PCAN-USB Pro FD

User Manual

Relevant Product

Product name	Part number
PCAN-USB Pro FD	IPEH-004061

Imprint

PEAK-System Technik GmbH
Otto-Röhm-Straße 69
64293 Darmstadt
Germany

Phone: +49 6151 8173-20
Fax: +49 6151 8173-29

www.peak-system.com
info@peak-system.com

Document version 2.0.0 (2023-08-18)

Contents

Imprint **2**

Relevant Product **2**

Contents **3**

1 Introduction **5**
- 1.1 Properties at a Glance 6
- 1.2 System Requirements 7
- 1.3 Scope of Supply 8

2 Configuration **9**
- 2.1 Open the Casing 10
- 2.2 Voltage Supply of External Devices 11
- 2.3 Internal Termination 13

3 Installation **15**
- 3.1 Install Device Driver Setup 15
- 3.2 Connection 16
- 3.3 Check Operational Readiness 16
- 3.4 Connecting the CAN and LIN Bus 16
- 3.5 CAN Cabling 17
- 3.6 Example Application under Windows 18

4 Operation **19**
- 4.1 Status LEDs 19
- 4.2 Unplugging the USB Connection 20
- 4.3 Distinguishing several interfaces 20

5 CAN Monitor PCAN-View **21**
- 5.1 Initialize CAN interface 22
- 5.2 Transmit CAN message 24
- 5.3 Additional Tabs 25

6 API PCAN-Basic **30**
- 6.1 Features of PCAN-Basic 31
- 6.2 Principle Description of the API 32

7 LIN Monitor PLIN-View Pro **33**
- 7.1 Features 34
- 7.2 Start and Initialize PLIN-View Pro 35

7.3 Receive / Transmit Tab 37
7.4 Transmit a LIN Frame 37
7.5 Trace Tab 40
7.6 Scripting Tab 41
7.7 LIN Interface Tab 42
8 PLIN-API 43
8.1 Features 44
9 Technical Data 45
Appendix A CE Certificate 47
Appendix B UKCA Certificate 48
Appendix C Dimension Drawings 49
Appendix D Quick Reference 50
Appendix E Linux 53
Appendix F Disposal 54

1 Introduction

The PCAN-USB Pro FD interface enables the connection of CAN FD and LIN networks to a computer via USB. Two field buses can be connected at the same time, up to four with appropriate adapter cables (2 x CAN FD, 2 x LIN). Each CAN FD channel is separately isolated against USB and LIN with a maximum of 500 Volts. Its robust aluminum casing makes the PCAN-USB Pro FD interface suitable for mobile applications.

The CAN FD standard (CAN with Flexible Data rate) is primarily characterized by higher bandwidth for data transfer. The maximum of 64 data bytes per CAN FD frame (instead of 8 so far) can be transmitted with bit rates up to 12 Mbit/s. CAN FD is downward compatible to the CAN 2.0 standard, CAN FD nodes can be inserted into already existing CAN networks without CAN FD extensions.

The monitor software PCAN-View and the programming interface PCAN-Basic for the development of applications with CAN connection are included in the scope of supply and support the standard CAN FD. The monitor application PLIN-View Pro as well as the PLIN programming interface are also included in the scope of supply.

This manual describes the use of the CAN interface with **Windows**. CAN device drivers and application information for **Linux**: www.peak-system.com/quick/DL-Driver-E

At the end of this manual you can find a Quick Reference with brief information about the installation and operation of the interface.

1.1 Properties at a Glance

Common

- Interface for High-speed USB 2.0 (compatible to USB 1.1 and USB 3.0)
- Transmitting and receiving of CAN FD and LIN messages using 2 D-Sub connections (both with pin assignment for the CAN FD and LIN bus)
- Time stamp resolution 1 µs
- 5-Volt supply at the D-Sub connector can be activated through a solder jumper, e.g., for external bus converter
- Voltage supply via USB
- Extended operating temperature range from -40 to +85 °C (-40 to +185 °F)

CAN

- Complies with CAN specifications 2.0 A/B and FD
- CAN FD support for ISO and Non-ISO standards switchable
- CAN FD bit rates for the data field (64 bytes max.) from 25 kbit/s up to 12 Mbit/s
- CAN bit rates from 25 kbit/s up to 1 Mbit/s
- FPGA implementation of the CAN FD controller
- NXP TJA1044GT CAN transceiver
- Each CAN FD channel is separately opto-decoupled against USB and LIN up to 500 V
- CAN termination can be activated through solder jumpers, separately for each CAN channel
- Measurement of bus load including error frames and overload frames on the physical bus
- Induced error generation for incoming and outgoing CAN messages

LIN

- Bit rates from 1 kbit/s up to 20 kbit/s
- TJA1028 LIN transceiver
- Compliant with all LIN specifications (up to version 2.2)
- Both LIN channels (common ground) are opto-decoupled against USB and CAN FD
- Can be used as a LIN master or slave (1 ms master task resolution)
- Automatic bit rate, frame length, and checksum type recognition
- Autonomous scheduler with support for unconditional, event, and sporadic frames
- Hardware can work through a schedule table (up to 8 schedule tables can be configured with a total of 256 slots)

1.2 System Requirements

Computer with:

- Operating system Windows 11 (x64/ARM64), 10 (x64), or Linux
- a vacant USB port (USB 1.1, USB 2.0, or USB 3.0) or
- a vacant USB port at a self-powered USB hub

LIN

Per channel a supply voltage between 8 and 18 V for the LIN transceiver

1.3 Scope of Supply

- PCAN-USB Pro FD in aluminum casing

Downloads

- Device driver setup for Windows 11 (x64/ARM64), 10 (x64) including:
  - CAN FD device driver
  - LIN device driver
  - CAN monitor PCAN-View
  - LIN monitor PLIN-View Pro
- Device driver for Linux
- Programming interface PCAN-Basic for developing applications with CAN connection
- Programming interface PLIN-API for developing applications with LIN connection
- Programming interfaces for standardized protocols from the automotive sector
- Manual in PDF format

Optional accessories

Pre-configured cable set for PCAN-USB Pro FD and PLIN-USB (IPEK-003013)

2 Configuration

The following describes the configuration for the power supply of external devices and the internal termination. If you do not need any of these options, skip this chapter.

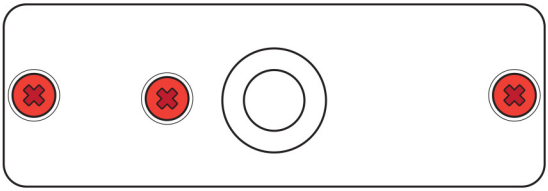
The interface must be opened for the configuration.

Risk of short circuit! Soldering on the interface may only be performed by qualified electrical engineering personnel.

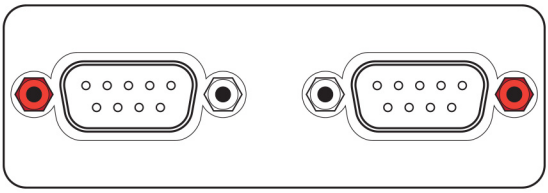
Attention! Electrostatic discharge (ESD) can damage or destroy components on the card. Take precautions to avoid ESD.

2.1 Open the Casing

1. Remove the three screws at the back of the casing.



2. Remove the two outer screws next to the D-Sub connectors at the front of the casing.



3. Remove the casing cover.
4. Lift the back of the case and slide the circuit board in direction of the front over the lower part of the casing.
5. When you have finished the configuration, close the interface.

Close interface

1. Slide the circuit board back into the lower part of the casing.
2. Close the casing cover.
3. Tighten the screws at the front and back side.

