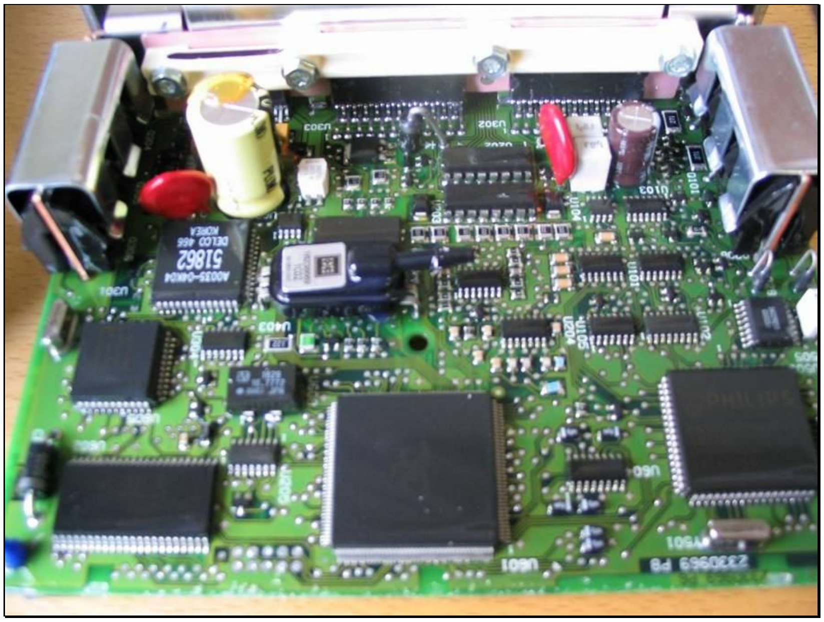
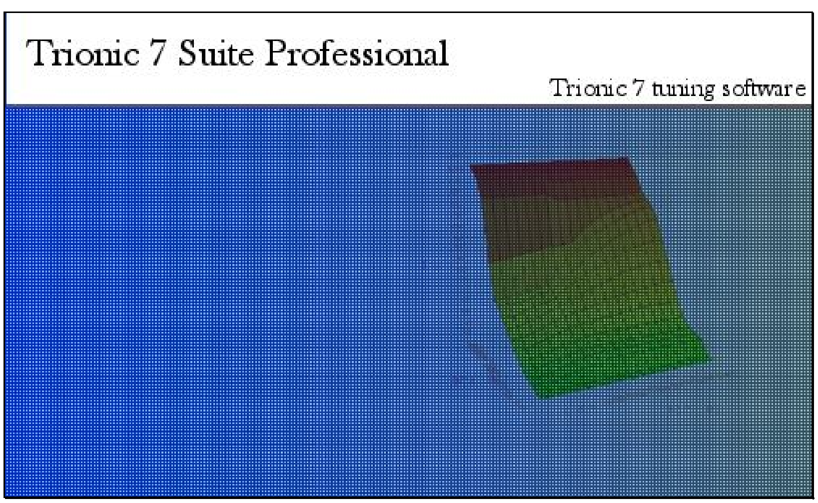
Trionic 7





Preface

This document is intended for Saab fanatics and engineers who want to start understanding the Saab Trionic 7 motor management system. It will give as much information as possible about the technical part of the system. The only limitation will be the knowledge of the author.

In short the content of this document will enable you to understand Trionic 7 better and give you hands-on information about altering the maps it uses. Prerequisites are minor electronics and computer knowledge and of course some understanding of how a turbo charged engine works.

Throughout the document the T7Suite software will be referenced. This software will enable you to really “get into” the Trionic. The T7Suite software can be downloaded from the T7Suite website.

http://trionic.mobixs.eu

Acknowledgements

The author would like to thank everyone on ecuproject for their help on getting all this information together. Special thanks go out to General Failure, J.K. Nilsson, Hook, Hma, Vigge, Mackan, Sandy Rus, JKB, L4staero and Steve Hayes.

These icons are used throughout the document to denote:

References

Advanced technical topics

Table of contents

Hardware

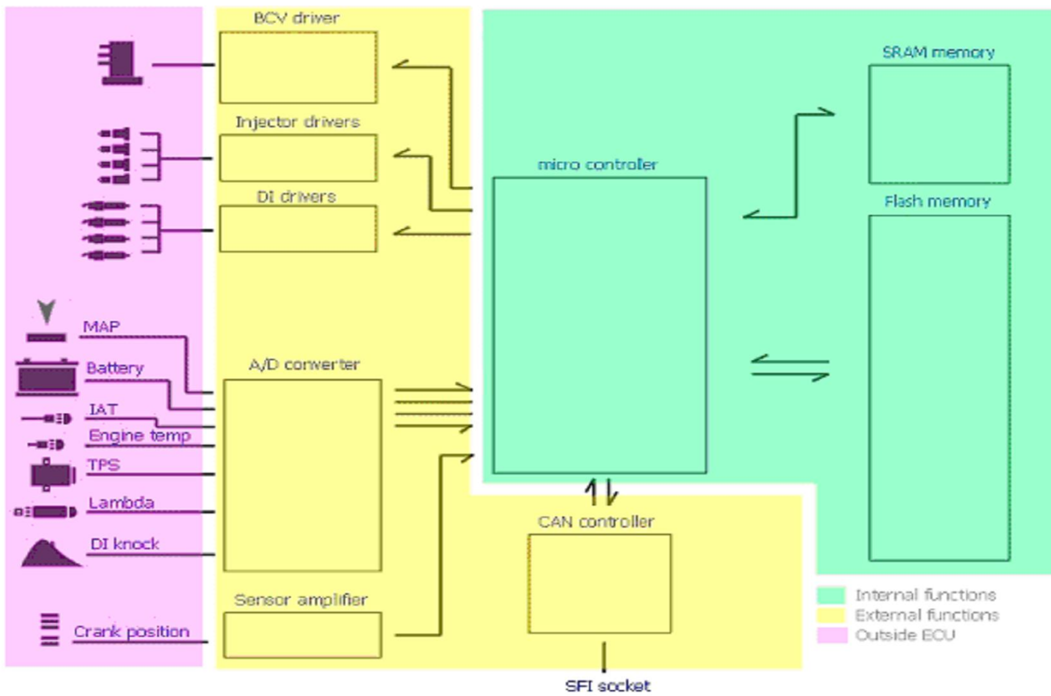
The T7 is build around a Motorola MC68332 (CPU32) microcontroller. This is a 32 bit controller that handles the entire motor management including fuel injection, ignition timing and boost pressure control.
The processor has a vast 4Mb (512 Kbyte) flash memory to its disposition for fetching program code and maps. This flash memory consist of a AM29F400BT-90SI (AMD) holds the program memory.
There is a coprocessor from Philips, a P83C592FHA/019. This is a 8-bit 8051-based microcontroller with a CAN (Controller Area Network) module. As the CAN physical line driver there is an Intel AN82527 (same family as used in Trionic 5). RAM memory is done by two 32 Kbit SRAM chips (U62H256S1K). There is also a special component that would appear to be a barometric pressure sensor.

Integrated circuit list

The table below lists almost all IC's on the board. This is just to give you an idea on what to expect.

Partnumber	Function	Usage	# on board
TC55257DFI-85L	SRAM (working memory)		1
16233970	Microcontroller	Main 32-bit CPU	1
PC83C592	Microcontroller with CAN contr.	8-bit coprocessor	1
AM29F400	Flash memory	4 Mbit	1
51862	DA converter		1
AN82527	CAN controller	CAN line driver	1
16238669 0H11	Pressure sensor?		1

Block schematic diagram



PCB details

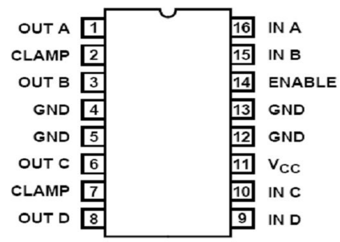
The PCB layout is not entirely known of course because SAAB did not release details about this, even in the service manuals. Finding out how things are setup is not so very difficult though, once you know what the system should do and what hardware components are on the board.
The image will give you some idea on what is what on the board.

Power supply

DI cartridge triggering

The DI cartridge has a trigger input for firing the four individual sparkplugs. These are triggered by signals from the ECU on pin 9, 10, 11 and 12 which are generated in the power driver IC CA3236 on the Trionic PCB (topside, 16 pin DIL housing). Internally these four pins are connected as show in the table and the image below.

DI cartridge pin	ECU pinnumber	CA3236 pinnumber	Description
2	7	1 (OUT A)	Trigger cylinder 1
3	8	3 (OUT B)	Trigger cylinder 2
4	67	6 (OUT C)	Trigger cylinder 3
5	68	8 (OUT D)	Trigger cylinder 4



BPC drivers (MTP3055V)

Injector drivers (MTD3055VL)

Crankshaft position sensor (LM1815)

Flash

Downloading with PEMicro USB BDM interface

Checksum

Preface

The Trionic 7 ECU binary images uses several checksums to verify integrity. Most of them have been easy to figure out, but one of them is so complicated, that it seems to been done to deter map changing. There is still some unknowns, that would be nice to figure out. For example some binaries don't seem to have all four checksums I've discovered. And of course there could be more checksums that have gone unnoticed. Two of the checksums are in the end of the binary and the two other ones are scattered in the code. The latter ones can be found be using pattern searching. Again the calculations are pretty simple, and even the harder checksum is easy to implement. Big thanks to solving these things goes to Tomi and General Failure.

Checksum lexicon

First of all, there are four different checksums. They have been given names by Tomi: FB checksum, F2 checksum, Misc checksum and Area 70000 checksum.

F2 and FB checksums

The first two checksums, FB and F2, can be found at the end of the binary. This end area has been called the file header (footer would be more logical). *See also chapter Trionic 7 file header*. The F2 checksum is not present in all binaries, so be aware of this. Finding the two other checksums is more of a challenge.

Misc checksum

The Misc checksum resides inline with the code. It is usually found in the area of 0x02000...0x05000. The checksum address can be found by pattern searching the bin file using a set of hex values along with mask bits. If a mask bit is not set, the corresponding hex value does not have to match. Here is the hex values, and the masks.

Pattern

0x48,0xE7,0x00,0x3C,0x24,0x7C,0x00,0xF0,0x00,0x00,0x26,0x7C,0x00,0x00,0x00,0x00,0x28,0x7C,0x00,0xF0,0x00,0x00,0x2A,0x7C

Mask

1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1

So, we are searching the binary for a string of bytes beginning with 0x48, 0xE7, 0x00, 0x3C... Then we mask out the bytes that change from binary to binary. When we've located this pattern, we know where to start. Now we start primitively disassembling the code. we search for byte patterns [0x48,0x6D], [0x48,0x78], [0x48,0x79], [0x2A,0x7C] and [0xB0,0xB9]. The three first patterns reveal addresses and lengths of checksum areas. There are 15 checksums areas from which the Misc checksum is calculated. The [0x2A,0x7C] pattern gives a base address for the [0x48,0x6D] addresses. Bare with me. These [0x2A,0x7C] addresses are summed with the base address to make the actual address. This way the address is only 2 bytes long. On the [0x48,0x79] addresses it's 4 bytes long without any base address. And the [0x48,0x78] pattern gives 2 bytes which correspond with the length of the checksum area. Finally the [0xB0,0xB9] pattern is followed by 4 bytes to the address of the Misc checksum.

Area 70000 checksum

This checksum refers to an area in the region of 0x70000. Like the Misc checksum, there is no clean way of finding out the length of the area along with the checksum address. Using pattern searching with this also has results. There are binaries that are incompatible with this approach, so this requires some fixing in the future.

Pattern

0x20,0x3c,0x00,0x00,0x11,0x52,0x2F,0x00,0x20,0x3C,0x00,0x00,0x09,0xD0,0x2F,0x00,0x20,0x3C,0x00,0x00,0x00,0xCC,0xD0,0x9F

Mask

1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1

After finding that pattern, the masked addresses are summed together. In the pattern, the original addresses are 0x00001152, 0x000009D0 and 0x000000CC. Summing these gives the Area 70000 length of 0x1BEE. At the same time this is the address where to find the Area 70000 checksum.

How to calculate a checksum

The FB checksum shares the same calculation method as Misc and Area 70000 checksums. It's simply a sum of the bytes from the checksum area. Four bytes are made into a 32 bit value and summed with the next 32-bit value. This goes on until there are fewer than 4 bytes left. The last 1...3 bytes are then individually summed together with the checksum.

Misc checksum

The Misc checksum is a sum of the individual checksums calculated from 15 areas. The used checksum calculation the same as with the FB checksum.

Area 70000 checksum

Once you have found the length of the Area 70000, you can calculate the checksum by using the function described in section "FB checksum". The start address is 0x70000. Notice that your binary might not have this area present.

Firmware

General

Once you are done with dumping the flash contents and you want to do more than only alter variables and maps you can start analyzing the binary. This is a difficult task because there are a lot of different firmware versions, stock ones – maybe different per MY and tuned ones that differ for every manufacturer and stage. In every case the code can be disassembled using a 6833x disassembler like the one in IDAPro. There are scripts available to automatically disassemble the code and make it more readable by replacing addresses by variable names that are extracted from the symboltable inside the binary.

Memory map

Disassembling the code

Symbol tables

General

Each T7 firmware file contains a symbol table describing data structures in the program. The major problem is that from some point in time SAAB started to compress the symbol tables in the binary file. Probably just to save space in the flash memory but it has made tuning a little harder. We actually need these symbol names because they tell us what a certain memory location means. For unpacked binaries these symbols can be extracted together with their corresponding memory addresses (ROM and RAM).

Image 4 gives a general idea on what these symbol tables look like.

```
000015d0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ; ................
000015e0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ; ................
000015f0h: 00 00 00 00 00 00 00 00 00 00 42 6C 6F 63 6B 54 ; ..........BlockT
00001600h: 79 70 65 00 42 6C 6F 63 6B 2E 54 69 6D 65 72 00 ; ype.Block.Timer.
00001610h: 42 6C 6F 63 6B 2E 54 69 6D 65 72 31 00 FF 42 6C ; Block.Timer1.ÿBl
00001620h: 6F 63 6B 2E 54 69 6D 65 72 32 00 FF 42 6C 6F 63 ; ock.Timer2.ÿBloc
00001630h: 6B 2E 41 44 5F 54 68 72 6F 74 74 6C 65 44 65 6D ; k.AD_ThrottleDem
00001640h: 61 6E 64 00 42 6C 6F 63 6B 2E 41 44 5F 54 68 72 ; and.Block.AD_Thr
00001650h: 6F 74 74 6C 65 53 75 6D 00 FF 42 6C 6F 63 6B 2E ; ottleSum.ÿBlock.
00001660h: 4E 65 77 00 42 6C 6F 63 6B 2E 4D 69 6E 00 42 6C ; New.Block.Min.Bl
00001670h: 6F 63 6B 2E 4D 61 78 00 42 6C 6F 63 6B 2E 6D 73 ; ock.Max.Block.ms
00001680h: 5F 43 6F 75 6E 74 65 72 00 FF 42 6C 6F 63 6B 2E ; _Counter.ÿBlock.
00001690h: 41 44 5F 54 6D 70 54 68 72 42 6C 6F 63 6B 00 FF ; AD_TmpThrBlock.ÿ
000016a0h: 42 6C 6F 63 6B 32 54 79 70 65 00 FF 42 6C 6F 63 ; Block2Type.ÿBloc
000016b0h: 6B 32 2E 46 43 4F 41 63 74 69 76 65 00 FF 42 6C ; k2.FCOActive.ÿBl
000016c0h: 6F 63 6B 32 2E 54 68 72 6F 74 74 6C 65 4F 4E 00 ; ock2.ThrottleON.
000016d0h: 42 6C 6F 63 6B 50 6F 74 69 32 54 79 70 65 00 FF ; BlockPoti2Type.ÿ
```

Image 1: Symbol table in T7 firmware

While examining the symbol table you can see that the separator is 0x00. In contrast to T5 where symbol and SRAM addresses reside in the same table, we now only find the symbol name. In another table in the binary we can find flash addresses and lengths in the same sequence as the symboltable.

Finding start of symbol table:
Search binary for string the first sequence of 15 zeros.

Finding start of address lookup table:
Search binary for 20 00 00 00 XX YY 00 F0 where XX YY is the index of the first symbol found in the symbollist.

To save you the time to lookup all addresses manually the T7Suite application will extract all symbol information in one run. Symbol name, flash address and length will be displayed all together.

This *image (6)* will give you an idea of what the symbol table should look like once it has been extracted. See appendix I for a complete list of known symbols.

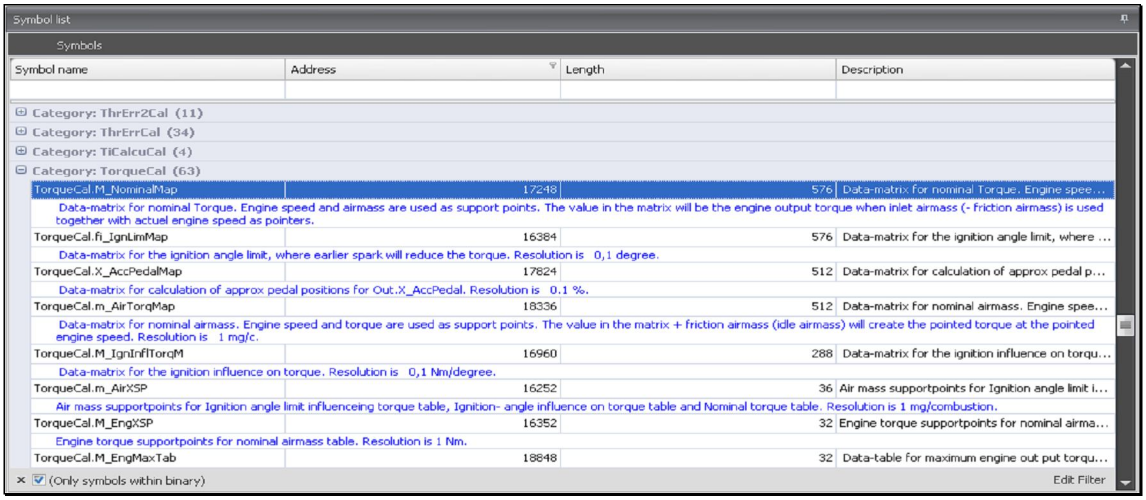
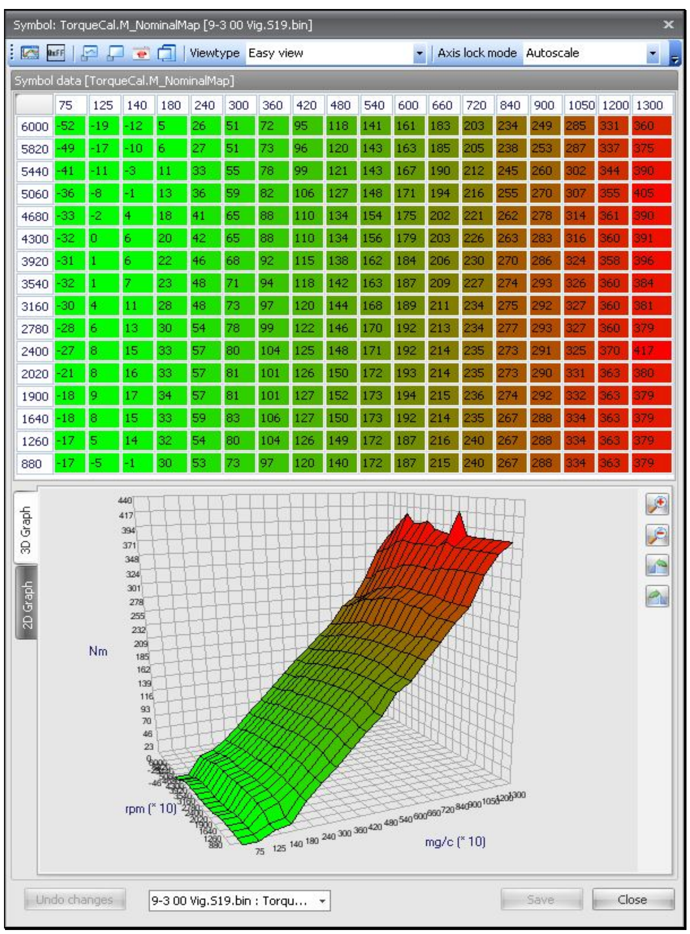


Image 2: Screenshot of a part of the symbol table

When the user double clicks one of the symbols that has a flash address attached to it, T7Suite will display the corresponding symbol in a viewer. This viewer will display the data in table form was well as in graphical form.



Maps

General

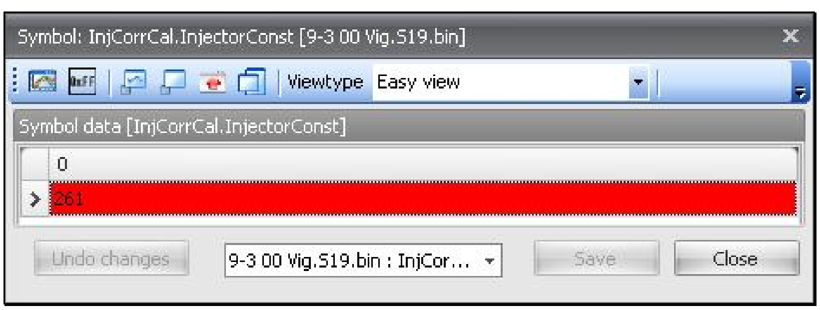
A lot of maps in the T7 are not only made up of a piece of raw data. It also includes x-axis and y-axis information. T7Suite will automatically display all known axis information when a map is opened. IN Trionic 7 most symbol have an English name (Trionic 5 has lots of Swedish names) that explains lots about its function. Also, the symbols are categorized by name, which makes browsing the symbols much easier. All torque calibration symbols start with "TorqueCal.". T7Suite groups all symbols by their respective category by default.

Fuel

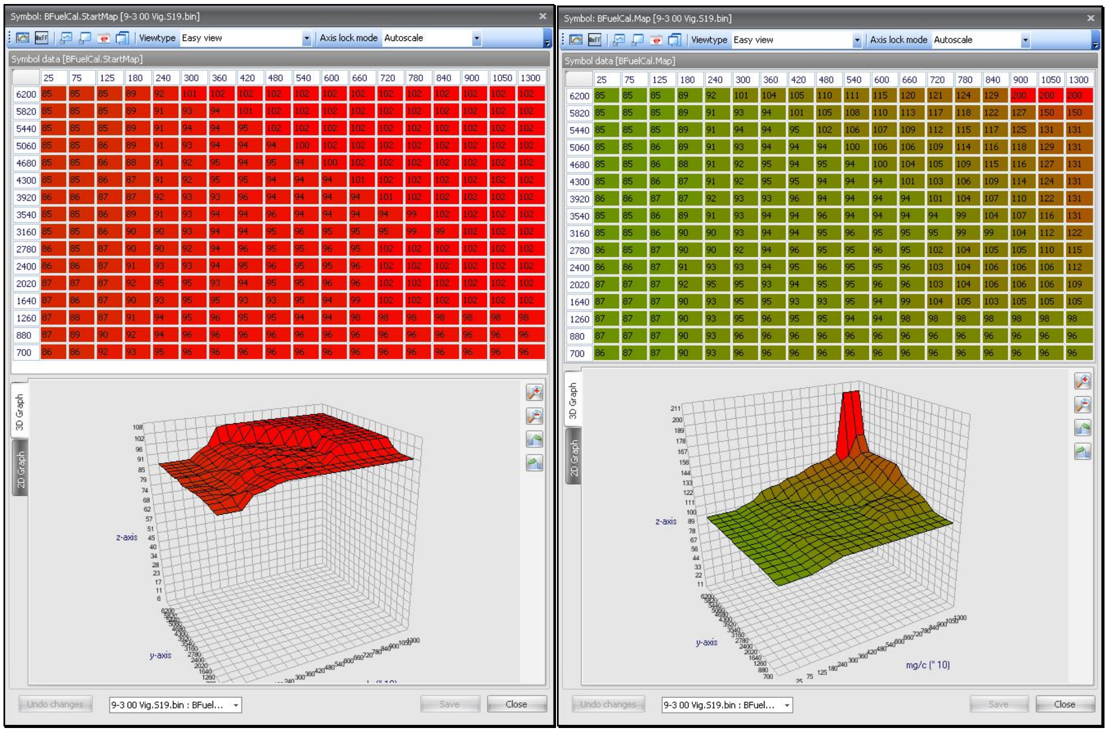
Fuel calculation in Trionic 7 is based on the Airmass entering the engine. In rough steps this seems to be the calculation's flow:

	Description	Explaination
1	Basic calculation of fuel quantity per combustion	The current air mass/combustion is divided by 14.7 and sent to box 2. The unit is now in mg fuel/combustion
2	Compensation	In case of a cold engine, shortly after starting, rapid load changes, knocking or high loads, the current value is multiplied by a compensation factor
3	Closed loop	The closed loop value is used as a multiplier. The value is then sent to box 4
4	Correction for purge	Multiply by the value for purge adaptation. The value is sent to box 5
5	Multiplicative adaptation (long term fuel trim)	The multiplicative adaptation value is used as a multiplier and the new value is sent to box 6
6	Additive adaptation	The additive adaptation value is added and the new value is sent to box 7
7	Starting fuel quantity	If the engine has not yet started, starting fuel is selected. The value is sent to box 8
8	Fuel quantity per combustion to be injected	The fuel quantity per combustion is the amount of petrol to be supplied to the engine. The value is sent to box 9
9	Injector opening duration	Converts the value to the time during which the injector must be open and the new value is sent to box 10
10	Injection twice per combustion	Injection takes place twice per combustion until the camshaft position has been found. Injection duration is divided by two. The value is sent to box 11
11	Voltage dependant needle lift duration added (battery correction)	Adds the injector time delay, which is voltage dependant. The value is sent to box 12
12	Fuel cut	The value is sent to box 13 unless fuel cut is active
13	Activation of injector	At a DETERMINED crank shaft angle, the microprocessor will control the transistor for the injector that is next in the firing order

The basic fuel quantity is calculated based on Airmass and Injector constant. This injector constant is called InjCorrCal.InjectorConstant.



If the engine is not warmed up yet, an alternate fuel map is used called BFuelCal.StartMap. If the engine has reached operating temperature the normal map "BfuelCal.Map" is used.



You will see the areas calibrated to be run in closed loop have a value of around 1.00. It can be sort of .98-1.02 or so. Then you will notice the high load part of the maps ramp up to enrich the mixture.

The trick is to set the closed loop part of the map first, the areas that are represented by values of 1.00.
You will need to switch closed loop off and drive around with a wideband in the tailpipe.

What you want to achieve is an AFR of 14.7 for petrol while driving around under light loads where the value is 1.00. What this does is T7 calculates the injection time to be say 5ms, if that is not correct it is multiplied by this map. Say you need to enter a value of 1.1 in the map to get correct AFR it will change injection time to 5.5ms (5ms*1.1=5.5ms)

The idea is to get closed loop area correct first to stop any negative adaption once the tune is finished and allowed to run in closed loop again. If your closed loop areas of your fuel map is too rich it will negatively adapt over a long period of time. This will can have the effect of leaning your AFR's across the board.

Example: you make a tune, closed loop AFR's are fine (because the O2 is making it fine through feedback) but unknown to you its rich and short term fuel trim is driving negatively 13%. I.e. is leaning off injection time by 13%.
You don't notice this and make some full power runs to check AFR its fine at say 12.5 AFR.
After several weeks the multiplicative adaptation (Long term fuel trim) has absorbed some adaption and has earned a value of -13% this will now subtract 13% from whole fuel calculation including full power. All of a sudden your full power AFR has jumped up to 14.0 AFR, engine failure happens very easily from here.

Now back to fuel mapping. To set closed loop area of fuel map monitor the AFR in this light load area if its wrong after adjusting for large injectors start by just adjusting the Injector constant up and down accordingly instead of altering the closed loop area of the BfuelCalMap. This will affect the whole map

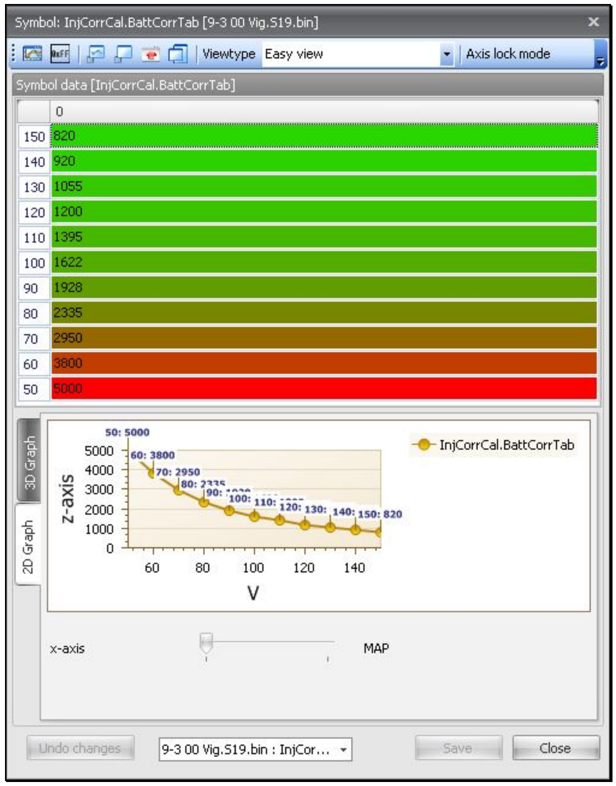
instead of one point. By doing it this way you can almost get AFR spot on in closed loop area just by a few goes at adjusting the injector constant.
It can be found in InjCorrCal.InjectorConst, the value represents the injectors flow in mg of fuel (not capacity or cc's as injectors are normally rated in). The injector constant is a calculation factor used by T7 to calculate injection time.

Once closed loop area is done the high load area can be mapped. If its too lean just increase the values in the relative column relating to what site of the map you are running in. Once your high load areas are done, activate closed loop again so you can see how it all runs. Monitor fuel adaptations and AFR etc.

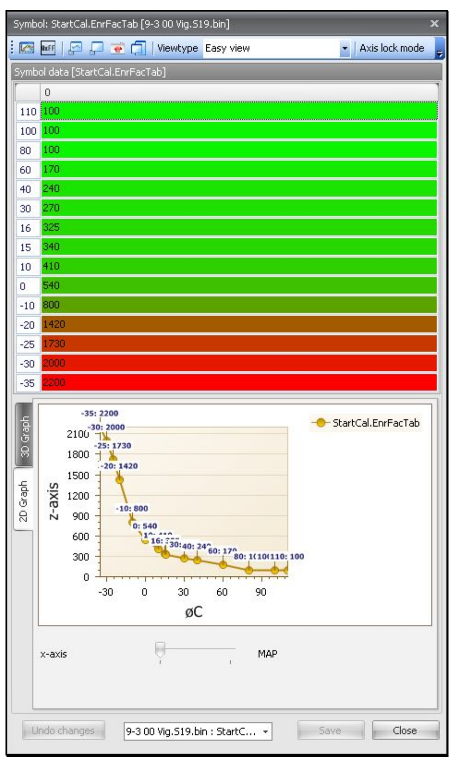
One thing to note is how quickly it drops into open loop under full throttle.
By going into open loop the o2 sensor is "masked" where the ECU listens to what its saying but ignores it, this allows afr's to go beyond 14.7 and injection correction is directly taken from the fuel map we just adjusted allowing much enrichment to cool charge etc.

To alter open loop enrichment you can change at what Airmass point does it change to open loop. This map is called LambdaCal.MaxLoadNormTab. Also, open loop entry (so, leaving closed loop situation) has a delay attached to it. This way, short overruns of the maximum load will not immediately result in leaving closed loop. Stock bins often have this set to 2000 milliseconds which seems quite long. If you want to ECU to leave closed loop faster after overrunning the load limit, just decrease the time in LambdaCal.TimeOpenLoop.

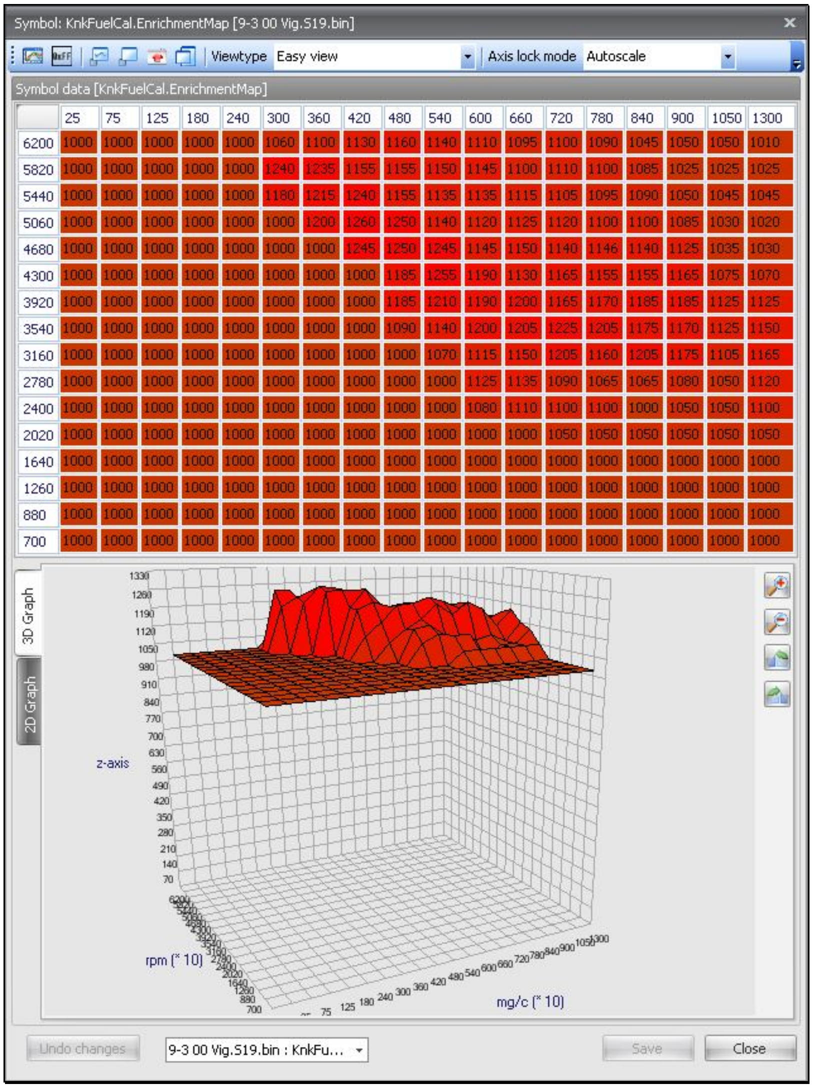
Battery correction values for injector latency



Water temperature correction



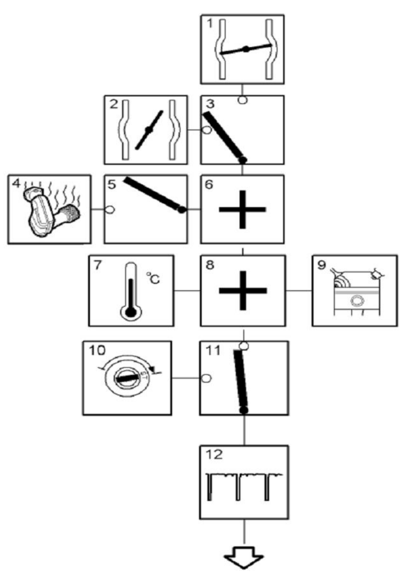
Fuel injection correction map for knock conditions



	25	75	125	180	240	300	360	420	480	540	600	660	720	780	840	900	1050	1300
6200	1000	1000	1000	1000	1000	1060	1100	1130	1160	1140	1110	1095	1100	1090	1045	1050	1050	1010
5820	1000	1000	1000	1000	1000	1240	1235	1155	1155	1150	1145	1100	1110	1100	1085	1025	1025	1025
5440	1000	1000	1000	1000	1000	1180	1215	1240	1155	1135	1135	1115	1105	1095	1090	1050	1045	1045
5060	1000	1000	1000	1000	1000	1000	1200	1260	1250	1140	1120	1125	1120	1100	1100	1085	1030	1020
4680	1000	1000	1000	1000	1000	1000	1000	1245	1250	1245	1145	1150	1140	1146	1140	1125	1035	1030
4300	1000	1000	1000	1000	1000	1000	1000	1000	1185	1255	1190	1130	1165	1155	1155	1165	1075	1070
3920	1000	1000	1000	1000	1000	1000	1000	1000	1185	1210	1190	1200	1165	1170	1185	1185	1125	1125
3540	1000	1000	1000	1000	1000	1000	1000	1000	1090	1140	1200	1205	1225	1205	1175	1170	1125	1150
3160	1000	1000	1000	1000	1000	1000	1000	1000	1000	1070	1115	1150	1205	1160	1205	1175	1105	1165
2780	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1125	1135	1090	1065	1065	1080	1050	1120
2400	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1080	1110	1100	1100	1000	1050	1050	1100
2020	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1050	1050	1050	1050	1050	1050
1640	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1260	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
880	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
700	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000

Ignition

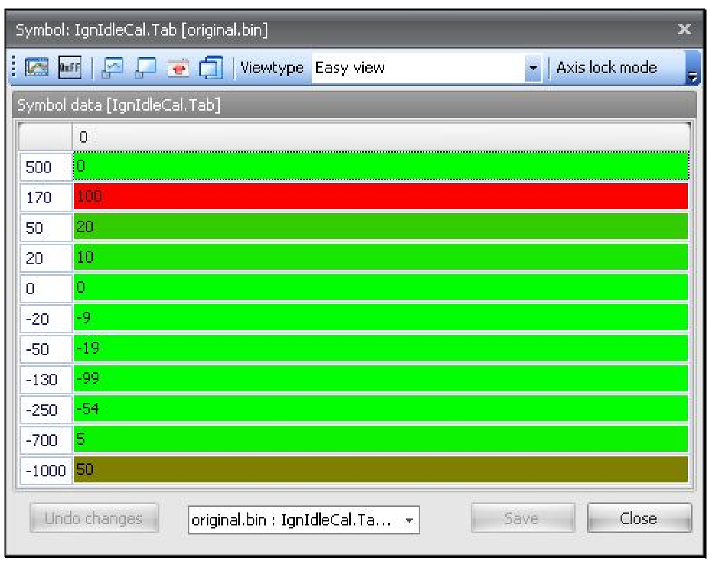
	Description	Explanation
1	Idling speed ignition timing	With idle speed control active, the timing is adjusted to stabilize idle engine speed. The value is sent to box 3
2	Normal ignition timing	When idle speed control is inactive, the ignition timing is read from a load and engine speed depending matrix. The value from the matrix is optimized for lowest fuel consumption (best engine torque) and sent to box 3
3	Selection of ignition timing	One of the ignition timing calculation is selected depending on which function is active. The value is sent to box 6
4	Catalytic converter heating timing	In order to heat up the catalytic converter as fast as possible after start, the ignition will be retarded. This is a compensation matrix that is added to the value in box 3. The matrix is dependent on load and engine speed
5	Engagement of catalytic converter heating timing	The function is active when coolant temperature is above -10 degrees Celsius and below +64 degrees Celsius
6	Total	The value from box 5 is added to the value of box 3
7	Compensation	The ignition timing is corrected depending on engine coolant temperature and intake air temperature. The value is sent to box 6.
8	Knock control	If knocking occurs, a timing retardation will be calculated. The value is sent to box 6
9	Total	The compensation angle and knock retardation are totalled to give the current ignition timing. The value is sent to box 7
10	Selection of ignition timing	Starting ignition timing is selected when the engine has not been started. The value is sent to box 9
11	Starting ignition timing	Starting ignition timing is selected when the engine has not yet been started. The value is sent to box 9
12	Activate relevant trigger	At the calculated crankshaft angle, the microprocessor controls the transistor for the trigger that is next in firing order



Ignition cassette

The ignition cassette is mounted on the valve cover on top of the spark plugs. The ignition cassette houses four ignition coils/transformers whose secondary coil is direct connected to the spark plugs. The ignition cassette is electrically supplied with battery voltage from the main relay (B+) and is grounded in an earth point. When the main relay is activated the battery voltage is transformed to 400 V DC which is stored in a capacitor. The 400 V voltage is connected to one of the poles of the primary coil in the four spark coils. To the ignition cassette there are four triggering lines connected from the Trionic ECU, pin 9 (cyl. 1), pin 10 (cyl. 2), pin 11 (cyl. 3) and pin 12 (cyl. 4). When the ECU is grounding pin 9, the primary coil for the first cylinder is grounded (via the ignition cassettes B+ intake) and 400 V is transformed up to a maximum of 40 kV in the secondary coil for cyl. 1. The same procedure is used for controlling the ignition on the rest of the cylinders.

