## **CAN Data Miner**



Software to access vehicle information via the OBD-II connector.

# Contents

CAN Data Miner	1
Software to access vehicle information via the OBD-II connector	1
Introduction	3
Why	
Overview	
Disclaimer	
Protocols & Features	
Known Bugs/Limitations	
Installation	5
Execution	6
Configuration	9
Config Parameters	
Communication Speeds	. 10
Units	. 11
PID Groups	. 11
PID Group	. 11
Item ID.	. 12
Data	. 12
Modes	. 14
Vehicles	. 15
Vehicle Table	. 15
General Tab	. 15
ECU and PID Groups Tab	. 16
Addresses Tab	
Flags and Parameters Tab	. 17
Connecting	
Interface	. 19
Smart Serial Interface	. 21
Protocol	
CAN Logging	. 24
OBD-II	
Current Data	. 27
Chart data	. 28
Snapshot Data	. 29
Howtos	. 30
Data Log	. 30
Supported Hardware Reference	. 32
Unsupported Devices	. 33

## Introduction

### Why

Why make just another tool for the CAN-bus when there are so many tools available already? I have two reasons;

- 1. Learning how it really works, it's useful as an added experience.
- 2. Because I can.

What I want to provide to the world is a flexible and configurable tool. This will of course also mean that it may be complicated for the average person to understand every detail.

Also notice that for J2534 devices there is no automatic detection of protocol. This is one of the current limitations of the application.

This tool is still under development so this manual may not be accurate at all times.

#### Overview

At installation you will get a set of default values in the tool, and the vehicle that I have tested on is a Subaru, which means that some parts are specific for that brand and that if you test on another vehicle you may need to change the configuration for that vehicle.

In the document there are references to PID, it may in some cases be seen as "PID or other item", e.g. address, TID etc.

Information that is obvious is intentionally left out - like explanation of Name fields.

#### Disclaimer

Data provided together with this application is provided as-is and may not be entirely correct. Especially when it comes to data that is specific for a vehicle brand or model.

### **Protocols & Features**

Supported protocols are: J1850 VPW & PWM, ISO9141, ISO14230, ISO 15765 (11 and 29 bit) and raw CAN. Notice that the actual support also depends on the interface device.

The application supports a number of OBD adapters, some J2534, some serial with AT-commands. The first page of the application displays the supported adapters.

The current version will look up J2534 adapters from the registry so even if your adapter isn't listed it may still work. Notice that the adapter must support the 04.04 version of the J2534 API at the moment. The older version is not supported. If you don't know which version your adapter support you can always try and see if it is discovered or not.

One of the features of this application is that it supports multiple adapters at the same time so you can connect two vehicles at the same time and compare data from them as long as you use different adapters. Multiple adapters of the same model is not supported. Just watch out for stray currents if you connect two vehicles at the same time and you for example use battery chargers on them or if there is another connection between them.

#### Known Bugs/Limitations

- The Tactrix adapter seems to not have complete support for 29 bit addressing.
- The ISO 9141 support on the Volvo UWA seems to be incomplete.

## Installation

This application shall be distributed in a ZIP file where the MSI file, this document and a Setup.exe file shall be present. As you read this you have probably already unpacked the ZIP file into an empty directory.

If you have an earlier version of this program installed I recommend you to uninstall that version first.

Execute the Setup.exe file and install the program using the default values.

If you have a normal installation you it will be installed at the following locations:

Windows 7 64-bit (x64 version)

Executable	C:\Program Files (x86)\Bedug.com\CAN Data Miner
Data	C:\ProgramData\Bedug.com\CAN Data Miner

Windows 7 32-bit (x86 version)

Executable	C:\Program Files\Bedug.com\CAN Data Miner
Data	C:\ProgramData\Bedug.com\CAN Data Miner

#### Windows XP

Executable	C:\Program Files\Bedug.com\CAN Data Miner
Data	C:\Documents and Settings\All Users\Application
	Data\Bedug.com\CAN Data Miner

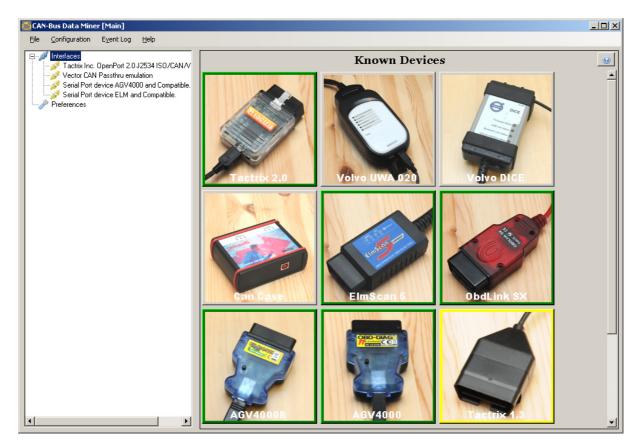
Most of the testing has been done under 64-bit Windows 7, but the application is built for 32-bit and should work well there too.

You should try to find the data directory, since the files there will be updated if you change the configuration. And if you uninstall the program the data files will be removed. So it may be a good idea to copy the files to a safe location before you update the application. Notice that a new version of the application may not be compatible with the old files, even if I'm trying to avoid changes that are incompatible.

If all goes well with the installation you will find the program in the start menu under **Bedug.com**.

# Execution

This program can be used without an adapter, but you will in that case not be able to communicate with the vehicle for obvious reasons. There is a simulated driver alternative that may be used if you like to try it out without connecting it to a car. However the simulated driver is a bit limited in functionality.

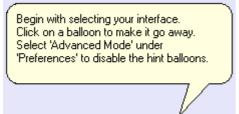


At the first page there is a tree view to the left with the interfaces and the preferences branches. The first page displayed contains a display of the supported devices and some devices that aren't supported. The items will get a border that indicates the level of support provided as listed in the following table:

Color	Support
Green	Supported and recognized. Notice that serial devices are set as recognized as soon as the system has a serial port so it's a bit of false advertising here.
Gray	Supported but not recognized.
<b>Yellow</b>	Currently not supported, will be added in future version.
Red	Won't be supported.

If you have a J2534 adapter that the application supports it must be connected and the drivers for it loaded before the application is started.

If you don't have "Advanced Mode" selected in the preferences page you will get a lot of hint balloons with tips on various items:



To hide a single one for the moment you can click on it. But it will reappear when you move the window to which it's pointing.

The File menu will give you the option to open and close log files of various kinds.

- CAN Data log file this is a file written in the Vector Canalyzer ASC format and is only available when using the CAN protocol. The ASC format is a text format.
- Event log file this is a file that gets the same information as is written in the log window.
- Measurement data log file this file is in the Microsoft Excel CSV format and contains sampled data from the vehicle. One column per data source, data sources are selected before logging is started.

