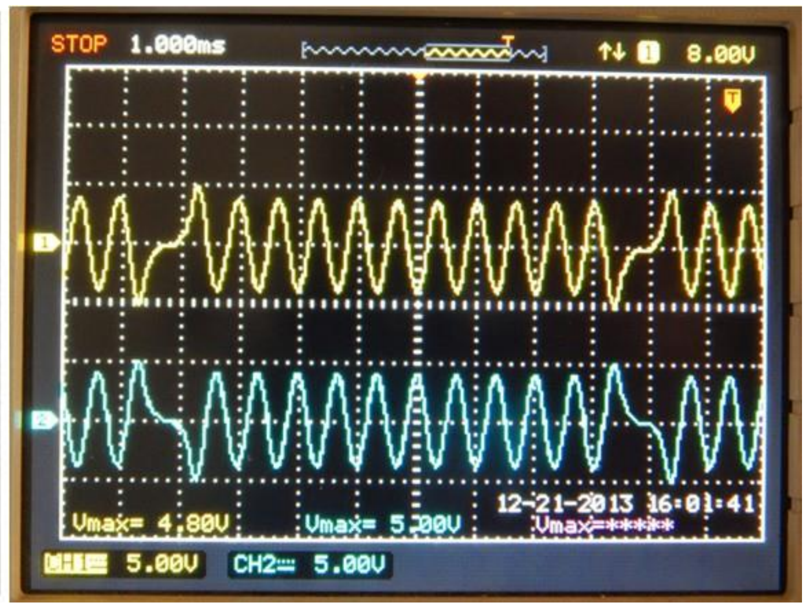
VRS sensor installation for the multi-tooth trigger-wheel

The VRS (Variable Reluctant Sensor) generates AC pulses instead of square waves.

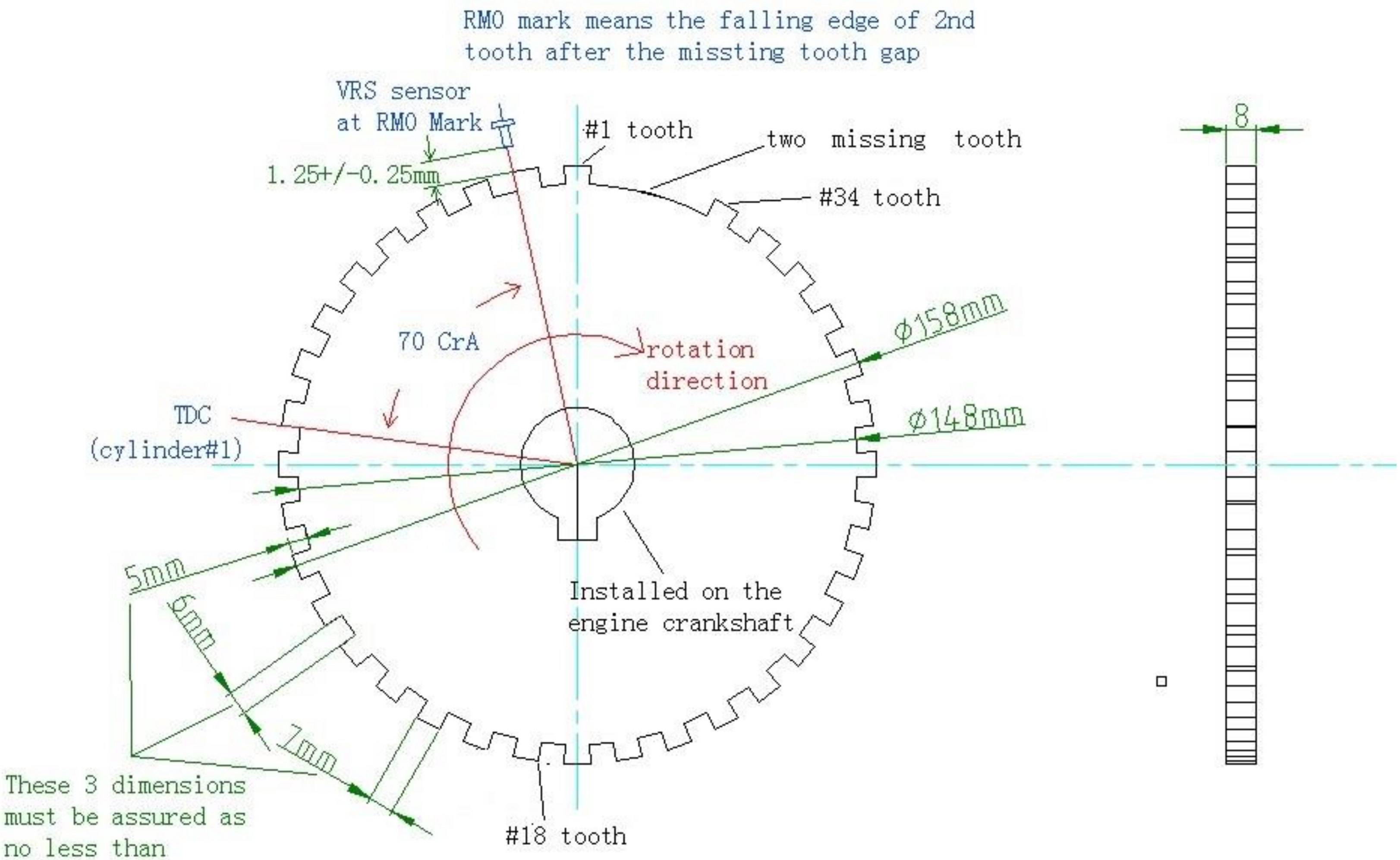
The metal sensing element (sensor core) needs to align well with the multi-tooth wheel. See below for the examples.

The tooth width needs to be bigger than the sensor metal core, otherwise, the VRS signal will not be correct, and ECU will the distorted VRS signal, and EFI could not work.





Measure the signal + wire and signal – wire output with oscilloscope (12-1 tooth-wheel)
[Note:] CKP+: CH1 (yellow); CKP –: CH2 (Blue)



trigger wheel 36-2

Trigger wheels supported

Our EFI works with a wide range of multi-tooth trigger wheels. The typical trigger wheels have 36-2 tooth-wheel, or 12-1, 24-1, 36-1 or 60-2 tooth wheels. Here, “36-2” tooth wheel means there are totally 36 teeth with 2 missing teeth.
12-1 means totally 12 teeth with 1 missing tooth, and so on. Our ECU supports different combinations of total teeth and missing tooth setting. The total number of teeth cannot be more than 60; the missing tooth can be 1 or 2.

Usually the higher the engine speed redline, the less the teeth. This is because at higher speed, the tooth signal becomes smaller and smaller, and eventually ECU cannot read it anymore. For most 2 stroke engines, 12-1 tooth wheel , or 24-1 tooth wheel is typical, because usually 2 stroke engines are high speed ones. For most 4-stroke motorcycle engines, the tooth wheel is 36-2 setting.

If you want to have a good ignition control system, a high-tooth wheel is definitely way to go. And if you have the mechanical capability to add the tooth wheel to your engine, we recommend you to do so.
For multi-cylinder engines, and special engines, such as a V-twin engine, if you want EFI system to control the ignition, you need to install the tooth wheel.
Our CDI controlled ignition system is meant to be used for single cylinder engine. And the accuracy of the ignition timing is not as good as the trigger wheel based.

Let’s see one example with 36-2 tooth-wheel.
Some calibration variables need to be defined:

VAL_nTeethTot: ----- Total number of teeth on the tooth-wheel.

VAL_nTeethMiss: ---- number of missing tooth on the tooth-wheel.

VAL_ToothTDC: ------- tooth number of the ignition TDC for single cylinder.

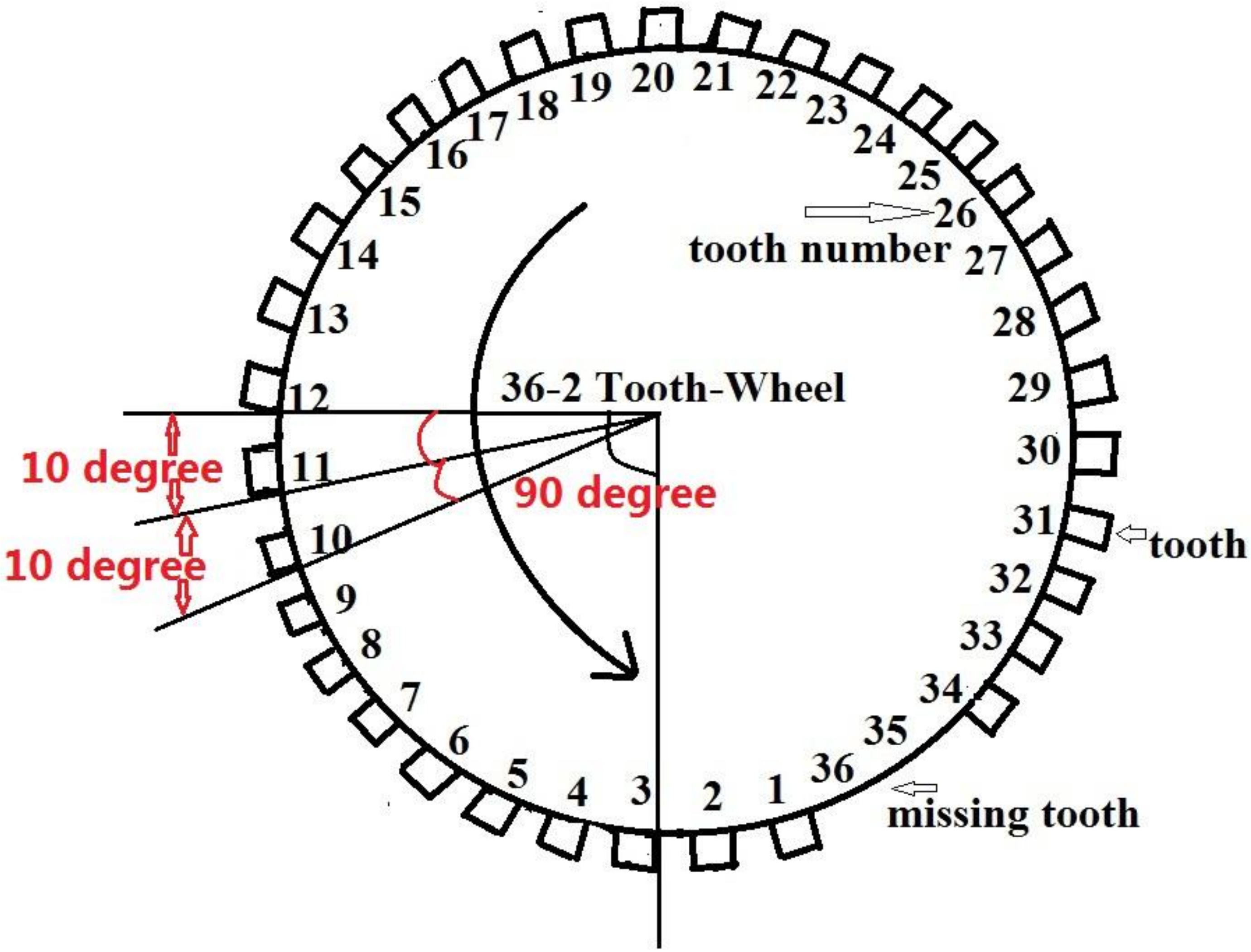
VAL_ToothTDC2:------ tooth number of Cylinder #2 ignition TDC for two cylinder engines.

MAP_Iga_N_Ld: --------basic ignition angle table

MAP_InjAngEnd_N_Ld- injection end angle table

Below is an illustration of the 36-2 trigger wheel and the engine positions.

Each tooth is 10 degree (360/36=10). The first tooth after missing teeth is marked as the #1 tooth.


36-2 Tooth-Wheel
tooth number
tooth
missing tooth
10 degree
10 degree
90 degree
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36

First you need calibrate the parameters of the tooth-wheel:

VAL_nTeethTot=36;

VAL_nTeethMiss=2;

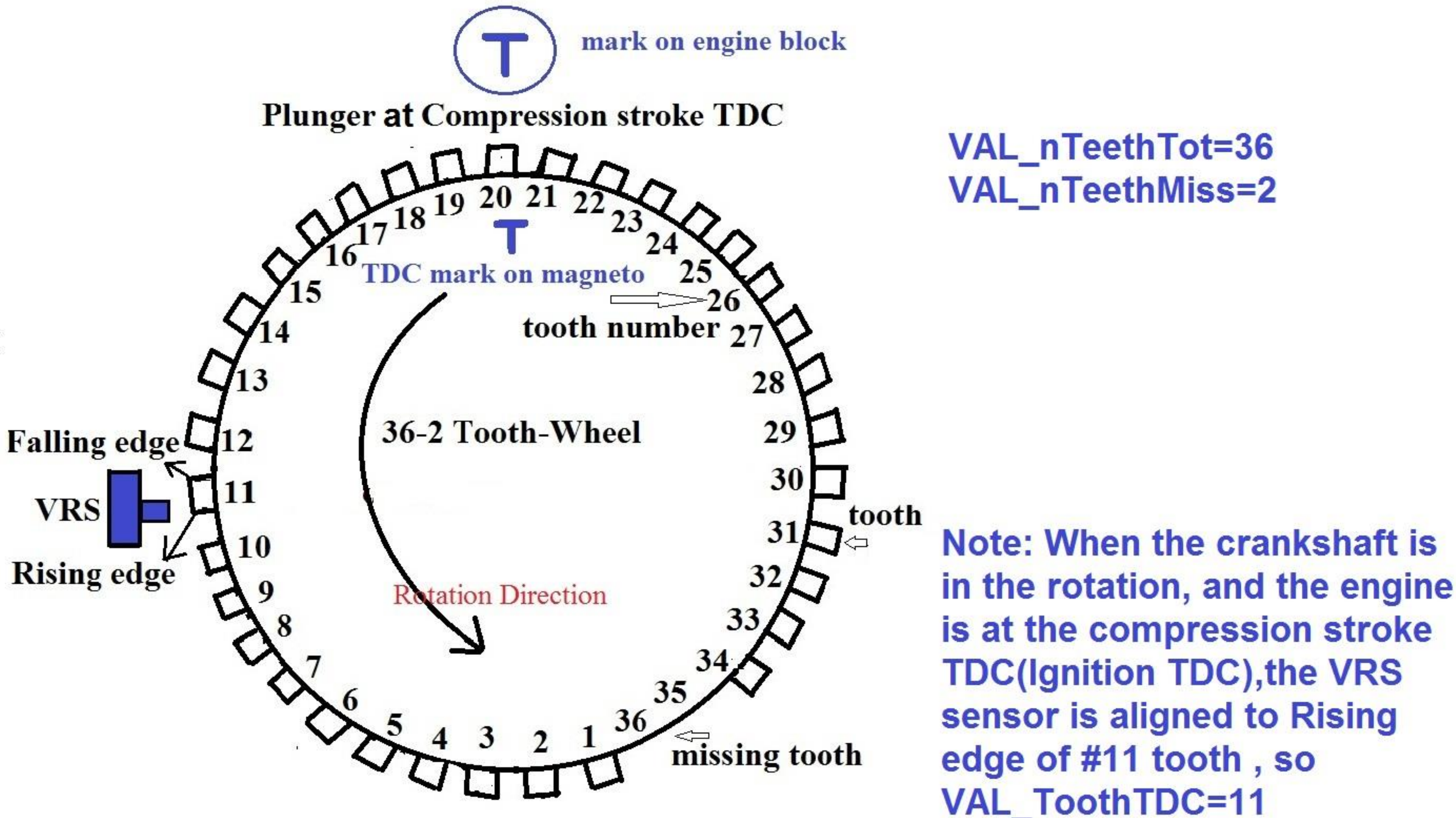
Four-stroke engine:

1) Single cylinder engine:

You need calibrate VAL_ToothTDC

VAL_ToothTDC is the tooth number of the ignition TDC.

To find it: Rotate the engine crankshaft and make the piston at the TDC position, and count the tooth number after the missing tooth, which tooth is aligned to the VRS sensor. See the below picture, it is very clear.



In the above picture, you can see the #11 tooth is at TDC, so VAL_ToothTDC=11

2) Tow-cylinder engines:

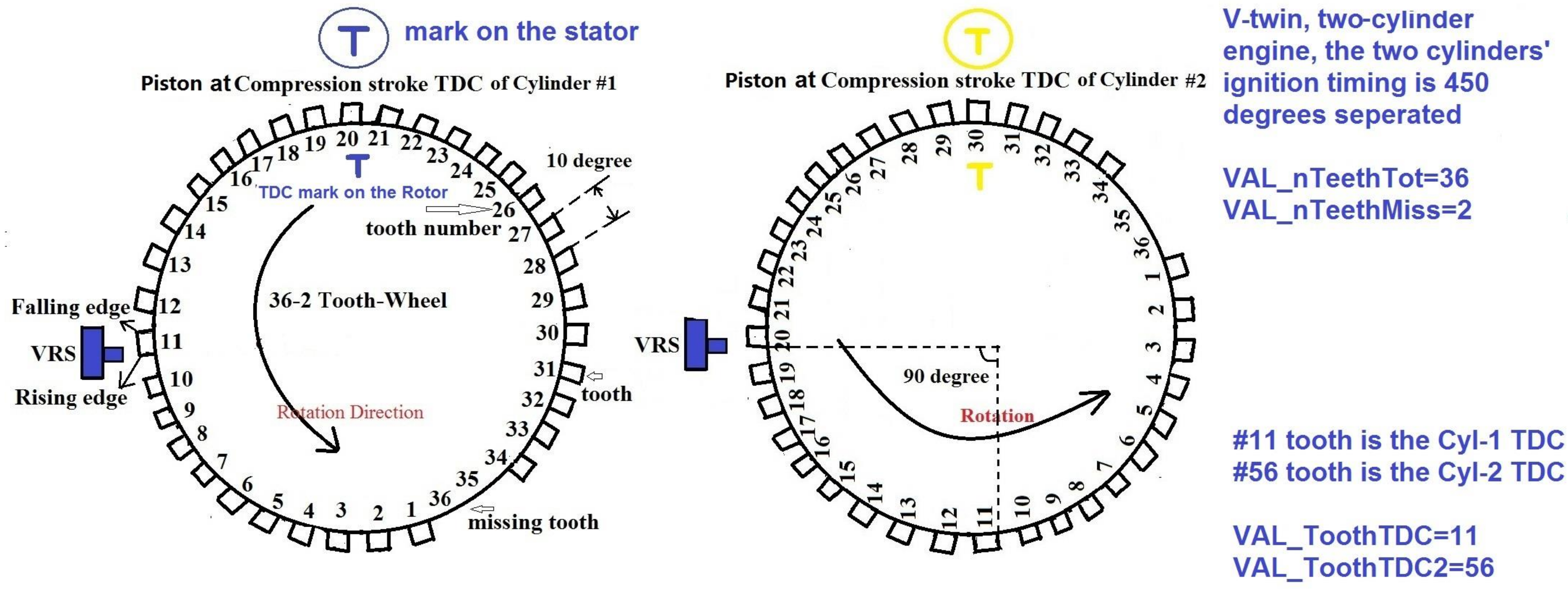
You need calibrate VAL_ToothTDC, VAL_ToothTDC2

The method to calibrate two-cylinder engine is same as a single cylinder engine. One important thing is to find out how many degrees the two cylinders are separated.

Use a V-twin engine as an example; the two cylinders are separated by 90 degrees.

For 36-2 tooth-wheel, V-twin, a two-cylinder engine, the ignition TDC tooth of cylinder -1 is #11, and the two cylinders are 90 degree apart, so the ignition TDC of cylinder #2 is 450 (360+90) degrees after the Cyl-1's. So the ignition TDC tooth of Cyl-2 is #56.

