



Document type
DESCRIPTION
 Title
 XCOM User Guide
 File name
 XCOM User Guide
 Date
 2012-10-10
 Issue

Approved by (department acronym name)

Issued by (department acronym name phone)
NEVE Fransson, Jonas 50 400

Info class
Internal
 Page
1(33)

To (department acronym name)

For information (department acronym name)

XCOM User Guide

Introduction	2
General.....	2
Concepts/abbreviations	2
Document history	3
Use cases	4
Use case: Connect to CAN.....	4
Use case: Retrieve data from vehicle	6
Use case: Program E2 parameter configuration values	8
Use case: Clear memory areas	11
Memory area.....	11
DCMN memory area	13
OPC4/RET1 statistic tables	13
Use case: Reset ECU	14
Use case: Recover defective ECU	15
GUI	17
General.....	17
File menu	18
ECU explorer.....	18
Configuration	20
I/O control.....	23
Measurement	24
Read/Write Memory	27
KWP signalling	29
Trace	31

Introduction

General

This document describes how to use XCOM – a diagnostic tool to read, save and modify ECU data such as trouble codes, parameters, statistic data etc.

XCOM supports modification of ECU data of the following types:

- 1) EMS – Engine management system
- 2) GMS – Gearbox management system
- 3) RET – Retarder system
- 4) AWD – All wheel drive system
- 5) CLS – Clutch Control
- 6) EEC – Exhaust Emission Control
- 7) COO (COO7) – Coordinator
- 8) BWS (BC11) – Bodywork system

XCOM supports resolving trouble codes to text description for the following types:

- 1) EMS (S6, S7, S8)
- 2) GMS (OPC4, OPC5)
- 3) RET (RET1, RET2)
- 4) AWD (AWD1)
- 5) EEC (SCR1, EEC3)
- 6) COO (COO6, COO7)
- 7) HMS (HMS1)

XCOM is installed in “C:\Program Files\Scania XCOM\” on 32 bit computers and in “C:\Program Files (x86)\Scania XCOM\” on 64 bit computers.

Concepts/abbreviations

<i>What</i>	<i>Meaning</i>
DTC	Diagnostic Trouble Code
DEC	Diagnostic Event Code
FFs	Freeze frame data
DCMN	Duty cycle monitor data
EOL	End Of Line
VIN	Vehicle Identification Number



Document history

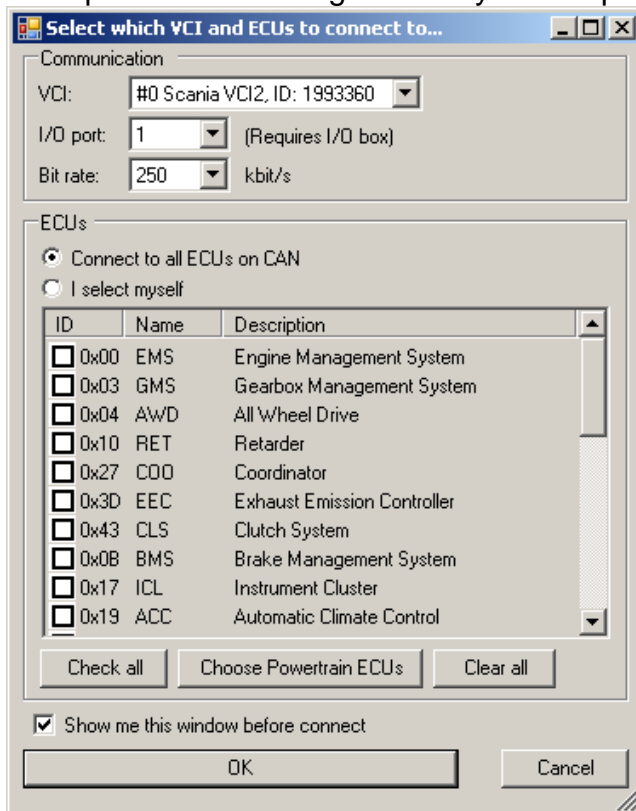
<i>Author</i>	<i>Revision</i>	<i>Date</i>	<i>Comment</i>	<i>Approved</i>
sssfjs	3.4	2012-08-23	Updated to XCOM 2.18.0	
sssfjs	3.3	2012-06-08	Updated to XCOM 2.17.0	
sssfjs	3.2	2011-09-07	Updated to XCOM 2.13.0	
sssfjs	3.1	2011-03-07	Updated to XCOM 2.12.1	
sssfjs	3.0	2011-02-28	Updated to XCOM 2.12.0	
sssfjs	2.4	2010-12-13	Updated to XCOM 2.11.1	
sssfjs	2.3	2010-11-24	Updated to XCOM 2.11.0	
sssfjs	2.2	2010-04-27	Updated to XCOM 2.10.0	
sssfjs	2.1	2009-09-08	Updated to XCOM 2.7.1	
sss339	2.0	2009-08-06	Updated to XCOM 2.7.0	
sssfjs	1.9	2009-06-22	Updated to XCOM 2.6.1	
sssfjs	1.8	2009-06-03	Updated to XCOM 2.6.0	
sssfjs	1.7	2009-04-06	Updated to XCOM 2.5.0	
sssfjs	1.6	2008-12-18	Updated to XCOM 2.4.0	
sssfjs	1.5	2008-11-04	Updated to XCOM 2.3.2	
sss339	1.4	2008-08-13	Updated to XCOM 2.2.0	
sssfjs	1.3	2007-11-02	Updated to XCOM 2.0.0	
sssfjs	1.3	2007-09-05	Updated to XCOM 1.3.0	
sssfjs	1.2	2007-05-04	Updated to XCOM 1.2.0	
sssfjs	1.1	2007-05-29	Updated to XCOM 1.1.0	
sssghd	1	2007-04-13	First draft	

Use cases

Use case: Connect to CAN

To connect to CAN via Scania VCI interface and the connected ECU the user performs the following steps

- 1) Start XCOM application
- 2) Connect to CAN by either
 - a. Select the menu choice File -> Connect to CAN
 - b. Press the "F4" key
 - c. Press the keys "shift" and "F4"
- 3) If the user has the option setting "display the select ECUs" is enabled the "Select ECUs" dialog will be displayed. If the shift + F4 were pressed the dialog will always be displayed.

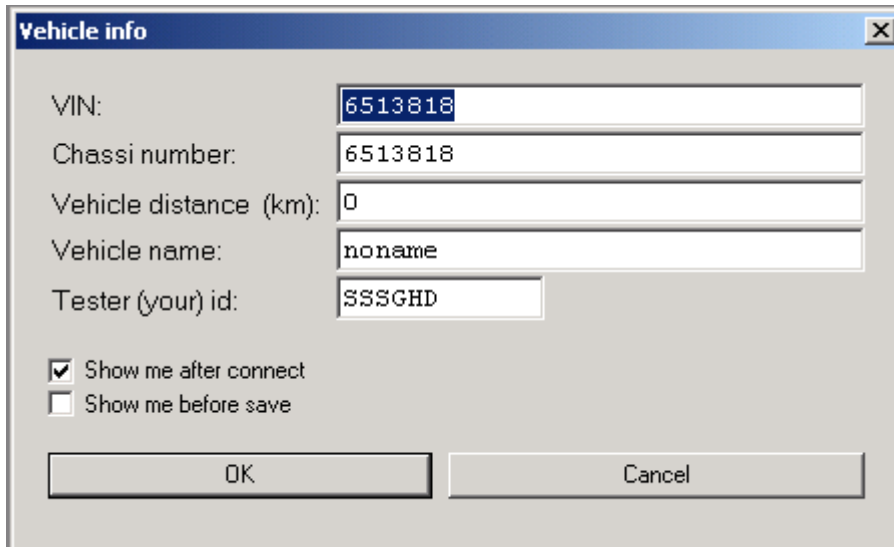


- 4) In the "Select ECUs" dialog the user can either...
 - a. Scan for all ECUs available on CAN (a functional SetSession KWP command will be transmitted on CAN) by selecting the radio button choice "Connect to all ECUs on CAN" and pressing the OK-button

- b. Scan for those ECUs that has the check mark set in the list (a physical SetSession KWP command will be transmitted to the marked ECU's CAN addresses) by selecting the "I select myself" radio button choice and press the OK-key.

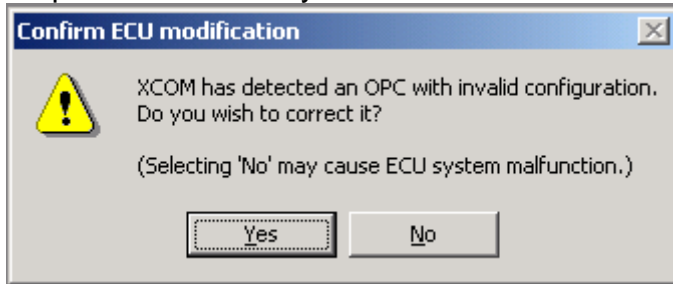
If there are more than one VCI connected to the computer the user can select which VCI to use when connecting to ECUs. The VCI ID can be found on the back of the VCI.

- 5) The application now connects to CAN via KWP.
- 6) When the connect sequence is complete the "Edit Vehicle Info" dialog is displayed. The data in the dialog is used when creating filenames when data is saved to file in various function/data forms.



- 7) In the dialog the user can change information about the connect session such as VIN, chassinumber, vehicle distance, vehicle name and tester name. Vehicle distance is only gathered from COO. VIN and chassinumber are gathered from COO. If no COO is found the information is retrieved from EMS, then GMS or as last resort the RET. Any changes of values by the user is kept throughout the connect session (until the user disconnects from CAN) and used to create proposal of filename when saving data. The vehicle name is resolved from the chassinumber.
- 8) If the application detects ECUs with invalid configuration, a message box will be displayed to query the user whether to allow the application to correct the ECU or not. Pressing "yes" will do the modification (recommended) and "no" will proceed the connect

sequence without any action.

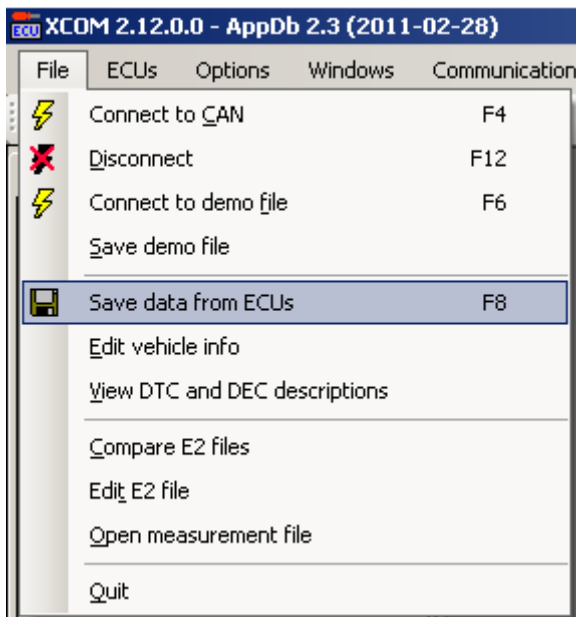


- 9) The sequence is now complete and the connected ECUs are shown in the ECUs menu and in the ECU explorer tree.

Use case: Retrieve data from vehicle

To save ECU data to file the user performs the following steps.