Event logging is done in a separate window to allow for some flexibility. It also contains some information about open log files.

The **Event Log** menu alternative will bring forward the log window in case it has been hidden.

Event Log CAN Traffic	
Events (limited to last 10000 characters)	Clear Loglevel Info
CAN Data Miner	
Standards: - SAE J2534 - SAE J1979	
000 000,024: Data Directory: C:\ProgramData\Bedug.com\CAN Data Miner\AppData' 000 000,042: Loading Preferences. 000 000,177: 32 Preferences Loaded from XML file. 000 000,177: this.gpsSerialPortName='COM19' 000 000,635: Port='COM19' 000 000,641: Loading Commspeeds. 000 000,641: Loading Commspeeds. 000 000,641: g Commspeeds Loaded from XML file.	λ
000 000,742: Loading Units. 000 000,833: sr=System.I0.StreamReader 000 000,838: 49 Units Loaded from XML file. 000 000,838: Loading Parameters. 000 000,942: 35 Parameters Loaded from XML file. 000 000,943: Loading vehicles. 000 001,063: 8 vehicles Loaded from XML file.	
000 001,063: Vehicle 0: Generic 11-bit, bits=11, numEcus=0	
CAN Data Log File None *	
Event Log File  * None *	
Measurement Data Log File   * None *	

In the event log window there is also a second tab that presents the CAN bus traffic. Currently this will only be presented when the Vector CanCase is used.

Log CAN Traffic															
Timestamp	Dir.	Address		Pa	yloa	Ы									Clear
458,3602	In2	0x07E8		41	20	FF	FF	FF	FF						
458,3602	Out2	0x07E0		01	40										
458,3822	Out1	0x07E0		02	01	40	00	00	00	00	00				
458,4052	Inl	0x07E8		06	41	40	FF	FF	FF	FF	00,	flags=0x00,	resla		
458,4052	In2	0x07E8		41	40	FF	FF	FF	FF						
458,4302	Out2	0x07E0		01	60										
458,4572	Outl	0x07E0		02	01	60	00	00	00	00	00				
458,4892	Inl	0x07E8		06	41	60	FF	FF	FF	FF	00,	flags=0x00,	resl		
458,5332	In2	0x07E8		41	60	FF	FF	FF	FF						
458,5652	Out2	0x07E0		01	80										
458,5652	Outl	0x07E0		02	01	80	00	00	00	00	00				
458,6022	Inl	0x07E8	Ш	06	41	80	FF	FF	Fl	91	00,	flags=0x00,	resl		
458,6452	In2	0x07E8		41	80	FF	FF	Fl	91						
458,6902	Out2	0x07E0		01	AO										
458,7342	Out1	0x07E0	Ш	02	01	AO	00	00	00	00	00				
458,7832	Inl	0x07E8		06	41	AO	00	00	00	01	00,	flags=0x00,	resla		
458,8302	In2	0x07E8	Ш	41	AO	00	00	00	01						
458,8732	Out2	0x07E0	Ш	01	CO										
458,8742	Out1	0x07E0		-			00								
458,9232	Inl	0x07E8		06	41	CO	00	00	00	01	00,	flags=0x00,	resla		
458,9743		0x07E8					00	00	01				_		
459,0253		0x07E0			EO										
459,0263	Out1	0x07E0		-			00								
459,0743	Inl	0x07E8		_						00	00,	flags=0x00,	resla		
459,0753	In2	0x07E8		41	EO	00	00	00	00					-	

This can be useful when debugging traffic on the CAN bus. The "Dir" column is displaying the direction of data and which communication layer. There are three possible layers, 1 to 3 where 1 is the low level CAN and 3 is the top level.

# Configuration

Under this menu item there are several alternatives, and each of them will be described.

For simplicity I have tried to use the Windows controls in their default manner, and I do a quick intro here:

	Name	Speed	Default J1850 (only check one)	Default ISO9141 (only check one)	Default ISO15765 (only check one)	Valid J1850 (Check one or more)	Valid ISO9141 (Check one or more)	Valid ISO15765 (Check one or more)	Save Close
•	1.2 kbps	1200							
	4.8 kbps	4800							
	9.6 kbps	9600							
	10.4 kbps	10400	V	~			~		
	41.6 kbps	41600							
	125 kbps	125000							
	250 kbps	250000							
	500 kbps	500000			~				
	1 Mbps.	1000000							
*									

To add a record, select the last line in the table and start to type.

In a few cases you will need to double-click on the empty row to add data in a separate form instead.

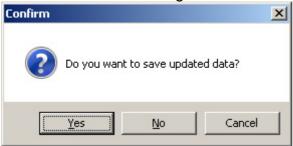
To update data you will select the field you want to update and click on it with the mouse to get it into edit mode.

(miled)

	Name	Speed	Default J1850 (only check one)	Default ISO9141 (only check one)	Default ISO15765 (only check one)	Valid J1850 (Check one or more)	Valid ISO9141 (Check one or more)	Valid ISO15765 (Check one or more)	Save Close
	1.2 kbps	1200							
	4.8 kbps	4800							
	9.6 kbps	9600							
	10.4 kbps	10400	~	~			~		
	41.6 kbps	41600							
	125 kbps	125000							
•	250 kbps	250000							
	500 kbps	500000			~				
	1 Mbps.	1000000							
*									

To delete data you need to select the whole row and press the **Del** button on the keyboard. A confirmation pop-up shall appear to prevent you from accidental deletes.

If you have updated data you will get a confirmation pop-up before any changes are lost. The application is a bit sensitive, so it's sufficient to put a field into edit mode to make it think that a change is made.



### **Config Parameters**

This is a list of the IOCTL configuration parameters that the J2534 API accepts. This list shall not need any change unless there is an update to the API. The current version is the 04.04 version. Later versions may have additional parameters.