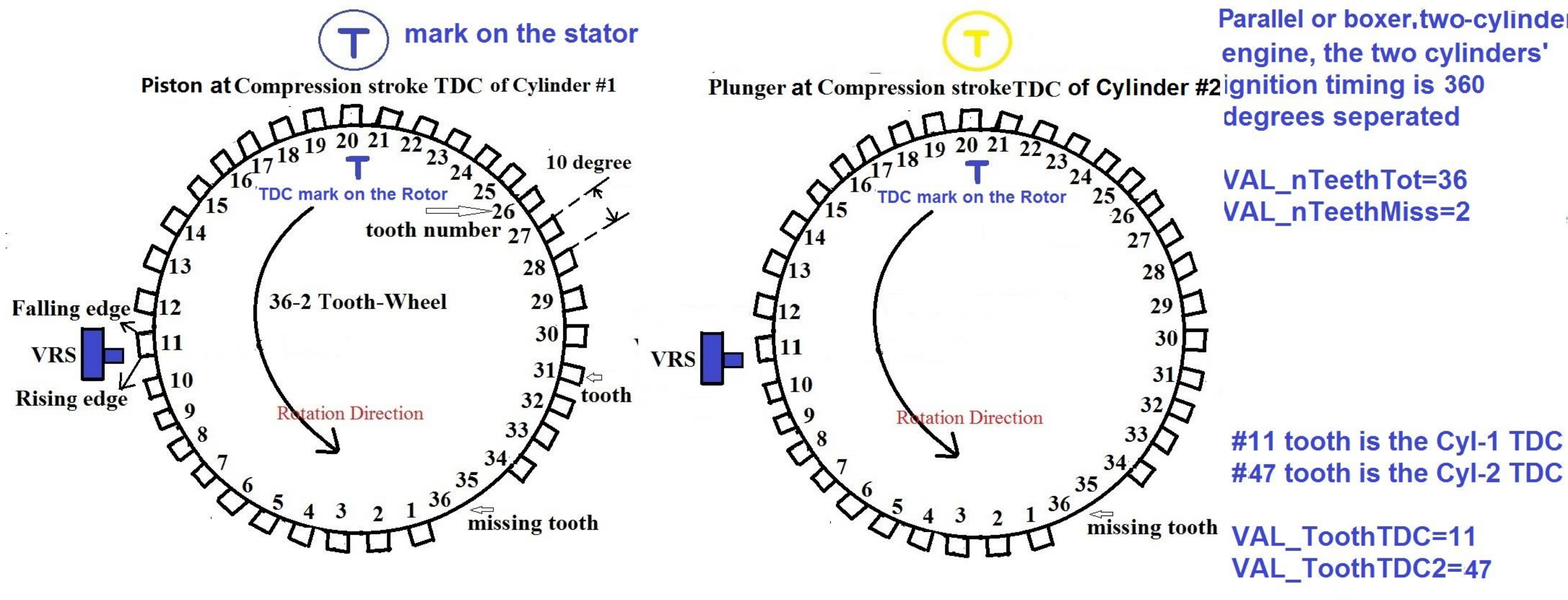
Note: **The value of the tooth number for 4 stroke engines is form 1 to 72, because 2 revolutions are one full engine cycle. The tooth number will increment by 1 into the next revolution, until 72. After number 72, the tooth number reset to 1.**



So VAL_ToothTDC=11, VAL_ToothTDC2=56.

If the two cylinders are separate from 360 degrees, like a boxer or parallel twin engine:

VAL_ToothTDC=11, VAL_ToothTDC2=47



Note: **The method to calibrate 12-1, 24-1, 60-2, etc. tooth-wheels for four-stroke engines is same as to the 36-2 tooth-wheel.**

Two-stroke engines:

1) Single cylinder 2 stroke engines

You need calibrate VAL_ToothTDC

VAL_ToothTDC is the tooth number of ignition TDC.

For example, for a 36-2 tooth-wheel, #11 tooth is the ignition DTC, so VAL_ToothTDC=11

2) Two-cylinder 2 stroke engines

You need calibrate VAL_ToothTDC, VAL_ToothTDC 2

If the two cylinders are separate by 180 degrees,

VAL_ToothTDC=11, VAL_ToothTDC2=29 (each tooth angle is 10 degree)

Note: **The value of the tooth number is form 1 to 36 for 2 stroke engines, because 1 revolution is one full engine cycle. The tooth number will be reset to 1 in the next cycle.**

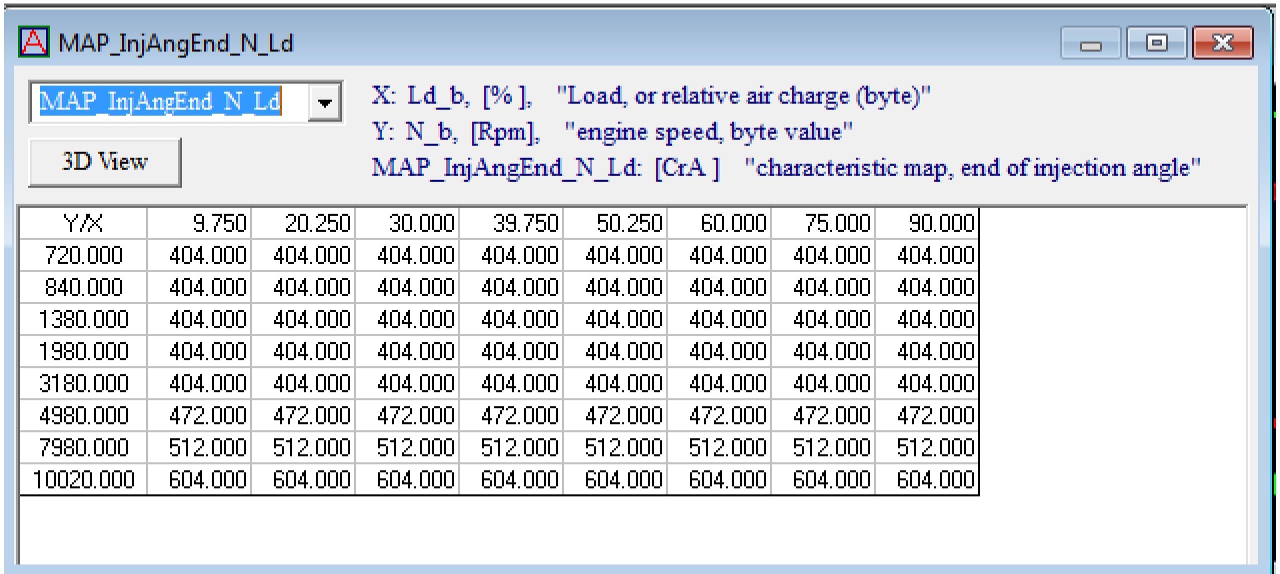
The method to calibrate 12-1, 24-1, 60-2, and etc. tooth wheels is same as to the 36-2 tooth-wheel.

Calibration of ignition timing and injection timing

The ignition angle MAP table: MAP_Iga_N_Ld

Injection end angle: MAP_InjAngEnd_N_Ld

Both of them can be calibrated in the engine dyno or chassis dyno.



Y/X	9.750	20.250	30.000	39.750	50.250	60.000	75.000	90.000
720.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000
840.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000
1380.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000
1980.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000
3180.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000	404.000
4980.000	472.000	472.000	472.000	472.000	472.000	472.000	472.000	472.000
7980.000	512.000	512.000	512.000	512.000	512.000	512.000	512.000	512.000
10020.000	604.000	604.000	604.000	604.000	604.000	604.000	604.000	604.000

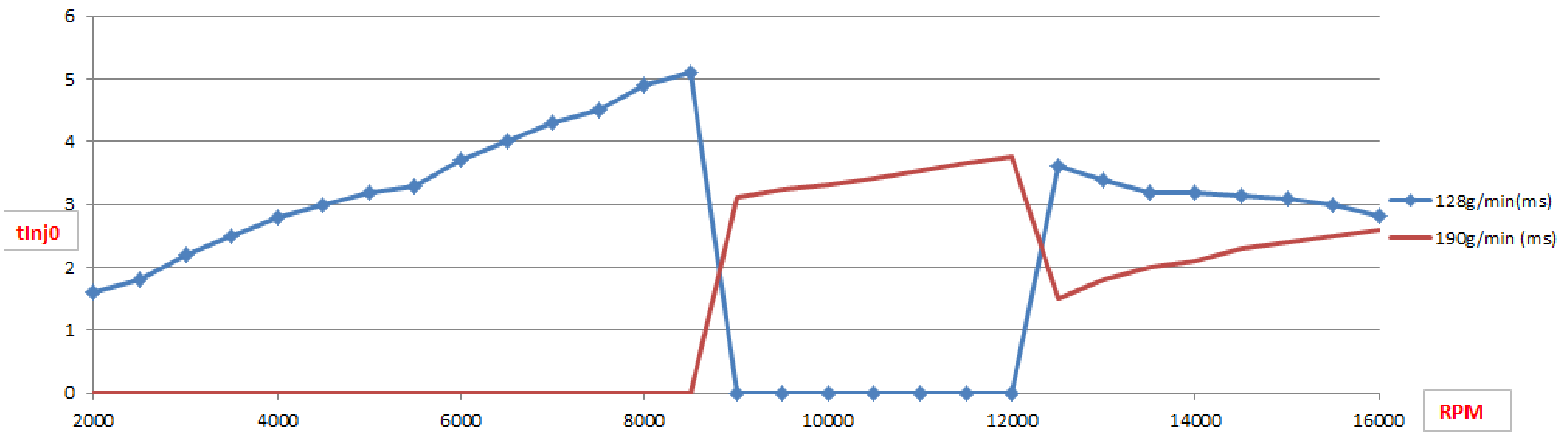
7.5 Dual injectors advanced tuning

Why 2 injectors it need for 2-strokes and high RPM engine?

The whole idea of 2 injectors (one small, one big) is based on the need: 2-stroke engines have only half the time to inject fuel compared to 4-stroke (360 vs 720 degrees), and the 2-strokes usually (esp. high-end engines) have such a wide range of RPM (2000 - 16000rpm, for example). One injector is not able to cover the whole RPM range, simply because every injector has its physical limit: the fixed flow rate. If you use a big injector to cover the high end, then you will have too-rich idle. If you use a small injector, to have a good idle, then you don't have enough fuel for WOT. Given a certain inject flow rate, you can only run an engine either at low RPM range or high RPM range, but not both (from 2000 to 16000rpm).

So, for 2-stroke engines with 10k-16k RPM, usually, we provide EFI system with dual fuel injectors, one smaller and one larger, the smaller injector is used start, idle, low RPM, and the larger one is used high RPM, and in WOT condition, the 2x injectors work together.

The following is the working example, 1 cylinder, 2-stroke engine with 16000 RPM (128g/min and 190g/min).



Control strategy:

There is one important thing for two injectors system you need understand.

For example, if the RPM is 10000, the time of 1 RPM is 6ms. (1 min=60 s=60000ms, 60000ms/10000RPM=6ms/RPM)

If the first injector injected time (tInj0) is bigger than 4ms (6ms- VAL_tInjBrkMin), this means the fuel injection can't match the need of engine, it will change to the second injector to inject fuel. Otherwise, the first injector continues to inject fuel.

If each injector can't match engine need, the two injectors will work together.

Default setting: VAL_tInjBrkMin=2000 us=2ms

You can adjust the fuel injection MAP table "RAM_MAP_LdTp_Tps_N" based on your engine running, the value of this table is bigger, the fuel will be more, otherwise, the fuel will be less. The ECU can't know how many fuel is enough, you need calibrate it. If it is rich, you can reduce the fuel injection.

And you also can calibrate VAL_tInjBrkMin to decide in which condition to change to the #2 injector

(Go to Menu->Advanced->Add Advanced Calibrations, to find the VAL_tInjBrkMin).

7.6 Altitude tuning

Our EFI system can work in different altitude, and it can use the Map (Baro) sensor to estimate the altitude, then to correct the fuel automatically, but it also needs some fine tuning.

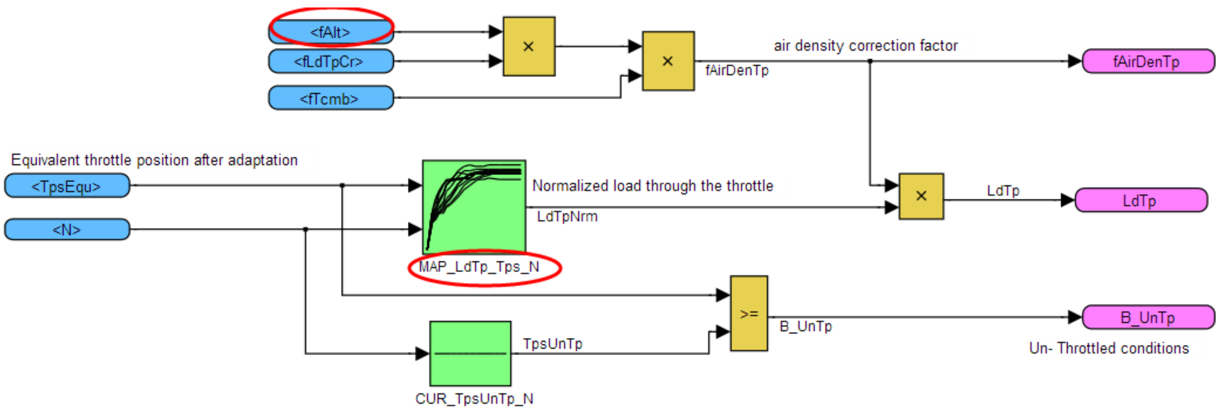
There is an altitude factor to correct the fuel injection, fAlt, it is a universal factor, and it is a multiplication coefficient to correct fuel injection. **fAlt =Pam/1013,**

So, if you are at sea level, Pam is 1013, so the fAlt is 1,And at 5000 feet altitude, the Pam is 850hPa, so the fAlt = 850/1013, it is 0.83.

For example, at sea level, the fuel injection is 5ms @ 2000RPM, so the fuel injection will be 5*0.83 (fAlt) =4.2ms @2000RPM at 5000 feet altitude.

Of course, the intake air pressure Map is also be different, this just explains to the altitude factor how to work.

The fAlt is not used to correct fuel injection time directly, it is used to correct the fVe and Tps based mapping, then to correct the fuel injection.



Chapter 8 Sensors and actuators

8.1 Sensors' characteristics calibration

You only need to calibrate sensors and actuators if you are using the ones other than what we provide. (For example, you installed a throttle body from after-market or from another FI engine, which comes with the OEM's TPS, MAP sensors, and injector); or if you use our ECU only to replace the stock ECU on an EFI engine.

The best way to tune the sensor and actuator's characteristics is to find the manufacturer's data sheets. If you cannot get them, here are the ways to get coarse data for them.

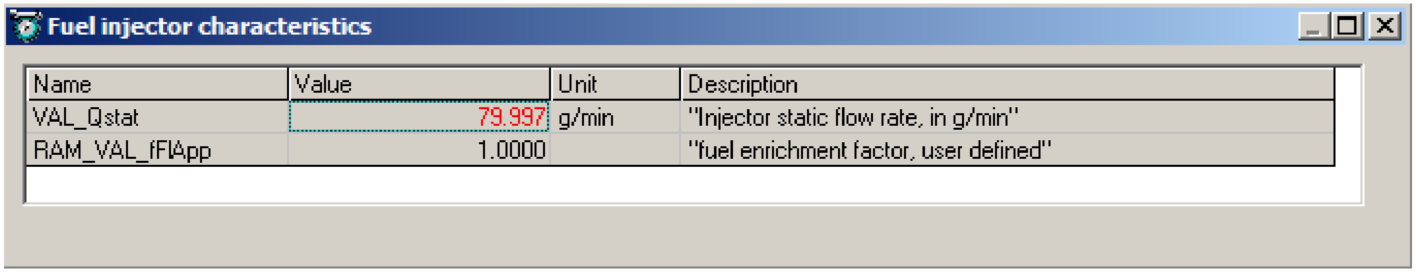
8.1.1 Injector characteristics

Static flow rate:

VAL_Qstat: injector static flow rate (g/min, or gram per minute)

This must be found out from the manufacturer's datasheet. Or you have to hire a professional company to measure it.

Menu->Calibrations->Fuel system->Fuel injector characteristics


Fuel injector characteristics

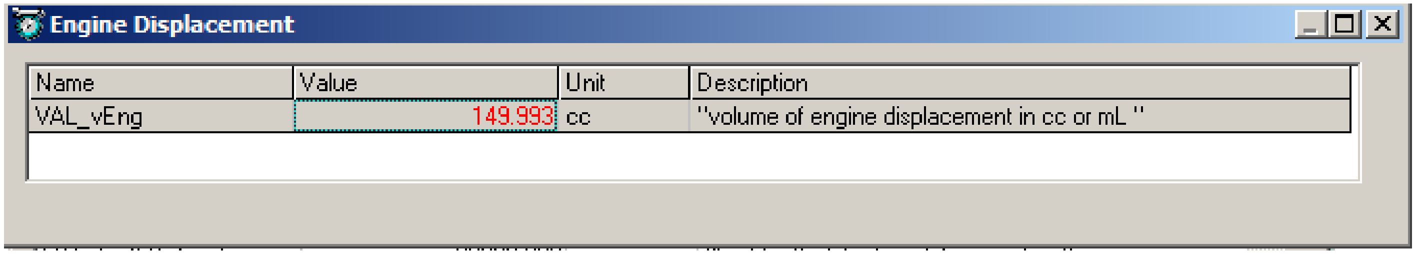
Name	Value	Unit	Description
VAL_Qstat	79.997	g/min	"Injector static flow rate, in g/min"
RAM_VAL_fFlApp	1.0000		"fuel enrichment factor, user defined"

Together, it may be necessary to find out what is the fuel pressure in the fuel rail, or high pressure line. Make sure the "VAL_Qstat" has the value which is associated with this fuel pressure. Otherwise it may need adjusted. Ask us how to.

8.1.2 Engine displacement

VAL_vEng: volume of engine displacement (cc or milliliter)

Menu->Calibrations->System parameters->Engine displacement


Engine Displacement

Name	Value	Unit	Description
VAL_vEng	149.993	cc	"volume of engine displacement in cc or mL "

This might be the easiest one to figure out.

Note: **If it's a 2 cylinder engine, this value needs to be divided by 2. Meaning this is the single-cylinder displacement.**

8.1.3 MAP sensor

Map: manifold absolute pressure (hPa)

uMap: voltage of MAP sensor signal (V)

Map = uMap * VAL_PmapGrd + VAL_PmapOfs

Analog signal from the MAP sensor, uMap, is converted into absolute pressure in hPa. the conversion is linear, with slope of VAL_PmapGrd, and offset of VAL_PmapOfs.

"VAL_PmapGrd" and "VAL_PmapOfs" are calibrate-able variables in the ECU.

To find out these 2 values, you have below options:

Get it from the manufacturer, either from the vehicle manufacturer, or component manufacturer.

Measure it yourself, and you need 2 pairs of (pressure vs voltage) data points:

Use an accurate pressure gauge, measure the pressure point #1, P1, and at the same time use a multi-meter, measure the voltage of the MAP sensor signal wire, u1;

Measure the pressure point #2, P2, and at the same time use a multi-meter, measure the voltage of the MAP sensor signal wire, u2;

These 2 data points are preferred to be representative to the range of MAP sensor.

For example;

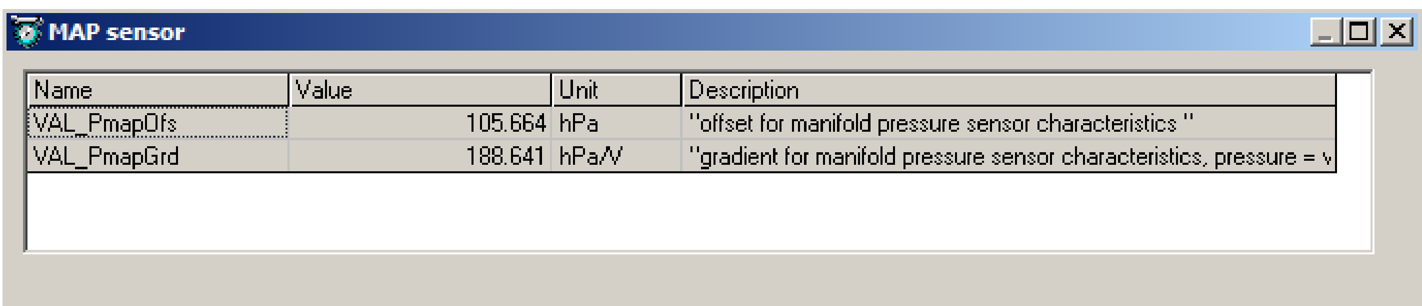
P1 can be the idle air pressure: 500hPa.