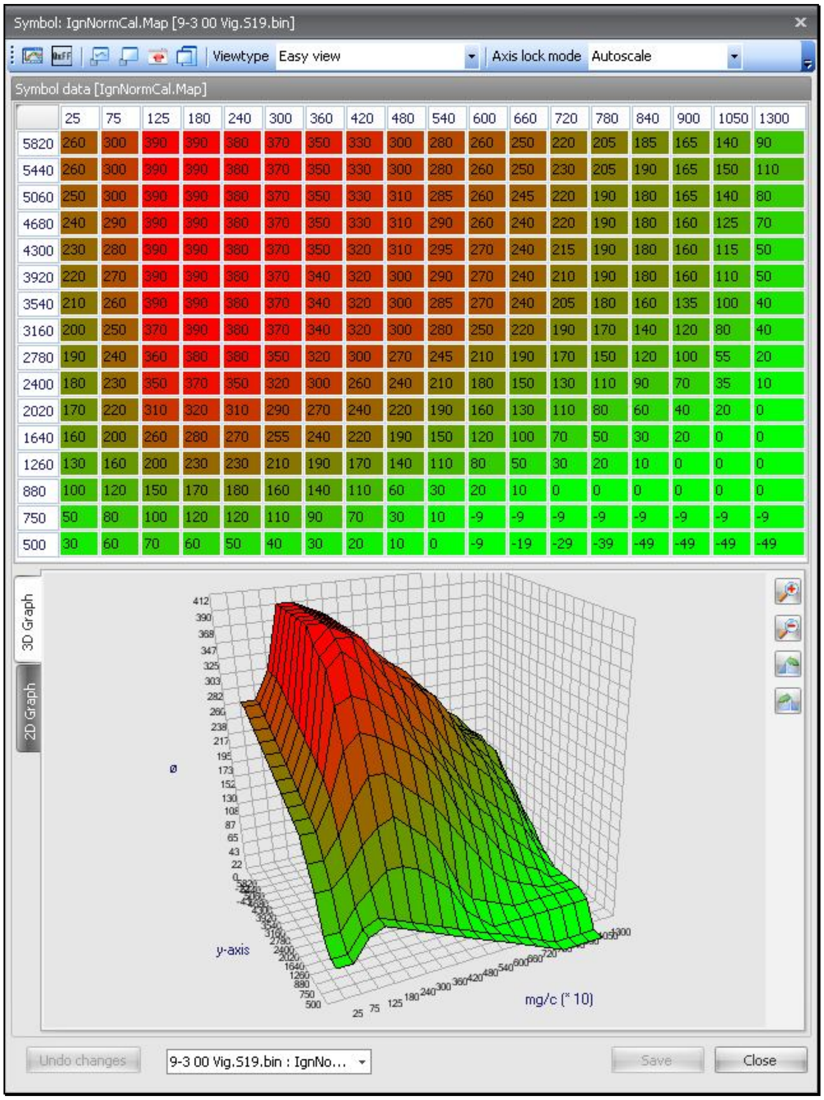
Idle control



	0
500	0
170	100
50	20
20	10
0	0
-20	-9
-50	-19
-130	-99
-250	-54
-700	5
-1000	50

Normal mode ignition

Ignition is normally controlled by the main ignition matrix: IgnNormCal.Map



	25	75	125	180	240	300	360	420	480	540	600	660	720	780	840	900	1050	1300
5820	260	300	390	390	380	370	350	330	300	280	260	250	220	205	185	165	140	90
5440	260	300	390	390	380	370	350	330	300	280	260	250	230	205	190	165	150	110
5060	250	300	390	390	380	370	350	330	310	285	260	245	220	190	180	165	140	80
4680	240	290	390	390	380	370	350	330	310	290	260	240	220	190	180	160	125	70
4300	230	280	390	390	380	370	350	320	310	295	270	240	215	190	180	160	115	50
3920	220	270	390	390	380	370	340	320	300	290	270	240	210	190	180	160	110	50
3540	210	260	390	390	380	370	340	320	300	285	270	240	205	180	160	135	100	40
3160	200	250	370	390	380	370	340	320	300	280	250	220	190	170	140	120	80	40
2780	190	240	360	380	380	350	320	300	270	245	210	190	170	150	120	100	55	20
2400	180	230	350	370	350	320	300	260	240	210	180	150	130	110	90	70	35	10
2020	170	220	310	320	310	290	270	240	220	190	160	130	110	80	60	40	20	0
1640	160	200	260	280	270	255	240	220	190	150	120	100	70	50	30	20	0	0
1260	130	160	200	230	230	210	190	170	140	110	80	50	30	20	10	0	0	0
880	100	120	150	170	180	160	140	110	60	30	20	10	0	0	0	0	0	0
750	50	80	100	120	120	110	90	70	30	10	-9	-9	-9	-9	-9	-9	-9	-9
500	30	60	70	60	50	40	30	20	10	0	-9	-19	-29	-39	-49	-49	-49	-49

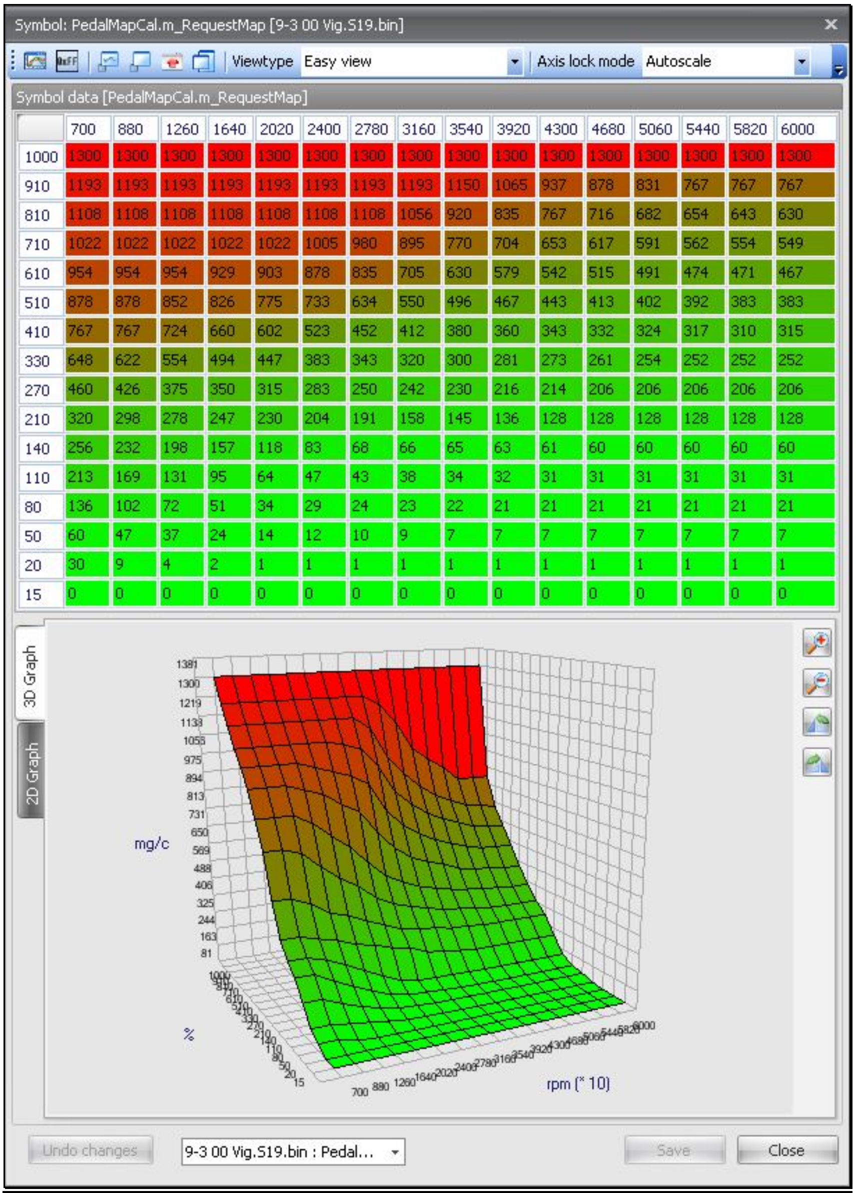
Torque

Trionic 7 is a torque/Airmass request system instead of a boost request system like Trionic 5 is.
The basic procedure for the Airmass controller is like in the table below.

	Description	Explaination
1	Driver request	The control module reads pedal potentiometer 1 and converts the voltage to Airmass per combustion (mg/c). The value is sent to box 3
2	Cruise control request	When cruise control is active, the air mass per combustion required to maintain the set speed is calculated. The value is sent to box 3
3	Select highest value	The control module selects the highest of the two values (box 1 or box 2). The value is sent to box 5
4	Engine torque limitation	The maximum permissible air mass per combustion varies depending on the engine type. During operation, the maximum permissible mg/c must also be limited to protect the engine, gearbox, brakes and turbo
5	Select lowest value	The control module selects the lowest value and sends it to box 8
6	Compensation request	When the AC compressor is on, and when the heated rear window or radiator fan is on, the mg/c required to compensate for the increased load is calculated. The value is sent to box 8
7	Other air request	The control module calculates the mg/c required for idle speed control. The value is sent to box 8
8	Totalling values	The control module totals all the values. The total is sent to box 9
9	Total requested mg/c	
10	Total Airmass request	
11	Throttle control	The requested mg/c is converted to requested voltage for throttle position sensor 1. The charge air pressure and intake air temp are used to correct this conversion. The throttle motor rotates the throttle until the current voltage for throttle position sensor 1 corresponds with the requested voltage
12	Current mg/c	The requested mg/c is also compared with the current mg/c (MAF reading). If needed the requested voltage for throttle position sensor 1 is finely adjusted
13	Turbo control	If mg/c is too high for throttle alone the turbo control will take over. The excess is converted to a PWM which controls the charge air control valve. The absolute pressure sensor is used to correct the conversion
14	Current mg/c	The requested mg/c is compared to current mg/c and the charge air control vale PWM is finely adjusted if required

Torque request

So, if the driver (or cruise control for that matter) pressed the accelerator pedal he actually requests a certain Airmass from the system. This value is fetched from the PedelMapCal.m_RequestMap shown below.



	700	880	1260	1640	2020	2400	2780	3160	3540	3920	4300	4680	5060	5440	5820	6000
1000	1300	1300	1300	1300	1300	1300	1300	1300	1300	1300	1300	1300	1300	1300	1300	1300
910	1193	1193	1193	1193	1193	1193	1193	1193	1150	1065	937	878	831	767	767	767
810	1108	1108	1108	1108	1108	1108	1108	1056	920	835	767	716	682	654	643	630
710	1022	1022	1022	1022	1022	1005	980	895	770	704	653	617	591	562	554	549
610	954	954	954	929	903	878	835	705	630	579	542	515	491	474	471	467
510	878	878	852	826	775	733	634	550	496	467	443	413	402	392	383	383
410	767	767	724	660	602	523	452	412	380	360	343	332	324	317	310	315
330	648	622	554	494	447	383	343	320	300	281	273	261	254	252	252	252
270	460	426	375	350	315	283	250	242	230	216	214	206	206	206	206	206
210	320	298	278	247	230	204	191	158	145	136	128	128	128	128	128	128
140	256	232	198	157	118	83	68	66	65	63	61	60	60	60	60	60
110	213	169	131	95	64	47	43	38	34	32	31	31	31	31	31	31
80	136	102	72	51	34	29	24	23	22	21	21	21	21	21	21	21
50	60	47	37	24	14	12	10	9	7	7	7	7	7	7	7	7
20	30	9	4	2	1	1	1	1	1	1	1	1	1	1	1	1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

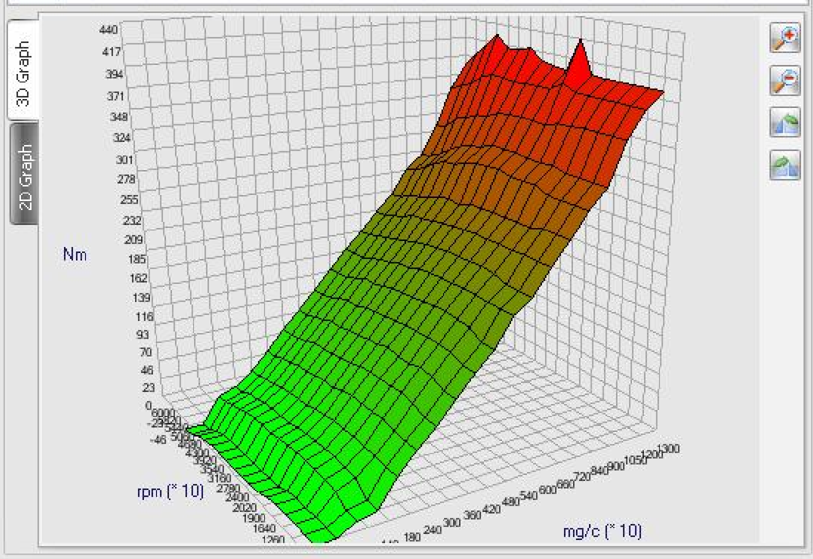
The table holds Airmass values for each position of the accelerator pedal and each rpm site. Trionic now looks up the estimated engine output (torque) based on Airmass and rpm. This is done through map "TorqueCal.M_NominalMap" as shown next.

Symbol: TorqueCal.M_NominalMap [9-3 00 Vig.S19.bin]

Viewtype Easy view | Axis lock mode

Symbol data [TorqueCal.M_NominalMap]

	75	125	140	180	240	300	360	420	480	540	600	660	720	840	900	1050	1200	1300
6000	-52	-19	-12	5	26	51	72	95	118	141	161	183	203	234	249	285	331	360
5820	-49	-17	-10	6	27	51	73	96	120	143	163	185	205	238	253	287	337	375
5440	-41	-11	-3	11	33	55	78	99	121	143	167	190	212	245	260	302	344	390
5060	-36	-8	-1	13	36	59	82	106	127	148	171	194	216	255	270	307	355	405
4680	-33	-2	4	18	41	65	88	110	134	154	175	202	221	262	278	314	361	390
4300	-32	0	6	20	42	65	88	110	134	156	179	203	226	263	283	316	360	391
3920	-31	1	6	22	46	68	92	115	138	162	184	206	230	270	286	324	358	396
3540	-32	1	7	23	48	71	94	118	142	163	187	209	227	274	293	326	360	384
3160	-30	4	11	28	48	73	97	120	144	168	189	211	234	275	292	327	360	381
2780	-28	6	13	30	54	78	99	122	146	170	192	213	234	277	293	327	360	379
2400	-27	8	15	33	57	80	104	125	148	171	192	214	235	273	291	325	370	417
2020	-21	8	16	33	57	81	101	126	150	172	193	214	235	273	290	331	363	380
1900	-18	9	17	34	57	81	101	127	152	173	194	215	236	274	292	332	363	379
1640	-18	8	15	33	59	83	106	127	150	173	192	214	235	267	288	334	363	379
1260	-17	5	14	32	54	80	104	126	149	172	187	216	240	267	288	334	363	379
880	-17	-5	-1	30	53	73	97	120	140	172	187	215	240	267	288	334	363	379



Undo changes | 9-3 00 Vig.S19.bin : Torqu... | Save | Close

Second lambda sensor

In the years Trionic 7 was shipped on cars, several things changed in these cars setups. One of the major changes was the introduction of the second lambda (oxygen, O2) sensor that is placed after the catalyst to ensure the catalyst is working properly. If you want to run software from a double lambda sensor car in a single lambda car, you have to make some changes in the settings. This information is courtesy of L4staero.

Turning off second lambda sensor

You can use this procedure in cars having only one lambda sensor and in cars having two lambda sensors, but with a missing catalyst.

Map	Value	Description
LambdaCal.ST_AdapEnable	0	Second lambda sensor disabled
LambdaCal.ST_AdapEnable	1	Second lambda sensor enabled

Alternative solution to turning off second lambda

Change low limit on O2heaterPostCal.I_LowLim to 0 mA to disable sensor heater error, and change CatDiagCal.LoadHi and LoadLo to values never seen normally, like 30 and 20.

Map	Value	Description
O2HeatPostCal.I_LowLim	0	Second lambda sensor disabled
CatDiagCal.LoadLo	20	Second lambda sensor disabled
CatDiagCal.LoadHi	30	Second lambda sensor disabled
O2HeatPostCal.I_LowLim	230	Second lambda sensor enabled
CatDiagCal.LoadLo	140	Second lambda sensor enabled
CatDiagCal.LoadHi	425	Second lambda sensor enabled

Calibration of OBD2 and LEV EVAP systems

If we want to run a file that was developed for OBD2 or a LEV car in an earlier car we run into problems because the early car is missing a second catalyst, a tank pressure sensor and a purge canister behind the fuel tank. If you have an early B205E/L engine you simply couldn't run a later B205R software version in it because it would through CEL's for the missing hardware. We need to make changes to the file before we can run in on an earlier car (e.g. switch of the control of the new hardware).

OBDCal.OBD2Enabled= This is self explanatory, if car is OBD2 put value at 1 if it's not OBD2 put value at 0.
On this point in later bins(compressed) there is EOBDEnable which is always on in EC2000 EU files and LOBDEnable which is always on in EC2000 RW files. File type is shown in firmware information under engine type.

OBDCal.EnableOBD2Limit= As above but its a 4 byte value. If done in Hex value for a OBD2 car is 00000001 and for non-OBD2 car is 00000000. As shown in T7suite is 2 values. OBD2 cars top value is 1 and bottom value is 0. In non OBD2 car both values are 0.

OBDCal.evapEquipmentExist= If car is equipped with a canister at rear of tank and a tank pressure sensor value will be 1. If neither exist value should be set to 0.

Info on LEV (Low Emission Vehicle)

For example take a 2001 9-3 Aero (or SE as called in USA) equipped with a B205R, Saab's decision to take all "R" engines and clean them so to speak by developing new emission systems for them leaves them with some differences to their low level engine relatives. The term LEV(Low emission vehicle) in this sense refers to Saab's decision to add a second catalytic converter and a tank pressure sensor and a large purge canister behind fuel tank, as well as adding a 2nd oxy to monitor condition of first cat.

Footer information

If we look at the footer in the binary (last page in hex viewer) we see a set of reversed strings. Each of these strings contains an identifier. These identifiers have a hardcoded meaning.

Identifier	Length	Description
0x91	0x09	Ecuid.vehicleidnr
0x94	0x07	Ecuid.ecuhardwversnr
0x95	0x0C	Ecuid.ecusoftwnr
0x97	0x1E	Ecuid.ecusoftwversnr
0x9A	0x04	Ecuid.softwaredate
0x9C	0x04	variable name table crc (not really sure)
0x9B	0x04	Symboltable (packed table with symbol names)
0xF2	0x04	F2 checksum
0xFB	0x04	Romchecksum.piareachecksum
0xFC	0x04	Romchecksum.BottomOffFlash
0xFD	0x04	RomChecksumType
0xFE	0x04	Romchecksum.TopOffFlash
0xFA	0x05	Lastmodifiedby
0x92	0x0F	Ecuid.partnralphacode (IMMO)
0x93	0x07	Ecuid.ecuhardwnr
0xF8	0x02	?
0xF7	0x02	?
0xF6	0x02	?
0xF5	0x02	?
0x90	0x11	Ecuid.scaletable (VIN)
0x99	0x06	Ecuid.testerserialnr
0x98	0x0D	Ecuid.enginetype
0xF9	0x01	Romchecksum.Error

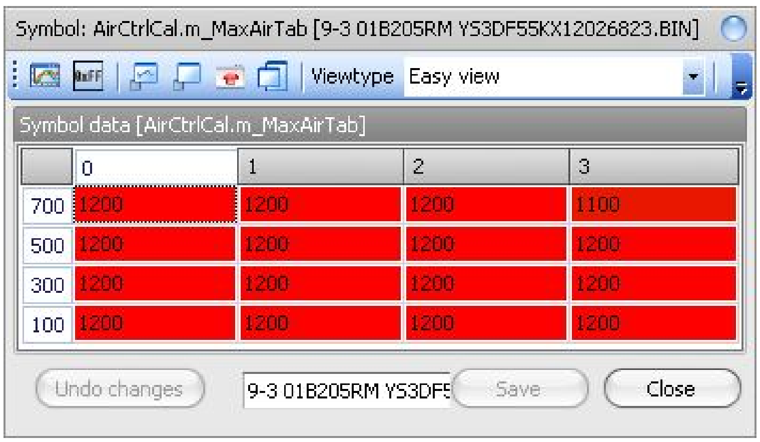
```
0007ff20h: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ; ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0007ff30h: FF FF FF FF FF FF FF FF FF FF 30 30 37 37 33 30 ; ÿÿÿÿÿÿÿÿÿÿ007730
0007ff40h: 33 58 32 43 35 34 46 45 33 53 59 90 11 31 F9 01 ; 3X2C54FE3SY□.1ù.
0007ff50h: 30 31 33 20 72 4E 20 4E 41 43 50 50 98 0C 34 32 ; 013 rN NACPP˜.42
0007ff60h: 38 30 30 30 99 06 35 34 33 32 31 30 33 59 35 43 ; 8000™.5432103Y5C
0007ff70h: 35 35 46 45 33 53 59 90 11 BD E5 F5 02 34 81 F6 ; 55FE3SY□.½åõ.4□ö
0007ff80h: 02 60 A3 F7 02 B9 3B F8 02 36 37 30 30 38 33 35 ; .`£÷.¹;ø.6700835
0007ff90h: 93 07 33 32 32 30 38 34 32 30 31 47 4E 30 31 36 ; “.322084201GN016
0007ffa0h: 38 92 0F FF FF FA FB 42 FA 05 40 B3 06 00 FE 04 ; 8’.ÿÿúûBú.@³..þ.
0007ffb0h: 00 00 00 00 FD 04 FF FF 07 00 FC 04 1D DB 9A 72 ; ....ý.ÿÿ..ü..Ûšr
0007ffc0h: FB 04 31 30 37 30 9A 04 20 20 20 20 20 20 20 20 ; û.1070š.
0007ffd0h: 55 45 20 45 35 30 32 42 20 35 2D 39 97 14 44 37 ; UE E502B 5-9—.D7
0007ffe0h: 34 2E 43 30 42 46 58 31 41 45 95 0C 35 34 38 30 ; 4.C0BFX1AE•.5480
0007fff0h: 38 33 35 94 07 20 20 38 33 37 30 38 33 35 91 09 ; 835”.  8370835‘.
```

Tuning the T7

Tuning with T7Suite

To get the ECU to produce more engine output, several parameters (maps) have to altered. This chapter will give you a general idea on what to change – and why – for getting to an approximate stage II equivalent. The example is a 9-3 B205R.

AirCtrlCal.m_MaxAirTab
Airmass value from controller where area map has reached max-area and there is no point to increase the I-part. Resolution is 1 mg/c



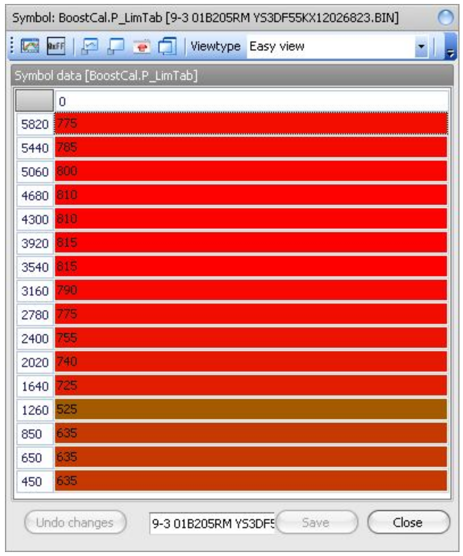
	0	1	2	3
700	1200	1200	1200	1100
500	1200	1200	1200	1200
300	1200	1200	1200	1200
100	1200	1200	1200	1200

.

AirCtrlCal.m_MaxAirE85Ta (if running on E85)
Same as above for E85
BoostCal.I_LimTab
Load limit tab. to enable the I Part of boost regulator. If the load request from Airmass master is above this value plus the hysteresis is the I Part enabled and the throttle closed loop is disabled. If the load request from Airmass master is below this value is the I Part disabled and the throttle is allowed to run in closed loop.

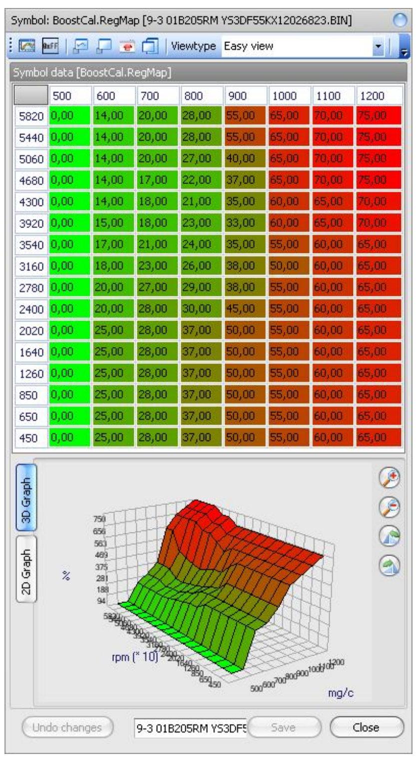
	0
5820	775
5440	785
5060	800
4680	810
4300	810
3920	815
3540	815
3160	790
2780	775
2400	755
2020	740
1640	725
1260	2000
850	2000
650	2000
450	2000

BoostCal.P_LimTab
Load limit tab. to enable the P Part of boost regulator. If the load request from Airmass master is above this value plus the hysteresis is the P Part enabled. If the load request from Airmass master is below this value is the P Part disabled.



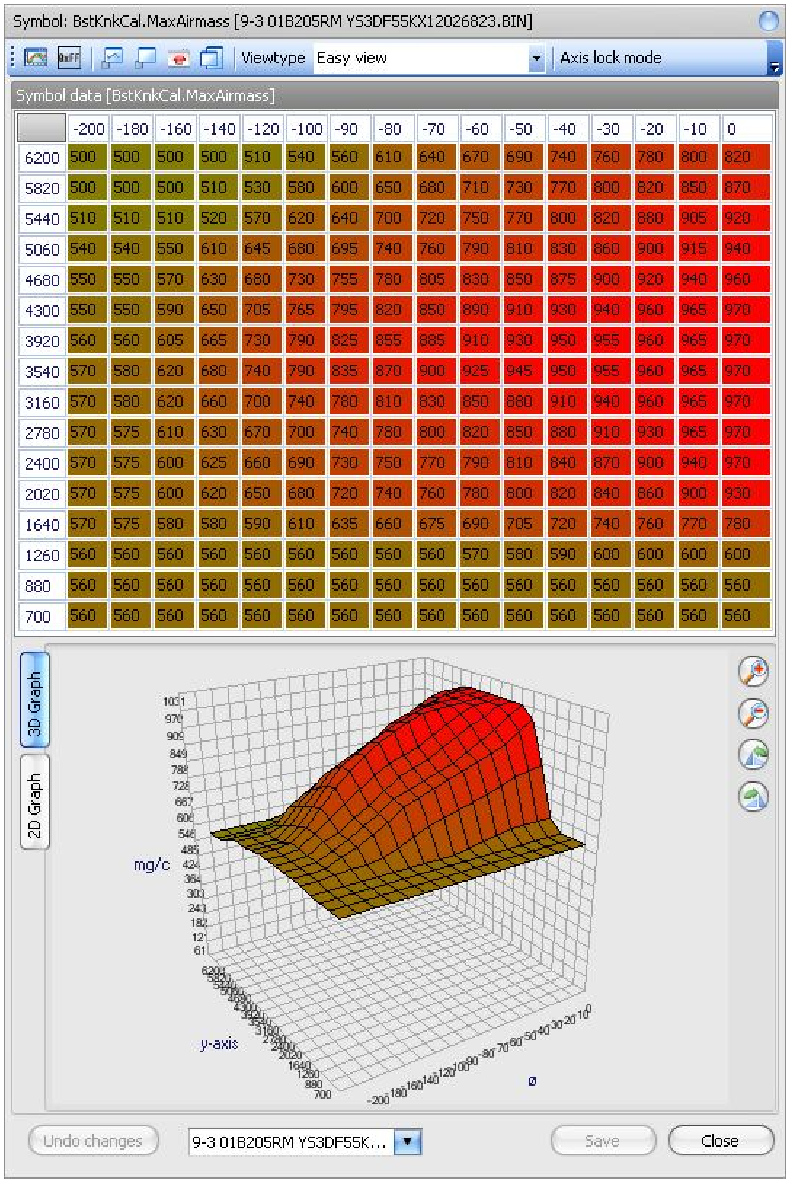
	0
5820	775
5440	785
5060	800
4680	810
4300	810
3920	815
3540	815
3160	790
2780	775
2400	755
2020	740
1640	725
1260	525
850	635
650	635
450	635

BoostCal.RegMap
Main constant matrix. Resolution is 0.1 %.



	500	600	700	800	900	1000	1100	1200
5820	0,00	14,00	20,00	28,00	55,00	65,00	70,00	75,00
5440	0,00	14,00	20,00	28,00	55,00	65,00	70,00	75,00
5060	0,00	14,00	20,00	27,00	40,00	65,00	70,00	75,00
4680	0,00	14,00	17,00	22,00	37,00	65,00	70,00	75,00
4300	0,00	14,00	18,00	21,00	35,00	60,00	65,00	70,00
3920	0,00	15,00	18,00	23,00	33,00	60,00	65,00	70,00
3540	0,00	17,00	21,00	24,00	35,00	55,00	60,00	65,00
3160	0,00	18,00	23,00	26,00	38,00	50,00	60,00	65,00
2780	0,00	20,00	27,00	29,00	38,00	55,00	60,00	65,00
2400	0,00	20,00	28,00	30,00	45,00	55,00	60,00	65,00
2020	0,00	25,00	28,00	37,00	50,00	55,00	60,00	65,00
1640	0,00	25,00	28,00	37,00	50,00	55,00	60,00	65,00
1260	0,00	25,00	28,00	37,00	50,00	55,00	60,00	65,00
850	0,00	25,00	28,00	37,00	50,00	55,00	60,00	65,00
650	0,00	25,00	28,00	37,00	50,00	55,00	60,00	65,00
450	0,00	25,00	28,00	37,00	50,00	55,00	60,00	65,00

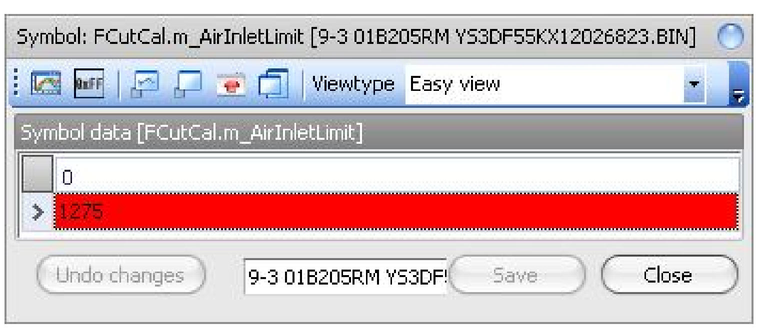
BstKnkCal.MaxAirmass (divide by 3,1 for approx torque, ignition, airtemp etc affect this!)
Map for max allowed Airmass for manual gearbox, m_nHigh. Resolution is 1 mg/c.



	-200	-180	-160	-140	-120	-100	-90	-80	-70	-60	-50	-40	-30	-20	-10	0
6200	500	500	500	500	510	540	560	610	640	670	690	740	760	780	800	820
5820	500	500	500	510	530	580	600	650	680	710	730	770	800	820	850	870
5440	510	510	510	520	570	620	640	700	720	750	770	800	820	880	905	920
5060	540	540	550	610	645	680	695	740	760	790	810	830	860	900	915	940
4680	550	550	570	630	680	730	755	780	805	830	850	875	900	920	940	960
4300	550	550	590	650	705	765	795	820	850	890	910	930	940	960	965	970
3920	560	560	605	665	730	790	825	855	885	910	930	950	955	960	965	970
3540	570	580	620	680	740	790	835	870	900	925	945	950	955	960	965	970
3160	570	580	620	660	700	740	780	810	830	850	880	910	940	960	965	970
2780	570	575	610	630	670	700	740	780	800	820	850	880	910	930	965	970
2400	570	575	600	625	660	690	730	750	770	790	810	840	870	900	940	970
2020	570	575	600	620	650	680	720	740	760	780	800	820	840	860	900	930
1640	570	575	580	580	590	610	635	660	675	690	705	720	740	760	770	780
1260	560	560	560	560	560	560	560	560	560	570	580	590	600	600	600	600
880	560	560	560	560	560	560	560	560	560	560	560	560	560	560	560	560
700	560	560	560	560	560	560	560	560	560	560	560	560	560	560	560	560

BstKnkCal.MaxAirmassAu
Map for max allowed Airmass for automatic gearbox, m_nHigh. Resolution is 1 mg/c.

FCutCal.m_AirInletLimit
If the "MAF.m_AirInletFuel" is higher than this limit during m_AirInletTime will the fuelcut be activated (pressure guard).