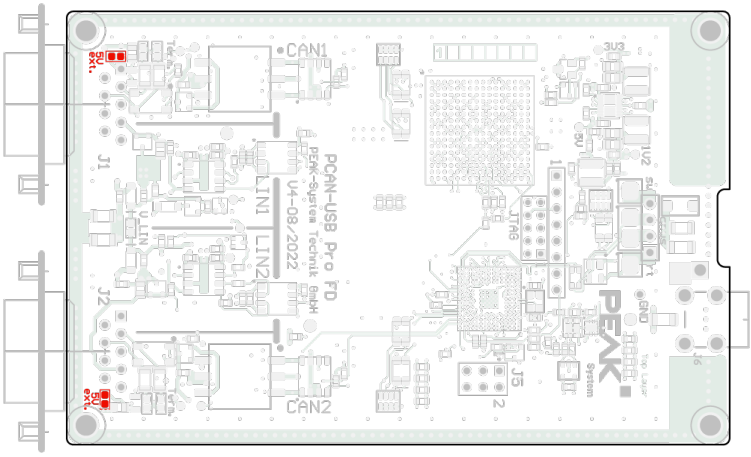
2.2 Voltage Supply of External Devices

> **Attention!** The voltage supply for external devices is not protected separately. Therefore, turn off the computer before you connect and disconnect CAN cables or peripheral systems. Consider that some computers still supply the USB ports with power even when they are turned off (standby operation).

Optionally, an external power supply can be connected via the D-Sub connector using solder bridges at pin 1 on the D-Sub connector. This allows external devices to be supplied with a voltage of 5 V DC, such as the PCAN-TJA1054 bus converter for Low-speed CAN. The power supply is not activated at delivery. The current output is limited to 50 mA.

For PCAN-USB Pro FD from S/N 24548

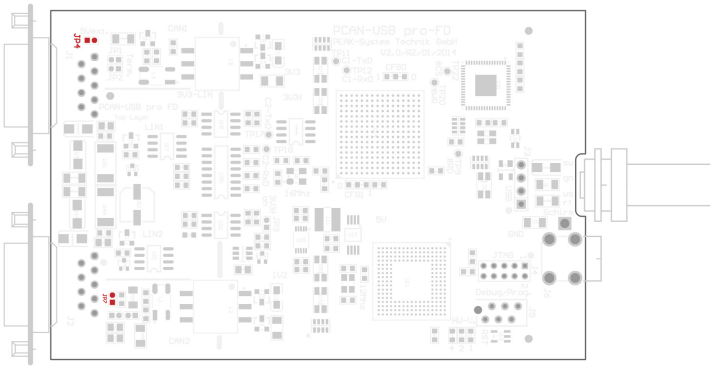
Set the solder bridge(s) according to the desired configuration.



D-Sub connector	Solder field	5-Volt supply at Pin 1	
		Without (Standard)	Active
CAN 1	5V ext.		
CAN 2	5V ext.		

For PCAN-USB Pro FD to S/N 24547

Set the solder bridge(s) according to the desired configuration.



D-Sub connector	Solder field	5-Volt supply at Pin 1	
		Without (Standard)	Active
CAN 1	JP4		
CAN 2	JP7		

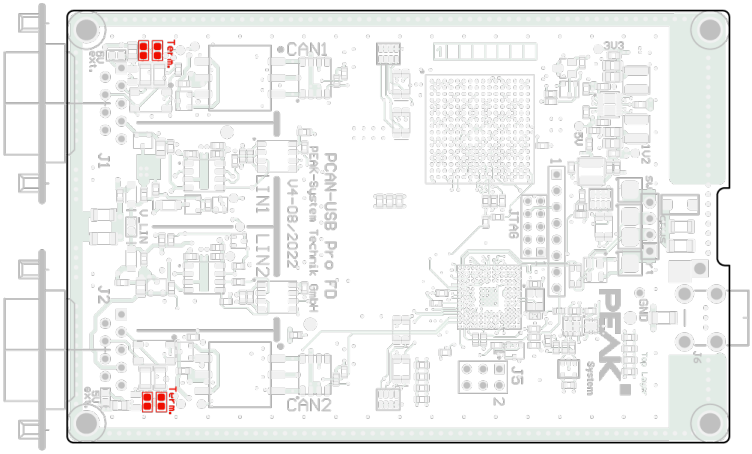
2.3 Internal Termination

The termination can be activated by solder jumpers on the circuit board, to terminate one end of the CAN bus. At delivery the termination is not activated.

Tipp: We recommend to do termination at the CAN cabling, for example with the terminating resistors PCAN-Term (IPEK-003002) or PCAN-MiniTerm (IPEK-003002-Mini). Thus, CAN nodes can be flexibly connected to the bus.

For PCAN-USB Pro FD from S/N 24548

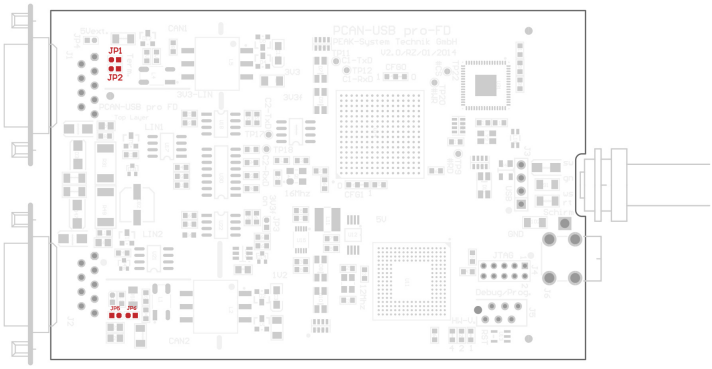
Set the solder bridges corresponding to the desired settings.



D-Sub connector	Solder field	Termination	
		Without (Standard)	Active
CAN 1	Term.		
CAN 2	Term.		

For PCAN-USB Pro FD to S/N 24548

Set the solder bridges corresponding to the desired settings.



D-Sub connector	Solder field	Termination	
		Without (Standard)	Active
CAN 1	JP1 and JP2		
CAN 2	JP5 and JP6		

3 Installation

This chapter covers the software setup for the interface PCAN-USB Pro FD under Windows and the connection of the interface to the computer.

> **Note:** For an installation of the CAN interface under Linux read Appendix E *Linux*.

> Install the driver package before you connect the PCAN-USB Pro FD.

3.1 Install Device Driver Setup

1. Download the device driver setup from our website: www.peak-system.com/quick/DL-Driver-E
2. Unpack the file `PEAK-System_Driver-Setup.zip`
3. Double-click the file `PeakOemDrv.exe`
   The driver setup starts.
4. Confirm the *Start* and the *License agreements*.
5. Follow the instructions. When getting to the *Components Selection*, add the LIN Driver.

3.2 Connection

Attention! Do not use a USB extension cable to connect the interface to the computer. Extension cables do not comply with the USB specification.

1. Connect the interface to a USB port of the computer or of a connected USB hub. Windows detects the new hardware and completes the driver installation.
2. Check the LEDs on the interface. If the LED „USB“ is blinking and the other LEDs light up green, the driver was initialized successfully.

3.3 Check Operational Readiness

1. Open the Windows Start menu.
2. Type `Peak Settings` and press Enter.
   The window *PEAK-Settings* appears.
3. Select *CAN Hardware*.
   The connected CAN interface is displayed.
4. Select *LIN Hardware*.
   The connected LIN interface is displayed.

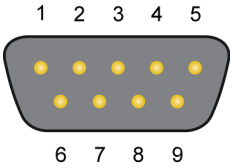
3.4 Connecting the CAN and LIN Bus

The two connectors CAN1/LIN1 and CAN2/LIN2 can each be connected simultaneously to a CAN and a LIN bus. Depending on the actual application, the D-Sub sockets on the bus side must be assigned accordingly. The assignment for CAN corresponds to the specification CiA® 106.

For each CAN connection there is galvanic isolation up to 500 V against USB and LIN.

Both LIN connections have a common galvanic isolation up to 500 V against USB and CAN. The LIN connections are not galvanically isolated against each other.

The transceiver of the LIN interface requires a power supply between 8 and 18 V DC. This must be connected via pin 9 at the D-Sub connector. Both LIN connections need their own supply.