Note: **When measuring the pressure, create a stable vacuum. Small engine manifold pressure is too dynamic for this measurement.**

P2 can be the barometric pressure: 1013 hPa at sea level.

Then
VAL_PmapGrd = (P2 - P1) / (u2 - u1)

VAL_PmapOfs = P2 - u2* VAL_PmapGrd


MAP sensor

Name	Value	Unit	Description
VAL_PmapOfs	105.664	hPa	"offset for manifold pressure sensor characteristics "
VAL_PmapGrd	188.641	hPa/V	"gradient for manifold pressure sensor characteristics, pressure = v

Menu->Calibrations->sensor characteristic-> Map sensor

8.1.4 IAT sensor

IAT: intake air temperature sensor

Ta: Tempearature of intake air (°C)

uTa: voltage of air temperature sensor signal (V)

Usually temperature sensors are NTC type: negative temp characteristics:

Assuming, as a first-order approximation, that the relationship between resistance and temperature is linear, then:

ΔR = k * ΔT

Where

ΔR = change in resistance

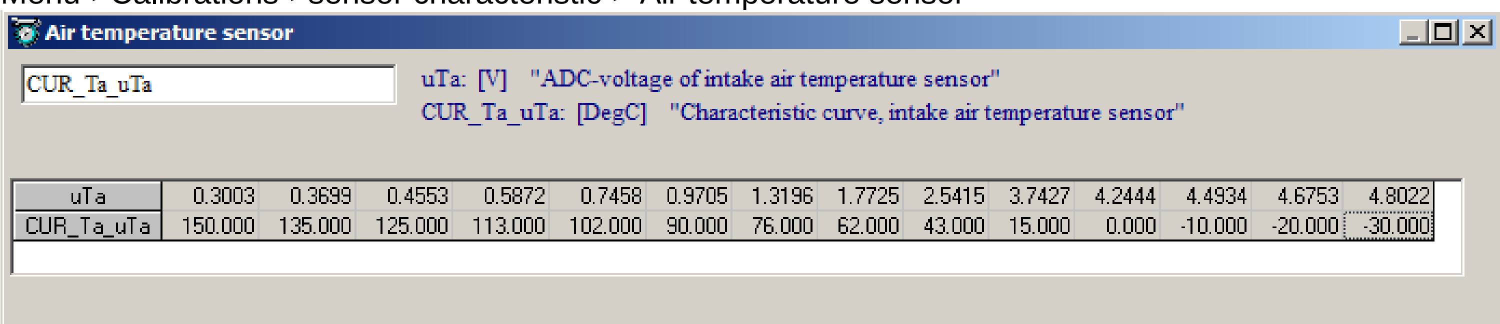
ΔT = change in temperature

k = first-order temperature coefficient of resistance

Contact us if you do not have sensor manufacturer's data.

By these relations, you can calculate the below table and fill in HAECAL:

Menu->Calibrations->sensor characteristic-> Air temperature sensor



uTa	0.3003	0.3699	0.4553	0.5872	0.7458	0.9705	1.3196	1.7725	2.5415	3.7427	4.2444	4.4934	4.6753	4.8022
CUR_Ta_uTa	150.000	135.000	125.000	113.000	102.000	90.000	76.000	62.000	43.000	15.000	0.000	-10.000	-20.000	-30.000

8.1.5 ECT sensor

ECT: engine coolant temperature sensor

Tm: Engine (motor)Tempearature (°C)

uTm: voltage of engine temperature sensor signal (V)

Usually temperature sensors are NTC type: negative temp characteristics:

Assuming, as a first-order approximation, that the relationship between resistance and temperature is linear, then:

$\Delta R = k * \Delta T$

Where

ΔR = change in resistance

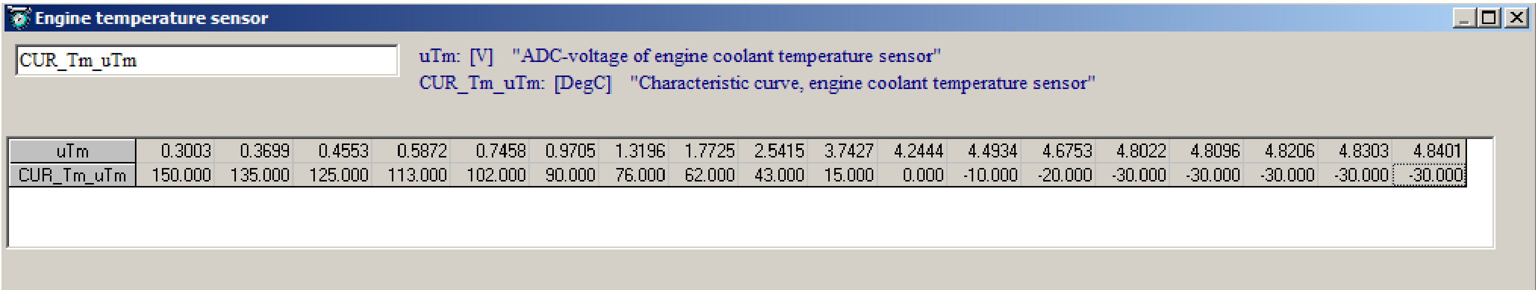
ΔT = change in temperature

k = first-order temperature coefficient of resistance

Contact us if you do not have sensor manufacturer's data.

By these relations, you can calculate the below table and fill in HAECAL:

Menu->Calibrations->sensor characteristic-> engine temperature sensor



uTm	0.3003	0.3699	0.4553	0.5872	0.7458	0.9705	1.3196	1.7725	2.5415	3.7427	4.2444	4.4934	4.6753	4.8022	4.8096	4.8206	4.8303	4.8401
CUR_Tm_uTm	150.000	135.000	125.000	113.000	102.000	90.000	76.000	62.000	43.000	15.000	0.000	-10.000	-20.000	-30.000	-30.000	-30.000	-30.000	-30.000

Calibration for your own temperature sensor (IAT or ECT):

If you have the characteristic from the manufacturer, you can fill the table directly.

If not, you need measure it by yourself.

A rough but working one: put your temperature sensor into some coolant, which you can heat it up at different temp., and use a thermo-stat measure the coolant temp, and also measure the voltage with our software HAECAL, and fill in the CUR_Tm_uTm or CUR_Ta_uTa.

8.1.6 TPS sensor

Tps: throttle position sensor (%)

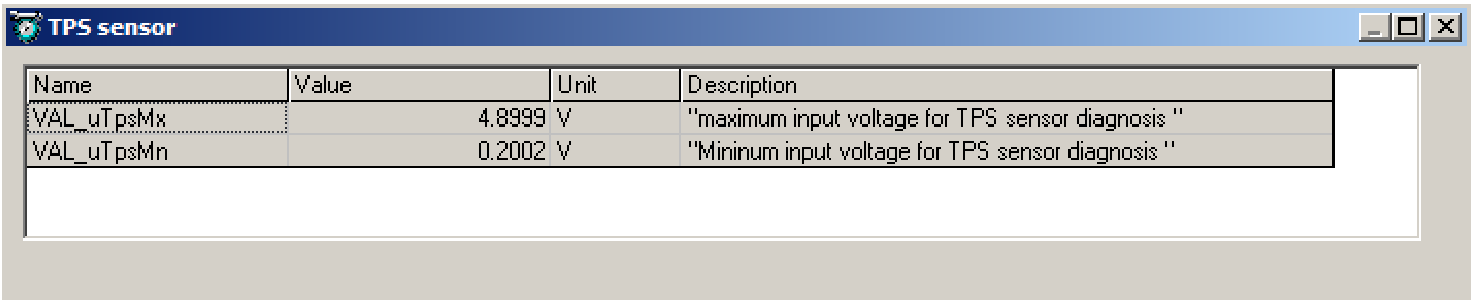
uTps: voltage of TPS sensor signal (V)

For throttle position sensors, you need to measure the idle position "uTps1", and WOT position "uTps2".

Then find the values for VAL_uTpsMx and VAL_uTpsMn, to make sure that:

+5V > VAL_uTpsMx > (uTps2 + 0.1v)

(uTps1 - 0.1v) > VAL_uTpsMn > 0



Name	Value	Unit	Description
VAL_uTpsMx	4.8999	V	"maximum input voltage for TPS sensor diagnosis "
VAL_uTpsMn	0.2002	V	"Mininum input voltage for TPS sensor diagnosis "

TPS self-adaptation

When will the ECU run the TPS self-adaptation?

1. When a new ECU is installed;
2. When either the TPS or the throttle body is replaced:
3. Your ECU cannot reach 100% TPS.
4. When ECU is first on after the TPS is mechanically adjusted.

How to do TPS self-adaptation?

1. Power on, and disconnect the ECU (power-fail) to reset the self-adaptation;
   For example, disconnect the ECU from the connector with key on.
2. Power off, re-connect the ECU, and power on again.
3. Keep the throttle at “idle position” for more than 5s.

If you look closely in HAECAL, you can watch the TPS reading is changing from a small value to 0% by itself.

Now you can read the “TPS” is 0% in HAECAL.

		tInj	1.963	20ms	ms	0	65.535
		Tm	54.000	20ms	DegC	-50	205
		TPS	0.000	20ms	%	0	100
		TpsEqu	0.000	20ms	%	0	99.99847
		UbAdc	15.781	20ms	V	0	25.6
		uLsb	1.1670	20ms	V	-1	4

4. Keep the throttle at “WOT” (wide open throttle) for more than 5S.

Now you can read the “TPS” is almost 100% or 99.99% in HAECAL.

		StepPosDsr	190.000	20ms		0	640
		Ta	19.000	20ms	DegC	-50	205
		TexBfCat_uEGT	280.957	20ms	DegC	-50	1230
		tInj	1.963	20ms	ms	0	65.535
V		Tm	53.000	20ms	DegC	-50	205
V		TPS	99.998	20ms	%	0	100
		TpsEqu	99.998	20ms	%	0	99.99847
		UbAdc	15.794	20ms	V	0	25.6
		uLsb	1.1719	20ms	V	-1	4
		uMan	1.8408	20ms	V	0	5

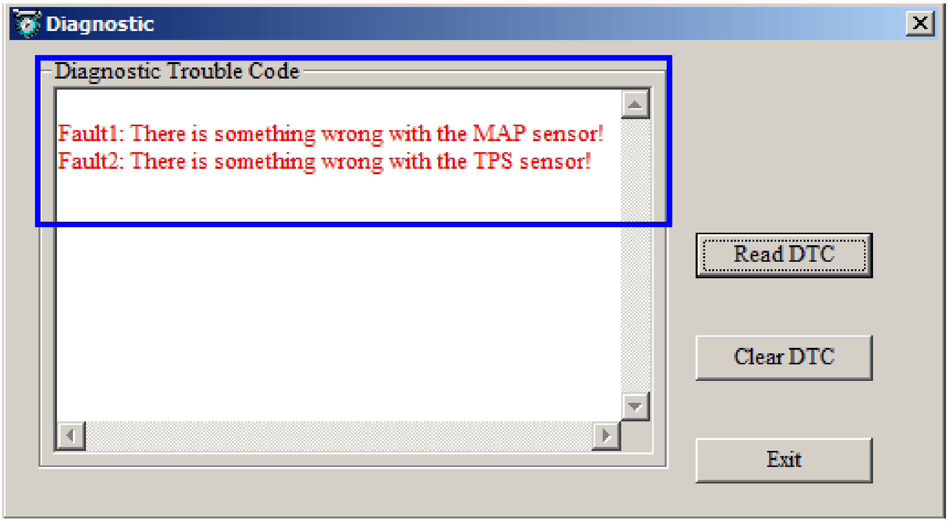
5. Double check TPS in “HAECAL”.

It is 0% when the throttle is at “idle position”, and it is 100% when the throttle is at “WOT” position.

8.2 Fault and Heal

Power on the EFI system, and if the MIL lamp (LED) is on, it means there is an error in the system. Please use HAECAL to read DTC.

For example



Diagnostic Trouble Code:

Fault1: There is something wrong with the MAP sensor!

Map sensor report error, please check the connection.

Fault2: There is something wrong with the TPS sensor!

Throttle body position sensor error, please check the TPS connection, and see the sensor voltage of fully closed and wide opened (4.8V>uTps>0.2V)

Fault3: There is something wrong with the O2 sensor!

Narrow band O2 sensor error, please check the installation and connection, and see the voltage of O2S is range from 0 to 1V

Fault4: There is something wrong with the CKP sensor!

Pick-up signal error, check the connection

Fault5: There is something wrong with the IAT sensor!

Intake air temperature sensor error, check the harness connection

Fault6: There is something wrong with the ECT sensor!

Engine temperature sensor error, check the harness connection

Fault7: There is something wrong with the VPWR!

Battery voltage error, it he voltage of battery is low to 10V and high to 16V, it will report error.

8.2.1 TPS sensor diagnostic

ECU reports TPS error

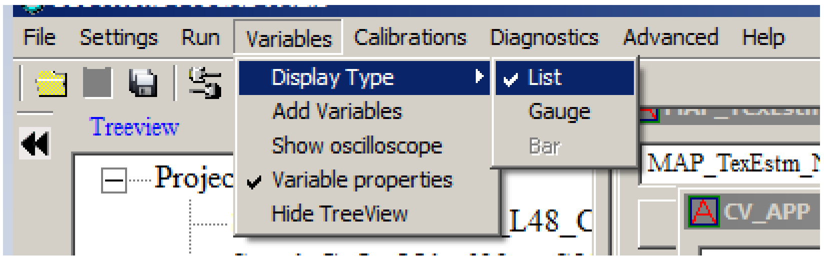
1) Connection problem

The Throttle position sensor is not connected well, please check the wire and reconnect it.

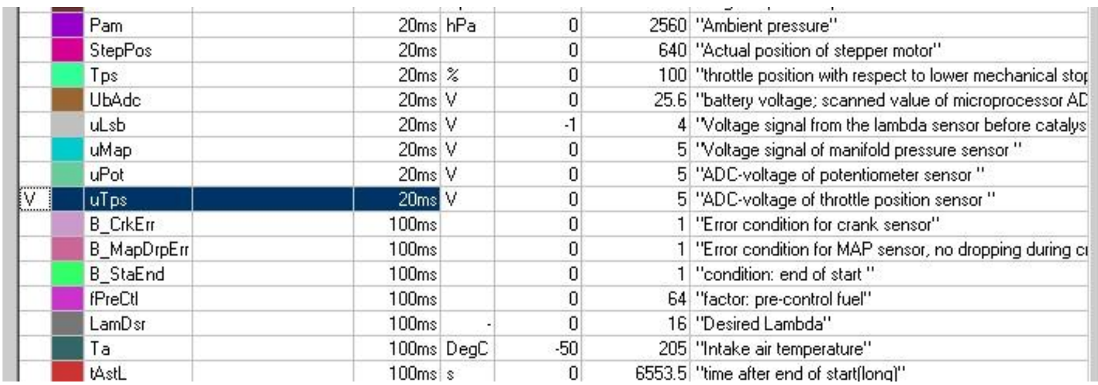
2) The voltage of TPS is out of measuring range

If the signal of TPS is out of range, you need to adjust it by using the following method

(1) Open HAECAL, then go to "Variables Type ->List"



(2) Read the voltage of throttle position Sensor :" uTps"



	Pam	20ms	hPa	0	2560	"Ambient pressure"
	StepPos	20ms		0	640	"Actual position of stepper motor"
	Tps	20ms	%	0	100	"throttle position with respect to lower mechanical stop
	UbAdc	20ms	V	0	25.6	"battery voltage; scanned value of microprocessor AD
	uLsb	20ms	V	-1	4	"Voltage signal from the lambda sensor before catalys
	uMap	20ms	V	0	5	"Voltage signal of manifold pressure sensor "
	uPot	20ms	V	0	5	"ADC-voltage of potentiometer sensor "
V	uTps	20ms	V	0	5	"ADC-voltage of throttle position sensor "
	B_CrkErr	100ms		0	1	"Error condition for crank sensor"
	B_MapDrpErr	100ms		0	1	"Error condition for MAP sensor, no dropping during cr
	B_StaEnd	100ms		0	1	"condition: end of start "
	fPreCtl	100ms		0	64	"factor: pre-control fuel"
	LamDsr	100ms	-	0	16	"Desired Lambda"
	Ta	100ms	DegC	-50	205	"Intake air temperature"
	tAstL	100ms	s	0	6553.5	"time after end of start(long)"

(3) Connect to ECU, to click "Run→Start Measuring"

(4) Remove the Tps sensor form the throttle body, see the below picture.

(5) Install the throttle sensor.

(6) After installed, to mount the screw, it is better not to tight completely so you can adjust it conveniently.

(7) When the throttle fully closed, to see the value of voltage of TPS by HAECAL. (uTps> 0.3V)

	Name	Value	Rate	Unit	Min	Max	Description
	B_CrkErr	0.0000	100ms		0	1	"Error condition for crank sensor"
	tAstL	0.0000	100ms	s	0	6553.5	"time after end of start(long)"
	B_StaEnd	0.0000	100ms		0	1	"condition: end of start "
	uLsb	0.4529	20ms	V	-1	4	"Voltage signal from the lambda sensor before catalys
V	uTps	0.6384	20ms	V	0	5	"ADC-voltage of throttle position sensor "
	fLc	1.0000	20ms	-	0	2	"Lambda controller output (word)"
	LamDsr	1.0000	100ms	-	0	16	"Desired Lambda"
	fPreCtl	1.0000	100ms		0	64	"factor: pre-control fuel"
	uMap	4.5679	20ms	V	0	5	"Voltage signal of manifold pressure sensor "
	uPot	4.9695	20ms	V	0	5	"ADC-voltage of potentiometer sensor "
	Tps	7.7423	20ms	%	0	100	"throttle position with respect to lower mechanical sto

(8) When the throttle is widely open, to see the voltage. (uTps<4.8V)

	Name	Value	Rate	Unit	Min	Max	Description
	tAstL	0.0000	100ms	s	0	6553.5	"time after end of start(long)"
	B_StaEnd	0.0000	100ms		0	1	"condition: end of start "
	uLsb	0.4529	20ms	V	-1	4	"Voltage signal from the lambda sensor before catalys
V	uTps	4.4141	20ms	V	0	5	"ADC-voltage of throttle position sensor "
	fLc	1.0000	20ms	-	0	2	"Lambda controller output (word)"
	LamDsr	1.0000	100ms	-	0	16	"Desired Lambda"
	fPreCtl	1.0000	100ms		0	64	"factor: pre-control fuel"
	uMap	4.5667	20ms	V	0	5	"Voltage signal of manifold pressure sensor "
	uPot	4.9695	20ms	V	0	5	"ADC-voltage of potentiometer sensor "

(9) If the voltage of throttle sensor is not from 0.3V to 4.8V, you can rotate the sensor to adjust the voltage.