ID	Name	Description	Pres. Type	Min value	Max value	Protocols	Secured	▲ Save
0x01	DATA_RATE	TBD	Deci 💌	5	1000000			Close
0x03	LOOPBACK	TBD	Hex 💌	0×00	0x01		V	
0x04	NODE_ADDRESS	TBD	Hex 💌	0×00	Oxff		V	
0x05	NETWORK_LINE	TBD	Hex 💌	0×00	0x02		V	
0x06	P1_MIN	TBD	Hex 💌	0×00	Oxffff		V	
0x07	P1_MAX	TBD	Hex 💌	0×00	Oxffff		V	
0x08	P2_MIN	TBD	Hex 💌	0×00	Oxffff		V	
0x09	P2_MAX	TBD	Hex 💌	0×00	Oxffff		V	
0x0a	P3_MIN	TBD	Hex 💌	0×00	Oxffff		V	
0x0b	P3_MAX	TBD	Hex 💌	0×00	Oxffff		V	
0x0c	P4_MIN	TBD	Hex 💌	0×00	Oxffff		V	
0x0d	P4_MAX	TBD	Hex 💌	0×00	Oxffff		<b>v</b>	
0x0e	W1	TBD	Hex 💌	0×00	Oxffff		~	
0x0f	W2	TBD	Hex 💌	0×00	Oxffff		~	-

## **Communication Speeds**

This page allows you to configure the permitted speeds that can be used with the interface. I haven't tested all the listed speeds, and unless you want to add a speed to suit your case you can leave this as is.

	Name	Speed	Default J1850 (only check one)	Default ISO9141 (only check one)	Default ISO15765 (only check one)	Valid J1850 (Check one or more)	Valid ISO9141 (Check one or more)	Valid ISO15765 (Check one or more)	Save Close
•	1.2 kbps	1200					V		
	4.8 kbps	4800					V		
	9.6 kbps	9600					~		
	10.4 kbps	10400	V	~		•	~		
	41.6 kbps	41600				•	~		
	125 kbps	125000							
	250 kbps	250000							
	500 kbps	500000			~				
	1 Mbps.	1000000						•	
*									

### Units

This page allows you to configure units of various types to be used during presentation, and you can also categorize them. This is intended to allow for filtering between Metric and Imperial units at presentation.

	Unit	Description	Category	<b>_</b>	Save
•		When the mind goes empty.	0 = General 💽		Chara
	%	Percent.	0 = General 💽		Close
	*	Degrees (angle)	0 = General 💽		
	°C	Temperature in Celsius	1 = Metric		
	۴F	Temperature in Fahrenheit.	2 = Imperial		
	A	Amperes.	0 = General 💽		
	bit	When a single bit is used.	0 = General 💽		
	Bitmap	Separate interpretation needed for presentation o	0 = General 💽		
	bits	When a few bits are used for a value.	0 = General 🗖		
	DTC	Diagnostic Trouble Code, special interpretation.	0 = General 💌		

There is a safeguard built in to prevent you from deleting a unit that is used.

## **PID Groups**

The PID group page is one of the key pages in the application. There are several items of information that can be configured here.

ID Gro	oup										Iten	n ID (PID	, TID, Addre	ss etc.)											
	Group ID	Name							Has PID Map	Prim. Mode		ID	em N	ame			Ca Ple	an Pl ot H	ot eight	ls PID Map	Refresh		Ê	Sav	
•	1	SAE PIDs			•	0x01		0x <sup>*</sup>	16 Ba	ank 1, Se	nsor 3: C	xygen se	e 🖡	7 15	0		1:1	•			-				
	2	Extended	IPIDs.							0x22		0x	17 Ba	ank 1, Se	nsor 4: C	xygen se	e F	15	0		1:1		-		
	5	Tests (Or	nly Non-C	AN .	vehicles)				~	0x05		0x	18 Ba	ank 2, Se	nsor 1: C	xygen se	e 🖡	7 15	0		1:1	•			
	6	Tests (C4	N + Nor	h-CAN	V vehicles]	Î.			V	0x05		0x	19 Ba	ank 2, Se	nsor 2: C	xygen se	e 🖡	15	0		1:1				
	7	OBDMID	S						•	0x06		0x	1a Ba	ank 2, Se	nsor 3: C	lxygen se	e 🖡	15	0		1:1				
	8	Vehicle I	nformatio	n					•	0x09	_	0x	16 Ва	ank 2, Se	nsor 4: C	lxygen se	e F	15	0		1:1	•	- I		
	10	SSM Dat	ata		M Data 🗖 0xa8		SM Data 🗆 Oxa		0xa8			0x	1c 01	3D stand	ards this	vehicle o	:o [	15	0		1:1		·		
	11	Subaru E	xtended	PIDs	1					0x22		0x	1d 0:	kygen sei	nsors pre	sent	Γ	15	0		1:1	•			
ata	12	ReadDI	C Inform	ation	/ISO 1422	90				0v19	<u> </u>	0	1e Å:	wiliaru im	uit etatue		r	1 15			1.1		. <b>I –</b> I	Ren	umber
	#	Name	Unit		Address Offset [Hex]	Byte Count	Starl Bit	End Bit	Scale Offset	Scale Factor	Pres. Min	Pres. Max	Format	Red Zone Min	Red Zone Max	Red2 Zone Min	Red2 Zone Max	Yellow Zone Min	Yellow Zone Max	Blue Zone Min	Blue Zone Max	Green Zone Min	Green Zone Max	Plot Color	
►	1	Output	V	-	0x00	1			0	0,005	0	1,3		0	0	0	0	0	0	0	0	0	0	None	-
	2	Short	%	•	0x01	1			-128	0,78	-100	100		0	0	0	0	0	0	0	0	0	0	Green	-
*				-																					-

### **PID Group**

The **PID Group** is an administrative item to group the data items in different categories. Some PID groups has PID maps, that is when a specific PID contains information about which other PIDs that are supported.

The Group ID column contains an unique numeric identifier for that group.

The **Prim Mode** column contains the primary mode for the PID group, it can be used to select default mode value when a specific PID Group is selected.

#### Item ID

The **Item ID** is the table for all items within a PID group. This is PIDs, Addresses and other items.

The column **Item ID** must contain a hex value of the form **0xNN**. That value is used in communication with the ECU.

The **Can Plot** column indicates if the value can be plotted in a graph or not. Text strings can't be plotted, static numerical values makes no sense to plot while changing numerical values makes sense to plot and can therefore be plotted.

**Plot Height** is the height of the graph to plot in pixels, and it can range from 60 to 800. [This value is now obsolete, setting it does not have any impact on the new charts.]

**Is PID Map** means that the item in question is used to map which other items that are supported by the ECU. E.g. on Mode 0x01, PID 0x00 there is a map for the supported PIDs in range 0x01 to 0x20 where 0x20 is a new map and so on.

**Refresh** is the refresh frequency of the parameter when doing logging and plotting. Some data is slower in update than other and can therefore be requested at a lower rate. E.g. engine coolant temperature can have the value 1:30 which means that it's only requested once every 30<sup>th</sup> sample period.

#### Data

The **Data** table is where the interpretation of the received data is configured. It contains a lot of fields intended to control the behaviour of data interpretation and presentation.