Pin	Assignment	D-Sub socket
1	CAN_V+ (external, optional)	
2	CAN_Low	
3	CAN_GND	
4	LIN	
5	LIN_GND	
6	LIN_GND	
7	CAN_High	
8	Not assigned	
9	LIN_V_{Bat}	

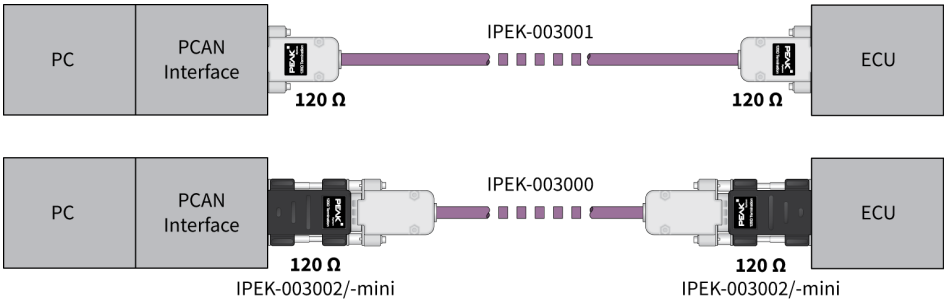
PEAK System offers a pre-configured LIN connection cable (IPEK-003013), wich allows the tapping of individual lines. The LIN and the supply lines are lead directly out to separate plugs and the CAN lines are forwarded to a D-Sub plug.

3.5 CAN Cabling

3.5.1 Termination

The High-speed CAN bus (ISO 11898-2) must be terminated with 120 Ω on both ends. The termination prevents interfering signal reflections and ensures the proper operation of the transceivers of the connected CAN nodes (CAN interfaces, control devices).

3.5.2 Example of a Connection



This example shows a connection between the PCAN Interface and a control unit (ECU). The upper example shows a connection with a cable which is terminated with 120 Ω at both ends. At the lower example the connection is made with termination adapters.

3.5.3 Maximum Bus Length

The maximum bus length depends primarily on the bit rate:

Nominal bit rate	Buslength
1 Mbit/s	40 m
500 kbit/s	110 m
250 kbit/s	240 m
125 kbit/s	500 m
50 kbit/s	1.3 km
25 kbit/s	2.5 km

The listed values have been calculated on the basis of an idealized system and can differ from reality.

Note: For CAN FD, the same maximum bus lengths applys as for CAN, despite the higher data bit rate of CAN FD. The dependency is based on the bit rate during the arbitration, called nominal bit rate.

3.6 Example Application under Windows

As an example application for accessing the interface, run the CAN monitor PCAN-View and/or the LIN monitor PLIN-View Pro from the Windows Start menu.

4 Operation

4.1 Status LEDs

For indication of operating conditions the PCAN-USB Pro FD interface has several LEDs:

USB

Status	Meaning
Green on	High-speed USB is connected (≥USB 2.0)
Green blinking	Communication via High-speed USB
Orange on	Full-speed USB is connected (USB 1.1)
	The PCAN-USB Pro FD interface is in suspend mode (only voltage supply via USB cable, e.g. during computer standby).
Green on	Communication via Full-speed USB

CAN1 and CAN2

Status	Meaning
Green on	The CAN interface is initialized. There's a connection to a driver of the operating system.
Green slow blinking	A software application is connected to the CAN channel.
Green quick blinking	Data is transmitted via the connected CAN bus.
Red quick blinking	An error is occurring during the transmission of CAN data.

LIN1 and LIN2

Status	Meaning
Green on	There's a connection to a driver of the operating system.
Green slow blinking	The LIN channel is initialized with a valid bit rate. A software application is connected to the LIN channel.
Green quick blinking	Data is transmitted via the connected LIN bus.

4.2 Unplugging the USB Connection

The interface can be disconnected from the computer without further actions. In Windows, the interface is not listed under "Safely Remove Hardware".

4.3 Distinguishing several interfaces

You can operate several interfaces on a single computer at the same time. For this purpose, the device ID for CAN and the hardware ID for LIN can be determined in order to distinguish the interfaces in a software environment. The ID's are permanently stored in the interface.

PCAN-View / PLIN-View Pro

To set the device ID in PCAN-View or PLIN-View Pro:

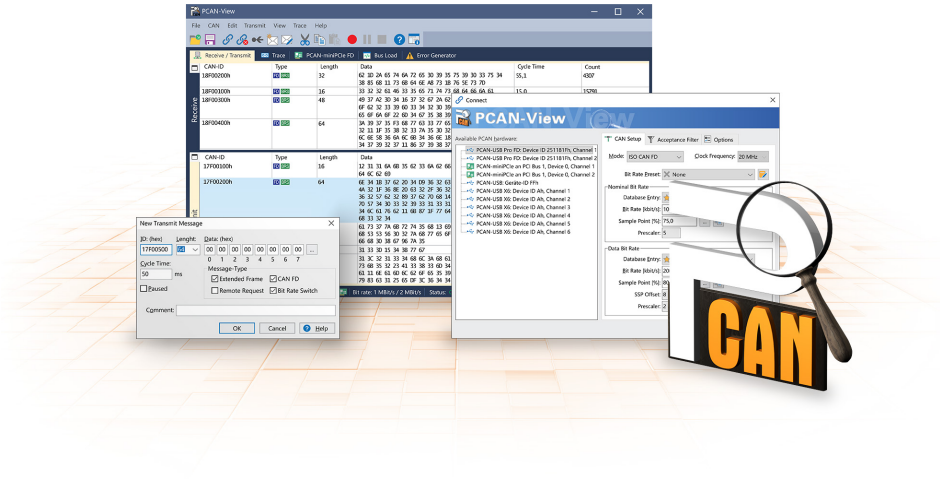
1. Open the tab *PCAN-USB Pro FD* of the respective program.
2. Enter a hexadecimal number with suffix “h” as the new hardware ID.
3. Confirm the entry with *Set*.

PEAK-Settings

To set the device ID in PEAK-Settings:

1. Click on *CAN Hardware* or *LIN Hardware*.
   The connected hardware is displayed.
2. Enter a hexadecimal number with suffix “h” as the new hardware ID.
3. Confirm the entry with *Set*.

5 CAN Monitor PCAN-View



The CAN monitor PCAN-View is a Windows software for viewing, sending, and recording CAN and CAN FD messages. The software is installed with the installation of the device driver package under Windows.

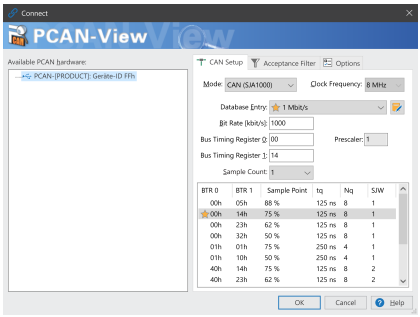
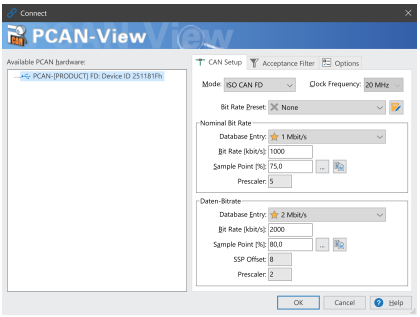
In the following the initialization of a CAN interface is described as an example.

Detailed information about using PCAN-View can be found in the program window under the menu item *Help*.

5.1 Initialize CAN interface

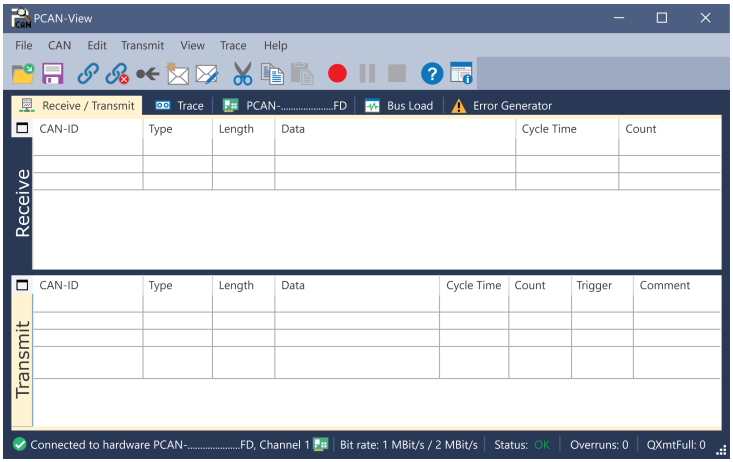
1. Open the program *PCAN-View* via the Windows Start menu.