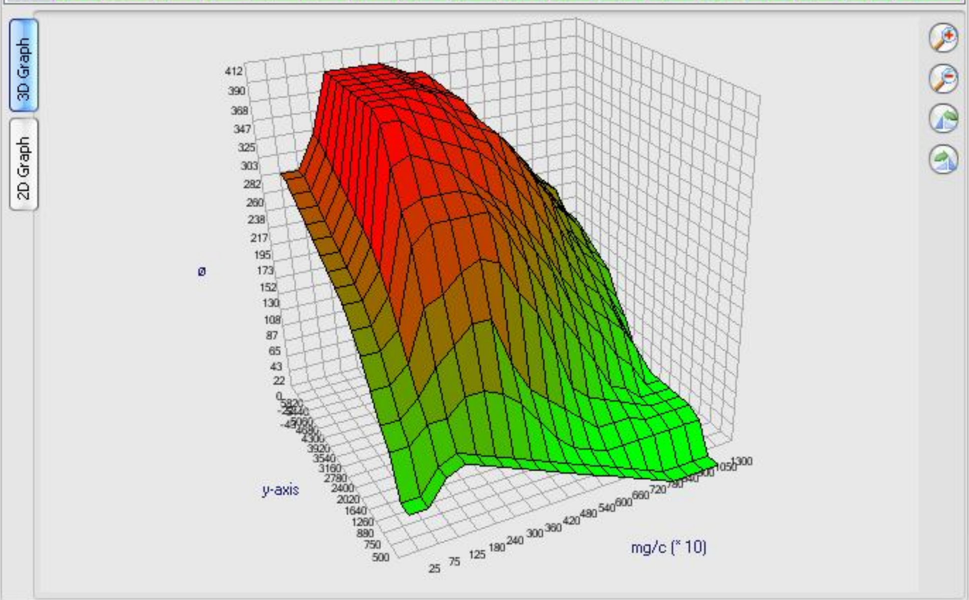
IgnE85Cal.fi_AbsMap (if you want to change the ignition)
Ignition map for E85 fuel. Resolution is 0.1 °.
IgnNormCal.Map (if you want to change the ignition)
Normal ignition map. Resolution is 0.1 °.

Symbol: IgnNormCal.Map [9-3 01B205RM YS3DF55KX12026823.BIN]

Viewtype Easy view Axis lock mode Autoscale Mathematics

Symbol data [IgnNormCal.Map]

	25	75	125	180	240	300	360	420	480	540	600	660	720	780	840	900	1050	1300
5820	28,00	32,00	39,00	39,00	39,00	39,00	38,00	37,00	36,00	33,50	31,00	29,50	26,50	24,00	22,00	20,00	16,00	14,00
5440	28,00	32,00	39,00	39,00	39,00	39,00	38,00	38,00	36,00	33,50	31,00	29,50	26,50	23,50	21,50	20,00	15,00	13,00
5060	27,00	31,00	39,00	39,00	39,00	39,00	38,00	38,00	36,00	33,50	31,00	29,00	26,50	23,00	21,00	18,00	14,00	12,00
4680	26,00	30,00	39,00	39,00	39,00	39,00	38,00	37,00	36,00	33,00	31,00	28,00	26,50	23,00	20,00	17,50	13,00	11,00
4300	25,00	29,00	39,00	39,00	39,00	39,00	38,00	37,00	36,00	33,00	31,00	28,00	25,00	21,50	19,50	17,00	12,00	10,00
3920	24,00	28,00	39,00	39,00	39,00	39,00	38,00	36,00	35,00	33,00	31,00	28,00	24,50	19,50	17,50	15,00	9,50	7,00
3540	23,00	27,00	39,00	39,00	39,00	38,00	37,00	35,00	34,00	33,00	30,00	27,00	24,00	19,50	16,00	14,00	8,00	5,00
3160	22,00	26,00	39,00	39,00	38,00	37,00	36,00	34,00	33,00	30,00	28,00	25,00	21,00	17,00	14,00	11,50	6,50	2,00
2780	21,00	25,00	38,00	38,00	36,00	35,00	34,00	33,00	31,00	28,00	25,00	21,50	17,50	14,00	12,00	9,00	5,00	1,00
2400	20,00	23,00	34,00	35,00	34,00	33,00	31,00	29,00	27,00	24,00	21,00	17,00	14,00	11,50	9,50	7,50	4,00	0,00
2020	18,00	21,00	30,00	31,00	31,00	31,00	30,00	28,00	25,00	22,00	19,00	14,50	11,50	9,00	5,50	3,50	0,00	0,00
1640	16,00	18,00	27,00	29,00	29,00	29,00	29,00	26,00	22,00	18,00	14,00	10,50	7,00	3,00	2,00	0,00	0,00	0,00
1260	13,00	15,00	21,00	22,00	24,00	25,00	24,00	20,00	17,00	12,00	6,00	4,00	3,00	2,00	1,00	0,00	0,00	0,00
880	10,00	12,00	15,00	17,00	19,00	19,00	13,00	9,00	7,00	5,00	3,00	1,00	1,00	1,00	0,00	0,00	0,00	0,00
750	8,00	10,00	10,00	12,00	12,00	11,00	9,00	6,00	3,00	1,00	0,00	-1,00	-1,00	-1,00	-1,00	-1,00	-1,00	-1,00
500	3,00	6,00	7,00	6,00	5,00	4,00	3,00	2,00	1,00	0,00	-1,00	-2,00	-3,00	-4,00	-5,00	-5,00	-5,00	-5,00



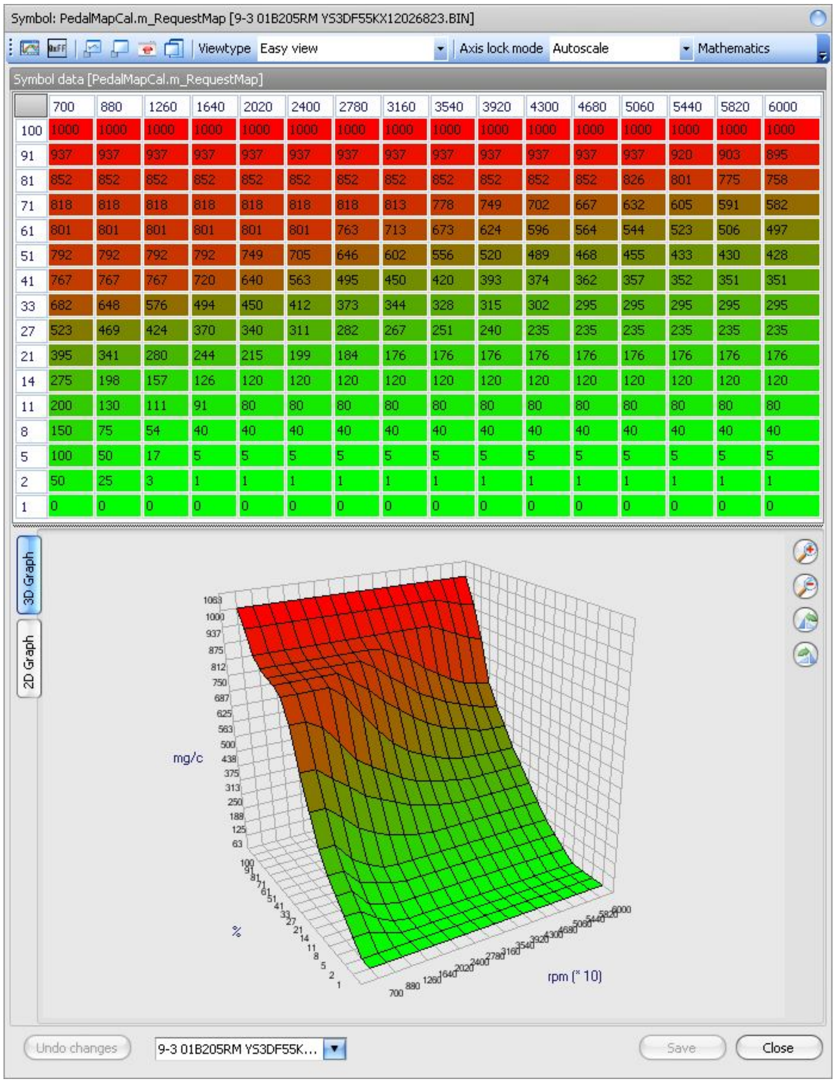
Undo changes 9-3 01B205RM YS3DF55K... Save Close

MapChkCal.CheckSum (automatically updated in between every map change with T7suite!)

MaxVehicCal.v_MaxSpeed(max speed)

PedalMapCal.m_RequestMap

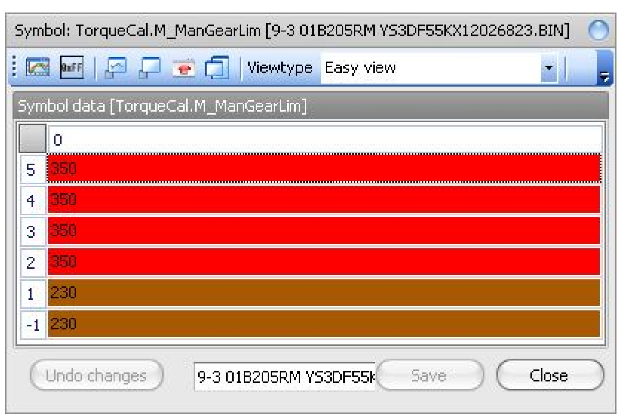
Requested Airmass from the driver as a function of rpm and accelerator pedal position. Resolution is 1 mg/c.



	700	880	1260	1640	2020	2400	2780	3160	3540	3920	4300	4680	5060	5440	5820	6000
100	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
91	937	937	937	937	937	937	937	937	937	937	937	937	937	920	903	895
81	852	852	852	852	852	852	852	852	852	852	852	852	826	801	775	758
71	818	818	818	818	818	818	818	813	778	749	702	667	632	605	591	582
61	801	801	801	801	801	801	763	713	673	624	596	564	544	523	506	497
51	792	792	792	792	749	705	646	602	556	520	489	468	455	433	430	428
41	767	767	767	720	640	563	495	450	420	393	374	362	357	352	351	351
33	682	648	576	494	450	412	373	344	328	315	302	295	295	295	295	295
27	523	469	424	370	340	311	282	267	251	240	235	235	235	235	235	235
21	395	341	280	244	215	199	184	176	176	176	176	176	176	176	176	176
14	275	198	157	126	120	120	120	120	120	120	120	120	120	120	120	120
11	200	130	111	91	80	80	80	80	80	80	80	80	80	80	80	80
8	150	75	54	40	40	40	40	40	40	40	40	40	40	40	40	40
5	100	50	17	5	5	5	5	5	5	5	5	5	5	5	5	5
2	50	25	3	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

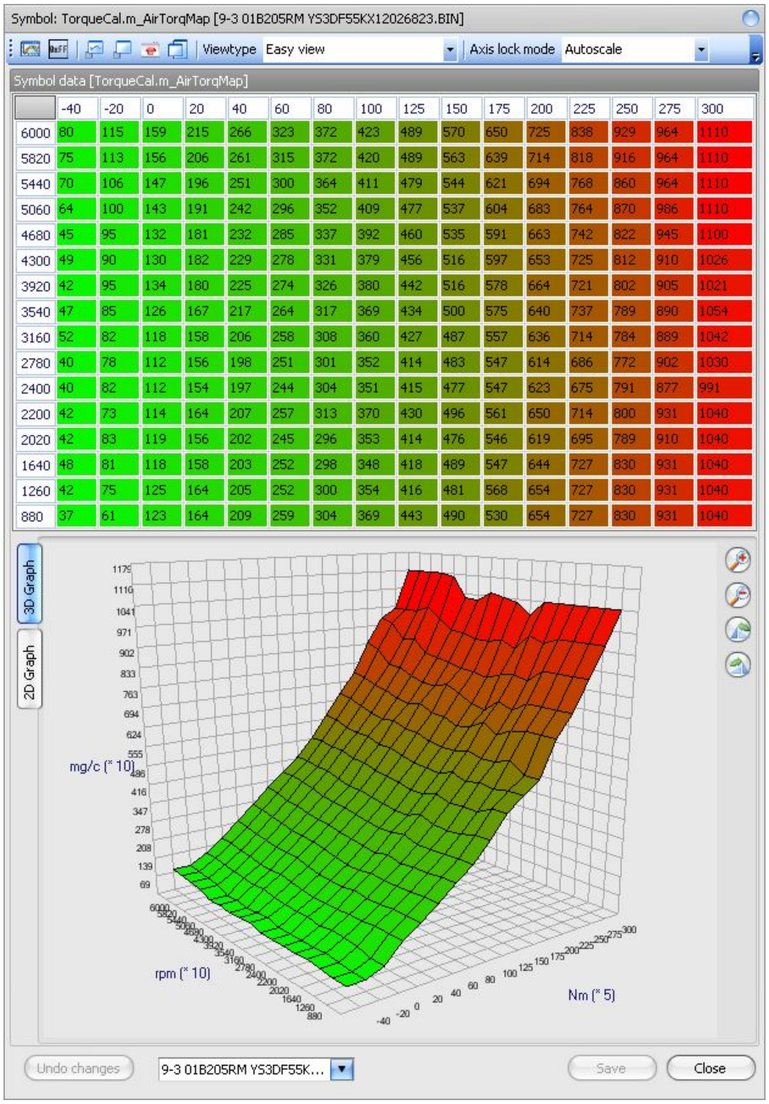
TorqueCal.M_ManGearLim

Maximum engine torque limit for each gear in the manual gearbox. Resolution is 1 Nm.



	0
5	350
4	350
3	350
2	350
1	230
-1	230

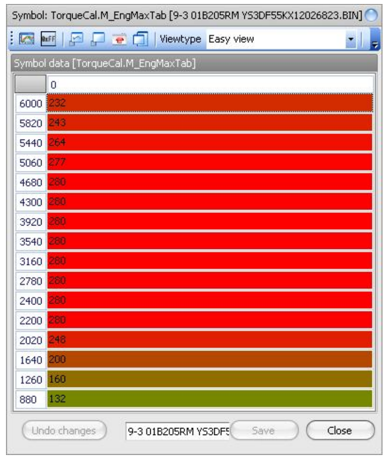
TorqueCal.m_AirTorqMap (This is where all torque limiters take their data from and therefore needs to be "fooled" if you are running 400nm+ or an automatic!)
Data-matrix for nominal Airmass. Engine speed and torque are used as support points. The value in the matrix + friction Airmass (idle Airmass) will create the pointed torque at the pointed engine speed. Resolution is 1 mg/c.
axis to the above map: **TorqueCal.m_AirXSP**



	-40	-20	0	20	40	60	80	100	125	150	175	200	225	250	275	300
6000	80	115	159	215	266	323	372	423	489	570	650	725	838	929	964	1110
5820	75	113	156	206	261	315	372	420	489	563	639	714	818	916	964	1110
5440	70	106	147	196	251	300	364	411	479	544	621	694	768	860	964	1110
5060	64	100	143	191	242	296	352	409	477	537	604	683	764	870	986	1110
4680	45	95	132	181	232	285	337	392	460	535	591	663	742	822	945	1100
4300	49	90	130	182	229	278	331	379	456	516	597	653	725	812	910	1026
3920	42	95	134	180	225	274	326	380	442	516	578	664	721	802	905	1021
3540	47	85	126	167	217	264	317	369	434	500	575	640	737	789	890	1054
3160	52	82	118	158	206	258	308	360	427	487	557	636	714	784	889	1042
2780	40	78	112	156	198	251	301	352	414	483	547	614	686	772	902	1030
2400	40	82	112	154	197	244	304	351	415	477	547	623	675	791	877	991
2200	42	73	114	164	207	257	313	370	430	496	561	650	714	800	931	1040
2020	42	83	119	156	202	245	296	353	414	476	546	619	695	789	910	1040
1640	48	81	118	158	203	252	298	348	418	489	547	644	727	830	931	1040
1260	42	75	125	164	205	252	300	354	416	481	568	654	727	830	931	1040
880	37	61	123	164	209	259	304	369	443	490	530	654	727	830	931	1040

TorqueCal.M_EngMaxTab

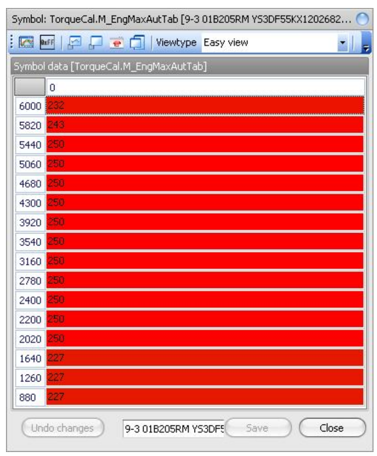
Data-table for maximum engine out put torque for manual cars. Resolution is 1 Nm.



	0
6000	232
5820	243
5440	264
5060	277
4680	280
4300	280
3920	280
3540	280
3160	280
2780	280
2400	280
2200	280
2020	248
1640	200
1260	160
880	132

TorqueCal.M_EngMaxAutTab

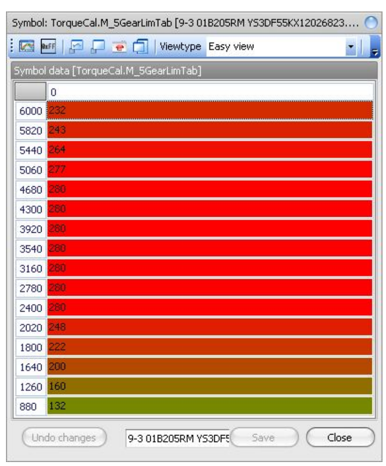
Data-table for maximum engine output torque for automatic cars. Resolution is 1 Nm.



	0
6000	232
5820	243
5440	250
5060	250
4680	250
4300	250
3920	250
3540	250
3160	250
2780	250
2400	250
2200	250
2020	250
1640	227
1260	227
880	227

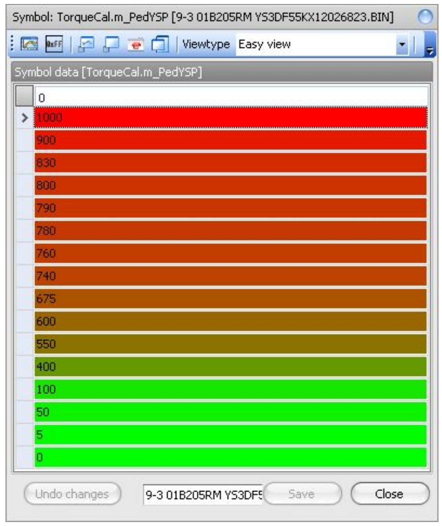
TorqueCal.M_5GearLimTab

Data-table for maximum engine output torque for manual cars on fifth gear. Resolution is 1 Nm.



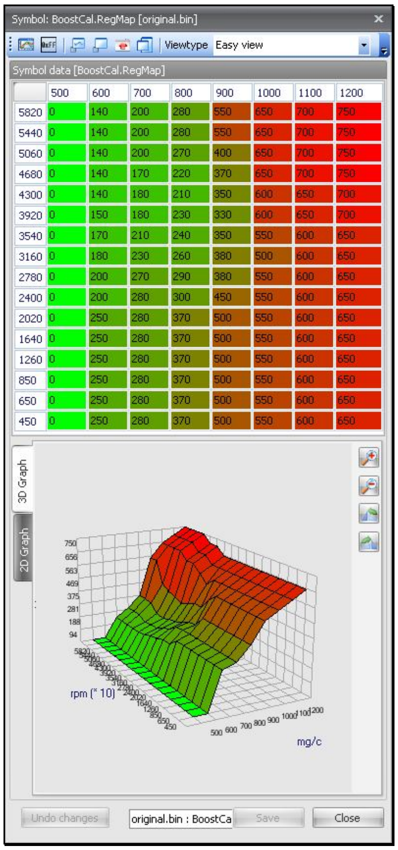
	0
6000	232
5820	243
5440	264
5060	277
4680	280
4300	280
3920	280
3540	280
3160	280
2780	280
2400	280
2020	248
1800	222
1640	200
1260	160
880	132

TorqueCal.M_EngMaxE85Tab (if running on E85)
Data-table for maximum engine output torque when running on E85. Resolution is 1 Nm.
TorqueCal.m_PedYSP
Air mass support points for (Calc) X_AccPedalMap. Resolution is 1 mg/combustion.

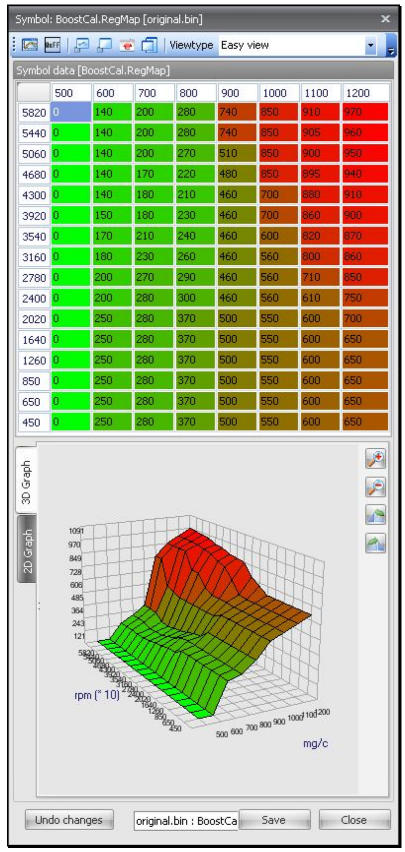


Tuning Boost calibration

This map holds percentages (0.1% accurate) of how much air should be passed to the return hose of the boost control value. The higher the value, to more air is bled off and the less the wastegate will open (and thus, the more air the turbo will be spooling). As you can see, the more Airmass is requested (x – axis) the more the wastegate is held shut and thus, the more Airmass the turbo will be providing. If we want more Airmass from the turbo, we need to keep the wastegate shut longer and thus we have to enter higher numbers on the right side of the table.



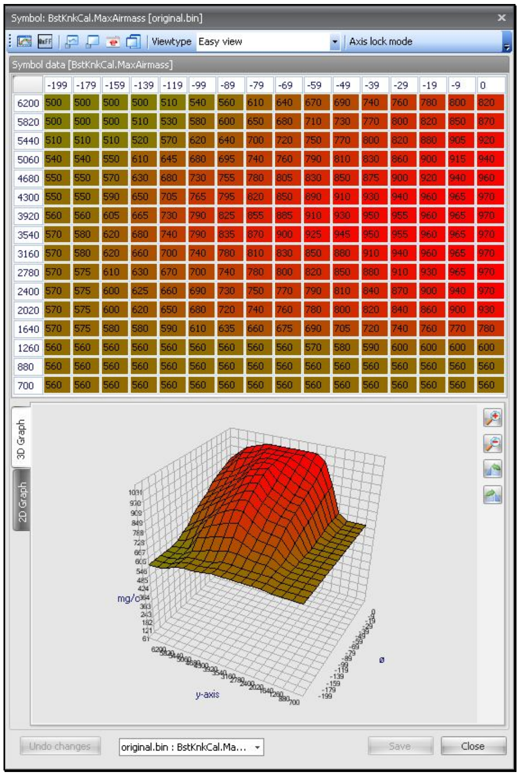
	500	600	700	800	900	1000	1100	1200
5820	0	140	200	280	550	650	700	750
5440	0	140	200	280	550	650	700	750
5060	0	140	200	270	400	650	700	750
4680	0	140	170	220	370	650	700	750
4300	0	140	180	210	350	600	650	700
3920	0	150	180	230	330	600	650	700
3540	0	170	210	240	350	550	600	650
3160	0	180	230	260	380	500	600	650
2780	0	200	270	290	380	550	600	650
2400	0	200	280	300	450	550	600	650
2020	0	250	280	370	500	550	600	650
1640	0	250	280	370	500	550	600	650
1260	0	250	280	370	500	550	600	650
850	0	250	280	370	500	550	600	650
650	0	250	280	370	500	550	600	650
450	0	250	280	370	500	550	600	650

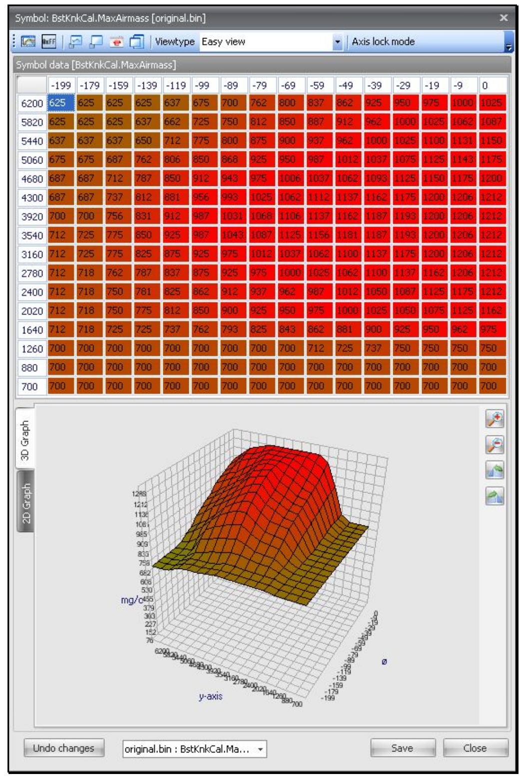


	500	600	700	800	900	1000	1100	1200
5820	0	140	200	280	740	850	910	970
5440	0	140	200	280	740	850	905	960
5060	0	140	200	270	510	850	900	950
4680	0	140	170	220	480	850	895	940
4300	0	140	180	210	460	700	880	910
3920	0	150	180	230	460	700	860	900
3540	0	170	210	240	460	600	820	870
3160	0	180	230	260	460	560	800	860
2780	0	200	270	290	460	560	710	850
2400	0	200	280	300	460	560	610	750
2020	0	250	280	370	500	550	600	700
1640	0	250	280	370	500	550	600	650
1260	0	250	280	370	500	550	600	650
850	0	250	280	370	500	550	600	650
650	0	250	280	370	500	550	600	650
450	0	250	280	370	500	550	600	650

Altering Airmass limiter

To be able to flow more air though the engine that is allowed in the stock configuration we will have to modify the Airmass limiter tables as well. Note that there are two different ones, one for manual gearbox and one for automatic gearbox. This example will only show the manual gearbox table (BstKnkCal.MaxAirmass) but for automatic cars BstKnkCal.MaxAirMassAu needs to be changes.



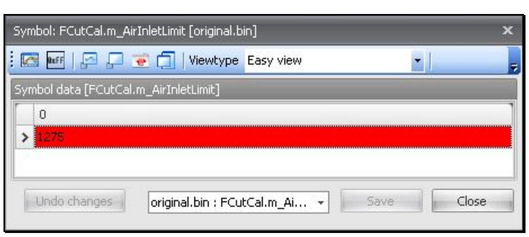


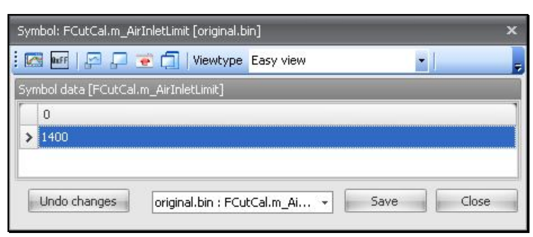
As you can see, the maximum amount of Airmass allowed is approximately 970 mg/c. We need to change the table so that it will allow more Airmass. In this case we just up the table with 25% with the math functions in T7Suite.

NOTE: Please do not simply turn off this limiter by setting it way higher than the actually intended level because it is an important limiter to provide engine safety.

Altering fuelcut

Then there's the fuelcut function to worry about. We need to increase the limit of what the fuel cut function will accept to prevent it from shutting of fuel too early.

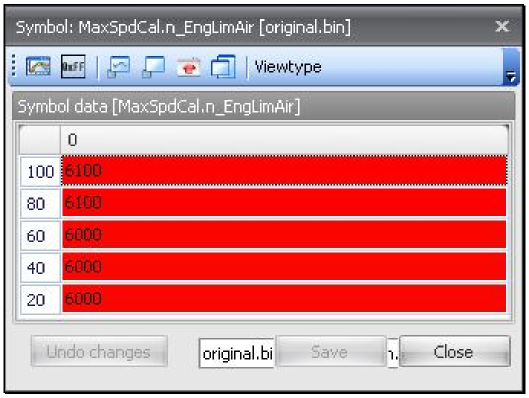


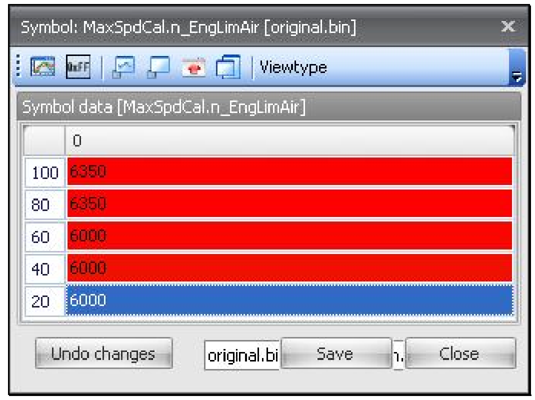


NOTE: Please do not simply turn off this limiter by setting it way higher as the actually intended level because it is an important limiter to provide engine safety.

Engine speed limiter

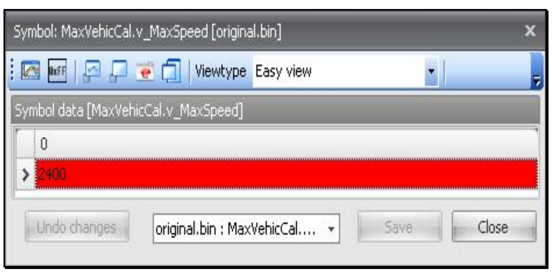
To prevent the system to reduce Airmass above engine speeds that are still acceptable we need to change MaxSpdCal.n_EngLimAir as well. Y axis values are engine temperature (coolant). Please note that 200 rpm above this limit, the fuel cut mechanism will become active!

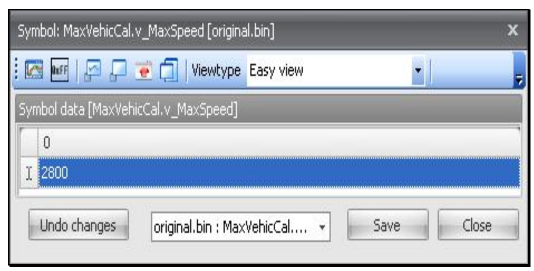




Vehicle speed limiter

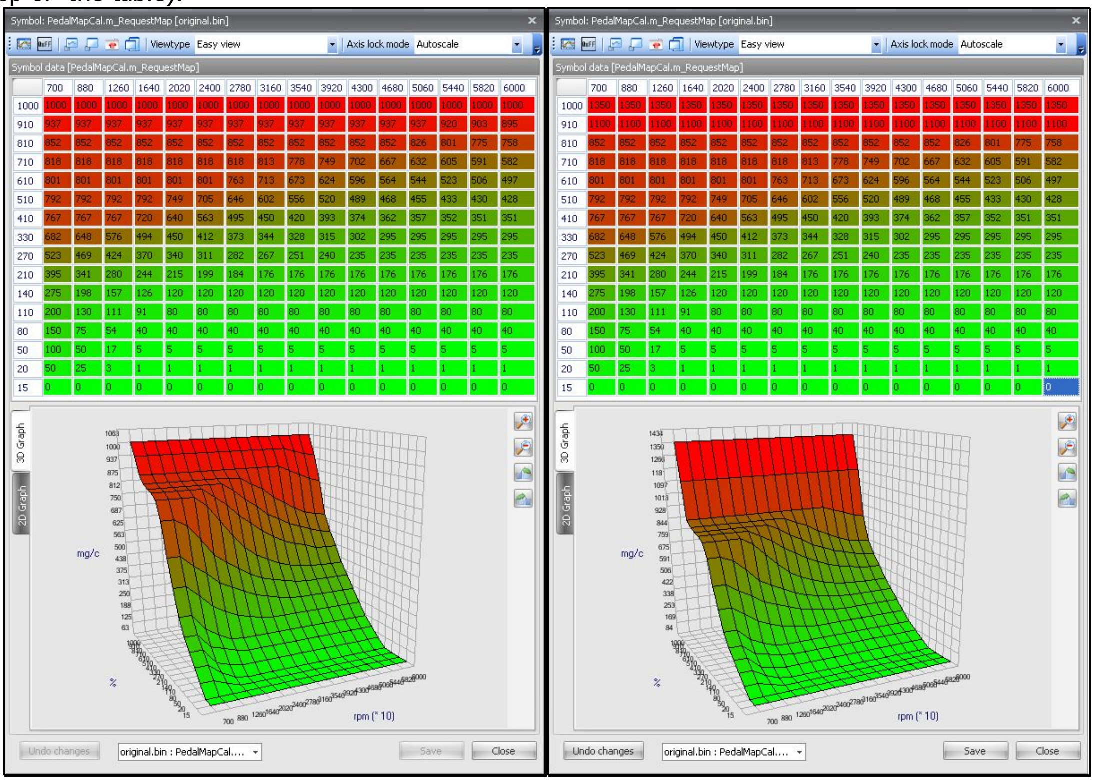
An option is to increase the vehicle speed limiter as well. In this stock binary the vehicle speed is limited to 240 km/h. We can change it to – for example 280 km/h.





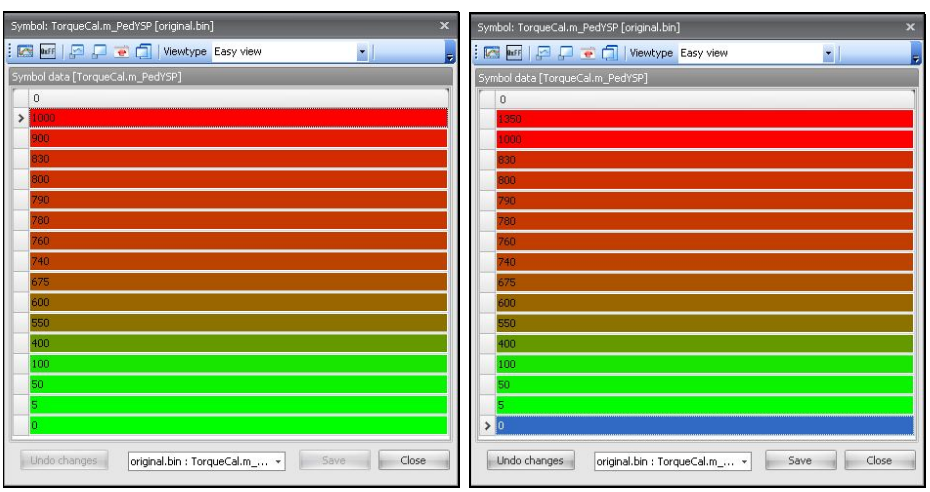
Airmass request

To get more from the engine than in the stock configuration we need to actually request more Airmass for a certain pedal position and rpm site. This can be done through PedalMapCal.m_RequestMap. Because we want more power at wide open throttle (from the drivers perspective) we need to increase the Airmass request at pedal positions in the high percentage range (top of the table).

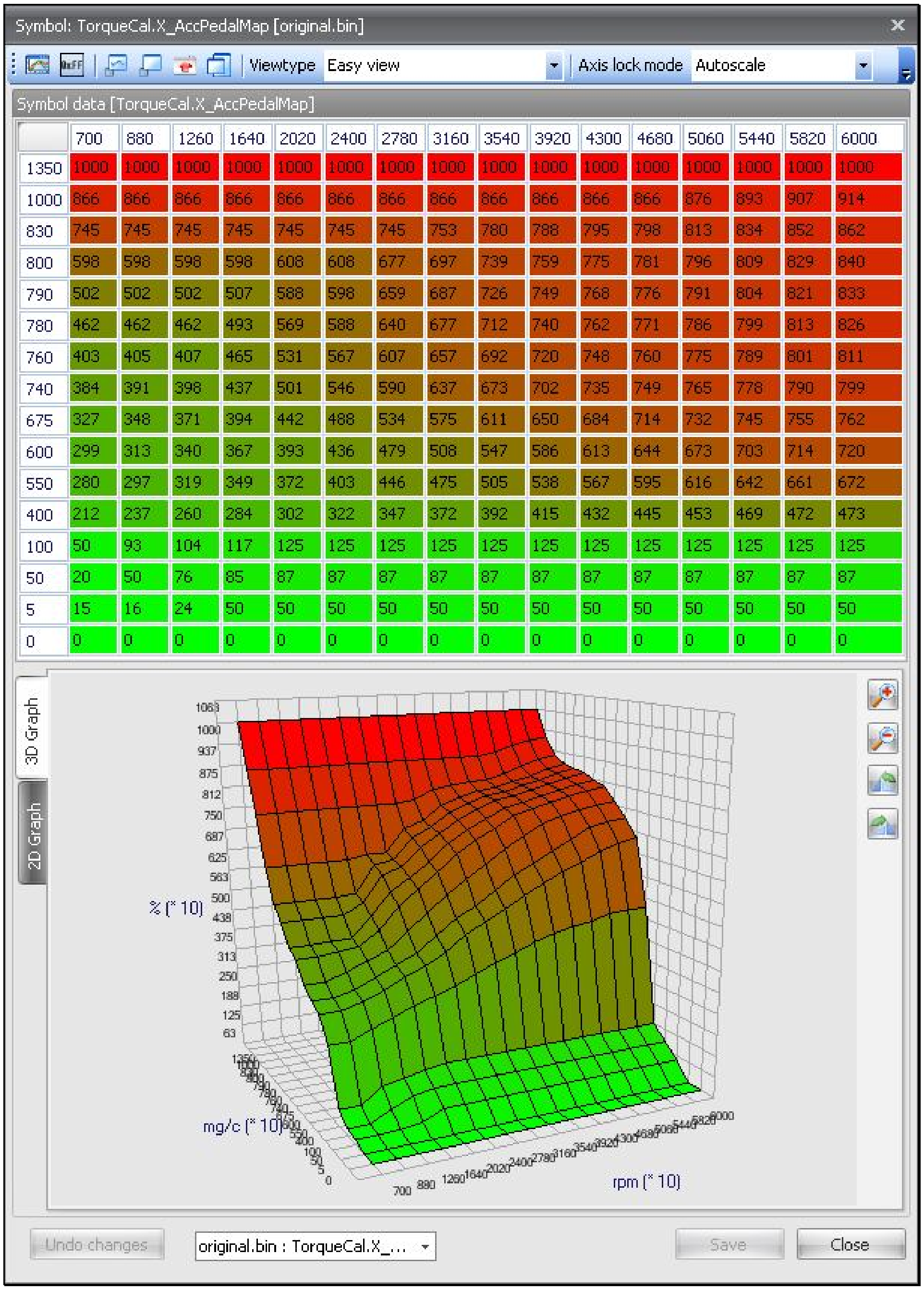


As you can see we increased the top two rows so that a maximum of 1350 mg/c will be requested.