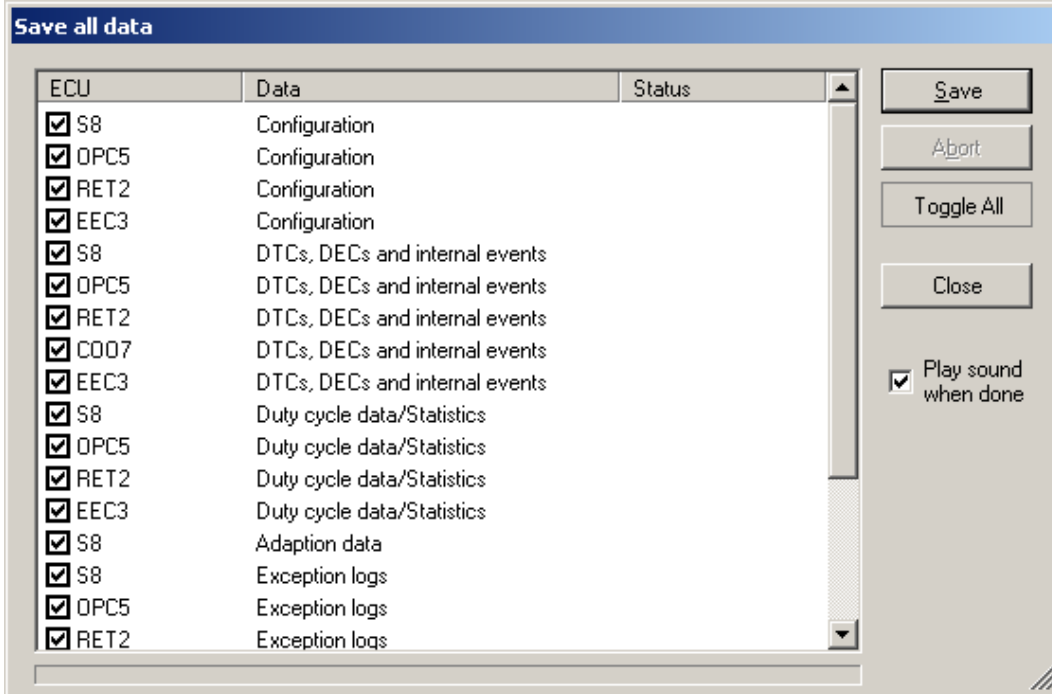
- 1) Connect to CAN (see use case “Connect to CAN”)
- 2) Select in the file menu “Save data from ECUs...”



Approved by (department acronym name)

Issued by (department acronym name phone)
NEVE Fransson, Jonas 50 400

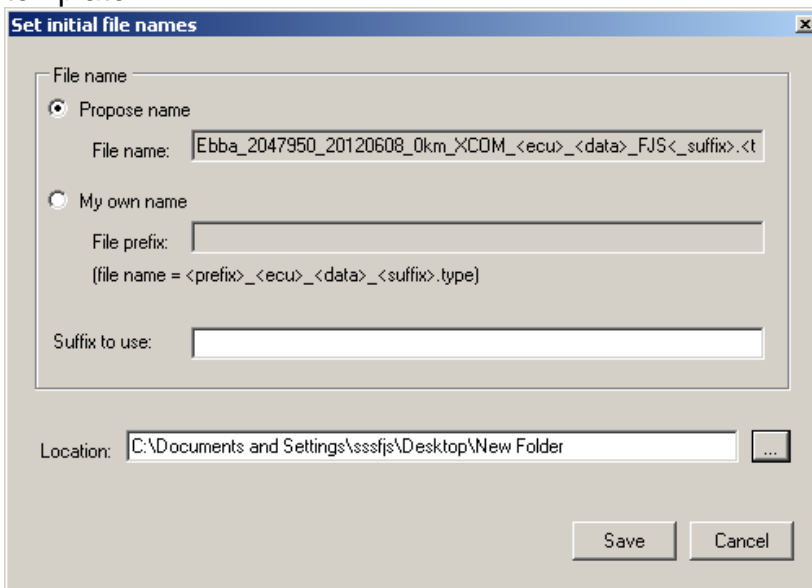
3) A dialog will appear:



4) Select the data that you want to save data from by checking the boxes in the right field.

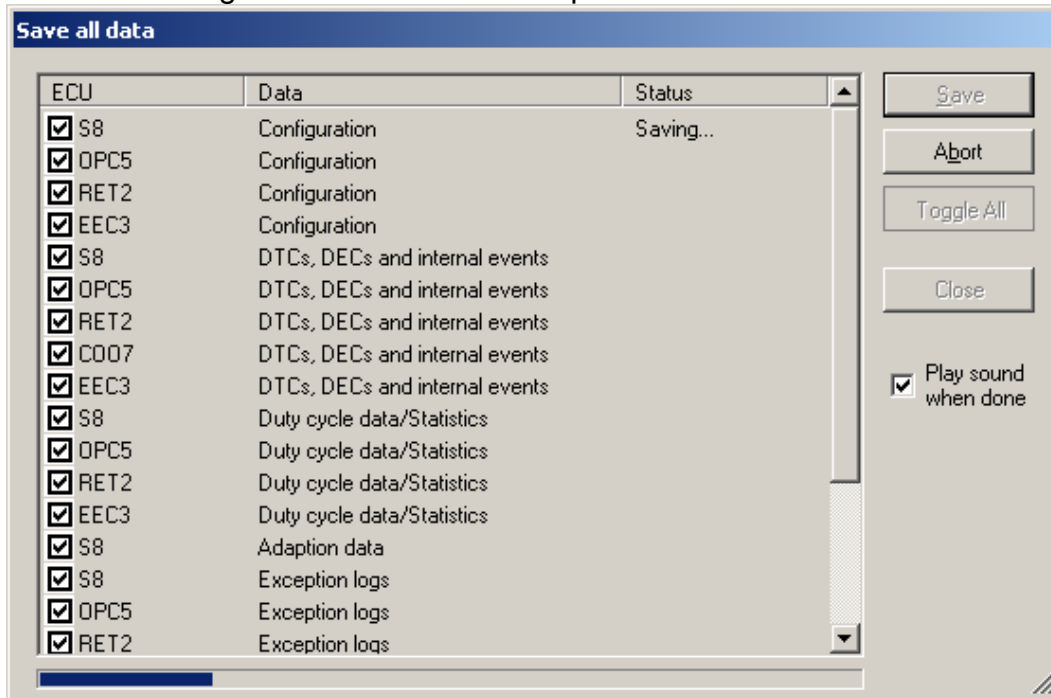
5) Press the "Save" button.

6) A new dialog will appear and prompt the user to choose a file name template.



Specify the location of where the application shall store the output files by click on the "..." button next to the "Location" text area.

- 7) Press the "Save" button to start retrieving and saving of the ECU data. The dialog is closed and the save procedure is started.



The save procedure will take some minutes.

- 8) The procedure is complete when window is enabled again.
- 9) The user verifies that a number of files have been created in the directory previously specified.
- 10) Press the "Close" button to close the dialog.

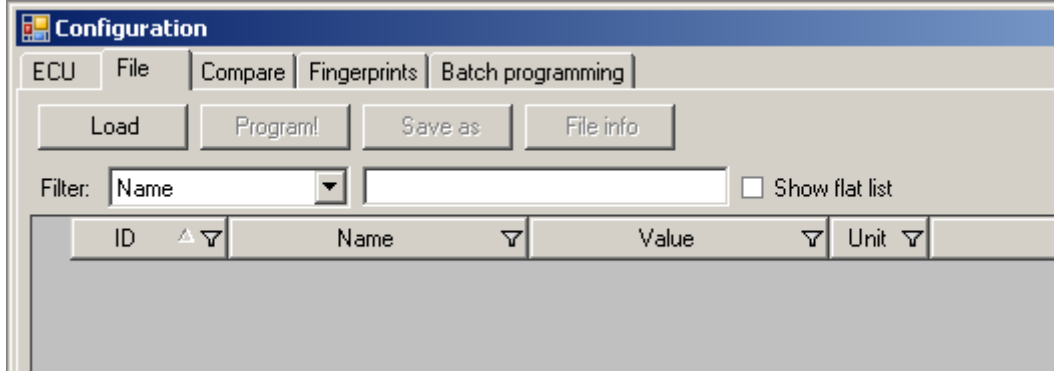
Use case: Program E2 parameter configuration values

To load configuration data into ECU, perform the following steps.

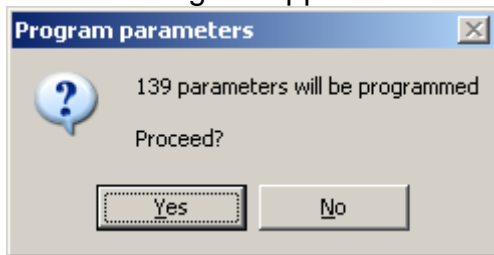
- 1) Connect to CAN (see use case "Connect to CAN")
- 2) Open the Configuration form for the ECU that is to be programmed. This is done either by expanding the ECU node in explorer tree and double clicking on the "Configuration" function/data node or by selecting the menu choice ECUs->(the EcuName)->Configuration.

The Configuration form is displayed.

- 3) Go to the "File" tab.

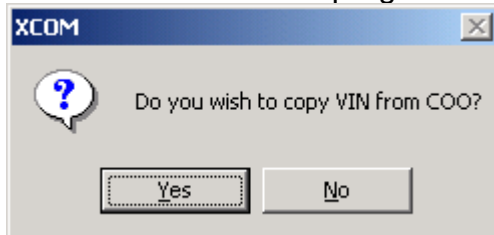


- 4) Load the XCOM E2 xml file that contains configuration data for the ECU by clicking the "Load..." button. (This can take some seconds.)
- 5) Push the "Program!" button in the "File" tab to program the ECU. A confirm dialog will appear:



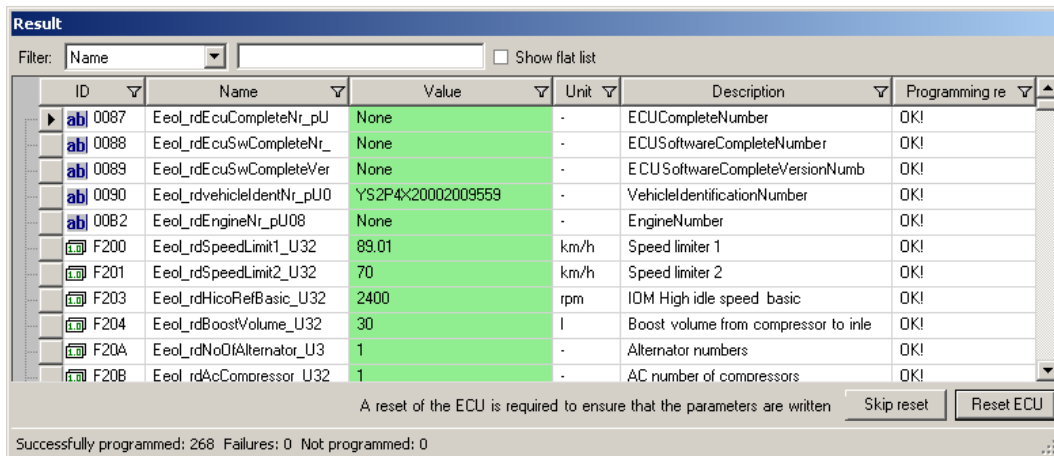
Press "Yes" to confirm that the ECUs parameters is to be programmed.

- 6) A new dialog is displayed to query if VIN shall be copied from COO to the ECU that will be programmed.



Press "Yes" to confirm (you must be connected to a COO in order for it to work) or "No" to skip this operation.

- 7) The parameters that were not successfully programmed from file will be marked with red, those that were programmed successfully will be marked green. Parameters that did not exist in the file are marked yellow.

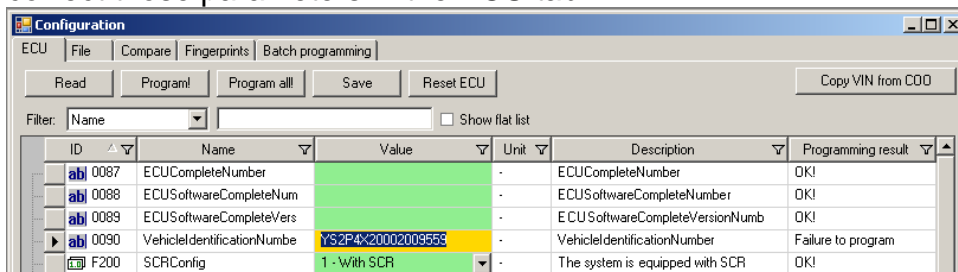


ID	Name	Value	Unit	Description	Programming re
ab 0087	Eeol_rdEcuCompleteNr_pU	None	-	ECUCompleteNumber	OK!
ab 0088	Eeol_rdEcuSwCompleteNr_	None	-	ECUSoftwareCompleteNumber	OK!
ab 0089	Eeol_rdEcuSwCompleteVer	None	-	ECU SoftwareCompleteVersionNumb	OK!
ab 0090	Eeol_rdvehicleIdentNr_pU0	YS2P4x20002009559	-	VehicleIdentificationNumber	OK!
ab 0082	Eeol_rdEngineNr_pU08	None	-	EngineNumber	OK!
F200	Eeol_rdSpeedLimit1_U32	89.01	km/h	Speed limiter 1	OK!
F201	Eeol_rdSpeedLimit2_U32	70	km/h	Speed limiter 2	OK!
F203	Eeol_rdHicoRefBasic_U32	2400	rpm	IDM High idle speed basic	OK!
F204	Eeol_rdBoostVolume_U32	30	l	Boost volume from compressor to inle	OK!
F20A	Eeol_rdNoOfAlternator_U3	1	-	Alternator numbers	OK!
F20B	Eeol rdAcCompressor_U32	1	-	AC number of compressors	OK!