   Depending on the CAN interface the *Connect* dialog is displayed with or without settings for CAN FD.

CAN	CAN FD
	

CAN interface	List entry in *Available Hardware*
USB Interface, 1-channel	see example above
USB Interface, 2-channel	PCAN-USB Pro FD: Device ID 251181Fh, Channel 1 PCAN-USB Pro FD: Device ID 251181Fh, Channel 2
PCIe Interface, 2-channel	PCAN-PCI Express at PCI Bus 1, Device 0, Channel 1 PCAN-PCI Express at PCI Bus 1, Device 0, Channel 2

2. If there are several CAN interfaces, select the desired interface. For multiple channels, select the desired channel from the list.
3. Enter the *bit rate(s)* and other settings according to the connected CAN bus.
4. Confirm the entries with *OK*. The main window appears and displays the *Receive / Transmit* tab.

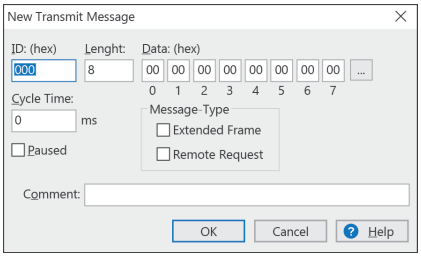
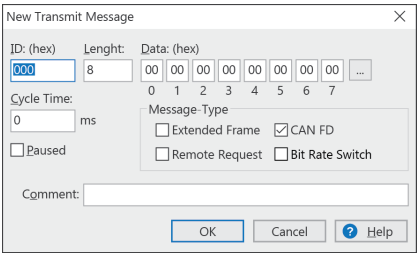


5. For initializing another channel or CAN interface, open another instance of *PCAN-View*.

5.2 Transmit CAN message

1. Select the menu command *Transmit / New Message*.

   Depending on the CAN interface, the dialog box *New Transmit Message* is displayed with or without settings for CAN FD.

CAN	CAN FD
	

1. Enter the *ID*, *Length* and *Data* of the message. Other settings can be made according to the connected CAN bus.
2. Enter a value into the *Cycle Time* field to choose manually or periodically message transmission.

   To transmit periodically enter a value greater than 0.

   To transmit only manually enter the value 0.
3. Confirm the entries with *OK*.

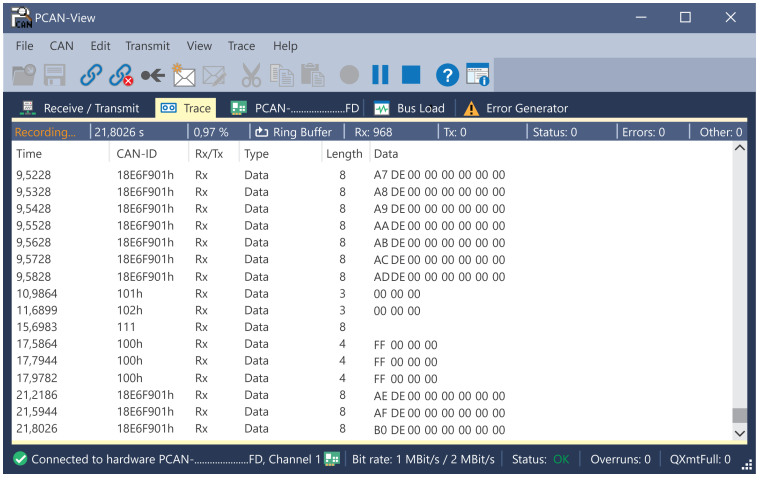
   The created transmit message appears on the *Receive / Transmit* tab.
4. To send the message manually, select the menu command *Transmit* > *Send* or press the space bar.

   The manual transmission process is performed additionally for periodically transmitted CAN messages.

5.3 Additional Tabs

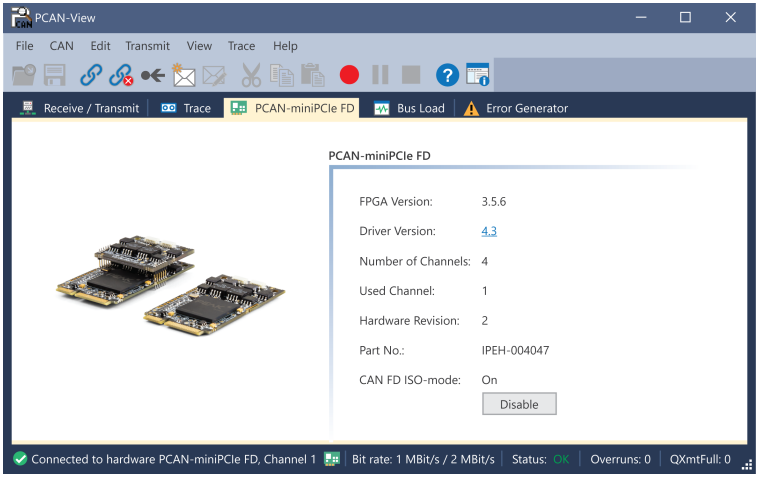
Depending on the CAN interface, additional tabs are available.

5.3.1 Trace Tab



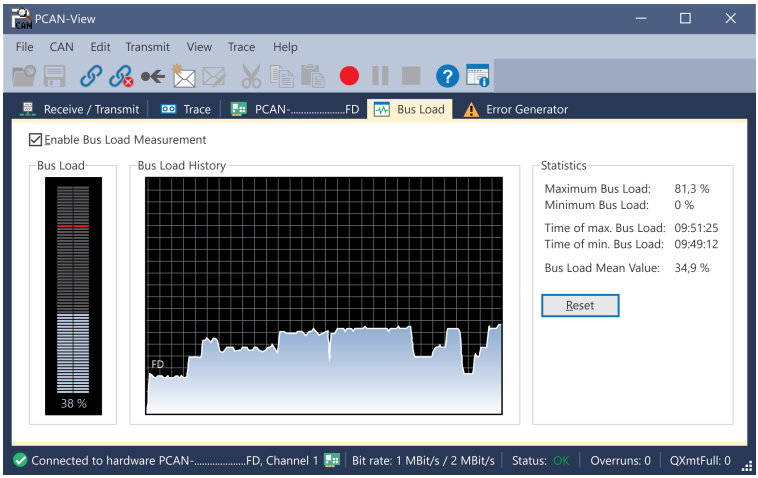
The tracer (data logger) records the communication of the CAN bus in linear or ring buffer mode. The trace data can be saved to a file.

5.3.2 CAN interface Tab



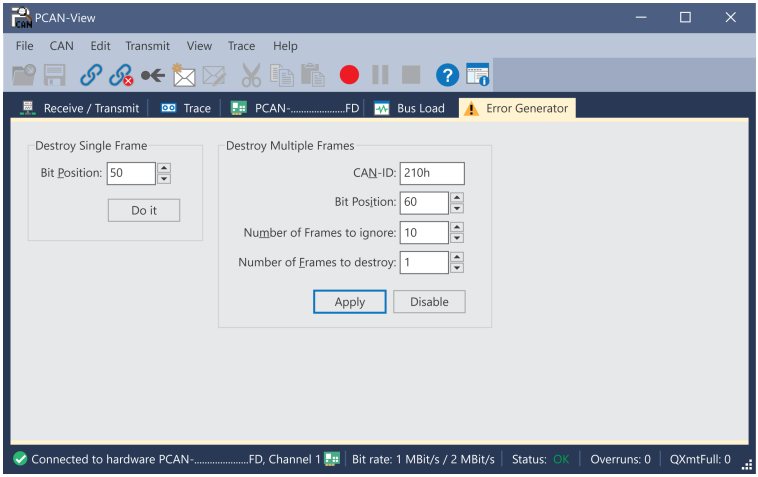
The *CAN interface* tab shows information about the hardware and the used Windows device driver. In this case exemplary for the PCAN-miniPCIe FD. Depending on the CAN interface, a hardware ID can be determined to distinguish several interfaces of the same type. For interfaces with CAN FD a transmission according to "ISO" or "Non-ISO" can be set as default of the hardware.

