Marelli Weber 16F Technical Information

Purpose of The Guide

To supply information on the operation of the electronic control unit (single point injection with static ignition) – IAW 16F and variants.

Defining the methods used to verify which component is faulty:

Point 1 – System Constitution

Point 2 – Operation of the System

Point 3 – Verifying functionality of individual defective components

Appendix 1 – Diagnostic Parameters of the system

Document Amendments

Amendment	Date	Description
-	Nov 1992	Edition 1 - New
A	Sept 1993	Models list updated. Text on 3.1 and 3.2 corrected See Appendix 1
B	June 1997	Appendix 1 amended to carry immobilizer information for IAW 16F. Corrected §A for model extensions
C	Oct 1997	Modernised with respect to 6F models, System Outline updated Text in 3.2.1 and point 1.2.3 and title of Appendix 1 amended
--	Nov 1997	Edition 2 – Diagnostic Protocol in Appendix 1 amended

§A - Applicable Models

CAR MODEL	NOTES	ISO CODE	MANF. CODE	SUPP. CODE
Cinquecento 899 SPI ECE F2		DO 85 85 94 43	46475180	61602.072.02
Cinquecento 1108 SPI ECE F2		DO 85 86 94 C4	46475181	61602.068.05
Panda 899 SPI ECE F2		DO 85 89 94 C7	46475176	61602.083.02
Panda 1000 SPI ECE F2		DO 85 08 94 46	46475177	61602.071.02
Panda 1108 4x4/4x4 ECE F2		DO 85 8A 94 C8	46475178	61602.070.02
Panda 1108 SPI CA ECE F2		DO 85 0B 94 49	46475179	61602.069.02
Punto 55 1.1 SPI 5M/6M ECE F2		DO 85 02 94 40	46480662	61602.102.00
Punto 60 1.2 SPI CM ECE F2 T.i.T.		DO 85 04 94 C2	46475175	61602.075.04
Punto Selecta 1.2 SPI ECE F2		DO 85 07 94 45	46467014	61602.079.03
Punto 1.1 SPI Em.04 "East Europe"		DO 85 01 94 BF	7787315	61602.058.01
Lancia Y "Young" 1.1 SPI ECE F2		DO 85 02 94 40	46480662	61602.102.00
Lancia Y 1.2 SPI CM ECE F2	See Point 1.2 of the attached Appendix 1	D0 85 8F 15 CE	46448299	61602.074.02
Lancia Y 1.2 SPI CA ECE F2		DO 85 10 15 4F	46448300	61602.073.02
Palio 1108 SPI Em.04 95 RON "Turkey"		Under Development	46463848	61602.091 .AA
Palio 1108 SPI Em.04 91 RON "South Africa"		Under Development	46463849	61602.092.AA
Palio 1108 SPI Em.04 90 RON "Morocco"		Under Development	46463850	61602.093.AA
Tipo 1372 SPI ECE 04 (TOFAS)		DO 85 91 16 51	46462900	61602.078.00
Tipo/Tempra 1.6 SPI USA'83 (TOFAS)		D0 85 0E 15 CD	46416940	61602.067.01
Tipo/Tempra 1.6 SPI Em.04 (TOFAS)		D0 85 8C 15 CB	46425974	61602.085.01
131 Bn/Sw1.6SPI USA'83 (TOFAS)		DO 85 92 16 52	46480845	61602.077.AA
Seicento 899 cc. SPI F2 Base		Under Development	46467015	61602.095.AA
Seicento 899 cc. SPI F2 Electronic Clutch		Under Development	46467016	61602.096.AA
Seicento 1108cc. SPI F2 Base + Condizione.		Under Development	46467017	61602.097.AA

NOTE: For vehicles in development or with undefined ISO Codes, use the IAW 16F Family ISO Code: 55-DO-85-80-80-2A.

1

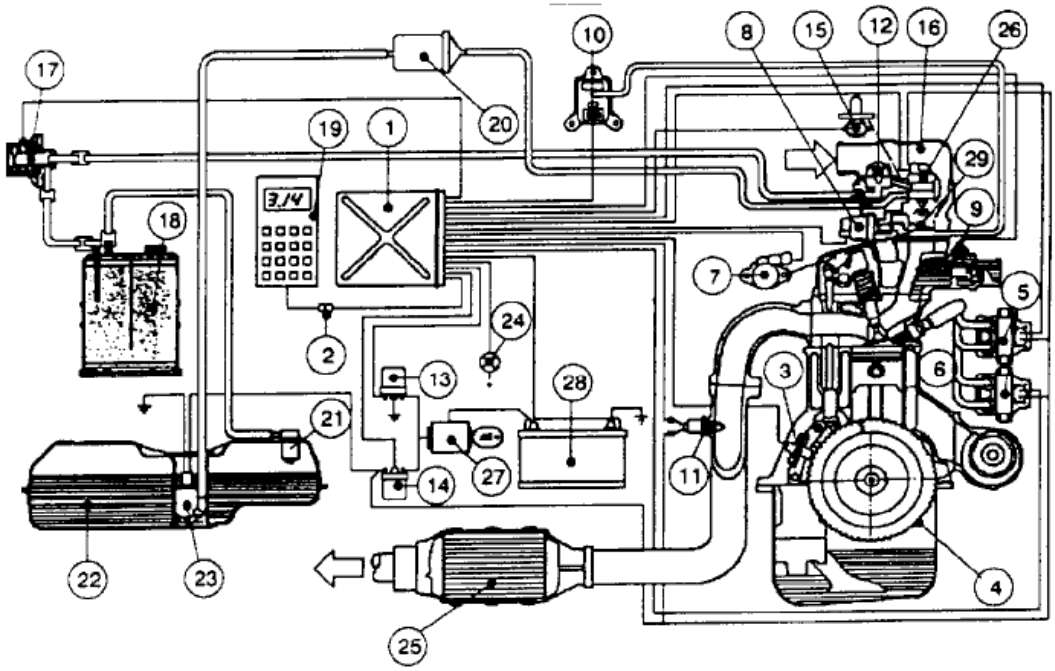
System Constitution

The system is constituted from the following members, their characteristics are in the following data sheets:

Number	Part	Document
1	ECU	CAP. 9.93240/01
2	ECU Wiring Loom	CAP. 9.95209/02
3	ECU System Warning Lamp	
4	Test Instrumentation	
5	Spark Plug	CAP. 9.92120
6	Ignition Coils	CAP. 9.93207
7	Idle Control Valve	CAP. 9.93239
8	Injector	CAP. 9.93231
9	Air Temperature Sensor	CAP. 9.93225
10	Water Temperature Sensor	CAP. 9.93225
11	Throttle Position Sensor	CAP. 9.93228/02
12	Manifold Absolute Pressure (MAP) Sensor	CAP. 9.93241
13	Oxygen (Lambda) Sensor	CAP. 9.93233/50
14	RPM / TDC Sensor	CAP. 9.93206
15	Charcoal Canister	CAP. 9.92605
16	Fuel Pump Relay	CAP. 9.92210

System Outline

M. Marelli 16F Family ECU system:



Key

1. ECU
2. Diagnostics Socket
3. RPM / TDC Sensor
4. TDC Sense Pulley (60 – 2 teeth)
5. Ignition Coils
6. Spark Plugs
7. Throttle Position Sensor
8. Idle Control Valve
9. Coolant Temperature Sensor
10. Manifold Absolute Pressure (MAP) Sensor
11. Lambda Sensor
12. Fuel Pressure Regulator
13. Relay
14. Relay
15. Air Temperature Sensor
16. Throttle Body
17. Fuel Vapour Release Valve
18. Charcoal Canister
19. Fiat-Lancia Tester or Equivalent
20. Fuel Filter
21. Multifunction Valve
22. Fuel Tank
23. Fuel Pump
24. ECU System Warning Light
25. Catalytic Converter
26. Fuel Injector
27. Ignition Amplifier
28. Battery
29. Butterfly (In Throttle Body)

2
OPERATION OF THE SYSTEM

The “IAW 16F” electronic control unit controls the ignition and the fuelling of the motor using closed loop “SPEED DENSITY” sensing.

2.1

Ignition

The ECU fires two ignition coils alternately, using the wasted spark principle. The rate of firing and ignition advance is dependent on absolute pressure (of intake charge) and engine speed.

The amount of ignition advance is calculated the using absolute pressure of the air in the inlet manifold, the engine speed and the amount of cut-off before TDC.

The ignition advance is variable, dependent on the following factors:

- Coolant Temperature
- Variation of Engine Speed
- Variation of Manifold Absolute Pressure

2.2

Fuelling

The ECU calculates the exact amount of fuel to send to the motor based on its speed and the absolute pressure in the manifold, carrying out corrections based on the air temperature, coolant temperature, throttle position and the feedback given from the lambda sensor for all fields of engine operation.

- **Synchronous** (used in conditions of normal operation of the motor);
- **Asynchronous** (used in specific (error) conditions).

These operating methods work together to manage the fuelling in all conditions: fuel cutting at high rpm (cut-off), idle control, management during transitions, correcting running conditions according to water temperature, air temperature, battery voltage, emissions control.

The ECU also controls the operation of the pump fuel (determining the amount of fuel to add in if the engine speed does not exceed a prefixed threshold in a prescribed time, or in case of emergency) and the Idle Control Valve.

2.3

Emissions

The ECU works to ensure one amount of the air-fuel mixture burns at the stoichiometric ratio in all the working conditions, aiming to ensure correct and long-lasting operation of the catalytic converter; the system alters itself to take into account the age of the engine and fuel quality

The system also controls the amount of fuel vapour in the exhaust and inlet by using the charcoal canister to redirect unburnt fuel into the inlet manifold, thusly, avoiding large variations of the mixture inside it.

2.4

Self Diagnostics

The unit performs passive self – diagnostics. It systematically tests the inputs and outputs of each of its sensors, and checks them for correct operation. Any errors are logged in the unit's EEPROM and the ECU System Warning light is lit up.

It is possible to test each actuator and sensor independently of the ECU system using an appropriate test instrument. Equally, an appropriate test instrument can modify / erase the contents of the EEPROM, e.g.: Tests may be carried out and specific errors cancelled.

The ECU warns the user of errors by lighting up the ECU System Warning Light if it detects a sensor \ actuator is not working correctly, replacement parts and / or repair may be necessary.

The EEPROM contents can only be erased using the diagnostic instrument. In error conditions the ECU will default to a safe state (ignoring most inputs) in order to keep the engine running. Under certain conditions, this is not possible e.g.: Failure of main power relay, Failure of RPM / TDC Sensor, Fuel Pump or fuel injector failure.

2.5

Other Functions

The ECU can be made to control other specific functions when necessary:

- Control of Air Conditioning Compressor,
- Fuel Pump firing 1 impulse every 180^{o} (Crankshaft) for analysis purposes,
- Sending an On-Off signal to an automatic gearbox ECU, or an EGR valve if coolant temperature reaches a threshold.

2.5.1

Connection to Immobilizer ECU

The unit can be connected to an "Electronic Key" Immobilizer ECU.

Once the procedure of memorizing keys into the Immobilizer ECU is complete, the unit must be connected to the Immobilizer ECU it was matched to.

The Immobilizer does not allow the starter motor to fire if the Transponder key does not match the codes held in the Immobilizer ECU.

Once the key is turned to the "MARCH" position, information is exchanged between the two ECU's; if the procedure completes correctly, the starter motor is allowed to work, otherwise it is not.

The communication is done on a serial line that links the two ECU's.

3
VERIFYING FUNCTIONALITY OF INDIVIDUAL DEFECTIVE COMPONENTS

3.1

If all physical connections to sensors and actuators are correct, and there is still an error in the system, other methods need to be employed.

This verification procedure is only 100% correct when the tests are carried out with the motor at idle as per the Production Standard 3.00093.

3.2

Test Procedure – Motor Diagnosis (Carry out at Idle)

This operation must be carried out with the aid of the equipment designed for testing ECU's that conform to FIAT xxx design.(IAW 16 yyyy), or with the aid of the FIAT—LANCIA Tester if other test equipment does not exist of if the aforesaid equipment has not received the correct modifications.

3.2.1

If the test equipment is the interface card to the FIAT $JFLT_2$ Standard, using the supplied programs, any unusual test results should be referred to Fiat.

D.T. — F.V. — S.I.E.E. — Experimentation - Self-diagnosis and Simulation.

If the equipment is lacking the interface card, D.T. can only supply the application software for the FLT.