A reset of the ECU is required to ensure that the parameters are written Skip reset Reset ECU

Successfully programmed: 268 Failures: 0 Not programmed: 0

- 8) To complete programming:
- If all parameters are OK, press “reset ECU” to complete the programming.
 - If there are red or yellow parameters, press “skip reset” and correct those parameters in the ECU tab:



ID	Name	Value	Unit	Description	Programming result
ab 0087	ECUCompleteNumber		-	ECUCompleteNumber	OK!
ab 0088	ECUSoftwareCompleteNum		-	ECUSoftwareCompleteNumber	OK!
ab 0089	ECUSoftwareCompleteVers		-	ECU SoftwareCompleteVersionNumb	OK!
ab 0090	VehicleIdentificationNumbe	YS2P4x20002009559	-	VehicleIdentificationNumber	Failure to program
F200	SCRConfig	1 - With SCR	-	The system is equipped with SCR	OK!

And press “program!”.

- 9) The sequence is now complete.

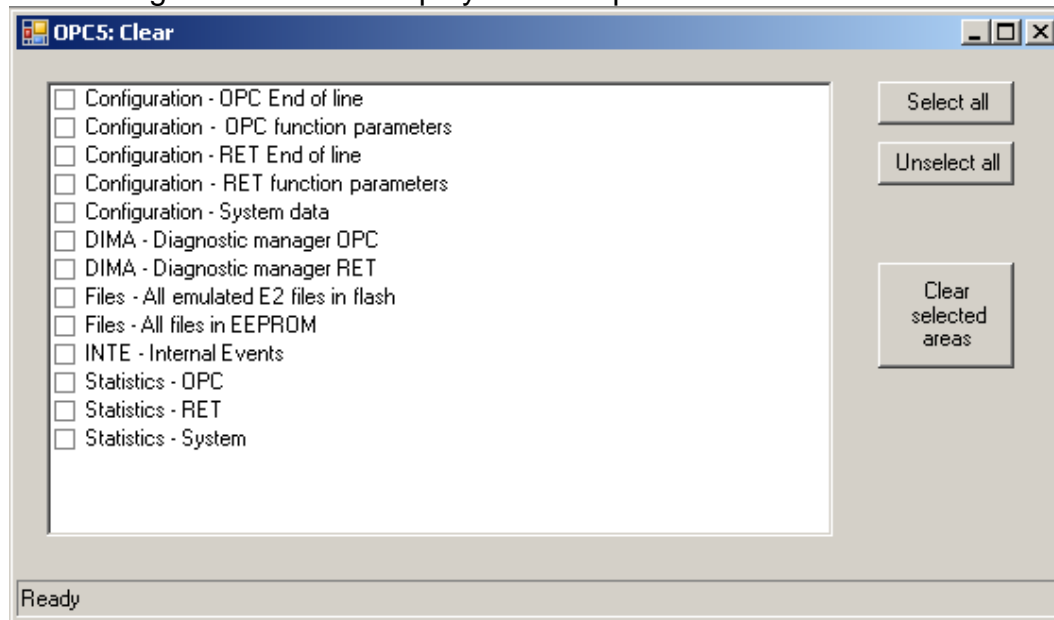
Use case: Clear memory areas

Memory area

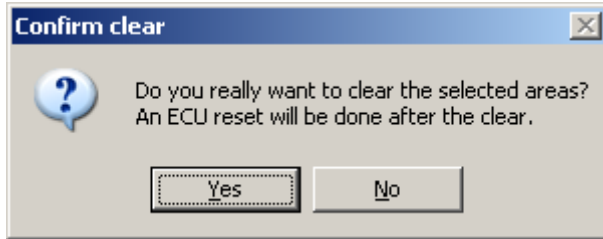
To clear (reset) data areas, perform the following steps.

- 1) Connect to CAN (see use case "Connect to CAN")
- 2) Open the "Clear" form for the related ECU. This is done either by expanding the ECU node in explorer tree and double clicking on the "Clear" function/data node or by selecting the menu choice ECUs->(the EcuName)->Clear.

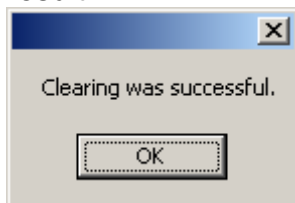
The Configuration form is displayed. Example:



- 3) Select the area that is to be programmed by making the check mark visible for the name of the area in the list box.
- 4) Press the "Clear selected areas" button to reset the data area.
- 5) When the button is pushed a dialog is displayed to confirm the operation:



- 6) Press "Yes" to start the reset sequence. An ECU reset will be performed when the clear is done.
- 7) When the sequence is complete a dialog is displayed to report the result:

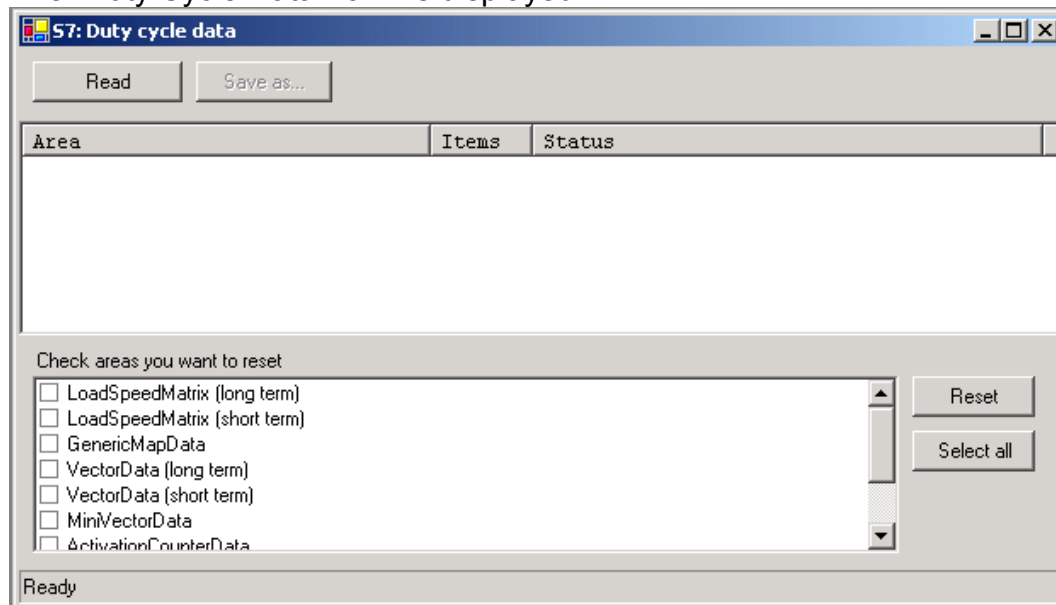


DCMN memory area

To clear (reset) DCMN data areas, perform the following steps.

- 1) Connect to CAN (see use case "Connect to CAN") if not already connected.
- 2) Open the "Duty Cycle Data" form for the related ECU. This is done either by expanding the ECU node in explorer tree and double clicking on the "Duty Cycle Data" function/data node or by selecting the menu choice ECUs->(the EcuName)->Duty Cycle Data.

The "Duty Cycle Data" form is displayed

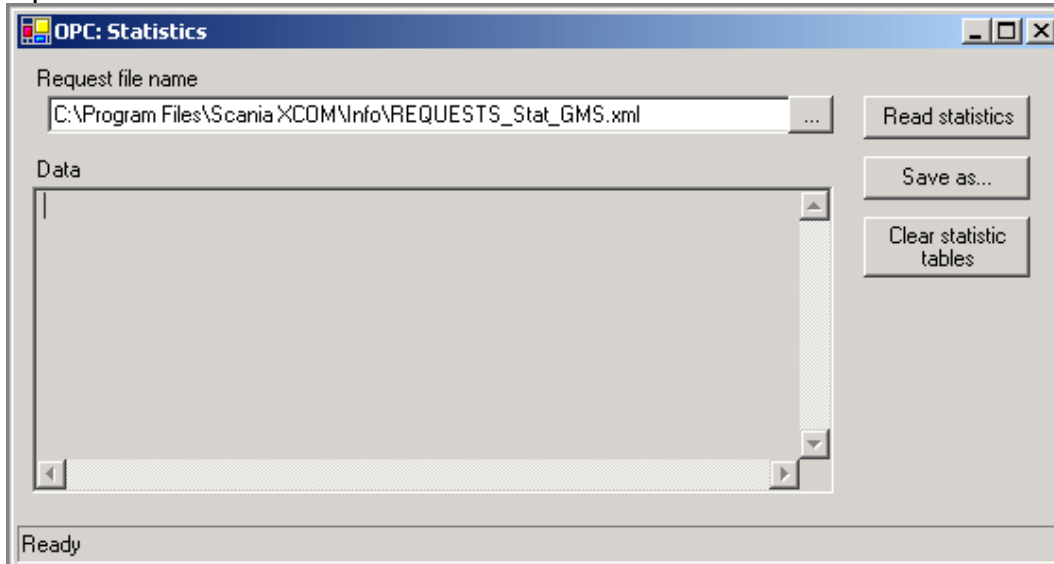


- 3) Select the areas that are to be cleared by putting the check mark in the boxes of the names in the list box.
- 4) Clear the areas by pressing the "Reset" button.
- 5) The application will erase the areas and display the result in the main form's status window.
- 6) The sequence is now complete

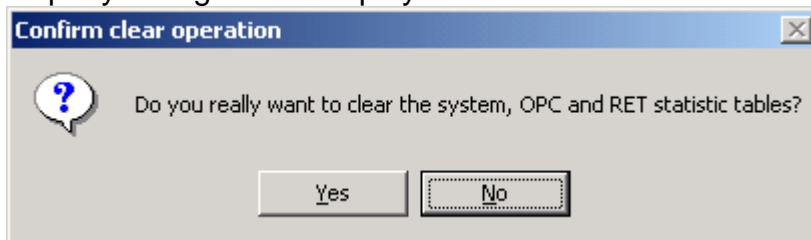
OPC4/RET1 statistic tables

To clear the OPC/RET statistics tables, perform the following steps.

- 1) Connect to CAN (see use case “Connect to CAN”) if not already connected.
- 2) Open the ”Statistics” form for the a OPC or RET.



- 3) Click the “Clear statistics tables” to clear the tables.
- 4) A query dialog will be displayed to confirm the action.



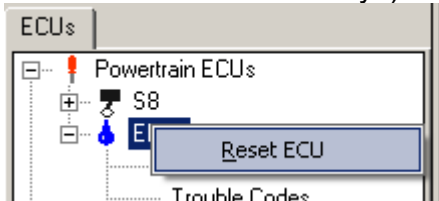
- 5) Press ‘Yes’ to proceed and ‘No’ to abort the sequence.
- 6) The result of the action is displayed in the main form’s status window.
- 7) The sequence is now complete.

Use case: Reset ECU

To reset an ECU, perform the following steps.

- 1) Connect to CAN (see use case “Connect to CAN”) if not already connected.