5.3.3 Bus Load Tab



The *Bus Load* tab displays the current bus load, its time history and statistical information of the connected CAN channel.

5.3.4 Error Generator Tab



Via the *Error Generator* tab the communication on the CAN bus in test environments or during the development of CAN buses can be disturbed in a controlled way by 6 consecutive dominant bits. This is a violation of the CAN protocol on the CAN bus which must be recognized as an error by the connected CAN nodes.

> **Note:** The Error Generator should only be used by experienced users and in the development environment. For further information, please contact our customer support: support@peak-system.com

You can destroy CAN frames with the error generator by one of two methods:

- once after activation
- repeatedly at specific intervals related to a CAN ID

Destroy Single CAN Frame

The *Destroy Single Frame* area refers to the next CAN frame that is recognized after activation.

1. Enter the *Bit Position* where in the CAN frame the error is to be generated. The bit position must start after the identifier. The count includes the stuff bits.
2. Execute the destroy action with *Do it*.

   The next received or transmitted CAN frame will be destroyed at the selected bit position.

Destroy Multiple CAN Frames

1. Enter the *CAN ID* of the CAN frame that is intended to be destroyed multiple times. The following specifications refer to this ID.
2. Enter the *Bit Position* where in the CAN frame the error is to be generated. The bit position must start after the identifier. The count includes the stuff bits.
3. If CAN messages are to be sent unharmed before being destroyed, specify the *Number of Frames to ignore*.
4. Determine the *Number of Frames to destroy*.
5. Confirm the entries with *Apply* to activate the error generator.
6. Stop destroying further CAN frames with *Disable*.

6 API PCAN-Basic

The intended use of PCAN-Basic requires compliance with the license rights. Read the license agreement for end users at: https://www.peak-system.com/quick/eula

The programming interface (API) PCAN-Basic provides basic functions for the connection of own programs to the CAN and CAN FD Interfaces of PEAK-System. PCAN-Basic is the interface between the program and the device driver. In Windows operating systems this is a DLL (Dynamic Link Library) and in Linux operating systems an SO (Dynamic Shared Object). PCAN-Basic is designed to be cross-operating system compatible. Software projects can be ported between supported systems with little effort.

With the installation of the device driver package under Windows the DLL files of the API PCAN-Basic are placed in the system folder. Examples for all common programming languages as well as libraries and help files are available as a download package at: www.peak-system.com/quick/DL-Develop-E

For Linux, a download of the API is available under this link. For a use of PCAN-Basic another driver package with a chardev driver is needed, because an access under SocketCAN is not possible. The "Driver Package for Proprietary Purposes", the user manual, and further information about the implementation can be found at www.peak-system.com/linux

6.1 Features of PCAN-Basic

- API for developing applications with CAN and CAN FD connections
- Support for CAN specifications 2.0 A/B and FD
- Application development for the platforms Windows 11 (x64/ARM64), 10 (x64), and Linux
- Multiple PEAK-System applications and your own can be operated on a physical channel at the same time
- Use of a single DLL for all supported hardware types
- Use of up to 16 channels for each hardware type (depending on the PEAK CAN interface used)
- Simple switching between channels of a PEAK CAN interface
- Access to the CAN channels of a PCAN-Gateway via the new PCAN-LAN hardware type
- Driver-internal buffering of up to 32,768 CAN messages per CAN channel
- Precision of time stamps on received messages up to 1 µs (depending on the PEAK CAN interface used)
- Supports PEAK-System‘s trace formats version 1.1 and 2.0 (for CAN FD applications)
- Access to specific hardware parameters, such as listen-only mode
- Notification of the application through Windows® events when a message is received
- Support of CAN error frames
- Confirmation of physical transmission by CAN echo frames
- Extended system for debugging operations
- Multilingual debugging output
- Output language depends on operating system
- Debugging information can be defined individually
- Thread-safe API

6.2 Principle Description of the API

The sequence of accessing the CAN interface is divided into three phases:

Initialization

A CAN channel must be initialized before using it. This is done by the simple call of the function `CAN_Initialize` for CAN and `CAN_InitializeFD` for CAN FD. Depending on the type of the CAN hardware, up to 16 CAN channels can be opened at the same time. After a successful initialization the CAN channel is ready. No further configuration steps are required.

Interaction

For receiving and transmitting messages the functions `CAN_Read` and `CAN_Write` as well as `CAN_ReadFD` and `CAN_WriteFD` are available depending on the initialization mode. Additional settings can be made, such as setting up message filters to confine to specific CAN IDs or setting the CAN controller to Listen-only mode.

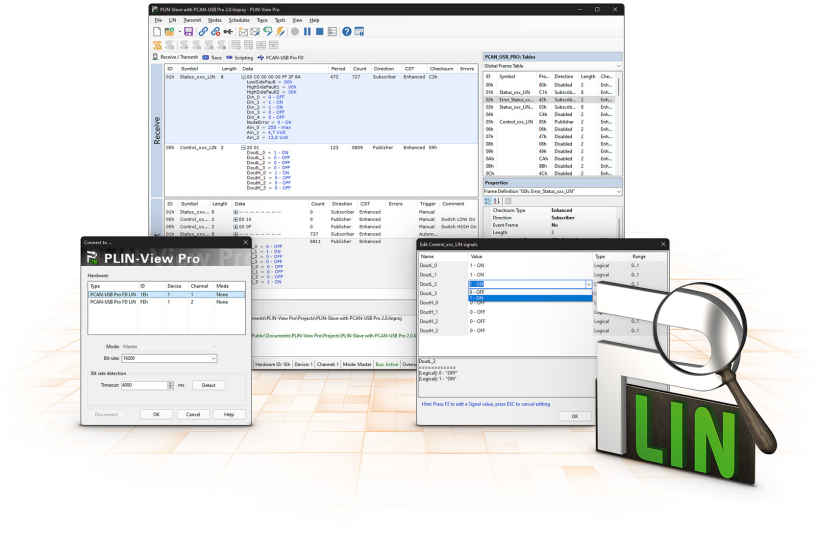
When receiving CAN messages, events are used for an automatic notification of an application (client). This offers the following advantages:

- The application no longer needs to check for received messages periodically (no polling).
- The response time at reception is reduced.

Completion

To end the communication the function `CAN_Uninitialize` is called in order to release the reserved resources for the CAN channel, among others. In addition the CAN channel is marked as "Free" and is available to other applications.

7 LIN Monitor PLIN-View Pro



The LIN monitor PLIN-View Pro is a Windows software for viewing, sending and recording LIN messages. The software is installed ready for operation under Windows with the installation of the device driver package.

In the following the initialization of a LIN interface is described as an example.

Detailed information on the use of PLIN-View Pro can be found in the program window under the menu item *Help*.

