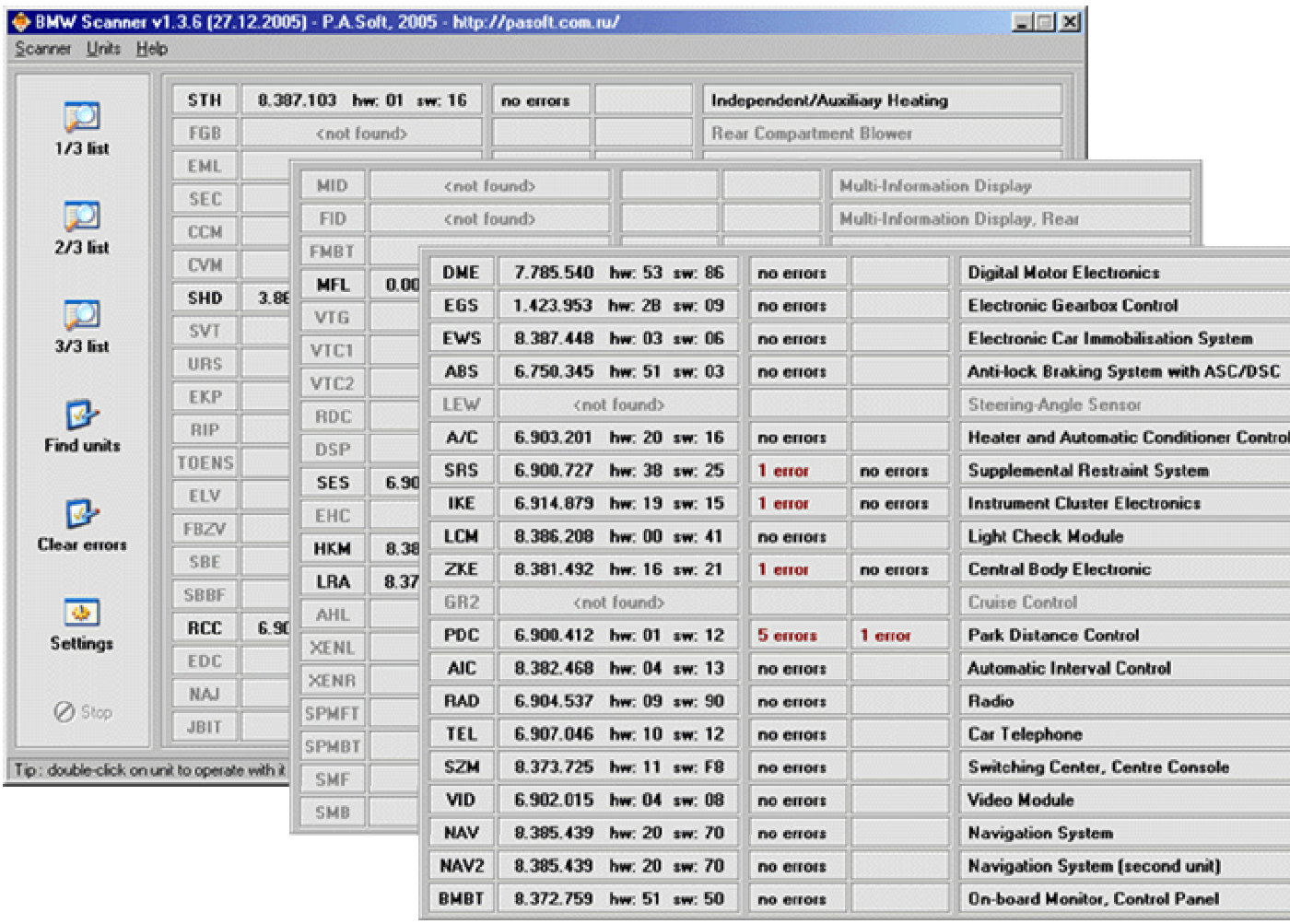
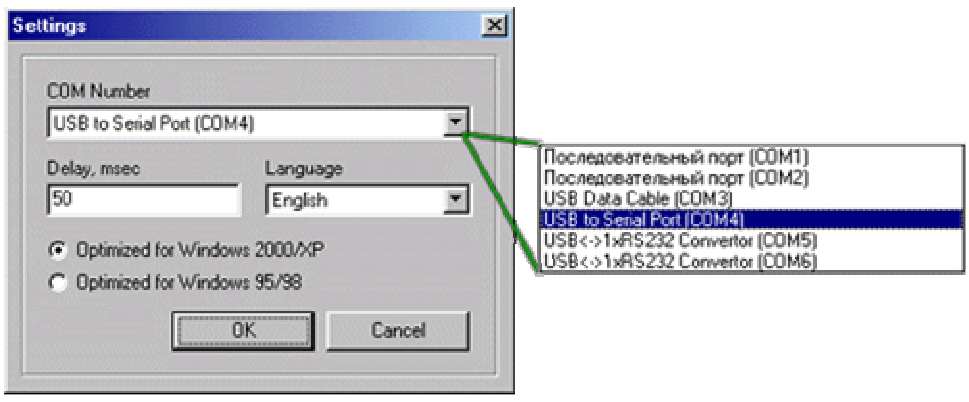
Differences from previous versions

First of all the list of supported units is much bigger for now. At this moment software supports almost all electronic units than could be installed on "old" chassis.



In main menu you can see new [CLEAR ERRORS] function. It was cutted from [FIND UNITS] procedure. The reason of doing this was a lot of customers asks because in previous versions [FIND UNITS] procedure contained also automatically malfunctions codes reset that was making malfunctions analysis more difficult.

A little changes was done also in settings menu. You can see it on next screenshot :



For standart COM port selection in [COM Number] field now you can see list of all available COM ports. This list was read from windows registry, so that's no more necessary to select working COM port by checking all of them. All you need to do is only select necessary for you COM port and use it.