The **#** (number) column is a numeric ID of the sensor, and controls the order of presentation. The sensors can be sorted by any column of choice and the numbering can be revised by clicking on the **Renumber** button.

The **Name** field can be left blank when it's considered redundant. E.g. when the name of the PID is sufficient.

**Unit** is a list of units that can be selected for presenting the value. This is not used in the calculation, only in presentation.

Address Offset is the position in the payload packet that the data is located at. This shall be entered in hex form.

Byte Count is the number of bytes the raw data consists of.

**Start Bit** is used when processing bit-coded data, leave blank if the data isn't bit-coded.

**End Bit** if it's the same as **Start Bit** it's handled as a single bit, which usually represents a switch On/Off. If it's greater than Start Bit it will mean that the bits in that range are used to form a numeric value that is in the range permitted by the number of bits. Notice that the plotted graphs add an offset for single bit plots to allow you to distinguish between the bits in a byte presented.

**Scale Offset** and **Scale Factor** are two floating point values that are used to convert the raw data to user-friendly data. E.g. converting the temperature 0x50 to a readable value in Celsius will use the **Scale Offset** of -40 and the **Scale Factor** of 1 as follows:

T = (0x50 + [Scale Offset]) \* [Scale Factor]Or in figures: T = (0x50 + (-40)) \* 1

This will give you a value of 40 degrees Celsius (Which is about 104 degrees Fahrenheit). To get the value in Fahrenheit instead you will need to use an offset value of -22.22222222 and a scale factor of 1.8.

This of course has the limitation of only permitting linear scaling of the data, so no advanced formulas can be used.

**Pres. Min** and **Pres Max** are values specifying the range that can be used for plotting data where the min value is the lowest position on the Y axis and the max value is the highest. Any value outside the bounds is considered to be max or min value to avoid runaway plots. Currently these values are only useful when handling the Unit "**bit**".

**Format** is the format to be used when doing a text presentation of the data. Normal alternatives are "0" – no decimals, "0.00" – two decimals etc. The string is fed right into the string formatter for C#, so you have some freedom to choose other format masks as well. If it's empty the format "0.000" is selected. If an invalid format is encountered the default format is used and a "1" is added after the value.

**Zone** fields are used to allow for configuration of colored zones in graphs to indicate when a value is outside recommended range. If the Min and Max value for a zone is identical it's disabled. Recommended is to set the values to zero if they aren't used.

**Plot Color** is controlling the color of some graphs. In other cases it is only used to control if a sensor shall be plotted or not. Any sensor with the color **None** shall not be plotted.

### Modes

The Modes Page is used for the basic modes (also called Service Identifier) supported by the ECU.

Mode [Hex]	Name	Secured	PID Group	ŕ	Save
0x01	Show current data	•	1 SAE PIDs	-	Close
0x02	Show freeze frame data	•	1 SAE PIDs	-	Definition of Mode (also called Service Identifier) Ranges and reference to
0x03	Show stored Diagnostic Trouble Codes.	•		-	definition.
0x04	Clear Diagnostic Trouble Codes and stored values	•		-	00 - 0F Request [SAE J1979] 10 - 1F Request (bit 6 = 0) [SSF 14230-3]
0x05	Test results, oxygen sensor monitoring (non CAN	•		-	20 - 2F Request (bit 6 = 0) [SSF 14230-3]
0x06	Test results, other component/system monitoring	•		-	30 - 3E Request (bit 6 = 0) [SSF 14230-3] 3F - Not Applicable
0x07	Show pending Diagnostic Trouble Codes (detect	•		-	40 - 4F Response SAE J1979 50 - 5F Response to Services (\$10 - \$3E) (bit 6 = 1) (SSF 14230-3)
0x08	Control operation of on-board component/system	◄		•	60 - 6F Response to Services (\$10 - \$3E) (bit 6 = 1) [SSF 14230-3]
0x09	Request vehicle information	•	8 Vehicle Information	-	70 - 7E Response to Services (\$10 - \$3E) (bit 6 = 1) [SSF 14230-3] 7F - Negative Response [SAE J1979], [SSF 14230-3]
0x0a	Permanent DTC's (Cleared DTC's)			-	80 - Request 'ESC' - Code [SSF 14230-3] 81 - 8F Request (bit 6 = 0) [SSF 14230-2]
0x10	Diagnostic Session Control (ISO 14229)	•		-	90 - 9F Request (bit 6 = 0) [Reserved for future exp. as needed]
0x11	ECU Reset (ISO 14229)	•		-	A0 - B9 Request (bit 6 = 0) [Defined by vehicle manufacturer] BA - BF Request (bit 6 = 0) [Defined by system supplier]
0x14	Clear Diagnostic Information (ISO 14229)	2		-	C0 - Positive Resp. 'ESC' - Code [SSF 14230-3] C1 - CF Positive Response (bit 6 = 1) [SSF 14230-2]
0x19	Read DTC Information (ISO 14229)	•		-	D0 - DF Positive Response (bit 6 = 1) [Reserved for future exp. as needed
0x22	Read Extended PID (ISO 14229)	2	2 Extended PIDs.	-	E0 - F9 Positive Response (bit 6 = 1) [Refined by vehicle manufacturer] FA - FF Positive Response (bit 6 = 1) [Refined by system supplier]
0x23	Read Memory By Address (ISO 14229)	~		-	
0x24	Read Scaling Data By Identifier (ISO 14229)	~		-	
0x27	Security Access (ISO 14229)	~		-	
0x28	Communication Control (ISO 14229)	•		-	

The **Mode** column is sent to the ECU in combination with the **PID** value defined above to get or set data in an ECU. The data is used when interpreting responses from the ECU, and in those cases the returned data has 0x40 added to the mode to indicate that it's an answer. There is no need to declare those added values since the application takes care of the 0x40 adding.

**Secured** means that the record is protected against accidental deletion because deleting the record can cause problems in the application.

The **PID Group** column is for informative purposes and is also used when parsing responses from ECU.

### Vehicles

The Vehicles Page is where you as a user can define a vehicle configuration for a specific vehicle. It is possible to copy an existing vehicle and create a new definition that can be changed to suit your needs.

📷 ¥ehicl	es							
Vehicle	Filter on Brand	No Filter	]					
	Name	Brand	Model	Addre Bits	ess	Secured	P	Сору
•	Generic 11-bit			11	-			Save
	Generic 29-bit			29	-			Close
	ISO9141			11	•			I
	Subaru CAN-Bus	Subaru		11	-		$\vdash$	
	Subaru ISO9141	Subaru		11	-			
	Test			29	•		-	
Genera	ECUs and PID Groups Ad	dresses Flags and Parame	ters					
Protoc			•	C	Capa	bilities		
IS01	5765 🗾			Γ		SM - Subar		ect Monitor
Speed						/C1 - Volvo I	CAN	-bus Workaround
250	kbps 💌 N/A 💌							
Π Α	dd Checksum							
Additi	onal OBD codes file Browse	e						
22								

#### Vehicle Table

In this table the **Address Bits** field is important, since it is used to select between the 11-bit addressing mode and the 29 bit addressing mode of ISO15765. It also controls the functionality of the application in some cases.