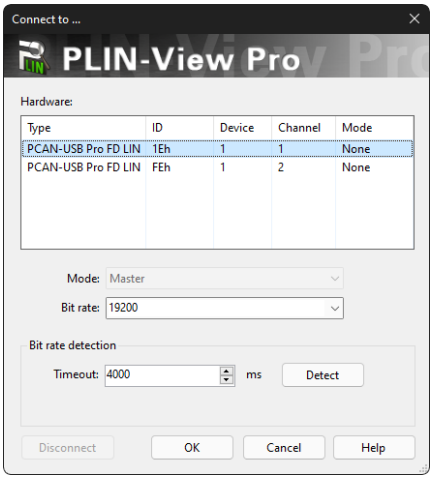
7.1 Features

- Display of incoming LIN frames
- Symbolic display of LIN messages (LDF files)
- Master or Slave mode
- Administration and processing of schedule tables
- Configurable recording of LIN frames (trace)
- Display and recording of frame events such as bus sleep, bus wake-up, and overrun
- Automation of various processes with VBScript
- Automation of LIN data and elements with C# scripts; optional use of C# assemblies
- Integrated text editor for C# with syntax highlighting
- Separate views for:
  - Transmit and Receive
  - Trace (data logger)
  - Scripting
  - Connected LIN interface

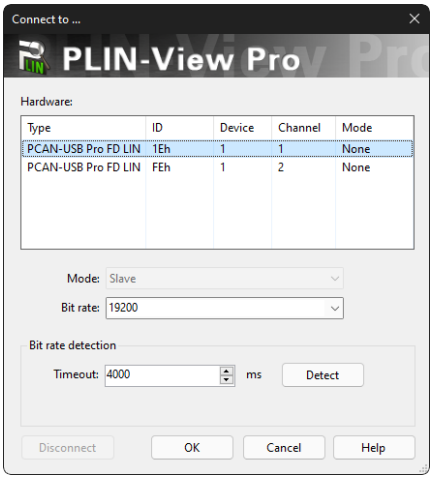
7.2 Start and Initialize PLIN-View Pro

1. From the Windows Start menu, select *PLIN-View Pro*.

   The main window and the *Connect to ...* dialog box for selecting the LIN hardware appear. The parameters for the LIN interface are set in the dialog window.



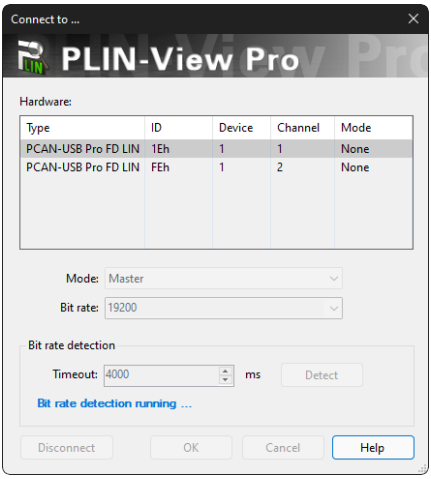
Selection of the hardware as master.



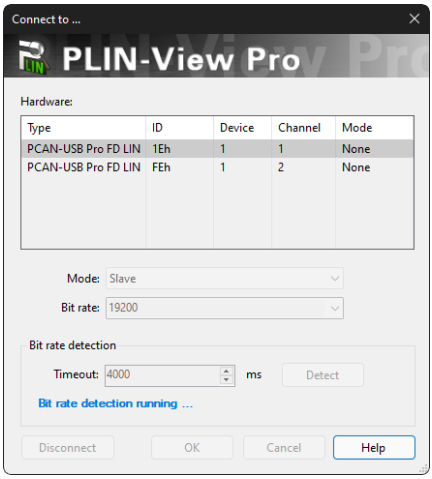
Selection of the hardware as slave.

2. If there are several LIN interfaces, select the desired interface. If there are several channels, select the desired channel from the list.
3. Determine the operation *Mode* to be used for the LIN connection.

4. If the bit rate is known: Select the bit rate of the LIN bus from the *Bit rate* list.
   If the bit rate is unknown: Determine the bit rate under *Bit rate detection* with *Detect*.
   Note: The LIN interface must not be initialized by any other software.



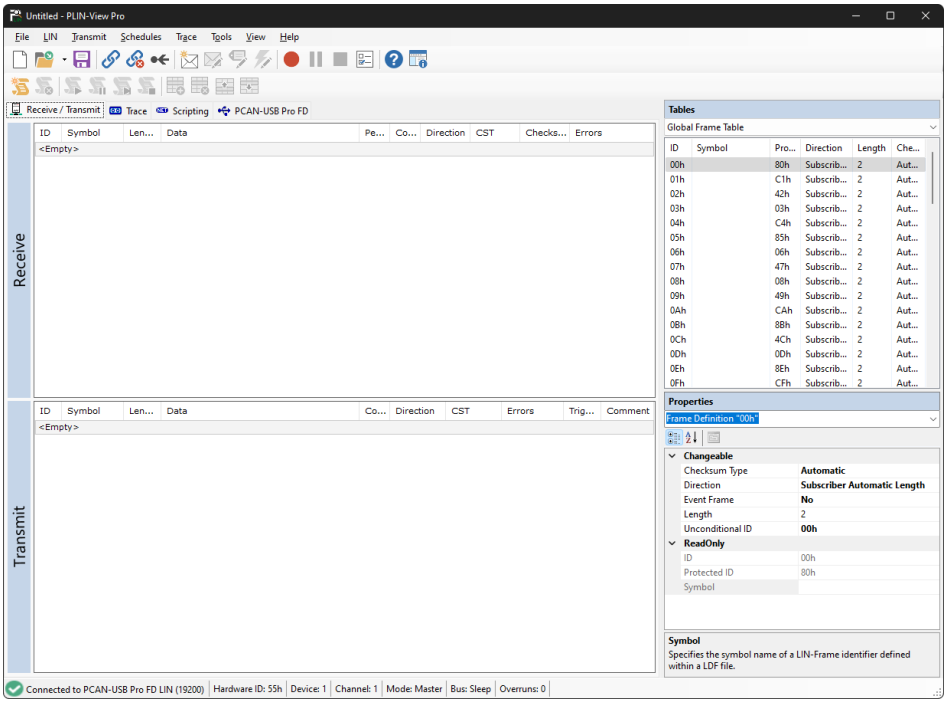
Bit rate detection as master.



Bit rate detection as slave.

5. Confirm the settings with *OK*.
6. Optional: To initialize another channel or LIN interface, open another instance of PLIN-View Pro.

7.3 Receive / Transmit Tab



In the upper area, the *Receive / Transmit* tab displays the Recieve window for received LIN frames. Depending on the operation mode Master or Slave, the lower area shows the *Transmit* window for the operation mode „Master“ or *Publish* for „Slave“. If the master requests data from a slave, the slave can publish the data in the LIN frame.

In the *Global Frame Table* all 64 defined LIN frame entries are stored, which can be processed with the LIN interface (LIN ID 0x00 to 0x3F). To send a LIN frame, the underlying frame definition must be adapted in the *Properties* window.

7.4 Transmit a LIN Frame

Depending on the customer's requirements, different scenarios for sending LIN frames are possible.

7.4.1 With LDF (LIN Description File)

> **Note:** An LDF must be provided by the system manufacturer of the LIN bus or created by the customer.

Application examples with an LDF:

- **Master with scheduler:** The Publisher data is edited in the *Transmit* window and assigned to the *Scheduler* with the Space bar.
- **Slave (Listen Only):** Selecting "All - Listen Only" will receive the data as a silent listener.
- **Slave (LIN node simulation):** The publisher data of the slave are changed in the *Publish* window. With the Space bar the data is sent to the hardware and thus made available on the LIN bus.
- **Master with LIN diagnostic frames**: Diagnostic frames 3C/3D are processed via a script to be created by the customer for the LDF used. Examples can be found in the *Help*.

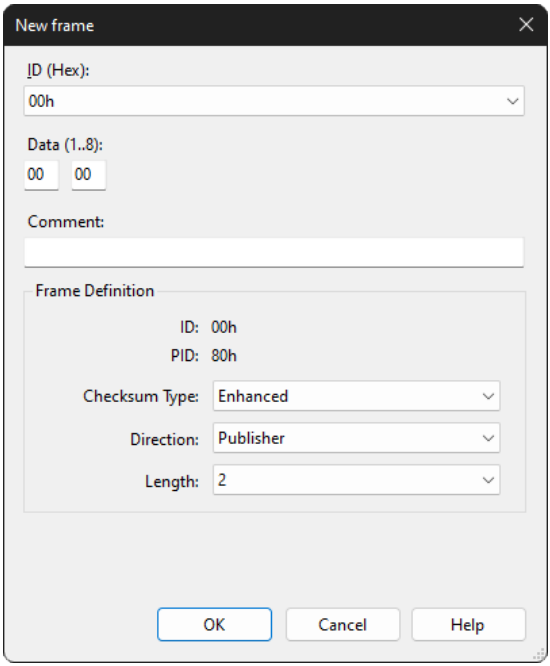
7.4.2 Manually

Manual sending is done according to the connected hardware as master or slave. Frames are configured beforehand for this. For periodic sending, a scheduler can also be created.