1 DIAGNOSTIC CHARACTERISTICS OF SYSTEM

1.1 Diagnostic Table

PO = Key On, CK = Engine Starting , ER = Enginr Rotating, VR = Vehicle in Motion

Subgroup	Obtainable Data Type	Format	PO	CK	ER	VR	Notes
	Identification of the system:						**Protocol of communication with diagnosis instrument**:
	ISO Code Read	[6 Bytes/Hex]	*	*	*	*	
	Supplier Code Read	[11 Bytes/ACII]	*	*	*	*	The data is coded and transmitted to you in positive logic NRZcon at 7812.5Baud. The structure of the Bytes is following: 1 Start Bit (“0”), 8 Bit of Data, 1 Stop Bit (“1”), no parity. At the Power On, the ECU transmits to the 6 byte ISO Code at 1200baud. After at least 500 msec from the arrival of code ISO, the diagnosis instrument must transmit a code of acknowledgment to the ECU composed from three hexadecimal Bytes 0Fh, AAh, CCh, transmitted at 1200 speed of Baud with a time of interbytes (time between a Byte and successive) of 110 msec +/- 10 msec.
	Visualization of:	n° Bytes/valore	*	*	*	*	
	RPM	[2 Bytes/ Rpm]	*	*	*	*	
	Injector Duration	[2 Bytes/msec]	*	*	*	*	
	Ignition Advance	[1 Byte/°Ang.]	*	*	*	*	If this procedure has positive outcome, the ECU gets ready for the serial communication with instrument of diagnosis at 7812.5 Baud and modes as required (see PRODUCTION STANDARD).
	Manifold Absolute Pressure	[1 Byte/mmHg]	*	*	*	*	
	Air Temperature	[1 Byte / ° C]	*	*	*	*	The communication is like ISO_4: L line used L for initializiation and transferring Bytes from diagnosis instrument to ECU, and K line used in order to transfer Bytes from ECU to diagnosis instrument.
	Water Temperature	[1 Byte / ° C]	*	*	*	*	
	Throttle Position	[1 Byte/°Ang.]	*	*	*	*	
	Battery Voltage	[1 Byte/Volt]	*	*	*	*	**Acknowledgment of the errors**:
	Lambda Sensor Correction	[1 Byte / %]	*	*	*	*	
	Idle Stepper Position	[1 Byte / Step]	*	*	*	*	
	Minimum Integral Correction	[1 Byte / Step]	*	*	*	*	The error comes found through a validation process that works in the following way: When an error is found, this is held for a time in order to avoid the possible noises on the diagnosed line; if this stage passes it comes considered present and it is passsed to memory in ERR—CO—xx (filtered error), after that is passed to the successive phase in which it is controlled that the error is always present for an other time (validation time). If the error comes exceeded this phase it is passed to memory ERR—VA—xx (validated error) and, it lights the warning lamp. The phase of writing of the error in EEPROM happens, with the exception of the other systems, immediately (after 4 msec).
	Minimum Proportional Correction	[1 Byte / Step]	*	*	*	*	
Central Computer	Trimmer Title	[1 Byte]	*	*	*	*	
	ERR-CO-XX	[3Byte/Hex]	*	*	*	*	
	FGSTAT	[1Byte/Hex]	*	*	*	*	
	ERR-VA-XX	[3Byte/Hex]	*	*	*	*	
	Error Code (ECU Sensor Failed)	[11Bytes/ASCII]	*	*	*	*	
	Idle Stepper Correction	[2 Bytes / Step]	*	*	*	*	**Memorization of the errors**:
	Idle Stepper Correction w\Air Conditioning	[2 Bytes / Step]	*	*	*	*	
	Target Minimum RPM	[1 Byte/Rpm]	*	*	*	*	Beginning from the state of OK, an error activates a filter procedure that consists in sampling the line for a time T1. If to the term of T1 the number of incorrect results exceeds a prefixed value Q1, it exceeds the filter and gets written to RAM (ERR—CO—xx); if instead the filter stage is not passed, it is returned in state OK. In this condition the warning lamp doesnt light, because the error still is not validated.
	Difference between setpoint and current speed	[1 Byte/Rpm]	*	*	*	*	
	Minimum Stepper Value	[1 Byte / Step]	*	*	*	*	
	External Correction for ICV	[1 Byte / Step]	*	*	*	*	After the overcoming of the filter, a validation procedure is activated, that the line is sampled for a time T2. If the number of wrong results exceeds a prefixed value Q2, the error is validatted and written in memory E2PROM (ERR-VA-xx) and the warning light is lit
	FGSTAT2	[1 Byte/Hex]	*	*	*	*	
	SE-CO-XX, SE-VA-XX	[6 Bytes/Hex]	*	*	*	*	
	Stepper Offset below minimum - Charcoal Canister Off	[2 Bytes/msec]	*	*	*	*	
	Stepper Offset below minimum - Charcoal Canister On	[2 Bytes/msec]	*	*	*	*	If the procedure of validation of the error does not succeed, the content of RAM is cancelled and the system returns to OK condition. To every variable representative of confirmed or validated errors one variable with the sense of the errors will correspond: open circuit, short circuit to Gnd or + Vbatt. If a confirmed error or a sure sense is present and the ECU finds error of the opposite sense, then the process of validation of the error goes ahead but the the error is inverted. If it is instead present a validated error and the ECU finds an error of opposite polarity, the sense of the validated error does not come invert. The memorization happens alone if the error were not present previously. Therefore in case of errors of various type on the same line, the sign memory errors will contain the sign of the first validated error.
	Stepper Offset reduction	[2 Bytes/msec]	*	*	*	*	
	Stepper Offset reduction from midpoint	[2 Bytes / %]	*	*	*	*	
	Errors UNIVAS, EEVAS	[2 Bytes/Hex]	*	*	*	*	
	CRDVAS	[1 Byte/Hex]	*	*	*	*	
	Signalling of:						
	Error RAM (not overcoming test) — WL=On		*				The values of T1, Q1, T2, Q2 are specific values in calibration for every line.
	Error EPROM (wrong checksum of the memory ROM) — WL=On		*	*	*	*	Some “particularly serious” errors come immediately validated to the first survey and are logged in permanent memory; for these errors also the lamp is lit immediately
	Error EEPROM (wrong checksum memory EEPROM) - WL=On		*				
	Microprocessor Error (functionality of Micro) - WL=On		*	*	*	*	
	N.B. : **WL** = **ECU System Warning light lit**						
	Recovery:						
	System Emulation not successful						

Translated by: Arse
Date: 18/05/2005

Errors + Omissions Excepted

Subgroup	Type	Conv Formula	POR	CK	ER	VR
	RPM	(15 x 10^6)/$01$02				
	Injector Duration	(2 x $03 $04)/$10^3$	*	*	*	*
	Ignition Advance	$05/2	*	*	*	*
	Manifold Absolute Pressure	$06x3				
	Air Temperature	$07 — 40	*	*	*	*
	Water Temperature	$08 — 40	*	*	*	*
	Throttle Position	($09 x 4.234)-2.9638	*	*	*	*
	Battery Voltage	$0A x 0.0625	*	*	*	*
	Lambda Sensor Correction	($0B x 0,002656) + 0.66	*	*	*	*
	Idle Stepper Position	$0C	*	*	*	*
	Minimum Integral Correction	$0D/2	*	*	*	*
	Minimum Proportional Correction	$0E/2	*	*	*	*
	Trimmer Title	80H= 00, FFH=+127, 00H=-127	*	*	*	*
	ERR-CO-XX	$10 (INP), $11 (OUT),$12(FUNZ)	*	*	*	*
	FGSTAT	$13	*	*	*	*
	ERR-VA-XX	$14 (INP), $15 (OUT), $16 (FUNZ)	*	*	*	*
	Error Code (ECU Sensor Failed)	$17..$21	*	*	*	*
Central Computer	Idle Stepper Correction	$22$23	*	*	*	*
	Idle Stepper Correction w\Air Conditioning	$24$25	*	*	*	*
	Target Minimum RPM	$26x8	*	*	*	*
	Difference between setpoint and current speed	$27x8	*	*	*	*
	Minimum Stepper Value	$28 — 128	*	*	*	*
	External Correction for ICV	$29 — 128	*	*	*	*
	FGSTAT2	$2A	*	*	*	*
	SE-CO-XX, SE-VA-XX	$2B..$30	*	*	*	*
	Stepper Offset below minimum - Charcoal Canister Off	$32$33	*	*	*	*
	Stepper Offset below minimum - Charcoal Canister On	$34$35	*	*	*	*
	Stepper Offset reduction	$36$37	*	*	*	*
	Stepper Offset reduction from midpoint	$38$39	*	*	*	*
	Errors UNIVAS, EEVAS	$71 (UNIVAS), $72(EEVAS) -	*	*	*	*
	CRDVAS	$73 (valore max = FF Hex)	*	*	*	*
			*	*	*	*

Errors in system components:

Throttle Position Sensor	*	*	*	*
Manifold Absolute Pressure Sensor	*	*	*	*
Lambda Sensor			*	*
Water Temperature Sensor	*	*	*	*
Air Temperature Sensor	*	*	*	*
Battery Voltage	*	*	*	*
RPM/TDC Sensor		*		*
Injector Unit		*	*	*
Ignition Coil 1		*	*	*
Ignition Coil 2		*	*	*
Charcoal Canister			*	
Air Conditioner			*	
Fuel Pump	*	*	*	

Notes

Error Code Memory Locations:

ERR-CO-xx = FILTERED ERRORS

ERR-VA-xx = VALIDATED ERRORS

SE-CO- xx = Sign of the FILTERED ERRORS

SE-VA-xx = Sign of the VALIDATED ERRORS

FGSTAT/FGSTAT2 = SYSTEM STATUS

Two Immobilizer Specific bytes:

UNIVAS = FILTERED IMMOBILISER ERRORS

EEVAS = VALIDATED IMMOBILIZER ERRORS

Error Cancellation:

Error canceling can happen in 2 ways:

Through the diagnosis instrument (FLT) using specific commands from active diagnosis ($84 Hex)

Through continually incrementing Trip in calibration until value N-AVV (in calibration), when powering on cancels contents of RAM and from validated errors from the EEPROM

Immobilizer Errors are as follows: CRDVAS is set to FF Hex with a validated error and decremented of 1 to every Key. On error absence EEVAS is set to 0.

Warning Lamp Management:

In order to verify of the functionality of the system, the lamp is lit every Power—On, for 4 seconds. In the event of a validated error, the lamp will remain lit, and it will extinguish once the error passes. This is also logged in the EEPROM as a “history”.

Ignition of the Warning Lamp:

It is possible to define for every error whether the lamp is lit and/or limits are imposed on engine ratings (speed limitation). The lamp will activate, for the errors for which the activation is stated, the error is present in ERR-CO and ERR-VA-xx (error filtered and validated) and cleared once the error disappears:

WL = On=> ERR-CO-xx = On ERR-VA-xx = On

WL = Off => all the other conditions

Information on the State of System and the Motor:

2 Bytes exist that contain information on the state of the system and engine (FGSTAT, FGSTAT2)

FGSTAT	FGSTAT2
Bit_0 = 1 Diagnostics OK, stepper in Run and TH20 OK	Bit_0 = Free (free)
Bit_1 = 1 Motor rotating	Bit_1 = Free (free)
Bit_2 = 1 Square Wave Signal OK	Bit_2 = 1 Stepper Calibration OK
Bit_3 = 1 ICV at Minimum or Maximum	Bit_3 = Free (free)
Bit_4 = 1 Operating in Closed Loop	Bit_4 = Free (free)
Bit_5 = 1 Air conditioner On	Bit_5 = Free (free)
Bit_6 = 1 System Calibration OK	Bit_6 = 1 Power line failure
Bit_7 = 1 Testing Stepper Motor	Bit_7 = Free (free)

Subgroup	Obtainable Data	PO	CK	ER	VR	Notes	Error Memory Coding	
Central Computer	Stepper Motor Control	*	*	*	*		**[ERR-CO/ERR-VA] _INP:**	**[ERR-CO/ERR-VA] _OUT:**
	RAM Error	*						
	EPROM Error	*	*	*	*		Bit_0 = 1 Throttle Position Sensor Error	Bit_0 = 1 Injector Error
	EEPROM Error	*					Bit_1 = 1 MAP Sensor Error	Bit_1 = 1 Ignition Coil 1 Error
	uC (Microcontroller) Error	*	*	*	*		Bit_2 = 1 Lambda Probe Error (KO2 outside range)	Bit_2 = 1 Ignition Coil 2 Error
	Immobilizer	*					Bit_3 = 1 Water Temperature Sensor Error	Bit_3 = 1 ICV Error
							Bit_4 = 1 Air Temperature Sensor Error	Bit_4 = 1 Charcoal Canister Error
	Memory Retained Errors:						Bit_5 = 1 Battery Voltage Error	Bit_5 = 1 Air Conditioner Error
							Bit_6 = 1 Estimate Error Steps (NU)	Bit_6 = 1 Fuel Pump Error
	Autocalibration Errors			*	*		Bit_7 = 1 NU	Bit_7 = 1 Relay Error (NU)
							[ERR-CO / ERR-VA]_FUN :	**[SE-CO/SE-VA]_FUN:**
							Bit_0 = 1 Calibration Error	Bit_0 = 1 Calibration Parameters too Rich
							Bit_1 = 1 RAM Error	Bit_0 = 0 Calibration Parameters too Lean
							Bit_2 = 1 EPROM Error	Bit_1 = 1 NU
							Bit_3 = 1 EEPROM Error	Bit_2 = 1 NU
							Bit_4 = 1 uC (Microcontroller) Error	Bit_3 = 1 NU
							Bit_5 = 1 RPM / TDC Sensor Error	Bit_4 = 1 NU
							Bit_6 = 1 NU	Bit_5 = 1 NU
							Bit_7 = 1 NU	Bit_6 = 1 NU
								Bit_7 = 1 NU
							[SE-CO/SE-VA]_OUT:	**[SE-CO/SE-VA]_INP:**
							Bit_0 = 1 Injector Short circuited	Bit_0 = 1 Throttle Position Sensor shorted to Gnd
							Bit_0 = 0 Injector Open circuited	Bit_0 = 0 Throttle Position Sensor shorted to Vbatt
							Bit_1 = 1 Coil 1 Shorted to Vbatt	Bit_1 = 1 MAP Sensor shorted to Vbatt
							Bit_1 = 0 Coil 1 Shorted to Gnd	Bit_1 = 0 MAP Sensor shorted to Gnd
							Bit_2 = 1 Coil 2 Shorted to Vbatt	Bit_2 = 1 NU
							Bit_2 = 0 Coil 1 Shorted to Gnd	Bit_2 = 0 NU
							Bit_3 = 1 ICV Stepper Motor Short circuited	Bit_3 = 1 Water Temperature Sensor shorted to Gnd
							Bit_3 = 0 ICV Stepper Motor Open Circuited	Bit_3 = 0 Water Temperature Sensor shorted to Vbatt
							Bit_4 = 1 Charcoal Canister shorted to Vbatt	Bit_4 = 1 Air Temperature Sensor shorted to Gnd
							Bit_4 = 0 Charcoal Canister shorted to Gnd	Bit_4 = 0 Air Temperature Sensor shorted to Vbatt
							Bit_5 = 1 Air Conditioner shorted to Vbatt	Bit_5 = 1 Vbatt larger than upper threshold
							Bit_5 = 0 Air Conditioner shorted to Gnd	Bit_5 = 0 Vbatt lower than lower threshold
							Bit_6 = 1 Fuel pump shorted to Vbatt	Bit_6 = 1 NU
							Bit_6 = 0 Fuel pump shorted to Gnd	Bit_6 = 0 NU
							Bit_7 = 1 Generic Relay shorted to Vbatt	Bit_7 = 1 NU