If your vehicle isn't represented you can start with doing a copy of an existing vehicle that is similar.

#### **General Tab**

The **Protocol** is where you select the communication protocol used on the OBD-II connector. All newer vehicles (2008 and later on US market and most vehicles 2008 and later on other markets) supports ISO15765 and CAN while older are using a lot of different protocols.

**Speed** is one of the speeds defined above. Take care to select a speed that is supported by the protocol you selected.

**Tester Addr** is the address of the application in some configurations. E.g. 29-bit addressing uses this, 11-bit does not. The value is usually 0xF1 or 0xF0, but some vehicles uses other values.

**Add Checksum** is for older protocols that needs checksum on the communicated data. Not applicable for ISO15765.

Additional OBD codes file This is the name of the file containing vendor-specific OBD codes in the data directory. Notice that the vendor specific file is always combined with the general set of OBD codes provided with the application. Vendor specific definitions are the primary alternative if the same code is defined in both data sets.

**Capabilities** Indicates special capability support to be enabled for the vehicle. Currently only SSM (Subaru Select Monitor) is supported which enables some limited extra functionality.

#### **ECU and PID Groups Tab**

This tab contains two lists, one of ECU:s and their addresses and the other of the supported PID groups for the vehicle.

Notice that the address format for the ECU:s depends on the number of address bits and protocol. For ISO15765 11-bit only permits the range 0x00 to 0x07 while 29-bit permits 0x00 to 0xFF.

The PID group table is edited by double-clicking on a row to get a pop-up where the data is configured.

#### Addresses Tab

On this tab the sending and receiving addresses are configured. If you have defined ECU:s in the previous tab you shall only define one broadcast and one fixed address in the Tx table and one address in the Rx table. Values depends on number of bits and protocol.

	Broadcast	Тх	Rx
ISO15765 11-bit	0x07df	0x07e0 to 0x07e7	0x07e8 to 0x07ef
8 addresses			
ISO15765 29-bit	0x18db3300	0x18da0000	0x18da0000
256 addresses		to 0x18daff00	to 0x18da00ff
		mask 0xffffff00	mask 0xffff00ff

#### **Flags and Parameters Tab**

On this tab the specific connection and message flags can be configured, as well as additional IOCTL parameters to be applied at connection.

The IOCTL parameter list varies depending on which protocol that is selected.

For a detailed description of the flags and parameters you need to look at the J2534 spec and the API vendor documentation.

# Connecting

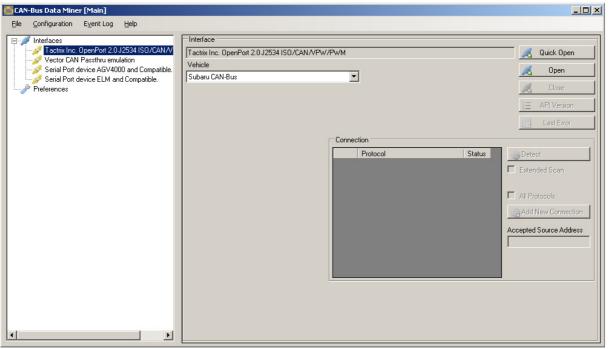
There are two ways to connect to a device, either by selecting the device in the tree to the left or by clicking on the device in the first page.

If you click on a device in the first page a pop-up similar to the following:

m Information	- 🗆 🗙
About Tactrix 2.0	
Tactrix OpenPort 2.0 is supported through the J2534 API.	
OK Cancel	li

If the OK button is active it can be clicked and the device shall be selected in the tree to the left. The item in the tree can also be selected manually.

## Interface



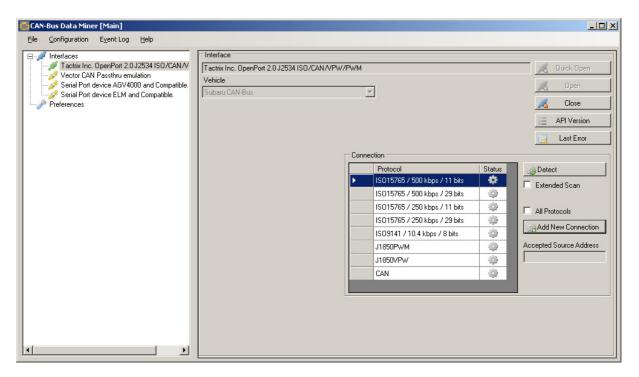
When selecting a device in the tree the page above will be displayed. The content varies depending on the selected device. E.g. serial devices will offer additional fields where the user can select COM port and COM port parameters.

If you have configured your vehicle correctly you may click on the **Quick Open** which is the fast and easy way to get to the OBD data page. Notice that the Quick Open function is pretty stupid and is not able to resolve encountered problems very well.

By using the **Open** button you will instead be able to do a step by step connection to the device and vehicle. This is useful in case you want to try alternatives like using the CAN protocol instead of ISO15765.

Since some devices (and vehicles) can handle multiple protocols simultaneously the user is offered the ability to select protocol for the connection. The default protocol depends on which vehicle that is selected.

All known protocols will be selectable by checking the the checkbox "All Protocols", but be aware that this also means that protocols not documented as being supported by the application, device or vehicle will be provided. This is primarily for testing and shall normally be left unchecked.



Select the protocol to use and click "Add New Connection". This shall bring up a new page where the protocol parameters can be configured.

### Smart Serial Interface

When using a smart serial interface like the AGV4000 and ELM327 there will be a few more interface parameters to consider in order to inform the application of which port the device is connected to. Communication speed, parity, data bits and stop bits can also be configured here if necessary.

🔂 CAN-Bus Data Miner [Main]		
<u>File Configuration Event Log H</u> elp		
Interfaces Tactrix Inc. OpenPort 2.0 J2534 ISD/CAN/V Vector CAN Passthru emulation Serial Port device AGV4000 and Compatible. Serial Port device ELM and Compatible Preferences	Interface         Serial Port device ELM and Compatible.         Vehicle         Subaru CAN-Bus         Port         Port         Databits         Parity         Stopbits         8         Image: Stoppits         9         Image: Stoppits         115200         Image: Stoppits         Image: S	Quick Open

If you don't know which serial port your device was assigned you can disconnect and reconnect it while the page above is displayed. The correct alternative will then be selected in the **Port** drop-down. However you will still need to configure the speed and the other parameters if they are different from the default for the device.