> **Note:** In the following example, a frame is sent manually from a master as publisher without a scheduler. For automated sending, further expertise in configuring LIN frames and at least one additional LIN node are required.

1. Connect your LIN interface as master, see chapter *Start and Initialize PLIN-View Pro*.
2. Select the menu command *Transmit* > *New Frame*.

   The *New frame* dialog box appears.



3. Select a frame from the *ID* list.
4. Select "Publisher" for *Direction*.

   The *Data* fields can now be filled.
5. Enter the data of the LIN frame in the *Data* fields.
6. Confirm the entries with *OK*.

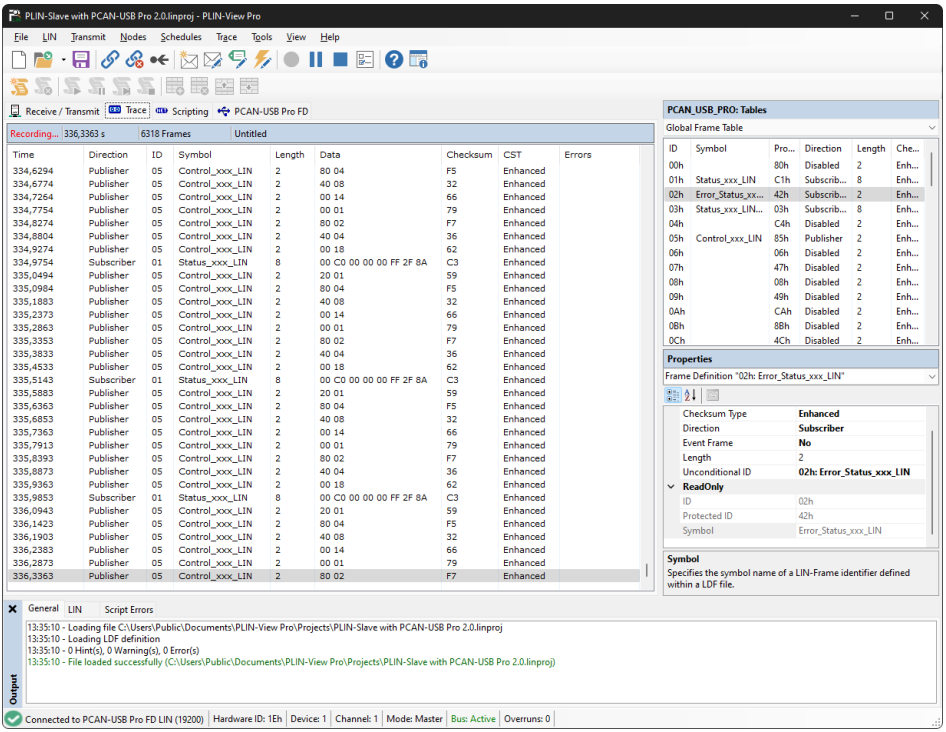
   The configured message appears in the *Transmit* window. „0" is displayed in the *Count* column.
7. Send the selected frame with the menu command *Transmit* > *Send* or with the Space bar.

   The message is sent on the LIN bus and appears in the *Receive* window. „1" for *Transmit* and *Receive* is displayed in the *Count* column.

Change data

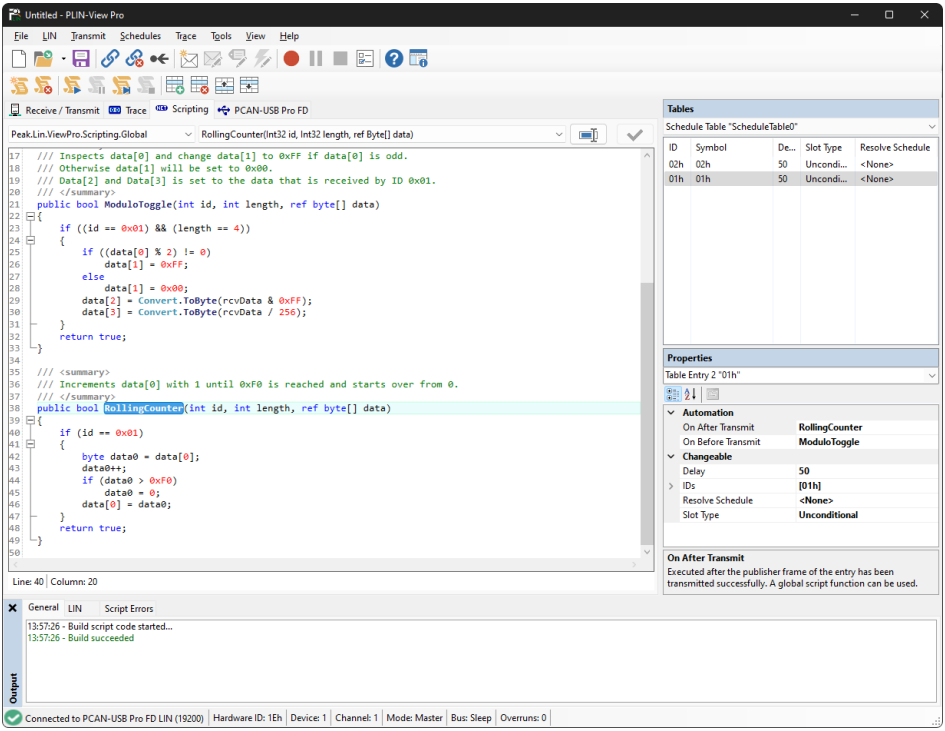
1. Double-click the message in the *Transmit* window.
   The *New frame* dialog box appears again.
2. Change the data and confirm with *OK*.
   The changed data will be displayed in the *Transmit* window.
3. Send the frame again.
   The data in the *Receive* window is updated. The value in the *Count* column is increased by one for *Transmit* and *Receive*.

7.5 Trace Tab



The tracer records all sent and received LIN frames if required. The header displays the current status, the complete runtime and the number of recorded LIN frames. Newly recorded LIN frames are appended to the bottom of the list. Depending on the selected setting, recording is done temporarily or directly to a file.

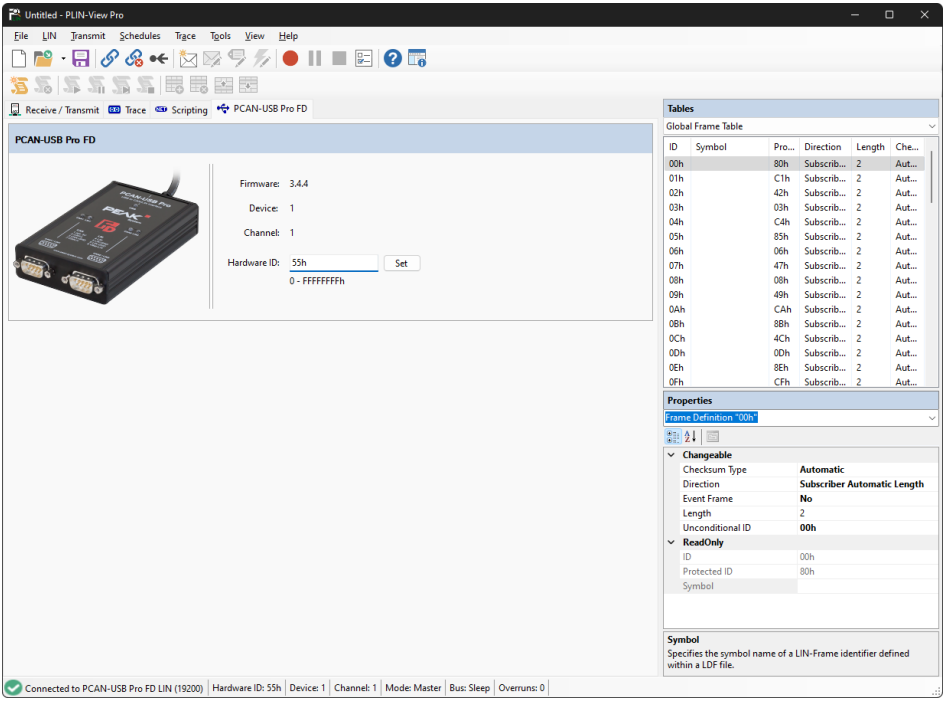
7.6 Scripting Tab



The *Scripting* tab is a text editor with syntax highlighting for the C# programming language. Scripts can be written to automate LIN data and LIN elements. Compiling and deploying a script is done with the check mark in the upper right corner. Feedback, warnings, and errors for the script are displayed in the *Output* section below.