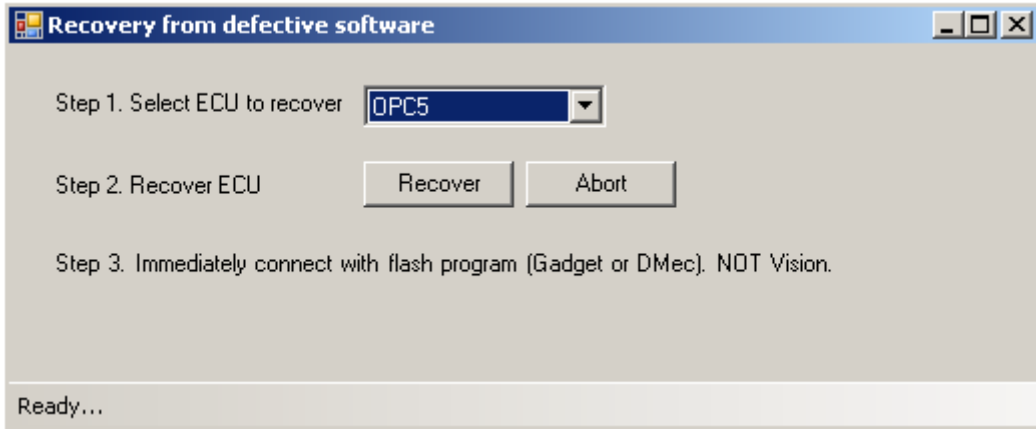
- 2) Display the ECU pop-up menu either by clicking the ECU node in the ECU explorer tree with the right-side button on the mouse or pressing the "pop-up" key when the related ECU is selected (which is done with the arrow keys).



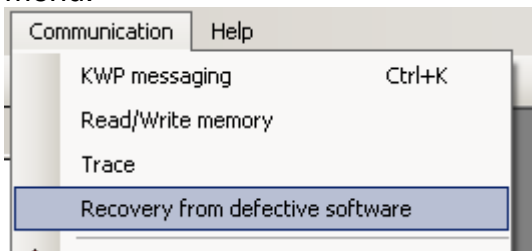
- 3) Pressing "OK" will reset the ECU, close all open windows and redraw the tree of ECUs. The ECU is reset with "keyOn".
- 4) The sequence is now complete.

Use case: Recover defective ECU

If an ECU has been flashed with a defective software and other ECUs reports that it is missing it can be recovered with XCOM.



- 1) Select "recovery from defective software" from the Communication menu.

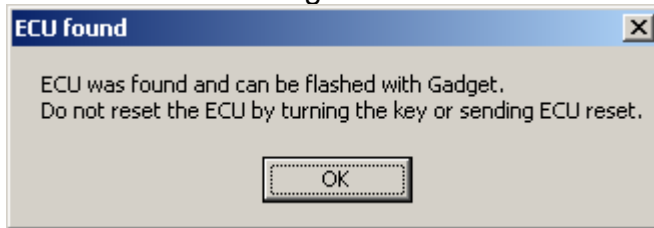


- 2) Select the defective ECU.



3) Select to recover it. XCOM will try to connect to the ECU now and should succeed within seconds. In case if XCOM fails to connect to another CAN bus directly on the ECU.

4) On success a message will be shown:



5) Immediately connect with a flash program (Gadget or DMec) and repair the ECU with a no defective software. Do not turn the key or reset the ECU in another way, in that case the whole procedure have to be redone.

GUI

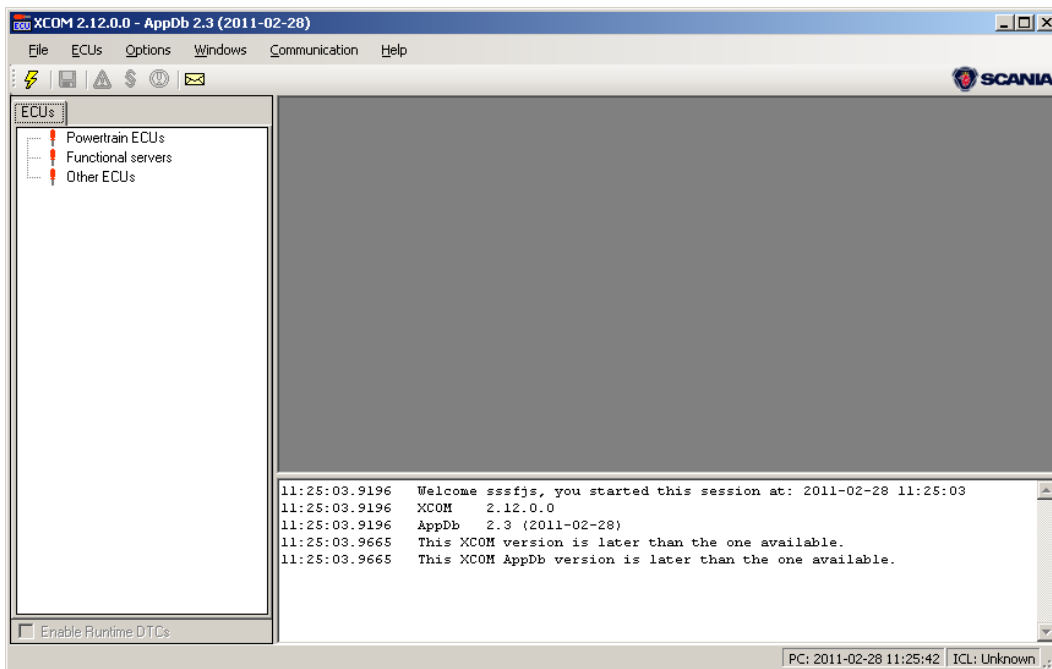
General

XCOM is a windows desktop application and tested on the following platforms:

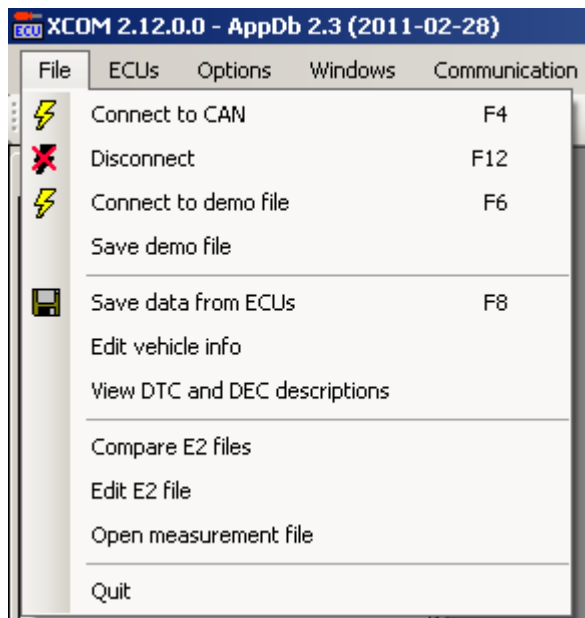
Windows 7

Windows XP

When the application is started, the following window appears:



File menu



In the file menu the following selections are available:

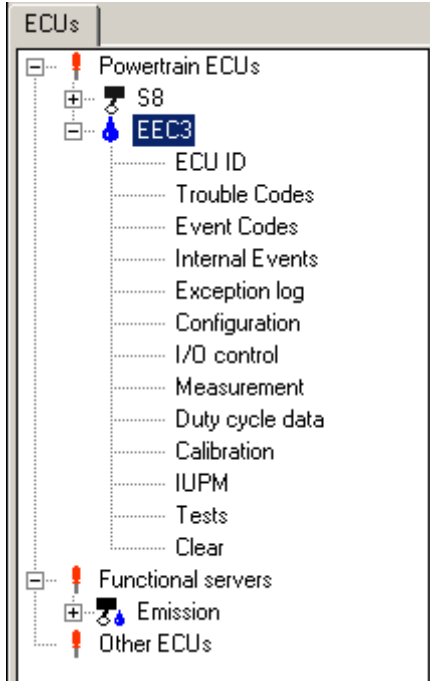
<i>Selection</i>	<i>Shortcut</i>	<i>Purpose</i>
Connect to CAN...	F4 (or shift F4)	Connect to CAN and display any found ECUs on CAN.
Disconnect	F12	Disconnects from CAN and close all forms.
Connect to demo file	F6	Connect to a previously saved SCOMM demo file.
Save demo file		Saves CAN traffic to a SCOMM demo file.
Save data from ECUs	F8	Shows the Save all data dialog
Edit vehicle info		Shows the Edit vehicle info-dialog
View DTC/DEC descriptions		Opens a window which loads all available descriptions.
Compare E2 files		Compares two E2 configuration files
Edit E2 file		Edits an E2 configuration file
Open measurement file		Opens an measurement file in a new measurement window.

ECU explorer

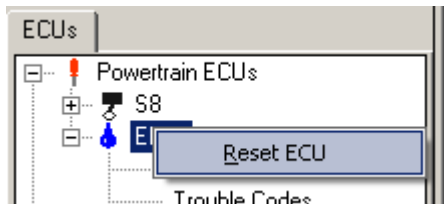
The ECU explorer holds information of what kind of data and operations are available per ECU. The information is (almost) the same as in the menu selection "ECUs". In the tree there are two root nodes, ECU nodes with their function/data node.

Approved by (department acronym name)

Issued by (department acronym name phone)
NEVE Fransson, Jonas 50 400



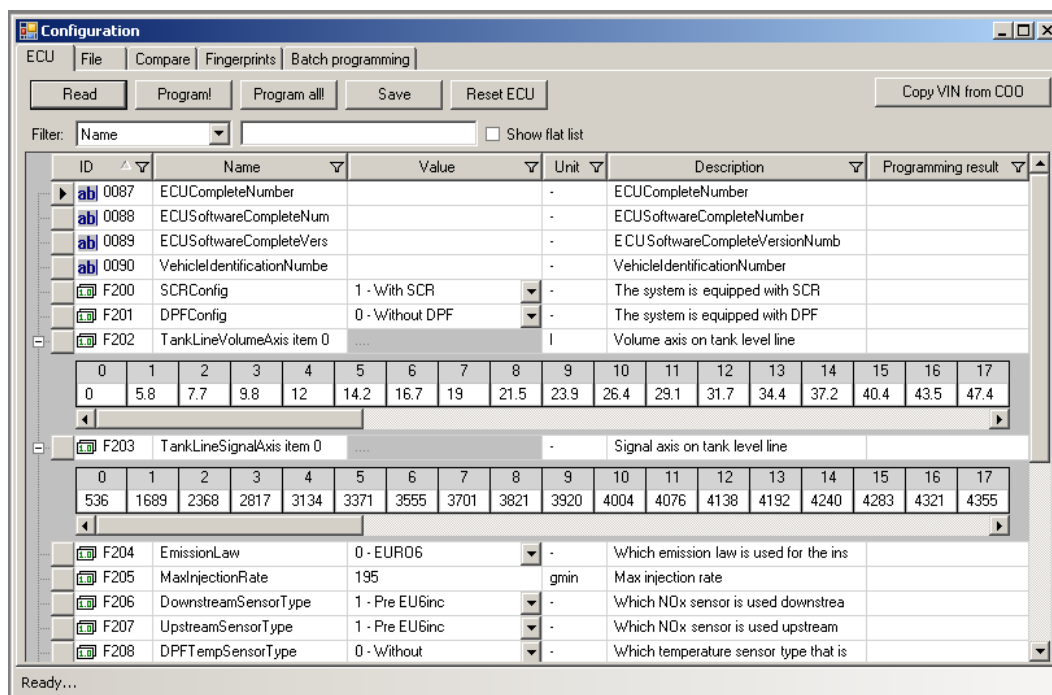
When right-clicking on an ECU node, a pop-up menu appears. See picture below.



All ECU nodes have the choice “Reset ECU”.

Configuration

The purpose of the configuration form is to modify values of the ECUs EOL parameter setup.



The screenshot shows the 'Configuration' software window with the 'ECU' tab selected. The main area displays a table of parameters with columns for ID, Name, Value, Unit, Description, and Programming result. Below the table, there are two data tables for 'TankLineSignalAxis item 0' and 'TankLineVolumeAxis item 0'.