## Protocol

The Protocol page offers the ability to set protocol specific parameters before connecting with the vehicle. These parameters can be specified in the vehicle configuration. Which parameters that are available for configuration depends on which device and protocol that is selected.

🔂 CAN-Bus Data Miner [Main]		
<u>File </u> _onfiguration E⊻ent Log <u>H</u> elp		
Elle Configuration Event Log Help Interfaces Tactite Inc. OpenPort 2.0.32534 ISO/CAN/V Serial Port device AGV4000 and Compatible. Serial Port device ELM and Compatible. Preferences	Protocol         IS015765         Connection Parameters       Ecus & Flags         Connection Flags         Source (Tester) Address Bits         CAN_ID_BOTH         CAN_28BIT_ID         Source (Tester) Address         IS09141_K_LINE_ONLY         IS09141_NO_CHECKSUM         IS09141_FORD_HEADER	Close Connect Disconnect Raw CAN BDJI Gauges
٩	Add Checksum Timeout 1000 = J2534 Speed (kbps) 500 kbps •	Overruns

Notice that J2534 devices may support multiple protocols at the same time, so you can set up both ISO15765 and ISO9141 at the same time using the same device. This will of course require that the device and the vehicle you connect to supports it.

🛅 CAN-Bus Data Miner [Main]			_ 🗆 X
Eile ⊆onfiguration E⊻ent Log Help			
E-∮ Interfaces E-∮ Tactrix Inc. OpenPort 2.0 J2534 ISO/CAN/V	Protocol	Close	
	ISO9141		
S09141	Connection   Connection Parameters   Ecus & Flags	Connect	
	Connection Flags Message Options		
Preferences	CAN_ID_BOTH		
	CAN_29BIT_ID Source (Tester) Address	N Raw CAN	
	□ ISO9141_K_LINE_ONLY □ ISO9141_NO_CHECKSUM	a obd-li	
	ISO9141_NO_CHECKSUM_DT ISO9141_FORD_HEADER Ux6a Broadcast	6 Gauges	
	Ox6a Broadcast		
	Timeout		
	1000 -		
	J2534 Speed (kbps)		
	10.4 kbps		
		Overruns	_

When connecting using a serial port device (AGV or ELM) there are no options to configure and the adapter takes care of all necessary low level data.

1-1-1

Instead a progress bar and an information field about which protocol that was detected will be displayed:

Can-Bus Data Miner [Main]		<u> </u>
Eile Configuration Event Log Help		
Interfaces     Tactrix Inc. OpenPort 2.0 J2534 ISO/CAN/V     S015765     IS015741	Protocol           [SERIAL_AT_ELM           Progress	Close
Vector CAN Passthru emulation Serial Port device AGV4000 and Compatible. Serial Port device ELM and Compatible.	Initialization Detected Protocol	Connect  Disconnect  Raw CAN
- Preferences		Gauges
		Overruns
K b		

The progress bar is stepping up as the initialization commands are stepped through to the adapter and finally the field for the detected protocol shall be filled in with the protocol that the adapter did discover when communicating with the vehicle.

# **CAN Logging**

The CAN logging can currently only be done with the Tactrix adapter, and it requires you to set the connection protocol to **CAN** and the connection flags **SNIFF\_MODE** and **CAN\_ID\_BOTH** and then connect. The data that is logged is raw CAN data and you will need a separate database for the vehicle in question to be able to interpret it.

When clicking on the **Raw CAN** button you will get a new window that is used for CAN data logging and control of the logging.

Pressing on the **Start** button will start the listening for CAN messages and they will be presented in the window as they come in and are updated. By checking the **Append Logging** checkbox the data in the log window is appended otherwise there is a lookup performed and the data with the same MsgID is replaced with the latest. For large volumes it is a good idea to log to file.

Start	Stop 🔽 Ap	ppend Logging Tim	neout (ms) 🚺 🕂	Пт
#	Timestamp	MsgID	Data	
0	-1092494502	00 00 07 df	02 01 00 00 00 00 00 00	
1	-1092493164	00 00 07 e8	06 41 00 b8 7b 30 10 00	
2	-1092343449	00 00 07 df	02 01 00 00 00 00 00 00	
3	-1092342121	00 00 07 e8	06 41 00 b8 7b 30 10 00	
4	-1091090356	00 00 07 df	02 01 00 00 00 00 00 00	
5	-1091089030	00 00 07 e8	06 41 00 b8 7b 30 10 00	
6	-1081699725	00 00 07 df	02 01 Oc 00 00 00 00 00	
7	-1081698339	00 00 07 e8	04 41 Oc 18 c0 00 00 00	

This is the RAW CAN packets that are logged, which is useful for some but the data is often very specific for a brand and a vehicle model.

If you are lucky and have access to Vector Canalyzer you can open the log file in that application for further analysis and processing.

At the bottom of the window you have a text field and an "Add" button which allows you to add comments to the log file that can be used for tracking events, e.g. when switching on the wipers.

# OBD-II

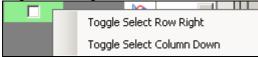
	🕏 Refresh 🛛 🕞	Snapshot Da	ata	Plot Data Interval (ms) 1000	3	Go 🔵	Stop	🔀 Clear
Iso15765     Current Data	ESave 🚺	Load Selecti	ion	🗖 Log Data	E Save	Data	🕒 Sav	ve Charts
	Mode	PID	Freq.	Description	ECM 0x07e8	TCU 0x07e9	Plot	Plot Group
SSM	0x01	0x47	1:1	Absolute throttle position B				- 💌
Vector CAN Passthru emulation	0x01	0x48	1:1	Absolute throttle position C				- 🔻
Serial Port device ELM and Compatible.	0x01	0x49	1:1	Accelerator pedal position D				•
Preferences	0x01	0x4a	1:1	Accelerator pedal position E				•
	0x01	0x4b	1:1	Accelerator pedal position F				- 💌
	0x01	0x4c	1:1	Commanded throttle actuator				• •
	0x01	0x4d	1:1	Time run with MIL on				•
	0x01	0x4e	1:1	Time since trouble codes cleared				· •
	0x01	0x5a	1:1	Relative accelerator pedal position				. 🔻
	0x01	0x5b	1:1	Hybrid battery pack remaining life				- 🔻
	0x01	0x5c	1:1	Engine oil temperature				-
	0x01	0x61	1:1	Driver's demand engine - percent torque				• 🔻
	0x01	0x62	1:1	Actual engine - percent torque				- 🔻
	0x01	0x63	1:1	Engine reference torque				· 🔻
	0x01	0x64	1:1	Engine percent torque data				- 🔻
	0x01	0x65	1:1	Auxiliary input / output supported				- 🔻
	0x01	0x7f	1:1	Engine run time				. 🔻
	0x22	0x114b	1:10	Estimated catalyst temp				. 🔻
	0v22	0v114c	1.10	Estimated DPE temp	Г			

This window is actually a bit more than OBD-II, but OBD-II is the base functionality that is used. Some brand specific items are added here too – like part of SSM (Subaru Select Monitor).