In addition we need to alter the y axis support point for the pedal map that lets Trionic lookup a pedal position for a given Airmass. This map is called TorqueCal.m_PedYSP. This axis map should support the maximum Airmass we're requesting in the m_Requestmap, so in our case we need to modify the map to match the 1350 mg/c we are requesting as a maximum.



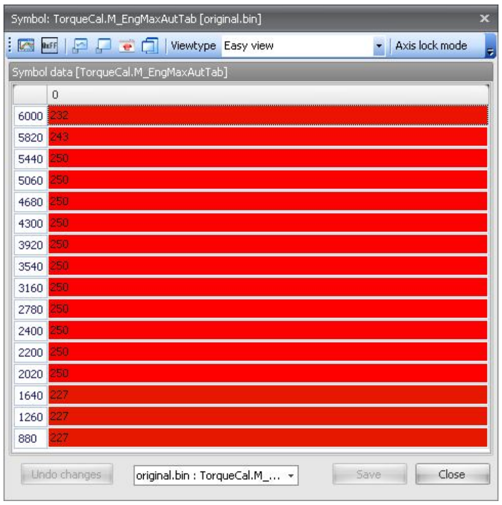
The map that uses this axis is called TorqueCal.x_AccPedalMap. It it shown below with the altered axis values for clarification.

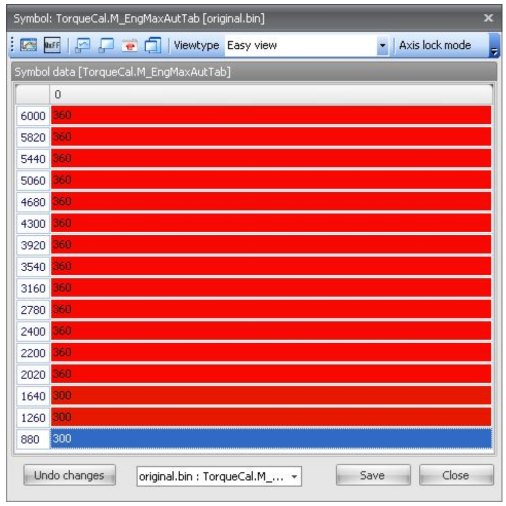


	700	880	1260	1640	2020	2400	2780	3160	3540	3920	4300	4680	5060	5440	5820	6000
1350	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1000	866	866	866	866	866	866	866	866	866	866	866	866	876	893	907	914
830	745	745	745	745	745	745	745	753	780	788	795	798	813	834	852	862
800	598	598	598	598	608	608	677	697	739	759	775	781	796	809	829	840
790	502	502	502	507	588	598	659	687	726	749	768	776	791	804	821	833
780	462	462	462	493	569	588	640	677	712	740	762	771	786	799	813	826
760	403	405	407	465	531	567	607	657	692	720	748	760	775	789	801	811
740	384	391	398	437	501	546	590	637	673	702	735	749	765	778	790	799
675	327	348	371	394	442	488	534	575	611	650	684	714	732	745	755	762
600	299	313	340	367	393	436	479	508	547	586	613	644	673	703	714	720
550	280	297	319	349	372	403	446	475	505	538	567	595	616	642	661	672
400	212	237	260	284	302	322	347	372	392	415	432	445	453	469	472	473
100	50	93	104	117	125	125	125	125	125	125	125	125	125	125	125	125
50	20	50	76	85	87	87	87	87	87	87	87	87	87	87	87	87
5	15	16	24	50	50	50	50	50	50	50	50	50	50	50	50	50
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

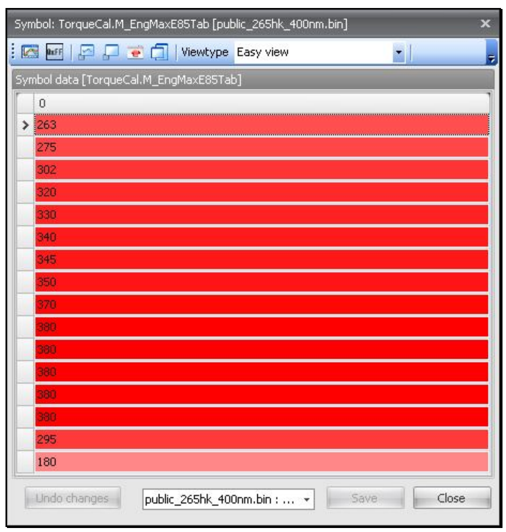
Torque limiter

To prevent to system to reduce Airmass above a certain engine output, the torque limiter needs to be increased according to expected engine output.

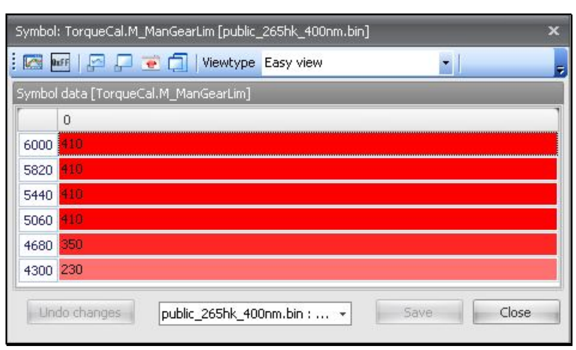




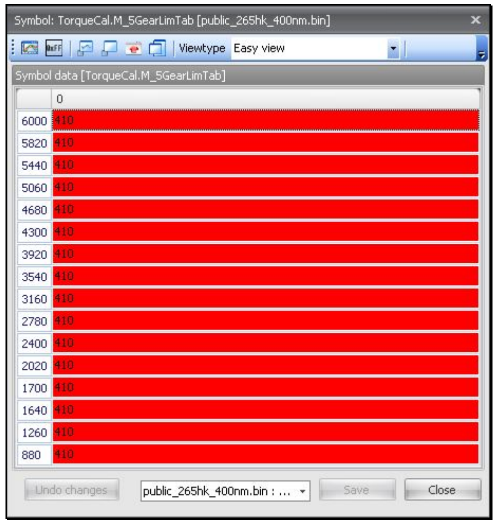
Torque limiter for E85 fuel



Torque limiter for manual gearbox in higher revs

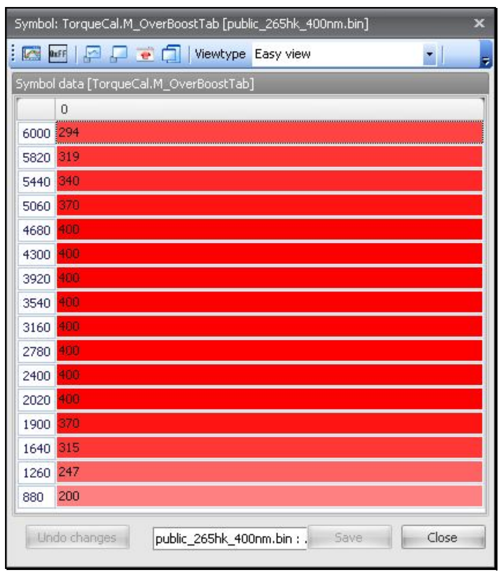


Torque limiter in 5th gear



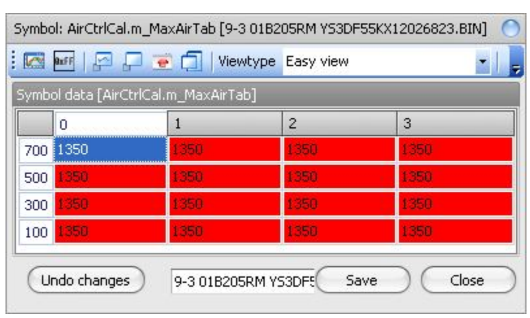
	0
6000	410
5820	410
5440	410
5060	410
4680	410
4300	410
3920	410
3540	410
3160	410
2780	410
2400	410
2020	410
1700	410
1640	410
1260	410
880	410

Overboost table



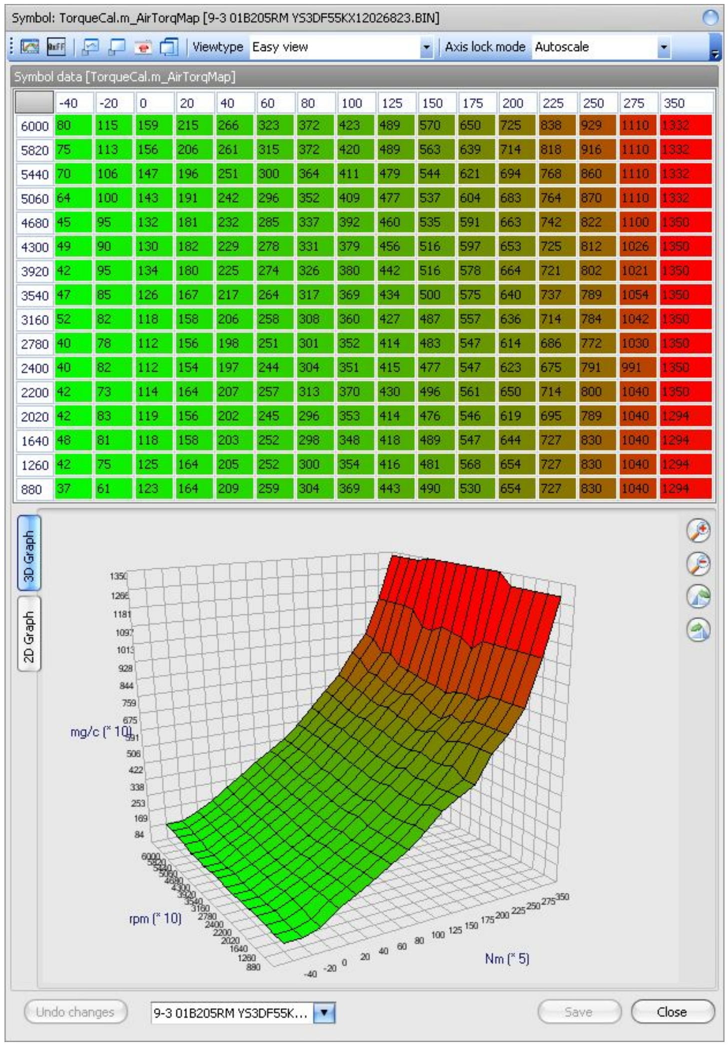
	0
6000	294
5820	319
5440	340
5060	370
4680	400
4300	400
3920	400
3540	400
3160	400
2780	400
2400	400
2020	400
1900	370
1640	315
1260	247
880	200

Maximum Airmass for I-part of PID controller (**AirCtrlCal.m_MaxAirTab)**



	0	1	2	3
700	1350	1350	1350	1350
500	1350	1350	1350	1350
300	1350	1350	1350	1350
100	1350	1350	1350	1350

TorqueCal.m_AirTorqMap (This is where all torque limiters take their data from and therefore needs to be "fooled" if you are running 400nm+ or an automatic!)
Data-matrix for nominal Airmass. Engine speed and torque are used as support points. The value in the matrix + friction Airmass (idle Airmass) will create the pointed torque at the pointed engine speed. Resolution is 1 mg/c.



	-40	-20	0	20	40	60	80	100	125	150	175	200	225	250	275	350
6000	80	115	159	215	266	323	372	423	489	570	650	725	838	929	1110	1332
5820	75	113	156	206	261	315	372	420	489	563	639	714	818	916	1110	1332
5440	70	106	147	196	251	300	364	411	479	544	621	694	768	860	1110	1332
5060	64	100	143	191	242	296	352	409	477	537	604	683	764	870	1110	1332
4680	45	95	132	181	232	285	337	392	460	535	591	663	742	822	1100	1350
4300	49	90	130	182	229	278	331	379	456	516	597	653	725	812	1026	1350
3920	42	95	134	180	225	274	326	380	442	516	578	664	721	802	1021	1350
3540	47	85	126	167	217	264	317	369	434	500	575	640	737	789	1054	1350
3160	52	82	118	158	206	258	308	360	427	487	557	636	714	784	1042	1350
2780	40	78	112	156	198	251	301	352	414	483	547	614	686	772	1030	1350
2400	40	82	112	154	197	244	304	351	415	477	547	623	675	791	991	1350
2200	42	73	114	164	207	257	313	370	430	496	561	650	714	800	1040	1350
2020	42	83	119	156	202	245	296	353	414	476	546	619	695	789	1040	1294
1640	48	81	118	158	203	252	298	348	418	489	547	644	727	830	1040	1294
1260	42	75	125	164	205	252	300	354	416	481	568	654	727	830	1040	1294
880	37	61	123	164	209	259	304	369	443	490	530	654	727	830	1040	1294

Finally, we're all done! ☺

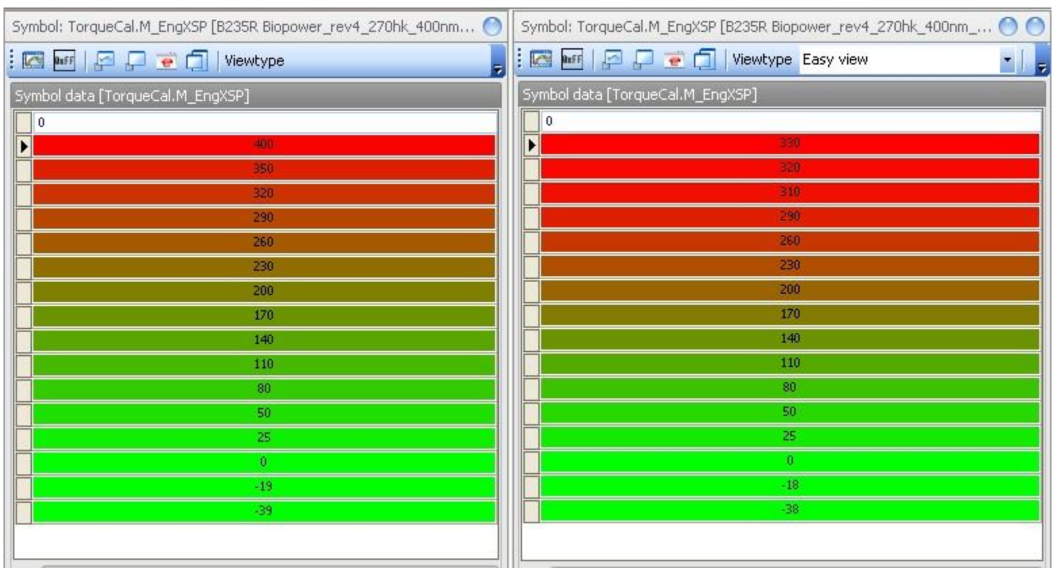
Automatic transmission specifics

In automatic Trionic 7 cars the TCM (Traction Control Module) sends a torque limit over can to the ECU (Trionic). This means – theoretically - you cannot achieve more torque than the torque limit the TCM dictates. The only known way around this at present time is to make Trionic THINK it's not making that much torque, so we have to fool the ECU into thinking it is still below the torque limit set by the TCM.
There are different TCM limits depending on year, engine type, gearbox etc.
A MY01-AERO AUT has a 330NM limiter while a 5 speed automatic gearbox has a 350Nm limit.

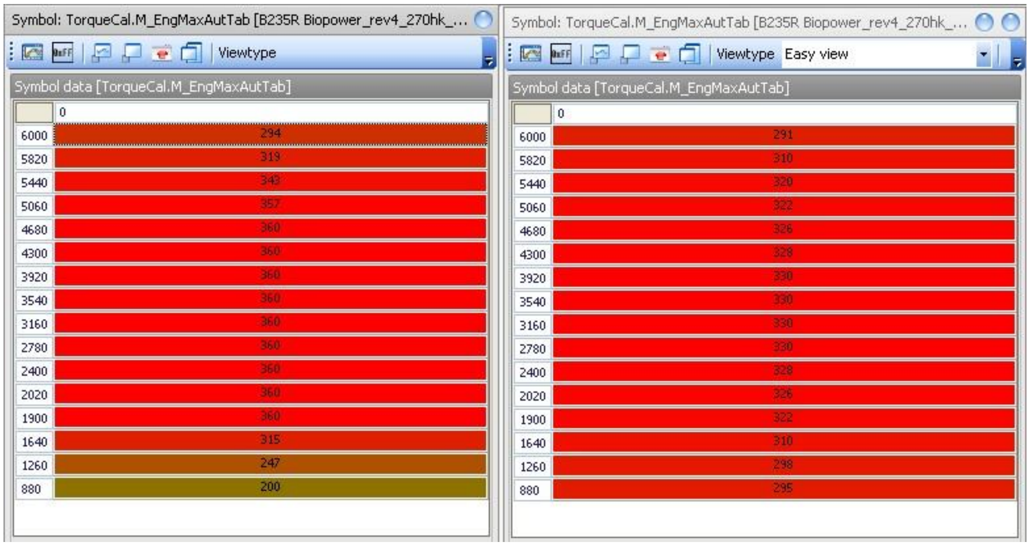
To fool the ECU into thinking it is making less torque is to rescale the x-axis for **TorqueCal.mAirTorqMap** which is **TorqueCal.M_EngXSP**. The top value in this list must be no more than the TCM limit.

In this case the top three rows have been altered to keep the calculated torque below 330Nm.

400 -> 330Nm
350 -> 320Nm
320 -> 310Nm



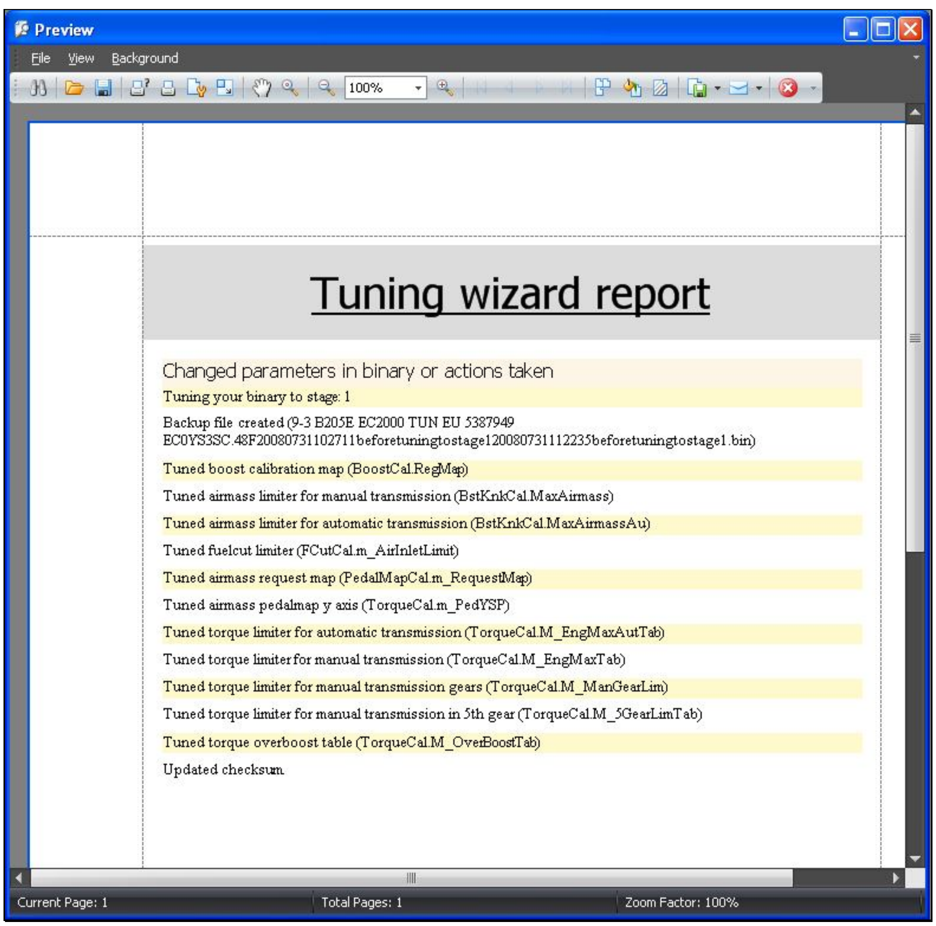
This means that when requesting 330Nm you will actually get 400, 320 will get you 350 and so on. The torque limiters in **TorqueCal.M_EngMaxAutTab** must be scaled with this in mind...
In this case 400Nm between 2780 and 3920rpm. Values in between you need to recalculate, at 4300rpm the user wanted 390Nm, (400=330, 350=320) means that 322=360, 324=370, 326=380, 328=390nm



If you want to use the same bin in manual cars all manual limiters must be calculated and set correctly!
NOTE: In Bio power bins *TorqueCal.M_EngMaxE85Tab*!

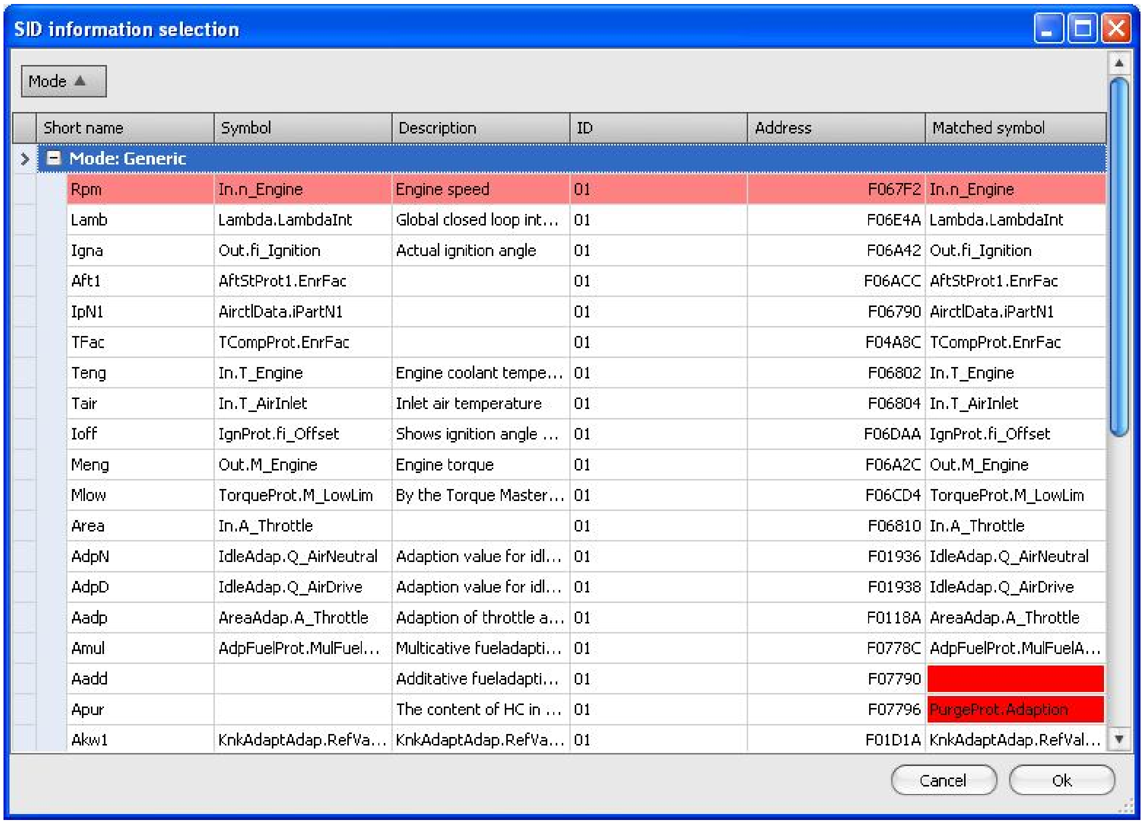
Using the tuning wizard

T7Suite incorporates a tuning wizard. This wizard allows you to automatically alter the maps in the binary file to get it to a stage I equivalent file. The wizard can be activated by selecting "Tuning" > "Easy tune to stage I" from the menu. A dialog will appear in which you can confirm that you want to tune the file to stage I. Once you've selected this the process will start. After a few seconds a report will appear showing all actions taken on your file.

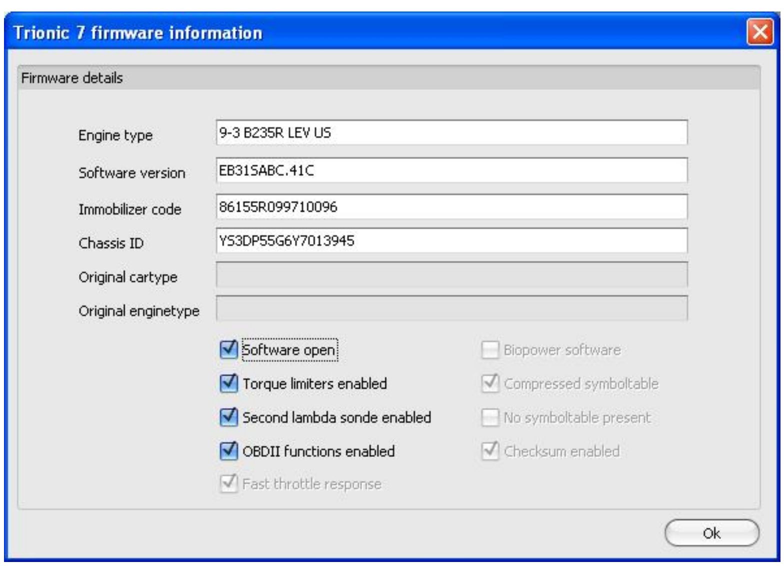


Stuff for SID information display

T7Suite incorporates a function to allow visualization of information on the SID (System Information Display). This way you can view real-time information without utilizing the Canbus interface. You can select the variables you want the SID to display using the SID information selection option in T7Suite.



Also, the software should be opened to be able to view the selected data on the SID. This is done in the firmware information screen.



Some tips on how to use the SID information option:

- ECMStat.ST_ActiveAirDem shows the current Airmass limiter
- ECMStat.P_Engine shows calculated engine power (hp)
- ECMStat.AirFuelRatio shows calculated AFR
- ECMStat.p_Diff shows boost pressure (manifold – ambient) in 0.1 kPa units
- BstKnkProt.MapPointer shows the offset in 0.1 degrees for BstKnk.MaxAirmass (so, the ignition offset for knock)
- ExhaustCal.ST_Enable allows you to enable and disable the EGT algorithm. These algorithms are based on the stock engine and won't be properly calibrated for a stage 3+ setup.
- KnkDetAdap.KnkCntCyl first 2 bytes show cylinder 1 knock count, next 2 bytes shows cylinder 2 knock count etc.

CAN Bus interface

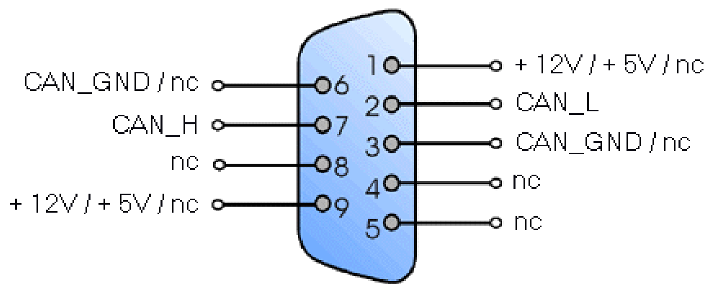
Interfacing with the Trionic T7 unit through the CAN bus is possible.

General information

Chip used on Trionic side: Intel AN825257
Communication speed used: 615 Kbit/s

The most frequently used interface for this is the Lawicel CANUSB interface that can be found on www.canusb.com. This interface can convert CAN signals onto you USB port and vice versa. The interface has a USB port on one side – that connects to you computer – and an male RS232 (DB9) connector on the other side. This side connects to the CAN bus of the Trionic.

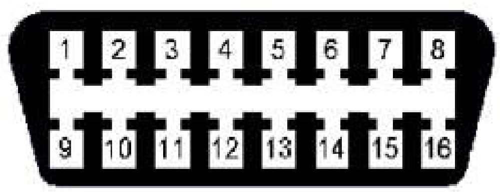
The Lawicel interface has the following pin out on the DB9 connector.



Connecting to CAN bus with ECU on your desk

OBDII socket pin out

On some models, the OBDII port enables you to connect to the I bus directly. On most models, you need to wire into the P-pus (preferably, because data transmission rates are tenfold of that on the I-bus) or into the I-bus directly. The P-bus can be found at the pins of the ECU (as described on the previous page), the I-bus can be found in a lot of places like the CD changer connector in the trunk (other spots are shown in the table below).



Pinnumber	Description
1	
2	J1850 Bus+
3	
4	Chassis Ground
5	Signal Ground
6	CAN High (J-2284) **Note: not on all models**
7	ISO 9141-2 K Line
8	
9	
10	J1850 Bus-
11	Airbag Controller (?)
12	ABS Controller (?)
14	CAN Low (J-2284) **Note: not on all models**
15	ISO 9141-2 L Line
16	Battery Power

I-Bus There components are hooked up to the I-Bus (instrumentation bus):

SID	Saab Information Display
ACC	Automatic Climate Control
RADIO	Radion control unit
CDC	CD-Changer – in trunk
PSM	Power Seat Memory
STC	Soft Top Control – for carbios
Twice	Theft Warning Integrated Control Electronics
DICE	Dashboard Integrated Control Electronics

P-Bus These components are hooked up to the P-Bus (powertrain bus)

TC/ABS	Traction Control/ABS – in case of ABS only (without Traction Control) it is connected to MIU
Trionic 7	Motormanagement for petrol engines
VP44-PSG16	Motormanagement for diesel engines
TCM	Transmission Control System – in autmatic geared cars (petrol only)

Real-time symbols in Trionic 7

This table gives a summary of interesting symbols to monitor in Trionic 7.

SID name	Symbolname	Description
Rpm	In.n_Engine	Engine speed UNIT : rpm MAX : 8000 MIN : 25 (set to 10 when engine starts to move) TRANS : V = P. Resolution is 1. Interval is Every combustion / 5 ms when engine is still.
Lamb	Lambda.LambdaInt	Global closed loop integrator. Update : every combustion. V6: Bank 1. Resolution is 0.01 %.
Igna	Out.fi_Ignition	Actual ignition angle. A positive value is before TDC and a negative value is after TDC. Resolution is 0.1 °. Interval is Every combustion.
Teng	In.T_Engine	Engine coolant temperature UNIT : (C MAX : 150 MIN : -40 TRANS : V = P. Resolution is 1. Interval is 1000 ms.
STAd	E85Adap.ST_Adap	
Tair	In.T_AirInlet	Inlet air temperature UNIT : (C MAX : 140 MIN : -40 TRANS : V = P. Resolution is 1. Interval is 1000 ms.
Ioff	IgnProt.fi_Offset	Shows ignition angle output from offset functions. Resolution is 0.1 °.
Meng	Out.M_Engine	Engine torque UNIT : Nm MAX : 400 MIN : -100 TRANS : V = (P+. Resolution is 1. Interval is 10ms.
Mlow	TorqueProt.M_LowLim	By the Torque Master selected lowest torque limit request, corrected with adaption value made at idle.
nErr	obdNoOfFaults	Number of errors stored
Pbef	In.p_AirBefThrottle	Engine inlet air pressure UNIT : kPa MAX : 300 MIN : 0 TRANS : V = P * 10. Resolution is 0.1. Interval is Every combustion.
Pinl	In.p_AirInlet	Engine inlet air pressure UNIT : kPa MAX : 300 MIN : 0 TRANS : V = P * 10. Resolution is 0.1. Interval is Every combustion.
Pair	In.p_AirAmbient	Barometric air pressure UNIT : kPa MAX : 120 MIN : 50 TRANS : V = P * 10. Resolution is 0.1. Interval is 250 ms.
mReq	m_Request	Requested airmass
mAIR	MAF.m_AirInlet	Airmass in milligram per combustion. This airmass is the actual load value in the ECM. (Unfiltered) Calculated from ActualIn.Q_AirInlet. Resolution is 1 mg/c. Interval is every combustion.
Miss	Missf.nrOfCountedMisfire	Counts the nr of misfire that has not been filtered or rpm diff filtered.
Pfac	BoostProt.PFac	Calculate P part for regulator. load diff * P const P = 100. Update : every 10 msec. Resolution is 0.1 %.
Ifac	BoostProt.IFac	Calculated I part for regulator. load diff * I const I = I + 1000. Update : every 10 msec. Resolution is 0.1 %.
PWM	Out.PWM_BoostCntrl	Duty-cycle for boost pressure valve. Resolution is 0.1 %. Interval is every 20 ms.
tSta	ECMStat.t_StartTime	Engine start time, measured by measuring the time from that the battery volatage decreases 1.0V to the time engine speed reached 1000 rpm.
LIMP	OBDAdap.ThrLimpHomeNr	Last reported throttle limphome number.
Mode	SID.ST_Mode	Mode settings to see different "values"
Me85	In.X_EthanolSensor	
Ad85	E85.X_EthanolActual	

Ca85	E85Prot.X_EthanolActual	
Amul	AdpFuelProt.MulFuelAdapt	Multicative fueladaption value. Resolution is ñ 0.01 %.
FFac	Purge.FuelFac	The fuelfactor from the purge function. Resolution is 0.01 %.
ReFu	E85Adap.ST_ReFuel	
Crnk	CrnkCas.ST_Fuel	
MxLo	LambdaProt.MaxLoadNorm	Max load (airmass) for closed loop during normal conditions. Update : every combustion. Resolution is 1 mg/c.
SFuL	E85Adap.V_SavedFuelLevel	
VFue	In.V_FuelTank	Fuel level UNIT : l (litre) MAX : 100 MIN : 0 TRANS : V = P * 10. Resolution is 0.1. Interval is 1000 ms.
Aadd	AdpFuelProt.AddFuelAdapt+2	Additative fueladaption value. Resolution is 0.01 mg/c.
Aadp	AreaAdap.A_Throttle	Adaption of throttle area. Interval is 250ms.
AdpD	IdleAdap.Q_AirDrive	Adaption value for idlespeed regulation (drive activated). This value is added to the PID and Constant part of the regulator. If the I-part is limited will the adaption stop. Resolution is 0.01 g/s.
AdpN	IdleAdap.Q_AirNeutral	Adaption value for idlespeed regulation (drive not activated). This value is added to the PID and Constant part of the regulator. If the I-part is limited will the adaption stop. Resolution is 0.01 g/s.
Akw1	KnkAdaptAdap.RefValueWind	
Akw2	KnkAdaptAdap.RefValueWind+2	
AMR	CanIn.ST_EngineInterv	Engine intervention is requested from ESP (AMR)
Apur	Purge.HCCont	The content of HC in the purge air. Resolution is 0.1 %. ay CanIn.a_Lateral Lateral acceleration, only implemented on cars with ESP. Resolution is 0.5 m/s2.
Badp	BoostAdap.Adaption	Adaption value for boost control. Interval is Every 100ms.
BMR	CanIn.ST_BrakeInterv	Brake intervention is requested from ESP (BMR)
CLUi	Out.CMD_CoastLUInhibit	Inhibit coast slip lock up
Cmem	EngTip.ST_Active	Status flag showing if tipin is active 0 = Not active 1 = Tip-in active 2 = Tip-out active
CSLU	In.ST_TCMCSLU	Coast Lock up slip state 0 = No request 1 = Fuel cut inhibit 2 = Fuel cut allowed
DTI	Out.M_DTI	Drivers Torque Intention. The torque that the driver requests converted from air to torque. Limitations from all functions excluding TCM and TCS are included in the signal UNIT : Nm MAX : 400 MIN : -100
Fcod	obdFaults	codes for errors stored
FFAd	Purge.m_FuelPrg	Fuel flow from purge. Resolution is 0.01 mg/c.
Flow	Purge.Flow	The actual purge flow. Resolution is 1 mg/s.
FMXF	PurgeProt.FuelFacMaxFlow	Maximum allowed purge flow in respect to maximum allowed fuel factor at actual load. Resolution is 1 mg/s.
Frez	PurgeProt.AdpFreeze	Adaption freeze status.
Fuel	BFuelProt.CurrentFuelCons	
Gear	In.X_ActualGear	Actual gear on automatic gearbox. 2 - Reverse 3 - Neutral 5 - Gear 1 6 - Gear 2 7 - Gear 3 8 - Gear 4 11 - Gear 3, lock up 12 - Gear 4, lock up Interval is every 50 ms.
GSI	Out.CMD_GearShiftInhibit	Prevent TCM from shifting.