ID	Name	Value	Unit	Description	Programming result																																				
0087	ECUCompleteNumber		-	ECUCompleteNumber																																					
0088	ECUSoftwareCompleteNum		-	ECUSoftwareCompleteNumber																																					
0089	ECUSoftwareCompleteVers		-	ECUSoftwareCompleteVersionNumb																																					
0090	VehicleIdentificationNumbe		-	VehicleIdentificationNumber																																					
F200	SCRConfig	1 - With SCR	-	The system is equipped with SCR																																					
F201	DPFConfig	0 - Without DPF	-	The system is equipped with DPF																																					
F202	TankLineVolumeAxis item 0		l	Volume axis on tank level line																																					
<table border="1"> <tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th></tr> <tr><td>0</td><td>5.8</td><td>7.7</td><td>9.8</td><td>12</td><td>14.2</td><td>16.7</td><td>19</td><td>21.5</td><td>23.9</td><td>26.4</td><td>29.1</td><td>31.7</td><td>34.4</td><td>37.2</td><td>40.4</td><td>43.5</td><td>47.4</td></tr> </table>						0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	0	5.8	7.7	9.8	12	14.2	16.7	19	21.5	23.9	26.4	29.1	31.7	34.4	37.2	40.4	43.5	47.4
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17																								
0	5.8	7.7	9.8	12	14.2	16.7	19	21.5	23.9	26.4	29.1	31.7	34.4	37.2	40.4	43.5	47.4																								
F203	TankLineSignalAxis item 0		-	Signal axis on tank level line																																					
<table border="1"> <tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th></tr> <tr><td>536</td><td>1689</td><td>2368</td><td>2817</td><td>3134</td><td>3371</td><td>3555</td><td>3701</td><td>3821</td><td>3920</td><td>4004</td><td>4076</td><td>4138</td><td>4192</td><td>4240</td><td>4283</td><td>4321</td><td>4355</td></tr> </table>						0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	536	1689	2368	2817	3134	3371	3555	3701	3821	3920	4004	4076	4138	4192	4240	4283	4321	4355
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17																								
536	1689	2368	2817	3134	3371	3555	3701	3821	3920	4004	4076	4138	4192	4240	4283	4321	4355																								
F204	EmissionLaw	0 - EURO6	-	Which emission law is used for the ins																																					
F205	MaxInjectionRate	195	gmin	Max injection rate																																					
F206	DownstreamSensorType	1 - Pre EU6inc	-	Which NOx sensor is used downstream																																					
F207	UpstreamSensorType	1 - Pre EU6inc	-	Which NOx sensor is used upstream																																					
F208	DPFTempSensorType	0 - Without	-	Which temperature sensor type that is																																					

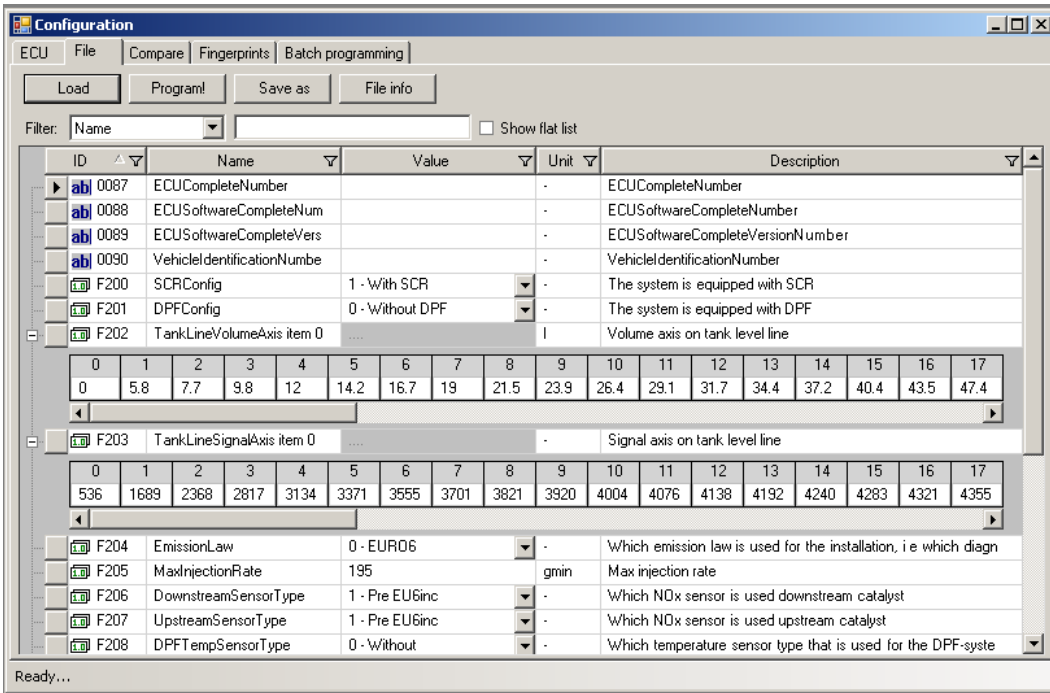
The form consists of five tab pages holding the following information:

- 1) ECU - Parameter values read from ECU.
- 2) File – Parameter values read from a file.
- 3) Diff – Compare ECU parameter values to a file.
- 4) Fingerprints – Fingerprint data read from ECU.
- 5) Batch Programming – Programming a series of parameter based on selection

In the ECU tab the user can read all parameters and their values from ECU. Once the values are read, the information can be saved to a file (xml) by pushing the "Save"-button. The user can also modify each value by selecting the value cell and type the new value or in the case where a list of pre-defined choices exists, select the preferred choice in order to change the value. The new value will be written to ECU when the user pushes the "Program!" button. To save the programmed parameters the ECU should be reset by pressing the "Reset ECU" button.

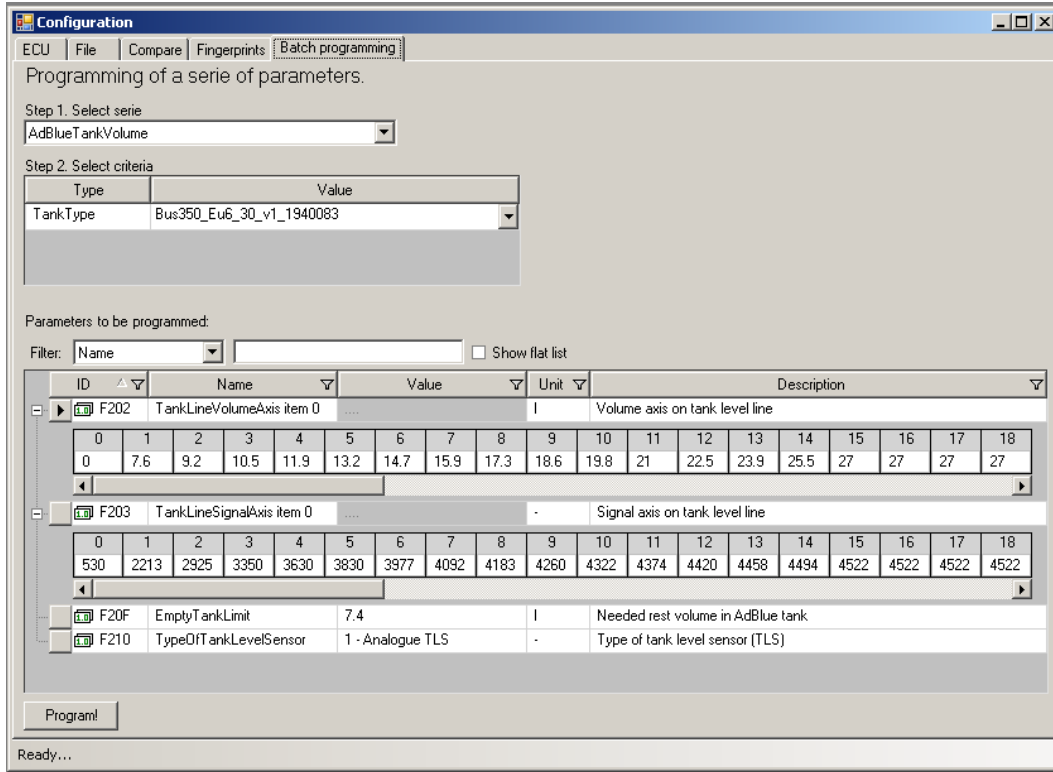
There is also a button for copying VIN from COO to the ECU the form is connected to. The user must have a COO (on CAN address 0x27) in order for the function to work.

In the File tab the user can load a previously save configuration file and program all values to the ECU. The loaded file can be edited by clicking in a value cell and entering a new value. The edited file can then be saved with the “Save as” button.



Once a file is loaded the user can program the values by pushing the “Program!” button. All values that exist in the ECU will then be programmed. If the button “Show file info” is pushed a new window will be displayed showing file information such as fingerprints and ECU ID.

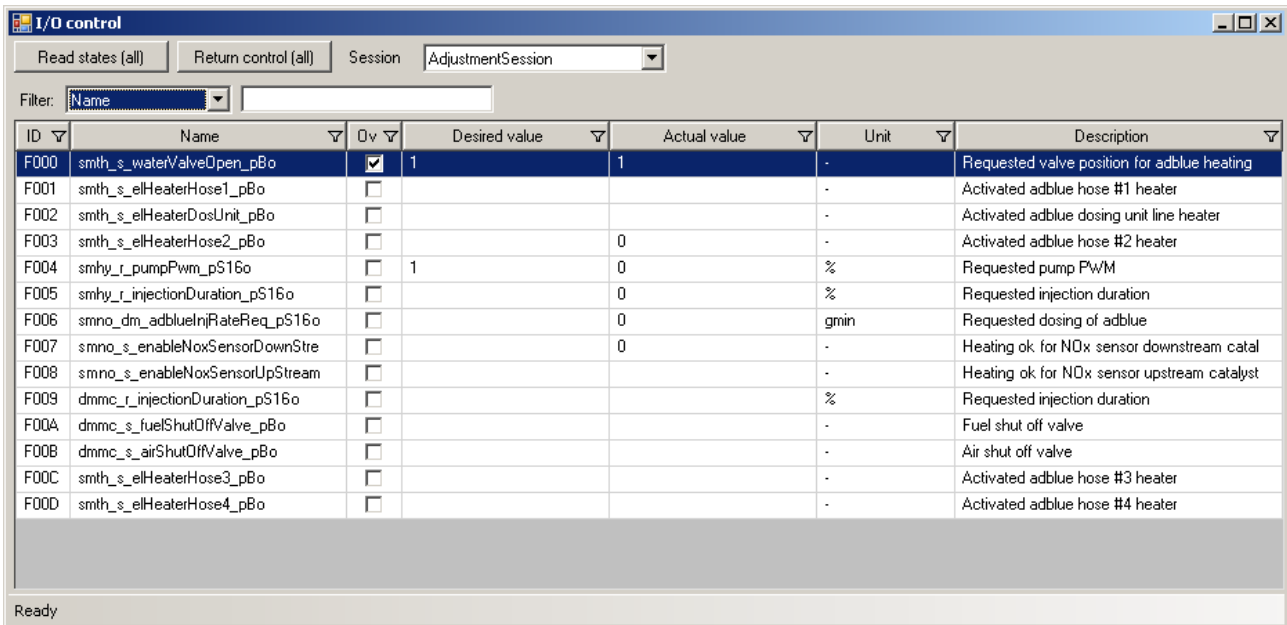
When the user wants to program special handled parameters or a series of parameters, the “batch programming”-tab can be used to set the values in ECU.



A list of a parameter series is available in the "Step 1: Select serie" combo box. Once selected a list of criterias are show in the "Select criteria" list. The "Type" holds the name of the criteria and the "Value"-column holds the criteria value. The values are selected with a drop down list. The parameters which will be changed by the batch programming are shown in the list below the selected criteria. When the criteria values are set the user pushes the "Program ECU" button to write parameter values to ECU. A message box will be displayed showing the result of the program action.

I/O control

I/O signals can be controlled by the user with the I/O control form.