Due to the amount and type of information it is split into several pages.

The checkbox columns decide which ECU to communicate with and which data to request. Green background means that the ECU has confirmed support for the item, dark grey means that the item is unavailable while yellow means that the ECU **MAY** support the item.

Right-clicking on a field in one of the ECU columns will bring up a menu:



In this menu you can select all checkboxes from the current box to the last on the row or column. Notice that when selecting column only those items that have the same **Plot** state as the current cell will be selected.

## Current Data

On this page there are several buttons, a table and a few other items.

Buttons:

- **Refresh** Requests data from the vehicle about supported fields. Also clears all selected items.
- **Snapshot data** Requests one shot of data values from the vehicle using the data selected by the checkboxes in the table. If the **Plot Data** checkbox is checked the data ends up in charts otherwise in text form.
- **Save** Save the current selection configuration. Useful when you have a specific configuration that you want to re-use or share with friends.
- Load Selection Load a saved configuration from file.
- **Go** Used to start logging or plotting using data requested by the interval given by the **Interval (ms)** field. The button is only activated if Plot Data and/or Log Data checkboxes are checked.
- **Stop** Stops request of logging/plotting data.
- Clear Clear the current set of sub panels to the data panel.
- Save Data Save the data in each sub panel to a directory selected by the user. The user is prompted to choose between saving data in one single file with joined data or multiple files with one file for each PID.
- **Save Charts** Save all charts in sub-panels as PNG images of size 1024x768.

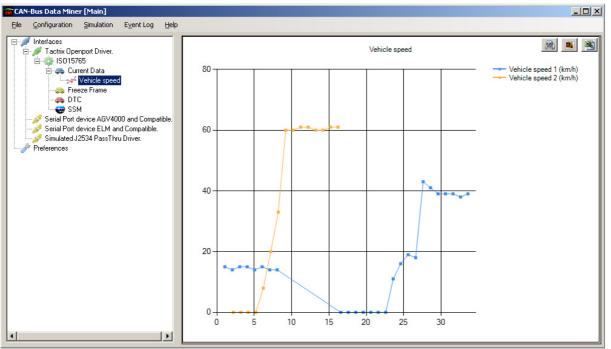
Checkboxes:

- Plot Data Checked if data shall be plotted to charts.
- Log Data Checked if data shall be logged to file. This is disabled if no log file is opened.

Data Table Columns:

- **Mode** The mode (service ID) value for the item.
- **PID** PID, Address or other identifier for the item.
- Freq Sample frequency, provided for information.
- **Description** Text describing the PID.
- **<ECU Column(s)>** One or more columns presenting ECUs supported and which PIDs they support. This can be one or more columns.
- **Plot** Has an icon indicating if the data can be plotted on a chart. (Data that varies over time has this)
- **Plot Group** Used when two or more values shall appear in the same chart. Use the same number for each value that shall appear in the same chart. Up to nine groups are supported.

## Chart data



Above is a simple chart. The chart offers three buttons, Print, Save image and Save data. The buttons are hidden while the chart is updated and displayed when the logging of data is stopped.

It is possible to stop the plotting and re-start it. A question will be asked at restart if it shall continue on the same series or start a new series. The selection will impact all charts, there is no way to select different action for each chart plotted.

## Snapshot Data

✓ Interfaces	ECU ECM	Save 📄 Print	👃 Get	🥩 Clear & Get
🚊 🪙 Current Data	PID	Description	Unit	Data 🔺
ECM Freeze Frame	0x04	Calculated engine load value	%	20,0
	0x05	Engine coolant temperature	°C	-1
SSM	0x0B	Intake manifold absolute pressure	kPa	101,000
	0x0C	Engine RPM	rpm	1069,250
Serial Port device ELM and Compatible.     Preferences	0x0D	Vehicle speed	km/h	4,000
	0x0F	Intake air temperature	°C	0,000
	0x10	MAF air flow rate	g/s	20,490
	0x11	Throttle position	%	78,824
	0x23	Fuel Rail Pressure (diesel, or gasoline direct inject)	kPa	36150,000
	0x33	Barometric pressure	kPa	100,000
	0x42	Control module voltage	V	14,589
	0x46	Ambient air temperature	°C	6,000
	0x49	Accelerator pedal position D	%	19,216
	0x4A	Accelerator pedal position E	%	19,216
	0x4C	Commanded throttle actuator	%	79,216
	0x07	Engine Load	%	100,0
	0x08	Coolant Temperature.: Celsius	°C	0
	0x08	Coolant Temperature.: Fahrenheit	۴F	32,000
	0x09	Air/Euel Correction #1	%	99 219

Snapshot data is the data that is the result of a single data request. The **Get** button appends data to the list while the **Clear & Get** button clears the table and refreshes it with latest data.

Data fields with light yellow color are fields that are potentially invalid (the data was all ones from the vehicle indicating a raw value of 255, 65535 etc.).

The **Save** button saves the data to a CSV file that can be opened in Excel and other programs supporting that format.

The **Print** button will print the data to a printer of your choice. It is designed to work with standard paper sizes like **Letter** or **A4**.

## Howtos

Here a few common tasks are described. More will be added from time to time.

## Data Log

Connect the adapter to the vehicle and computer, start the application. Select the adapter and then the vehicle model.