Note: **The voltage of throttle sensor is normal in the range from 0.3 to 4.8V**

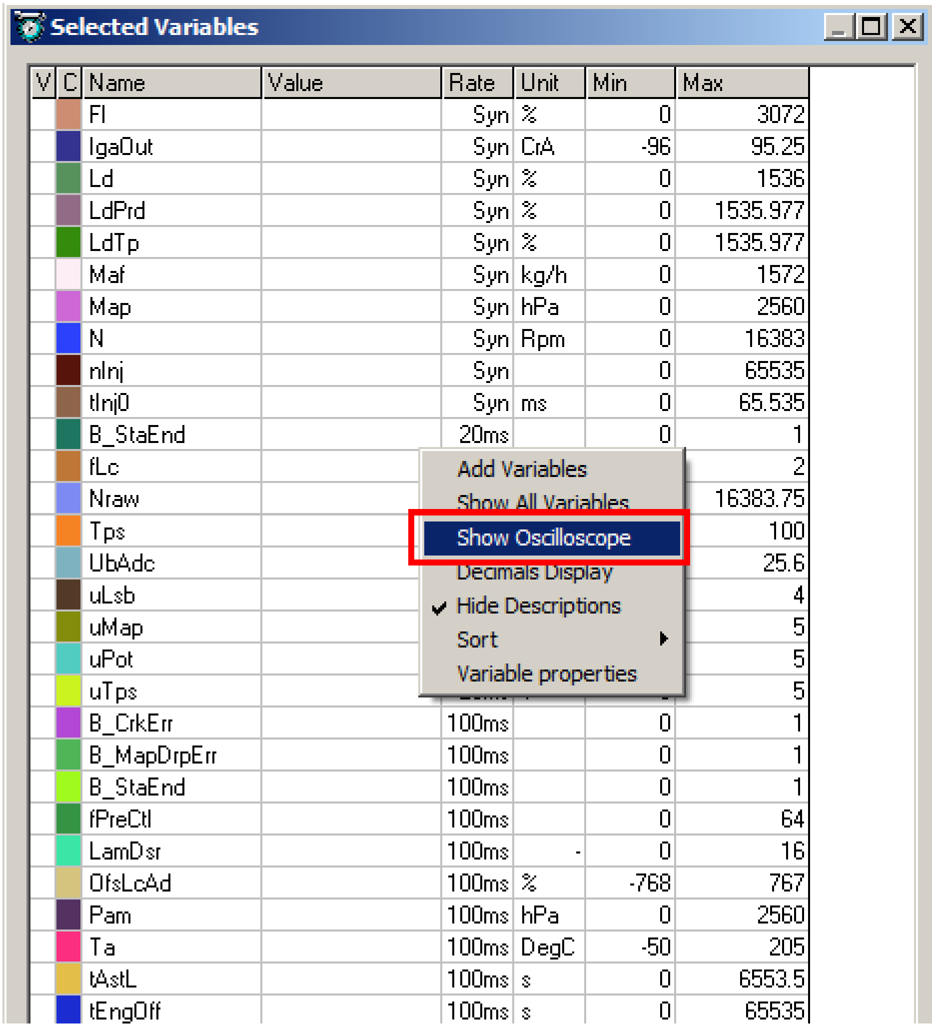
3) Reset the TPS position if Tps is not 0 when idle

Key on the system and key on only, disconnect the ECU from the main harness and reconnect it with keeping power on. Then you can see the Tps is changed to 0 when throttle is fully closed.

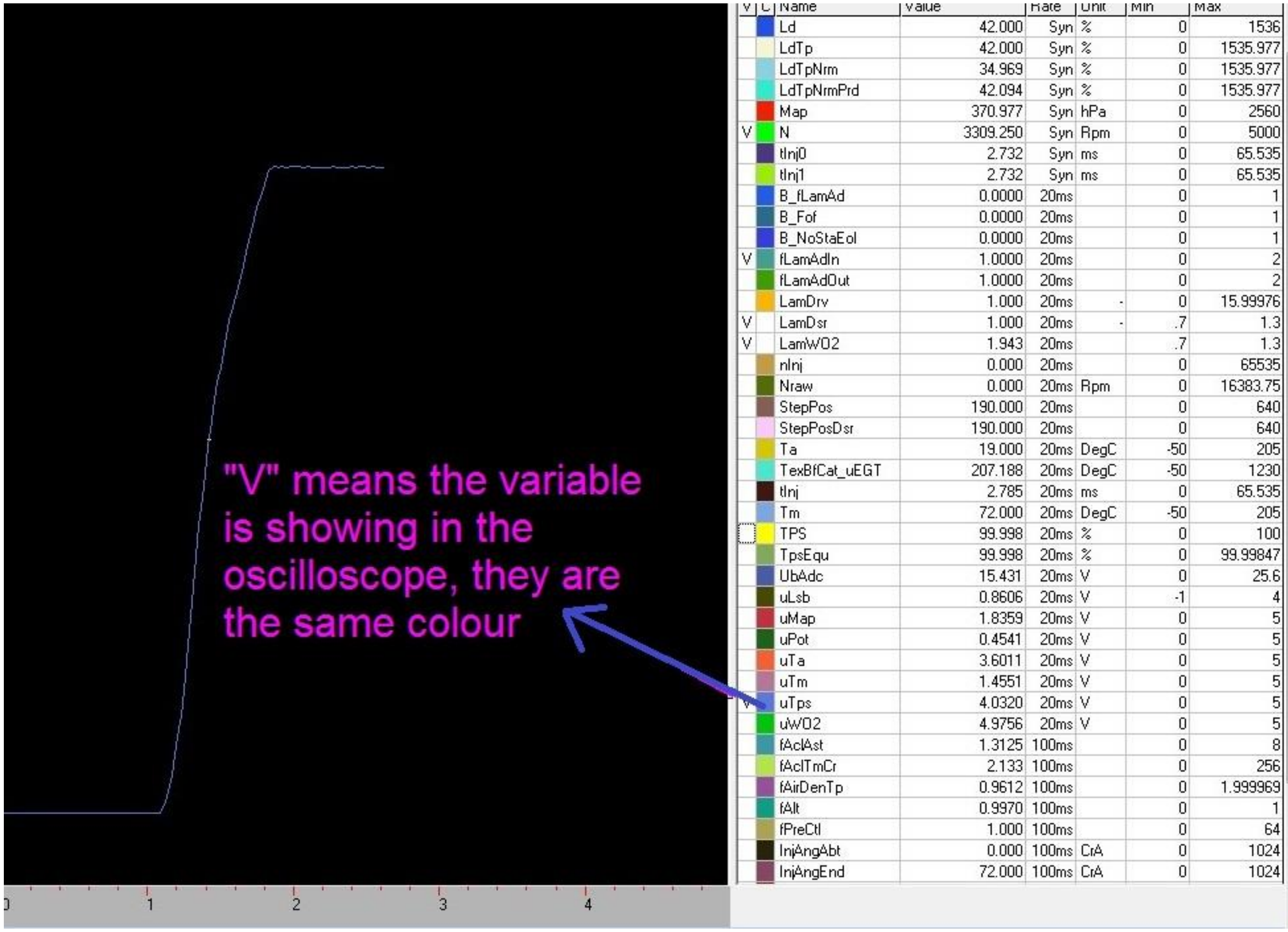
4) How to check the TPS signal rationality?

With the TPS is installed, the output voltage should be 0.3~4.8V DC. You can tell whether the TPS is broken by reading the virtual scope of the HAECAL

1) Show the “uTps” in the oscilloscope (right click on the list window).



V	C	Name	Value	Rate	Unit	Min	Max
		Fl		Syn	%	0	3072
		IgaOut		Syn	CrA	-96	95.25
		Ld		Syn	%	0	1536
		LdPrd		Syn	%	0	1535.977
		LdTp		Syn	%	0	1535.977
		Maf		Syn	kg/h	0	1572
		Map		Syn	hPa	0	2560
		N		Syn	Rpm	0	16383
		nInj		Syn		0	65535
		tInj0		Syn	ms	0	65.535
		B_StaEnd		20ms		0	1
		fLc					2
		Nraw					16383.75
		Tps					100
		UbAdc					25.6
		uLsb					4
		uMap					5
		uPot					5
		uTps					5
		B_CrkErr		100ms		0	1
		B_MapDrpErr		100ms		0	1
		B_StaEnd		100ms		0	1
		fPreCtl		100ms		0	64
		LamDsr		100ms	-	0	16
		OfsLcAd		100ms	%	-768	767
		Pam		100ms	hPa	0	2560
		Ta		100ms	DegC	-50	205
		tAstL		100ms	s	0	6553.5
		tEngOff		100ms	s	0	65535

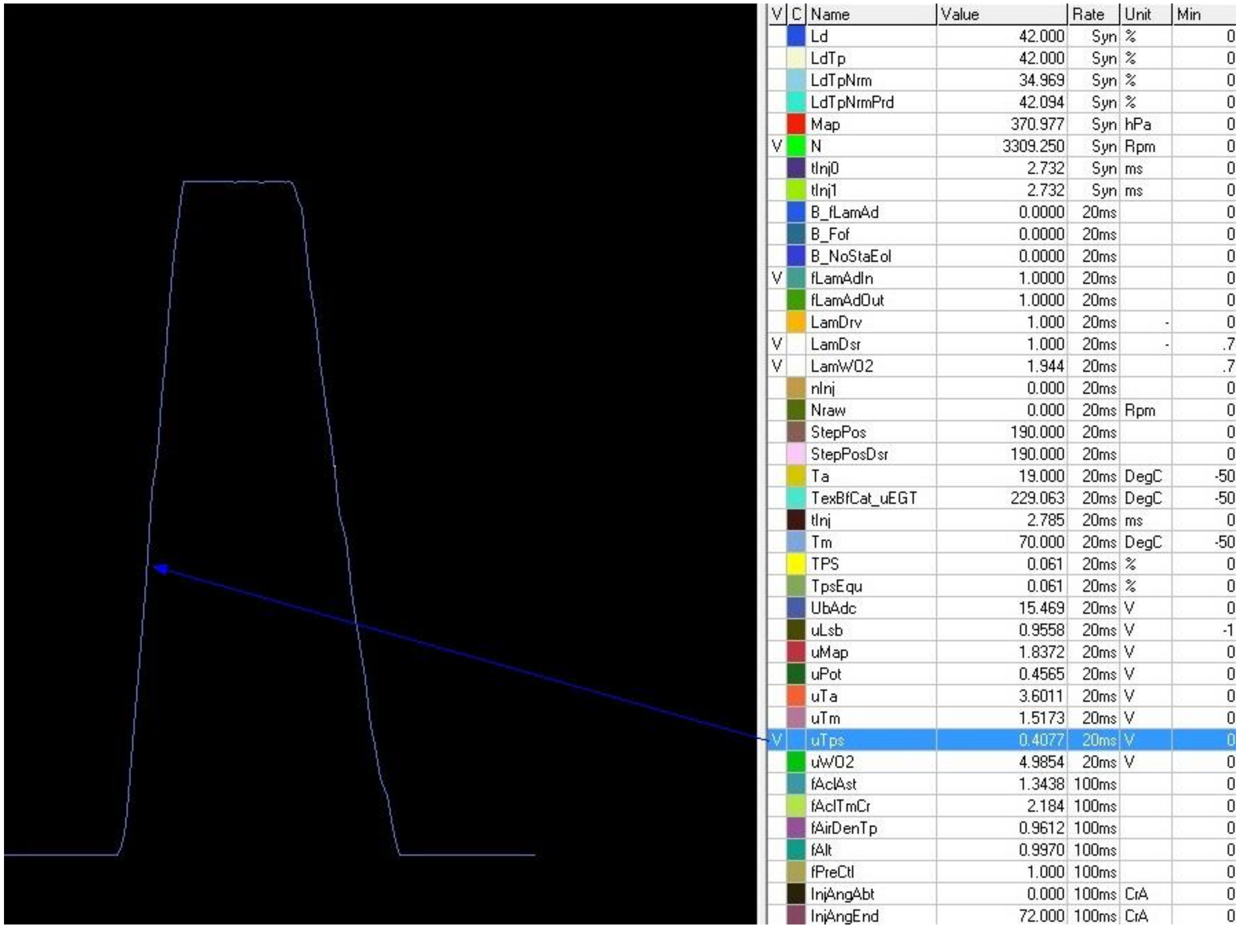


V	C	Name	Value	Rate	Unit	Min	Max
		Ld	42.000	Syn	%	0	1536
		LdTp	42.000	Syn	%	0	1535.977
		LdTpNrm	34.969	Syn	%	0	1535.977
		LdTpNrmPrd	42.094	Syn	%	0	1535.977
		Map	370.977	Syn	hPa	0	2560
V		N	3309.250	Syn	Rpm	0	5000
		tInj0	2.732	Syn	ms	0	65.535
		tInj1	2.732	Syn	ms	0	65.535
		B_fLamAd	0.0000	20ms		0	1
		B_Fof	0.0000	20ms		0	1
		B_NoStaEol	0.0000	20ms		0	1
V		fLamAdIn	1.0000	20ms		0	2
		fLamAdOut	1.0000	20ms		0	2
		LamDrv	1.000	20ms	-	0	15.99976
V		LamDsr	1.000	20ms	-	.7	1.3
V		LamWO2	1.943	20ms		.7	1.3
		nInj	0.000	20ms		0	65535
		Nraw	0.000	20ms	Rpm	0	16383.75
		StepPos	190.000	20ms		0	640
		StepPosDsr	190.000	20ms		0	640
		Ta	19.000	20ms	DegC	-50	205
		TexBfCat_uEGT	207.188	20ms	DegC	-50	1230
		tInj	2.785	20ms	ms	0	65.535
		Tm	72.000	20ms	DegC	-50	205
		TPS	99.998	20ms	%	0	100
		TpsEqu	99.998	20ms	%	0	99.99847
		UbAdc	15.431	20ms	V	0	25.6
		uLsb	0.8606	20ms	V	-1	4
		uMap	1.8359	20ms	V	0	5
		uPot	0.4541	20ms	V	0	5
		uTa	3.6011	20ms	V	0	5
		uTm	1.4551	20ms	V	0	5
V		uTps	4.0320	20ms	V	0	5
		uWO2	4.9756	20ms	V	0	5
		fAclAst	1.3125	100ms		0	8
		fAclTmCr	2.133	100ms		0	256
		fAirDenTp	0.9612	100ms		0	1.999969
		fAlt	0.9970	100ms		0	1
		fPreCtl	1.000	100ms		0	64
		InjAngAbt	0.000	100ms	CrA	0	1024
		InjAngEnd	72.000	100ms	CrA	0	1024

2) Connect to ECU, to click "Run—Start Measuring"

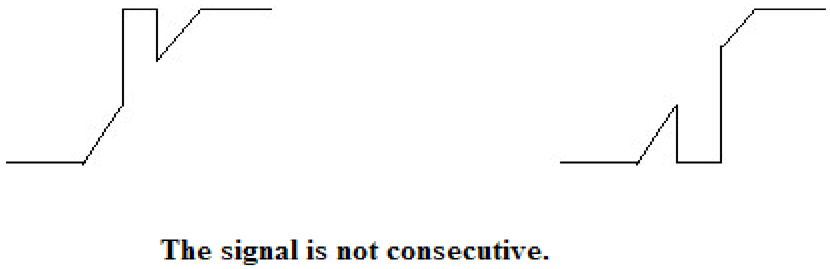
3) Slowly open and close the throttle, and you can get a symmetrical curve on the virtual oscilloscope, which is 0.3V~4.8V, like picture 1.

If the line is not consecutive, like picture 2, you should replace it with a new TPS sensor.



V	C	Name	Value	Rate	Unit	Min
		Ld	42.000	Syn	%	0
		LdTp	42.000	Syn	%	0
		LdTpNrm	34.969	Syn	%	0
		LdTpNrmPrd	42.094	Syn	%	0
		Map	370.977	Syn	hPa	0
V		N	3309.250	Syn	Rpm	0
		tInj0	2.732	Syn	ms	0
		tInj1	2.732	Syn	ms	0
		B_fLamAd	0.0000	20ms		0
		B_Fof	0.0000	20ms		0
		B_NoStaEol	0.0000	20ms		0
V		fLamAdIn	1.0000	20ms		0
		fLamAdOut	1.0000	20ms		0
		LamDrv	1.000	20ms	-	0
V		LamDsr	1.000	20ms	-	.7
V		LamWO2	1.944	20ms		.7
		nInj	0.000	20ms		0
		Nraw	0.000	20ms	Rpm	0
		StepPos	190.000	20ms		0
		StepPosDsr	190.000	20ms		0
		Ta	19.000	20ms	DegC	-50
		TexBfCat_uEGT	229.063	20ms	DegC	-50
		tInj	2.785	20ms	ms	0
		Tm	70.000	20ms	DegC	-50
		TPS	0.061	20ms	%	0
		TpsEqu	0.061	20ms	%	0
		UbAdc	15.469	20ms	V	0
		uLsb	0.9558	20ms	V	-1
		uMap	1.8372	20ms	V	0
		uPot	0.4565	20ms	V	0
		uTa	3.6011	20ms	V	0
		uTm	1.5173	20ms	V	0
V		uTps	0.4077	20ms	V	0
		uWO2	4.9854	20ms	V	0
		fAclAst	1.3438	100ms		0
		fAclTmCr	2.184	100ms		0
		fAirDenTp	0.9612	100ms		0
		fAlt	0.9970	100ms		0
		fPreCtl	1.000	100ms		0
		InjAngAbt	0.000	100ms	CrA	0
		InjAngEnd	72.000	100ms	CrA	0

Picture 1



Picture 2

8.2.2 MAP sensor diagnostic

If you have installed the EFI system and key on the bike, engine is not starting, the ECU reports Map error, and there are two possible causes.

1) Connection error

The Map sensor is not connected intact. Please check it and reconnect it to ECU harness.

2) A bad sensor

The Map sensor is broken. Read the signal in HAECAL to check it.

Power on ECU and connect to ECU. Go to menu "Variables->Display Type->List". Then "Run->Start measuring"

uTps	0.2380	20ms	V	0	5	"ADC-voltage of throttle position sensor "
uLsb	0.4529	20ms	V	-1	4	"Voltage signal from the lambda sensor be
fLc	1.0000	20ms	-	0	2	"Lambda controller output (word)"
fPreCtl	1.00	100ms		0	64	"factor: pre-control fuel"
LamDsr	1.00	100ms	-	0	16	"Desired Lambda"
DfsLcAd	1.05	100ms	%	-768	767	"additive adaptive correction of the relativ
uMap	4.6448	20ms	V	0	5	"Voltage signal of manifold pressure senso
uPot	4.9046	20ms	V	0	5	"ADC-voltage of potentiometer sensor "
UbAdc	12.18	20ms	V	0	25.6	"battery voltage; scanned value of microp
Ta	27.00	100ms	DegC	-50	205	"Intake air temperature"
Tm	71.00	100ms	DegC	-50	205	"Engine temperature"
Map	1029.77	20ms	hPa	0	2560	"Intake manifold pressure measured with M
Pam	1030.20	100ms	hPa	0	2560	"Ambient pressure"
tEngOff	65535.00	100ms	s	0	65535	"stop time"

If the voltage of Map sensor (uMap) is 0V or 4.99V, it shows that the Map sensor is broken, but makes sure the Map sensor is connected well at first.

If the Map sensor is broken, contact us to get a good one. The normal voltage (uMap) is about 4.7V (1 Bar sensor); and about 1.8V (2.5Bar sensor) when system is just powered on and engine doesn't start.

3) Fault indication when cranking engine

If ECU reports MAP error (B_MapDrpErr=1) when the engine starts or is running. The possible reason is the intake air vacuum is too less or there is some leaking air.

uPot		20ms	V	0	5	"ADC-voltage of potentiometer
uTps		20ms	V	0	5	"ADC-voltage of throttle positio
B_CrkErr		100ms		0	1	"Error condition for crank sensc
B_MapDrpErr		100ms		0	1	"Error condition for MAP senso
B_StaEnd		100ms		0	1	"condition: end of start "
fPreCtl		100ms		0	64	"factor: pre-control fuel"
LamDsr		100ms	-	0	16	"Desired Lambda"

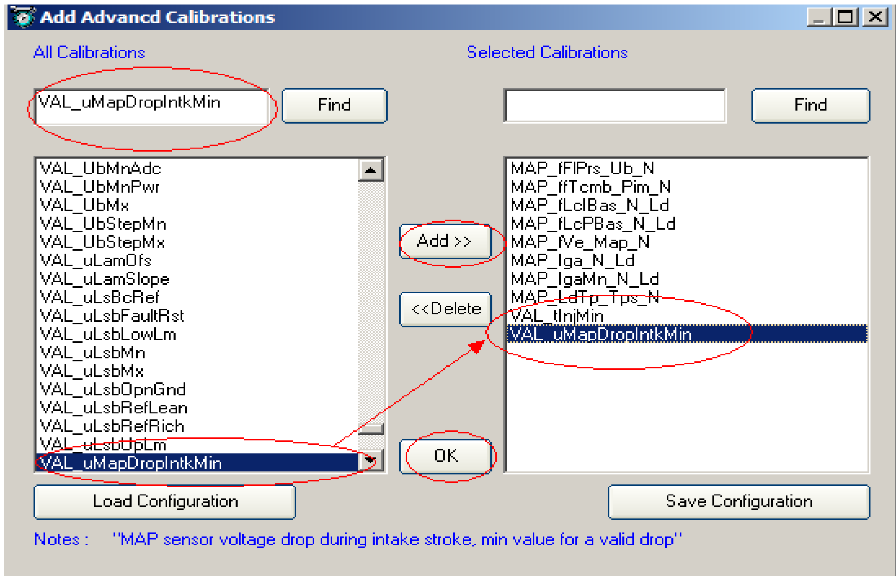
(3) First, check the Map sensor location, and connect the Map sensor to throttle body by a using a short hose.