Its also making virtual COM port detection more easy. Virtual COM using when adapter works with USB to COM converter, that making possible using adapter on computers that have no COM port (for example it can be modern laptops). A little changes you can see also in log file structure:

```
-------------------------------------------------------------------------------------------
BMW Scanner v1.3.6 (27.12.2005) - P.A.Soft, 2005                  Time: 14:33:58 28.12.2005
-------------------------------------------------------------------------------------------
---------------------------
 Clear errors start
---------------------------
DME*  ->  7.785.540   hw: 53  sw: 86          ODO: -           VIN: WBAGE61080DN44432   ADFG: 000000
EGS   ->  1.423.953   hw: 2B  sw: 09          ODO: 165010 km   VIN: WBAGE61080DN44432   ADFG: 000000
EWS   ->  8.387.448   hw: 03  sw: 06          ODO: 165022 km   VIN: WBAGE61080DN44432   ADFG: FFFFFF
ABS   ->  6.750.345   hw: 51  sw: 03          ODO: -           VIN: -                   ADFG: -
A/C   ->  6.903.201   hw: 20  sw: 16          ODO: -           VIN: -                   ADFG: 004432
SRS   ->  6.900.727   hw: 38  sw: 25          ODO: -           VIN: DN44432             ADFG: 000000
IKE   ->  6.914.879   hw: 19  sw: 15          ODO: 165026 km   VIN: DN44432             ADFG: FFFFFF
LCM   ->  8.386.208   hw: 00  sw: 41          ODO: 165000 km   VIN: DN44432             ADFG: -
ZKE   ->  8.381.492   hw: 16  sw: 21          ODO: -           VIN: -                   ADFG: FF4432
PDC   ->  6.900.412   hw: 01  sw: 12          ODO: -           VIN: -                   ADFG: 014432
AIC   ->  8.382.468   hw: 04  sw: 13          ODO: -           VIN: -                   ADFG: 210799
RAD   ->  6.904.537   hw: 09  sw: 90          ODO: -           VIN: -                   ADFG: -
TEL   ->  6.907.046   hw: 10  sw: 12          ODO: -           VIN: -                   ADFG: -
SZM   ->  8.373.725   hw: 11  sw: F8          ODO: -           VIN: -                   ADFG: 000000
VID   ->  6.902.015   hw: 04  sw: 08          ODO: -           VIN: -                   ADFG: FF4432
NAV   ->  8.385.439   hw: 20  sw: 70          ODO: -           VIN: -                   ADFG: -
NAV2  ->  8.385.439   hw: 20  sw: 70          ODO: -           VIN: DN44432             ADFG: 004432
BMBT  ->  8.372.759   hw: 51  sw: 50          ODO: -           VIN: -                   ADFG: FFFFFF
MFL   ->  0.000.000   hw: 01  sw: 08          ODO: -           VIN: -                   ADFG: -
SES   ->  6.903.886   hw: 05  sw: 31          ODO: -           VIN: -                   ADFG: -
HKM   ->  8.381.988   hw: 01  sw: 02          ODO: -           VIN: -                   ADFG: FFF928
LRA   ->  8.375.964   hw: C6  sw: 05          ODO: -           VIN: -                   ADFG: 05FFFF
STH   ->  8.387.103   hw: 01  sw: 16          ODO: -           VIN: -                   ADFG: 004432
SHD   ->  3.861.949   hw: 05  sw: 31          ODO: -           VIN: -                   ADFG: 264432
RCC   ->  6.903.660   hw: 20  sw: 02          ODO: -           VIN: -                   ADFG: -
```

```
ABS   ->  6.750.345   hw: 51  sw: 03          no errors
A/C   ->  6.903.201   hw: 20  sw: 16          no errors
SRS   ->  6.900.727   hw: 38  sw: 25          1 error    <- Clear ->  no errors
IKE   ->  6.914.879   hw: 19  sw: 15          1 error    <- Clear ->  no errors
LCM   ->  8.386.208   hw: 00  sw: 41          no errors
ZKE   ->  8.381.492   hw: 16  sw: 21          1 error    <- Clear ->  no errors
PDC   ->  6.900.412   hw: 01  sw: 12          5 errors   <- Clear ->  1 error
AIC   ->  8.382.468   hw: 04  sw: 13          no errors
RAD   ->  6.904.537   hw: 09  sw: 90          no errors
TEL   ->  6.907.046   hw: 10  sw: 12          no errors
SZM   ->  8.373.725   hw: 11  sw: F8          no errors
```

```
 Errors details (dec)
----------------------------
SRS   ->  26/160
          Errors after clear:
          no errors

IKE   ->  191/72
          Errors after clear:
          no errors

ZKE   ->  194/2
          Errors after clear:
          no errors
```

Except standart data about installed units mileage and VIN in log file you also can see detail information about found malfunctions codes and ADFG data. More information about it you can read in "ADFG - "present" from BMW ?"

In new version you can also read log file from main program window using [Scanner]/[View log-file] option or [F5] key.

There is also different way to save log files. In previous versions log file has fixed name (bmwscan.log) and all the time when program was run more and more times all previously saved data in file was erased. In new version log file name contains also date and time when program was started, that makes impossible to lost previous data and more easy to find necessary file in arhives.