CAN-Bus Data Miner [Main]		<u>_                                    </u>
Eile Configuration Simulation Event Log Help		
Interfaces     Vector CanCard / CanCase     Vector CanCard / CanCase     Serial Port device AGV4000 and Compatible.     Serial Port device ELM and Compatible.     Simulated J2534 PassThru Driver.     Preferences	Interface Tactrix Inc. OpenPort 2.0 J2534 ISO/CAN./VPW/PWM Vehicle Generic 11-bit	Quick Open
	Protocol	Add New Connection

Click on "Quick Open" if you know that the combination of vehicle and protocol is supported and you shall get the following page:

Vector CanCard / CanCase	💈 Refresh 🛛 🚺	Snapshot [		Plot Data Interval (ms)		G		Stop	😢 Clear	-	
🖻 🎲 ISO15765	Save [	Save Log Data				E Save Data			Cave Charts		
Current Data	Mode	PID	Freq.	Description	ECU 0x7e8	Plot	Sel.	Multi- Sel.	Plot Group		
a DTC	▶ 0x01	0x01	1:1	Monitor status since DTCs cleared.	Supported			1	•	•	
Serial Port device AGV4000 and Compatible.	0x01	0x03	1:1	Fuel system status	Supported			1		•	
Simulated J2534 PassThru Driver.	0x01	0x04	1:1	Calculated engine load value	Supported			1	•	•	
Preferences	0x01	0x05	1:30	Engine coolant temperature	Supported			1	•	Ŧ	
	0x01	0x0a	1:1	Fuel pressure	Supported			1	•	•	
	0x01	0x0b	1:1	Intake manifold absolute pressure	Supported			1	· .	Ŧ	
	0x01	0x0c	1:1	Engine RPM	Supported			1	•	•	
	0x01	0x0d	1:1	Vehicle speed	Supported			1	•	•	
	0x01	OxOf	1:10	Intake air temperature	Supported			1	•	Ŧ	
	0x01	0x10	1:1	MAF air flow rate	Supported			1	· .	•	
	0x01	0x13	1:1	Oxygen sensors present	Supported			Ţ	•	•	
	0x01	0x14	1:1	Bank 1, Sensor 1: Oxygen sensor voltag	Supported			Ļ	•	Ŧ	
	0x01	0x1c	1:1	OBD standards this vehicle conforms to	Supported			1	-	•	

Under the "File" menu select "Open Measurement Data Log File..." and select a suitable location and name for the file. The data shall be written in Excel CSV format (actual format differs depending on the locale setting of your computer). US format is using "," as a field separator and "." as a decimal separator while some European countries uses ";" as a field separator and "," as a decimal separator.

CAN-Bus Data Miner [Main]										- 0
Eile ⊆onfiguration ≦imulation E⊻ent Log Help										
E- Ø Interfaces ↓ Vector CanCard / CanCase	🕏 Refresh 🛛 🦷	🔲 Plot Data 🛛 Interval (ms)	s) 🚺 📑 🗔 Go 🖉 Stop 🔀 Clear					ar		
	Save 🚺	🖺 Save Data 📔 Save Charts								
Current Data	Mode	PID	Freq.	Description	ECU 0x7e8	Plot	Sel.	Multi- Sel.	Plot Group	
- 🚑 DTC	0x01	0x01	1:1	Monitor status since DTCs cleared.	Supported			↓.	•	•
Serial Port device AGV4000 and Compatible.	0x01	0x03	1:1	Fuel system status	Supported			↓ ↓	•	•
Simulated J2534 PassThru Driver.	0x01	0x04	1:1	Calculated engine load value	Supported			Ļ	· [	-
	0x01	0x05	1:30	Engine coolant temperature	Supported			↓ ↓	· [	-
	0x01	0x0a	1:1	Fuel pressure	Supported			↓ ↓	. [	• •
	0x01	0х0Ь	1:1	Intake manifold absolute pressure	Supported			↓ ↓	· .	-
	0x01	0x0c	1:1	Engine RPM	Supported		~	1	· [	-
	0x01	0x0d	1:1	Vehicle speed	Supported		◄	1	· [	-
	0x01	0x0f	1:10	Intake air temperature	Supported			1		-
	0x01	0x10	1:1	MAF air flow rate	Supported		◄	1	•	-
	0x01	0x13	1:1	Oxygen sensors present	Supported			Ţ	· [	•
	▶ 0x01	0x14	1:1	Bank 1, Sensor 1: Oxygen sensor voltag	Supported			1	•	-
	0x01	0x1c	1:1	OBD standards this vehicle conforms to	Supported			Ļ	•	•

Check the "Log Data" checkbox and the items you want to log to file as in the following example:

Set the "Interval (ms)" value to the interval between each log event that you want to have. Then click on the "Go" button to start the logging.

To indicate that the application is working there is a progress bar that moves in the field below the "Interval (ms)" label.

Click the "Stop" button to stop logging. Don't forget to close the data log file before trying to read it with another program.

No threshold value exists to trig start and stop of logging.

## **Supported Hardware Reference**

Since this software supports several hardware devices a reference to the devices is in order.

Device	Description
Tactrix 2.0	An OBD adapter provided by Tactrix Inc.
	http://www.tactrix.com/
UWA 020	The Volvo Trucks standard adapter.
	http://www.volvotrucks.com/
DiCE	The Volvo Cars standard adapter.
	http://www.volvocars.com/
CanCase XL	A device for CAN bus traffic logging and analysis.
	https://www.vector.com/vi_index_en.html
ElmScan 5	One device using the ELM327 (or compatible) chip:
	http://www.scantool.net/elmscan5-compact.html
ObdLink SX	One device using the ELM327 (or compatible) chip:
	http://www.scantool.net/scan-tools/pc-based/obdlink-sx.html
AGV4000	A device with a chip that's similar to the ELM327 but not
	entirely compatible. Notice that the AGV4000 and AGV4000B
	are two different devices that looks almost identical, but the
	AGV4000 has some bugs that the AGV4000B corrects.
	AGV4000 is supported, but not recommended due to the bugs. <u>http://agv4000.de/</u>
AGV4000B	Similar to the ELM device but not entirely compatible, does not
	have the bugs that the AGV4000 has.
	http://agv4000.de/
Tactrix 1.3	An older cable from Tactrix inc which is limited in functionality
	and requires the PC to implement the protocol. Will only work
	with ISO9141 and ISO14230 when it's supported.
	http://www.tactrix.com/
Vag-COM KKL	A cable similar to the Tactrix 1.3 which requires the PC to implement the protocol.
	http://www.scantool-direct.co.uk/vag-com-kkl-usbdiagnostic-
	interface-49-p.asp
L	

For further information about ELM chips see; <u>http://elmelectronics.com/obdic.html</u>

There is a chip STN1110 which claims to be compatible with the ELM chip but have better performance. No tests have been performed with that chip, and it may work or it may not work. See <u>http://www.obdsol.com/stn1110/</u> for further info.

Other J2534 devices shall work if they support the 04.04 version of the API. The application searches for the devices in the registry, so don't be surprised if your odd device is listed.

# **Unsupported Devices**

Since some devices are problematic they are listed here as "Unsupported" devices.

Device	Description			
ELM327 Ver 1.5a	A Chinese clone, the one in the picture doesn't even work.			
	want to try it you can do so since it's not blocked in the software, but don't expect it to work.			