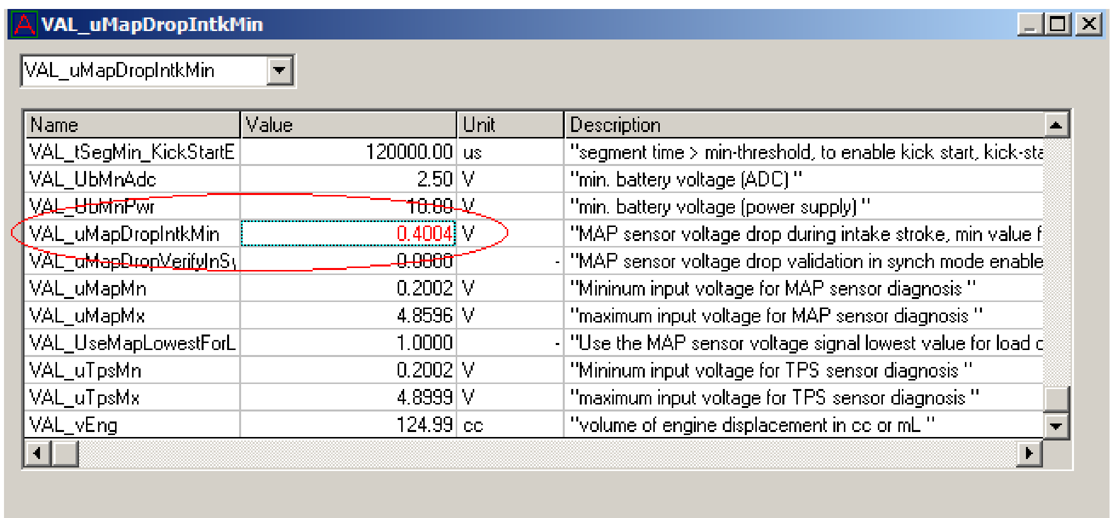
Check it for air leaks and make sure that it is sealed properly.

Turn the idle air screw or stepper motor position to get more idle air.

Change the value of minimum Map voltage drop "VAL_uMapDropIntkMin" during cranking.

In HAECAL, go to "menu->Advanced->Add Advance Calibrations"


Add Advancd Calibrations
All Calibrations
Selected Calibrations
VAL_uMapDropIntkMin
Find
Find
VAL_UbMnAdc
VAL_UbMnPwr
VAL_UbMx
VAL_UbStepMn
VAL_UbStepMx
VAL_uLamOfs
VAL_uLamSlope
VAL_uLsBcRef
VAL_uLsbFaultRst
VAL_uLsbLowLm
VAL_uLsbMn
VAL_uLsbMx
VAL_uLsbOpnGnd
VAL_uLsbRefLean
VAL_uLsbRefRich
VAL_uLsbUpLm
VAL_uMapDropIntkMin
Add >>
<<Delete
OK
MAP_fFlPrs_Ub_N
MAP_ffTcmb_Pim_N
MAP_fLclBas_N_Ld
MAP_fLcPBas_N_Ld
MAP_fVe_Map_N
MAP_Iga_N_Ld
MAP_IgaMn_N_Ld
MAP_LdTp_Tps_N
VAL_tInjMin
VAL_uMapDropIntkMin
Load Configuration
Save Configuration
Notes : "MAP sensor voltage drop during intake stroke, min value for a valid drop"


VAL_uMapDropIntkMin
VAL_uMapDropIntkMin
Name | Value | Unit | Description
VAL_tSegMin_KickStartE | 120000.00 | us | "segment time > min-threshold, to enable kick start, kick-sta
VAL_UbMnAdc | 2.50 | V | "min. battery voltage (ADC) "
VAL_UbMnPwr | 10.00 | V | "min. battery voltage (power supply) "
VAL_uMapDropIntkMin | 0.4004 | V | "MAP sensor voltage drop during intake stroke, min value f
VAL_uMapDropVerifyInSy | 0.0000 | - | "MAP sensor voltage drop validation in synch mode enable
VAL_uMapMn | 0.2002 | V | "Mininum input voltage for MAP sensor diagnosis "
VAL_uMapMx | 4.8596 | V | "maximum input voltage for MAP sensor diagnosis "
VAL_UseMapLowestForL | 1.0000 | - | "Use the MAP sensor voltage signal lowest value for load c
VAL_uTpsMn | 0.2002 | V | "Mininum input voltage for TPS sensor diagnosis "
VAL_uTpsMx | 4.8999 | V | "maximum input voltage for TPS sensor diagnosis "
VAL_vEng | 124.99 | cc | "volume of engine displacement in cc or mL "

Note: **The value of "VAL_uMapDropIntkMin"cannot be smaller than 0.3V**

8.2.3 Temperature sensor (IAT & ECT) diagnostic

IAT: Intake air temperature sensor

ECT: Engine temperature sensor

If ECU reports IAT or ECT error, there are two possible reasons for this

Possible reason #1: the connect wire break off

Possible reason #2: the temperature sensor is broken

Note: **If there is an error about the temperature sensor, please repair it or replace it as soon as possible, because it will affect the fuel injection, it may be too rich or too lean, and the engine can't run normally.**

8.2.4 NB O2 sensor diagnostic

If you have installed an O2 sensor and the O2 sensor signal is not changed, it still 450mv or bigger than 1V when engine is finished warmed up (engine is running about 3 minutes) and anything is connected well. It shows the O2 sensor is broken, please change another good O2 sensor.

And if the system reports O2S error, and you also need to check the connection whether it is short or broken circuit.

8.2.5 VRS signal diagnostic

No matter which pick-up sensor you used, crankshaft sensor or hall sensor, you must make sure there is a clean signal to input ECU, otherwise, the EFI system will not work. The installation and connection are very important. Check previous chapter on how to install the VRS sensor.

8.2.6 Battery diagnostic

If the system reports "VPWR" error, please check the health of battery or the 12V charging system integrity.

8.2.7 Actuator Diagnosis

EFI system has some Actuators, such as Relay, fuel pump, injector, CDI.

If one of actuators doesn't work, the EFI system will not run, so make sure all of these in good shape.

Chapter 9 Advance functions of EFI

9.1 Fuel Dial Function

Fuel Dial means input an analog voltage signal to ECU, and use this signal to adjust the desired lambda (LamDsr) to modify the real-time lambda (AFR).

We use the "Perf-SW", or performance switch, wire to connect the external signal input, such as a potentiometer.

LamDsr = 1/ (uPot * VAL_uPot2Fct+VAL_fLamDrvBas);

VAL_ LamLeanLim and VAL_ LamRichLim are the limitation of LamDsr Max and Min value.

VAL_LamLeanLim < LamDsr < VAL_LamRichLim;

lamDsr: desired lambda which you want

uPot: The voltage signal of potentiometer input to ECU

VAL_fLamDrvBas: Basic factor to calculate the lamDsr

VAL_uPot2Fct: Change the input voltage signal to actual factor.

VAL_LamLeanLIm: lambda lean condition limited

VAL_LamRichLim: lambda rich condition limited

Default settings in CAL:

VAL_fLamDrvBas=0.5;

VAL_uPot2Fct=0.2;

VAL_LamLeanLim=2;

VAL_LamRichLim=0.5;

So, if uPot=0V, then LamDsr=2; and If uPot=5V, LamDsr=0.67.

You also can change the limitation value to limit the Max and Min value of lamDsr.

Example:

If you want the value of LamDsr is 1, then

uPot=(1/lamDsr- VAL_fLamDrvBas)/ VAL_uPot2Fct; so uPot=2.5V.

So if you have not changed the default settings, the output voltage of your potentiometer should be 2.5V.

If you want to adjust the AFR (Lambda), you can pre-adjust the uPot to 2.5V (LamDsr=1) first, then you can change the potentiometer output to be bigger or smaller to get a rich or lean desired lambda, basing on the actual lambda.

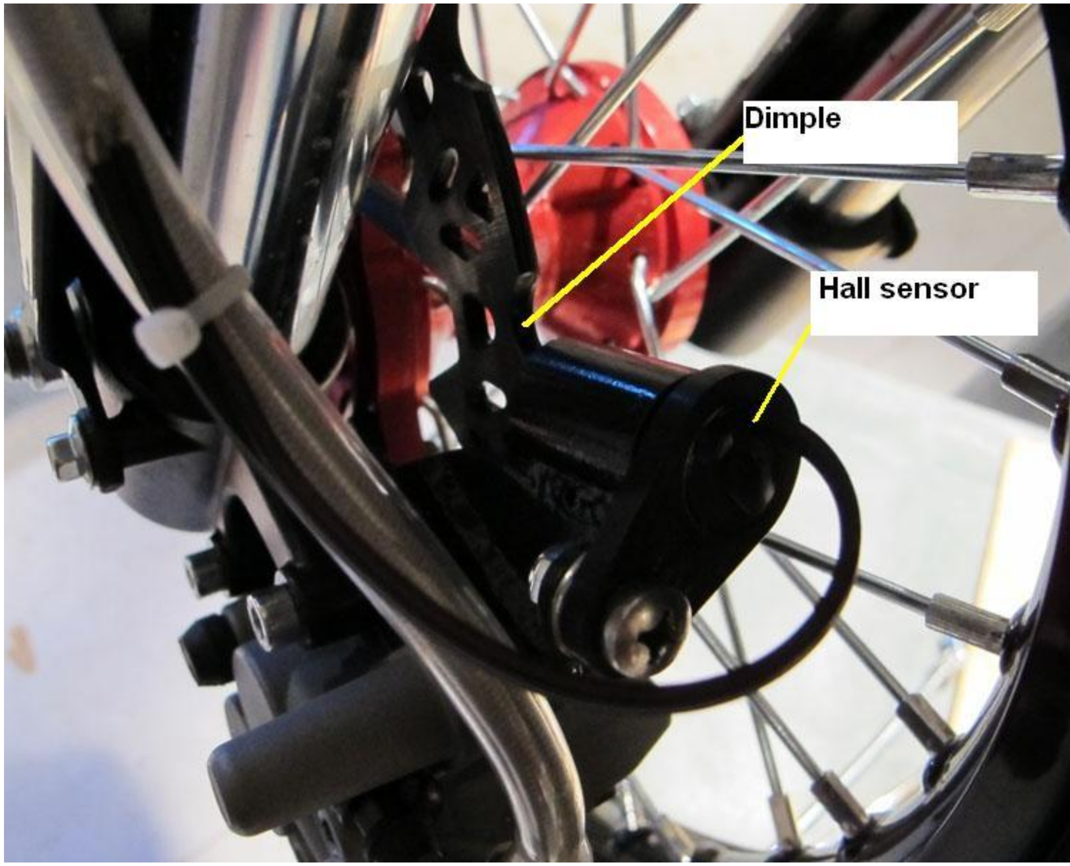
9.2 Traction Control function

Traction Control function with HAE EFI system

For traction controls of 2 wheelers or 3 wheelers, there must be 2 wheel speed sensors to detect the wheel speeds of non-driven wheel and driven wheel.

The ECU will compare the driven wheel speed to non-driven wheel speed. If driven wheel speed is higher than the non-driven wheel, which means there is a slip event. ECU will reduce the engine torque by retarding the ignition angle and / or cut the fuel to eliminate the slipping event.

For ECU to detect the wheel speeds, 2 wheel speed sensors are used, (Hall effect sensor, for example). User must install 2 trigger wheels on the 2 wheels. Or there may be already some "trigger wheels", like in the picture below that the bike comes with 2 wheels with dimples built-in from the OEM. You can use those dimples as the tooth patterns. In this case, all you need is to install 2 Hall Effect sensors.



There are some variables need to be calibrated if you use the traction control function and use two wheel speed sensors to measure the vehicle speed. So you need do some advanced calibration.

Find the variables in HAECAL: go to menu->Advanced->Add Advanced Calibrations. You can calibrate the variables based on your bike spec...

VAL_DiamWhlNonDrv: Diameter of front Wheel (non-driven wheel), the tire size in meters.

VAL_DiamWhlDrv: Diameter of rear Wheel (driven wheel), the tire size in meters.

VAL_nTeethWhl: the total number of teeth on the trigger wheel of wheel speed sensor.

VAL_dWhlSpdAsr - Delta wheel speeds to detect the slip event!

Measure-able variables:

Go to menu->Variables->Add Variables. Add the below variables in 20ms, so you can read them in HAECAL.

tToothDrv: "tooth period of driven wheel speed sensor"

tToothNonDrv: "tooth period of non-driven wheel speed sensor"

VspWhlDrv: "Vehicle Speed from the driven wheel speed sensor (Rear wheel)"

VspWhlNonDrv; "Vehicle Speed from the non-driven wheel speed sensor (Front wheel)"

Note:

The method to calculate vehicle speed as below:

V------- Vehicle speed in km/h.

D-------Diameter of wheel, form the tire size; converted into meter (Pi*D) is the circumference.

1km/h=1m/3600ms

V= (D*3.1416*3600)/ (VAL_nTeethWhl*tooth period)

If (VspWhlDrv- VspWhlNonDrv)> VAL_dWhlSpdAsr, then the traction control function is enabled.

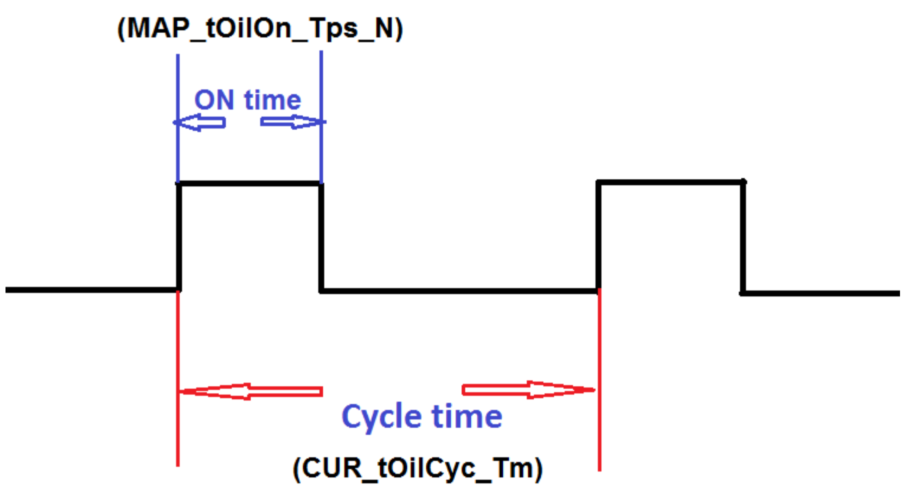
9.3 Oil pump control tuning

For some 2 stroke engines, maybe there needs extra oil pump for lubrication or mixture with fuel, our EFI can use PWM signal to control the oil pump. This is also suitable for 4 stroke engines.

There are two control strategies to control the oil pump to work

1) When Key on, the oil pump will work some seconds, just like fuel pump control. The working time can be tuned, VAL_tOilPrimMx (s).
   You can set VAL_tOilPrimMx to be 5, the oil pump will work 5 seconds after Key on.
2) When you crank to start, there is another MAP table to control the oil pump, MAP_tOilOn_Tps_N, this table is based on engine temperature (Tm) and engine speed (RPM), and it is ON time. VAL_tOilOnMx and VAL_tOilOnMn are the Max and Min value of on time limitation.

And there is also a cycle table, CUR_tOilCyc_Tm, based on engine temperature (Tm).



Chapter 10 Auto-tuning with ALM

How to connect ALM to ECU and enable the auto-tuning feature?

Auto-Tuning means ALM and ECU will work together and automatically tune the AFR as you desire. In ECU, AFR is represented by Lambda (equivalent AFR). Lambda = 1 means AFR 14.7 for gasoline.

The default target Lambda is 1.0 across the board. You can define your own desired Lambda dependent on the RPM and TPS. Usually, a little rich AFR at high RPM / high TPS is preferred to have a better performance as well as engine cooling effect. A typical desired Lambda table could be 0.85 at high RPM and high TPS (>90%); and 1.0 everywhere else. The desired Lambda table should be engine specific. Some engines don't like 14.7 AFR at idle, and can only be stable if it is a little rich. In that case, you should define the desired Lambda to match the engine characteristics.

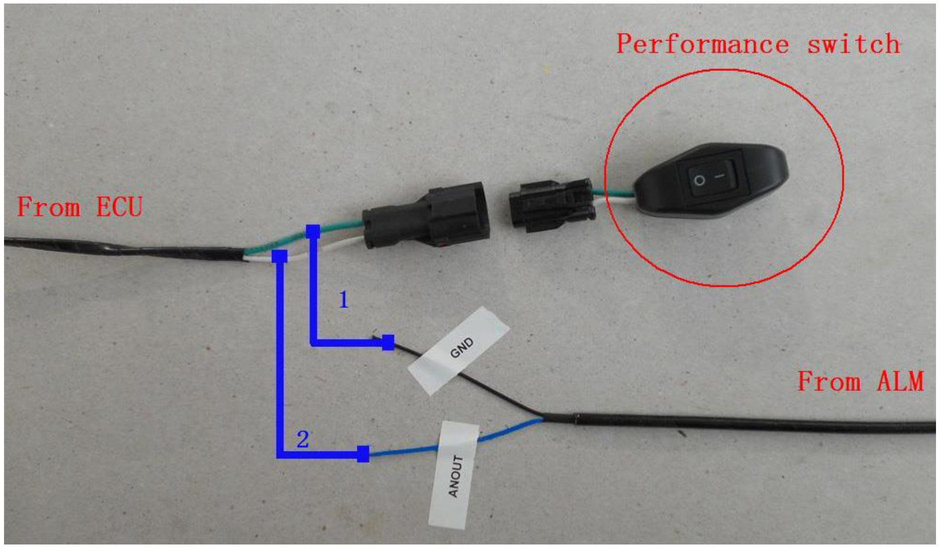
With the Auto-Tuning feature, ECU will read the ALM's real-time lambda input, and automatically adjust the fuel towards the desired Lambda at that RPM and TPS range. ECU will store the learnt data in its own memory. After you run the engine at different operating conditions (RPM & TPS) for a while; it will eventually learn most of operating points. And the engine is tuned then.

All you need to do is to drive your vehicle in different throttle positions and different RPM in “steady state driving”. (Steady state means holding throttle position at fixed position and let the engine run at a certain RPM for about 10 seconds.) , ECU-ALM will take care of the rest.

Note: Auto-tuning only works if the manual tuning is making the engine running in the ballpark. Auto-tuning is supposed to take care of the fine tuning only! It is not supposed to replace the manual tuning. If your engine is not even having a stable running, don't try auto-tuning!

Section 1 Connect ALM to ECU

Previously we used the "Perf-SW", performance switch, wire to connect the ALM input. That causes the loss of dual fuel map feature.

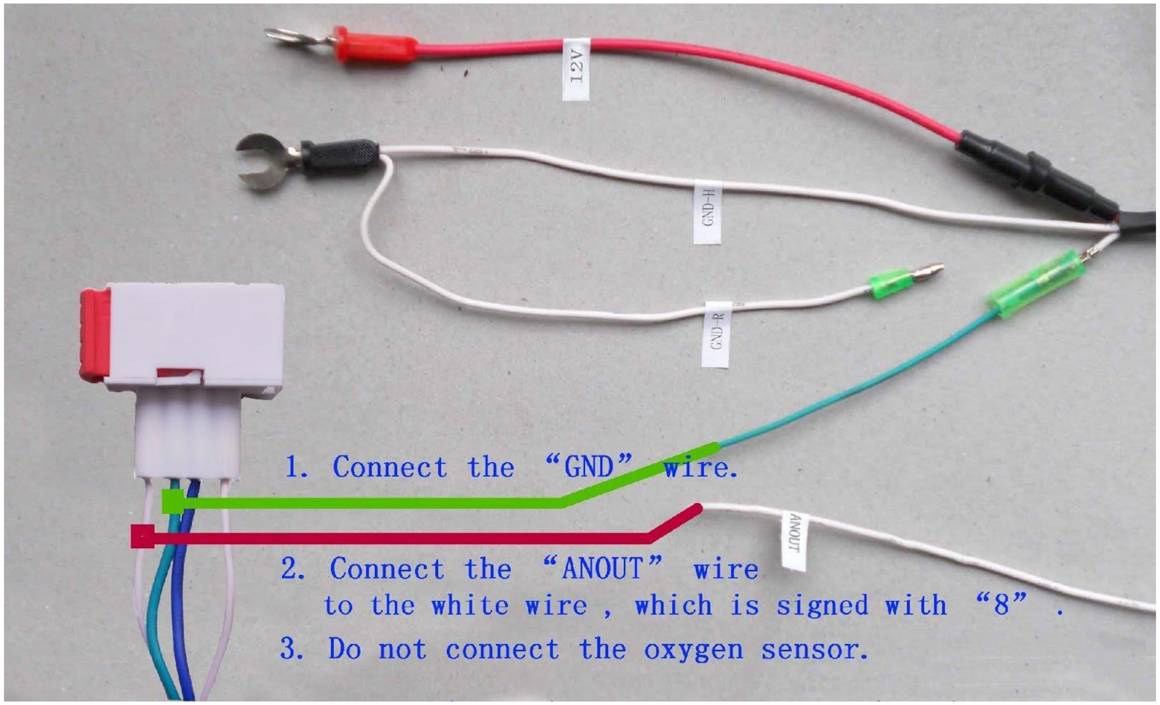


Previous connection

Recently, we have changed that. We are now using the Narrow-band (NB) O2 input wire to connect the ALM analog voltage input, which allows for real-time lambda.