HCnt	Purge.HCCont	The content of HC in the purge air. Resolution is 0.1 %.
In.X	In.X_AccPedal	Pedal position UNIT : % MAX : 130 MIN : 0 TRANS : V = P * 10. Resolution is 0.1. Interval is 20 ms.
Iput	ActualIn.n_GearBoxIn	Transmission input rpm (turbine speed) Used to detect when the load is changed for the engine when gear is engaged. Resolution is 1 rpm. Interval is every 50 ms.
JeLi	JerkProt.JerkFactor	Threshold value for changing shift pattern to "no lockup"
Jerk	ECMStat.JerkFactor	This factor describes the jerking of the engine. The formula for calculating this is abs(ECMStat.n_EngineDelta2) * factor. The factor is for scaling it so it will be possible to filter it. The calibratable value used for filtering is nEngCal.FilterFactor. Since the jerk factor is based on every combustion, it is not possible to compare the numbers for 6 cylinder engines and 4 cylinder.
Kph1	ActualIn.v_Vehicle	Left front wheel speed UNIT : km/h MAX : 300 MIN : 0 (detection of min. 1.0 km/h) TRANS : V = P * 10. Resolution is 0.1. Interval is 100 ms.
Kph2	ActualIn.v_Vehicle2	Vehicle speed, measured on the rear wheel, sent from MIU. UNIT : km/h MAX : 300 MIN : 0 TRANS : V = P * 10. Resolution is 0.1. Interval is 100 ms. LwsI CanIn.fi_SteeringAngle Stearing angle (LwsIn), only implemented on cars with ESP . Resolution is 3 °.
mAir	MAF.m_AirInlet	Airmass in milligram per combustion. This airmass is the actual load value in the ECM. (Unfiltered) Calculated from ActualIn.Q_AirInlet. Resolution is 1 mg/c. Interval is every combustion.
Mair	In.M_TCSTorqueReq	Maximum torque request from TCS system via CAN. Resolution is 1 Nm. Interval is Every 20 ms.
MiFi	Missf.nrOfFilteredMisfire	Number of missfires occurred
Mnom	Torque.M_Nominal	Nominal engine output torque at a certain enginespeed and inlet airmass. Read from matrix.
MTCM	ActualIn.M_TCMLimitReq	Maximum engine torque request from TCM UNIT : Nm MAX : 400 MIN : -100 TRANS : V = P. Resolution is 1. Interval is 10 ms.
Mtot	In.M_TCSTotalReq '	Total torque request from ESP equiped cars. The differance in torque between In.M_TCSTorqueReq and In.M_TCSTotalReq is taken with ignition retardation. Resolution is 1 Nm. Interval is Every 20 ms.
NoIg	Out.ST_NoIgnitionRetard	Ignition retardation is not allowed due to overheating the catalytic converter Oput DiffPSProt.v_GearBoxOut TCM gearbox output speed converted to vehicle speed. Resolution is 0.1 km/h. Interval is Every 100ms.
Pdif	ECMStat.p_Diff	Differance between inlet manifold air pressure and external air pressure. Resolution is 0.1 kPa.
Peng	ECMStat.P_Engine	Calculated engine power. Measured in horsepower.
Perc	PurgeProt.PurgePercent	Purge flow/Air mass flow ratio. Resolution is 0.01 %.
PMXF	PurgeProt.PdiffMaxFlow	Maximum flow allowed by the diff. pressure. Resolution is 1 mg/s.
Ppwm	Purge.Valve	Purge valve PWM. Resolution is 0.1 %.
PrSt	Purge.Status	Status of the purge function.

ReqF	PurgeProt.ReqFlow	Requested purge flow. Resolution is 1 mg/s.
ShPn	In.ST_TCMShiftPattern	Active TCM shift pattern 0 = ECO 1 = Pwr 2 = Wusp 3 = Wnt 4 = US1 5 = US2 6 = Hot1 7 = Hot2 8 = Jerk 9 = Rep 10 = DS 11 = Tap U/D
Tign	TorqueCal.T_NoIgnRet	No retardation of ignition above this exhaust temperature
Tlmp	unknown	
TngA	ActualIn.T_Engine	Engine coolant temperature UNIT : (C MAX : 150 MIN : -40 TRANS : V = P. Resolution is 1. Interval is 1000 ms.
TTCM	In.T_TCMOil	Oil temperature in automatic gearbox
tTCM	In.t_TCMTrqLimDuration	Maximum engine torque duration UNIT : ms MAX : 2500 MIN : 0 TRANS : V = P. Resolution is 1. Interval is 10 ms.
vGiF	CanIn.fi_YawVelocity	Yaw velocity (vGiF), only implemented on cars with ESP. Resolution is 0.02 °.
vVLF	In.v_Vehicle	Left front wheel speed UNIT : km/h MAX : 300 MIN : 0 (detection of min. 1.0 km/h) TRANS : V = P * 10. Resolution is 0.1. Interval is 100 ms.
vVRF	In.v_Vehicle3	Right front wheel speed UNIT : km/h MAX : 300 MIN : 0 (detection of min. 1.0 km/h) TRANS : V = P * 10. Resolution is 0.1. Interval is 100 ms.
Xacc	Out.X_AccPedal	Pedal position UNIT : % MAX : 100 MIN : 0 TRANS : V = P*1. Resolution is 0.1. Interval is 20ms.

SAAB I-bus communication

Courtesy of Tomili and General Failure

The I-Bus is an internal bus (Instrumentation Bus) that connects together instruments such as the radio, ACC and the SID (Information Display). The I-bus is the non-critical bus which means that vehicle critical information is not sent over it. That is done through the P (powertrain) bus.
The I-bus enables us to communicate with several devices in the car including the TWICE, DICE etc. Both the I and the P bus are CAN bus based communication buses which enables us to use a normal CAN adapter to interface with them.

The I bus follows the ISO 11898-2 standard which is the high speed variant, although it only implements a maximum communication speed of 47,619 Kbit/s. Messages on a CAN bus are sent within CAN frames that include an identifier number, number of data bytes, the actual data (called the payload) and checksum (to verify it the data was transferred correctly). There are two CAN frame formats: the basic frame (referred to as CAN2.0A) and the extended frame (referred to as CAN2.0B). The basic frame has a 11-bit identifier field, when the extended frame has a 29-bit identifier which makes it possible to extend the number of message types that can be sent over the bus.
The I bus uses two signal (just like any other CAN bus connection) which are called I+ and I-. These refer to CAN High and CAN Low signals in general CAN terminology.
Details on I bus communication:

Frame type:	CAN 2.0A (11-bit identifiers)
Bus speed:	47,619 Kbit/s
Timing register settings:	BTR0 0xCB, BTR1 0x9A

As an example the engine speed is located as a 16-bit integer value on message ID 460h (hexadecimal) in the second and third bytes (I define first byte as the most significant byte). Speed is in the same message in fourth and fifth bytes. The speed value is multiplied by ten, so you have to divide the 16-bit integer by 10 to get the real speed value.
Here's an example ID 460h message: 00 03 9C 00 2A 00 00 00
Engine rpm is 039Ch (hexadecimal format) = 924 (decimal format) = 942 RPM
Speed is 002Ah = 42 = 4,2 km/h

220h - Trionic data initialization

This message must be sent before any queries to the Trionic can be sent. Trionic responds with a 238h message.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
220h	3Fh	81h	01h	33h	02h	40h	00h	00h

238h - Trionic data initialization reply

Trionic sends this message after a 220h message has been sent on the bus.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
238h	40h	BFh	21h	C1h	01h	11h	02h	58h

240h - Trionic data query

This message is sent as a command to the Trionic. It can be used to query "OBD-II" information. I use the quotation marks because basically this information is also available from the OBD-II interface. The problem with the OBD-II interface is that it's slow. By using directly the I-Bus (or even better, the P-Bus) you can achieve refresh rates 10 Hz...~50 Hz. The DATA byte below indicates what data is requested. The COMMAND byte is 01h for requesting OBD-II information. With COMMAND byte as 1Ah you can request information from the Trionic software header. The header comprises of ASCII and hexadecimal data about e.g. Vehicle Identification Number (VIN), Immobilizer code, Software Saab part number, Hardware Saab part number and Software version string.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
240h	40h	A1h	02h	COMMAND	DATA	00h	00h	00h

Here's a list of DATA values you can query (that I know of) with the COMMAND byte as 01h.

DATA	Description	How to handle reply value
00h	Some generic request (requires more research)	?
04h	Calculated load value	unit is %
05h	Coolant temperature	subtract 40, unit °C
0Bh	Manifold Air Pressure, MAP	unit hPa
0Ch	Engine RPM	divide by 4, unit 1/min
0Eh	Engine timing advance	?
0Fh	Intake air temperature	subtract 40, unit °C
10h	Mass Air Flow, MAF	divide by 100, unit g/s
14h	O2 sensor 1, Bank 1	?
15h	O2 sensor 2, Bank 1	?

When COMMAND is 1Ah, here's what you can get with the DATA byte.

DATA	Description
90h	Vehicle Identification Number, VIN
91h	Hardware Saab part number
92h	Immobilizer code
94h	Software Saab part number
95h	Software version
97h	Engine type
98h	Additional info
99h	Software date

Example messages:
240h [40 A1 02 01 10 00 00 00]
258h [C1 BF 04 41 00 00 00 00]
266h [40 A1 3F 81 00 00 00 00]
258h [80 BF 10 04 AB 00 00 00]
266h [40 A1 3F 80 00 00 00 00]
First, Mass Air Flow value is requested from the Trionic. It responds with a two row answer, to which both rows are acknowledged. The MAF value is 4ABh, which translates to 1195dec, so the actual value is 11.95 g/s.

258h - Trionic data query reply

Trionic sends these reply messages after receiving a 240h data query. There are several different ways the Trionic can respond. Below are three different ones. The ROW byte tells the row index of the reply (i.e. how many messages are still coming). It always seems to have the most significant bit, bit 7 set. Also, the first row has bit 6 set. So, a one line reply has a ROW byte of C0h, two line has ROW bytes C1h and 80h, three line has C2h, 81h, 80h, and so on. The LENGTH byte gives the length of the answer. However there's something a bit illogical about the LENGTH byte, since in the first response way detailed below, the LENGTH byte is either 03h or 04h. In the last response way LENGTH is logical, it tells the number of REPLY bytes plus two (5Ah and DATA byte). The last reply row is padded with 00h's if necessary.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
258h	ROW	BFh	LENGTH	41h	00h	00h	00h	00h
258h	ROW	BFh	DATA	VALUE1	[VALUE0]	00h	00h	00h

Example messages:
240h [40 A1 02 01 0B 00 00 00]
258h [C1 BF 03 41 00 00 00 00]
266h [40 A1 3F 81 00 00 00 00]
258h [80 BF 0B 64 00 00 00 00]
266h [40 A1 3F 80 00 00 00 00]
Asks Manifold Air Pressure, MAP, from Trionic, with a response of 64h or 100dec (100 hPa).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
258h	ROW	BFh	LENGTH	41h	DATA	VALUE1	[VALUE0]	00h

Example messages:
240h [40 A1 02 01 0C 00 00 00]
258h [C0 BF 04 41 0C 24 DC 00]
266h [40 A1 3F 80 00 00 00 00]
Asks Engine RPM from Trionic, with a response of 24DCh or 9436dec. Divide by 4 and you get 2359 RPM.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
258h	ROW	BFh	LENGTH	5Ah	DATA	REPLY	REPLY	REPLY
258h	ROW	BFh	REPLY	REPLY	REPLY	REPLY	REPLY	REPLY
258h	ROW	BFh	REPLY	REPLY	REPLY	REPLY	REPLY	REPLY

Example messages:
240h [40 A1 02 1A 90 00 00 00]
258h [C3 BF 13 5A 90 59 53 33] YS3
266h [40 A1 3F 83 00 00 00 00]
258h [82 BF 45 46 35 38 43 39] EF58C9
266h [40 A1 3F 82 00 00 00 00]
258h [81 BF 59 31 32 33 34 35] Y12345
266h [40 A1 3F 81 00 00 00 00]
258h [80 BF 36 37 00 00 00 00] 67
266h [40 A1 3F 80 00 00 00 00]
Asks Vehicle Identification Number from Trionic, with a response of YS3EF58C9Y1234567.

266h - Trionic reply acknowledgement

This message must be sent after every Trionic data query reply. Notice that bit 6 (which tells if this is the first reply row) of ROW byte is cleared in this acknowledgement message.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
266h	40h	A1h	3Fh	ROW	00h	00h	00h	00h

Example messages:
240h [40 A1 02 01 0C 00 00 00]
258h [C1 BF 04 41 00 00 00 00]
266h [40 A1 3F 81 00 00 00 00]
258h [80 BF 0C 17 58 00 00 00]
266h [40 A1 3F 80 00 00 00 00]
The first 266h response is with 81h in the ROW byte and the second with 80h.

280h - Pedals, reverse gear

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. The GEAR byte is 02h if the reverse gear selected, otherwise the byte is FFh. When the brake pedal is pressed lightly only bit 1 of the PEDAL byte will be set. Normal braking will set bits 1, 3 and 4. The BRAKE/CLUTCH bit will also be set if the clutch is pressed. The ACTIVE bit in the CRUISE byte is set when cruise control has been activated. The ?? byte indicates that the engine is running or something like that (needs closer inspection).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
280h	STATE	GEAR	PEDAL	-	CRUISE	??	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
PEDAL	-	-	KICKDOWN	BRAKE	BRAKE/CLUTCH	-	BRAKE	-
CRUISE	-	-	ACTIVE	-	-	-	-	-

Example message:
80 FF 0A 00 00 C0 82 00

Information has changed from the last message, reverse gear is not selected, brake pedal has been pressed and the engine is running (?).

290h - Steering wheel and SID buttons

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. The audio button information and SID button information are duplicated to bytes 4 and 5.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
290h	STATE	-	AUDIO	SID	AUDIO	SID	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
AUDIO	VOL-	VOL+	SRC	SEEK+	SEEK-	NXT	-	-
SID	CLR	SET	DOWN	UP	NPANEL	CLOCK+	CLOCK-	-

Example message:
80 00 40 00 40 00 00 00
Information has changed from the last message and the Volume Up audio button has been pressed.

320h - Doors, central locking and seat belts

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. Bit LOCKED is 1 if doors are locked. Bit FL reports if the front left door is open, FR reports the same from the front right door (RL and RR bits indicate the rear doors). Bit TRUNK reports if the trunk is open. The BULBS byte indicates if there are any broken light bulbs, that the SID should report (Information provided by Magnus Lirell).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
320h	STATE	DOOR	BELT	-	BULBS	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
DOOR	LOCKED	FL	FR	RL	RR	TRUNK	-	-
BELT	-	-	??	??	-	-	-	-
BULBS	-	Left park	-	-	-	-	-	-

Example message:
00 E0 00 00 00 CC 80 00
Information has not changed from the last message and both front doors are open.

328h - SID audio text

A group of three messages are sent with an interval of 1 second and if a value changes. Messages are sent with about 10 milliseconds apart. The CHANGED bit in the ROW byte will be set if information changes. The two bits ORD0 and ORD1 in the ORDER byte are for sequence numbering. A new message group starts with the NEW bit set and both ORD0 and ORD1 bits. On the second message the NEW and ORD1 bits are cleared and the third message has also the ORD0 bit cleared. So the ORDER byte will be 42, 01 and 00 for the three sequential messages. The ROW byte tells to which row of the SID the text will be displayed on. The byte can have a value of 2 or 1, but in these audio text messages the row number is always 2. The TEXT4...TEXT0 bytes are plain ASCII coded characters that will be displayed on the SID.

Note that the last message will contain only one normal character in the TEXT4 byte. The TEXT3 byte should contain a integer value that will be shown as a small number to indicate the selected radio station number. The last TEXT2...TEXT0 bytes are always zeros.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
328h	ORDER	-	ROW	TEXT4	TEXT3	TEXT2	TEXT1	TEXT0

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ORDER	-	NEW	-	-	-	-	ORD1	ORD0
ROW	CHANGED	-	-	-	-	-	ROW1	ROW0

Example message group:
42 96 02 55 31 20 4B 49 U 1 _ K I
01 96 02 53 53 20 46 4D S S _ F M
00 96 02 20 01 00 00 00 _ 1
The text "U1 KISS FM " will be displayed on the SID display and the radio station number is 1.

32Ch - ACC to SID text

This message is identical to the 32Fh message. The only difference is that this message is sent by the ACC (Automatic Climate Control).
Information provided by Magnus Lirell.
Example message group:
45 96 81 53 54 41 52 54 S T A R T
04 96 81 41 20 4D 4F 54 A _ M O T
03 96 81 4F 52 20 20 20 O R
02 96 82 46 D7 52 20 41 F Ö R _ A
01 96 82 43 43 20 20 20 C C
00 96 82 20 20 20 20 20
The text "STARTA MOTOR FÖR ACC" will be displayed on the SID display. This message should come up when press Auto + OFF on the ACC with the ignition turned ON and motor not running.

32Fh - TWICE to SID text

This SID full-screen text message is almost identical to the previous 328h SID audio text message. This 32Fh message is used by the TWICE (Theft Warning Integrated Central Electronics) to display text on the SID. The row number can be one in this message. And the ORDER byte contains an ORD2 bit so that six messages can be sent. Also the last three bytes of messages #3 and #6 (that's with ORDER bytes 03h and 06h) are 20h, not 00h like in the audio text message.
Example message group:
45 96 01 32 20 52 45 4D 2 _ R E M
04 96 01 4F 54 45 20 4B O T E _ K
03 96 01 45 59 20 20 20 E Y
02 96 02 32 20 54 52 41 2 _ T R A
01 96 02 4E 53 50 4F 4E N S P O N
00 96 02 44 52 20 20 20 D R
The texts "2 REMOTE KEY" and "2 TRANSPONDR" will be displayed on rows one and two of the SID display.

337h - SPA to SID text

This message is identical to the previous 32Fh message. The only difference is that this message is sent by the SPA (Saab Park Assist).
Information provided by Magnus Lirell.
Example message group:

42 96 01 50 41 52 4B 2E P A R K .
01 96 01 48 4A 15 4C 50 H J Ä L P
00 96 01 20 20 00 00 00 _ _
The text "PARK.HJÄLP" will be displayed on row one of the SID display.

348h - SID audio text control

Message is sent with an interval of 1 second. The normal value (i.e. don't display text) of byte STATUS is FFh. When a 328h SID audio text message group is sent, the STATUS byte is changed to 04h and immediately after that to 05h for the duration of the messages. After the 328h messages, the status byte returns to FFh.
Notice that the control messages use a 20h increment relative to the actual text messages in their IDs. Bytes 0 and 3 probably have something to do with priority (which text is displayed if several texts should be displayed at the same time).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
348h	11	02	STATUS	19	00	00	00	00

Example message:
11 02 FF 19 00 00 00 00
SID should not display the text provided with message 328h.

34Ch - ACC to SID text control

Message is sent with an interval of 1 second. The normal value of byte STATUS is FFh. When a 32Ch ACC to SID text message group is sent, the STATUS byte changes between 03h and 05h for the duration of the messages. After the 32Ch messages, the status byte returns to FFh. Byte UNKNOWN1 seems to stay at 00h and UNKNOWN2 byte at 23h.
Information provided by Magnus Lirell.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
34Ch	18	UNKNOWN1	STATUS	UNKNOWN2	00	00	00	00

Example message:
18 00 05 23 00 00 00 00
.
.
18 00 FF 23 00 00 00 00
First line: requesting SID to display text provided with message 32Ch. Second line: SID should not display ACC text.

34Fh - TWICE to SID text control

Message is sent with an interval of 1 second. The normal value of byte STATUS is FFh. When a 32Fh TWICE to SID text message group is sent, the STATUS byte is changed to 04h and immediately after that to 05h for the duration of the messages. After the 32Fh messages, the status byte returns to FFh.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
34Fh	1B	00	STATUS	2D	00	00	00	00

Example message:
1B 00 05 2D 00 00 00 00
Requesting SID to display text provided with message 32Fh.

357h - SPA to SID text control

Message is sent with an interval of 1 second. The normal value of byte STATUS is FFh. When a 337h SPA to SID text message group is sent, the STATUS byte is changed to 04h and immediately after that to 05h for the duration of the messages. After the 337h messages, the status byte returns to FFh.

Byte UNKNOWN1 is 00h when STATUS is FFh and 01h when STATUS is 05h. UNKNOWN2 byte also changes from 08h (STATUS is FFh) to 12h (STATUS is 05h).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
357h	1F	UNKNOWN1	STATUS	UNKNOWN2	00	00	00	00

Example message:
1F 01 05 12 00 00 00 00
.
.
1F 00 05 08 00 00 00 00
First line: requesting SID to display text provided with message 337h. Second line: SID should not display SPA text.

368h - SID text priority

Message is sent with an interval of 1 second and if a value changes. Message is sent in a group of three messages. The first byte, ROW, has the values 0h, 1h, 2h corresponding to lines on the SID. The 0h line needs more research, since as you know the SID has only two lines of text it can display. If the PRIORITY byte is FFh, there is nothing shown on the SID. However, when a control unit sends text messages (e.g. messages 337h and 357h), its requested priority is shown on the PRIORITY byte. If another control unit wants to override the currently shown text message, it needs to send messages with a higher priority, i.e. with a lower PRIORITY byte value.
Information provided by David Goncalves.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
368h	ROW	PRIORITY	-	-	-	-	-	-

Example message group:
00 FF 00 00 00 00 00 00
01 FF 00 00 00 00 00 00
02 19 00 00 00 00 00 00
SID shows text on row 2 with priority 19h. No text on row 0 and 1.

380h - Audio RDS status

Message is sent with an interval of 1 second and if a value changes. This message tells the RDS status of the Audio system.
Information provided by David Goncalves.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
380h	STATE	UNKNOWN	STATUS	-	-	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
STATUS	RDS activ.	No TP signal	??	TP activ.	PTY activ.	??	??	No RDS signal

Example message group:
80 00 D0 00 00 00 00 00
RDS has been activated and the radio is receiving a RDS signal, TP has been activated but no TP signal received.

3B0h - Head lights

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. LIGHT byte indicates the park and daylight head-lights with the PARK and DAY bits. When ignition signal is off, the OFF bit is set in the LIGHT byte. The ?? byte is 20h when ignition signal is off and B8h when the signal is on.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
3B0h	STATE	LIGHT	-	??	-	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
LIGHT	-	1	1	1	OFF	1	PARK	DAY
??	-	-	-	-	-	-	-	-

Example message:
00 76 74 B8 00 00 00 00
Information has not changed from the last message and the head-lights are in park mode.

3C0h - CD Changer control

Message is sent with an interval of 1 second. The most significant bit of the first byte, CHANGED, indicates when information has changed in the message. Byte 1, COMMAND, specifies what CD Changer controlling button has been pressed. The PARAM byte is used by "Change to CD" command to inform which CD was selected.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
3C0h	CHANGED	COMMAND	PARAM	-	-	-	-	-

COMMAND	Description
59h	Next (NXT button)
35h	Seek next (Seek+)
36h	Seek previous (Seek-)
68h	Change to CD according to PARAM byte
B0h	Audio mute (off?)
B1h	Audio mute (on?)
76h	Random play (long press of CD/RDM button)

Example message:
80 68 02 00 00 00 00 00
Information has changed since last message and CD 2 has been selected to be played.

3C8h - CD Changer information

Message is sent with an interval of 1 second. The most significant bit of the first byte, CHANGED, indicates when information has changed in the message. MAGAZINE byte informs which disc slots are occupied and which are empty. The lower nibble of DISC byte (DISC3...DISC0) tells which CD disc is being played in BCD coding. TRACK byte tells the track number being played in BCD coding. The bytes MIN and SEC tell the play minutes and seconds of the current track, also in BCD coding. When there's no disc in play, the TRACK, MIN and SEC bytes are FFh. The STAT0...STAT2 bits indicate the status of the CD changer. If the STAT value is 0h, the CD changer is inactive (not spinning). Value 3h means power-up / spin-up, but not ready yet. And value 4h means that everything is OK - playing disc. SECURITY byte informs the Audio head unit if the CD Changer has not been married or is married to a different car. If the byte is D0h, everything is alright. Values 50h and 10h mean that the CD Changer needs to be married to the car (text "CDC CODE" will be displayed on SID). Value 90h means the CD

Changer has been married to a different car and the VIN doesn't correspond to the one stored in the CD Changer (text "CDC LOCKED" will be display on SID).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
3C8h	CHANGED	-	MAGAZINE	DISC	TRACK	MIN	SEC	SECURITY

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CHANGED	CHANGED	-	??	-	-	-	-	-
MAGAZINE	-	-	CD6	CD5	CD4	CD3	CD2	CD1
DISC	-	STAT2	STAT1	STAT0	DISC3	DISC2	DISC1	DISC0

Example message:
20 00 17 45 33 01 53 D0
Information has not changed since last message and CD slots 4 and 6 are empty. The current track, number 33, is being played from disc 5 and has lasted 1 minute and 53 seconds. The CD Changer has been married and is operating in the married car.

3E0h - Automatic Gearbox

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. The GEAR byte indicates in what state the gearbox is. Value 05h means Forward, value 03h Neutral and value 02h Reverse. The GEAR SHIFT byte tells in what position the gear shift is. Value 01h means Park, value 02h Reverse, value 03h Neutral, value 04h Drive, value 08h "limit to 4", value 07h "limit to 3" and value 06h "low speed gear".
Information provided by Magnus Lirell.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
3E0h	STATE	GEARBOX	GEAR SHIFT	MODE1	Adaptation?	MODE2	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
GEARBOX	-	-	-	-	-	-	-	-
GEAR SHIFT	-	-	-	-	POS	POS	POS	POS
MODE2	-	-	-	-	-	-	SPORT/WINTER	SPORT/WINTER

Example message:
00 00 00 00 00 00 00 00
Example text...

410h - Light dimmer and light sensor

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. DIMM1 and DIMM0 form a 16-bit integer for instrumentation panel dimmer position. Integer value is between about 4600h and FD00h. LIGHT1 and LIGHT0 form also a 16-bit integer for the SID light sensor. Typical values range from about 1800h to 2C00h. The NPANEL byte indicates if the Night Panel function is on (Information provided by Magnus Lirell).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
410h	STATE	DIMM1	DIMM0	LIGHT1	LIGHT0	NPANEL	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-

Example message:
00 46 41 18 04 00 00 00
Information has not changed from the last message and dimmer is at minimum position.

430h - SID beep request

This message is sent when the SID is requested to beep (what unit sends this request is still unknown).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
430h	80	04	00	00	00	00	00	00

Example message:
80 04 00 00 00 00 00 00
Request for the SID to beep.

439h - SPA distance

Message is sent with an interval of ??. The DIST byte indicates the distance to objects behind the car according to the SPA (Saab Park Assist).
Information provided by Magnus Lirell.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
439h	STATE	-	DIST	-	-	-	-	-

Example message:
00 00 00 00 00 00 00 00
Example text...

460h - Engine rpm and speed

Message is sent with an interval of 100 milliseconds. The ENGOFF bit of the first ENG byte indicates if the engine is off (the bit is 0 when engine is on). RPM1 and RPM0 form a 16-bit integer for engine speed (in RPM). Integer value is between 0000h and 1B00h. SPD1 and SPD0 form also a 16-bit integer for the car speed. The unit for the value is 100 meters per hour, so in order to get speed in kilometers per hour you must divide the integer value by ten.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
460h	ENG	RPM1	RPM0	SPD1	SPD0	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ENG	ENGOFF	-	-	-	-	-	-	-

Example message:

00 03 9C 00 2A 00 00 00
Engine is running, engine speed is 924 RPM and car speed 4,2 km/h.

4A0h - Steering wheel, Vehicle Identification Number

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. The WIPER byte indicates the position of the windshield wiper control. The least significant bit of the WIPER byte indicates that the park lights are on. The LEFT and RIGHT bits of the SIGNAL byte indicate if a signal light is on. So when the signal light blinks, so does the bits. The REARFOG bit indicates if the rear fog light is on. Vehicle Identification Number, VIN, is transmitted on bytes 4...6 (VIN2...VIN0). To be more precise, the car serial number part of the VIN is transmitted. It is coded so that the last number of the VIN, which probably is a checksum, is moved to the upper part of the first (VIN2) byte. For example if your VIN is YS3DC55C612123456, your serial number part is that in bold, 123456. The last number, 6, is the checksum number. Moving it to the upper part of the VIN2 byte, the VIN bytes would be VIN2 = 61h, VIN1 = 23h, VIN0 = 45h.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
4A0h	STATE	WIPER	SIGNAL	-	VIN2	VIN1	VIN0	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
WIPER	-	DRIZZLE	NORMAL	BACK	-	-	-	PARK
SIGNAL	-	-	-	-	-	LEFT	RIGHT	REARFOG

Example message:
80 41 00 00 61 23 45 00
Information has changed from the last message and the windshield wiper is in drizzle mode. The Vehicle Identification Number's serial number part is 123456.

520h - ACC, inside temperature

Message is sent with an interval of 1 second. Inside temperature is reported with the 8-bit byte TEMP. In order to get the correct temperature, the value must be subtracted with 40. This is done to encode negative temperatures. So a value of 58 (3Ah) is in fact +18 degrees Celsius and on the other hand a value of 29 (1Dh) would give an temperature of -11 degrees Celsius. The REARHEAT bit in the ACC byte indicates if the rear window heater is on.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
520h	-	ACC	-	-	-	TEMP	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ACC	-	REARHEAT	-	-	-	-	-	-

Example message:
00 00 00 00 00 3A 00 00
Inside temperature is 18 degrees Celsius.

530h - ACC

Message is sent with an interval of 1 second. The ACCON bit in the ACC byte tells if the ACC (Automatic Climate Control) is turned on. When the bit is set, the ACC is on and when the bit is cleared the ACC has been turned off. ACPRESS byte shows the A/C Pressure (unit is bar).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
530h	-	ACC	-	ACPRESS	-	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ACC	-	-	-	-	-	ACCON	-	-

Example message:
00 00 00 00 00 00 00 00
Text here.

590h - Position Seat Memory

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message.
Information provided by Magnus Lirell.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
590h	STATE	MEM	-	-	-	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
MEM	-	-	-	-	-	STORE	MEM1	MEM0

Example message:
80 05 00 0 00 00 00 00
Store current seat and mirror positions into memory number 1.

5C0h - Coolant temperature, air pressure

Message is sent with an interval of 1 second. Temperature is reported with a 8-bit byte. In order to get the correct coolant temperature, the value must be subtracted with 40. This is done to encode negative temperatures. So a value of 58 (3Ah) is in fact +18 degrees Celsius and on the other hand a value of 29 (1Dh) would give an temperature of -11 degrees Celsius. The 16-bit value combined from PRES1 and PRES0 gives the Ambient air pressure in hehtopascals [hPa].

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
5C0h	-	COOLANT	PRES1	PRES0	-	-	-	-

Example message:
00 6A 6A 03 F9 00 00 00
Coolant temperature is 66 degrees Celsius and air pressure is 1017 hPa.

630h - Fuel usage

Message is sent with an interval of 1 second. Fuel usage since last start is reported with an 16-bit integer value (bytes FUEL1...FUEL0). The FUEL1 byte is the most significant byte. The unit is milliliters of fuel (used since start). The amount of fuel left in the tank is indicated by 16-bit integer values AVGTANK and TANK. The TANK value seems to be a raw measurement value and the AVGTANK an averaged value of the raw measurements. A full tank is about 02A0h and a soon empty tank below 0040h. Dividing the value by 10 could give the available fuel amount in liters (this is a guess). The first byte VALID at least informs if the tank values should be disregarded. For example during a engine start the two least significant bits (TANKVALID) are set, meaning that the AVGTANK and TANK values are not correct (they seem to be zeros).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
630h	VALID	FUEL1	FUEL0	AVGTANK1	AVGTANK0	TANK1	TANK0	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
VALID	-	-	-	-	??	??	TANKVALID	TANKVALID

Example message:
00 00 00 00 00 00 00 00
Example text here.

640h - Mileage

Message is sent with an interval of 1 second. Mileage is reported with an 24-bit integer value (bytes MIL2...MIL0). The unit is 10 meters, so in order to get the mileage in kilometers the value must be divided by a hundred.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
640h	-	-	MIL2	MIL1	MIL0	-	-	-

Example message:
00 00 CA A1 DC 00 00 00
Traveled mileage is 132 797,08 kilometers.

6A1h - Audio head unit

The Audio head unit uses messages with ID 6A1h to communicate with the CD changer. Byte 3 seems to be the command for the CD changer. Value 13h is power-up / spin-up, value 12h play and 18h power-down / spin-down.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
6A1h	21	00	00	COMMAND	01	??	??	-

Example message:
Example description.

6A2h - CD changer

The CD changer uses messages with ID 6A2h to communicate with the Audio head unit.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
6A2h	-	-	-	-	-	-	-	-

Example message:
Example description.

720h - RDS time

Message is sent with an interval of 1 second. The RDSON byte indicates if a RDS signal has been detected.
Information provided by Magnus Lirell.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
720h	-	SEC	MIN	HOUR	DAY	MONTH	RDSON	-

Example message:
00 01 0E 13 0F 02 E1 00
The day is the 16th of February and the time is 19:14:01.

730h – Clock

Message is sent with an interval of 8 seconds. The ENGCOLD byte indicates how many minutes has passed since the ignition was turned off.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
730h	-	SEC	MIN	HOUR	DAY	MONTH	ENGCOLD	-

Example message:
00 01 0E 13 0F 02 00 00
The day is the 16th of February and the time is 19:14:01.

740h 750h - Security

Messages are sent when the car key is turned to ignition on. This leads me to believe that these two messages are somehow connected to the anti-theft system. The CODE bytes seem to have random numbers in them that change after a successful "handshake".
The 750h message is sent first and after about 20 ms the 740h message is sent. The "reply" can contain zeroed CODE bytes and the STATUS byte as FFh. In that case after about 190 ms the same 750h message is repeated. The second 740h "reply" seems to be always ok, since it has something in the CODE bytes and the STATUS byte is 55h.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
740h	CODE4?	CODE3?	CODE2?	CODE1?	CODE0?	STATUS	00	00
750h	CODE4?	CODE3?	CODE2?	CODE1?	CODE0?	00	00	00

Example messages:
750 4F C3 11 2A FA 00 00 00
740 FE 91 EA 5C CD 55 00 00

750 7E EE 9B FF 6E 00 00 00
740 00 00 00 00 00 FF 00 00
750 7E EE 9B FF 6E 00 00 00
740 58 67 92 77 4F 55 00 00

750 21 E9 FF F8 4A 00 00 00
740 00 00 00 00 00 FF 00 00
750 21 E9 FF F8 4A 00 00 00
740 2C DB 4A 7E 49 55 00 00

750 14 20 42 C7 C3 00 00 00
740 00 00 00 00 00 FF 00 00
750 14 20 42 C7 C3 00 00 00
740 04 B9 29 D2 03 55 00 00
In this example the first column indicates the message ID. These are real-world examples of the messages. If someone figures out how these messages work, it could mean that the anti-theft mechanism could be bypassed.

7A0h - Outside temperature

Message is sent with an interval of 1 second. There are two temperature values: AVGTEMP and RAWTEMP. AVGTEMP is the averaged outside temperature with a reading accuracy of 1 degree Celsius. The other temperature, RAWTEMP, shows the raw temperature readings. It has a reading accuracy of 0.5 degrees Celsius. Temperatures are reported as 16-bit values. In order to get the correct temperature, the value must be divided by 10 and subtracted with 40. In other words, a value of 0235h is first converted to decimal, 565, then divided by 10, so that we get 56.5 and finally subtract 40 to get 16.5 degrees Celsius. The STATUS byte most likely tells if the temperature readings are inside or outside values. The SID seems to display the inside temperature for some time. During this time the STATUS byte is 30h. After the a while the STATUS bytes changes to first to 80h and then right after to 00h. This probably means that the temperature readings are outside values.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
7A0h	STATUS	AVGTEMP1	AVGTEMP0	RAWTEMP1	RAWTEMP0	-	-	-

Example message:
00 01 72 01 6D 00 00 00
Average outside temperature is -4.0 degrees Celsius and raw outside temperature reading is -3.5 degrees Celsius.

SAAB P-bus communication

Courtesy of Tomili and General Failure

The P-Bus is an internal bus (Powertrain Bus) that connects together ECU, ABS control unit, TCS control unit and DICE.