ID	Name	Overridden	Desired value	Actual value	Unit	Description
F000	smth_s_waterValveOpen_pBo	<input checked="" type="checkbox"/>	1	1	-	Requested valve position for adblue heating
F001	smth_s_elHeaterHose1_pBo	<input type="checkbox"/>			-	Activated adblue hose #1 heater
F002	smth_s_elHeaterDosingUnit_pBo	<input type="checkbox"/>			-	Activated adblue dosing unit line heater
F003	smth_s_elHeaterHose2_pBo	<input type="checkbox"/>		0	-	Activated adblue hose #2 heater
F004	smhy_r_pumpPwm_pS16o	<input type="checkbox"/>	1	0	%	Requested pump PWM
F005	smhy_r_injectionDuration_pS16o	<input type="checkbox"/>		0	%	Requested injection duration
F006	smno_dm_adblueInjRateReq_pS16o	<input type="checkbox"/>		0	gmin	Requested dosing of adblue
F007	smno_s_enableNoxSensorDownStre	<input type="checkbox"/>		0	-	Heating ok for NOx sensor downstream catal
F008	smno_s_enableNoxSensorUpStream	<input type="checkbox"/>			-	Heating ok for NOx sensor upstream catalyst
F009	dmnc_r_injectionDuration_pS16o	<input type="checkbox"/>			%	Requested injection duration
F00A	dmnc_s_fuelShutOffValve_pBo	<input type="checkbox"/>			-	Fuel shut off valve
F00B	dmnc_s_airShutOffValve_pBo	<input type="checkbox"/>			-	Air shut off valve
F00C	smth_s_elHeaterHose3_pBo	<input type="checkbox"/>			-	Activated adblue hose #3 heater
F00D	smth_s_elHeaterHose4_pBo	<input type="checkbox"/>			-	Activated adblue hose #4 heater

If the ECU supports overriding signals in multiple sessions a session combo box is available. The session alternatives are listed in the "Session" combo box in the upper left corner. Once selected the signals are displayed in the main list.

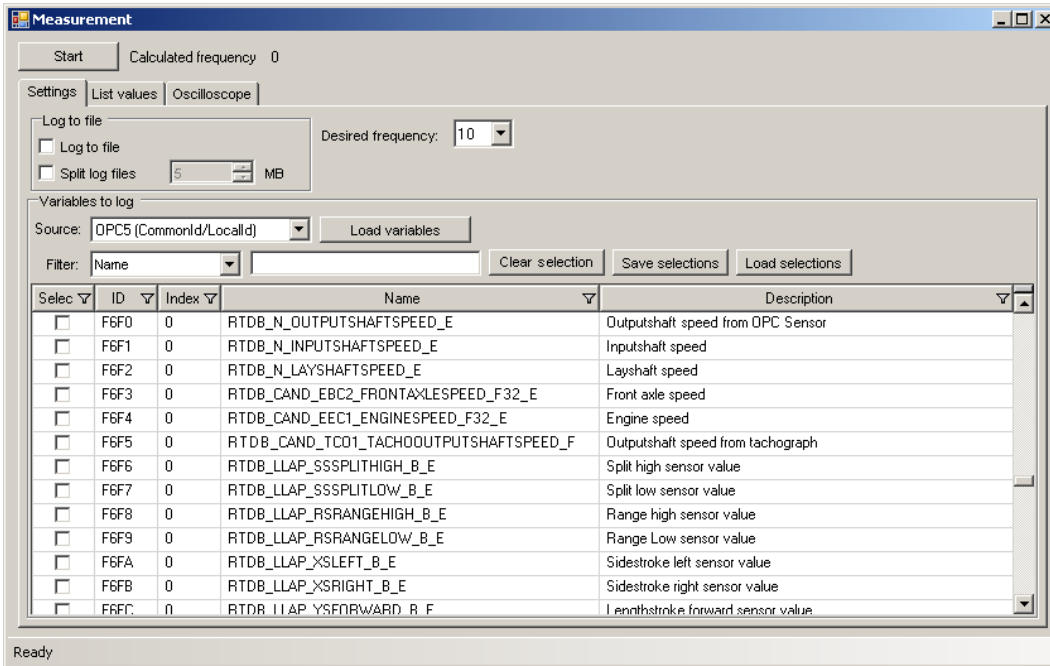
The current value of the signal is shown in the column "Actual value". To read the value either press the "read states (all)" button or right click on the signal(s) and select read.

To control a signal the user sets a value in the "desired value" and presses enter to accept it. If the ECU allowed the signal to be overridden the "overridden" checkbox will be checked. Otherwise an error message will be shown either telling that the value was out of range or it was not allowed in the current state of the ECU.

To release the override of the signal uncheck the "overridden" checkbox. To release all signals either use the "return control (all)" button or select StandardSession session.

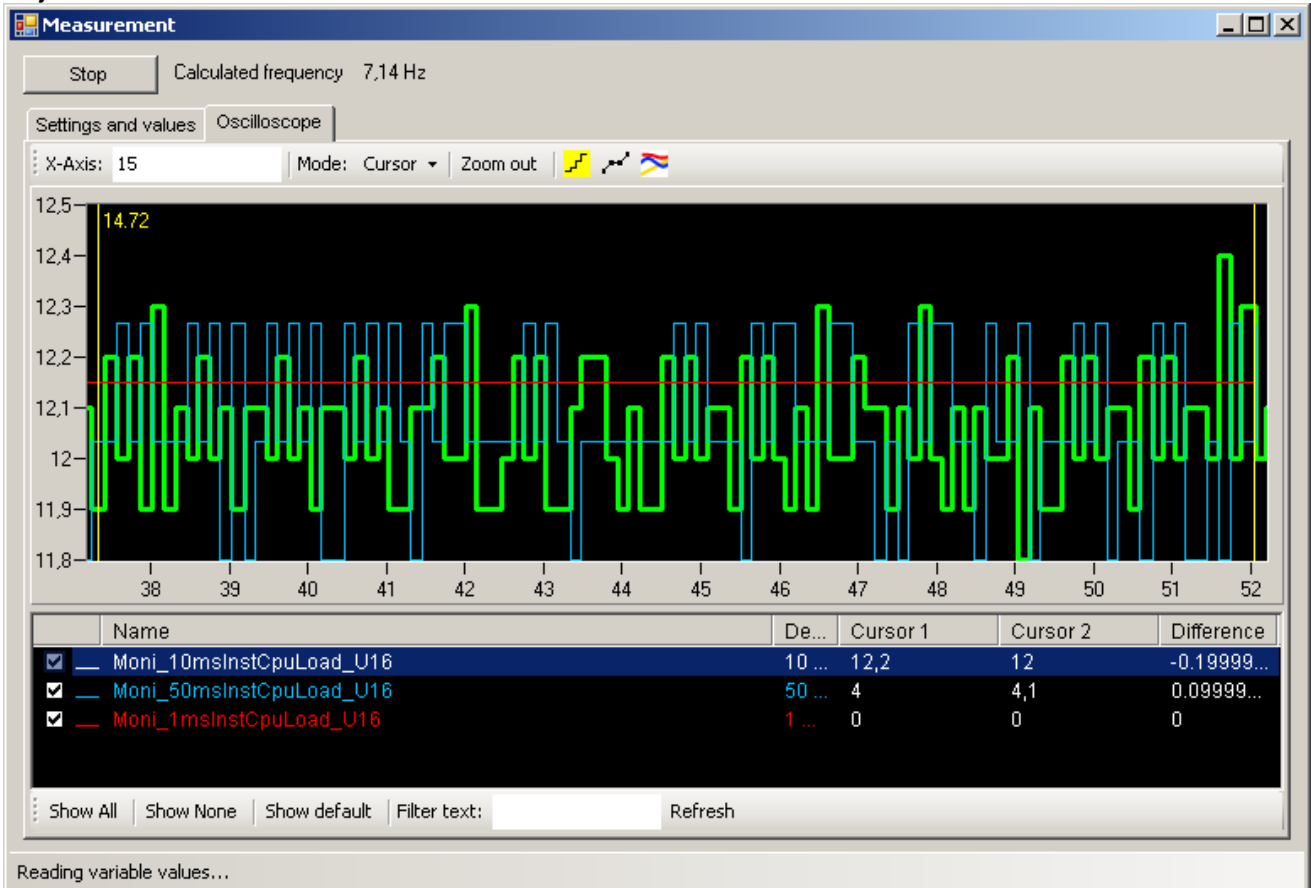
Measurement

The application can read variables continuously with the "Measurement" form which is displayed below.

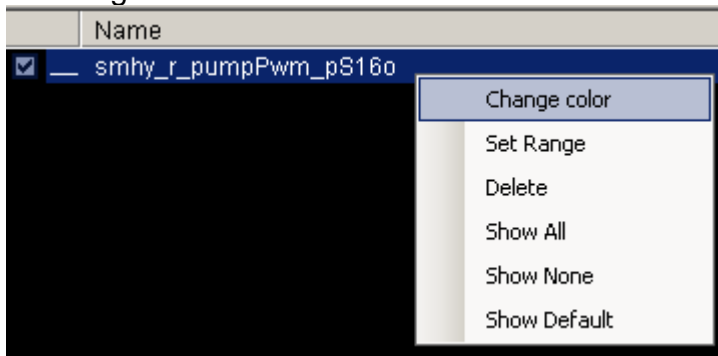


To select signals to log simply check the signals in the list. When the logging have been started the signals will be shown in the "list values" and "oscilloscope" tabs. The signals will only be saved to file if "log to file" have been checked before log started.

In the "Oscilloscope" tab, a graphical view of the data is presented. Y axis can be adjusted per variable and the length of X axis in seconds can be adjusted.



Set range of a variable:





The plot have two cursors. The cursors can be moved can be moved during logging and when the logging is stopped.

When the logging is stopped the pan and zoom modes can be used. To reset the view of the plot click on the “Zoom out” button.

Variables logged to a measurement file can be loaded with the “Load measurement” button. All modes will be enabled when viewing a loaded file.

Short cuts

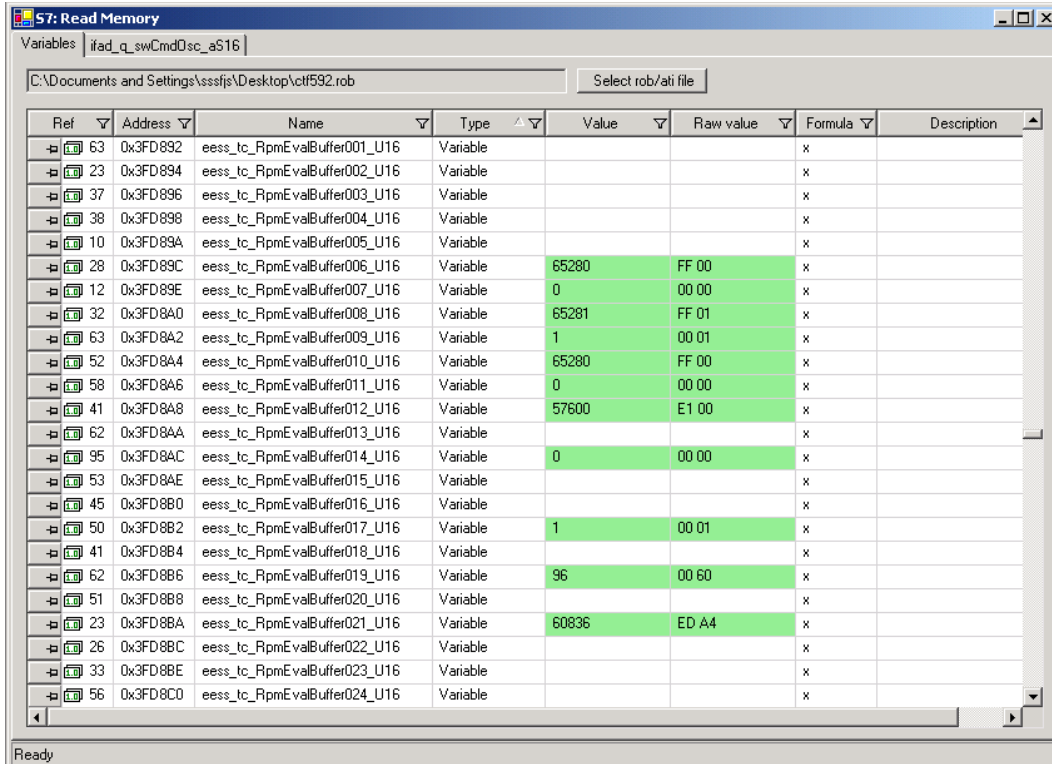
- Panning:
 - o Press the control key to be able to pan the graph horizontally. When the control key is released the pointer mode goes back to what it was before.
- Cursors:
 - o Double click on graph to set first cursor
 - o Shift key and double click to set second cursor
 - o The cursors can also be moved using the arrow keys left and right
 - o Move the entire selection and up/down to increase/decrease the span, while the graph component is focused.
- Zooming:
 - o Use + to zoom between the cursors and – to zoom out to the last position when + was hit or show the entire graph.
 - o Press control key and + or – to zoom in or out the entire graph.