This means our ECU is actually hardware-compatible to both NB O2 and wideband (WB) O2 with the same input wire. The only difference is to download a different CAL file into ECU. This simplifies a lot of wire work.

If you have one of our ALM (with LED display model) wideband kits, you shall connect the ALM to ECU as in the below picture:



Note: the white/red 4-pin connector is the NB O2 sensor connector coming with our harness.
When you connect ALM to ECU for tuning, DO NOT CONNECTS THE NB O2 SENSOR.

Pin definition of the Narrow-band oxygen sensor connector (white/red), from right to left in the picture:

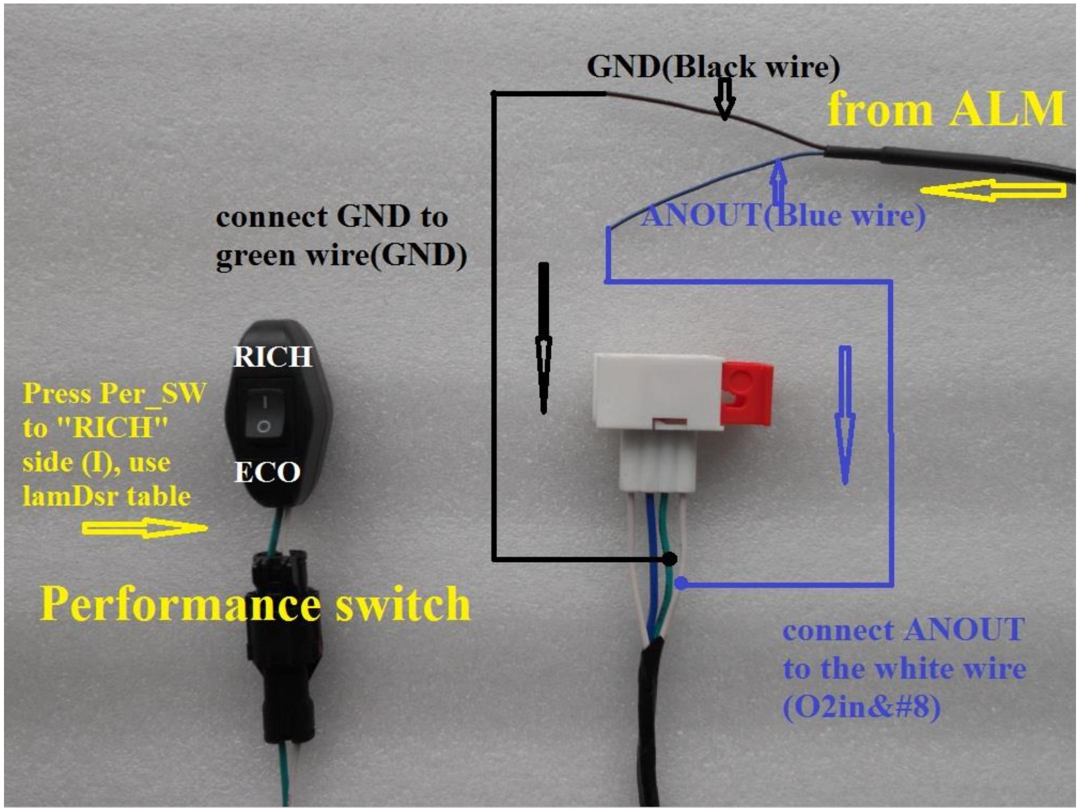
- Heater circuit - (White) – not used for ALM
- Heater circuit + (Blue) – not used for ALM
- Reference Ground (Green)
- **O2in - O2 sensor input (White #8)**

Connect the GND wire which labeled GND-R from harness to green wire from O2 sensor connector

Connect the ANOUT wire from ALM harness to the white wire which is signed #8

ALM-S

If you have one of our ALM-S wideband kits, you shall connect the ALM-S to ECU as in the below picture:



Most of our 4-stroke EFI harness comes with a NB O2 sensor connector (white/red) which has 4 wires. From left to right in the picture:

- Heater circuit - (White)
- Heater circuit + (Blue)
- Reference Ground (Green)
- O2in - O2 sensor input (White)

Our ALM-S harness comes with 2 wires:

- ANOUT (Blue) - analog output representing the lambda
- GND (Black) - reference ground

You need to connect the ANOUT (Blue) to O2in (White) and GND (Black) to Ground (Green).

If your performance switch (Per-SW) is pressed down to the ECO mode, the target Lambda for auto-tuning will be 1.0 (14.7AFR) across the board.

If your Per-SW is pressed down to the RICH mode, the target lambda will be your desired lambda table: MAP_LamDrv_N_TPS. You can open this table in HAECAL and change it as you want.

Section 2 Burn the correct CAL file

After you connect the ALM or ALM-S to ECU as the previous section, you need to burn the correct CAL file into ECU.

When you buy our ALM and ECU combo setup, you will get a special CAL files:

1) "xxxx_ALM_Auto-Tuning_Enabled.CAL". This CAL file will enable the auto-tuning and let ECU keep learning over the life.
2) "xxxx_ALM_Auto-Tuning_Finished.CAL". This CAL file will stop the auto-tuning and retain the learnt data in ECU memory, and then you can use the ECU to read the lambda (AFR).

The way to burn the CAL file:

Run HAECAL and load the correct A2L and the CAL files.

Note: when you burn the CAL file into ECU, always make sure your 12V battery is well charged. Low battery will cause a failed burn, and your ECU will malfunction in that case.
Select "burn to ECU" after you has connected HAECAL to ECU via the serial cable.

Section 3 Let Auto-Tuning Run

After you burn the Auto-tuning enabled CAL file into the ECU. You can start your engine and drive it out. Auto-tuning will start automatically once your engine is warmed up.

Note: engine warm-up condition is engine temperature is greater than 70 degree Celsius, or 158 degree Fahrenheit. Your engine needs to run more than 10-15 minutes to reach that temperature.

Note: To have better auto-tuning opportunities, you need to drive in steady-states with different throttle positions. For example, you drive the vehicle at 5%, 10%, 20% ... 100% throttles, and for every throttle position, you need to keep that for about 10 seconds. One indication of steady-state is that the RPM is relatively stable.

Note: Make sure your 12V battery is well charged, and your charging system has enough power for both EFI and ALM. All together, EFI and ALM require 3A to 5A current draw, which is about 70W total. You need also count the other electrical loads on your vehicle. Low battery or deficient charging system will cause malfunction of the auto-tuning.

Note: Remember, you can use the performance switch (Per-SW) to define what your target lambda (AFR) is. The auto-tuning will tune the lambda to 1.0 across the board in ECO mode, and tune the lambda to your desired lambda table in RICH mode.

Note: If you engine is not even able to keep running, (stall or sluggish), you may need to do some manual tune first to get the AFR to the ball park range. You can send us the logged data and we can do that for you. If your engine has already been setup by us or others with the kit, more than likely we have a calibration file available that’s very close to what you need.

Section 4 Auto-tuning finished

When will the auto-tuning finished? It is depending on your feeling. If you feel the engine has been running much better and stable compared to the beginning, at variant throttle and RPM conditions, you can think the auto-tuning is finished.

After you have finished Auto-Tuning with ALM, you can burn the "xxxx_ALM_Auto-Tuning_Finished.CAL" into the ECU. This CAL file will stop the auto-tuning but retain all the learnt tuning data in the ECU memory. ECU will still read ALM input, but not to use it for auto-tuning any more.

Note: If you load the default Cal file which comes with EFI kit and has no "enabled" or "finished" words in the name, you must disconnect the ALM from the ECU. Because the default CAL file is expecting NB O2 sensor input. You can re-install the NB O2 sensor back to the EFI system and running close-loop controls.

How to run your engine for ALM Auto-tuning?

If you run your engine with ALM auto-tuning function, you need run your engine in stable condition (stable TPS and stable RPM) about 10 seconds, then it will modify the real lambda to desired lambda automatically.

For you to have a high efficiency auto-tuning, you can run your engine with the following progress as a reference.

When you load the correct Enabled CAL file or enable the auto-tuning in HAECAL, and connect the ALM to ECU with above method correctly, then you can start your engine.

Auto-tuning function will be enabled automatically when engine have finished warm-up (engine temperature is higher than 70 degree) or have run about 15 minutes.

Then drive it on road, to have better auto-tuning opportunities, you need to drive in steady-states with different throttle positions at different RPM's. It's better in a long straight road without many traffic lights.

For example, you drive the vehicle at 5%, 10%, 20% ... 100% throttles, and for every throttle position, you need to keep that for about 10 seconds. One indication of steady-state is that the RPM is relatively stable.

Driving conditions: after the engine is warmed up (about 10-15minutes); drive the bike in some kind of steady state: for example, on the flat road, hold the throttle position at 5%; at relative steady vehicle speed, and run it for 10-30s. Then move the throttle opening a little higher, like 10%, and hold it at 10% at relative steady vehicle speed for 10-30s; repeat this for 20%, 30% ...100% throttle positions.

After done a round of throttle positions, key off and wait for a while. This is a full cycle of auto-tuning.

Then repeat the above again if you want to do more auto-tuning.

The reason for key off when finished a driving cycle is that ECU will update the tuned data, and in next running ECU will use the tuned data to run, so it will have a better performance than last running.

Every driving cycle running, the stable condition (TPS and RPM) is not necessary to keep much longer time, just 10 seconds is enough. When every driving cycle is finished, you need key off then key on, engine will be calibrated well in some duty cycles. The real lambda will be closed to desired lambda.

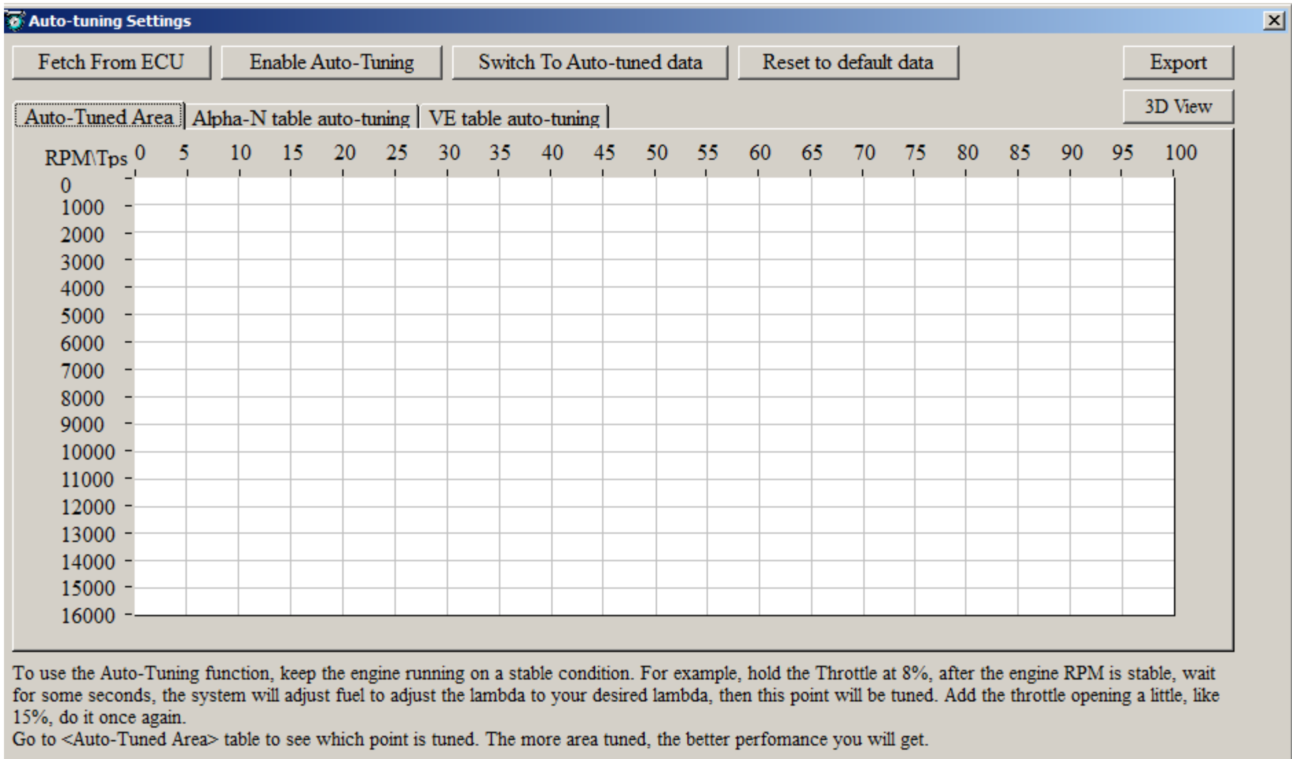
If you have an engine with a transmission, you need run the duty cycle in different gears.

HAECAL operation manual about Auto-tuning

How to bring up the auto-tuning window in HAECAL?

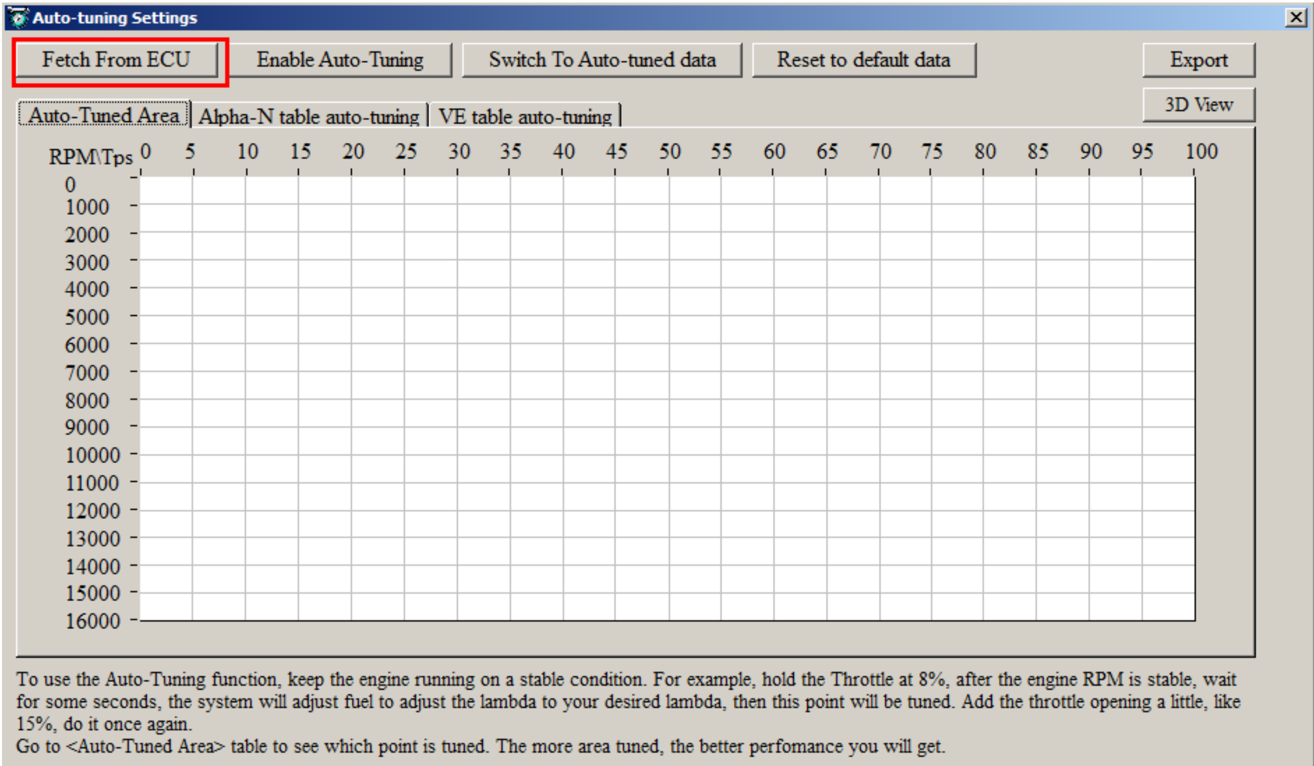
1. Load the correct A2l and CAL file with HAECAL V7.1.9.2 or later version.
2. Bring up the Auto-Tuning window:

Menu->Calibrations->Auto-Tuning settings



How to fetch the data from ECU?

Fetch the current data from ECU; click “Fetch from ECU”, you can get the data in ECU.



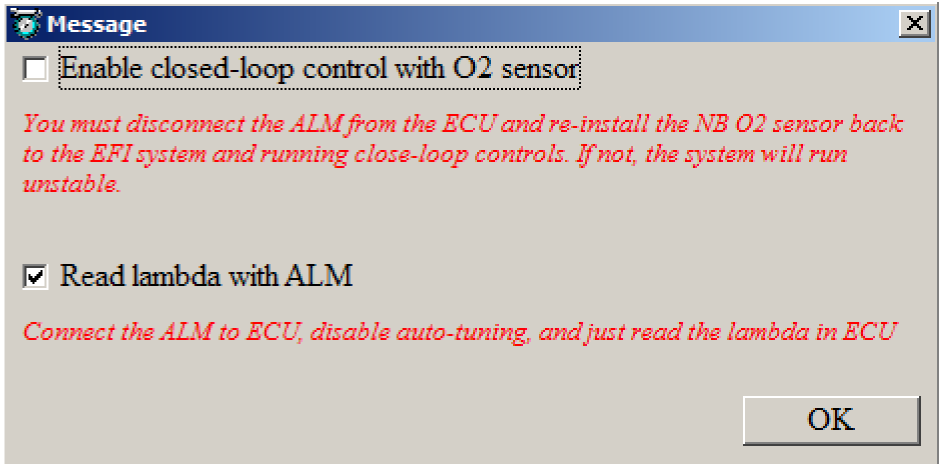
How to enable or disable auto-tuning with HAECAL?

If you load the default CAL file, you can enable or disable auto-tuning.

Enable Auto-Tuning: enable auto-tuning function

Disable Auto-Tuning:

(1): Enable closed-loop control with O2 sensor
(2): Read lambda with ALM



How to switch to use auto-tuned data or to use manual tuned data (default data)?

Switch to auto-tuned data: ECU will use the auto-tuned data to run engine

Switch to manual-tuned data: ECU will use the default MAP value to run engine.

Note: **If you have the Enabled auto-tuning CAL or other corresponding CAL file from us, you don't need change this in HAECAL.**

Load the corresponding CAL file, so you can do the test.