Subgroup	Obtainable Data Type	Format	PO	CK	ER	VR	Notes
Air Temperature Sensor	**Visualisation of:**						**Test Method:** The air temperature value returned by the sensor is tested to check it is within certain limits. If it is wrong, a default value will be used, or the current value if it is higher than the test result
	Air Temperature	[oC]	*	*	*	*	Shorted to Ground: TAIRA < MIN.TAIR Shorted to Vbatt: TAIRA > MAX.TAIR
	Signalling of:						Frequency of the test 100 msec.
	Shorting to Ground		*	*	*	*	**ECU Warning Light:**
	Shorting to Vbatt		*	*	*	*	Warning Light lit if ERR-CO-INP =10 Hex + ERR-VA-INP =10 Hex
	Conversion Formulae:						**Recovery:**
	Air Temperature = DATA-40	[oC]					If the error is temporary uses MT.AIR.L = MT.AIR.Lold (last valid value); If the error is confirmed with TH2O < DEF.TAI, error is logged and MT.AIR.L = T_H2O used
	DATA = $07 Hex	[1 Byte]					otherwise MT.AIR.L = DEF.TAIR;
Battery Voltage	**Visualisation of:**						**Test Method:** The battery voltage is tested to verify that the value is in the range of the permissible values. If the result is incorrect, a default value of will be taken if the ECU is starting the engine and an other if the engine is already started.
	Battery Voltage						Battery voltage over maximum threshold: V.BATT * MAX.VBAT (16.2 V)
	Signalling of:						Battery voltage under minimum threshold or power failure:
	Battery Voltage exceeding upper limit	[V]	*	*	*	*	V.BATT * MIN.VBAT (6V)
	Vbatt lower than lower limit or power failure						Frequency of the test 12 msec.
	Conversion Formulae:		*	*	*	*	**ECU Warning Light:**
			*	*	*	*	Off, but validated error if ERR-CO-INP = Hex 20 + ERR-VA-INP = Hex 20
	Battery Voltage = (DATA x 0.4234) - 2.9638	[V]					
	DATA = $0A Hex	[1 Byte]					**Recovery:** If the error then introduces during the starting phase V.BATT = V1. VBATT Otherwise V.BATT = V2.VBATT

Subgroup	Obtainable Data Type	Format	POK	CKER	VR	Notes
RPM / TDC Sensor	**Visualisation of:**					**Test Method:**
	Engine Speed	[rpm]	*	*	*	The time between MAXP and MINP will reduce as engine speed increases. CNMAP is the time between maxima and minima. NMAP (fixed value) may be used as an absolutely last resort if CNMAP becomes inaccurate . This is stopped if the RPM Sensor begins responding
	Signalling of:					Frequency of the test is every pulley tooth.
	Signal Not Present		*	*	*	**ECU Warning Lamp:**
	Conversion Formulae:					On, validated error if: ERR-CO-FUN = Hex 20 + ERR-VA-FUN = Hex 20
	Engine Speed = $(15x10^6)$/DATA	[rpm]				**Recovery:**
	DATA = $01$02 Hex	[2 Bytes]				None Possible

Signal Not Present:

If at Key-On condition, the ECU finds no signal from the RPM Sensor, it samples the MAP Sensor and if the throttle is open, immobilizer OK and key in On position. The MAP Sensor is continually sampled until an incremental decrease DELP1 is seen, this is sampled until a minimum pressure, MINP is reached. This is continued until a maximum, MAXP is found.

Subgroup	Obtainable Data Type	Format	POK	CER	VR	Notes Test Method:
						Lambda Sensor Failed:
						Whilst direct Lambda probe output tests are not possible, if the conditions for Closed-Loop operation are met, the ECU presumes that the lambda probe output is within limits: The system remains in closed loop operation unless the KO2 is found to be outside limits or the lambda sensor disappears. Errors are found if the probe signal is absent or outside limits or if KO2 is out of range. The confirmed error becomes invalid if the probe starts responding again.
	Visualisation of:					
Lambda Sensor	Lambda Sensor Status (Open / Closed Loop) via FGSTAT			*	*	Shorted to VBATT: Probe Failure recognised when TENS.SO2 * SLAMBCC
	KO2			*	*	Cut Off + System Running Rich; Acceleration + System Running Lean: If cut off stays for a time and the system is always rich or if under acceleration the system is always lean, probe failure is logged
	Signalling of:					KO2 outside range: If KO2 moves out of range for a set time, Probe Failure is logged
						Frequency of the every test 100 msec.
	Lambda Signal out of Range			*	*	
	Conversion Formulae:					**ECU Warning Lamp:**
	Sensor Status = [FGSTAT = $13], 10 Hex = Closed Loop					Off, validated error if ERR-CO-INP = Hex 04 + ERR-VA-INP = Hex 04
	KO2 FDT=$0Bx0.002656+0.66					**Recovery:**
						In the event of probe in failure the system is already in Open-Loop and
						In the event of KO2 going out of limits, the system is forced into Open-Loop and KO2 = Hex. 80 is used

Subgroup	Obtainable Data Type	Format	POR	CKR	ERR	VRR	Notes
							Test Method:
							The MAP output is tested to check it is within expected values for given Throttle Positions and Engine Speeds. If the value is incorrect, a default is assumed according to:
							Shorted to Gnd:
	Visualisation of:						Pressure P.ATM < S.PATM
							Pressure P2 < MIN.PRE
	Atmospheric Pressure		*				
	Manifold Presure			*	*	*	**Shorted to Vbatt:**
							Pressure P2>MAZ.PRE is Not Feasible.
	Signalling of:						
MAP Sensor							If the engine is running, and there is no Throttle Position Sensor Error: (MP2.mb8 - P.AN.PE) > DELT.PRE
	Shorting to Ground		*	*	*	*	
	Shorting to Vbatt		*	*	*	*	If PERIOD > S1.PRE then P2 < MIN.MP2;
							If PERIOD < S1.PRE then P2 > MIN.MP2;
	Conversion Formulae:						
							Frequency of the test is every 4 msec, then every Power-On and 90° in Engine-Running mode
	Manifold Pressure = DATA x 3	[mmHg]					
	DATA = $06Hex	[1 Byte]					**ECU Warning Lamp:**
							On, validated error if ERR-CO-INP = Hex 02 + ERR-VA-INP = Hex 02
							Recovery:
							If the error is temporary then MP2mb = MP2mbold (last valid value)
							If the error is confirmed then P.ATM = 1012 mbar; If Error present on ICV then MP2.mb = DEF.MP2 * 4
							If the error is present at Key-On and PERIOD > S.DEMAR (motor not started), then MP2.mb = 1012 mbar
							If the motor is started then MP2.mb = 4 * P.AN.PE (function of the ICV and RPM/TDC Outputs)

Subgroup	Obtainable Data: Type	Format	PO	CK	ER	VR	Notes
							Test Method:
							Shorting to Gnd:
							Throttle angle PARACQ < MIN.PAP
Throttle Potentiometer	**Visualisation of:**						**Shorting to Vbatt :**
							Throttle Angle PARACQ > MAX.PAP
	Throttle Position	[° Ang]	*	*	*	*	
							Frequency of the test 4 msec.
	Signalling of:						
							ECU Warning Light:
	Shorting to Ground		*	*	*	*	On, if ERR-CO-INP =01 Hex + ERR-VA-INP =01 Hex
	Shorting to Vbatt		*	*	*	*	
							Recovery:
	Conversion Formulae:						If the error is temporary it uses last valid value ANG.PAPO = ANG.PAP1
							If the error is confirmed it uses ANG.MIN = ANG.MIND and ANG.MAX = ANG.MAXD
	Throttle Position = DATA x 0.4234 - 2.9638	[° Ang]					If MP2mb8 < SP2.PAP or SPEED < SPEED.PAP then ANG.PAPO = ANG.PAP1 = ANG.MIN
	DATA = $09 Hex	[1 Byte]					If MP2mb8 > SP2.PAP and SPEED > SPEED.PAP then ANG.PAPO = ANG.PAP1 = VDEF.PAP
							In the Event of the TPS Failing, the signal is composed of a composite of the MAP Sensor and the RPM sensor, with the function eventually trying to make the TP zero wrt the position of the Idle Stepper Motor
							Test Method:
							Shorting to Gnd:
							Water Temp. T.H2O * T.H2OMIN
Water Temperature Sen	**Visualisation of:**						
							Shorting to Vbatt :
	Water Temperature	[oC]	*	*	*	*	Water Temp. T.H2O * T.H2OMAX
	Signalling of:						Frequency of the test 100 msec.
	Shorting to Ground		*	*	*	*	**ECU Warning Light:**
	Shorting to Vbatt		*	*	*	*	On, validated error if ERR-CO-INP = Hex 08 + ERR-VA-INP = Hex 08
	Conversion Formulae:						**Recovery:**
							If the error is temporary MT.H2O.L = MT.H2O.Lold (last valid value)
	Water Temperature = DATA - 40	[oC]					MT.H2O.L = MT.AIR.L if an eror is found error on first read
	DATA = $08 Hex	[1 Byte]					MT.H2O.L = DEF.TH2O if an error is found in the ATS
							If the error is confirmed then MT.H2O.L < DEF.TH2O is used. This becomes MT.H2O.L for 1 in every 8 samples of NBR.TH2Ox

Subgroup	Obtainable Data Type	Format	POK	CKER	VR	Notes
Calibration Parameters	**Visualisation of:**					**Test Method:** The Calibration values are tested to check they are within acceptable limits
	ADOFFSET			*	*	**At Maximum Threshold:** gain ADGAIN > MAX.GAIN
	ADOFFPURG			*	*	offset ADOFFSET > MAX.OFF
	ADOFFPL			*	*	offset ADOFFPURG > MAX.OFPU
	ADGAIN			*	*	offset ADOFFPL > MAX.OFPL
	Signalling of:					**At Minimum Threshold:** gain ADGAIN < MIN.GAIN
	Calibration Parameters reaching maximum threshold		*	*	*	* offset ADOFFSET < MIN. OFF
	Calibration Parameters reaching minimum threshold		*	*	*	* offset ADOFFPURG < MIN.OFPU
	Conversion Formulae:					offset ADOFFPL < MIN.OFPL
						Frequency of Test: Every time system is calibrated
	ADOFFSET = $32$33					
	ADOFFPURG = $34$35					**ECU Warning Light:**
	ADOFFPL = $36$37					Off, validated error if ERR-CO-FUN =01 Hex + ERR-VA-FUN =01 Hex
	ADGAIN = $38$39					
						Recovery:
						None Possible, Values are automatically configured
Charcoal Canister						**Test Method:**
						Shorting to Gnd:
						In the Canister Off state, the ECU tests that the Canister signal is high once per engine cycle, and logs errors otherwise
						Shorting to Vbatt:
	Signalling of:					In the Canister On state, the ECU tests that the Canister signal is low once per engine cycle, and logs errors otherwise
	Shorting to Ground			*	*	
	Shorting to Vbatt			*	*	Test Frequency: Every 100msec
	Activation					**ECU Warning Light:**
						Off, validated error if ERR-CO-OUT = Hex 08 + ERR-VA-OUT = Hex 08
	Activated for 20msec in every second, 7 times		*			
						Recovery:
						Canister Control Inhibited
						Canister Calibration Inhibited

Subgroup	Obtainable Data Type	Format	POK	CKER	VR	Notes
						Test Method:
						The ECU has a feedback device in it that it uses to check that the current in the ignition coil rises to 4.5A within the time the ECU is programmed to expect it to. If this time is too long the coil has probably shorted to Vbatt, if the time is too short, the coil will have shorted to Gnd. In either case, errors are logged. The same is valid if the voltage is outside the range expected
Ingnition Coils	**Visualisation of:**					**Shorting to Vbatt:** if TC.MES (1-2) < TMPMIN and MV.BATT < BAT.MAX
	Shorting to Ground			*	*	**Shorting to Gnd:**
	Shorting to Vbatt			*	*	if TC.MES (1-2) > TMPMAX and MV.BATT > BAT.MIN
	Activation:					Frequency of Test: Every time current data is fedback and each interrupt of the coil Off command
	Initial Activation: 20msec in every second, 5 times		*			**ECU Warning Light:** *On, Validated Error For Coil 1 if : ERR-CO-OUT = 02 Hex + ERR-VA-OUT = 02 Hex* *On, Validated Error For Coil 2 if : ERR-CO-OUT = 04 Hex + ERR-VA-OUT = 04 Hex*
						Recovery: Short to Vbatt: Command stays open, tries to turn engine again. Short to Gnd: Command closed, default value assumed
Fuel Injector	**Visualisation of:**					**Test Method:**
	Injector Opening Duration	[msec]		* * *		The ECU has a feedback element that checks that the injector is opening and closing correctly,or if it is shorted, or the fuel is too hot. If a eror situation is detected, the fuel is automatically cut off and the error is logged. The memory in polled by an INTERRUPT every 131msec, and this defines the state of the fuelling system to the error validation system, resetting it as necessary, allowing the injector to fire. This only happens when the engine is turning, and the current data from the feedback element is different to the current memory contents.
	Signalling of:					
	Open Circuit			* * *		If the injector shorts to Gnd, the fuel pump may be shut off. This happens if: NIB.INJ [CONF.TAR] = 1. Error Confirmation happens when the injector is next fired (and after 180° in synchronous management) and the error is validated after O_VAL.
	Short Circuit to Vbatt. Gnd, or Thermal Protection			* * *		
	Activation:					**All Conditions:**
	Activated for 4msec in every second, 5 times		*			Using dedicated hardware
	Conversion Formulae:					Test Frequency: Every 100msec
	Injector Duration = DATA / 500	[msec]				**ECU Warning Light:**
	DATA = $03$04 Hex	[1 Byte]				On, validated error if ERR-CO-OUT = 01 Hex + ERR-VA-OUT = 01 Hex
						Recovery: **Open Circuit:** Injector command shut off **Other Condition:** Fuel Pump shut off

Subgroup	Obtainable Data: Type	Format	PO	CK	ER	VR	Notes
	Visualisation of:						**Test Method:**
Idle Control Valve	Stepper Position		*	*	*	*	The ECU has a feedback device in fuelling stage that it uses to check that the Stepper is functioning correctly. In an error situation, the ECU automatically disables the Idle Stepper and logs an error to its memory. This is polled by the INTRPT every 131 msec, and if the sensor status is no longer erroneous, the error is cleared and normal operation resumed.
	Integral Value for Stepper		*	*	*	*	
	Proportional Value for Stepper		*	*	*	*	
							All Cases:
	Signalling of:						Using Specialised equipment
			*	*	*	*	
	Open Circuit		*	*	*	*	Frequency of Test: Every 100 msec
	Short Circuit to Vbatt. Gnd, or Thermal Protection						
							ECU Warning Light:
	Activation:						Off, Validated Error if : ERR-CO-OUT = 08 Hex + ERR-VA-OUT = 08 Hex
	Activated 32 steps +/- from neutral				*		**Recovery:**
							Stepper Motor Disabled
	Conversion Formulae:						Prohibit Charcoal Canister Calibration
							Limit Engine Speed to 1200rpm
	Stepper Position [steps]	FDT = $0C					
	Integ. Val [steps]	FDT = $0D / 2					
	Propnl. Val [steps]	FDT = $0E / 2					
							Test Method:
							The test is performed when the air conditioner is turned on. There is a feedback element equipped part of the ECU that will continually sample the signal to check whether it is permanent or not
							Shorting to Gnd:
Air Conditioner Relay	**Signalling of:**						If the A/C is On and the Recirculation is Off, the ECU will check for the A/C's signal to be maintained as High, otherwise an error is logged.
	Shorted to Gnd				*	*	**Shorting to Vbatt:**
	Shorted to Vbatt				*	*	If the Recirculation is On, the ECU will check for the A/C's signal to be maintained as Low, otherwise an error is logged.
	Activation:						
							Test Frequency: Every 131 msec
	Relay activates for 30 sec.		*				
							ECU Warning Light:
							Off, Validated Error if : ERR-CO-OUT = 20 Hex + ERR-VA-OUT = 20 Hex
							Recovery:
							Shorting to Vbatt: Control of A/C lost, temperature difference maintained

Subgroup	Obtainable Data Type	Format	PO	CK	ER	VR	Notes
Fuel Pump Relay	**Signalling of:**						**Test Method:** The ECU has a feedback device that constantly samples the signal and checks it for errors **Shorting to Gnd:** With key in Off position (Motor will stop running shortly), If the Relay level is high, an error is logged **Shorting to Vbatt:** With Key in On position (motor running), If the relay level is low, errors are logged Frequency of Test: Every 131 msec **ECU Warning Light:** On, Validated Error if : ERR-CO-OUT = 40 Hex + ERR-VA-OUT = 40 Hex **Recovery:** Shorting to Vbatt: Fuel Pump Shut down Prohibit diagnosis of coil, injector, lambda and canister
	Shorted to Gnd		*	*	*	*	
	Shorted to Vbatt		*	*	*	*	
	Activation:						
	Relay activates for 30 sec.		*				
RAM	**Signalling of:**						**Test Method:** At power on the RAM is tested to write and read characters $55 to $AA, after this the RAM stands by and the first 8 bytes of RAM data are verified to correspond with the VERSION variable and the adapted values are in the ranges: Offset MIADO < adapted * MXADO Adapted MIADG < gain * MXADG The 2 writing and reading cycle is repeated on any incoherent areas, and the coherent areas are cleared. In an error situation in Standby, the contents are dumped to the STBY table, on errors in VERSION, the contents are dumped to the VERSION table along with the contents of ER-CO-FUN and ER-PR-FUN Test Frequency: Every Power on **ECU Warning Light:** Off, Validated Error if : ERR-CO-FUN = 02 Hex + ERR-VA-FUN = 02 Hex **Recovery:** Write in the values in Standby mode
	RAM Checksum Incorrect (writing test reading $55 - $AA)		*				

Subgroup	Type	Obtainable Data Format	POR	CK	ER	VR	Notes
EEPROM	**Signalling of:** Hardware Checksum Wrong		*				**Test Method:** With the key in STOP position, the EEPROM contents are copied to RAM. The EEPROM is dedicated to hardware calibration and has 3 uniform areas that all have the checksum 55AA Hex. Any write errors or area inconsistancies trigger errors to be logged Frequency of Test: Every Power On
	RAM and EEPROM Checksum not matching (On every power on after initial failure)		*				**ECU Warning Light:** Off, Validated Error if : ERR-CO-FUN = 08 Hex + ERR-VA-FUN = 08 Hex **Recovery:** Default EEPROM values written in
EPROM	**Signalling of:** EPROM Checksum Incorrect		*	*	*	*	**Test Method:** The ROM contents are checked against the CHECKROM contents at startup. Any inconsistancies cause errors to be logged Frequency of Test: Every Power On **ECU Warning Light:** Off, Validated Error if : ERR-CO-FUN = 04 Hex + ERR-VA-FUN = 04 Hex **Recovery:** No Recovery Possible
Microprocessor	**Signalling of:** Microprocessor Error		*	*	*	*	**Test Method:** Interrupt caused by Illegal Op-Code Interrupt caused by Reset Watchdog Interrupt caused by Illegal Carrier (Signal Input) Interrupt caused by clock malfunction Test Frequency: Every Power on and every microprocessor interrupt **ECU Warning Light:** Off, Validated Error if : ERR-CO-FUN = 10 Hex + ERR-VA-FUN = 10 Hex **Recovery:** RESET the Microcontroller

Subgroup	Obtainable Data Type	Format	POR	CK	ER	VR	Notes
							Test Method:
							Uncoded ECU: The ECU is Uncoded and cannot accept key memorizing
							Received Universal Code: The Immobiliser ECU sends the Engine ECU the universal code on request. Both are effectively uncoded. The ECU Warning lamp will flash once with a 50% duty cycle (on for 0.5 sec in 1 sec) to show this status
							No Code Received / Link Down: If serial comms is interrupted (eg: by an open or short circuit) or if the immobilizer is bypassed, this error will result
	Visualisation of:						
	RAM Errors / RAM Status (UNIVAS)		*	*	*	*	**Incorrect / Unrecognised Code:** Where the Engine ECU code doesn't match the code sent by the Immobilzer ECU. Usually caused by two pre-programmed units
	Immobilizer EEPROM Errors (EEVAS)		*				
	Number of CRDRVAS Errors						
	Signalling of:						**Wrong Key Code:** If the Transponder is malfunctioning, the reader is broken or the key is memorised to a different ECU, this error will result
Immobilizer	ECU uncoded		*	*	*	*	
	Start Allowed / Disallowed?		*				
	ECU sent the "Universal Code"		*				**Disallowed Start:** If the codes held by the ECUs match, the start is allowed, if the codes do not match (or errors are present), the start is disallowed
	No Code Received / Link Down		*				
	Incorrect / Unrecognised code		*				
	Wrong Code sent from Key (Transponder)		*				
							Test Frequency: Every Power On
	Activation:						**ECU Warning Light:**
	Recovery Procedure to allow engine to start		*				On, DTC Code: U1600 (At same time as CODE Light)
	Conversion Formulae:						**Recovery:**
	UNIVAS = [FLAGS]	$71					Start Procedure through Accelerator Pedal
	EEVAS = [FLAGS]	$72					Send ECU Code through Test Instrument
	CRDRVAS =	$73					**Immobilizer Coding:**
							Bit_0 = 1 ECU is Uncoded
							Bit_1 = 1 Start Disallowed
							Bit_1 = 0 Start allowed
							Bit_2 = 1 ECU has received Universal Code
							Bit_2 = 0 ECU has not received universal code
							Bit_3 = Reserved Bit
							Bit_4 = 1 Received Backdoor Code
							Bit_5 = 1 Wrong Key Code (101010101 from immobilizer)
							Bit_6 = 1 Incorrect / Unrecognised Code
							Bit 7 = 1 No Code Received / Link Down

1.2

Communication Line Configuration and Initialisation

1.2.1

Lines Used

ECU Communication is done on two lines, Line L to send data to the ECU and Line K to receive data from the ECU.

Communication is to an ISO 4 Standard, the L line is used to transmit data to the ECU and the K line is used to receive data from it. Line G, if available is used to send communication data down.

The ECU Pinout is as follows:

- Pin number 10 – Line "L"
- Pin number 15 – Line "K"

1.2.2

Initialisation

The ECU must be initialised at least 500 msec after the Diagnosis Instrunent recieves the ISO Code through the K Line, while the system is in its "Key On" State. The Insturment should send an initialisation code of 0F-AA-CC Hex (To allow the ECU to understand what it is communicating with). It should be in the following format:

- 1200 Baud
- 1 Start bit, 1 Stop bit,
- 8 data bits
- No Parity

The initialisation bytes should be transmitted 110ms ±10ms apart. The ECU will not acknowledge recipt. Once the ECU recieves the Initialisation code, it prepares for communication as decribed in later chapters.

1.2.3

ISO Code

The ECU transmits the ISO code when its state is changed from "Power Off" to "Power On and Power line OK"

The ISO code is sent from the ECU on line K at 1200 baud. It is a 6 byte code, the 55 Hex code meaning the system is in Synchronous operation, a 4 byte keyword and a control checksum (the Sum of the keyword and the 55 Hex byte, excluding the parity bit)

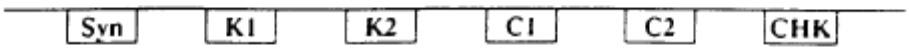
The ISO code can be requested by the diagnostic instrument at any time if communications is already estabilshed

N.B: If the Power-Latch causes communications to fail, reset the ECU using Key Off-Key On and await the ISO Code.

1.2.4

ISO Code Block Diagram

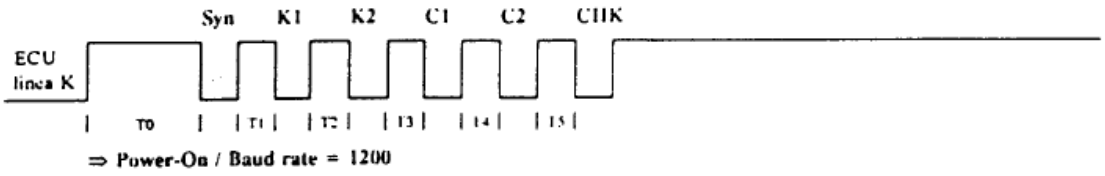
For a model specific listing of ISO Codes, see §A.



1.2.5

Technical Diagram of ISO Code

The ECU sends the ISO code down Line K at 1200 baud whenever it detects its state being changed from "Power Off" to "Power On and Power Latch OK"

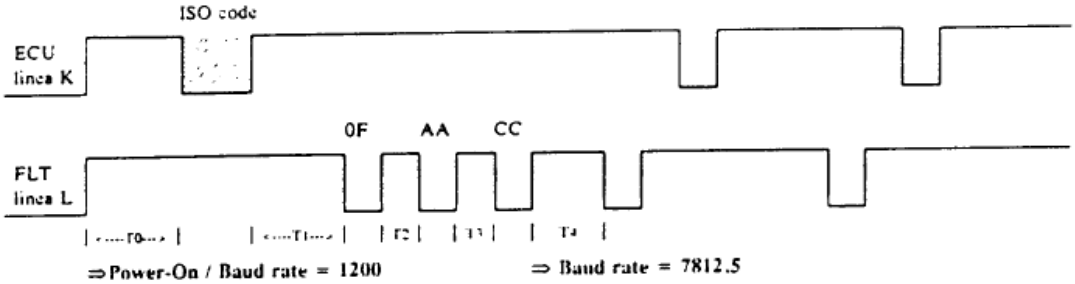


T0 = Time between Start-Up and Syn	150 ms < T0 < 1500 ms
T1 = Time between Syn and K1	2 ms < T1 < 1200 ms
T2 = Time between K1 and K2	2 ms < T2 < 1200 ms
T3 = Time between K2 and C1	2 ms < T3 < 1200 ms
T4 = Time between C1 and C2	2 ms < T4 < 1200 ms
T5 = Time between C2 and CHK	2 ms < T5 < 1200 ms

1.2.6

Technical Diagram of Initialisation

To initialise the ECU with test equipment, it is necessary to wait for time T0 ≥ 500msec after the ISO code is received before sending 0F-AA-CC Hex at 1200 baud down the L Line. Once this is received, the ECU enters a communicatiosn mode and begins communicating at 7812.5 Baud.



T0 = time between Power-On and preamble.	150 ms < T0 < 1500 ms
T1 = time between ISO Code and 0F Hex.	T1 > 500 ms
T2 = time between 0F and AA Hex.	T2 = 110 ms ± 10 ms
T3 = time between AA and CC Hex.	T3 = 110 ms ± 10 ms
T4 = time between CC Hex and first byte at 7812.5 baud	T4 > 110 ms

1.3

Communication

1.3.1

Communication Parameters

The communication to the ECU is in positive logic. It is done to the NRZ standard (Non Return to Zero) at 7812.5 baud, with a tolerance of ±5% per byte and in the following structure:

- 1 Start bit, logic state = 0
- 8 data bits
- 1 Stop bit, logic state = 1.

1.3.2

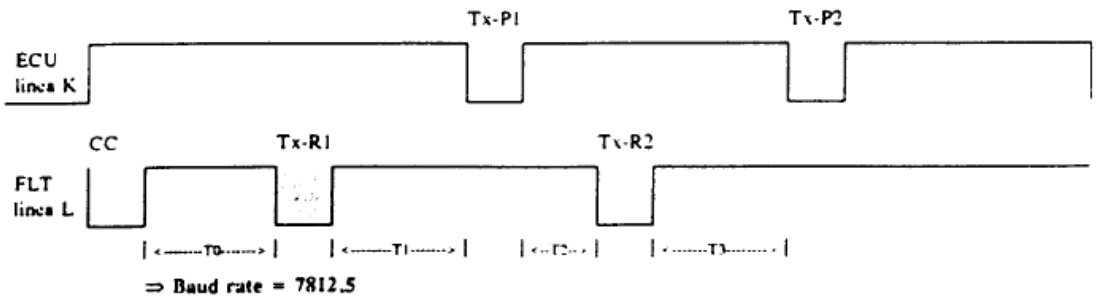
Communication Structure

Once initialised, the communication is of Masstter- Slave type, with data being sent to the ECU on Line L and Data being received at the Tester on line K. The Tester acts as the Master node.

ECU	**Tester**
Send ISO Code at "Key On" (1200 baud)	
	Send 0F-AA-CC Hex (Comms @ 7812.5 baud) Request Byte X
Reply by sending byte X	
	Etc etc

1.3.3

Technical Diagram of Communication



T0 = time between init finish and first request	T0 > 110 ms
T1 = time between request and answer	4 ms < T1 < 300 ms
T2 = time between last answer and new request	T2 > 4 ms
T3 = time between request and answer	4 ms < T3 < 300 ms

1.4

Data Available

The Tester can request certain data from the ECU by transmitting specific bytes (01-7F Hex). The data can be ECU status, Error codes, engine parameters. The techniques are described below.

If the tester requests a code that is invalid, the ECU will not respond. If the code is not available for viewing (eg: internal use only), the ECU will send back a possibly random response that will be of no use to the tester.

The first data demand should be sent at least 110msec after the last initialisation byte is sent (CC Hex.). For every correctly received code, the ECU will send out the relevant data.

If the data is requested as a word (2 Bytes – See Engine Speed or Injector Duration), the test instrument will have to send two separate requests. If is good practice to request MSB's first and LSB's last.

The ECU response time does not have to be advanced to 300msec (expect an instant response), but there should be at least a 4msec gap between data receipt and transmission of another data request.

In the "Power Latch" phase, the maximum possible duration should be used as the Microprocesor needs the time available to write EEPROM contents to RAM and reset the idle control valve.

1.4.1

Parameters Available

Code Requested by Tester (Hex.)	Parameter	Conversion Formula	Variable
01	Engine Speed MSB	Rpm = (15x 10^6) / $01 $02 [RPM]	PERIODE
02	Engine Speed LSB		PERIODE + 1
03	Injector Duration MSB	Tinij = (2 x $03$04) /10^3 [ms]	T.INJ.AP
04	Injector Duration LSB		T.INJ.AP + 1
05	Ignition Advance	Adv. = $05/2 [°Ang]	AVANCE
06	Manifold Absolute Pressure	P = $06x3 [mmHg]	MP2.MP8
07	Air Temperature	T.Air = $07 - 40 [°C]	MT.AIR.L
08	Water Temperature	T.H2O = $08-40 [°C]	MTEAU.L
09	Throttle Angle	A = $09x0.4234-2.9638 [V]	ANG.PAP0
0A	Battery Voltage	Vbatt. = $0A x 0.0625 [Volt]	M VBATT
0B	Lambda Probe Correction	KO2 = ($0B x 0.002656 + 0.66 [%]	K.O2
OC	Idle Stepper Motor Position	Pos. = $0C [Steps]	ALFAR
OD	Idle Stepper Integral Gain	Prop. = $0D / 2 (2's Compliment) [Steps]	INTEGR
0E	Idle Stepper Proportional Gain	Int. = $0E / 2 (2's Compliment) [Steps]	PROP
0F	Trimmer Position	00Hex = -127; 80Hex = 00; FFHex = + 127	TRIMRAM
10	Non – Validated Input Errors	ER-CO-INP = Coded Byte	ER-CO-INP
11	Non – Validated Output Errors	ER-CO-OUT = Coded Byte	ER-CO-OUT
12	Non – Validated Functionality Errors	ER-CO-FUN = Coded Byte	ER-CO-FUN
13	FGSTAT	FGSTAT = Coded Byte	FGSTAT
14	Validated Input Errors	ER-VA-INP = Coded Byte	ER-VA-INP
15	Validated Output Errors	ER-VA-OUT = Coded Byte	ER-VA-OUT
16	Non – Validated Functionality Errors	ER-VA-FUN = Coded Byte	ER-VA-FUN
17	Repair Code (xxH)	ASCII Character	CODRIC
18	Repair Code + 1 (xxH)	ASCII Character	CODRIC + 1
19	Repair Code + 2 (xxH)	ASCII Character	CODRIC+ 2
1A	Repair Code + 3 (xxH)	ASCII Character	CODRIC+ 3
1B	Repair Code + 4 (xxH)	ASCII Character	CODRIC+ 4
1C	Repair Code + 5 (xxH)	ASCII Character	CODRIC+ 5
1D	Repair Code + 6 (xxH)	ASCII Character	CODRIC+ 6
1E	Repair Code + 7 (xxH)	ASCII Character	CODRIC+ 7
1F	Repair Code + 8 (xxH)	ASCII Character	CODRIC+ 8
20	Repair Code + 9 (xxH)	ASCII Character	CODRIC+ 9
21	Repair Code + 10 (xxH)	ASCII Character	CODRIC + 10
22	AutoCalibration Correction MSB	ALFAU = $22$23	ALFAU
23	AutoCalibration Correction LSB		ALFAU +1
24	AutoCalibration Correction MSB with A/C	ALFAUC = $24$25	ALFAUC
25	AutoCalibration Correction LSB with A/C		ALFAUC + 1

Parameters Available (Continued)

Code Requested by Tester (Hex.)	Parameter	Conversion Formula	Variable
26	Minimum Engine Speed (Target)	CONS.REG = $26 x [Rpm]	CONS.REG
27	Minimum Offset turns	OFNNTR = $27 x 8 [Rpm]	OFNNTR
28	Stepper Correction (Delta)	DELREG = $28 - 128	DELREG
29	Stepper Correction from FLT	ALFAFTR = $29 - 128	ALFAFTR
2A	FGSTAT2	FGSTAT = Coded Byte	FGSTAT2
2B	Sign of Non - Validated Input Errors	SE-CO-INP = Coded Byte	SE-CO-INP
2C	Sign of Non - Validated Output Errors	SE-CO-OUT = Coded Byte	SE-CO-OUT
2D	Sign of Non - Validated Functionality Errors	SE-CO-FUN = Coded Byte	SE-CO-FUN
2E	Sign of Validated Input Errors	SE-VA-INP = Coded Byte	SE-VA-INP
2F	Sign of Validated Output Errors	SE-VA-OUT = Coded Byte	SE-VA-OUT
30	Sign of Non - Validated Functionality Errors	SE-VA-FUN = Coded Byte	SE-VA-FUN
31	FREE		
32	AC Offset Parameter MSB	ADOFFSET = $32$33 (Minimum with Canister deactivated)	ADOFFSET
33	AC Offset Parameter LSB		
34	AC Offset Parameter MSB	ADOFFPURG = $34$35 (Minimum with Canister activated)	ADOFFPURG
35	AC Offset Parameter LSB		
36	AC Offset Parameter MSB	ADOFFPL = $36$37 (Minimum)	ADOFFPL
37	AC Offset Parameter LSB		
38	AC Gain Parameter MSB	ADGAIN = $38$39 (Midrange value)	ADGAIN
39	AC Gain Parameter LSB		
70	Immobilizer ID Code	Not Viewable	VASTYPE
71	Immobilizer State - UNIVAS	UNIVAS = Coded Byte (Error Present and Status)	UNIVAS
72	Immobilizer Errors - EEVAS	EEVAS = Coded Byte (Only Errors)	EEVAS
73	Immobilizer Error Meter - CRDVAS	Decremented by 1 every Key - On unless errors are present otherwise is = to max value (FF Hex)	CRDVAS

1.4.2

Memory Structure and Error Memorisation

Considering self diagnosis and recovery systems, there are 21 possible errors of the system, from the following components:

- Throttle Position
- Manifold Absolute Pressure Sensor
- Lambda Sensor
- Water Temperature Sensor
- Air Temperature Sensor
- Battery Voltage
- Injector
- Ignition Coil Firing cyl. 1+4
- Ignition Coil Firing cyl. 2+3
- Idle Stepper Motor
- Charcoal Canister
- Air Conditioner
- Fuel Pump Relay
- Generic System Relays
- Auto Calibration Parameters
- RAM
- EPROM
- EEPROM
- Microprocessor
- TDC / RPM Sensor
- Immobilizer

The errors are subdivided in 3 bytes in RAM (ERR-CO-XX) + 3 bytes in EEPROM (ERR-VA-XX):

- **ERR-CO-INP** - Confirmed Input Errors (memorized in RAM)
- **ERR-CO-OUT** - Confirmed Output Errors (memorized in RAM)
- **ERR-CO-FUN** - Confirmed Functionality Errors (memorized in RAM)
- **ERR-VA-INP** - Validated Input Errors (memorized in E^2PROM)
- **ERR-VA-OUT** - Validated Output Errors (memorized in E^2PROM)
- **ERR-VA-FUN** - Validated Functionality Errors (memorized in E^2PROM)

A Temporary error will only show up in one memory (ERR-CO-XX) but may have been validated previously (present in ERR-VA-XX):

- **ERR-CO-XX -** Error Filtered
- **ERR-VA-XX** – Error Validated, held in E^2PROM (ERR-CO-XX copy)

There are also 2 bytes that define the state of the Engine and the ECU:

- **FGSTAT**
- **FGSTAT2**

1.4.3

ECU Warning Lamp Management

The ECU Warning Lamp is only lit when an error that makes it light is present and validated, that meaning both ERR-CO-XX and ERR-VA-XX must have data written in them.

ERR-CO-XX	ERR-VA-XX	WL	
0	0	Off	No Errors
1	0	Off	Temporary Error
0	1	Off	Error Already Memorised
1	1	On	Error in Calibration / Error Present and Validated

1.4.4

Error Cancellation

Cancellation of the errors held in the EEPROM can happen in two ways:

1. Decrement the Error term to zero (CRDxxx = 0), if there are no errors present on the line in question after five 6 minute periods, the errors are cleared (the engine doesn't try and stop itself rotating)
2. Through the Test Instrument using the correct commands.

1.4.5

Error and System State Memory Structure

The memory structure is as follows:

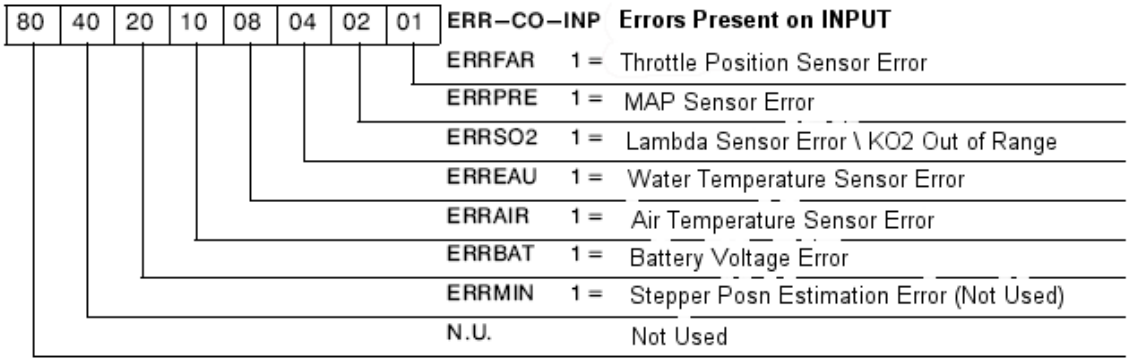
- 3 error bytes in RAM (ERR-CO-XX) + 3 bytes in E^2PROM (ERR-VA-XX)
- 3 bytes that show the sign of the error in RAM (SE-CO-XX0 + 3 bytes in E^2PROM (SE-VA-XX))
- 2 System status bytes (FGSTAT & FGSTAT2)

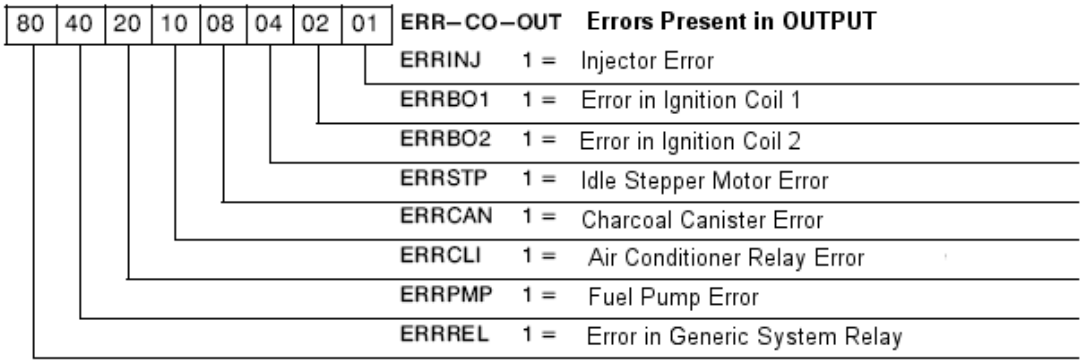
Code Requested by Tester (Hex.)	Variable	Description
10	ERR-CO-INP	Filtered Input Error, memorized in RAM (variable in RAM)
11	ERR-CO-OUT	Filtered Output Error, memorized in RAM (variable in RAM)
12	ERR-CO-FUN	Filtered Functionality Error, memorized in RAM (variable in RAM)
13	FGSTAT	System Status Byte (variable in E^2PROM)
14	ERR-VA-INP	Verified Input Error, memorized in E^2PROM (variable in E^2PROM)
15	ERR-VA-OUT	Verified Output Error, memorized in E^2PROM (variable in E^2PROM)
16	ERR-VA-FUN	Verified Functionality Error, memorized in E^2PROM (variable in E^2PROM)
2A	FGSTAT2	System Status Byte (variable in E^2PROM)
2B	SE-CO-INP	Sign of Confirmed Input Errors (variable in RAM)
2C	SE-CO-OUT	Sign of Confirmed Output Errors (variable in RAM)
2D	SE-CO-FUN	Sign of Confirmed Functionality Errors (variable in RAM)
2E	SE-VA-INP	Sign of Confirmed Input Errors (variable in E^2PROM)
2F	SE-VA-OUT	Sign of Confirmed Output Errors (variable in E^2PROM)
30	SE-VA-FUN	Sign of Confirmed Functionality Errors (variable in E^2PROM)

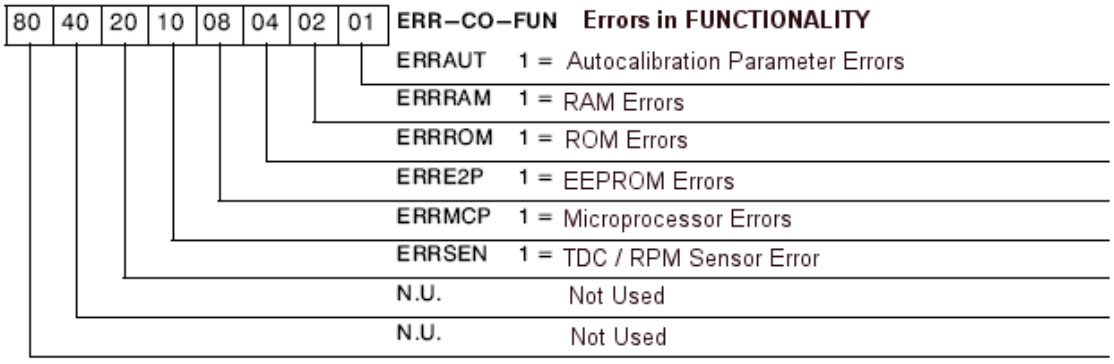
1.4.6

ERR-CO-XX Error Memory Byte Coding

Error Memory is held in RAM and is reset-able by removing battery power.



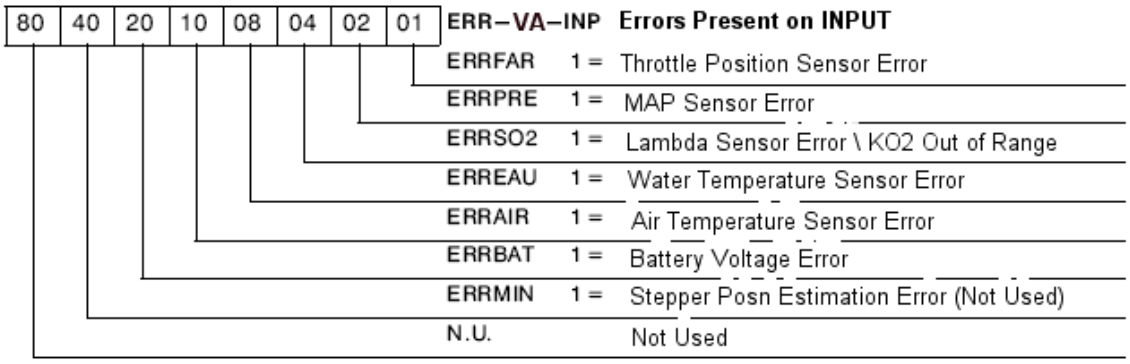


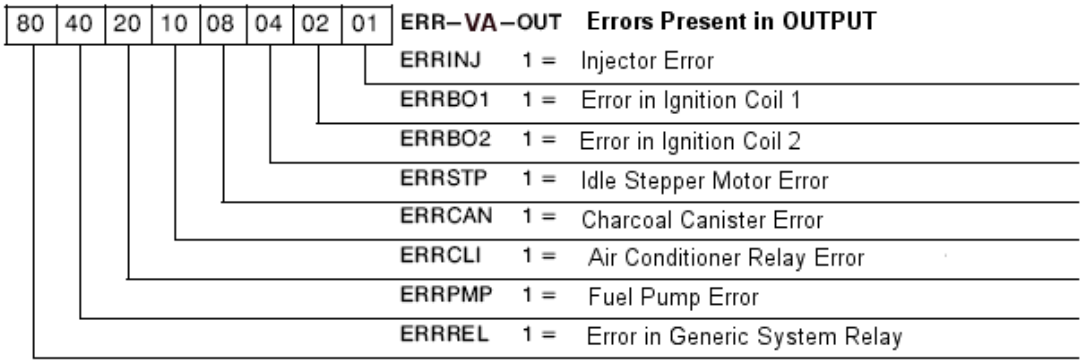


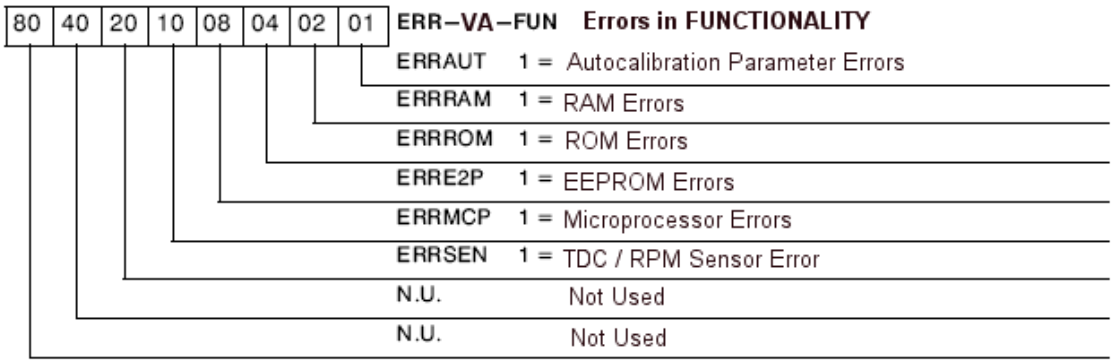
1.4.7

ERR-VA-XX Error Memory Byte Coding

Error Memory is held in EEPROM and is reset-able by using the test instrument or incrementing the appropriate memory.



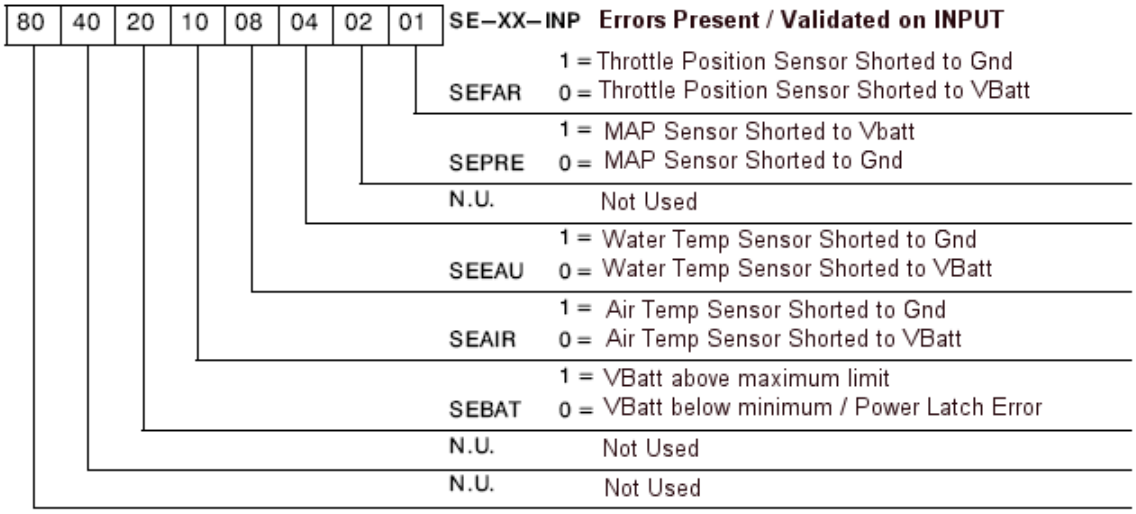


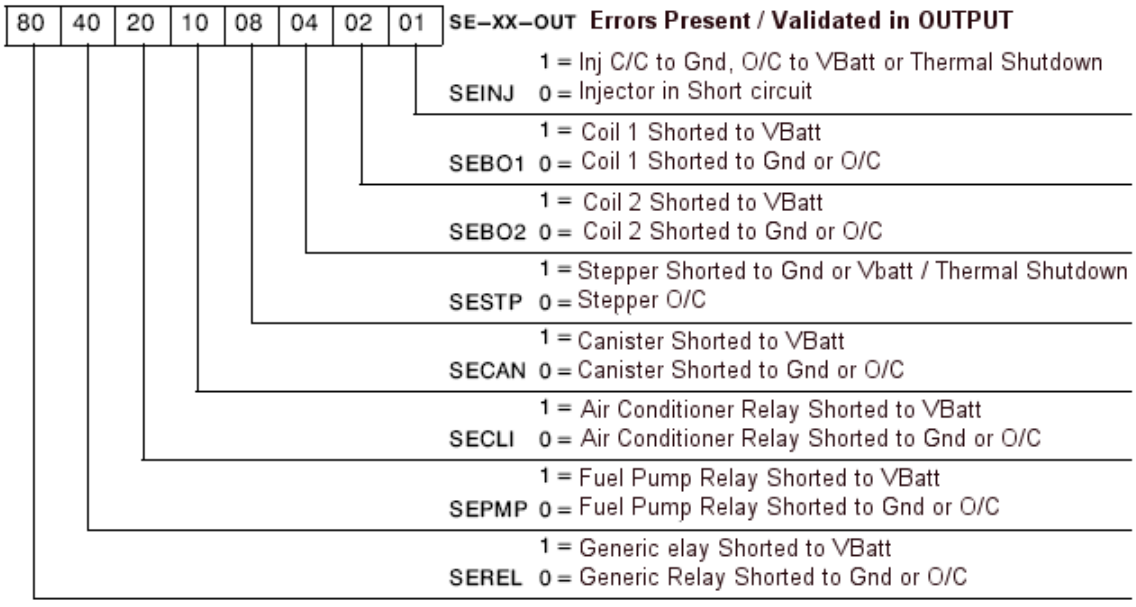


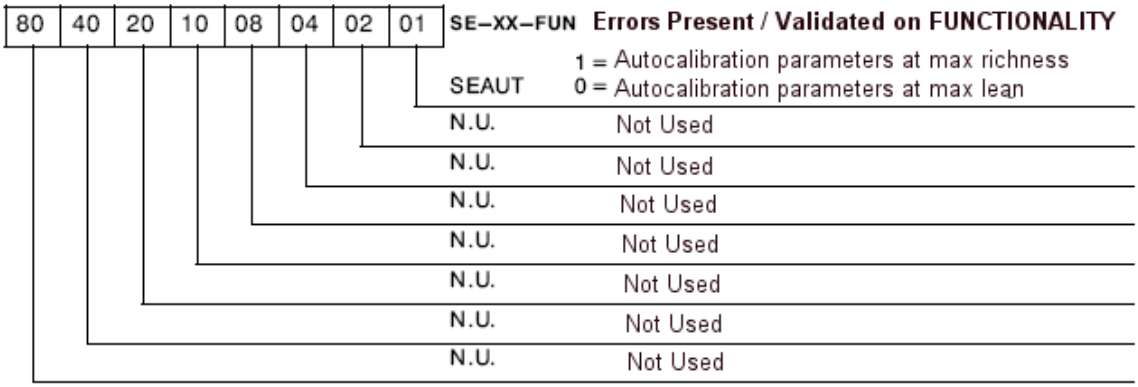
1.4.8

SE-CO-XX / SE-VA-XX Error Memory Byte Coding

SE-CO-XX Held and Variable in RAM
SE-VA-XX Held and Variable in EEPROM

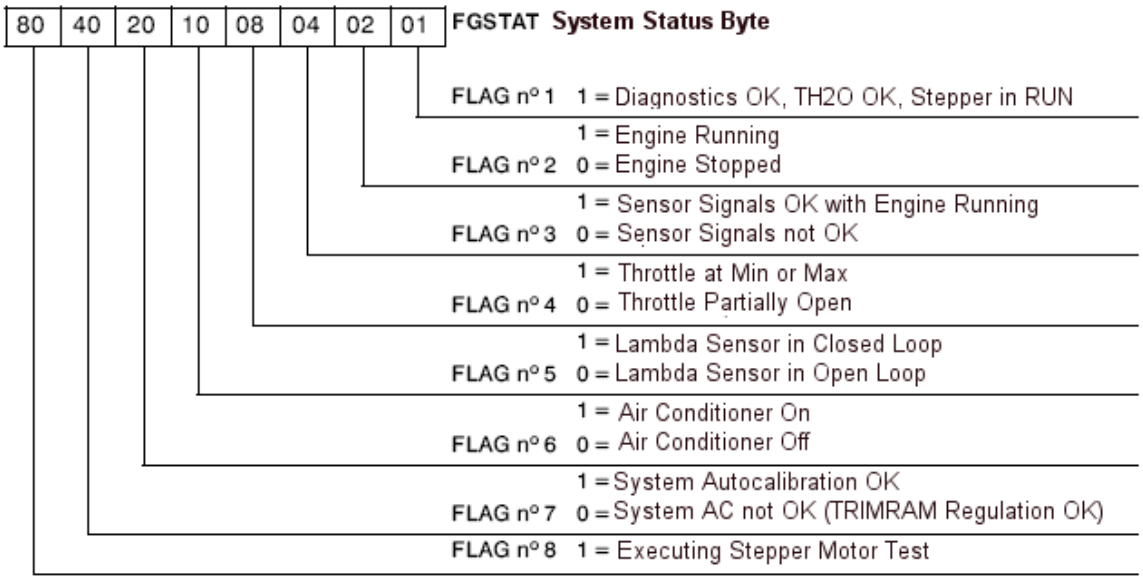


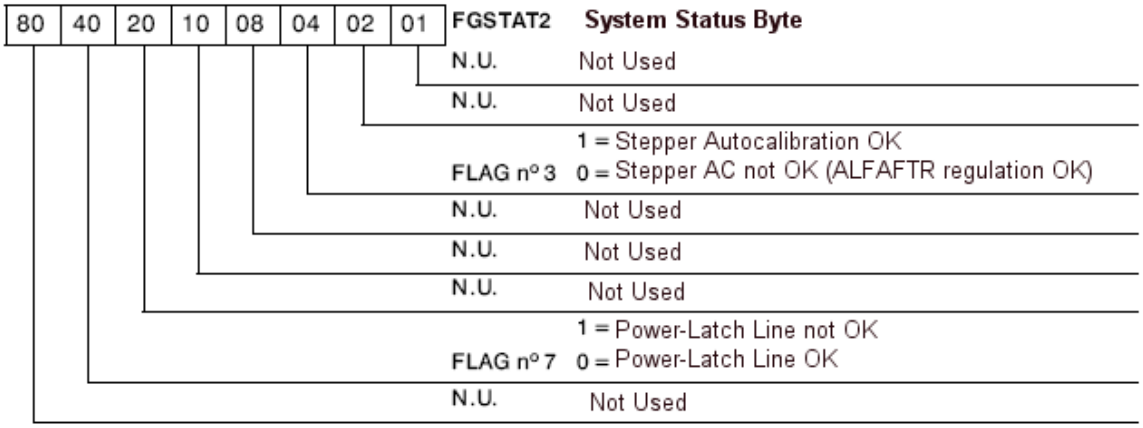




1.4.9

FGSTAT and FGSTAT2 Error Memory Byte Coding





1.5

“Electronic Key” (Immobilizer)

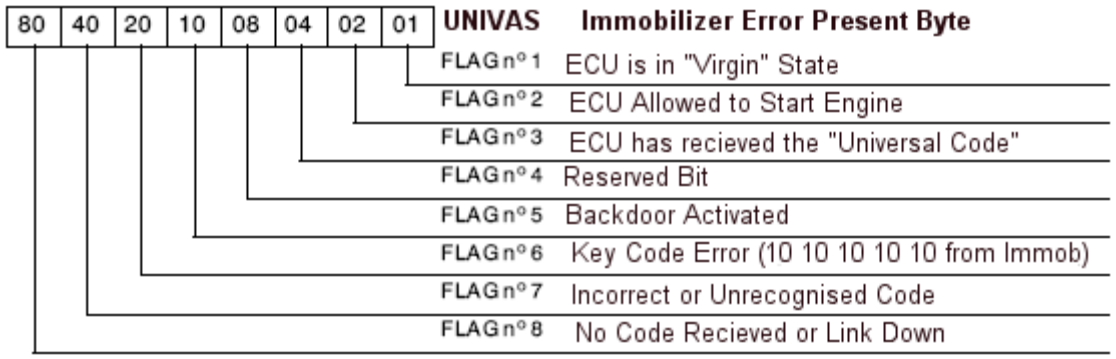
1.5.1

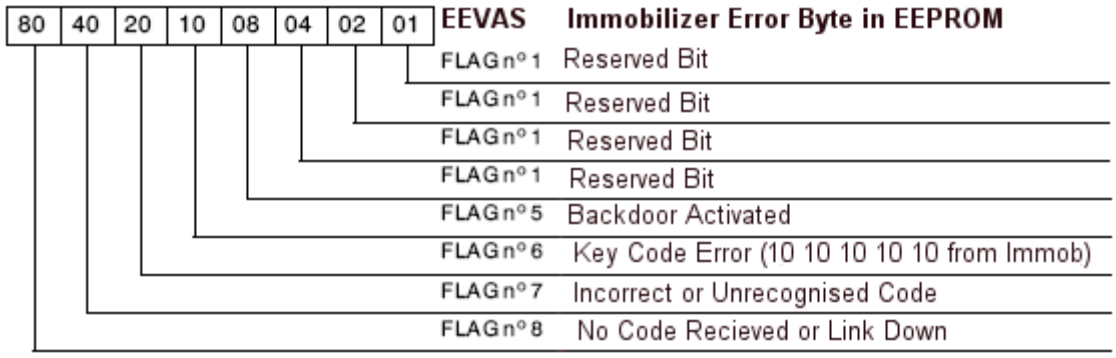
Error Memory Byte Coding

The Immobilizer has two error bytes in the ECU, one in RAM (UNIVAS) and one in EEPROM (EEVAS). CRDVAS resides in EEPROM and indicates the number of EEVAS errors that have been logged.

UNIVAS indicates the type of Immobilizer error present (Errors present when RAM reset)) and also indicates the State of the system, and there are 3 important conditions:

Flag n° 1 indicates if the ECU is “Virgin” – that meaning it is un-coded, with no keys programmed into it.
Flag n°2 indicates if the Immobilizer doesn’t allow the ECU to start the engine.
Flag n° 3 indicates if the ECU (whilst in Virgin State) has received the “Universal Code” from the Immobilizer ECU





N.B.: ECU - Virgin Code

In the event of the system not recognizing the code sent to it, if the ECU is in “Virgin State” and there are no Immobilizer errors, 05 Hex is written to **UNIVAS**. (at every Key-On the ECU Warning Lamp will stay on for 4 sec, and the CODE light flashes with fixed frequency of 1 Hz: T_{OFF} = 500ms and T_{ON} = 500 ms). If the Immobilizer ECU is not present, the Main ECU will remain in its “Virgin State” and **UNIVAS** = 01 Hex.

N.B.: Immobilizer ECU Code Memorisation

EEVAS will only have a value if any of 3 specific errors are present if the Immobilizer ECU is programmed for a certain key (e.g.: Not Virgin). The EEVAS byte will show all present errors, but the FF Hex value in CRDVAS will be due to the most recent confirmed error.

1.5.2

Immobilizer Error Counter CRDVAS

CRDVAS resides in EEPROM memory in the ECU and indicates the number of Key-On's that have passed since the last Immobilizer Error was logged. When errors are present, the CRDVAS value is equal to FF Hex (256 dec). It is decremented by 1 Hex. (1 dec) every "Key-On" (providing no errors are present) 00 Hex is reached, when EEVAS is reset to 00 Hex.

N.B. : The CRDVAS memory is only decremented when one transition from Key-Off to that one of Key-On (with valid code on the serial line), in absence of errors is detected by the Immobilizer ECU.

1.5.3

System Recovery using Test Instrument or Accelerator Pedal

If the Immobilizer Error: No Code Sent / Link Down is present, the Immobilizer ECU will not allow the engine to start.

A procedure exists that allows you to start the engine by entering the Immobilizer's unique 5 digit code (0…9) through the accelerator pedal.

The "Disallowed Start" Error is not always verified at "Key-On" and may not cause the ECU Warning Light to ignite with the Immobilizer ECU (CODE) Warning Light.

The Recovery procedure is as follows:

1) Press the accelerator to its greatest travel until the ECU Warning light remains on.
2) When the light goes out, release the accelerator pedal. The ECU Warning light will begin to flash On-Off, the first number on the CODE Card can be entered.
3) To Enter the number on the CODE Card, it is necessary to count the On-Off pulses of the ECU Warning Light and to interrupt the (after the light has flashed on the same amount of times as the number on the CODE Card) by pressing on the accelerator pedal and holding it until the ECU Warning Light remains on
4) Repeat the Process from 1) to enter the remaining numbers on the CODE Card.
5) If the Procedure is successful, the engine will start, otherwise the procedure has failed and will need repeating.

The other way to execute the system recovery is to use a test instrument, this is much more straightforward as the CODE Card number is entered through the instrument's keyboard.

In this case the Test Equipment will ask for the CODE Number and it can be entered in through the keypad, and the Immobilizer can be bypassed automatically by the Test Instrument.

Using the Tester, it is not possible to fail the system recovery unless the wrong code is entered (e.g.: not the same number as is on the CODE Card).

1.5.4

Procedure of Immobilizer Key Memorization

Prior to attempting to memorize keys into the ECU, it must be ascertained that the ECU is definitely virgin.

The system can be checked to see if it is virgin by whether the CODE Warning light remains on or off whilst the following is being done:

1) Verifying all keys (all blue keys and the red master key) do the following when inserted into the ignition and turned to "AVV": The CODE Warning light lights up for 0.5s, then remains off for 1.5s After this, the CODE warning light flashes on and off with a 50% Duty Cycle for as long as the ignition is on.

If the system is not virgin, or not working, the above procedure will have a different result. It is advisable to check the antenna that reads the code from they key is working correctly and that the Immobilizer ECU is communicating with the Engine ECU correctly.

2) The memorization procedure is started by inserting the master key (usually red) into the ignition and turning to "AVV". Return the key to the "Stop" Position once the CODE Warning light extinguishes.

3) Repeat 2) for any (blue) slave keys.

4) Finish the procedure by repeating 1) – inserting the master and turning to "AVV".

5) To check the procedure was successful, and that the system functions: reinsert the slave keys in turn (not the red master key) and check that the CODE warning light extinguishes after around 0.5secs of "Key-On"

1.6

Description of Active Diagnostics

Active Diagnostics tests allow a human tester to control individual actuators normally controlled by the ECU, in order to eliminate them in cases of multiple sensor / actuator failure or obscure ECU problems. It also allows a tester to clear the validated errors in EEPROM, and check some EEPROM values (System AutoCalibration, Stepper AutoCalibration and Minimum Stepper Regulator Turns)

Active Diagnostics tests are only possible with the key in the "On" position and the engine off.

To be able to do Active Diagnostics from the FLT or other testing device, you must send code $AA to the ECU once communications is established. It will send the echo if conditions are right for Active Diagnostics (It will not echo the correct code if the engine is on).

The code sent to the ECU will define what actuator / component is tested. All tests are listed in a table in this text. Once a code is sent to the ECU for a specific test, it will first echo the same code to the tester, then it will send a code that describes the test being performed, which starts a procedure of guided diagnosis – if the engine is running, the test will test the component silently (without affecting the engine), if the engine is off, the ECU will test the component directly.

The test can be interrupted from the tester, by sending code $FF Hex to the ECU. It responds by halting the test and sending an echo of the previous data.

During tests of output devices, they are tested to the voltage levels of +VBatt. If errors are detected, the test is halted, any commands to the devices are halted and code $EE is sent to the test instrument to signal an anomaly.

The ECU will send a code out after each test, there are two possible codes:

Code EE: Test FAIL (Error during execution)
Code FF: Test OK

To get out of Active Diagnostics, the tester must send a Stop code ($FF Hex) to the ECU whilst no tests are being performed.

If the K or L line is disconnected, the test instrument will have to TIMEOUT the communication as it is unable to receive the echo of the last sent code or the outcome of any tests underway. The TIMEOUT is variable on the test performed, e.g.: on the Fuel Pump test the TIMEOUT is 30s.

If serial communications is interrupted, a cycle of Key-On / Key-Off is needed to allow the car to be started again.

1.6.1

Test Functions Available in Active Diagnostics

Certain running modes must be adhered to for Active Diagnostics to be successful: The key must be in "On" position and the engine stopped (if the Test expects it) or the engine should be Running (Stepper Test, Engine Speed Regulation, Stepper Regulation (Steps), System Regulation); In the table all the codes of all possible tests are given (although some are only available in certain running states):

Code Requested by Tester (Hex.)	Parameter	Conversion Formula	Running Mode
80	Fuel Pump Relay Operation	Operate Relay for 30 secs	Key On
81	Injector Operation	Activate for 4 msec every sec, 5 times	Key On
82	Ignition Coil 1 Operation	Activate for 2 msec every sec, 5 times	Key On
83	Ignition Coil 2 Operation	Activate for 2 msec every sec, 5 times	Key On
84	Error Cancellation	Cancel all errors in EEPROM next Key-On	Key On
85	Charcoal Canister Operation	Activate for 20 msec every sec, 7 times	Key On
86	Engine Rev Limiter Operation	Activate at 125 Hz (3750rpm) for 2 sec	Key On
87	Air Conditioner Relay Operation	Operate Relay for 30 secs	Key On
88	Generic Relay Operation		Key On
*89	Toggle System AC / Trimmer CO (Limp Home)	Enable / Disable System AutoCalibration	Key On
8E	Stepper Operation	Activate 3 x +/- 32 steps (at 8 steps / sec)	Engine Running
**8F	TRIMRAM Regulation (inhibited from 2A21)	System Trimmer Regulation	Engine Running
*91	Toggle Stepper AC (Steps) / Manual Regulation	Enable / Disable Trimmer AutoCalibration	Key On
**92	Stepper Regulation (Steps) (inhibited from 2A21)	Stepper Regulation (Steps) ALFAFTR	Engine Running
98	Unblock Immobilizer	$98 + 5 bytes, terminating in $FF	Key On
AA	Enter into Active Diagnostics mode		Key On
EE	Error During Last Test		Key On
FF	Last test OK / Test finished		Key On

* - This only serves to test if the system is correctly autocalibrating. If the system did not have autocalibration programmed in from the factory, this is not available

** - Regulation parameters ALFAFTR and TRIMRAM are inhibited from SW v.2A21

1.6.2

Active Diagnostics Mode

This describes how to enter into Active Diagnostics, test parts, close down Active Diagnostics and what to expect from the ECU and test instrument.

Entering into Active Diagnostics:

FLT: Send code $AA Hex to the ECU with the Key on and the engine stopped
ECU: Echo back the received code ($AA Hex)

Running Tests:

FLT: Send code of appropriate Test to the ECU.
ECU: Echo back received code.

Engine State during Test activation:

The ECU does not allow the engine to run except for the test procedures: Stepper Test ($8C), System Regulation ($8F, $90), Engine Speed Regulation ($96, $97), Stepper Regulation (Steps) ($93, $94)

During Test Execution:

FLT: Await test Outcome ($EE or $FF)
ECU: Test part and send $EE if error execution fails

Forced Interruption during Testing:

FLT: Send $FF to ECU during test.
ECU: Send $FF back and interrupt test if part not in error, otherwise send $EE.

Test Outcome after Execution:

FLT: If $FF received, mark test OK, if $EE received, mark test FAIL
ECU: Send $FF if test had positive result, $EE if test had errors.

Exiting Active Diagnostics:

FLT: Send $FF when not executing tests
ECU: Echo back $FF.

1.6.2

Flow Diagram of Active Diagnostics

ECU	FLT
	← Tx $AA (Enter AD)
Tx $AA (Acknowledge = echo the code) →	
	← Tx the code to Activate the desired test
Tx echo of the desired code →	
	←Await Test Outcome
If the received code is recognized as OK, the test is allowed to start, otherwise no action is taken →	
Send $FF if Test OK, $EE if Test Error →	← Signal Test OK if $FF received ← Signal Test NOK if $EE received
	← Tx other codes / $FF (Exit AD)
Echo code received or sent $FF if Exit received →	
	If test code echo Rx'd await outcome.

N.B.: Transmission of $FF from FLT, it will have two different functions dependant on when it is transmitted:

1. If $FF is Tx'd from the FLT during test execution (after the test code is echoes back to the FLT), the ECU views this as a test interrupt and echoes the code received ($FF) back to the FLT. **The ECU is still in AD mode** so further test codes may be sent to it.

2. If $FF is sent in the absence of any test being executed and after the outcome of the last test was received by the FLT, **the ECU views this as a request to Exit AD**, so no further test codes may be sent.

1.6.3

Stepper AutoCalibration / Manual Stepper Regulation of Steps

The idle regulation is managed by the Idle Control Stepper Motor, and this is driven by the ECU. The number of steps needed on this stepper motor needed to keep the engine at the correct idle speed can be controlled by the ECU software using Auto Calibration strategies that take into account the air conditioner being on (if present), the air temperature and pressure and the status of the engine.

The Auto calibration is performed at each Power-On of the ECU. If the EEPROM is blank, a factory standard value can be copied across from another memory until the AC function can test for the correct value. Subsequently, the Auto-calibration function may be activated or disabled if it exists. That can be tested for in AD mode (Toggle AC Stepper / Manual Regulation). If the AC Function is present, there are two possible statuses:

Function present and enabled (FGSTAT2 Bit n° 3 = 04 Hex):
In this case it is possible to disable the AC function (Using the Toggle command), and manipulate the regulation values using the FLT, then save the new value in memory

Function Present and Disabled (FGSTAT2 Bit n° 3 = 00 Hex.):
In this case the regulation values may be directly manipulated or the AC function amy be enabled.

Notes:

1) If the test shows that the AC function is on, it can be disabled with the FLT (but the reverse is not possible)
2) Disabling AC involves setting ALFAU, ALFAUC and ALFAFTR to zero. The same thing happens as part of the Enabling process.
3) Configuration (disabling / enabling) of AC is written to EEPROM, so it is also maintained if the battery is disconnected.
4) New (from factory) ECUs write initial conditions to ROM at first Power-Latch and initalise ALFAUC / ALFAU to zero

N.B.: Regulation of ALFAFTR (Code $92 Hex in AD) is inhibited from SW vers.2A21

1.6.5

System AutoCalibration / Software Trimmer Regulation

This function influences T_{INJ}, the injector duration time. This is worked out using an ECU algorithm that also takes into account the decay rate of engine speed and it aims to harmonize the two parameters. The Auto-Calibration function is carried out at the first Power-On of the ECU (whilst the EEPROM is blank), after which a value is written into memory and the AutoCalibration function exists and is enabled. Subsequently, the AutoCalibration may be enabled or disabled if it exists in EEPROM memory using the AD command "Toggle AC / TRIMRAM Regulation". If the AC function exists, there are two possible states:

Function present and Enabled (FGSTAT Bit n^o 7 = 40 Hex).
The function may be disabled using the test instrument and TRIMRAM can be directly manipulated with the correct commands.

Function present and disabled (FGSTAT Bit n^o 7 = 00 Hex).
The AC Function may be re-enabled or TRIMRAM regulation may be carried out directly using the test instrument.

Notes:

1) If the test shows AC as being enabled, it is possible to disable it with the FLT, but if it doesn't exist (AC not programmed from the factory), it cannot be disabled.

2) Disabling AC involves setting OFFSET, GAIN and TRIMRAM to zero. The same thing happens when AC is re-enabled.

3) The status (enabled / disabled) of AC is stored in EEPROM, so it is maintained if the battery is disconnected

4) ECUs new from the factory write initial conditions into ROM at first Power-Latch and initialize TRIMRAM to zero.

N.B.: Regulation of TRIMRAM (code $8F in AD) is inhibited from SW vers. 2A21.

1.6.6

Testing System Regulation

Tests $89 and $91 in AD act to test the automatic calibration capability of the ECU. This is only possible if AC was programmed into the ECU from the factory. If AC is not present, the test will not be possible. With the system Auto Calibration disabled, T_{INJ} (injector duration) is a function of the TRIMRAM parameter, and that offset has an MSB and an LSB.
If the Offset is positive, it has a constant influence on $T_{INJ,}$ but if the Offset is negative, it will only affect T_{INJ} at the idle throttle position. With Auto Calibration disabled, the idle stepper motor opening is controlled by the ALFAFT parameter, the maximum idle stepper motor opening is ALFAM, and the idle position is ALFAMIN.

Altering TRIMRAM affects engine speed and Lambda values and the current value of TRIMRAM. Altering ALFAFT affects engine speed, ALFAFT, DELREG and the current value of ALFAFT. Altering OFNNT affects engine speed, CONS.REG and the current value of OFNNT only.

N.B.: Run the tests with the engine running and the coolant temperature above 70^{o}C.

1.6.7

Flow Diagram for Testing System Regulation

The initial configuration of System / Stepper Auto Calibration is as follows:

Stepper AutoCalibration Enabled → FGSTAT2 = **04 Hex**
System AutoCalibration Enabled → FGSTAT = **40 Hex**

Engine State	ECU	FLT
		← **Tx** code $AA (Enter AD)
	Tx code $AA (Acknowledge the code) →	
Key On		← **Tx code $89/ $91 Toggle**
	Tx Echo if Ok, $EE if NOk →	
		If Test OK Start Engine
		← **Tx code $92/ $8E (Man. Reg'n)**
	Tx Echo if Ok, $EE if NOk →	
		Echo Man Reg'n if Recieved *Signal NOk if $EE Receieved*
		← **Tx $ (Regulation Value)**
	Tx Echo of DATA if Ok, $EE if NOk →	
Engine Running		*The ECU will instantly memorize and apply the value sent and echo it to the FLT*
		← **Tx $FF to finish Reg'n test (value OK)**
	Tx Echo if Ok, $EE if NOk →	
		← **Tx $FF to Exit AD**
	Tx Echo if Ok, $EE if NOk →	
Key On / Off		*To finish the calibration, turn the Key to the "STOP" position, then turn the key to the "ON" position and re-send $0F $AA $CC to re-establish serial communications*

1.7

Wiring Connections / Pin-out: 16F ECU

Number	Description	Component / Function	Notes
1	Output	Ignition Coil 1	Cylinders 1+4
2	Output	Idle Control Stepper Motor	Phase B
3	Output	Idle Control Stepper Motor	Phase D
4	Output	Central Power Relay (Power-Latch)	Power to RAM
5	Input	Automatic Gearbox ECU input	Only on Version CA
6	Output	ECU Warning Lamp	Activated on -Vbatt
7	Input / Output	Immobilizer (If Present, otherwise NC)	Bidirectional
8	Input	Air Conditioner Relay on (+)	Activated on +Vbatt
9	NC	Not Connected	Not Connected
10	Input	Serial Line L	Unidirectional line (FLT->ECU)
11	Input	TDC / RPM Sensor Signal	Negative
12	Input	Lambda Sensor Signal	Negative
13	Input	Water Temperature Sensor Signal	
14	Output	Sensor +5v Line	5 volt DC
15	Output	Serial Line K	Bidirectional line (FLT->ECU)
16	Power	Signal Grounds	In Wiring Loom
17	Power	Primary Ground	Engine Ground
18	Output	Fuel Injector	Controls SPi Unit
19	Output	Ignition Coil 2	Cylinders 2+3
20	Output	Idle Control Stepper Motor	Phase A
21	Output	Idle Control Stepper Motor	Phase C
22	Output	Charcoal Canister Multifunction Valve	
23	Output	Engine Speed Limiting (Fuel Pump) Relay	Fuel Pump / Injector / Canister / Coils / Lambda
24	Output	Air Conditioner Relay on (-)	Activated on Gnd
25	NC	Not Connected	Not Connected
26	Input	Ignition in "On" Position	
27	NC	Not Connected	Not Connected
28	Input	TDC / RPM Sensor Signal	Positive
29	Input	Lambda Sensor Signal	Positive
30	Input	Throttle Position Sensor Signal	
31	Input	Air Temperature Sensor Signal	
32	Input	MAP Sensor Signal	
33	NC	Not Connected	Not Connected
34	Power	Secondary Ground	Engine Ground
35	Power	Positive feed from Main Relay	Positive out from ECU Relay / Power-Latch