Read/Write Memory

Some ECUs have the kwp-service ReadByMemoryAddress activated and have the Read Memory window in XCOM. To read from the memory a ROB, ATI or A2L definition file is required. When the file is loaded all variable definitions from that file is shown in a list.

XCOM will read the selected variable from the ECU if it is clicked. A green background means that it was successfully read, and a red background usually means that ReadByMemoryAddress service can't access that memory area. The value is shown both as scaled value and as the raw bytes.

Variables can be edited if the ECU supports WriteByMemoryAddress. To edit a variable click in the value or raw value cell and enter the new value.



The screenshot shows the '57: Read Memory' window in XCOM. The 'Variables' tab is active, showing a list of variables from a file located at 'C:\Documents and Settings\sjsfs\Desktop\ctf592.rob'. The table below represents the data shown in the screenshot.

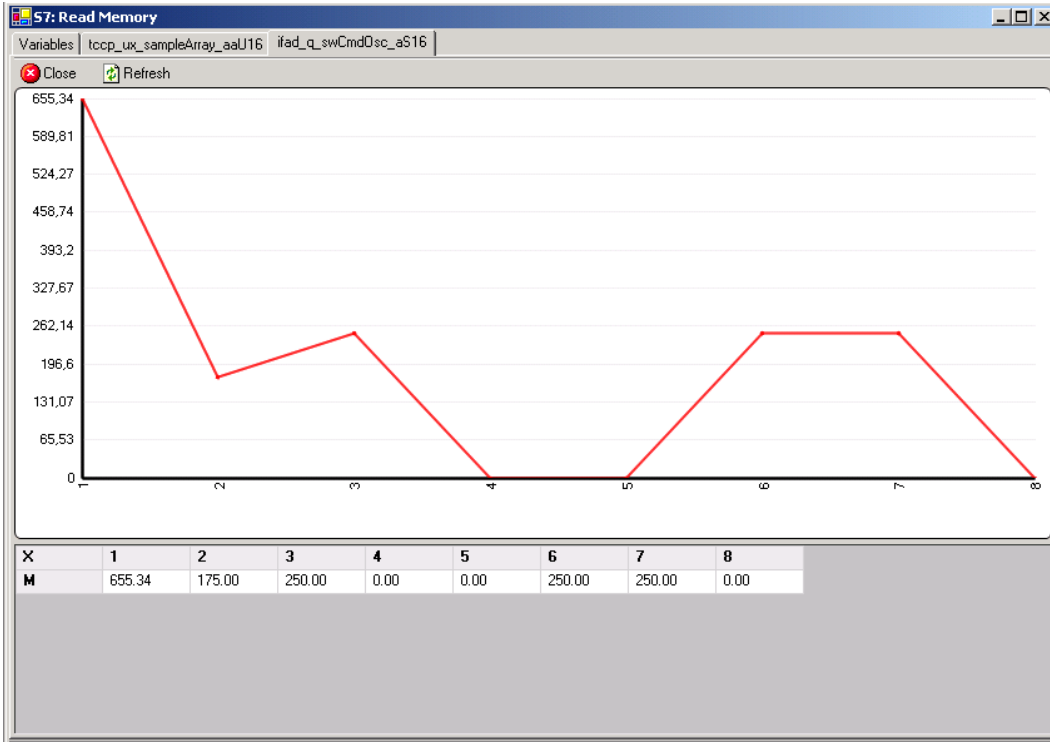
Ref	Address	Name	Type	Value	Raw value	Formula	Description
63	0x3FD892	eess_tc_RpmEvalBuffer001_U16	Variable			x	
23	0x3FD894	eess_tc_RpmEvalBuffer002_U16	Variable			x	
37	0x3FD896	eess_tc_RpmEvalBuffer003_U16	Variable			x	
38	0x3FD898	eess_tc_RpmEvalBuffer004_U16	Variable			x	
10	0x3FD89A	eess_tc_RpmEvalBuffer005_U16	Variable			x	
28	0x3FD89C	eess_tc_RpmEvalBuffer006_U16	Variable	65280	FF 00	x	
12	0x3FD89E	eess_tc_RpmEvalBuffer007_U16	Variable	0	00 00	x	
32	0x3FD8A0	eess_tc_RpmEvalBuffer008_U16	Variable	65281	FF 01	x	
63	0x3FD8A2	eess_tc_RpmEvalBuffer009_U16	Variable	1	00 01	x	
52	0x3FD8A4	eess_tc_RpmEvalBuffer010_U16	Variable	65280	FF 00	x	
58	0x3FD8A6	eess_tc_RpmEvalBuffer011_U16	Variable	0	00 00	x	
41	0x3FD8A8	eess_tc_RpmEvalBuffer012_U16	Variable	57600	E1 00	x	
62	0x3FD8AA	eess_tc_RpmEvalBuffer013_U16	Variable			x	
95	0x3FD8AC	eess_tc_RpmEvalBuffer014_U16	Variable	0	00 00	x	
53	0x3FD8AE	eess_tc_RpmEvalBuffer015_U16	Variable			x	
45	0x3FD8B0	eess_tc_RpmEvalBuffer016_U16	Variable			x	
50	0x3FD8B2	eess_tc_RpmEvalBuffer017_U16	Variable	1	00 01	x	
41	0x3FD8B4	eess_tc_RpmEvalBuffer018_U16	Variable			x	
62	0x3FD8B6	eess_tc_RpmEvalBuffer019_U16	Variable	96	00 60	x	
51	0x3FD8B8	eess_tc_RpmEvalBuffer020_U16	Variable			x	
23	0x3FD8BA	eess_tc_RpmEvalBuffer021_U16	Variable	60836	ED A4	x	
26	0x3FD8BC	eess_tc_RpmEvalBuffer022_U16	Variable			x	
33	0x3FD8BE	eess_tc_RpmEvalBuffer023_U16	Variable			x	
56	0x3FD8C0	eess_tc_RpmEvalBuffer024_U16	Variable			x	

When an adaptive line or map is clicked they will be shown in a new tab page. That tab page consists of two parts, a graph and a table with the values.

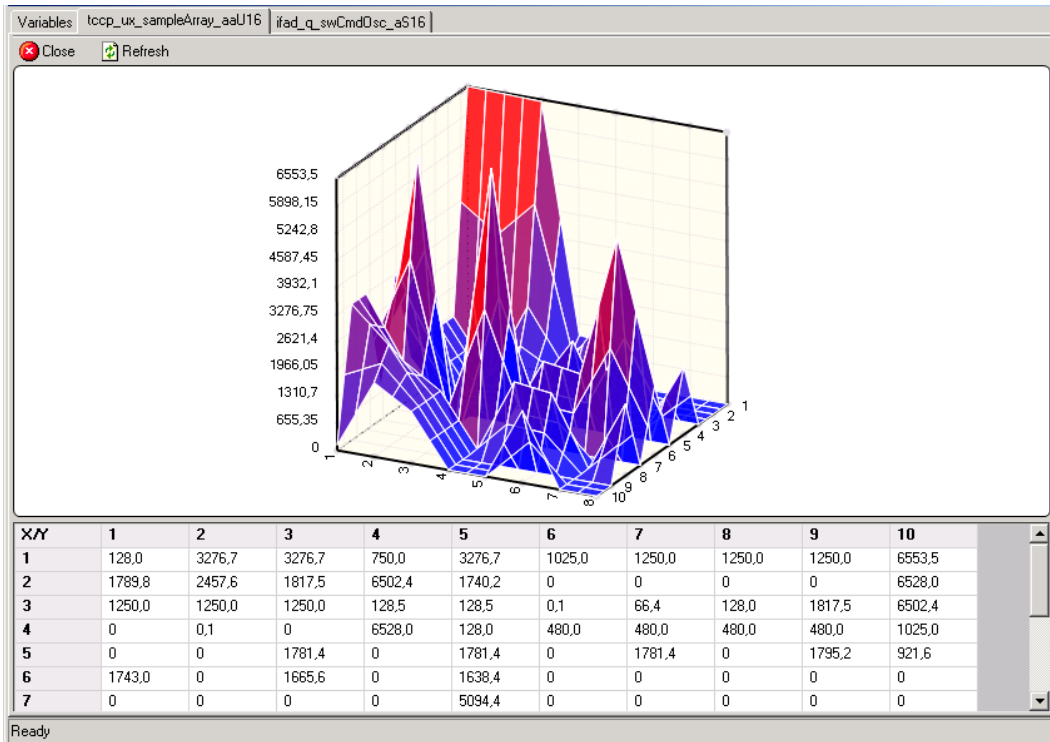
Approved by (department acronym name)

Issued by (department acronym name phone)
NEVE Fransson, Jonas 50 400

Info class
Internal
 Page
28(33)

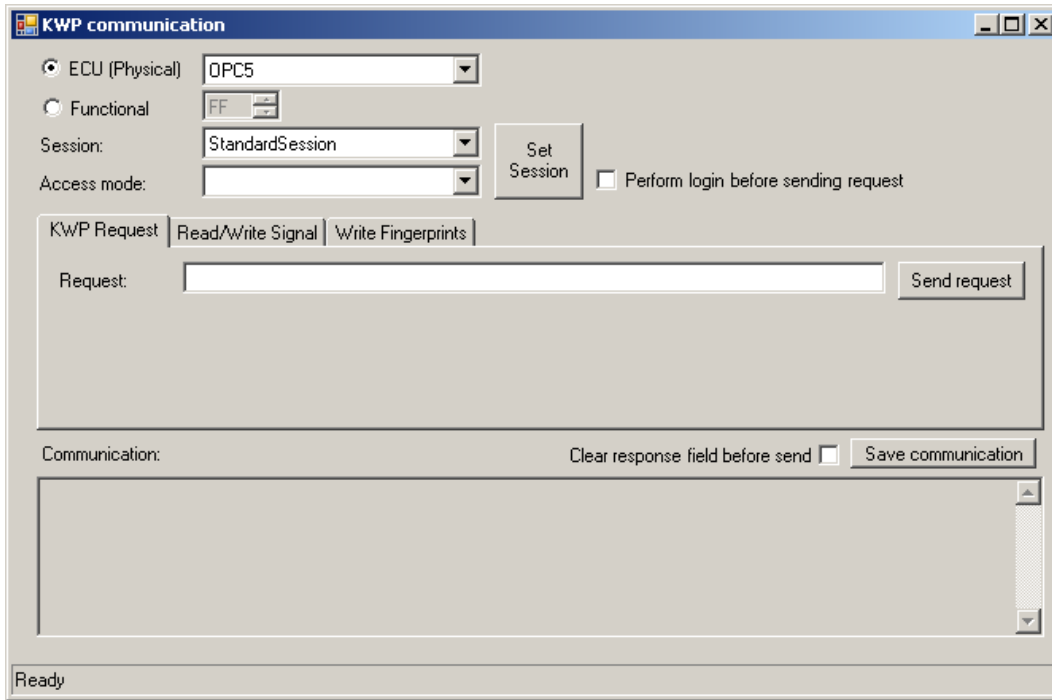


The graph of an adaptive map can be rotated by with ALT+left mouse button and it can be zoomed in or out with ALT+mouse scroll.



KWP signalling

The application allows the user to send specific KWP messages to an ECU and displays the response with hexadecimal byte numbers. This is done with the KWP communication form.



The form is activated with the menu choice Communication->KWP signalling or by pressing Ctrl+K.

Both physical and functional requests are supported.