The I bus follows the ISO 11898-2 standard which is the high speed variant, it implements a maximum communication speed of 500 kbit/s. Messages on a CAN bus are sent within CAN frames that include an identifier number, number of data bytes, the actual data (called the payload) and checksum (to verify it the data was transferred correctly). There are two CAN frame formats: the basic frame (referred to as CAN2.0A) and the extended frame (referred to as CAN2.0B). The basic frame has a 11-bit identifier field, when the extended frame has a 29-bit identifier which makes it possible to extend the number of message types that can be sent over the bus.
The P bus uses two signal (just like any other CAN bus connection) which are called P+ and P-. These refer to CAN High and CAN Low signals in general CAN terminology.
Details on P bus communication:

Frame type:	CAN 2.0A (11-bit identifiers)
Bus speed:	500 kbit/s
Timing register settings:	BTR0 0x00, BTR1 0x1C

1A0h - Engine information
Message is sent with an interval of 10 milliseconds. The ENGSTOPPED bit in the STATUS byte indicates if the engine is running or stopped. Engine RPM is shown as a 16-bit value (RPM1 and RPM0). MAP shows the Manifold Absolute Pressure in kilopascals. THROTTLE byte is the throttle pedal position, with a value range of 0...100 percent (0...64hex). MAF is the reading from Mass Air Flow sensor.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
1A0h	STATUS	RPM1	RPM0	MAP		THROTTLE	-	MAF

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATUS	CHANGED	-	-	-	ENGSTOPPED	-	-	-

2F0h - Vehicle speed
Message is sent with an interval of 20 milliseconds. Vehicle speed is indicated by the 16-bit value (bytes SPEED1 and SPEED0). You will need to divide the value by 10 to get the right scaling, kilometers per hour. The byte 3 seems to be 80h all the time.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
2F0h	-	SPEED1	SPEED0	?	-	-	-	-

280h - Pedals, reverse gear
Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. The GEAR byte is 02h if the reverse gear selected, otherwise the byte is FFh. When the brake pedal is pressed lightly only bit 1 of the PEDAL byte will be set. Normal braking will set bits 1, 3 and 4. The BRAKE/CLUTCH bit will also be set if the clutch is pressed. The ACTIVE bit in the CRUISE byte is set when cruise control has been activated. The ?? byte indicates that the engine is running or something like that (needs closer inspection).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
280	STATE	GEAR	PEDAL	-	CRUISE	??	-	-

h								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
PEDAL	-	-	KICKDOWN	BRAKE	BRAKE/CLUTCH	-	BRAKE	-
CRUISE	-	-	ACTIVE	-	-	-	-	-

Example message:
80 FF 0A 00 00 C0 82 00
Information has changed from the last message, reverse gear is not selected, brake pedal has been pressed and the engine is running (?).

338h - Trionic to SID text
A group of three messages are sent with an interval of 1 second and if a value changes. Messages are sent with about 10 milliseconds apart. The CHANGED bit in the ROW byte will be set if information changes. The two bits ORD0 and ORD1 in the ORDER byte are for sequence numbering. A new message group starts with the NEW bit set and both ORD0 and ORD1 bits. On the second message the NEW and ORD1 bits are cleared and the third message has also the ORD0 bit cleared. So the ORDER byte will be 42, 01 and 00 for the three sequential messages. The ROW byte tells to which row of the SID the text will be displayed on. The byte can have a value of 2 or 1, but in these audio text messages the row number is always 2. The TEXT4...TEXT0 bytes are plain ASCII coded characters that will be displayed on the SID.
Note that the last message will contain only one normal character in the TEXT4 byte. The TEXT3 byte should contain a integer value that will be shown as a small number to indicate the selected radio station number. The last TEXT2...TEXT0 bytes are always zeros.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
338h	ORDER	-	ROW	TEXT4	TEXT3	TEXT2	TEXT1	TEXT0

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ORDER	-	NEW	-	-	-	-	ORD1	ORD0
ROW	CHANGED	-	-	-	-	-	ROW1	ROW0

Example message group:
42 96 02 55 31 20 4B 49 U 1 _ K I
01 96 02 53 53 20 46 4D S S _ F M
00 96 02 20 01 00 00 00 _ 1
The text "U1 KISS FM " will be displayed on the SID display and the radio station number is 1.

358h - Trionic to SID text control
Message is sent with an interval of 1 second. The normal value of byte STATUS is FFh. When a 338h Trionic to SID text message group is sent, the STATUS byte is changed to 04h and immediately after that to 05h for the duration of the messages. After the 338h messages, the status byte returns to FFh. Byte UNKNOWN1 is 00h when STATUS is FFh and 01h when STATUS is 05h. UNKNOWN2 byte also changes from 08h (STATUS is FFh) to 12h (STATUS is 05h).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
358h	1F	UNKNOWN1	STATUS	UNKNOWN2	00	00	00	00

Example message:
1F 01 05 12 00 00 00 00
.
.
1F 00 05 08 00 00 00 00

First line: requesting SID to display text provided with message 338h. Second line: SID should not display Trionic text.

370h - Mileage

Message is sent with an interval of 100 milliseconds. The 24-bit value (comprised of MIL2, MIL1 and MIL0) shows the length that has been travelled since the engine was started. You will need to divide the value by 100 to get right scaling (the value's unit is meters). The value 000069h seems to tell that you haven't moved yet.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
370h	STATUS	MIL2	MIL1	MIL0	-	-	-	-

Example message:
00 02 D1 50 00 00 00 00
You have travelled 1 846,56 meters.

3A0h - Vehicle speed (MIU?)

Message is sent with an interval of 55 milliseconds. Vehicle speed is indicated by the 16-bit value SPEED (bytes SPEED1 and SPEED0). You must divide the 16-bit value by 10 to get the value in kilometers per hour.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
3A0h	-	-	-	SPEED1	SPEED0	-	-	-

3B0h - Head lights

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. LIGHT byte indicates the park and daylight head-lights with the PARK and DAY bits. When ignition signal is off, the OFF bit is set in the LIGHT byte. The ?? byte is 20h when ignition signal is off and B8h when the signal is on.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
3B0h	STATE	LIGHT	-	??	-	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
LIGHT	-	1	1	1	OFF	1	PARK	DAY
??	-	-	-	-	-	-	-	-

Example message:
00 76 74 B8 00 00 00 00
Information has not changed from the last message and the head-lights are in park mode.

3E0h - Automatic Gearbox

Message is sent with an interval of 1 second and if a value changes. The most significant bit of the first byte (byte 0) is set if information has changed from the last message. The GEAR byte indicates in what state the gearbox is. Value 05h means Forward, value 03h Neutral and value 02h Reverse. The GEAR SHIFT byte tells in what position the gear shift is. Value 01h means Park, value 02h Reverse, value 03h Neutral, value 04h Drive, value 08h "limit to 4", value 07h "limit to 3" and value 06h "low speed gear".
Information provided by Magnus Lirell.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
3E0h	STATE	GEARBOX	GEAR SHIFT	MODE1	Adaptation?	MODE 2	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATE	CHANGED	-	-	-	-	-	-	-
GEARBOX	-	-	-	-	-	-	-	-
GEAR SHIFT	-	-	-	-	POS	POS	POS	POS
MODE2	-	-	-	-	-	-	SPORT/WINTER	SPORT/WINTER

Example message:
00 00 00 00 00 00 00 00
Example text...

530h - ACC

Message is sent with an interval of 1 second. The ACCON bit in the ACC byte tells if the ACC (Automatic Climate Control) is turned on. When the bit is set, the ACC is on and when the bit is cleared the ACC has been turned off. The ACREQ bit indicates A/C request from DICE (not sure, a guess at the moment...). ACPRESS byte shows the A/C Pressure (unit is bar).

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
530h	-	ACC	-	ACPRESS	-	-	-	-

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ACC	-	-	-	-	ACREQ?	ACCON	-	-

Example message:
00 00 00 00 00 00 00 00
Text here.

5C0h - Coolant temperature, air pressure
Message is sent with an interval of 1 second. Temperature is reported with a 8-bit byte. In order to get the correct coolant temperature, the value must be subtracted with 40. This is done to encode negative temperatures. So a value of 58 (3Ah) is in fact +18 degrees Celsius and on the other hand a value of 29 (1Dh) would give an temperature of -11 degrees Celsius. The 16-bit value combined from PRES1 and PRES0 gives the Ambient air pressure in hehtopascals [hPA].

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
5C0h	-	COOLANT	COOLANT	PRES1	PRES0	-	-	-

Example message:
00 6A 6A 03 F9 00 00 00
Coolant temperature is 66 degrees Celsius and air pressure is 1017 hPa.

631h - ??
Message is sent with an interval of 1 second.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
631h	-	-	-	-	-	-	-	-

6B1h - ??
Message is sent with an interval of 1 second.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
6B1h	-	-	-	-	-	-	-	-

6B2h - ??
Message is sent with an interval of 1 second.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
6B2h	-	-	-	-	-	-	-	-

740h 750h - Security
Messages are sent when the car key is turned to ignition on. This leads me to believe that these two messages are somehow connected to the anti-theft system. The CODE bytes seem to have random numbers in them that change after a successful "handshake".
The 750h message is sent first and after about 20 ms the 740h message is sent. The "reply" can contain zeroed CODE bytes and the STATUS byte as FFh. In that case after about 190 ms the same 750h message is repeated. The second 740h "reply" seems to be always ok, since it has something in the CODE bytes and the STATUS byte is 55h.

ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
740h	CODE4?	CODE3?	CODE2?	CODE1?	CODE0?	STATUS	00	00
750h	CODE4?	CODE3?	CODE2?	CODE1?	CODE0?	00	00	00

Example messages:
750 4F C3 11 2A FA 00 00 00
740 FE 91 EA 5C CD 55 00 00

750 7E EE 9B FF 6E 00 00 00
740 00 00 00 00 00 FF 00 00
750 7E EE 9B FF 6E 00 00 00
740 58 67 92 77 4F 55 00 00

750 21 E9 FF F8 4A 00 00 00
740 00 00 00 00 00 FF 00 00
750 21 E9 FF F8 4A 00 00 00

740 2C DB 4A 7E 49 55 00 00

750 14 20 42 C7 C3 00 00 00
740 00 00 00 00 00 FF 00 00
750 14 20 42 C7 C3 00 00 00
740 04 B9 29 D2 03 55 00 00
In this example the first column indicates the message ID. These are real-world examples of the messages. If someone figures out how these messages work, it could mean that the anti-theft mechanism could be bypassed.

Common mistakes and FAQ

General

This chapter will describe some frequently made mistakes in handling the Trionic.

Ignition advance

When altering the ignition tables keep in mind that more than 35° advance from TDC is not good.

Ignition retard

When altering the ignition tables keep in mind that more than 5° retarding from TDC is not good.

FAQ

Question: At what AFR should I try to keep the engine?
Answer: Try to keep the AFR between 10.8 and 12.5 at WOT.

Question: At what EGT should I try to keep the engine?
Answer: Try to keep the exhaust gas temperature below 950°C.

Question: What is the approximate ignition advance at wide open throttle?
Answer: Try to keep the ignition advance at approximately 10° BTDC

Question: How can I determine what the maximum boost request should be for my turbo?
Answer: Check that the boost request level fits somehow to compressor map:
http://www.squirrelpf.com/turbocalc/index.php
In addition you can read appendix IV.

Question: What is a good intake air temperature?
Answer: Upton 60°C is good enough. If temperatures rise above 60°C consider replacing your stock intercooler with an aluminium, cross-flow type. These are available from Speedparts, Abbott, ETS and others.

Tools

T7Suite

T7Suite has the following functions:

- Checksum verification and correction
- Software ID adjustment
- Immobilizer code adjustment
- File comment adjustment
- Box number adjustment
- Extraction of symbol table
- Map visualization
- Compare maps in binary to another binary
- Move maps from one binary to another
- Downloading flash content from ECU through Canbus
- Flashing ECU through Canbus
- Real-time tuning (work in progress…)

For usage of this tool please refer to its user manual.

BD32

BD32.exe is the tool used to interface with the ECU through a BDM interface. It is DOS based and will run normally on Win95/98/Me. Ideally the user would boot into a DOS environment to use the tool. There is a windows version available but the author has no experience with that specific tool.

IDA Pro

IDAPro stands for Interactive DisAssembler Professional. It enables the user to disassemble binary files to its original source code. IDAPro is commercial software, not freeware.

Example of how to use IDA Pro with a **Trionic 7** box.
Open binary/raw file
Set processor to Motorola series: 68330
Check the 'Create RAM section', start address 0xF00000, size 0xFFFF.
Go to address ROM:00000000 in 'IDA View-A' and hit D-key three times to get dc.l $FFFFEFFC
Go to the next address ROM:00000004 and hit D-key three times to get reset vector address (this varies from binary to binary)
You should get e.g. dc.l unk_5169A or something like that, double-click the unk_5169A text
Your now in the place where the code execution starts, press C to disassemble
Now, from the menu select Options -> General, go to Analysis tab and press 'Kernel options1' button
Check 'Make final analysis pass' and hit OK
Press 'Reanalyze program' button and wait a while (this really takes some time, a minute or so)

Hex editor

Hexworkshop (or UltraEdit) is a tool that comes in handy often. It can be used to view, search and modify the raw binary file.

References

Web references

- ECUproject initiative [www.ecuprojct.com Steve Hayes and friends]
- Xendus [www.xendus.se General Failure]
- SaabCentral [www.saabcentral.com]
- Motorola datasheet on MC68332
- Trionic Wiki pages [http://en.wikipedia.org/wiki/Trionic]
- T7Suite homepage [http://trionic.mobixs.eu]
- Townsendimports [www.townsendimports.com]
- BDM Software [http://www.xendus.se/bdm/bd32-122.zip]
- Ion sensing for knock detection [http://www.fs.isy.liu.se/~larer/Projects/main.html]
- Turbo compressor maps [http://www.automotivearticles.com/Turbo_Selection.shtml]
- Saab9000.com [http://www.saab9000.com]
- JKBPower forum [http://jkbpower.egetforum.se/forum/index.php]

Appendix I : Symbol list

This appendix will give a short description of the most important maps in Trionic 7. To give a list of all symbols would be kind of stupid because there are approximately 4000 (!!!) symbols in a Trionic 7 binary.

AirCtrlCal.m_MaxAirTab
Airmass value from controller where area map has reached max-area and there is no point to increase the I-part. Resolution is 1 mg/c.

AirCtrlCal.m_MaxAirE85Ta (if running on E85)
Same as above for E85

BoostCal.I_LimTab
Load limit tab. to enable the I Part of boost regulator. If the load request from Airmass master is above this value plus the hysteresis is the I Part enabled and the throttle closed loop is disabled. If the load request from Airmass master is below this value is the I Part disabled and the throttle is allowed to run in closed loop.

BoostCal.P_LimTab
Load limit tab. to enable the P Part of boost regulator. If the load request from Airmass master is above this value plus the hysteresis is the P Part enabled. If the load request from Airmass master is below this value is the P Part disabled.

BoostCal.RegMap
Main constant matrix. Resolution is 0.1 %.

BstKnkCal.MaxAirmass (divide by 3,1 for approx torque, ignition, airtemp etc affect this!)
Map for max allowed Airmass for manual gearbox, m_nHigh. Resolution is 1 mg/c.

Knock limitation
Knock control first retards the ignition timing for each cylinder. If the mean value for the ignition retardation for all the cylinder exceeds a certain value, fuel enrichment will take place.
If the mean value for ignition retardation increases further, the maximum permissible air mass/combustion will be reduced with the values in the BstKnkCal.MaxAirmass maps. X axis represents degree of ignition retard. This reduction takes place continuously as the ignition retardation increases. The value constitutes the maximum air mass/combustion value permitted by knock control.

Note
Knock control on modern engines is not a safety function but a normal function. Consequently, it is considered normal when knock control reduces engine torque in certain cases. The engine knock control increases for e.g. high intake air temperatures or high coolant temperatures. Further influencing factors are driving at high altitudes and low octane fuel.
Certain engine variants require petrol with an octane rating of 98 RON in order to provide the specified engine torque/power.

BstKnkCal.MaxAirmassAu
Map for max allowed Airmass for automatic gearbox, m_nHigh. Resolution is 1 mg/c.
See text above

FCutCal.m_AirInletLimit
If the "MAF.m_AirInletFuel" is higher than this limit during m_AirInletTime will the fuelcut be activated (pressure guard).

IgnE85Cal.fi_AbsMap (if you want to change the ignition)
Ignition map for E85 fuel. Resolution is 0.1 °.

IgnNormCal.Map (if you want to change the ignition)
Normal ignition map. Resolution is 0.1 °.

MapChkCal.CheckSum (automatically updated in between every map change with T7suite!)

MaxVehicCal.v_MaxSpeed (max vehicle speed)

PedalMapCal.m_RequestMap
Requested Airmass from the driver as a function of rpm and accelerator pedal position. Resolution is 1 mg/c.

TorqueCal.M_ManGearLim
Maximum engine torque limit for each gear in the manual gearbox. Resolution is 1 Nm.
Manual gearbox, engine torque limitation

The maximum engine torque must be limited at low gear ratios for reasons of strength.
The control module calculates the engaged gear by comparing engine speed with vehicle speed.
If the gear ratio corresponds with gear 1 or R, the control module reads bus information "Reverse gear selected" from DICE to distinguish between them.
Engine torque is limited to:
Gear 1 or R is limited to 230 Nm on engine alternative B205E/B235E. Engine torque is limited to 350 Nm in other gears.
On engine alternative B235E, gear 5 is gradually limited if the engine speed is below 2400 rpm. This is to avoid vibration.
On engine alternative B235R, gear 1 is limited to 280 Nm and reverse gear, R, is limited to 230 Nm.
Torque limitation in gears 2-5 is 380 Nm.
The engine torque value is converted to mg air/combustion and constitutes the maximum air mass/combustion allowed by the gearbox.

TorqueCal.m_AirTorqMap This is where all torque limiters take their data from and therefore needs to be "fooled" if you are running 400nm+ or an automatic. 4sp automatic need the last row to be max 330nm, 5sp 350nm, as this is what the gearbox ECU is requesting as a max limit.
Data-matrix for nominal Airmass. Engine speed and torque are used as support points. The value in the matrix + friction Airmass (idle Airmass) will create the pointed torque at the pointed engine speed. Resolution is 1 mg/c.
axis to the above map: **TorqueCal.m_AirXSP**

TorqueCal.M_EngMaxTab
Data-table for maximum engine output torque for manual cars. Resolution is 1 Nm.

TorqueCal.M_EngMaxAutTab
Data-table for maximum engine output torque for automatic cars. Resolution is 1 Nm.
TCM, engine torque limitation

The maximum engine torque must be limited in gear R, 1 and 2 for reasons of strength. TCM will send continuous bus information specifying the maximum permissible engine torque. The maximum permitted engine torque is also limited during gear changing. TCM sends maximum permissible engine speed:

Reverse	*270 Nm*
1st gear	*330 Nm*
2nd gear	*330 Nm*
3rd gear	*330 Nm*
4th gear	*350 Nm*
5th gear	*350 Nm*

TCM then sends maximum permissible engine torque to protect the gearbox.

Trionic T7 converts Nm to mg air/combustion with help of the Airtorque map. The value constitutes the maximum air mass/combustion permitted by the automatic transmission and it is therefore we currently need to trick T7 to think more Airmass is still the same Nm.

TorqueCal.M_5GearLimTab
Data-table for maximum engine output torque for manual cars on fifth gear. Resolution is 1 Nm.

TorqueCal.M_EngMaxE85Tab (if running on E85)
Data-table for maximum engine output torque when running on E85. Resolution is 1 Nm.

TorqueCal.m_PedYSP
Air mass support points for (Calc) X_AccPedalMap. Resolution is 1 mg/combustion.

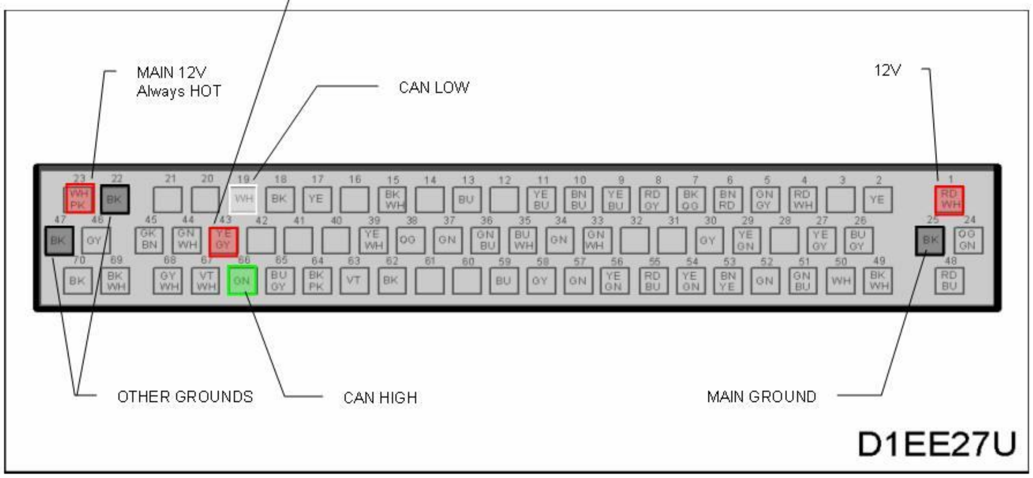
Appendix II : Trionic 7 pinout

70 pin connector

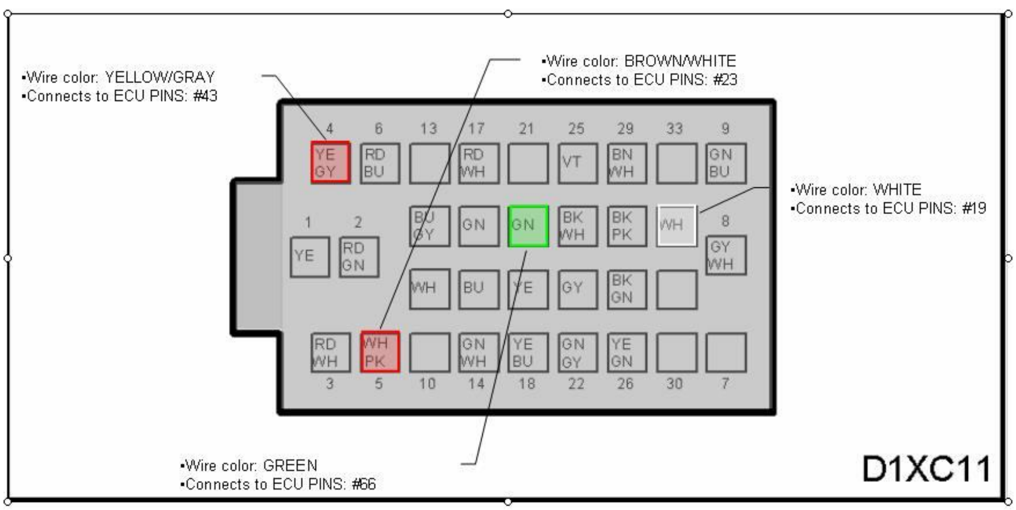
Pinnumber	Colour	Description	Range	In/out
1	Red/White	+12V	+12V	In
2	Yellow	Heating of rear lambda sonde	Ground	Out
4	Red/White	Relay of A/C compressor	Ground	Out
5	Green/Gray	Set switch cruise control	+54	In
6	Brown/Red	Combustion indicator cylinder 3 and 4 on 4 cylinder engine Combustion indicator cylinder 2, 4 and 6 on 6 cylinder engine		In
7	Black/Orange	Ignition trigger cylinder 1		Out
8	Red/Gray	Ignition trigger cylinder 4		Out
9	Yellow/Blue	Voltage from pedal potentiometer P2 in throttle body		In
10	Brown/Blue	Voltage from throttle potentiometer T1 in throttle body		In
11	Yellow/Blue	Pressure transducer on intake manifold		In
12		Relay SAI (Secondary air injection) on 6 cylinder engine	Ground	Out
13	Blue	Boost control valve	Ground	Out
15	Black/White	Temperature sensor delivery pipe (IAT)		
16		Optional wideband lambda input (spare AD input)		In
17	Yellow	Crankshaft sensor		
18	Black	Crankshaft sensor		
19	White	CAN Low (CAN_L)	Data	In/Out
20		Fuel injector cylinder 6	Ground	Out
21		Fuel injector cylinder 5	Ground	Out
22	Black	Ground for sensors		
23	White/PK	+12V, always connected		
24	Orange/Green	Throttle actuator in throttle body		
25	Black	Ground		
26	Blue/Gray	Main relay engine management system	Ground	Out
27	Yellow/Gray	Purge valve charcoal canister	Ground	Out
28				
29	Yellow/Green	Brake switch, cruise control	+12V	In
30		Cruise control resume switch	+54	In
31		Ignition trigger cylinder 5		Out
32		Ignition trigger cylinder 6		Out
33	Green/White	Voltage from throttle potentiometer T1 in throttle body?		In
34		Lambda signal first lambda sonde (front)	0-1 volt	In
35		Pressure sensor delivery pipe		In
36	Green/Blue	Limp home relay controller	Ground	Out
37	Green	EVAP shutoff valve	Ground	Out
38	Orange	Knock indicator from DI on 4 cylinder engine Knock indicator cylinders 1,3 and 5 on 6 cylinder engine		In
39	Yellow/White	Coolant temperature sensor signal		In
40		Optional EGT sensor input (PT100 thermo couple)		In
43	Yellow/Gray	Ignition (+12V)		
44	Green/White	Fuel injector cylinder 2		Out
45	GK/Brown	Fuel injector cylinder 1		Out
46	Gray	Feed for temperature and pressure sensors in delivery pipe and EVAP pressure transducer		Out
47	Black	Ground		
48	Red/Blue	Throttle actuator in throttle body		In/Out
49	Black/White	Same as pin 69 (connected inside throttle body)		
50	White	Fuel pump relay	Ground	Out
51	Green/Blue	Solenoid valve for turbo bypass valve	Ground	Out
52	Green	Cruise control on/off switch	+54	In
53	Brown/Yellow	Combustion indicator cylinder 1 and 2 for 4 cylinder engine Combustion indicator cylinder 1, 3 and 5 for 6 cylinder engine		In
54	Yellow/Green	Ignition trigger cylinder 2	Ground	Out
55	Red/Blue	Ignition trigger cylinder 3	Ground	Out
56	Yellow/Green	Voltage from pedal potentiometer P1 in throttle body		In
57	Green	Lambda signal rear lambda sonde	0-1 volt	In
58	Gray	Signal ground for lambda sondes	Ground	In
59	Blue	EVAP pressure transducer signal ?		
62	Black	Ground on 4 cylinder engine Knock indicator for cylinder 2, 4 and 6 on 6 cylinder engine		
63	Violet	Brake light switch	+30	In
64	Black/PK	ABS speed reading	Pulse	In
65	Blue/Gray	Air Mass Meter signal (might be Pink/White)		In
66	Green	CAN High (CAN_H)	Data	In/Out
67	Violet/White	Fuel injector cylinder 3		
68	Gray/White	Fuel injector cylinder 4		
69	Black/White	Voltage feed for the potentiometers in the throttle body. Connected to T7 pin 49 inside throttle body. Pressure transducer on intake manifold		
70	Black	Ground for sensors		

For your reference some schematic images for several connectors are included here.

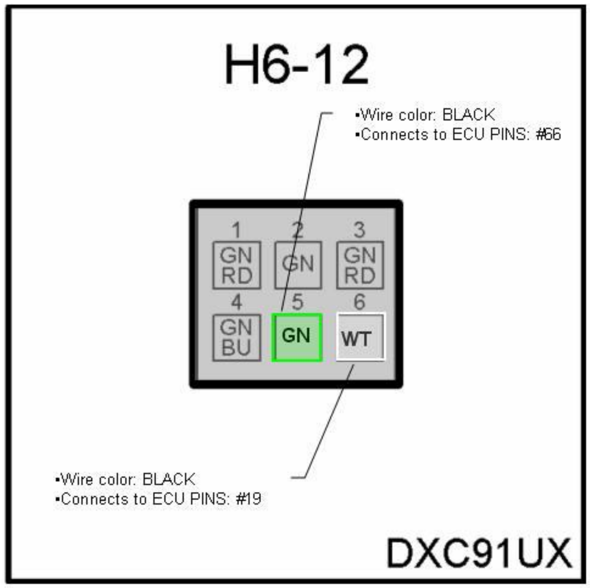
T7 ECU HARNESS (589b)



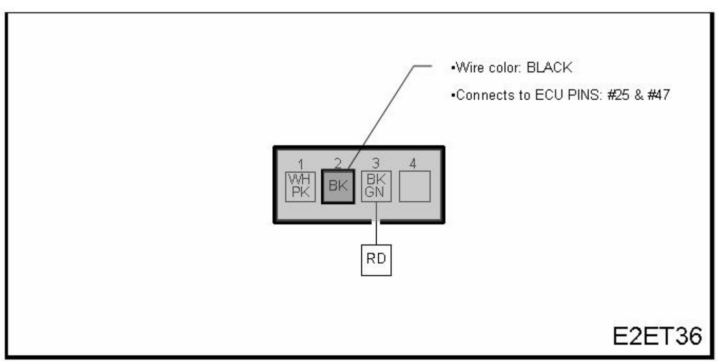
H33-5 HARNESS



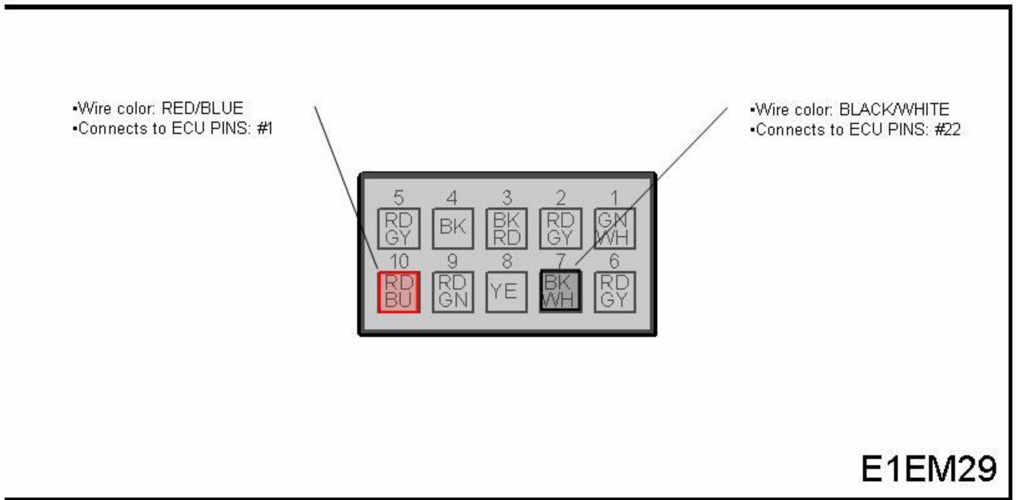
H6-12 HARNESS



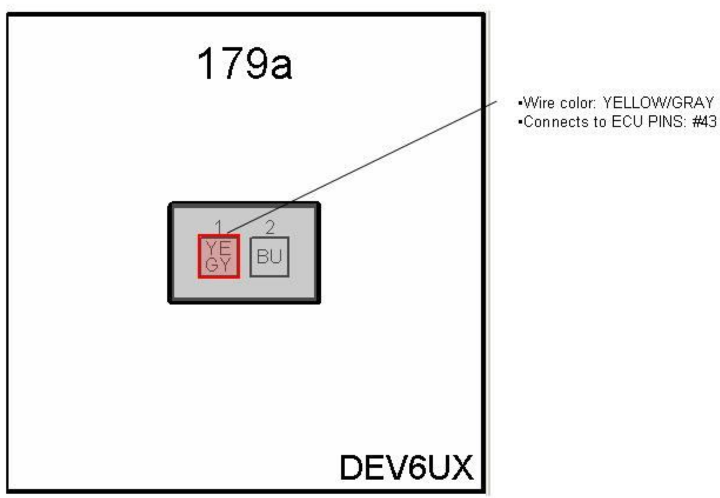
MAF HARNESS (205)



THROTTLE BODY HARNESS (604)



BOOST PRESSURE VALVE HARNESS (179a)



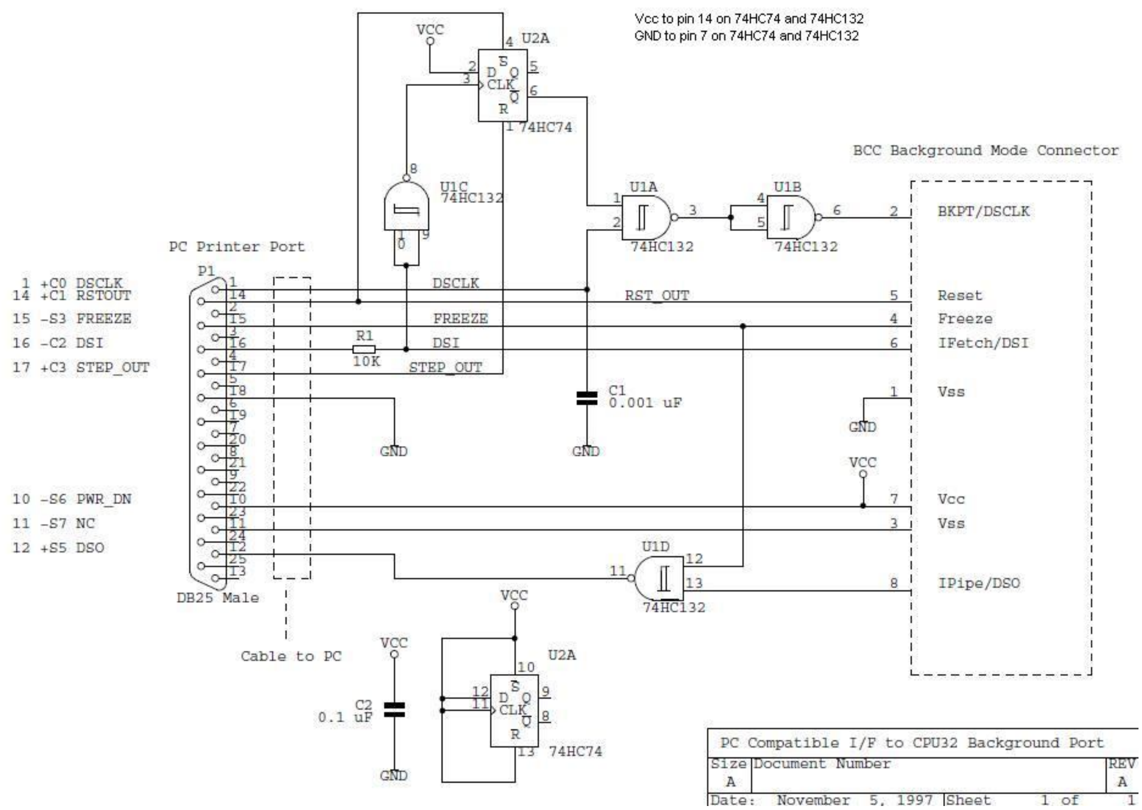
Appendix III : BDM technical information

General

BDM stands for Background Debug Mode. This refers to the mode the Motorola microcontroller is forced into when activating the BDM interface. This mode enables us to hold the processor in the program execution and read and write data from and to the memory inside the microcontroller and the memory connected to it. In this way we can download and program the flash contents which gives us access to the binaries we like so much! The BDM software you need can be downloaded from http://www.xendus.se/bdm/bd32-122.zip.

Home build 2 chips design schema

An alternative to buying a BDM interface can be building one yourself. This chapter will hand you all information needed to buy the components needed and the schema to build the interface. The image below shows the schema for the 2 chip design. There is also a 5 chip design and a GAL based design but these are more difficult to build at home.



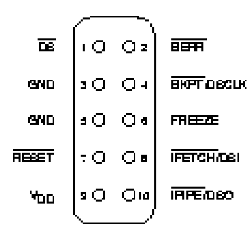
The table below shows the component list that you need to build the interface. Of course a soldering iron, PCB etc. are things that you also need.

Component	Amount	Description
74HC74	1	Dual JK Flip-Flop with Set and Reset
74HC132	1	Quad 2-input NAND Schmitt Trigger
Capacitor 0.1 uF	1	

Capacitor 0.001 uF	1	
Resistor 10 kΩ	1	
LPT cable + connector	1 meter	Smart is to get PCB type with normal LPT cable.
10 wire flat cable	20 cm	
10 pin female header for flat cable	1	

Pin out

Pinnumber	Pin name	Description	Remark
1	DS	Data strobe from target MCU. Not used in current interface circuitry	
2	BERR	Bus error input to target. Allows development system to force bus error when target MCU accesses invalid memory	
3	VSS	Ground reference from target	
4	BKPT/DSCLK	Breakpoint input to target in normal mode; development serial clock in BDM. Must be held low on rising edge of reset to enable BDM	
5	VSS	Ground reference from target	
6	FREEZE	Freeze signal from target. High level indicates that target is in BDM	
7	RESET	Reset signal to/from target. Must be held low to force hardware reset	
8	IFETCH/DSI	Used to track instruction pipe in normal mode. Serial data input to target MCU in BDM	
9	VCC	+5V supply from target.BDM interface circuit draws power from this supply and also monitors 'target powered/not powered' status	
10	IPIPE/DSO	Tracks instruction pipe in normal mode. Serial data output from target MCU in BDM	



Appendix IV : Turbo compressor maps

Each turbo has its own characteristics. These are determined by the size of the turbine housing, the size of the compressor wheel, the size of the turbine blades and many more parameters.
The most important identification of a turbocharger is by its compressor map. This is a graphical representation of its efficiency. In SAAB Trionic 5 cars there are 2 commonly used turbo chargers: the Garrett T25 for B204E, B204S, B204L, B234E and B234L engines and the Mitsubishi TD04-HL-15G/T (6cm2) for the B234R engines.

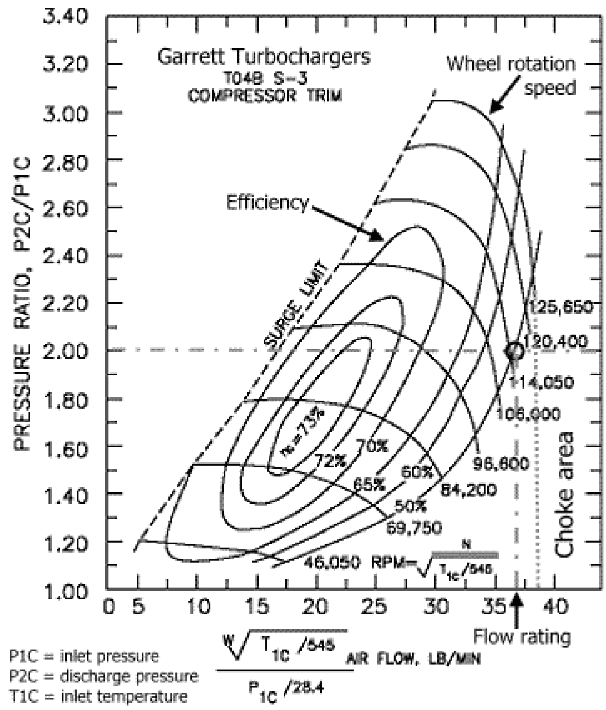
http://www.automotivearticles.com/Turbo_Selection.shtml

Terms to know:

- Compressor and turbine wheels. The turbine wheel is the vaned wheel that is in the exhaust gases from the engine. It is propelled by the exhaust gases themselves. The turbine wheel is connected to the compressor wheel by an axle. So, the compressor wheel will spin together with the turbine wheel. The compressor wheel also is vaned and these vanes compress the air and force it into the intercooler.
- Wheel “trim”. Trim is an area ratio used to describe both turbine and compressor wheels. Trim is calculated using the inducer and exducer diameters. As trim is increased, the wheel can support more air/gas flow.
- Compressor and turbine housing A/R. A/R describes a geometric property of all compressor and turbine housings. Increasing compressor A/R optimizes the performance for low boost applications. Changing turbine A/R has many effects. By going to a larger turbine A/R, the turbo comes up on boost at a higher engine speed, the flow capacity of the turbine is increased and less flow is wastegated, there is less engine backpressure, and engine volumetric efficiency is increased resulting in more overall power.
- Clipping. When an angle is machined on the turbine wheel exducer (outlet side), the wheel is said to be 'clipped'. Clipping causes a minor increase in the wheel's flow capability, however, it dramatically lowers the turbo efficiency. This reduction causes the turbo to come up on boost at a later engine speed (increased turbo lag). High performance applications should never use a clipped turbine wheel. All Garrett GT turbos use modern unclipped wheels.
- CFM = Cubic feet per minute.
- Lbs/minute = pounds (weight) per minute.
- M3/s = cubic meters per second.
- Corrected Airflow. Represents the corrected mass flow rate of air, taking into account air density (ambient temperature and pressure)
  Example:
  Air Temperature (Air Temp) - 60°F
  Barometric Pressure (Baro) - 14.7 psi
  Engine air consumption (Actual Flow) = 50 lb/min
  $$\text{Corrected Flow} = \frac{\text{Actual Flow}\sqrt{([\text{Air Temp}+460]/545)}}{\text{Baro}/13.95}$$
  $$\text{Corrected Flow} = \frac{50 * \sqrt{([60+460]/545)}}{14.7/13.95} = 46.3\ \text{lb/min}$$
- Pressure Ratio
  Ratio of absolute outlet pressure divided by absolute inlet pressure
  Example:
  Intake manifold pressure (Boost) = 12 psi
  Pressure drop, intercooler ($\Delta P_{Intercooler}$) = 2 psi
  Pressure drop, air filter ($\Delta P_{Air\ Filter}$) = 0.5 psi
  Atmosphere (Atmos) = 14.7 psi at sea level
  $$PR = \frac{\text{Boost} + \Delta P_{Intercooler} + \text{Atmos}}{\text{Atmos} - \Delta P_{Air\ Filter}}$$
  $$PR = \frac{12+2+14.7}{14.7-.5} = 2.02$$

How to read compressor maps

A 2-dimensional pressure map looks like this.



Turbo compressor map

The **curved lines** indicate the rotation speed (rpm) of the compressor wheel. In the sample map above these values are 45050, 69750, 84200 up to 125650 rpm. This is how fast a turbine wheel spins!!
The **elliptical circle** means the compressor's efficiency area. It's marked by the percent sign.
The **horizontal axis** is the amount of air before turbo, (1 m3/s = 2118.88 cfm, 10 lb/min = 144.718 cfm).

The **vertical axis** is the pressure ratio, the ratio of air pressure leaving to the turbo to air pressure entering the turbo.
Pressure Ratio=The pressure at compressor exducer vs. the pressure at compressor inducer.
In another word, the ratio of the pressure of the air after compression vs. the pressure before compression. As you can see, the pressure ratio depends on the ambient pressure. For example, at sea level, a turbo boosts 14.7psi. Ambient pressure is 14.7psi. That's 2 pressure ratio (PR) on the compressor map. Take that turbo to a higher elevation, the ambient pressure is less than 14.7psi. If the turbo still boosts 14.7psi, the pressure ratio would be higher. Now on the compressor map, you will see by moving up along a vertical line (to pump out the same cfm) and turbo efficiency has decreased as the elevation increases (PR increases). Simply put, turbos lose performance and become less efficient as elevation gets higher.

Choke area

The area to the right of the outer most elliptical circle is the least efficient area, the **choke area**.
It means when the compressor reaches certain rpm, the air moved by the compressor wheel in the diffuser area of the compressor housing is moving at or past the speed of sound. When the air speed reach sonic speed, the amount of air flow increase is very small as compressor wheel rpm increases. In plain words, the compressor has reached its limit. You can try to pump more psi, have the wheel

spin faster, but very little more air is pumped out the turbo compressor. You can see now, the compressor housing will need to properly match the compressor wheel. If you simply stuff a big wheel inside a small compressor housing, the diffuse area will be very small. This causes the air inside the housing to move at higher speed. That's why some of the so-called T28s which use a bigger compressor wheel inside the stock compressor housing does not produce good hp.

Compressor maximum flow

The max flow of a compressor is shown on the compressor map. On the map, look for the intersection of maximum compressor wheel speed (rpm) and the least compressor efficiency curve. Find that intersection. The horizontal coordinate is the max flow.
The area to the right of maximum flow is the 'choke area'.
The vertical coordinate is the pressure ratio at which the compressor reaches that maximum flow. From this boost level, as the boost increases, very little air flow is increased. For example, if a compressor reaches its maximum flow at 2 PR or 1 atm pressure or 14.7psi, higher boost does not pump more air into the motor. But higher boost may be needed to increase the manifold pressure for the motor to flow more air. A 5 liter motor with this turbo needs 15psi of manifold pressure to flow a certain CFM. A 3 liter motor with the same turbo will need much higher manifold pressure to flow the same amount of air although that turbo's compressor does not flow more air past 14.7psi.

Compressor maximum pressure

On the map, find the top-most point on the graph. The vertical coordinate is the max pressure ratio. For example, 2.8 pressure ratio at sea level is 1.8 times the atmospheric pressure, 1.8x14.7psi=26.46 psi. Compressor max pressure is limited by compressor wheel speed. It's physically impossible to boost higher than this maximum pressure for one particular turbo. Plus the pressure drop in the intercooler system, the actual maximum boost reading from a boost gauge that's plugged into the intake manifolds maybe a few psi lower than this maximum pressure.

What the compressor map reads

Most manufactures rate their turbos at 1 bar (15 psi). That's 2 pressure ratio. On the map, draw a horizontal line from 2PR. When the line intersects the right-most elliptical circle, the corresponding number on the x-axis is the maximum cfm the turbo can flow at 1 bar.
Use the TD04-15G's map for example, where the 2 PR line hits the right-most efficiency curve, it reads 428cfm as its flow rate at 15psi.

Comparing compressor maps

Well, compressor maps are really 3-dimensional maps. Any compressor map looks a hill/peak in 3 dimensions. Our compressor maps look like if you look at the hill directly from above vertically. The elliptical lines of elevations are the efficiency curves. Since in theory, we can always boost more and decrease turbo efficiency to get more cfm, let’s set the same Pressure Ratio and compare turbos at the same efficiency curves.

As a rule of thumb, a large turbo will be better at making a lot of pressure but will spool slower than a small turbo. A small turbo will build boost fast but is less capable to make big boost pressure.

Understanding information within the compressor map

1. The oblong ovals on the chart or “islands” as they are called represent the efficiency of the turbo in that range. As you can see on this map, the most efficient operation (73%) is in the very centre of the chart. This is general characteristic of most turbochargers. Without getting into the thermodynamics of adiabatic heat-pumps, we’ll just say that efficiency is a measure of how much excess heat the turbo puts into the compressed air coming out of the outlet. So intuitively, more efficient is better.

2. Wheel rotational speed is simply the rpm at which the compressor wheel is spinning.

3. The choke point, which is usually not indicated on flow maps, is the maximum flow rating the turbo is capable of regardless of pressure or efficiency.

4. Beyond the surge limit on the left of the plot, compressor surge occurs. In laymen's terms, this phenomenon is caused by a back pressure wave entering the exit of the compressor housing and disrupting flow through the compressor wheel. Surge will kill turbos and is to be avoided at all costs.

Surge Limit

To the left of the surge limit line on the flow map is the *surge area* where compressor operation can be unstable. Typically, surge occurs after the throttle plate is closed while the turbocharger is spinning rapidly and the by-pass valve does not release the sudden increase in pressure due to the backed-up air. During surge, the back-pressure build-up at the discharge opening of the compressor reduces the air flow. If the air flow falls below a certain point, the compressor wheel (the impeller) will loose its "grip" on the air. Consequently, the air in the compressor stops being propelled forward by the impeller and is simply spinning around with the wheel, which is still being rotated by the exhaust gas passing through the turbine section. When this happens, the pressure build-up at the discharge opening forces air back through the impeller causing a reversal of air flow through the compressor. As the back pressure eventually decreases, the impeller again begins to function properly and air flows out of the compressor in the correct direction. This sudden air-flow reversal in the compressor can occur several times and may be heard as a repetitive "Whew Whew Whew" noise if the surge is mild (such as when the by-pass valve is set a little too tight) to a loud banging noise when surge is severe. Surge should be prevented at all costs because it not only slows the turbocharger wheels so that they must be spooled back up again but because it can be very damaging to the bushings or bearings and seals in the center section.

Selecting a different turbo charger

Calculating your engine's flow requirements

Now that you can read and understand a compressor flow map, its time to figure out how to match a turbo to your engine, this involves selecting the proper compressor and turbine wheels along with the right combination of housing A/R. A mismatched turbo could not only result is extreme lag, but also wasted potential as a turbo can easily outflow an engine. I.e. bigger is not always better.

The only real calculation that needs to be done is to determine how much air you engine is actually flowing. This depends on a number of things including the RPM, absolute temperature (Rankin, equal to 460 + Fahrenheit temp), absolute manifold pressure (psi, equal to boost pressure plus atmospheric pressure), and lastly the engine volumetric flow or EVF in cfm.

First to calculate EVF use the following equation:

$$EVF = \left(\frac{engineCID}{1728}\right) * \left(\frac{RPM}{2}\right)$$

Engine CID = Engine displacement in cubic inches.
Next we'll use EVF to calculate the amount of air in lb/min the engine is flowing under boost and at temperature using this equation:

$$N = \frac{P * EVF * 29}{10.73 * T}$$

Where N is the airflow in lb/min, P is the absolute pressure in psi, and T is the absolute ambient temperature in Rankin.

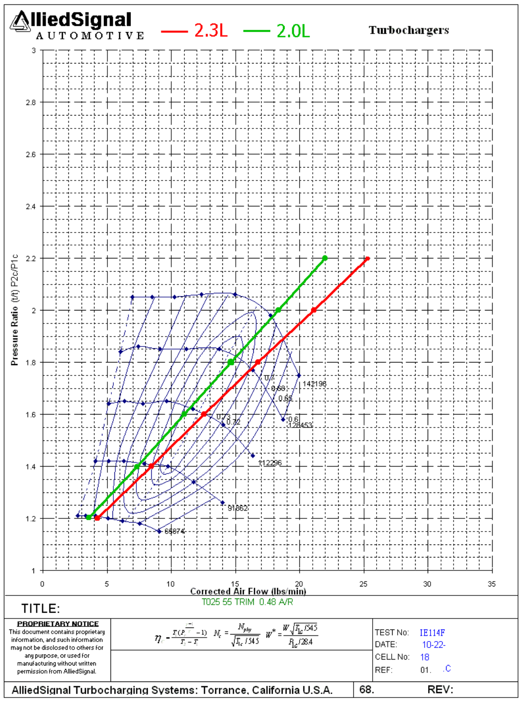
Finally, multiply N by the volumetric efficiency of your engine (VE). This compensates for the fact that upon every cycle of the engine, not all of the old air/fuel mix in the cylinders is forced out the exhaust. Thus there is a difference between the actual airflow through and engine and the predicted airflow. This discrepancy is equated to a VE. There is literally thousands of hours worth of online reading about volumetric efficiencies for just about every production engine. To get the most accurate results from this step I would suggest researching your engine and coming up with the most realistic VE possible as this does have a significant effect on engine flow. If you are just messing around with compressor flow maps and need a value for VE just to experiment with, 85% efficiency is a nice conservative number for most modified turbocharged cars at high rpm (6500-7500). Keep in mind though that on a forced induction setup VE can easily exceed 100% so again it will be very beneficial to research *your* engine.
For our SAAB engines these number apply.

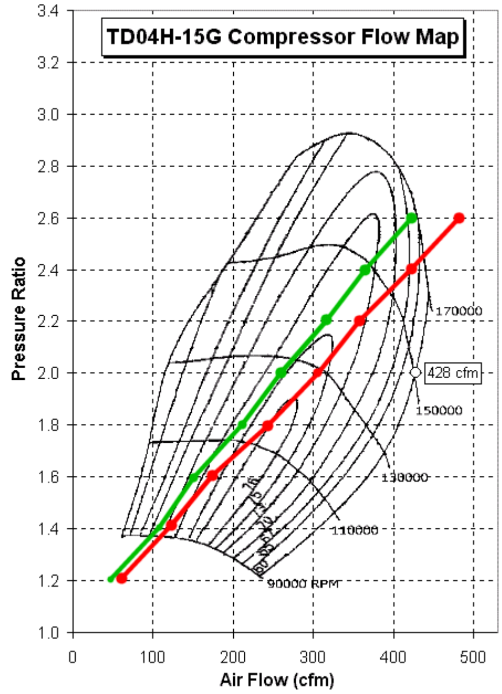
EVF / RPM	2.0L engine (122 cubic inch)	2.3L engine (140 cubic inch)
1000 rpm	35.3	40.6
2000 rpm	70.6	81.2
3000 rpm	105.9	121.8
4000 rpm	141.3	162.4
5000 rpm	176.6	203.1
6000 rpm	211.9	243.7
7000 rpm	247.2	284.3

Since the amount of air to be flowed by the turbo is largest when RPM is at its top we will take the worst case scenario and get EVF @ 7000 RPM. We have to make an assumption on the ambient temperature which we will set at 20°C. This is 68° Fahrenheit which is (460 + 68) = 528 Rankin.
Now we can calculate the airflow of the engine in lb/min for any given boost level.
If we want to draw a line into the compressor map for our engines needs we need to calculate the needed airflow for several boost levels.

Airflow / boost pressure	2.0L lb/min	2.0L cfm	2.3L lb/min	2.3L cfm
0.2 bar (2.9 psi)	3.7	53.1	4.2	61
0.4 bar (5.8 psi)	7.3	106.1	8.4	122
0.6 bar (8.7 psi)	11	159.2	12.6	183
0.8 bar (11.6 psi)	14.7	212.2	16.9	244
1.0 bar (14.5 psi)	18.3	265.3	21.1	305
1.2 bar (17.4 psi)	22	318.3	25.3	366
1.4 bar (20.3 psi)	25.7	371.4	29.5	427
1.6 bar (23.2 psi)	29.3	424.4	33.8	488

This all results in 2 simple lines in the compressor map which indicate the maximum flow required from the turbo by our engine.





Now we can clearly see where out engine leaves the compressor map and thus the limit for the combination of the two (engine and turbo) lies.
We also see that – even for the 2.3 litre engine – the TD04 can sustain a much higher boost pressure at higher rpms than the T25 can. Even at 1.4 bar boost (pressure ratio = 2.4) the TD04 is within its limits and would flow approximately 420 cfm. Would we have done the same with the T25 turbo we would most certainly be in the choke area and the turbo would be unable to get us the airflow that we required.

Determining the Best Wheel Trim-Housing A/R Combination

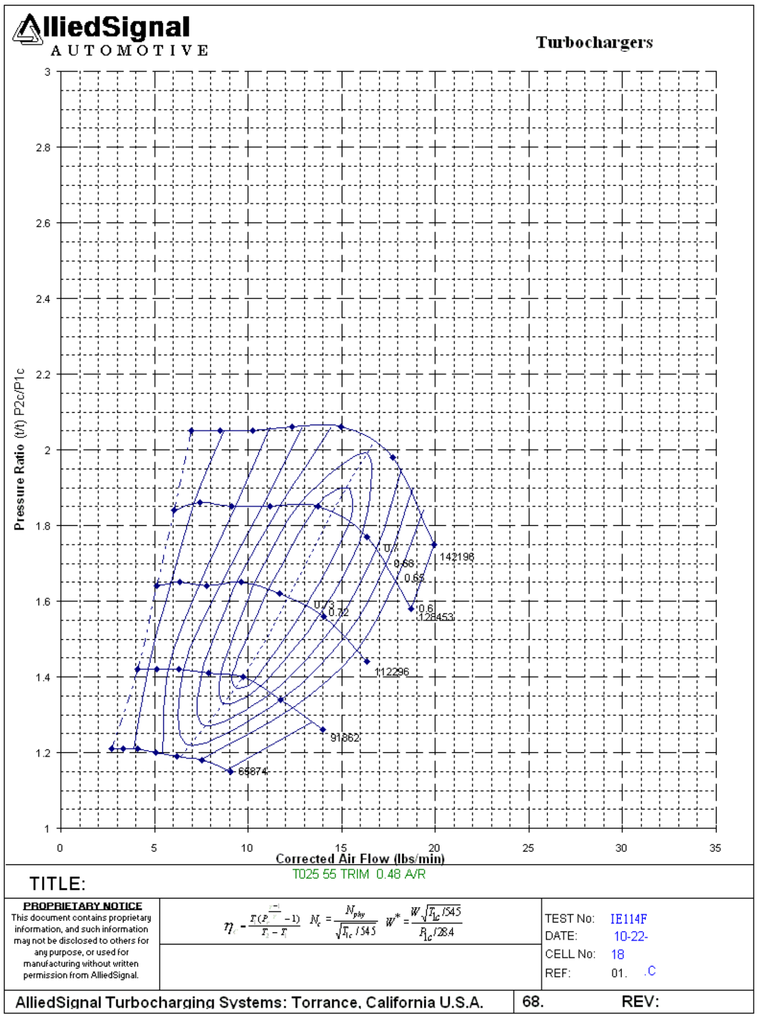
With the flow rate you have just calculated, you can look at compressor maps of different turbo chargers to see which ones give you the air flow you need at the pressures and efficiencies that you want to run.

When selecting a turbo, it is important to do the above calculations for a number of different RPM's and boost pressures because you will not always be at redline under full boost while driving you car. Checking the turbo performance at various engine speeds and pressures will give the overall picture of how well the turbo is sized to your vehicle.

Matching a flow map to your engine flow requirements will allow you to pick the compressor wheel trim for your application. However before you can go out and purchase that new turbo, you still have to settle on an exhaust wheel and turbine A/R. The real determining factor in this selection is maintaining compressor wheel speed. Remember the wheel RPM lines on the flow map? Well a properly sized exhaust wheel/housing combination will keep the compressor wheel operating within the maximum and minimum wheel speeds on the map as often as possible. Since different "hot side" combinations can affect your turbos performance, (i.e. a little more lag in return for more top end, or quicker spool up at the cost of overall power) the best thing to do is to contact a turbo manufacturer or distributor (www.forcedperformance.com, www.turbochargers.com) and they will be able to tell you the exact effects you can expect from all of the various hot side combos available for your turbo model.

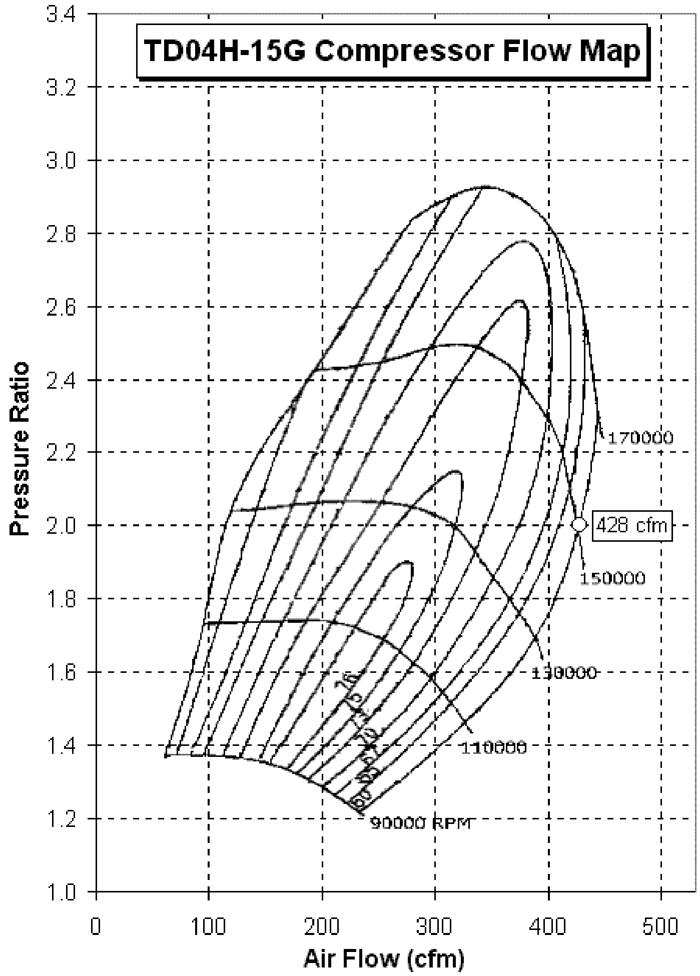
Garrett T25 specifications

Weight: 7,5kg
Compressor diameter: 60mm
Turbine diameter: 65mm and 59mm
Bearing span: 37,8mm
Moment of inertia: 5,4x10-5 kg/m2
Oil flow: 1,7L/min / SAE30 / 90C / 2,75bar
Compressor wheel: 54mm 55 trim, A/R 48
Turbine wheel: 53.8mm 61 trim, A/R 49



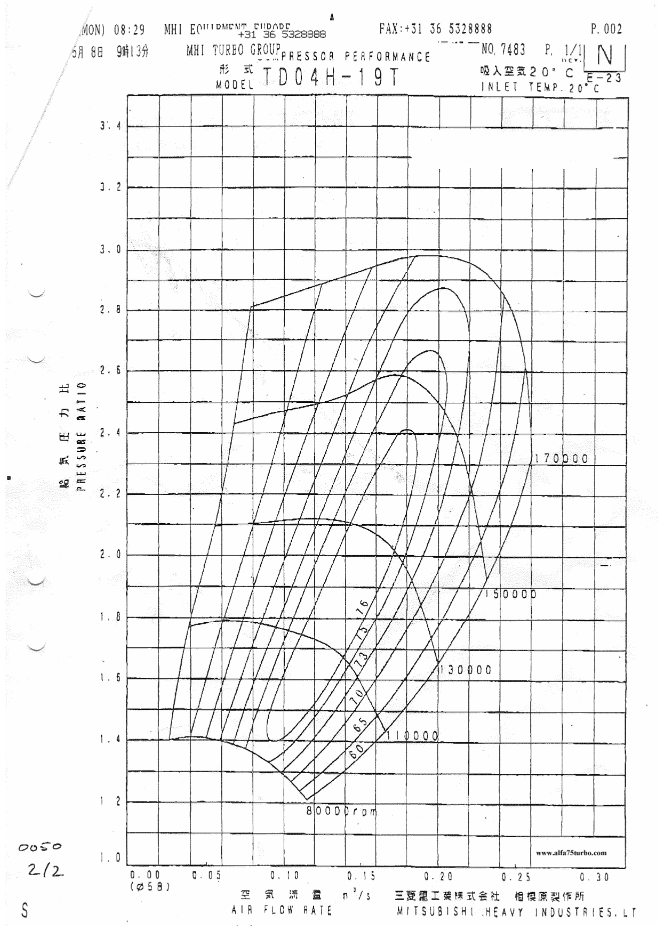
As you can see the T25 (trim 55) can flow 18 lbs of air per minute @ pressure ratio = 2 and efficiency will be ~65%. This 18 lbs/minute converts to about 260 cfm. The maximum efficiency zone (73%) reaches up to pressure ratio 1.9. This would be ~0.9 bar overpressure.
In terms of usage the T25 can take us up to ~1.1 bar boost pressure and bring up to 250 bhp.

Mitisubishi TD04-15G specifications



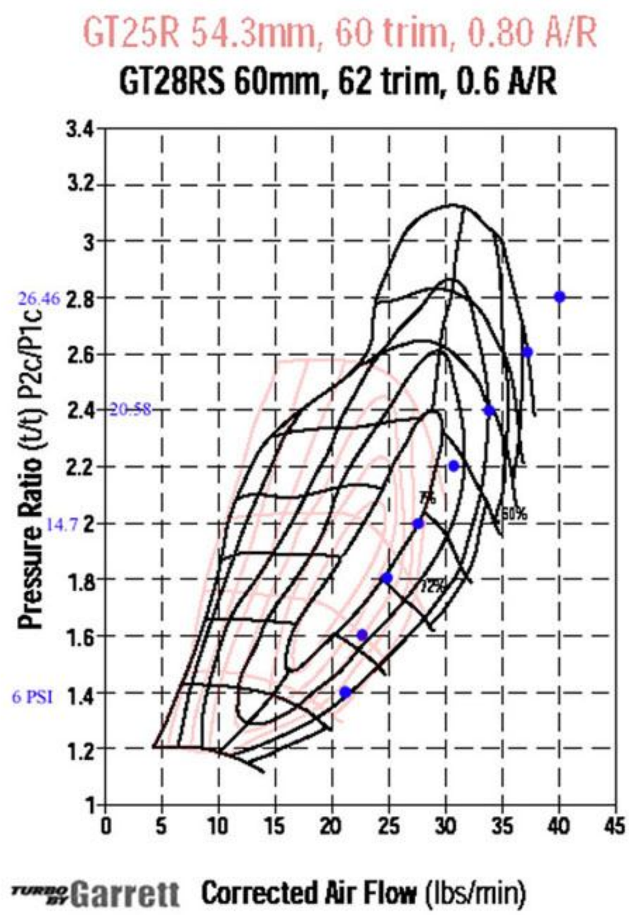
The TD04-15G from the 9000 Aero can flow more air than the "little" T25. Looking at the map we can see that the TD04-15G can flow ~428 cfm @ pressure ratio 2 and efficiency of 60%.
Also, the maximum efficiency zone (76% vs. 73% for the T25) reaches up to 1.9 pressure ratio with would be ~0.9 bar.
In terms of usage the TD04 can take us up to ~1.4 bar boost pressure. The TD04 can bring up to ~330bhp.

Mitsubishi TD04-19T specifications



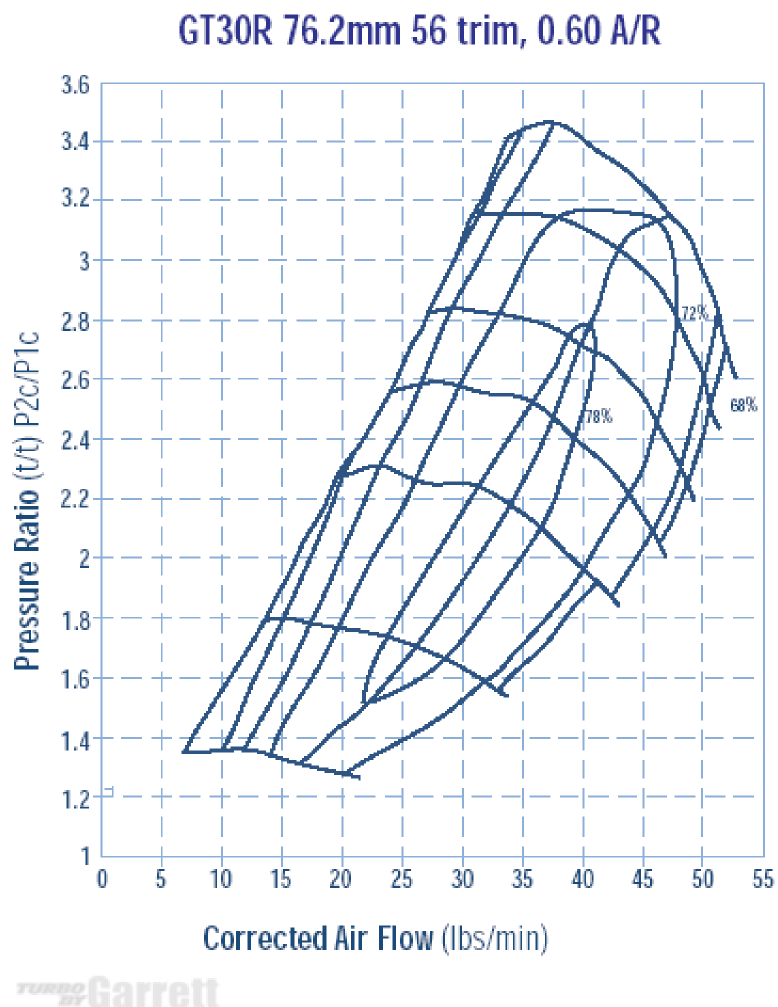
This TD04-19T compressor map has the air flow axis noted in m3/s. 1 m3 per second means 2118 cfm. So this turbo can flow approximately 0.26 m3/s = 550 cfm @ pressure ratio 2.3.
At the standard ratio of 2 it still flows 510 cfm which is more than a GT28RS. The TD04-19T can bring up to ~380 bhp.

Garrett GT28RS (GT2860R) specifications



The GT28RS is frequently used in modified cars. It can flow much more air than the TD04 and the T25. If we look at the compressor map for the GT28RS (black lines) we see that this turbo is capable of flowing ~33 lbs/minute which is ~477 cfm at pressure ratio 2. The maximum efficiency runs much higher than that. Up to 1.4 bar boost pressure this turbo will run at 75% efficiency.
In terms of usage the GT28RS can run nicely up to 1.6 bar at an air flow rate of ~37 lbs/min = 535 cfm. This is much more than the TD04-15G of course. The GT28RS can bring up to ~350 bhp.

Garrett GT30R specifications



The GT30R (GT3071) is given for reference reasons only. It can flow even more air than the GT28RS. If we look at the compressor map we see that this turbo is capable of flowing ~45 lbs/minute which is ~650 cfm at pressure ratio 2. The maximum efficiency runs up to 1.8 bar boost pressure. In terms of usage the GT30R can run nicely up to 1.8 bar at an air flow rate of ~52 lbs/min = 750 cfm. Even within the maximum efficiency zone this turbo will flow 40 lbs/minute which is more than the GT28RS will flow even at maximum.
The GT30R can bring up to ~500 bhp.

Conclusion

Comparing the two compressor maps for TD04 and T25 we can clearly see that the TD04 can flow more air at a higher pressure ratio and with a higher efficiency. Given the fact that the turbine blades are larger than in the T25, spool up will be a bit slower, but high end power will be much better.

Upgrading your turbo will affect more than meets the eye. The VE map in the Trionic would probably need adjustments because the hardware in the airflow has been changed. This means the volumetric efficiency also changes and thus the correction table needs changing too.
Also, when boost values rise, the intercoolers capacity for air flow comes into play. You must make sure that the intercooler is not so restrictive that upgrading the turbo will result in a burst intercooler. A high capacity cross flow intercooler would be a good option here.
And last but not least, upgrading the turbo charger means – that is the goal here – more air flow to the cylinders and thus more oxygen to burn. If we upgrade the turbo we need to consider the injectors too. If the injectors can't flow the amount of fuel needed to burn the amount of oxygen pushed into the cylinders we would have gained nothing.

Appendix V: Upgrade stages 1-7

Originally from T5 equipped cars: needs to be updated to T7 equipped cars!

If you want to go beyond the standard stages I – III there's more to alter than just the "silly" stuff like ECU, exhaust and catalyst. This chapter will describe what steps are needed for stages 1 up to 7.

Stage I

The target amount of power for stage I is about 235 bhp for FPT versions (T25 turbo) and 260 bhp for Aero models (TD04-15 turbo)

Component	Stock	Stage I minimum requirement
ECU	200/225 bhp	Stage I
Exhaust	2"	2"
Intake	---	---
Catalyst	---	---
Injectors	345 cc/min	345 cc/min
Fuel lines	---	---
Turbo	T25/TD04-15	T25/TD04-15
Exhaust manifold	---	---
Intercooler	---	---
Clutch	450 Nm	450 Nm
Camshafts	---	---
Fuel pump	---	---
Wastegate	---	---
Mapsensor	2.5 bar	2.5 bar
Air delivery pipe	---	---
Cylinder head	---	---

Stage II

The target amount of power for stage III is about 250 bhp for T25 models and 270 bhp for TD04 models.

Component	Stock	Stage II minimum requirement
ECU	200/225 bhp	Stage II
Exhaust	2"	3" cat-back
Intake	---	---
Catalyst	---	---
Injectors	345 cc/min	345 cc/min
Fuel lines	---	---
Turbo	T25/TD04-15	T25/TD04-15
Exhaust manifold	---	---
Intercooler	---	---
Clutch	450 Nm	450 Nm
Camshafts	---	---
Fuel pump	---	---
Wastegate	---	---
Mapsensor	2.5 bar	2.5 bar
Air delivery pipe	---	---
Cylinder head	---	---

Stage III

The target amount of power for stage III is about 270 bhp for T25 models and 280 bhp for TD-04 models.

Component	Stock	Stage III minimum requirement
ECU	200/225 bhp	Stage III
Exhaust	2”	3” turbo back
Intake	---	Open/sport air filter
Catalyst	---	Sport (3”) catalyst
Injectors	345 cc/min	345 cc/min
Fuel lines	---	---
Turbo	T25/TD04-15	T25/TD04-15
Exhaust manifold	---	---
Intercooler	---	---
Clutch	450 Nm	450 Nm
Camshafts	---	---
Fuel pump	---	---
Wastegate	---	---
Mapsensor	2.5 bar	2.5 bar
Air delivery pipe	---	---
Cylinder head	---	---

Stage IV

The target amount of power for stage IV is about 300 bhp. From stage 4 there’s no longer a difference between FPT and Aero models because the T25 turbo cannot reach a stage IV level and has to be replaced from this stage on.

Component	Stock	Stage IV minimum requirement
ECU	200/225 bhp	Stage IV
Exhaust	2”	3” turbo back
Intake	---	Open/sport air filter
Catalyst	---	Sport (3”) catalyst
Injectors	345 cc/min	413 cc/min
Fuel lines	---	---
Turbo	T25/TD04-15	TD04-15
Exhaust manifold	---	---
Intercooler	---	---
Clutch	450 Nm	Sports clutch (600 Nm)
Camshafts	---	---
Fuel pump	---	---
Wastegate	---	Reinforced model (Forge)
Mapsensor	2.5 bar	3 bar
Air delivery pipe	---	---
Cylinder head	---	---

Stage V

The target amount of power for stage V is about 350 bhp

Component	Stock	Stage V minimum requirement
ECU	200/225 bhp	Stage V
Exhaust	2"	3" turbo back
Intake	---	Open/sport air filter
Catalyst	---	Sport (3") catalyst
Injectors	345 cc/min	413 cc/min
Fuel lines	---	---
Turbo	T25/TD04	TD04-18T/19T
Exhaust manifold	---	---
Intercooler	---	Cross flow, high capacity
Clutch	450 Nm	Sports clutch (600 Nm)
Camshafts	---	---
Fuel pump	---	---
Wastegate	---	Reinforced model (Forge)
Mapsensor	2.5 bar	3 bar
Air delivery pipe	---	---
Cylinder head	---	---

Stage VI

The target amount of power for stage VI is about 400 bhp

Component	Stock	Stage VI minimum requirement
ECU	200/225 bhp	Stage VI
Exhaust	2"	3" turbo back
Intake	---	Open/sport air filter
Catalyst	---	Sport (3") catalyst
Injectors	345 cc/min	630 cc/min
Fuel lines	---	---
Turbo	T25/TD04	Garrett GTBB30
Exhaust manifold	---	---
Intercooler	---	Cross flow, high capacity
Clutch	450 Nm	Sports clutch (600 Nm)
Camshafts	---	Upgraded model for better engine breathing
Fuel pump	---	Walbro 255 or Bosch 044
Wastegate	---	Reinforced model (Forge)
Mapsensor	2.5 bar	3 bar
Air delivery pipe	---	---
Cylinder head	---	---

Stage VII

The target amount of power for stage VII is about 500 bhp

Component	Stock	Stage VII minimum requirement
ECU	200/225 bhp	Stage VII
Exhaust	2"	3" turbo back
Intake	---	Open/sport air filter
Catalyst	---	Sport (3") catalyst
Injectors	345 cc/min	750 cc/min
Fuel lines	---	Upgraded piping and rail
Turbo	T25/TD04	Garrett GT30R
Exhaust manifold	---	Tubular model with all mandrels of same length
Intercooler	---	Cross flow, high capacity
Clutch	450 Nm	Sports clutch (700 Nm)
Camshafts	---	Upgraded model for better engine breathing
Fuel pump	---	Walbro 255 or Bosch 044
Wastegate	---	Reinforced model (Forge)
Mapsensor	2.5 bar	3 bar
Air delivery pipe	---	Upgraded model (Abbott)
Cylinder head	---	Custom ported

Appendix VI: Check Engine Light (CEL)

Appendix VII: Knock and misfire detection

Trionic detects knocking and misfires by means of the ionization current that flows between the spark plug gaps. This appendix will explain how this works.

http://www.fs.isy.liu.se/~larer/Projects/main.html

Ionization current generation

In an ideal combustion reaction, hydrocarbon molecules react with oxygen and generate only carbon dioxide and water, e.g. isooctane gives

$$C_8H_{18} + \frac{25}{2}O_2 \longrightarrow 8CO_2 + 9H_2O.$$

In the combustion there are also other reactions present, that include ions, which go through several steps before they are completed.

$$\begin{array}{lcl} CH + O & \longrightarrow & CHO^+ + e^- \\ CHO^+ + H_2O & \longrightarrow & H_3O^+ + CO \\ \\ CH + C_2H_2 & \longrightarrow & C_3H_3^+ + e^- \end{array}$$

These ions, and several others, are generated by the chemical reactions in the flame front. Additional ions are created when the temperature increases as the pressure rises.

The processes creating the ionization current are complex and are also varying from engine cycle to engine cycle. *Image 25* shows ten (10) consecutive cycles of the cylinder pressure and the ionization current operating at constant speed and load.

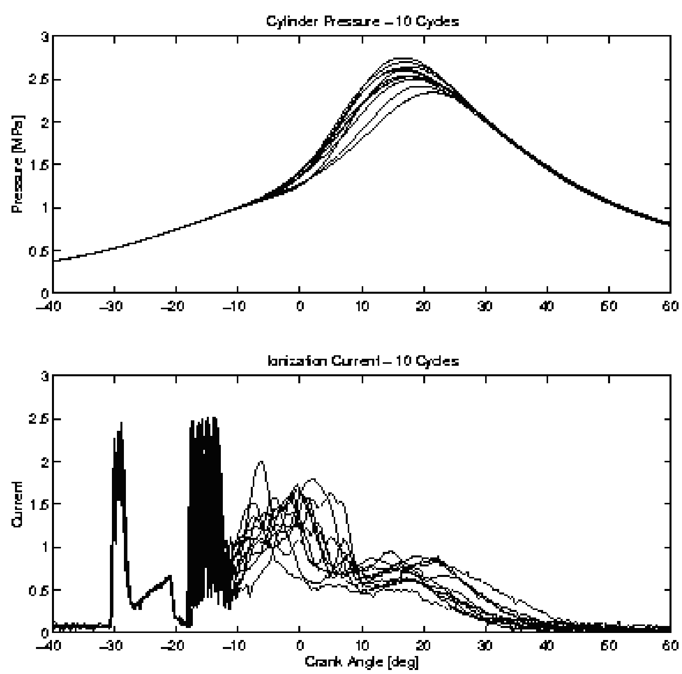


Image 3: Cycle to cycle variations in the combustion

As can be seen, the cycle-by-cycle variations are significant, which is a given problem in interpreting these signals.

Ionization current sensing

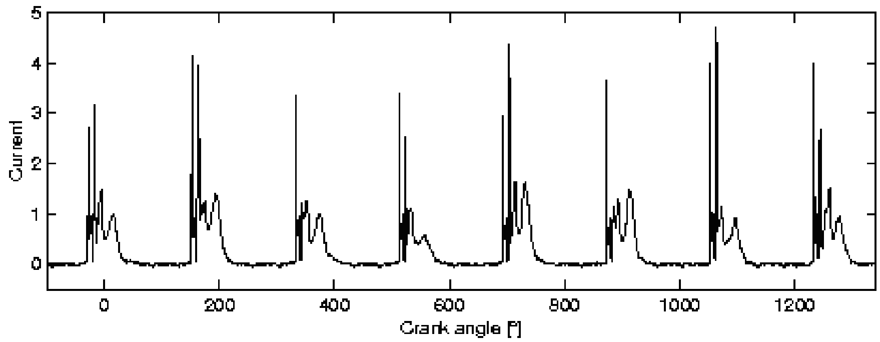


Image 4: Ionization current in one cylinder

Knock is a pressure oscillation in the cylinder with a frequency determined by the geometry of the combustion chamber. The oscillation is present in the current measurement and can be extracted mainly by using a band pass filter in a well chosen time window of the current signal. Knocking can destroy the engine. When there is a misfire, then there are no resulting ions and hence no current which is easily detected. Misfires can and will destroy the catalyst.
Ionization current interpretation can be used for both purposes, knock detection and misfire detection.

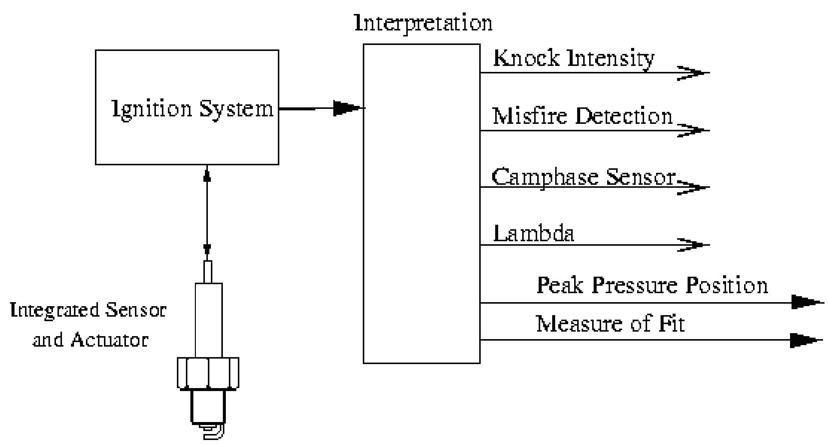


Image 5: Possible sensor information from ionization current

Detection

To detect the ions, a DC bias is applied to the spark plug, generating an electrical field. The electrical field makes the ions move and generates an ion current. A schematic illustration is shown in *Image 28 (a)*. The current is measured at the low-voltage side of the ignition coil, and does not require protection from the high-voltage pulses in the ignition, *Image 28 (b)*.

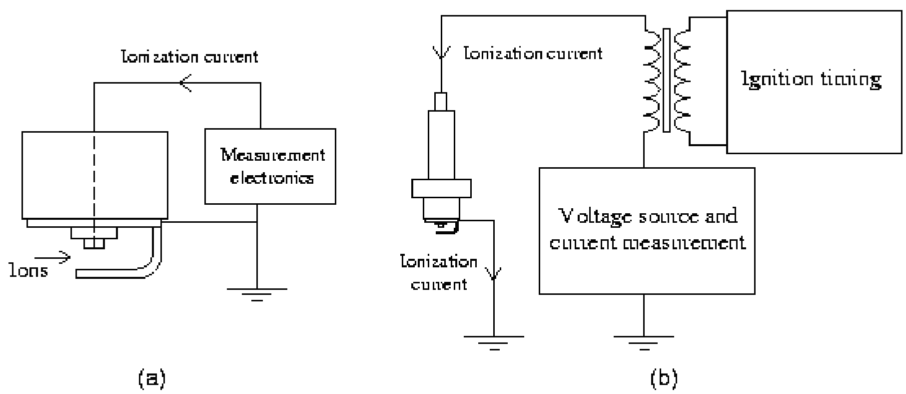


Image 6: Measuring the ionization current

Measurement of the ionization current. (a) The spark plug-gap is used as a probe. (b) Measurement on the low voltage side of the ignition coil.

The ionization current is an interesting engine parameter to study. It is a direct measure of the combustion result that contains a lot of information about the combustion, and several challenges remain in the interpretation of it. Some of the parameters that affect the ionization current are: temperature, air/fuel ratio, time since combustion, exhaust gas recycling (EGR), fuel composition, engine load, and several others.

Ionization Current Terminology

The ionization current typically has three phases: a phase related to ignition, a phase related to ions from the flame development and propagation, and a phase related to pressure and temperature development. In *Image 29*, the three phases of the ionization current are displayed. Each of these phases has varying characteristics and they also mix together in complicated ways. In the ignition phase, the ionization current is large, with reversed polarity. Due to the high current in the ignition the measured signal shown in the figure is limited. What can be seen in the image too is the ringing phenomenon in the coil after the ignition.

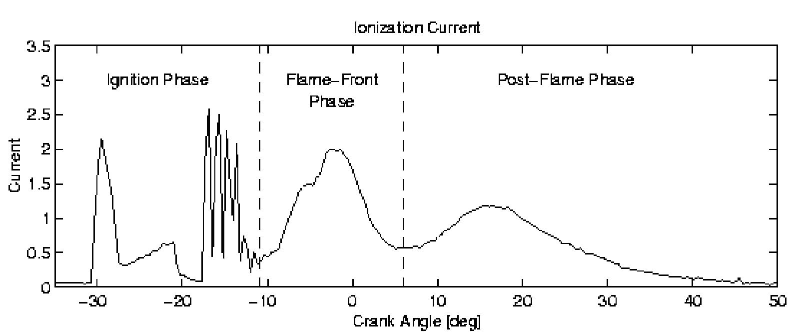


Image 7: Ionization current phases: ignition, flame front, and post flame

In the flame-front phase, the high level of ions associated with the chemical reactions in the flame produces one or more characteristic peaks. The ions generated by the flame have different recombination rates. Some ions recombine very quickly to more-stable molecules, while others have longer residual times. The result is a high peak which after some time decays as the ions recombine.

In the post-flame phase the most stable ions remain, generating a signal that follows the cylinder pressure due to its effect on the temperature and molecule concentration. Ions are created by the combination of the measurement voltage and the high temperature of the burned gases, since the temperature follows the pressure during the compression and expansion of the burned gases, when the flame propagates outwards and the combustion completes. The ionization current thus depends on the pressure.

Spark Advance and Cylinder Pressure

The spark advance is used to position pressure development in the cylinder such that the combustion produces maximum work. Under normal driving conditions the mixture is ignited around 15 – 30° in crank angle before the piston has reached top dead center (TDC), and the pressure peak comes around 20 degrees after TDC. In the graph below three different pressure traces, resulting from three different spark timings, are shown. Earlier spark advance normally gives higher maximum pressures and maximum temperatures that appear at earlier crank angles.

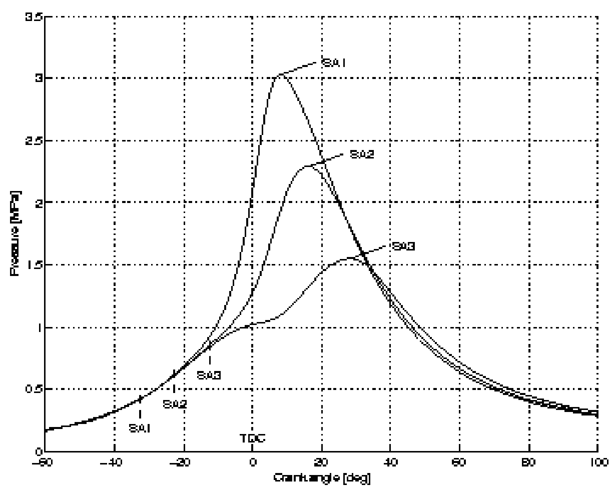


Image 8: Cylinder pressure vs. Ignition timing

Three different pressure traces resulting from three different spark advances. The different spark advances are; **SA1**: spark advance 32.5° before top dead center (TDC), **SA2**: 22.5° before TDC, **SA3**: 12.5° before TDC. The optimal spark advance is close to **SA2**.

The optimal spark advance for maximum output torque is close to **SA2** for the operating point in the figure, and the resulting peak pressure position lies around 17° after TDC. With too early ignition timing the pressure rise starts too early and counteracts the piston movement. This can be seen for the pressure trace with spark advance **SA1** where the pressure rise starts already at -20° due to the combustion. There are also losses due to heat and crevice flow from the gas to the combustion chamber walls, and with an earlier spark advance the loss mechanisms start earlier reducing the work produced by the gas. Higher pressures give higher temperatures which also decrease the difference in internal energy between the reactants and products in the combustion, thus resulting in lower energy-conversion ratios. The heat loss mechanisms and the lower conversion ratio can be seen in *Image 30*, at crank angles over 30°, where the pressure trace from the **SA1** spark advance is lower than the others.

Too late ignition gives a pressure increase that comes too late so that work is lost during the expansion phase. In *Image 30*, the pressure increase for spark advance **SA3** starts as late as at TDC. But work is also gained due to the later start of the effects mentioned above, which also can be seen in the figure. The pressure trace from the spark advance, **SA3**, is higher than the others at crank angles over 30˚. However, this gain in produced work can not compensate for the losses early in the expansion phase.

Peak Pressure Concept

Thus, optimal spark advance positions the pressure trace in a way that compromise between the effects mentioned above. To define the position of the in-cylinder pressure relative to TDC, the peak pressure position (PPP) is used, *Image 31*. The PPP is the position in crank angle where the in-cylinder pressure takes its maximal value. There also exist other ways of describing the positioning of the combustion relative to crank angle, e.g. based on the mass fraction burned curve.

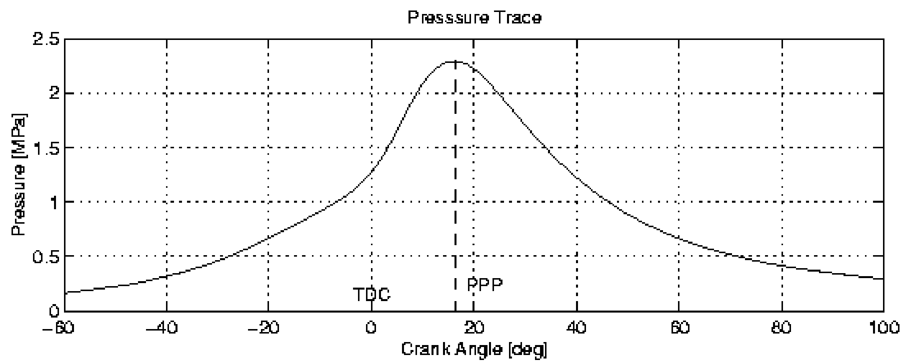


Image 9: Peak pressure position

Engine-tuning for efficiency

To be able to get the maximum torque from the engine at a given load point we have to investigate the torque development in different settings. In *Image 32*, mean values, over 200 cycles, of the PPP are plotted together with the mean value of the produced torque at four different operating points covering a large part of the road load operating range for the engine. Two of the operating points have an engine speed of 1500 rpm with different throttle angles, and for the two other operating points the engine speed is doubled to 3000 rpm. The PPP for maximum output torque in the figure lies around 15°ATDC (after TDC) for all these operating points, , even though the spark advance differs a lot.

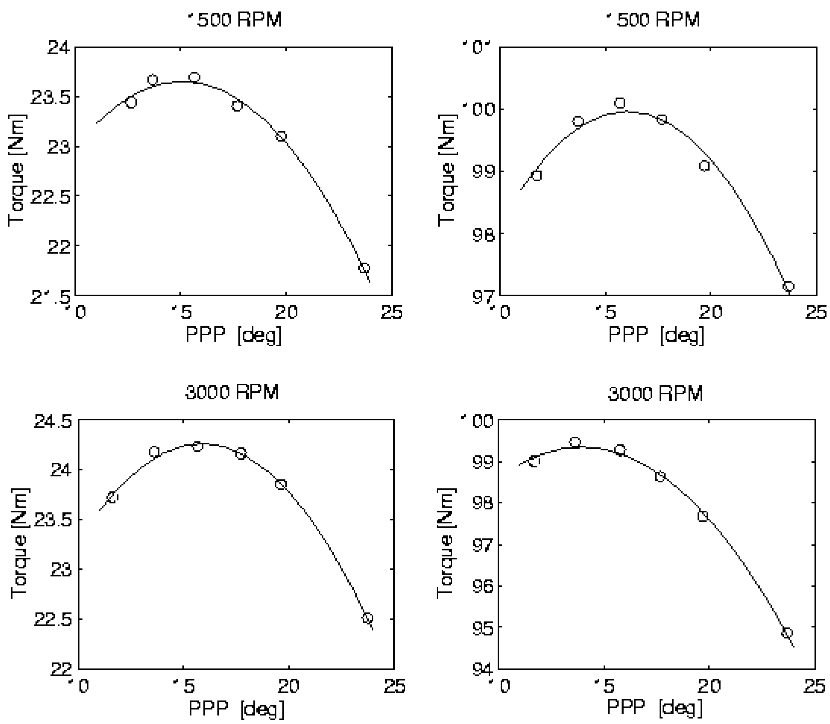


Image 10: PPP vs. torque in different settings

Note that the load and speed are changed over large intervals, and that the PPP for maximum output torque at the different operating points does not differ much. The PPP versus torque curve is flat around the position for the maximum. Therefore a spark schedule that maintains a constant PPP at 15° is close to optimum. Considering only the work produced, this motivates that an optimal spark schedule maintains almost the same position for the peak pressure. However, the optimal PPP changes slightly with the operating points. The efficiency can thus be improved a little bit further by mapping the optimal PPP for each operating point, and provide these values as reference signal to the spark timing controller. The peak pressure positioning principle can also be used for meeting emission standards.

Appendix VIII: Sensors and actuators

This appendix will list details about the sensors and actuators used in a T7 car.

General

Sensors are devices used to gather information. In a Trionic 7 car a lot of sensors are used to determine what actions to take inside the ECU. These sensors are all analogue, which means they output a signal that has to be converted to digital numbers for the ECU to be able to understand them. Actuators on the other hand are devices that enable the ECU to interact with the processes in the car. Actuators are driven (or activated) by the Trionic be it directly or indirectly.

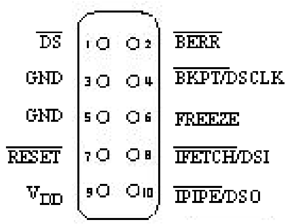
Sensors

Actuators

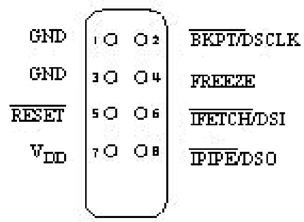
Appendix X: How to connect the PD BDM programmer to a T5/T7 ECU

Pin out

The standard BDM pin out:



The PD BDM/Willem pin out:



Please note that pins 1 & 2 on a standard BDM connector are not connected and that the row of pins are shifted one pin upwards. Pin 1 on the PD BDM/Willem is pin 3 on the standard BDM connector.

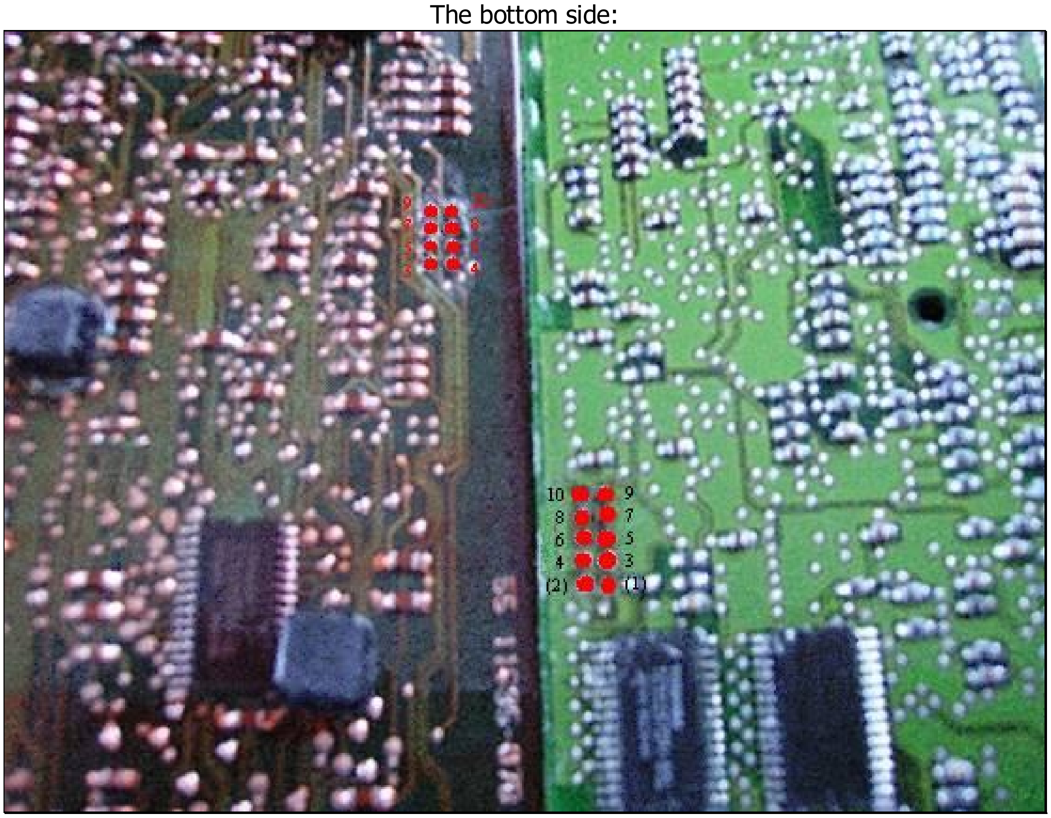
The BDM connector on the T5

On the T5 ECU there are eight pads for soldering the pins for the BDM programmer.

The BDM connector on the T7

Pay attention to that the BDM connectors are mirrored between the T5/T7 ECU. On the T7 ECU the pads for pin 1 & 2 are present but shall not be used when using a PD BDM or Willem programmer.

The bottom side:



On the other side of the PCB's there are only the other side of the pins. No bridging or something like that, just plain soldering.

Appendix XVI: Intercooler calculation

Description

This appendix will explain how to do calculations on intercooler flow capacity. A larger than stock intercooler is needed of you plan to go over 300 bhp with a Trionic 5 engine.

An intercooler is a heat exchanger. That means there are two or more fluids or gases that don't physically touch each other but a transfer heat or energy takes place between them. At wide open throttle and full boost the hot compressed air coming from a turbocharger is probably between 250 and 350 °F depending on the particular turbo, boost pressure, outside air temperature, etc.. We want to cool it down, which reduces its volume so we can pack more air molecules into the cylinders and reduce the engine's likelihood of detonation.

How does an intercooler work? Hot air from the turbo flows through tubes inside the intercooler. The turbo air transfers heat to the tubes, warming the tubes and cooling the turbo air. Outside air (or water in a watercooler intercooler) passes over the tubes and between fins that are attached to the tubes. Heat is transferred from the hot tubes and fins to the cool outside air. This heats the outside air while cooling the tubes. This is how the turbo air is cooled down. Heat goes from the turbo air to the tubes to the outside air.

There are some useful equations which will help us understand the factors involved in transferring heat. After we look at these equations and see what's important and what's not, we can talk about what all this means.

Equation 1

The first equation describes the overall heat transfer that occurs.
Q = U x A x DTlm

Q is the amount of energy that is transferred.
U is called the heat transfer coefficient. It is a measure of how well the exchanger transfers heat. The bigger the number, the better the transfer.
A is the heat transfer area, or the surface area of the intercooler tubes and fins that is exposed to the outside air.
DTlm is called the log mean temperature difference. It is an indication of the "driving force", or the overall average difference in temperature between the hot and cold fluids. The equation for this is:

$$DTlm = \frac{(DT1-DT2) * F}{\ln(DT1/DT2)}$$

where **DT1** = turbo air temperature in - outside air temperature out
DT2 = turbo air temperature out - outside air temperature in
F = a correction factor, see below

Note:

The outside air that passes through the fins on the passenger side of the intercooler comes out hotter than the air passing through the fins on the drivers side of the intercooler. If you captured the air passing through all the fins and mixed it up, the temperature of this mix is the "outside air temperature out".

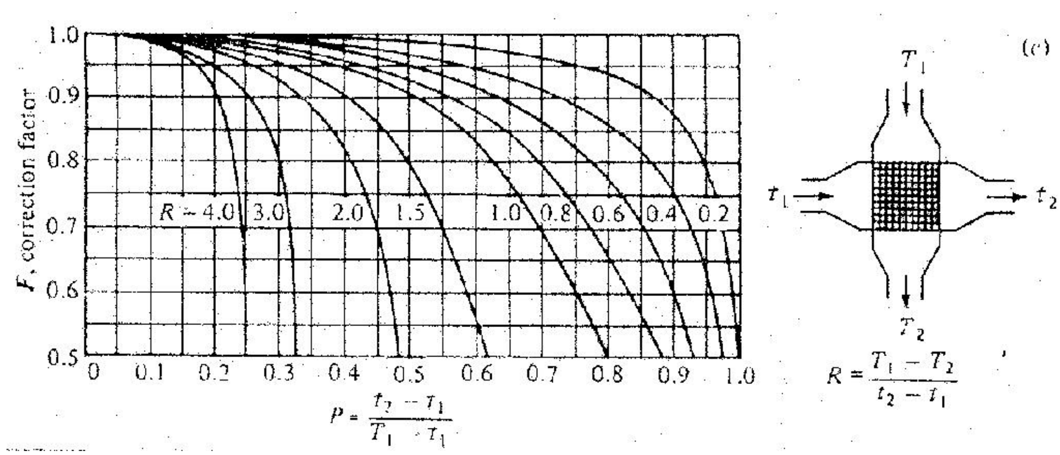
F is a correction factor that accounts for the fact that the cooling air coming out of the back of the intercooler is cooler on one side than the other.

To calculate this correction factor, calculate "P" and "R":

$$P = \frac{\text{turbo air temp out - turbo air temp in}}{\text{outside air temp in - turbo air temp in}}$$

$$R = \frac{\text{outside air temp in - outside air temp out}}{\text{turbo air temp out - turbo air temp in}}$$

Find P and R on "Fchart.jpg" (below) and read F off the left hand side.



This overall heat transfer equation shows us how to get better intercooler performance. To get colder air out of the intercooler we need to transfer more heat, or make Q bigger in other words. To make Q bigger we have to make U, A, or DTlm bigger, so that when you multiply them all together you get a bigger number. More on that later.

Equation 2

We also have an equation for checking the amount of heat lost or gained by the gas (or fluid) on one side of the heat exchanger (ie, just the turbo air or just the outside air):

Q = m x Cp x DT

Q is the energy transferred. It will have the exact same value as the Q in the first equation. If 5000 BTU are transferred from turbo air to outside air, then Q = 5000 for this equation AND the first equation.
m is the mass flow rate (lbs/minute) of fluid, in this case either turbo air or outside air depending on which side you're looking at.
Cp is the heat capacity of the air. This is a measure of the amount of energy that the fluid will absorb for every degree of temperature that it goes up. It is about 0.25 for air and 1.0 for water. Air doesn't do a great job of absorbing heat. If you put 10 BTU into a pound of air the temperature of it goes up about 40 degrees. If you put 10 BTU into a pound of water, the temperature only goes up about 10 degrees! Water is a great energy absorber. That's why we use water for radiators instead of some other fluid.
DT is the difference in temperature between the inlet and outlet. If the air is 200 deg going in and 125 deg coming out, then DT = 200 - 125 = 75. Again, on the cooling air side the outlet temperature is the average "mix" temperature.

If you know 3 of the 4 main variables on one side of the exchanger (the amount of heat transferred, the inlet and outlet temperatures, and the fluids flow rate) then this equation is used to figure out the 4th. For example, if you know the amount of heat transferred, the inlet temperature, and the flow rate you can calculate the outlet temperature. Since you can't measure everything, this equation is used to figure out what you don't know.

Caveat:
These equations are all for steady state heat transfer, which we probably don't really see too much under the conditions that we are most interested in – turbocharged engines! Cruising on the highway you would definitely see steady state.

So, now that we've got these equations, what do they **REALLY** tell us?

1. Heat transfer goes really well when there is a large temperature difference, or driving force, between the two gases. This is shown in equation 1 as a large DTlm. It doesn't go as well when there is a small temperature difference between the two gases (small DTlm). The closer you get the intercooler outlet temperature to the outside air temperature the smaller DTlm gets, which makes the heat transfer tougher.

2. The difference between the intercooler outlet temperature and the outside air temperature is called the approach. If it is 100 degrees outside and your intercooler cools the air going into the intake manifold down to 140 degrees, then you have an approach of 40 degrees (140 - 100 = 40). To get a better (smaller) approach you have to have more area or a better U, but there is a problem with diminishing returns. Lets rearrange the first equation to Q/DTlm = U x A. Every time DTlm goes down (get a better temperature approach) then Q goes up (transfer more heat, get a colder outlet temperature), and dividing Q by DTlm gets bigger a lot faster than U x A does. The upshot of that is we have a situation of diminishing returns; for every degree of a better approach you need more and more U x A to get there. Start with a 30 deg approach and go to 20 and you have to improve U x A by some amount, to go from 20 to 10 you need to increase U x A by an even bigger amount.

3. I would consider an approach of 20 degrees to be pretty good. In industrial heat exchangers it starts to get uneconomical to do better somewhere around there, the exchanger starts to get too big to justify the added expense. The one time I checked my car (stock turbo, stock IC, ported heads, bigger cam) I had an approach of about 60 deg. The only practical way of making the DTlm bigger on an existing intercooler is to only drive on cold days; if you buy a better intercooler you naturally get a better DTlm.

4. You can transfer more heat (and have cooler outlet temps) with more heat transfer area. That means buying a new intercooler with more tubes, more fins, longer tubes, or all three. This is what most aftermarket intercoolers strive for. Big front mounts, intercooler and a half, etc... are all increasing the area.

   A practical consideration is the fin count. The area of the fins is included in the heat transfer area; more fins means more area. If you try to pack too many fins into the intercooler the heat transfer area does go up, which is good, but the cooling air flow over the fins goes down, which is bad. Looking at the 2nd equation, Q = m * Cp * DT, when the fin count is too high then the air flow ("m") drops. For a given Q that you are trying to reach then you have to have a bigger DT, which means you have to heat up that air more. Then THAT affects the DTlm in the first equation, making it smaller, and lowering the overall heat transfer. So there is an optimum to be found. Starting off with bare tubes you add fins and the heat transfer goes up because you're increasing the area, and you keep adding fins until the it starts to choke off the cooling air flow and heat transfer starts going back down. At that point you have to add more tubes or make them longer to get more heat transfer out of the increased area.

5. Make U go up. You can increase the U by adding or improving "turbulators" inside the tubes. These are fins inside the tubes which cause the air to swirl inside the tube and makes it transfer its heat to the tube more efficiently. Our intercoolers have these, but I understand that more efficient designs are now available. One of the best ways to increase the U is to clean the tubes out! Oil film (from a bad turbo seal or from the stock valve cover breather) inside the tubes acts as an insulator or thermal barrier. It keeps heat from moving from the air to the tube wall. This is expressed in our equation as a lower U. Lower U means lower Qs which mean hotter turbo air temperatures coming out of the intercooler.

6. Air-to-water. If we use water as the cooling medium instead of outside air, we can see a big improvement for several reasons: Water can absorb more energy with a lower temperature rise. This improves our DTlm, makes it bigger, which makes Q go up and outlet temps go down. A well designed water cooled exchanger also has a much bigger U, which also helps Q go up. And since both DTlm and U went up, you can make the area A smaller which makes it easier to fit the intercooler in the engine compartment. Of course, there are some practical drawbacks. The need for a water circulation system is one. A big one is cooling the water down after it is heated (which means another radiator). This leads to another problem: You heat the water, and cool it down with outside air like the Syclone/Typhoon. You can't get it as cool as the outside air, but maybe you can get it within 20 degrees of it. Now you are cooling the turbo air with water that is 20 hotter than the outside air, and you can only get within 15 degrees of that temperature so coming out of the intercooler you have turbo air that is 35 degrees hotter than outside! (turbo air is 15 deg over water temp which is 20 deg over outside temp). You could have easily done that with an air to air intercooler! But... if you put ice water in your holding tank and circulate that... Then maybe the air temp coming out of the intercooler is 15 deg above that or 45 to 50 deg. Hang on! But after the water warms up, you're back to the hot air again. So, great for racing, not as good for the street.

7. Lower the inlet temperature. The less hard the turbo has to work to compress the air then the lower the temperature the air coming out of the turbo is. This actually hurts the DTlm, but still if it's cooler going in it will be cooler coming out. You can work the turbo less hard by running less boost, by improving the pressure drop between the air filter and the turbo, or by having a more efficient compressor wheel. You can also reduce the pressure drop in the intercooler, which allows you to run the same amount of boost in the intake manifold while having a lower turbo discharge pressure. More on this later. If you can drop the turbo outlet pressure by 2 psi, or raise the turbo inlet pressure by 1 psi, that will drop the turbo discharge temperature about 16 degrees (depending on the compression efficiency and boost level). If the turbo air is going into the intercooler 16 degrees colder then it may come out only 10 degrees colder than before, but that is still better than what it was.

Pressure drop

Another aspect of intercoolers to be considered is pressure drop. The pressure read by a boost gauge is the pressure in the intake manifold. It is not the same as the pressure that the turbocharger itself puts out. To get a fluid, such as air, to flow there must be a difference in pressure from one end to the other. Consider a straw that is sitting on the table. It doesn't having anything moving through it until you pick it up, stick it in your mouth, and change the pressure at one end (either by blowing or sucking). In the same way the turbo outlet pressure is higher than the intake manifold pressure, and will always be higher than the intake pressure, because there must be a pressure difference for the air to move.

The difference in pressure required for a given amount of air to move from turbo to intake manifold is an indication of the hydraulic restriction of the intercooler, the up pipe, and the throttle body. Let's say you are trying to move 255 gram/sec of air through a stock intercooler, up pipe, and throttle body and there is a 4 psi difference that is pushing it along. If your boost gauge reads 15 psi, that means the turbo is actually putting up 19 psi. Now we increase the amount of air travelling though to 450 gram/sec of air. At 15 psi boost in the intake manifold the turbo now has to put up 23 psi, because

the pressure drop required to get the higher air flow is now 8 psi instead of the 4 that we had before. More flow with the same equipment means higher pressure drop. So we put on a new front mount intercooler. It has a lower pressure drop, pressure drop is now 4 psi, so the turbo is putting up 19 psi again. Now we add the larger throttle body and the pressure drop is now 3 psi. Then we add the 3" up pipe, and it drops to 2.5 psi. Now to make 15 psi boost the turbo only has to put up 17.5 psi. The difference in turbo outlet temperature between 23 psi and 17.5 psi is about 40 deg (assuming a constant efficiency)! So you can see how just by reducing the pressure drop we can lower the temperatures while still running the same amount of boost.
Pressure drop is important because the higher the turbo charging pressure is, the higher the temperature of the turbo air. When we drop the turbo charging pressure we also drop the temperature of the air coming out of the turbo. When we do that we also drop the intercooler outlet temperature, although not as much, but every little bit helps. This lower pressure drop is part of the benefit offered by new, bigger front mount intercoolers; by bigger up pipes; and by bigger throttle bodies. You can also make the turbo work less hard by improving the inlet side to it. K&N air filters, these all reduce the pressure drop in the turbo inlet system which makes the compressor work less to produce the same boost which will reduce the turbo charge temperature.

Appendix XVII: Acronyms

Engine management specifics

Acronym	**Description**
ABS	Antilock Braking System
AMM	Air Mass Meter
BDM	Background Debug Mode
CANBUS	Controller Area Network (Car Area Network)
CPS	Crankshaft Position Sensor
DI	Direct Ignition
DICE	Dashboard Integrated Control Electronics
ECM	Engine Control Module
ECU	Engine Control Unit
EDU	Electronic Display Unit
FPR	Fuel Pressure Regulator
FPT	Full Pressure Turbo
HOT	High Output Turbo
IAT	Intake Air Temperature
LPT	Light Pressure Turbo / Line Printer Terminal
MAF	Mass Air Flow
MAP	Manifold Absolute Pressure
OBD	On Board Diagnostics
RAM	Random Access Memory
ROM	Read Only Memory
RPM	Revolutions Per Minute
SFI	Synchronous Flash Interface (Production test interface)
SID	System Information Display
TPS	Throttle Position Sensor
TWICE	Theft Warning Integrated Central Electronics
VSS	Vehicle Security System
WOT	Wide Open Throttle