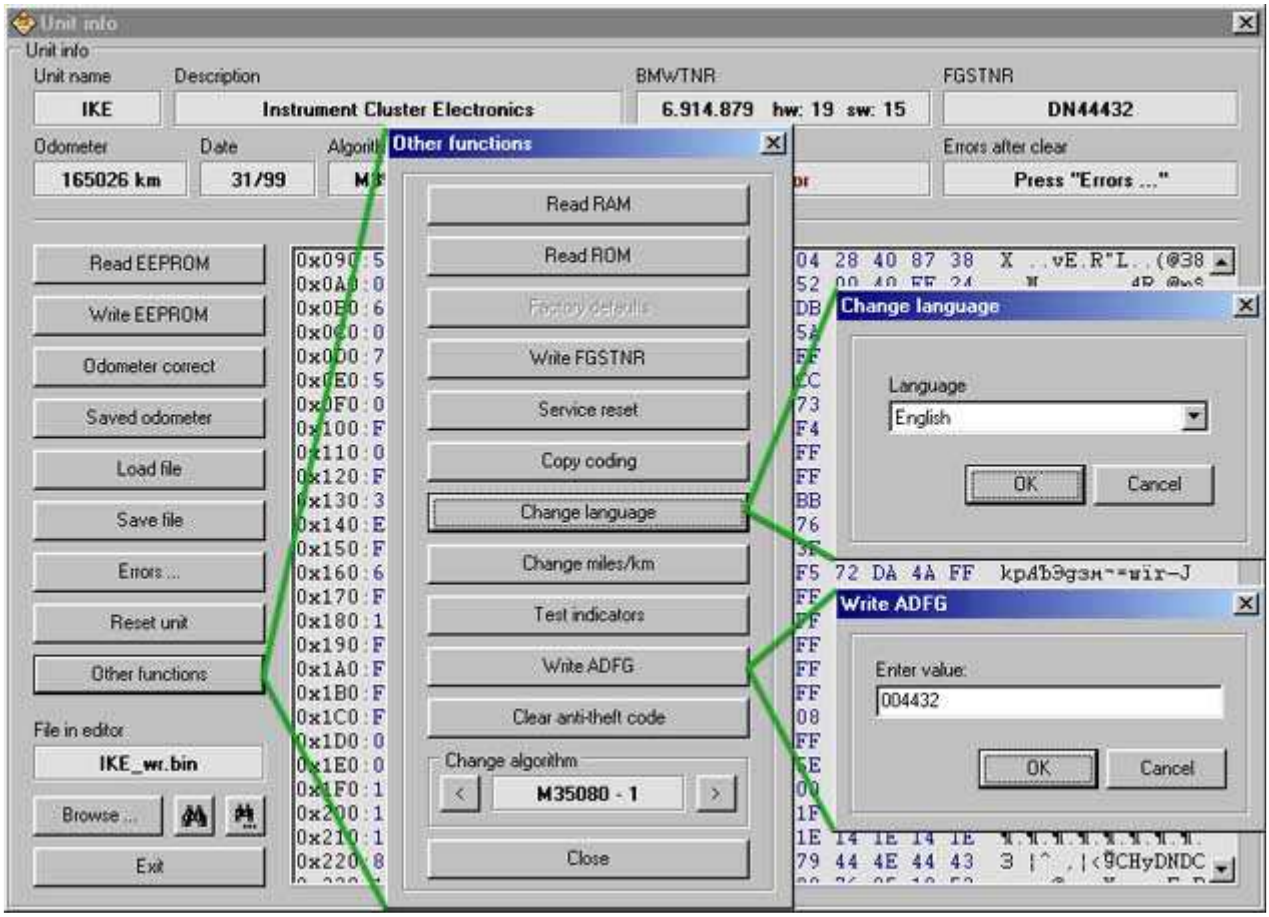
Program functions

Brief list of functions:

[	FIND UNITS	] - scanning installed units with reading identification data and malfunction codes
[	CLEAR ERRORS	] - reset fault codes in all of found units
[	ERRORS DETAILS	] - detail info about malfunction codes
[	READ MEMORY	] - reading unit software memory
[	READ EEPROM	] - reading EEPROM in units: IKE, LCM, ZKE, SRS, LEW, A/C, VID, BMBT, STH, AIC...
[	WRITE EEPROM	] - writing EEPROM in same units
[	ERRORS	] - reading detail info and reset malfunction codes in selected unit
[	RESET UNIT	] - unit software reset
[	WRITE FGSTNR	] - identification number writing
[	WRITE ADFG	] - more identification data writing
[	ODOMETER CORRECT	] - writing new mileage value to IKE & LCM units
[	SAVED ODOMETER	] - reading more detail info about mileage from IKE & EGS units
[	SERVICE RESET	] - IKE service reset
[	CHANGE LANGUAGE	] - changing language of main computer messages GE/EN/FR/IT
[	FACTORY DEFAULTS	] - factory defaults reset of IKE & LCM (for cars till 1999 year)

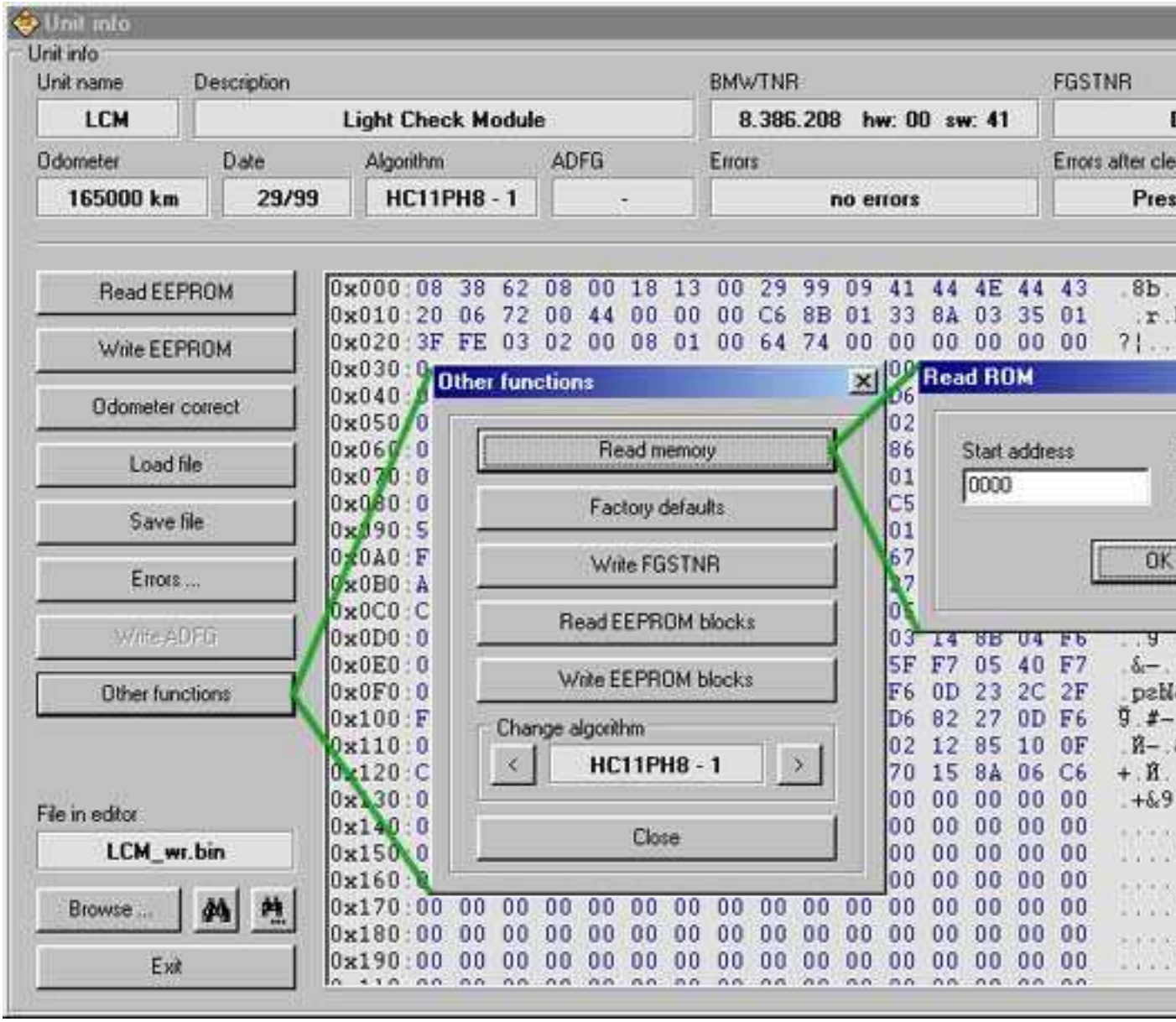
[COPY CODING] - base coding copying IKE to EWS or EWS to IKE units
[TV FUNCTIONS] - "video on when driving" setting in VID unit
[READ KEY] - reading ignition keys data from EWS unit
[ADAPTATION LEW] - steering wheel sensor (LEW) adaptation
[REGISTER HANDSET] - new handset register to TEL (BIT2) unit

In BMW Scanner V1.3.6 you can see the same windows structure, there are a few changes only in functions menu and program database.