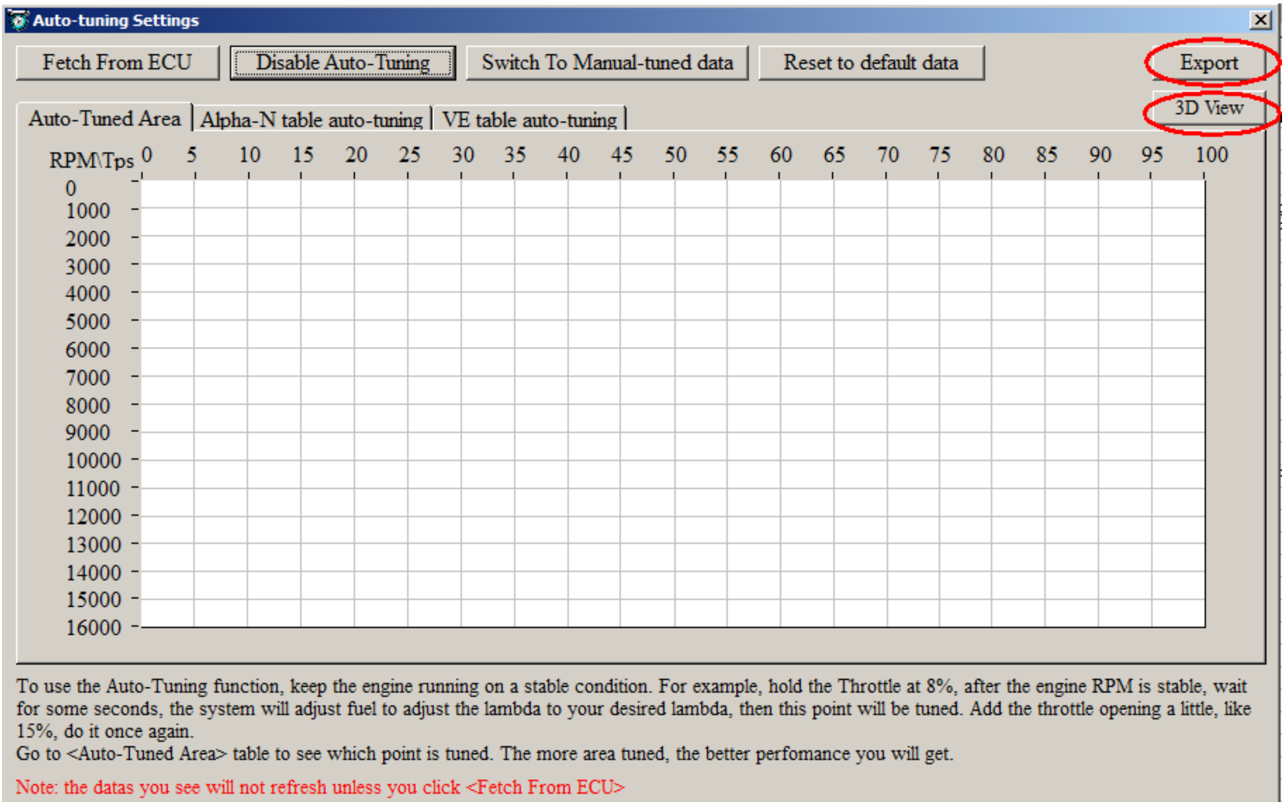
"**xxxx_ALM_Auto-Tuning_Enabled.CAL**". This CAL file will enable the auto-tuning and let ECU keep learning over the life.

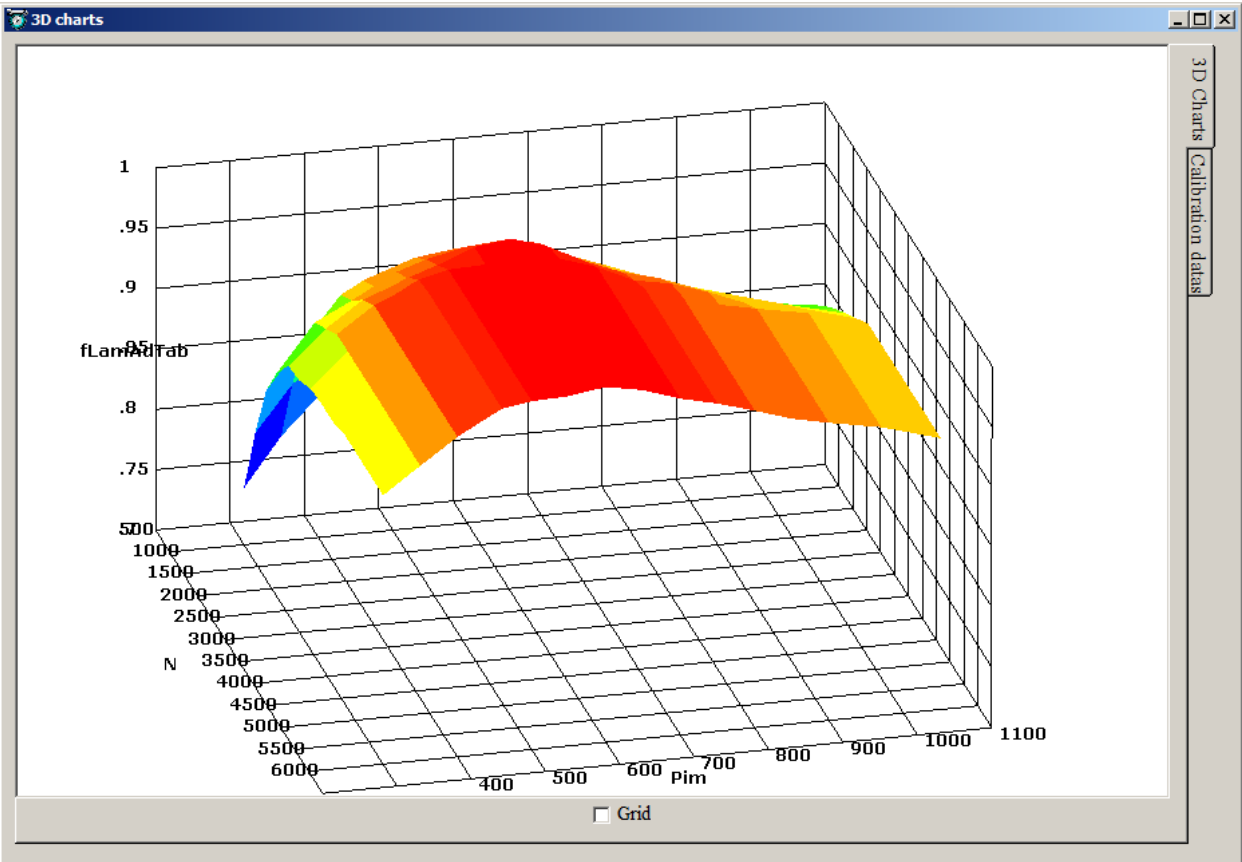
"**xxxx_ALM_Auto-Tuning_Finished.CAL**". This CAL file will stop the auto-tuning and retain the learnt data in ECU memory, and then you can use the ECU to read the lambda (AFR).

These two marked CAL file have been set to use auto-tuned data.

How to save the auto-tuned data or see the table graph?

Export the auto-tuned data to CSV, it's convenient to save or modify the data in this form. You can also use the "3D view" function to see the graph of data.

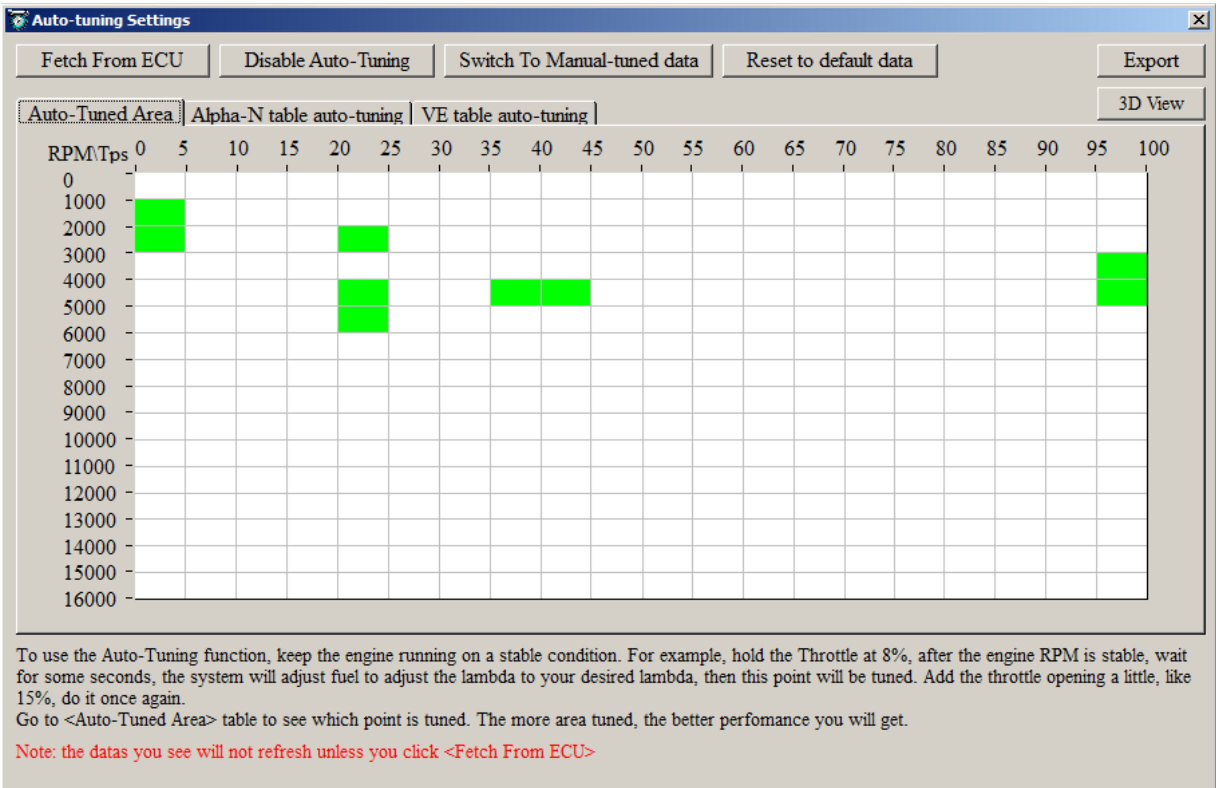




How to know the current operating mode (Tps and RPM) has finished auto-tuning?

The key point is the real-time lambda, making the desired lambda (LamWO2 close to LamDsr).

When you fetch from ECU, you can see the cell is changed to green color in Auto-tuned Area, it means the lambda is closed to desired lambda, in other words, it has finished auto-tuning.



Note: **when the color of cell in Auto-tuned Area has been changed to green, you can finish the auto-tuning, and disable it to see the performance.**

Variables:

SLM_StableCondition=1, means current condition is finished.

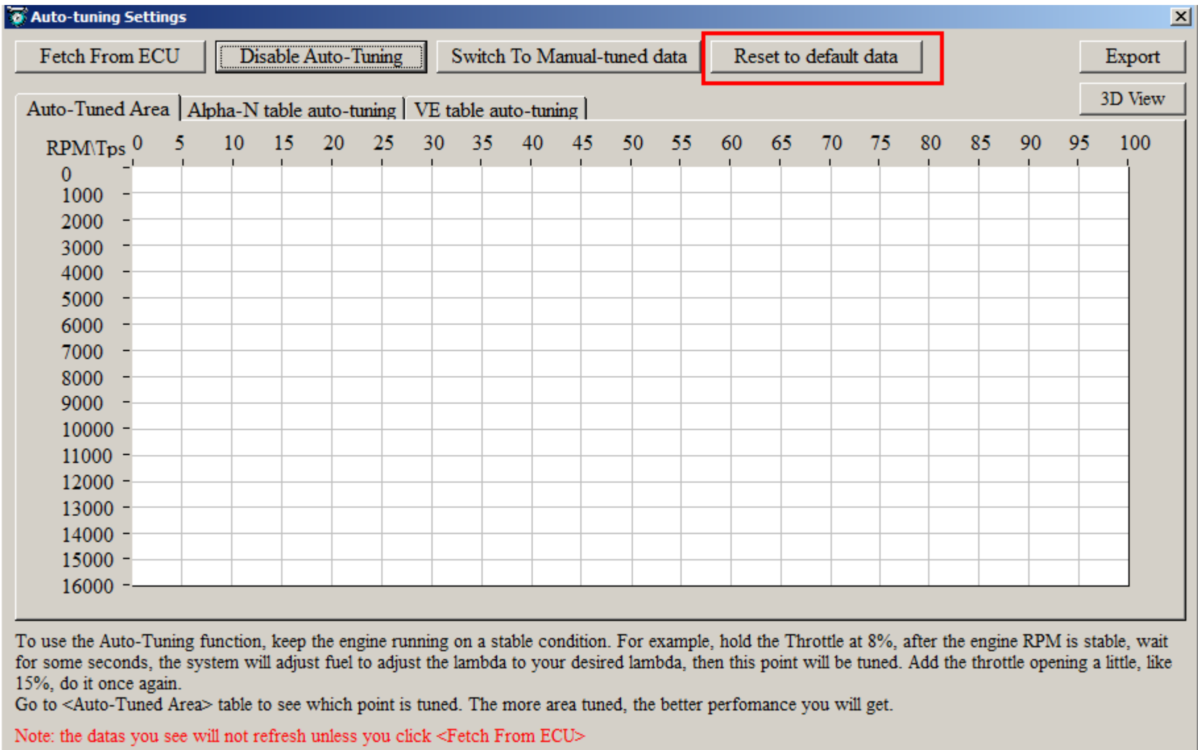
B_fLamAd=1, the auto-tuning function has been enabled, so you can use the auto-tuning.

(Add this variable: Menu->Variables->display type->list,
Then Right click in the selected variables list window, and add variable.)

The longer you run with the ALM auto-tuning on, the closer the calibration MAP table and lambda are to the real-time lambda.

How to reset to default data?

If the auto-tuned data is not stable, or you don't want the auto-tuned data and want to start a new auto-tuning, you can erase the auto-tuned data and reset it to default data.



Note: **You can only reset the data by using the HAECAL operation; other operations such as burn to ECU or power off can't do this.**

Attention again: when you have finished a phase of auto-tuning with ALM, you need to stop the engine and key-cycle (key off and then key on), the auto-tuned data will be saved in the ECU, in the next auto-tuning running, it will use the auto-tuned saved data to run.

(You have to enable "Switch to auto-tuned data" button)

Note: there is a more complicated method in our ECU to do auto tuning with an ALM, as a close loop control method, which can automatically tune the fuel for your "desired lambda table". But that requires a lot of detailed conditioning, to make it work right. We do not recommend the users to use it. That is for only very experienced users and be familiar with our software.

Chapter 11 On-the-fly Calibration Data

On-the-fly calibrations are something must to have when tuning the engine on the dyno, so that you can make you calibration changes taking effect immediately by hit a "ENTER" key. No need to "Burn to ECU" or reprogram the ECU. Only by this way, you can keep the engine running without interrupted and at the same time find the sweet spots of the certain operating conditions. And you can tune your fuel / spark maps very quickly.

But the drawback of the "on-the-fly" calibrations is to use a lot of memory of ECU. Basically any calibration data you want to do "on-the-fly" changes, you must double or even triple the size of the memory for that set of data. That would significantly increase the ECU cost.

We have a compromise to do this: only make certain critical calibration maps as "on-the-fly" capable. And most other non-critical calibrations, which are seldom changed by customers, stay the old way.

Mainly for most customers, the calibrations that need to be tuned for their engines are basic Fuel maps (VE table, TPS-load table), Spark maps, and some supplemental fuel/spark characteristic curves.

You can also use "**Save as**" feature at "File" menu to save your new calibrations to a new CAL file. And later on load it into HAECAL.

At this moment the below calibration maps, curves, and values are made "on-the-fly" capable:

RAM_MAP_LdTp_N_Tps:

"Tps based Load mapping" ; this is the most important one for dyno tuning. It determines the commanded fuel at different throttle position and RPM. Especially at WOT condition, fuel is solely determined by this table. Because on the dyno, the first thing to tune is the max power of the engine, and the max power is realized to optimize the AFR at the 100% TPS and max RPM. And the AFR at max RPM and 100%TPS is determined by this table.

RAM_MAP_fVe_Map_N:

"Volumetric efficiency, dependent on MAP sensor and RPM" ; this is the fundamental calibration map of an engine, also called "speed-density" method.

It determines the commanded fuel at different manifold pressure and RPM. Basically, the volumetric efficiency determines the fresh air amount in the cylinder, and therefore the fuel needed for optimized AFR.

Our ECU software use the blended "speed-density" and "Alpha-N" methods, and at most RPM ranges and part loads, speed-density is primary method to calculate Fuel. Actually Alpha-N is only used at WOT and high RPM. So for most operating conditions, you should tune the VE table for optimized AFR.

Other on-the-fly calibrations:

Basic Ignition angel table: **RAM_MAP_Iga_N_Ld**

After start and warm up factor: **RAM_CUR_fAst_TmSta, RAM_CUR_fWmp_Tm**
Globe enrichment factor: **RAM_VAL_fFlApp**

Note: when you tune the data on-the-fly, please "Stop Measuring". Otherwise it will not take effect.

And if the ECU power off, the data will be lost, so you need save the data and burn to ECU after you think the data is OK.

Chapter 12 about HAE software tools and ECU communications

Note: We do not support the 3rd party USB-RS232 adapters, even though they might work sometime. The problem is that those consumer electronics rated USB adapters only works in a noise-free environments. This means, once the engine is running, it generates a lot of electronic magnetic noises. And those USB adapters, though looking pretty, will not stay working when you are driving. That's why we developed our own.

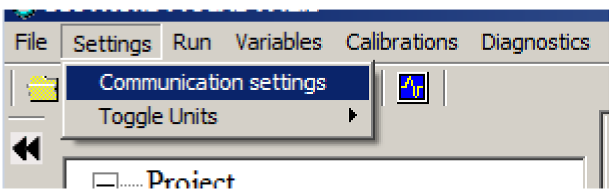
Your purchased one is only looking "nice"; they will not working nice once the engine is running.

Recommend you to buy a new USB adapter from us and we can help you from there.

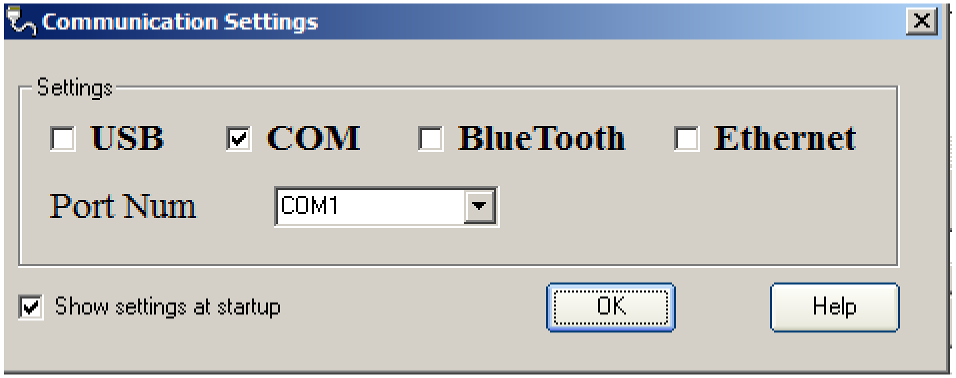
12.1 Check A2l or CAL which should be matching ECU

How to find the A2L or CAL files is matching with ECU?

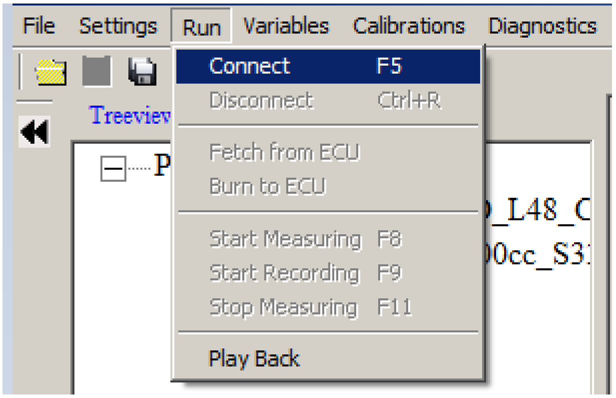
Connect your laptop with a USB or RS232 to ECU harness. First do some communications settings in HAECAL.



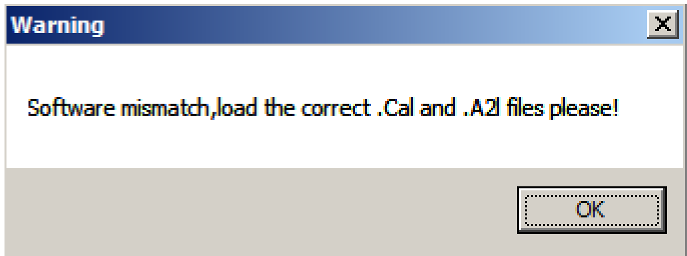
Select to the USB or COM or other according which you use. Then click "OK".