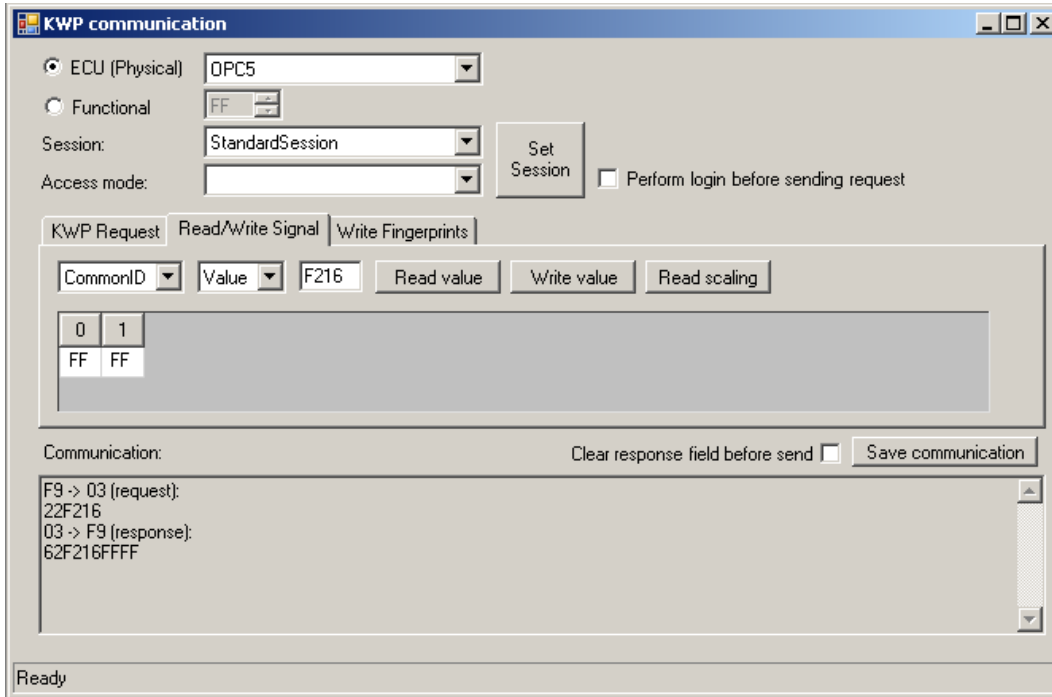
Select under which session the KWP command is going to be executed in by selecting a choice in the "Session" combo box.

Select under which security access mode the KWP command is to be executed in by selecting a choice in the "Access mode" combo box.

Enter a hexadecimal byte sequence in the "Request" text field. Each byte can be separated from the next by a space for greater visibility. The "Request" text field will autosuggest recently sent requests and common KWP services.

Send the request by pressing the "Send request" button.

If only the set session and security access mode commands shall be tested, press the "Do now" button (which covers the height of the session and access mode combo boxes).



To simplify reading and writing to a signal a KWP-message can be generated by specifying what kind of signal it is and what ID it has. When a signal is read all value bytes will be shown in the grid. The bytes can be edited and if the right session is used they can be written to the ECU.

This tab also supports reading of signal scaling. A KWP-message will be generated and both raw bytes and parsed scaling will be presented to the user.

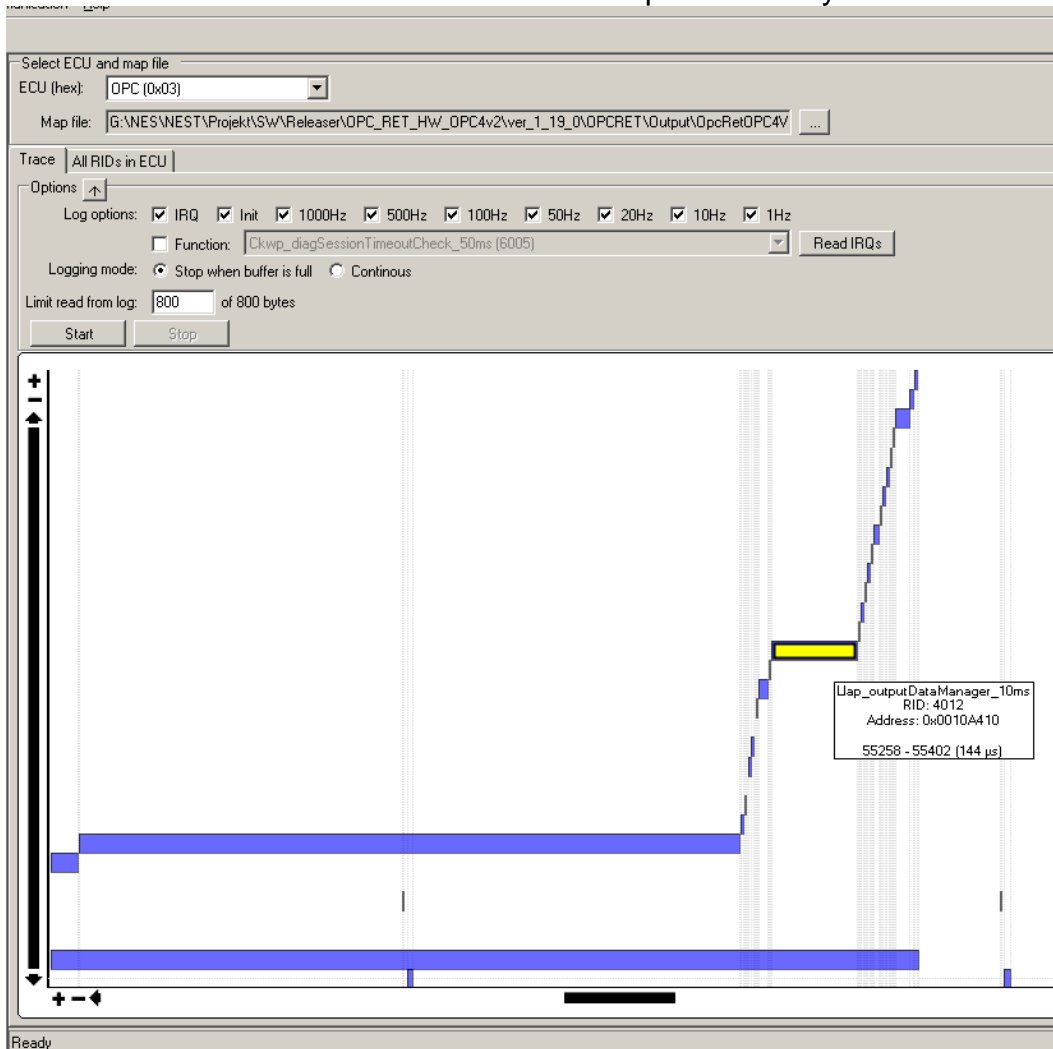
Trace

The trace window gives the user access to the timing analysis services implemented by the ECU.

The trace can be done in two modes:

- Log all functions of the selected types (10Hz, IRQ, ...)
- Log only a selected function

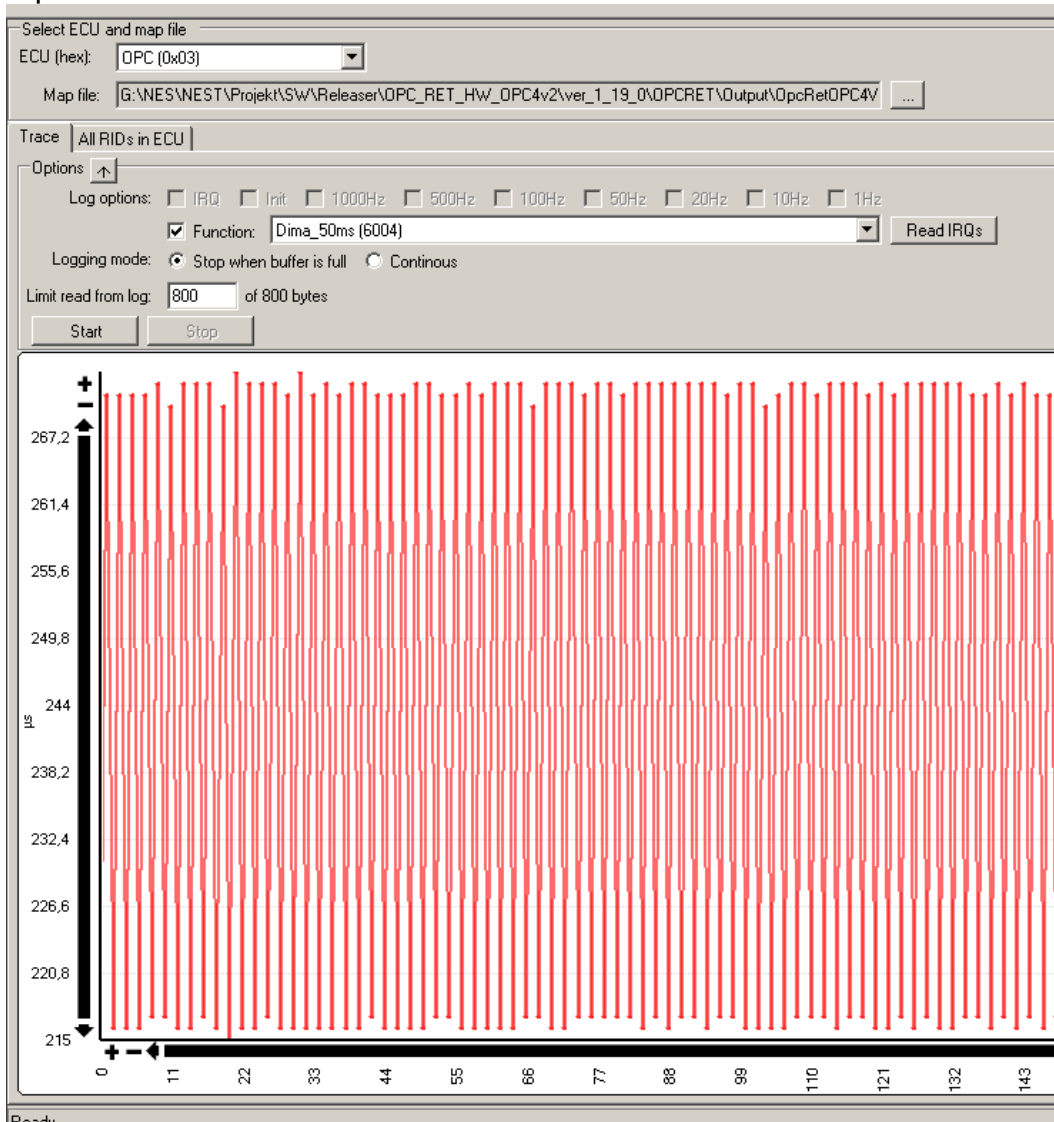
The first mode will result in a gantt diagram where the user can control execution times and that functions are interrupted correctly.



Approved by (department acronym name)

Issued by (department acronym name phone)
NEVE Fransson, Jonas 50 400Info class
Internal
Page
32(33)

The second type of logging results in a graph which reports how much time it took to execute the function. Each sample is the execution time in microseconds. The minimum, maximum, average, and total time is reported in the status window.





Approved by (department acronym name)

Date
2012-10-10

Info class
Internal

Issued by (department acronym name phone)
NEVE Fransson, Jonas 50 400

Issue

Page
33(33)

The trace window also support listing of all available functions which have a Runnable ID defined (and can be logged).

Trace All RIDs in ECU

Runnable IDs to read: IRQ Init 1000Hz 500Hz 100Hz 50Hz 20Hz 10Hz 1Hz Background

Read

Drag a column header here to group by that column.

RID	Group	Address	Function
1000	Init	0x0006256C	File_init
1001	Init	0x0005FC60	Main_canbusHw_init
1002	Init	0x00063684	Kwpm_kwpmManager_init
1003	Init	0x0010E38C	Cpplue_init
1004	Init	0x000604B4	Init_system_init
1005	Init	0x00109A58	Llap_init
1006	Init	0x00064D0C	Dtcm_setupGbox_init
1007	Init	0x00065000	Dtcm_setupRetx_init
1008	Init	0x0005FCDC	Main_setupInteEventRecords_init
4000	Log100Hz	0x00062834	File_10ms
4001	Log100Hz	0x00109CB8	Llap_inputDataManager_10ms
4002	Log100Hz	0x000606F0	Thdl_commonPreMainApp_10ms