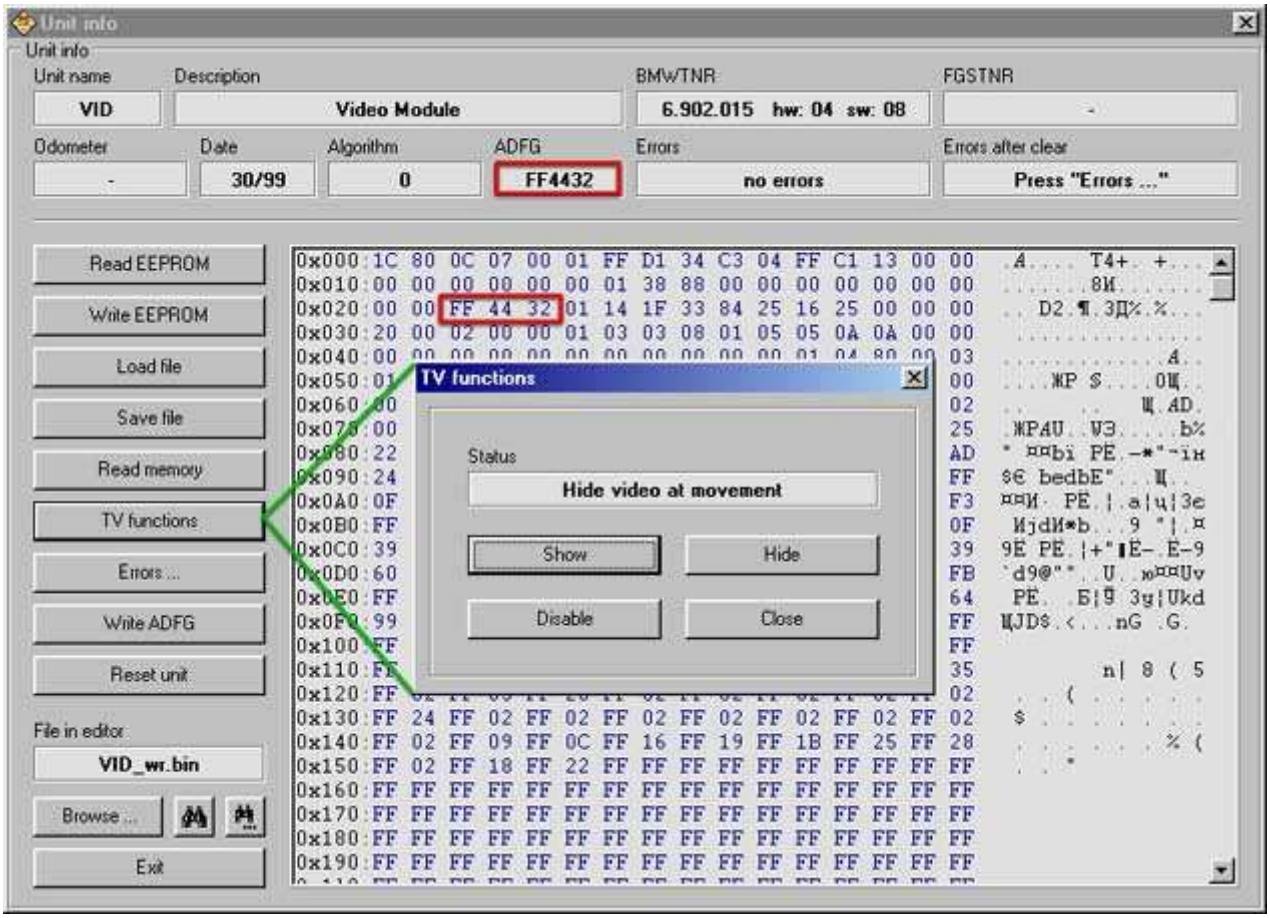


There is different dashboard classification also, using memory access method. For old variant with numbers only (1,2...13) now you can see new classification like: EEPROM memory type + comment (for example 93S56-3, 93S66-Tacho, M35080-E46, etc).

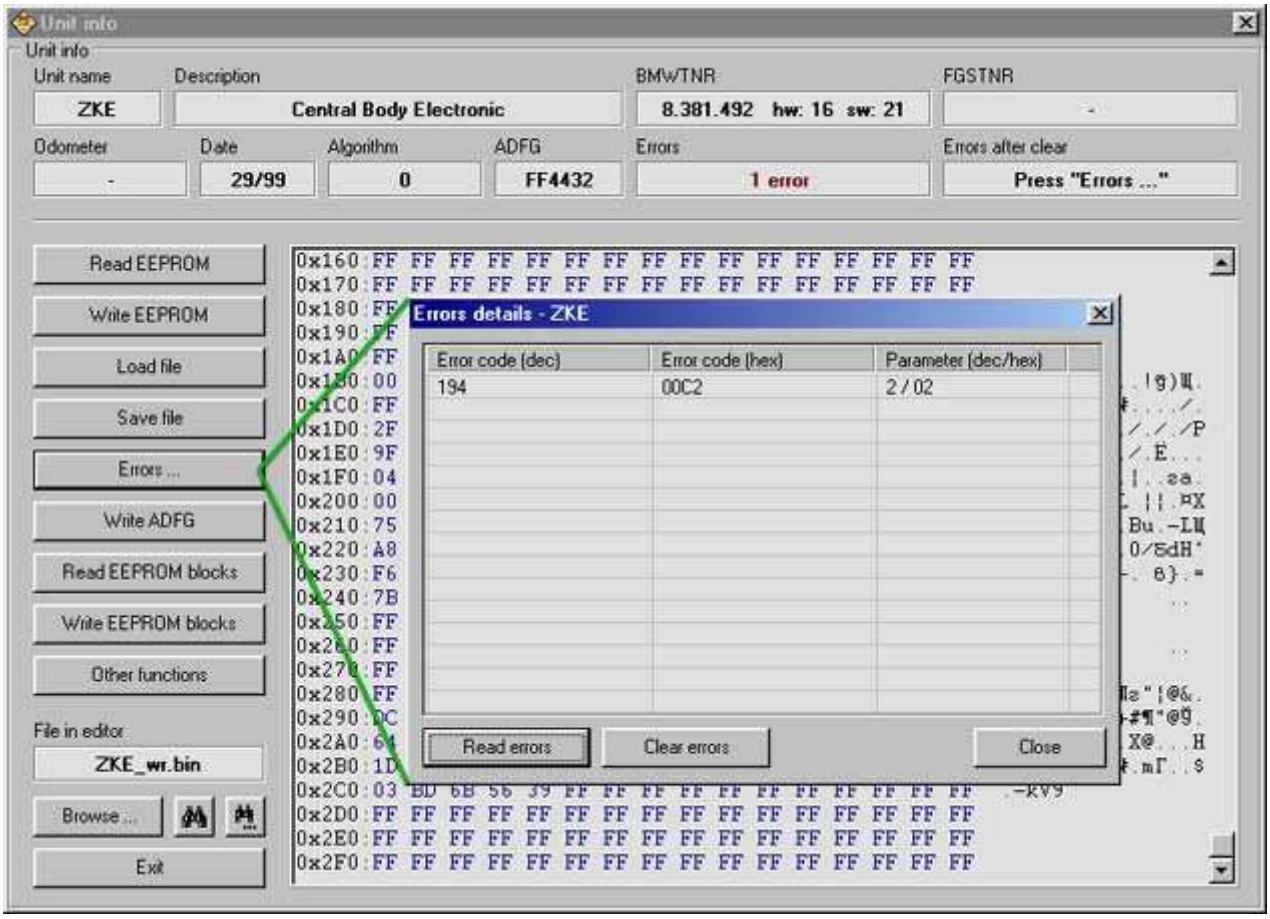
It gives to user more so important information if manual algorithm select using when works with unknown for user dashboard- see "List of supported IKE units".



LCM window is still same with old functions, added only new SW versions of unit. At this moment program can work with all known SW versions of LCM units till 09.2005, including SW76,SW77 - see. "List of tested LCM units".

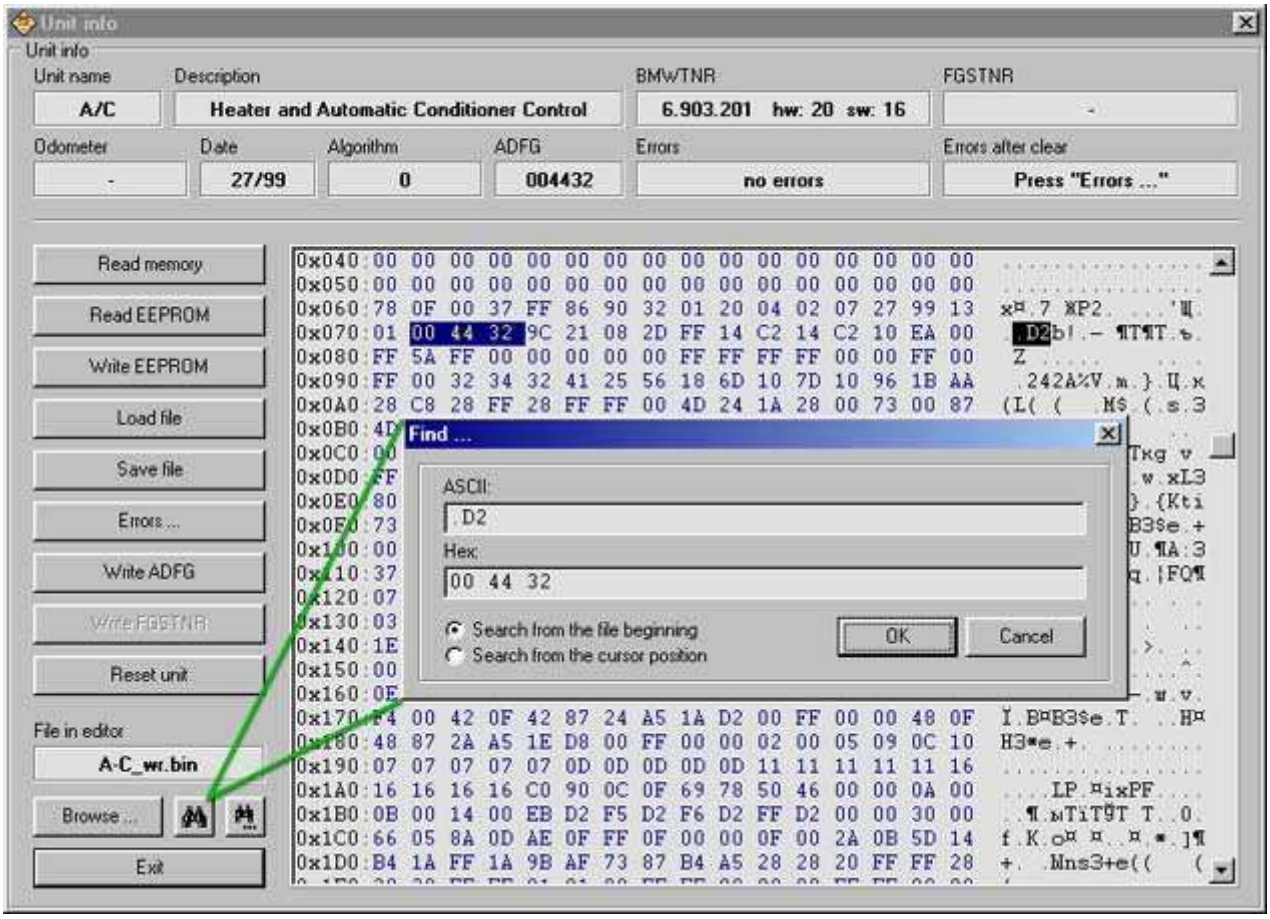


A lot of new functions are available now when work with video module (VID). Added functions like read/write EEPROM memory of unit, software memory reading, and also video when driving settings. This function will help you to set one of 3 possible video settings: show when driving / hide when driving / disable constantly.



On this screenshot you can see example of work with malfunction codes. Sure that is not enough for good diagnostic – you also have to know all malfunction codes decrypting. But its still better that nothing.

Using [ERRORS DETAILS] function is possible also from program main window (double click on mistake count output field).



One more interesting and good function was added to the program, it was done on customers demand – searching on file in editor window. It can be not only data that was read by program from unit, but any bin file that was loaded with [LOAD FILE] or [BROWSE] function.

List of supported IKE & LCM modules

IKE

BMWTNR	HW	SW	EEPROM-IDNR	BMWTNR	HW	SW	EEPROM-IDNR
3.413.124	04	07	M35080 - 2	6.948.237	23	20	M35080 - 1
3.413.132	04	07	M35080 - 2	8.352.207	06	08	93S56 - 1
3.414.370	05	08	M35080 - 2	8.364.005	06	09	93S56 - 1
6.901.921	05	15	93S66 - E46	8.369.051	07	12	93S56 - 1
6.901.922	05	15	93S66 - E46	8.372.354	09	09	93S56 - 3
6.901.923	05	15	93S66 - E46	8.372.354	11	09	93S56 - 3
6.902.362	07	16	93S66 - E46	8.372.354	13	09	93S56 - 3
6.902.362	09	16	93S66 - E46	8.372.357	13	09	93S56 - 3
6.902.363	07	16	93S66 - E46	8.372.359	09	09	93S56 - 3
6.902.374	07	16	93S66 - E46	8.372.359	10	09	93S56 - 3
6.902.375	09	16	93S66 - E46	8.372.359	11	09	93S56 - 3
6.903.748	08	12	93S66 - Tacho	8.372.359	12	09	93S56 - 3
6.903.794	12	16	93S66 - 1	8.372.359	13	09	93S56 - 3
6.903.804	12	16	93S66 - 1	8.372.359	14	09	93S56 - 3
6.906.110	09	13	93S66 - sw13	8.372.359	15	09	93S56 - 3

6.906.110	12	13	93S66 - sw13	8.372.361	11	09	93S56 - 3
6.906.110	14	13	93S66 - sw13	8.374.336	07	13	93S56 - 2
6.906.118	12	13	93S66 - sw13	8.375.669	04	11	93S56 - 4
6.906.119	12	13	93S66 - sw13	8.375.675	04	11	93S56 - 4
6.906.120	12	13	93S66 - sw13	8.375.895	16	10	93S56 - 3
6.906.122	12	13	93S66 - sw13	8.375.898	16	10	93S56 - 3
6.906.124	12	13	93S66 - sw13	8.375.900	16	10	93S56 - 3
6.906.124	14	13	93S66 - sw13	8.375.902	16	10	93S56 - 3
6.906.126	12	13	93S66 - sw13	8.376.192	07	14	93S56 - 5
6.906.148	12	13	93S66 - sw13	8.376.701	04	05	93S56 - Tacho
6.906.154	12	13	93S66 - sw13	8.376.707	04	05	93S56 - Tacho
6.906.889	10	20	M35080 - E46	8.376.708	04	05	93S56 - Tacho
6.906.890	10	20	M35080 - E46	8.378.983	07	15	93S56 - 5
6.906.897	10	20	M35080 - E46	8.380.144	04	07	93S66 - E46
6.906.901	10	20	M35080 - E46	8.380.145	04	07	93S66 - E46
6.906.991	16	14	M35080 - 1	8.380.146	04	07	93S66 - E46
6.906.992	15	14	M35080 - 1	8.381.195	06	15	93S56 - 6
6.906.992	16	14	M35080 - 1	8.381.195	07	15	93S56 - 6
6.906.998	15	14	M35080 - 1	8.381.788	05	07	93S56 - Tacho
6.906.999	15	14	M35080 - 1	8.381.790	05	07	93S56 - Tacho
6.906.999	16	14	M35080 - 1	8.381.806	05	07	93S56 - Tacho
6.907.015	16	17	M35080 - 1	8.381.808	05	07	93S56 - Tacho
6.907.018	16	17	M35080 - 1	8.381.810	05	07	93S56 - Tacho
6.907.021	14	17	M35080 - 1	8.381.812	05	07	93S56 - Tacho
6.911.287	12	23	M35080 - E46	8.386.092	05	13	93S66 - E46
6.911.315	12	23	M35080 - E46	8.386.110	05	13	93S66 - E46
6.914.873	19	15	M35080 - 1	8.387.044	04	11	93S66 - Tacho
6.914.874	19	15	M35080 - 1	8.387.044	05	11	93S66 - Tacho
6.914.879	19	15	M35080 - 1	8.387.044	06	11	93S66 - Tacho
6.914.884	18	15	M35080 - 1	8.387.045	04	11	93S66 - Tacho
6.914.885	18	15	M35080 - 1	8.387.045	05	11	93S66 - Tacho
6.914.885	19	15	M35080 - 1	8.387.045	06	11	93S66 - Tacho
6.914.905	19	15	M35080 - 1	8.387.047	04	11	93S66 - Tacho
6.914.908	19	18	M35080 - 1	8.387.047	06	11	93S66 - Tacho
6.914.913	19	18	M35080 - 1	8.387.063	04	11	93S66 - Tacho
6.914.930	19	15	M35080 - 1	8.387.065	05	11	93S66 - Tacho
6.915.237	21	31	M35080 - 2	8.387.075	04	11	93S66 - Tacho
6.931.244	22	32	M35080 - 2	8.387.075	05	11	93S66 - Tacho
6.932.907	27	35	M35080 - 2	8.387.604	05	14	93S66 - E46
6.940.589	21	17	M35080 - 1	8.387.605	05	14	93S66 - E46
6.942.546	24	21	M35080 - 1				

LCM

HW	SW	MCU-IDNR	HW	SW	MCU-IDNR
00	11	HC11P2-1	08	11	HC11P2-4
00	31	HC11P2-2	08	13	HC11P2-4
00	41	HC11PH8-1	08	36	HC912-4
00	42	HC11PH8-1	08	40	HC11PH8-2
00	43	HC11PH8-1	08	76	HC912-11
01	11	HC11P2-1	09	13	HC11P2-4
01	20	HC11P2-1	09	36	HC912-4
01	21	HC11P2-1	09	40	HC11PH8-2
01	31	HC11P2-2	09	76	HC912-11
01	41	HC11PH8-1	10	20	HC11P2-3
02	42	HC11PH8-1	10	31	HC11P2-2
02	43	HC11PH8-1	10	32	HC11P2-2
04	11	HC11P2-4	10	41	HC11PH8-1
04	51	HC912-2	11	41	HC11PH8-1

05	11	HC11P2-4	15	30	HC912-5
05	51	HC912-2	16	31	HC912-7
06	11	HC11P2-4	22	35	HC912-6
06	52	HC912-8	24	36	HC912-4
06	53	HC912-3	25	36	HC912-4
07	11	HC11P2-4	C1	40	HC11PH8-2
07	20	HC912-9	C1	41	HC11PH8-1
07	65	HC912-1	C1	43	HC11PH8-1
07	70	HC912-10	D1	41	HC11PH8-1
07	76	HC912-11	D1	42	HC11PH8-1
07	77	HC912-11			

About less known mileage copies in BMW of 1999-2005 year cars

---

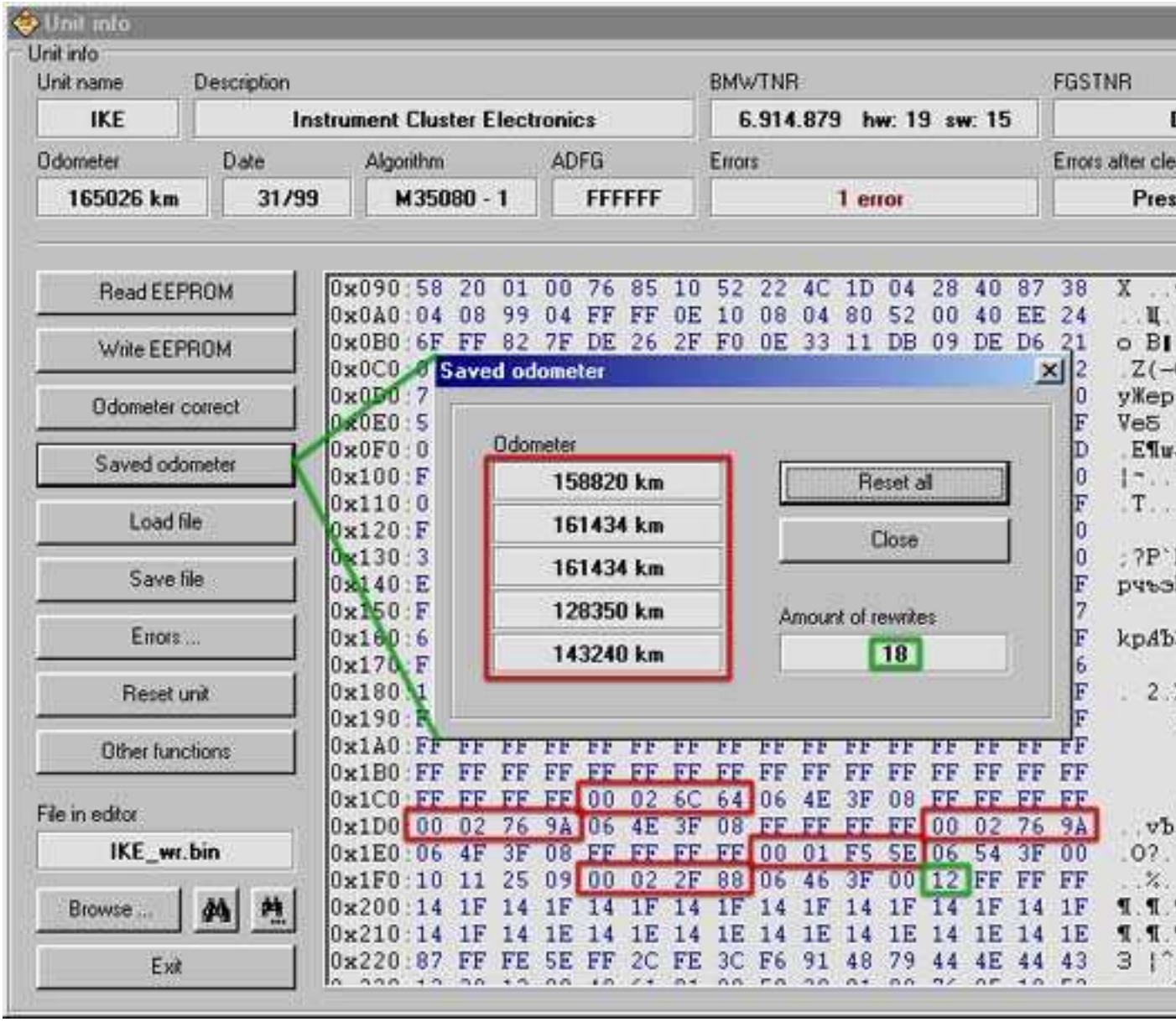
There is same information on almost all of the people about where mileage data in BMW is situated. Well known standard list of units contains IKE, LCM, EWS and malfunction codes parameters also. But, as usually, real situation is a little bit harder and some part of units contains hidden data also that helps us to know is digits on dashboards LCD true or false.

Calling this data as mileage copies is not absolutely correct because its not current mileage but saved when some of diagnostic procedures was done value. Good thing here is that reading of such data from units gives us very important information when car condition analyze has to be done.

Lets try to explain this all on examples.

First example is SW14-SW21 (M35080, E38/E39/E53 of 2000-2005 year) and SW13 (93S66 of 1999-2000 year) IKE units.

All necessary and interesting information about it you can see on next screenshot:



It looks like service data saving method in MB dashboards: serial cycle contains five write procedures + writing counter on #01FC area.

Most interesting here is that next writing procedure can be started when dashboard software reset will be done only. Sure that reason of such restart can be not only some diagnostic procedures using (like coding, service reset, etc). Next data writing will be done also if car battery or dashboard power fuse removed, and then installed back sure.

It means that data saving depends from random things, for example it can be service worker knowledge, some kind of malfunction or something else. Its hard to say for sure is this bad or good thing. Anyway, as result of such processes we have useful information that helps us to know not only about real mileage but also even about previous car using.

So lets talk a little about interesting and useful ways to work with this data.

First way is very short and easy, software reset of necessary units could be done using [RESET ALL] program function. This function contains five serial commands of software reset. As result of them previous mileage writing to 5 areas and counter value is +5. So when next status reading will be done, all necessary data is correct.

Second way is much more interesting. All necessary thing to do is write 5 different mileage

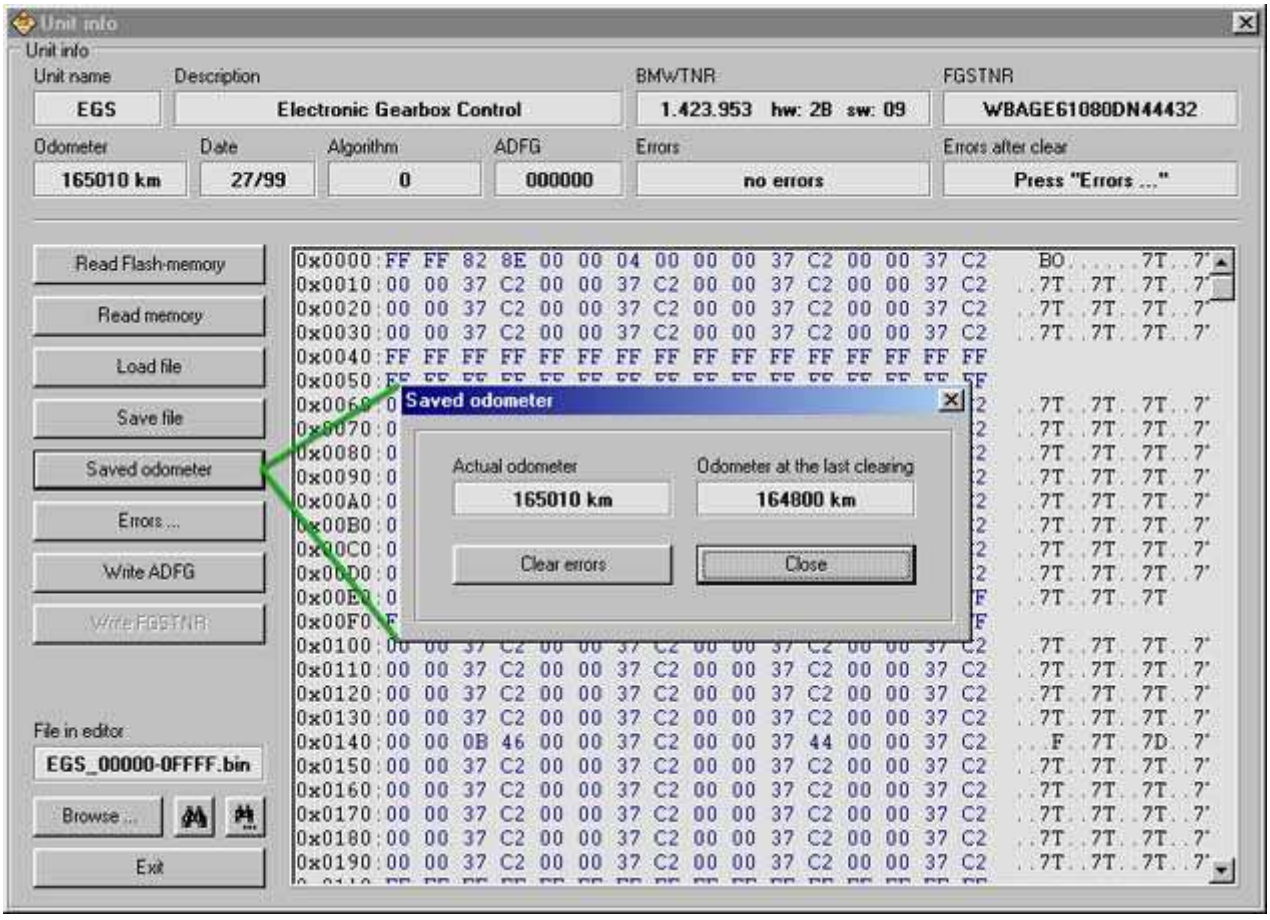
values less than current to their areas using [WRITE EEPROM] function. For example mileage difference between each of 5 values can be 15000km (like service resets). Most important thing here is to make biggest mileage value truly with writing counter.

Writing process is same serial cycle, starting from #01C4 memory area, continuing into #01D0,#01DC,#01E8,#01F4 and then to #01C4 again. If counter value is 9, then correct position of last writing is #01E8. Else we can see that mileage was corrected.

One more example of hidden mileage data is in automatic transmission (EGS) memory.

But I'm talking not about malfunction codes parameters because it's well known to everybody and solution of this is very simple – malfunction codes reset. After this procedure previously saved data is erased and new malfunction codes getting parameters that already contains new mileage value.

But not all of people know that most part of EGS units (HW23/28/29/2B/2C/2F/2H) as answer to standard diagnostic procedure except malfunction codes also sending two mileage values. First is current mileage, second is mileage value when last unit reset was done. Data output format is N/10(hex).



All diagnostic software as usually works with malfunction codes only and all other data is ignored. This situation making possible one bad mistake. After mileage correction in case when no malfunction codes wasn't found by diagnostic software malfunctions memory is not resetting and old mileage is saved there.

That's why working on car process must be done step by step. First step is full scanning of all units but without malfunction codes reset (you can use [FIND UNITS] function of main menu). It's making possible saving of all necessary data into log-file, including original mileage value.

Sure that last step is scanning of all units with malfunction codes resetting (using [CLEAR ERRORS] function of main menu). Manual malfunction codes reset could be done also from program main window, using [ERRORS DETAILS] function (see also "Program functions")

Some features of EEPROM M35080 read & write process

Well known thing that in all dashboards that was manufacted till 2000 year (with 93S56, 93S66 chip inside) EEPROM memory could be read via car diagnostic socket fully same as when using programmer.

A little different situation in new dashboards (manufacted after 2000) in E38/E39/E53 of 2000-2005 year (SW14-SW21, EEPROM M35080). When compare two dumps that was read via diagnose and using programmer we can see next picture:

DIAG.	=	PROG.
#0000-#0057	=	#0020-#0077
#0058-#01FF	=	#0200-#03A7
#0200-#021F	=	#0000-#001F
#0220-#03FF	=	#0220-#03FF

Well seen that full memory dump contains #0220-#03A7 data twice in #0078-#01FF and off course in #0220-#03A7 M35080 memory areas.

In first part of dump (that was read by programmer) we can see that developers tried to make special M35080 data format compatible with 93S56/93S66. Maybe it was done to make possible work with new cars using old software versions of diagnostic devices (to data was written into necessary memory fields correct).

But situation in second part of memory dump is not very good. You can get it from next example: some data in same (when read via diagnose) areas have different properties.

Lets look on this example more detail. Data writing into #0084-#0089 fields is denied by software protection for good known reasons. But data writing into #022C-#0231 area using standart commands of BMW protocol works perfect that is not have to be possible because its same areas in real (read by programmer) memory dump. So here we have one more mistake of BMW developers that continuing with more and more new SW units versions…

This situation also bad for BMW scanner users, because they getting a lot of problems with necessary memory fields writing.

We cant forget also that data in bought areas of dump read via diagnose is same area in M35080. That's why before data writing you have to change bought areas in working window of BMW scanner.

If data corrected in #0078-#01FF area only then it not will be written to chip. Actually it will be written twice. First data will be written to #0078-#01FF and then old one will be restored from #0220-#03A7 area.

If data will be written to #0220-#03A7 only - writing will be done, but program will finish writing procedure with mistakes error message (verification from #0078-#01FF area cant be done). The count of mistakes in error message has to be same as changed bytes count.

There is no so big deal here if user know about all processes that starts when write function used. But if not people usually think that program works bad =)

Also I want to tell that in more new dashboards SW versions (E83/SW07-SW08, E46/SW31-SW35) BMW developers fixed their bug. That's why all IKE units memory that

was detected like [M35080 - 2] could be read via diagnose absolutely same like when using programmer.

ADFG - a "present" from BMW ?

Here we have one more example that reading datasheets and units firmware analysis are very useful things sometimes.

As result of such analysis was discovered that EEPROM memory of almost all units contains special data in area of 3 bytes only but very interesting. BMW diagnostic protocol contains two special commands to work with this data. First of them to read memory area, second - to write it.

That's nothing strange here, except one thing - that's no any menu function in diagnostic devices (including GT-1) to work with this data. It means that this data can be written by manufacter only when identification memory programming.

Most interesting is containing of these 3 bytes. More than ten units (ZKE, A/C, NAV2, RAD, LEW, TEL, SHD, PDC, SMF, AHL, etc.) contains there last 4 digits of VIN. Come other units in this area contains date of programming. In all other units only 00 (than means VIN wasn't written) or FF (no any data wasn't written).

In this article you can see that BMW scanner V1.3.6 is more than enough to make expert car analysis. But its possible only with cars that was manufactured after 1998, which have ADFG data.