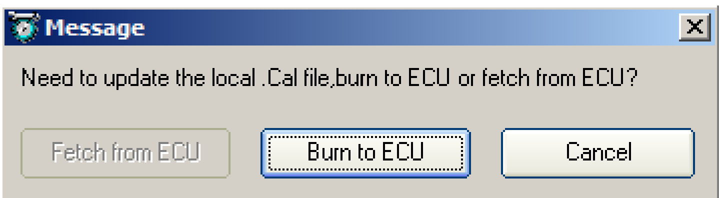
Connect to ECU: "Run->Connect"



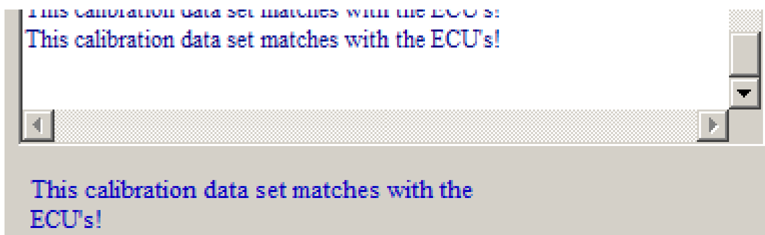
If it pop-up a warning box, so it is mismatching with your ECU. **Please connect us to get the right file.**



If it pop-up a message box, it means the settings of CAL is different from the ECU, you need to "Burn to ECU".



On the other hand, it will pop up a message "This calibration data set matches with the ECU's". It is right.



12.2 Burn to ECU successfully

How to make sure the "burn to ECU" is successful?

First, make sure the ECU is power on, and connect your laptop with a USB or RS232 to ECU harness.

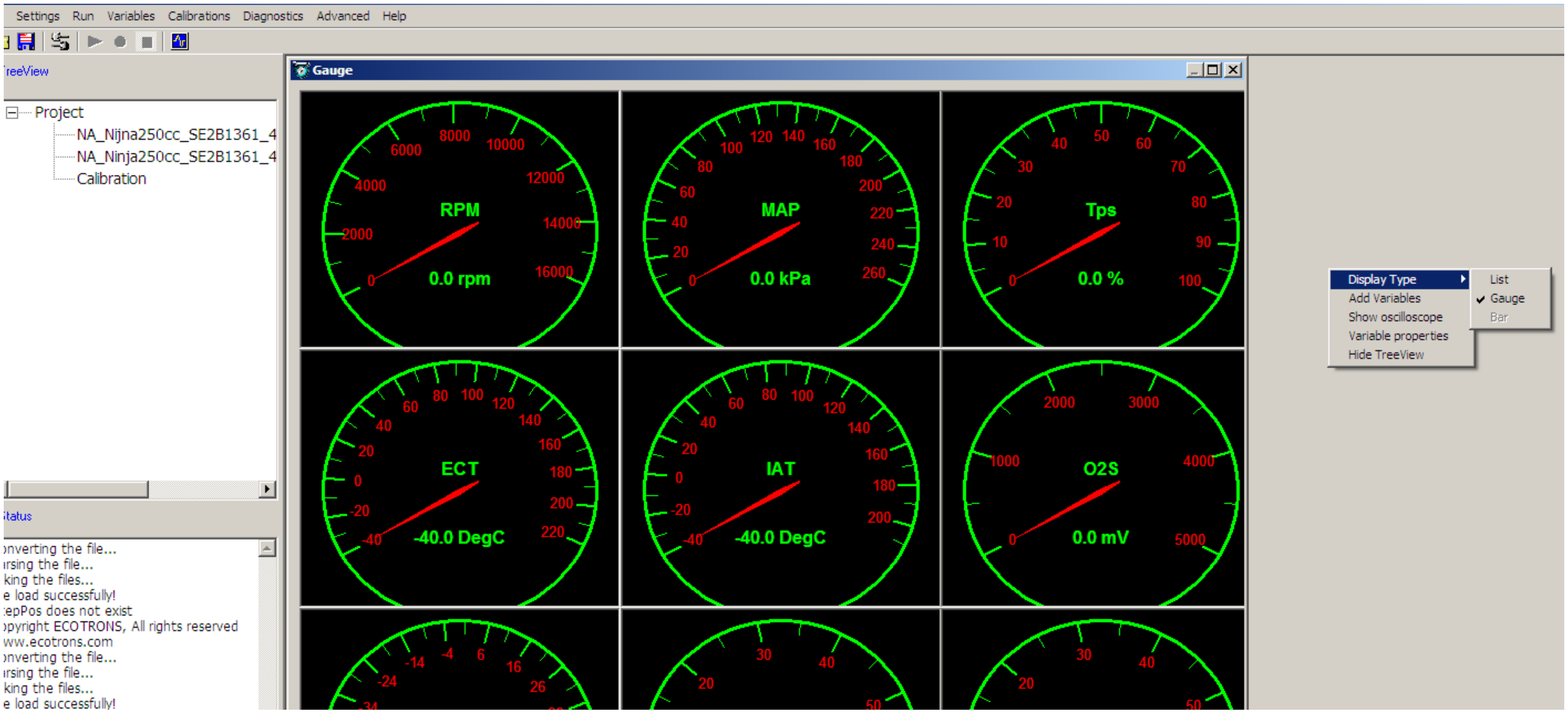
When to burn to ECU, the power is still on and don't put the power switch off.

And if burn to ECU successfully, you will hear fuel pump running for a couple seconds. If not, please do again and check all of connection is right.

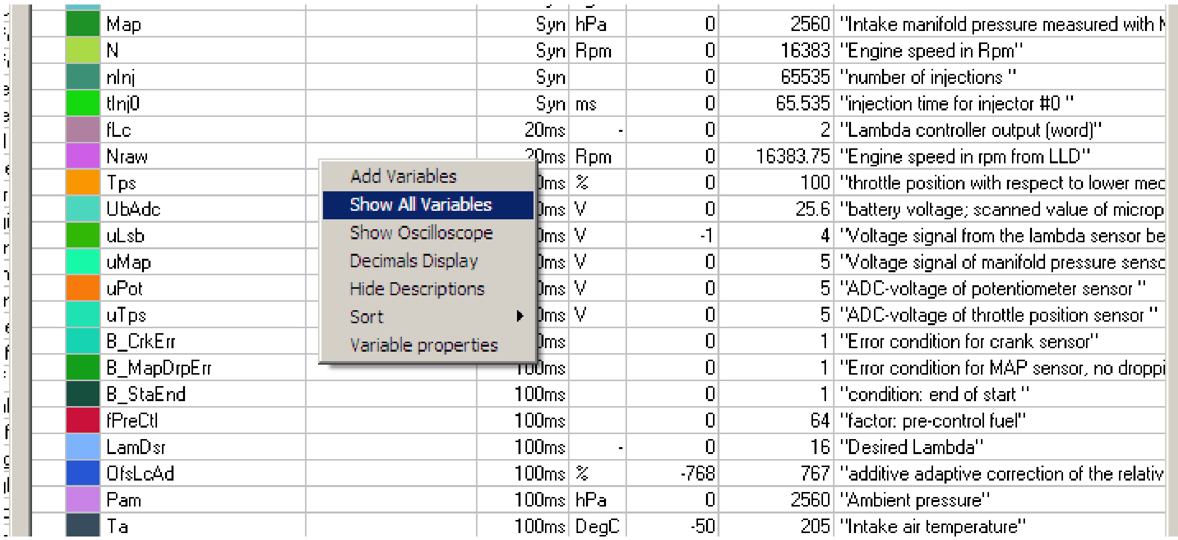
12.3 load configuration in variables list

For diagnosis and analyze the engine issue, sometime you need load the ".ini" file which we send you, in HAECAL, and log data. You can do this by following example operation.

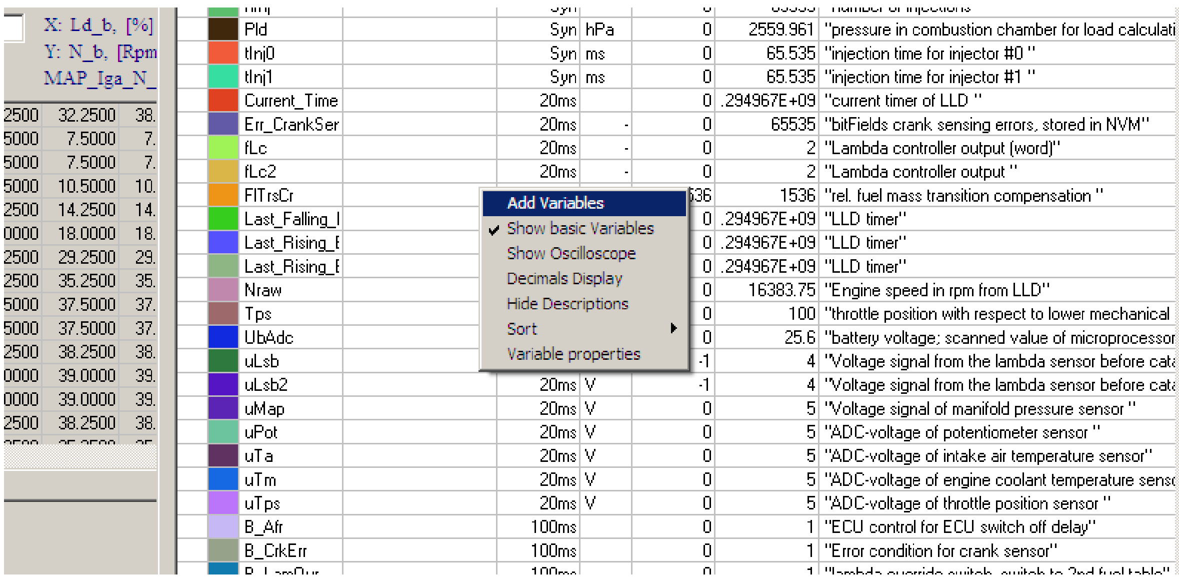
1) Choose "list": In the interface Right Click ,then click "Display type→List"



2) In the list view , right-click , first , Click “Show All Variables“ , then right-click choose “Add Variables”

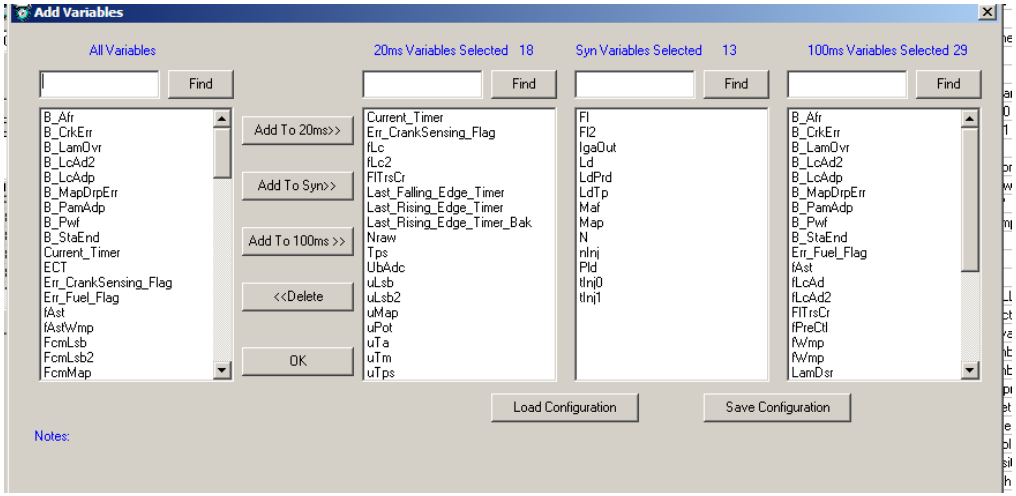


Map Syn hPa 0 2560 "Intake manifold pressure measured with I
N Syn Rpm 0 16383 "Engine speed in Rpm"
nInj Syn 0 65535 "number of injections "
tInj0 Syn ms 0 65.535 "injection time for injector #0 "
fLc 20ms - 0 2 "Lambda controller output (word)"
Nraw 20ms Rpm 0 16383.75 "Engine speed in rpm from LLD"
Tps ms % 0 100 "throttle position with respect to lower mec
UbAdc ms V 0 25.6 "battery voltage; scanned value of microp
uLsb ms V -1 4 "Voltage signal from the lambda sensor be
uMap ms V 0 5 "Voltage signal of manifold pressure sensc
uPot ms V 0 5 "ADC-voltage of potentiometer sensor "
uTps ms V 0 5 "ADC-voltage of throttle position sensor "
B_CrkErr ms 0 1 "Error condition for crank sensor"
B_MapDrpErr 100ms 0 1 "Error condition for MAP sensor, no droppi
B_StaEnd 100ms 0 1 "condition: end of start "
fPreCtl 100ms 0 64 "factor: pre-control fuel"
LamDsr 100ms - 0 16 "Desired Lambda"
OfsLcAd 100ms % -768 767 "additive adaptive correction of the relativ
Pam 100ms hPa 0 2560 "Ambient pressure"
Ta 100ms DegC -50 205 "Intake air temperature"
Add Variables
Show All Variables
Show Oscilloscope
Decimals Display
Hide Descriptions
Sort
Variable properties

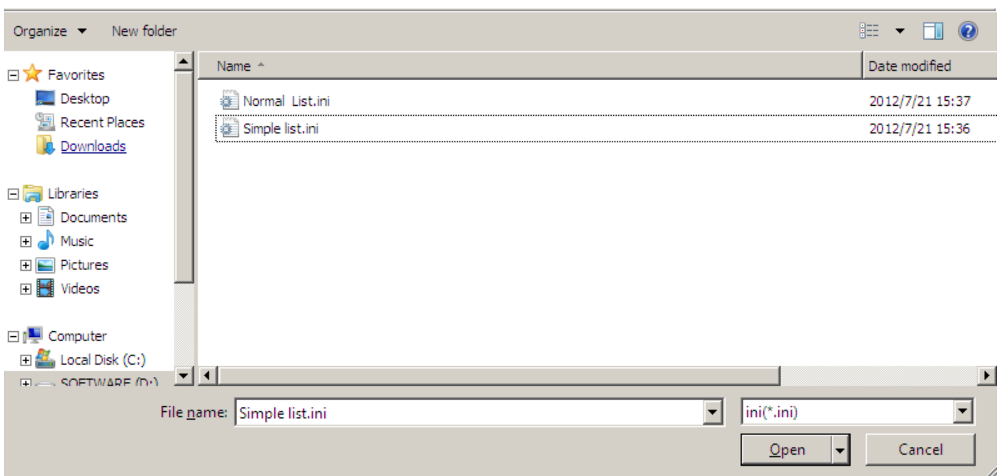


X: Ld_b, [%]
Y: N_b, [Rpm
MAP_Iga_N_
Pld Syn hPa 0 2559.961 "pressure in combustion chamber for load calculati
tInj0 Syn ms 0 65.535 "injection time for injector #0 "
tInj1 Syn ms 0 65.535 "injection time for injector #1 "
Current_Time 20ms 0 .294967E+09 "current timer of LLD "
Err_CrankSer 20ms - 0 65535 "bitFields crank sensing errors, stored in NVM"
fLc 20ms - 0 2 "Lambda controller output (word)"
fLc2 20ms - 0 2 "Lambda controller output "
FlTrsCr 1536 "rel. fuel mass transition compensation "
Last_Falling_I 0 .294967E+09 "LLD timer"
Last_Rising_I 0 .294967E+09 "LLD timer"
Last_Rising_I 0 .294967E+09 "LLD timer"
Nraw 0 16383.75 "Engine speed in rpm from LLD"
Tps 0 100 "throttle position with respect to lower mechanical
UbAdc 0 25.6 "battery voltage; scanned value of microprocessor
uLsb -1 4 "Voltage signal from the lambda sensor before cat
uLsb2 20ms V -1 4 "Voltage signal from the lambda sensor before cat
uMap 20ms V 0 5 "Voltage signal of manifold pressure sensor "
uPot 20ms V 0 5 "ADC-voltage of potentiometer sensor "
uTa 20ms V 0 5 "ADC-voltage of intake air temperature sensor"
uTm 20ms V 0 5 "ADC-voltage of engine coolant temperature sens
uTps 20ms V 0 5 "ADC-voltage of throttle position sensor "
B_Afr 100ms 0 1 "ECU control for ECU switch off delay"
B_CrkErr 100ms 0 1 "Error condition for crank sensor"
Add Variables
Show basic Variables
Show Oscilloscope
Decimals Display
Hide Descriptions
Sort
Variable properties

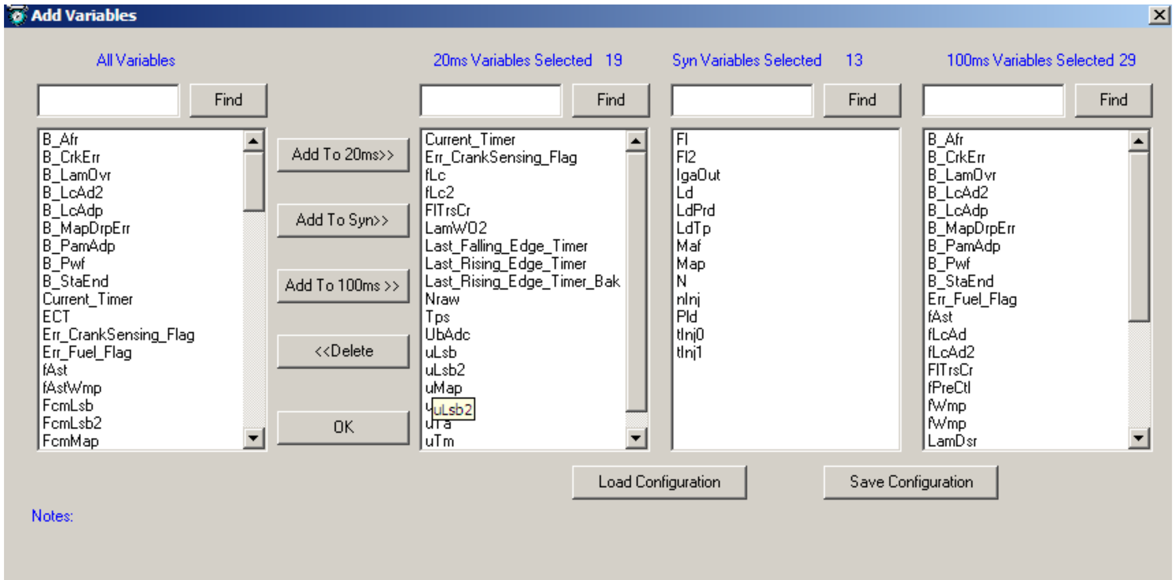
3) Click"Load Configuration"



4) Choose "Simple list.ini", and then click "open"



5) Click “OK”


Add Variables
All Variables
20ms Variables Selected 19
Syn Variables Selected 13
100ms Variables Selected 29
Find
Find
Find
Find
B_Afr
B_CrkErr
B_LamOvr
B_LcAd2
B_LcAdp
B_MapDrpErr
B_PamAdp
B_Pwf
B_StaEnd
Current_Timer
ECT
Err_CrankSensing_Flag
Err_Fuel_Flag
fAst
fAstWmp
FcmLsb
FcmLsb2
FcmMap
Add To 20ms>>
Add To Syn>>
Add To 100ms >>
<<Delete
OK
Current_Timer
Err_CrankSensing_Flag
fLc
fLc2
FlTrsCr
LamWO2
Last_Falling_Edge_Timer
Last_Rising_Edge_Timer
Last_Rising_Edge_Timer_Bak
Nraw
Tps
UbAdc
uLsb
uLsb2
uMap
uLsb2
uTa
uTm
Fl
Fl2
IgaOut
Ld
LdPrd
LdTp
Maf
Map
N
nInj
Pld
tInj0
tInj1
B_Afr
B_CrkErr
B_LamOvr
B_LcAd2
B_LcAdp
B_MapDrpErr
B_PamAdp
B_Pwf
B_StaEnd
Err_Fuel_Flag
fAst
fLcAd
fLcAd2
FlTrsCr
fPreCtl
fWmp
fWmp
LamDsr
Load Configuration
Save Configuration
Notes: