



## **Turbonator Tuners Guide**

Robert Lloyd  
BoostButton.com

Draft 1  
Updated - 1/12/11

## Section 1 - Fuel Tuning

### 1.1 - Wide-Open-Throttle

We'll start with WOT operation because it's the easiest to tune and the easiest to understand. The simplest situation for WOT is comes after the engine is at normal operating temperature. So, for this section, I'm going to assume that the engine is up to temperature. I'll go into the temperature compensation factors in a later section.

For Wide-open operation, the fuel is calculated from these values:

$$\text{(WOTTBL\_FuelFullThrottle + SparkScatterFuelStabilizationValue )} * \text{SumOfFuelModifiers - Factor\_StartFuelDecayIntoRunFuel}$$

WOTTBL is the Full-Throttle fuel PW table vs. MAP. The value in this table is an ideal PW based on MAP and a target A/F. This target A/F is hard-coded into the PW table – it's not a value that the computer uses for feedback or anything like that. It's just the value that Chrysler (or the tuner) chose when the table was created. Using CHeM or MP Tuner, you can see the effect of changing this target A/F on the PW

On the other hand, the spark scatter fuel stabilization value should only come into play at idle, and should be 0 at WOT. I'll cover this value in detail in the idle control section.

Also, the Factor\_StartFuelDecayIntoRunFuel only comes into play during the first few seconds after starting the engine and also can be considered to be zero for most WOT applications. You definitely should not be firing the engine and going straight to WOT. But, if you did, this factor would affect the fuelling.

So, that really leaves only the SumOfFuelModifiers and the WOTTBL PW values for WOT fuel calculation. WOTTBL is the 16-bit base PW lookup from MAP, the SumOfFuelModifiers value is a fraction (or percentage) and is calculated as follows (for WOT operation, there are additional factors for P/T):

$$\text{SumOfFuelModifiers} = \text{Cold Enrich factor} * \text{Cold Load enrich factor} * \text{Pump Eff}$$

The cold enrich factor is calculated from the 3 cold start tables (A,B,C) and the time constants that phase them out. Once the engine is fully warmed up, the A and C factors go to 1 and so they will not have an effect on WOT fuel. So, we can ignore them (for now, we'll get back to them for cold enrichment).

The CurveB factor is a little bit different. For CurveB there is no decay factor for time – it is always active. At normal operating temperature, the factor goes to 1. However, CurveB is also multiplied by a factor (FCRCVB) at WOT. This value comes from a lookup table vs. RPM and the scale is 100-200%.

The Cold Load Enrichment factor only comes into play during warm-up.

The Pumping Efficiency table is always used.

So, at WOT, and under normal operating conditions (ie, post warm-up period), there are only 3 values that affect fuel:

**WOTTBL (the main PW lookup) \* PEFTBL (Pumping Efficiency) \* CurveB  
\* FCRCVB (the CurveB WOT Multiplier)**

The only inputs to these tables are MAP, RPM, and Coolant temp.

### **WOT Tuning notes:**

When tuning for WOT, typically you set the target A/F using the main fuel table (WOTTBL). CHem or MP Tuner are very useful for helping you set these A/F values. If the actual WOT A/F is very far from this target, you should make adjustments with the PumpEff table.

I would advise caution when adjusting the CurveB or FCRCVB tables unless you are working with a highly modified cylinder head or intake system.

In setting up the Pumping Efficiency table, I would start with a stock table appropriate for your engine (is, use a 2.2 table for a 2.2 engine, 2.5 Masi, etc.). Generally speaking, cylinder head porting and/or high-lift cams tend to rotate the Pump Eff table counter-clockwise about the peak. So, the higher RPM range increases efficiency and the lower RPM range loses efficiency. It's also worth noting that Chrysler left the beginning and end of this table open. It's not realistic that the pumping efficiency is constant above 6000rpm. But, in setting the table up this way, they ensure that there is an excess of fuel. This helps keep cylinder temperatures and pressures down at these limits. This helps headgasket durability, but it kills power.

### 1.2.1 Part Throttle (Cruise)

Part throttle cruise is calculated similarly to WOT except that the adaptive memory and O2 feedback come into play. For P/T fuel:

$$(\text{BasePWLookup} + \text{SparkScatterFuelStabilizationValue}) * \text{SumOfFuelModifiers} - \text{Factor\_StartFuelDecayIntoRunFuel}$$

Where the BasePWLookup is the lowest of these 2 values:

$$(\text{MAPTBL\_FuelPartThrottle} * \text{Timer\_OpenLoopFraction}) \text{ or } \text{WOTTBL\_FuelFullThrottle}.$$

Like WOT, the SparkScatterFuelStabilizationValue should be zero except at idle, and Factor\_StartFuelDecayIntoRunFuel should be zero a few seconds after starting.

The 'SumOfFuelModifiers' value is calculated the same as for WOT with one exception: At part throttle, the adaptive memory value is included. The adaptive memory value can be +/- 25%. I'll show how the O2 controller works and how adaptive memory is calculated in the next chapter. For P/T operation, the 'SumOfFuelModifiers' value is calculated as:

$$\text{SumOfFuelModifiers} = \text{Cold Enrich factor} * \text{Cold Load enrich factor} * \text{Pump Eff} * 1 +/- \text{ValueFromAdaptiveMemory}$$

The Cold Enrich, Cold Load Enrich, and Pumping Efficiency factors are the same as for WOT. The adaptive memory value depends on the operating parameters of the engine and the O2 correction. There are 12 'cells' separated by RPM and MAP:

Auto Cal Factors Table

Closed throttle (idle) = cell 12, otherwise:

	8192	<b>1</b>	<b>3</b>	<b>5</b>	<b>7</b>	<b>9</b>	<b>11</b>
	R1	<b>0</b>	<b>2</b>	<b>4</b>	<b>6</b>	<b>8</b>	<b>10</b>
<b>RPM</b>	0	MP1	MP2	MP3	MP4	MP5	14.7
	<b>MAP</b>	→					

R1 and MP1 thru MP5 are all constants and can be changed by the tuner (if desired). Really, tuning these points should not normally be necessary.

The value stored in the cells comes from the O2 controller. Basically, if the O2 does not toggle soon enough, then the adaptive cell gets updated with a new value. The value is signed and can be +/- 25%.

The adaptive memory value is a RAM variable calculated by the O2 Ramp Controller. So, these values cannot be tuned directly. But, they are a good source of feedback regarding the state of tune.

### 1.2.2 O2 Controller Ramp Calibration

The O2 sensor is monitored ONLY for "rich/lean". A change from one to the other is a "toggle". The system constantly toggles between rich and lean so that the overall average is around stoichiometric.

This system divides a range of RPMs and MAP readings into "cells". The number of "kicks", or increases/decreases in fuel required to toggle the O2 reading, is different for each of these cells. This allows some fairly powerful flexibility in tuning. The only time you might want to adjust these values is to speed or slow the adjustment being performed.

Each cell has a 'kick' value. This is the amount of change applied to the running fuel until a rich/lean toggle occurs. The net result is the familiar rich/lean flashing that your O2 sensor gauge shows you.

The 'limits' are the maximum number of kicks that may be applied to each cell. If the O2 sensor fails to toggle after reaching the limit for more than a few sec, either the AutoCal cell gets changed (in the case of the secondary ramp) or a code for the O2 Sensor is stored (in the case of the primary ramp).

Primary Ramp vs. Secondary Ramp:

The Primary ramp is responsible only for the normal O2 controller feedback and setting the O2 sensor codes. The Secondary ramp is also responsible for O2 feedback, and also for updating the adaptive memory cells.

O2 Controller Cells

Closed Throttle (Idle) = Cell 9, otherwise:

<b>RPM</b> ↑	HiRPM					
		<b>2</b>	<b>5</b>	<b>8</b>		
	RPM2	<b>1</b>	<b>4</b>	<b>7</b>		
		<b>0</b>	<b>3</b>	<b>6</b>		
	RPM1					
		<b>0</b>				
		MAP1		MAP2		HiMAP
	<b>MAP</b>	→				

For each cell, there are 2 numbers. The first number is "rich" value, the second is "lean" value. If the O2 sensor is returning a lean value, "rich" kicks are added. If the O2 sensor is returning a rich value, "lean" kicks are added.

$$\text{KICK} = (\text{factor} * 512) - 512$$

example: 5% ->  $1.05 * 512 - 512 = 25.6 \text{ (dec)} = 0x19$  ; each bit is  $\sim 0.2\%$

**P/T Tuning Notes:**

**Turning Off AutoCal (Adaptive memory):**

Add this sub-section...

**Tuning for lean cruise:**

Add this sub-section...

### **1.2.1 Part Throttle Decel**

Part throttle decel fuel is calculated like the cruise fuel, except that additional factors are used to calculate the SumOfFuelModifiers value - Factor\_DecelLeanoutWhenThrottleDecreasing and Factor\_DecelLeanoutWhenMAPDecreasing. Essentially used to remove fuel when in decel (Accel fuel or accel pump function is handled separately and I'll show that in a little bit).

## 1.2.2 Cold Enrichment

### Cold Enrichment Factor Calculation (CNRICH)

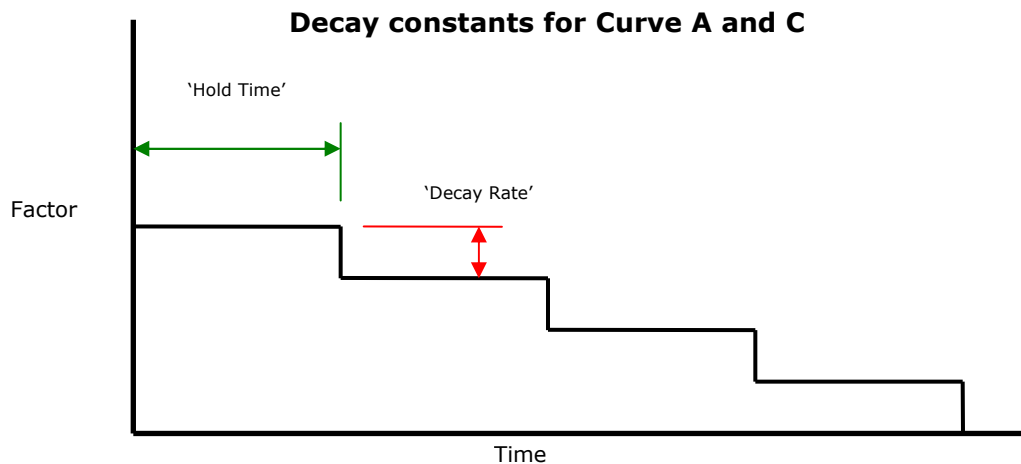
$$1 + ((\text{CURVEA} * \text{ACVDCY}/256 + \text{CURVEC} * \text{ACVDCY}/256 + \text{CURVEB} * \text{FCRCVB}/256) * 2 + (\text{HMPMUL} * \text{HIGHMP}/32) * \text{BAROCP}/256$$

CURVEA, CURVEB, CURVEC Tables:

$X = \text{Engine Temp DegC} + 128$   
(Engine Temp is scaled between -128C and +128C or -198F and +262F)

$Y = (\text{Factor} - 1) * 128$   
(where the factor is between 1.00 and 2.99)

ACVDCY is the calculated decay factor for the A and C cold enrichment tables. This value decreases as the engine warms up. The factor is calculated from TMPHLD, CLDHLD, HOTHLD, DCYTMC, DCYTMH, CCURV, and DCYCC. For Curve A, there are 2 decay rates – a hot decay and a cold decay. TMPHLD is the switch point for CurveA hot/cold decay. CLDHLD and DCYTMC are the CurveA cold hold time and decay rate; HOTHLD and DCYTMH are the CurveA hot hold time and decay rate. For CurveC, there is no temperature switch and the hold time is defined by CCURV and the decay rate by DCYCC. Basically, the 'Hold Time' is the time between reductions of the factor, and the 'Decay Rate' is the amount each factor is reduced per cycle. Eventually, the factors go to 1 either thru time or temperature changes.



The cold load enrichment factor is calculated from the 4 cold load tables (2 from MAP, 2 from coolant temp).

Load Enrichment:  
Cold Load Factor ==  $\text{HIGHMP} * \text{HMPMUL}/32$

HIGHMP Factors:  
X - Engine Temp



Y - (% enrichment per Torr/100) \* 32 \* 5.88 \* 256

HMPMUL Factors:

X - MAP in Torr \* 0.17

Y - MAP in Torr \* 0.17

And, finally, the pumping efficiency is taken from the PumpEff table. Each of the factors is a percentage. They are multiplied together to get the final SumOfFuelModifiers (also a percentage). That final FuelModifier is then used to scale the base WOT PW as stated above.

## **Section 2 - Ignition Tuning**

### **2.1 - Wide-Open-Throttle**

## **Section 3 – Boost Control Tuning**

### **3.1.1 – Boost Target (Except for VNT boost Control)**

The Chrysler boost target calculation is actually pretty simple. There are several tables that lookup the boost relative to some input value (RPM, coolant temp, charge temp, speed, time in boost, etc.). The result of each of these lookups is compared and the lowest value is chosen as the current boost target. Simple.

When setting up these tables for higher boost, make sure the highest boost setting in each table is set to your new desired boost. Personally, I would leave the lower boost settings in the coolant temp table. Boosting on a too-cold or too-hot engine is not a good idea. You can usually level the other tables to your max boost.

### **3.1.2 – Boost Target (VNT boost Control – T-SBEC only)**

This section not finished yet.

### **3.2.1 – Wastegate Duty Cycle Lookup and Calculation (Except for VNT boost Control)**

The wastegate duty cycle lookup/calculation is a little more complicated than the boost target lookup, but still pretty simple. Turbonator LM is different from Turbonator SMEC and SBEC in that it is based on T2 code and so uses the T2 style boost control. Turbonator SMEC/SBEC are based on the T1 code and so use the T1 style boost control.

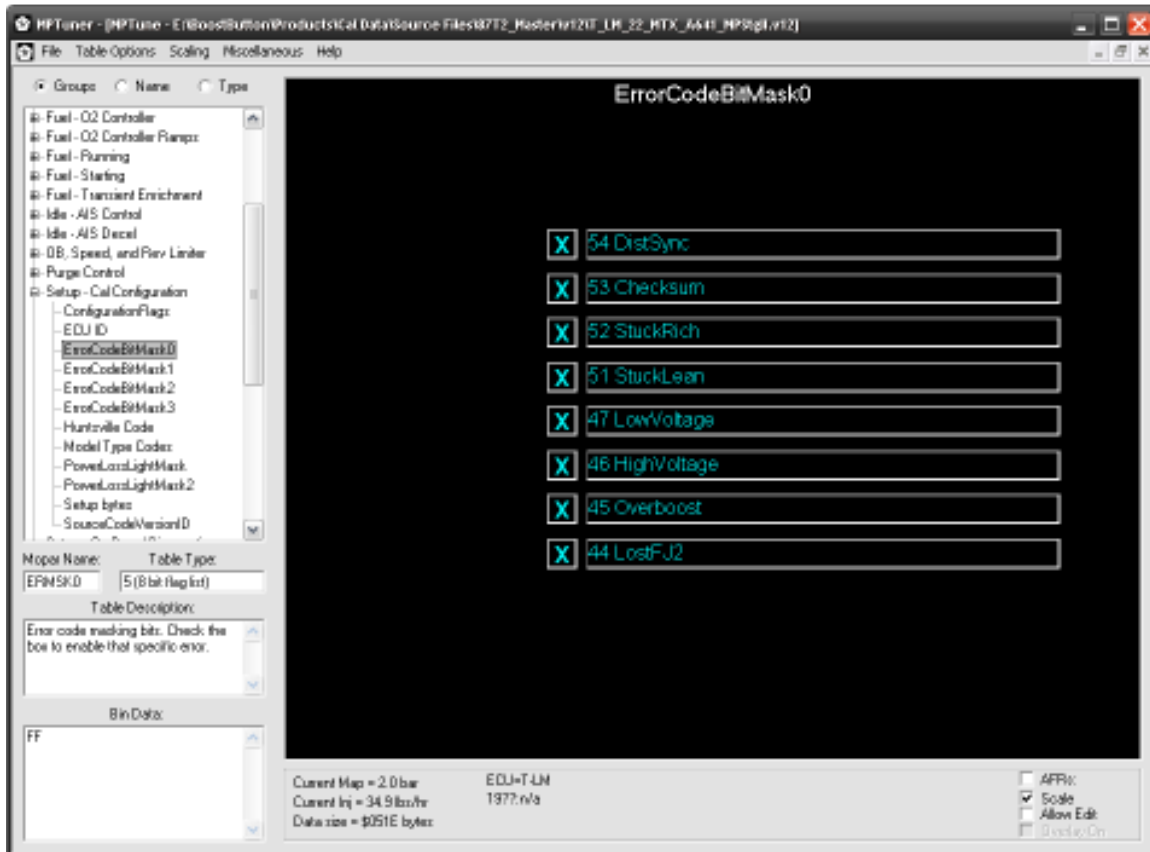
### **3.2.2 – VNT Solenoid Duty Cycle Lookup and Calculation (VNT boost Control – T-SBEC only)**

This section not finished yet.

## Section 10 – Error Code Masking

### 10.1.1 – T-LM

Error code masking in T-LM is done by unchecking the enable bit in 1 of the 4 error code masking bytes. In MP Tuner, these are located in the 'Setup – Cal Configuration' group. Find the error code you need to mask and simply remove the check mark next to it.



### 10.1.2 – T-SMEC and T-SBEC

Error code masking in T-SMEC/SBEC is a little more complicated. In some cases, there are actually more than 1 diagnostic routine that can result in a given error code. Also, Chrysler added error codes in the 61+ range. Chrysler has an internal masking code for each of these diagnostics. So, you have to mask each one individually. Unfortunately, Chrysler only gave 4 slots for these mask codes. T-SMEC and T-SBEC expand that to 8 slots. The error code masking slots are located in the 'Setup – Cal Configuration' group in MP Tuner.

Here is the key for error code to mask code:

<b>Code #</b>	<b>Description</b>	<b>Mask Code (HEX)</b>
63	Controller cannot write to EEPROM	2f
62	EMR mileage cannot be stored in EEPROM	30
61	EMR something else	31
54	Distributor Sync	01
53	Prom Checksum	02
52	Rich Condition	03,2d
51	Lean condition	04
47	Battery Volts Low	05
46	Battery Volts High	06
45	Nothing (overboost)	07
44	Loss of FJ2	08,2c
43	Ignition Coil Control	09,28,29,2b
42	Auto Shutdown Relay	0a
41	Charging System	0b
37	Barometric Read Solenoid	0c
36	Wastegate Solenoid	0d
35	Radiator Fan Relay	0e
34	Speed Ctl Solenoid	0f
33	AC Cutout Relay	10
32	Exh Gas Recirc Solenoid	11,2e
31	Purge Solenoid	12
27	Injector Driver interface	13,14,15
26	Injector Current Limit Not Achieved	16,17,18
25	AIS Motor	19
24	Throttle Position	1a,1b
23	Charge Temp Sensor	1c,1d
22	Coolant Temp Sensor	1e,1f
21	Oxygen Sensor	20
17	Engine Running Too Cool (thermostat)	21
16	Battery Voltage Sense	22
15	Vehicle Distance Sensor	23
14	MAP Electrical	24,25
13	MAP Pneumatic	26,27
12	Memory Standby Power Lost	00
11	Engine Not Cranked Since Battery Disconnected	28



## Notes and stuff I need to add in here somewhere...

I'd start by turning OFF the spark scatter idle control, forcing the AIS to control idle.

The primary means the Chrysler ECU uses to control idle is spark and fuel changes. So, it can be tough to get the idle fuel right while the spark scatter is running. I think all you really need to do is set IDLTMP and SSTEMP (in the 'Spark - Scatter Control' group) to FF to disable spark scatter. Though, you may want to copy over the complete 'Spark - Scatter Control' group from the 2.2 MP template. I'd start with IDLTMP - it controls the spark scatter fuel. Then try the SSTEMP to dis-able spark scatter completely.

The other effect is, this will likely keep your revs higher during shifts because the AIS will be wide open. So, you may also have to play around with the rolling AIS tables to bring the RPM down during shifts (if you need/want to).

The SSTEMP looks like it was already set to FF. That will dis-able the spark scatter. But, the IDLTMP is the one that adds fuel to control the idle speed. Now that that one is off, you should be able to adjust the pumpeff table effectively. After that, maybe try setting the SSTEMP to a value. If you set the scale on that table (double click on the table name to bring up the properties) to -200 to 260 deg F, you can set it to ~100F and see if that helps bring down the RPM between shifts (assuming you want it to).

Sounds like the idle adaptives are out of whack (or can't keep up). You need to dis-able them as well to see how lean you really are.

I think if you set 'MNRPO2\_MinimumRPMOverIdleForO2Feedback' to 0, it should completely dis-able to O2 feedback (forcing the system to go open-loop). I think this includes skipping the adaptives. That value should be at 0x8a25 and it will be set to 0xBC (188 decimal).

FuelNoThrottle sets the base PW for idle. There are many many factors that modify that value.

1. PumpEff - modifier based on engine RPM
2. Cold start curves A,B,C - based on coolant temp (A, and C are phased out with time, B is always active, but the factor is 1 at normal operating temp)
3. Spark Scatter fuel compensation - based on idle speed error and others
4. Adaptive fuel compensation

If you are leaning out as the engine gets warmer, it's likely to be due to the A,B,C Curves as the adaptives do not take effect until after the engine is warm and in closed loop. It could be that the A,B,C curves add too much fuel when cold, and/or that the FuelNoThrottle table or PumpEff table don't have enough fuel for that engine operating point (RPM/Vac).

Fuel baseline and fuel no throttle are the same thing. There's a fuel flow chart posted in the EFI section that will walk you thru how the main fuel tables are calculated and used.

The AFR line only shows up on those 3 tables because those are the only 3 tables that have the AFR built into it.

If you want to adjust the idle fuel, there are a few ways to do it.

- 1) Reduce the PumpEff at low RPM. For some reason, Chrysler left the PumpEff at like 60+% even at idle. When, in reality, this is probably not very realistic. So, reduce the PumpEff in maybe 5% increments below 1200rpm and see if that helps.
- 2) Reduce the baseline/no throttle fuel table near idle vacuum. If you have a different cam or cam timing than stock, it's likely that you're operating in a different part of this table than stock and are calculating a richer mixture.
- 3) Spark scatter fuel - Chrysler actually uses spark scatter and subtle fuel changes to control the idle speed primarily. The AIS motor is actually a last resort. This is because fuel and spark changes are much more responsive than the AIS motor. But, the constant changing spark and fuel make it hard to tune. There's another thread here about turning them off for tuning purposes.
- 4) Then, there is the adaptive fuel which also works at idle. You might need to adjust those kick values. You can also dis-able the adaptives for tuning purposes...

## Ignition Stuff

No, the total timing is referenced to TDC, not the base timing. So, 20deg should be 20deg, not 32.

But, if you set the distributor to 14deg, and the scan tool then reads 20, that's actually 22deg...

The 12 deg base timing doesn't get 'added' into the calculated timing. There's some math done to determine when to turn on/off the coil. It includes the dwell time, and an offset for the position of the rotor relative to TDC. I don't see how ChryCo got the offset, yet. But, I know that's what it is. I think it also must include some offset value for the width of the distributor rotor. I need to comb thru the patents again. I think there's probably something in there to explain it..

The timing the ECU calculates is referenced to TDC. So, it is actual advance. But, the ECU assumes the timing reference was set using the procedure, and that it was set to 12 degrees. So, it sets the coil on/off times based on the rising and falling edges of the shutter wheel. If you move the shutter wheel, then it's true, the computer no longer knows where it really is since it is all setup assuming 12 degrees. But, again, the 12 deg is not added to or subtracted from the stored timing value at all. It's hard-coded in the coil on/off timing calculation.