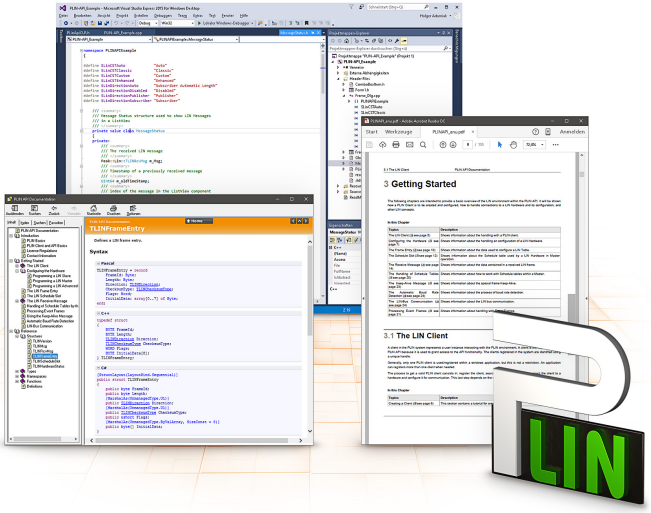
C# assemblies can optionally be included via the menu *Tools* > *Options* > *Tab References* > *Assemblies*. For more details open the *Help* with the key F1.

7.7 LIN Interface Tab



The LIN Interface tab receives the name of the connected hardware and shows information about the hardware and the firmware used. In this example for the interface PCAN-USB Pro FD. To distinguish several LIN interfaces of the same type, the *Hardware ID* of the LIN interface can be determined here.

8 PLIN-API



The intended use of PLIN-API requires compliance with the license rights. Read the license agreement for end users at: https://www.peak-system.com/quick/eula

The programming interface PLIN-API provides basic functions for the connection of own programs to the LIN hardware of PEAK-System. PLIN-API is the interface between the program and the device driver. In Windows operating systems this is a DLL (Dynamic Link Library).

The PLIN-API and examples for all common programming languages as well as libraries and help files are available as download package under www.peak-system.com/quick/DL-Develop-E

8.1 Features

- API for developing applications with LIN connection
- Windows DLLs for the development of x86-, x64-, and ARM64 applications
- Multiple applications can be operated on a physical channel at the same time
- Simple switching between the channels of a PLIN PC hardware
- Internal buffering of messages on software level (system service)
- Precision of time stamps on received messages up to 1 µs
- Allows storing custom data (max. 24 bytes) on the hardware
- Notification of the application through Windows events when a message is received and on plug-in or plug-out of a device
- Function to get error code descriptions in 4 languages

9 Technical Data

USB		
USB mode	High-speed USB 2.0, compatible with USB 1.1 and USB 3.0	
USB connector	Plug type A	

CAN		
Protocols	CAN and CAN FD according to ISO 11898-1; non-ISO CAN FD	
Physical transmission	ISO 11898-2 (High-speed CAN)	
CAN connector	D-Sub (m), 9 pins	
CAN bit rates	Nominal:	25 kbit/s to 1 Mbit/s
CAN FD bit rates	Nominal: Data:	25 kbit/s to 1 Mbit/s 25 kbit/s to 12 Mbit/s
Controller	FPGA implementation	
Transceiver	TJA1044GT	
Time stamp resolution	1 µs	
Galvanic isolation	up to 500 V per CAN connection against USB and LIN	
Power supply external devices	D-Sub pin 1; 5 V, max. 50 mA, disabled at delivery	
Internal Termination	via solder bridges, disabled at delivery	

LIN	
Transceiver power supply	8 to 18 V DC via D-Sub connector
LIN specification	2.2, downward-compatible
LIN connector	D-Sub, 9 pins
Transceiver	TJA1028 (TJA1021/20 up to S/N 24547)
Bit rates	1 to 20 kbit/s
Time stamp resolution	1 µs
Master task resolution	1 ms
Scheduler	Initiated by software, processing by hardware 8 schedule tables with a total of 256 slots configurable
Galvanic isolation	up to 500 V against USB and CAN, no isolation between LIN channels

Power supply	
Rated voltage	5 V DC (via USB connection) LIN operation: 8 - 18 V DC via D-Sub connector
Current consumption	max. 200 mA at 5 V via USB
Current consumption LIN transceiver:	Up to S/N 24547: max. 18 mA at 12 V From S/N 24548: max. 24 mA at 12 V

Measures	
Size	71,5 x 24 x 114 mm (W x H x D without connection cable)
Connection cable length	approx. 1,5 m
Weight	220 g (with connection cable)

Environment	
Operating temperature	-40 to +85 °C (-40 to 185 °F)
Temperature for storage and transport	-40 to +100 °C (-40 to 212 °F)
Relative humidity	15 to 90 %, not condensing
Ingress protection (IEC 60529)	IP20

Conformity	
RoHS	Directive 2011/65/EU (RoHS 2) + 2015/863/EU DIN EN IEC 63000:2019-05
EMC	Directive 2014/30/EU DIN EN 55032:2022-08 DIN EN 55035:2018-04

Appendix A CE Certificate

EU Declaration of Conformity

This declaration applies to the following product:

Product name:	**PCAN-USB Pro FD**
Item number(s):	**IPEH-004061**
Manufacturer:	PEAK-System Technik GmbH Otto-Röhm-Straße 69 64293 Darmstadt Germany

CE We declare under our sole responsibility that the mentioned product is in conformity with the following directives and the affiliated harmonized standards:

EU Directive 2011/65/EU (RoHS 2) + 2015/863/EU (amended list of restricted substances)

DIN EN IEC 63000:2019-05
Technical documentation for the assessment of electrical and electronic products with respect to the restriction of hazardous substances (IEC 63000:2016);
German version of EN IEC 63000:2018

EU Directive 2014/30/EU (Electromagnetic Compatibility)

DIN EN 55032:2022-08
Electromagnetic compatibility of multimedia equipment - Emission requirements (CISPR 32:2015);
German version of EN 55032:2015 + AC:2016 + A11:2020 + A1:2020

DIN EN 55035:2018-04
Electromagnetic compatibility of multimedia equipment - Immunity requirements (CISPR 35:2016, modified);
German version of EN 55035:2017

Darmstadt, 19 January 2023

Uwe Wilhelm, Managing Director

Appendix B UKCA Certificate

UK Declaration of Conformity

This declaration applies to the following product:

Product name: **PCAN-USB Pro FD**

Item number(s): **IPEH-004061**

Manufacturer:
PEAK-System Technik GmbH
Otto-Röhm-Straße 69
64293 Darmstadt
Germany

UK authorized representative:
Control Technologies UK Ltd
Unit 1, Stoke Mill,
Mill Road, Sharnbrook,
Bedfordshire, MK44 1NN, UK

We declare under our sole responsibility that the mentioned product is in conformity with the following UK legislations and the affiliated harmonized standards:

The Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment Regulations 2012

DIN EN IEC 63000:2019-05
Technical documentation for the assessment of electrical and electronic products with respect to the restriction of hazardous substances (IEC 63000:2016);
German version of EN IEC 63000:2018

Electromagnetic Compatibility Regulations 2016

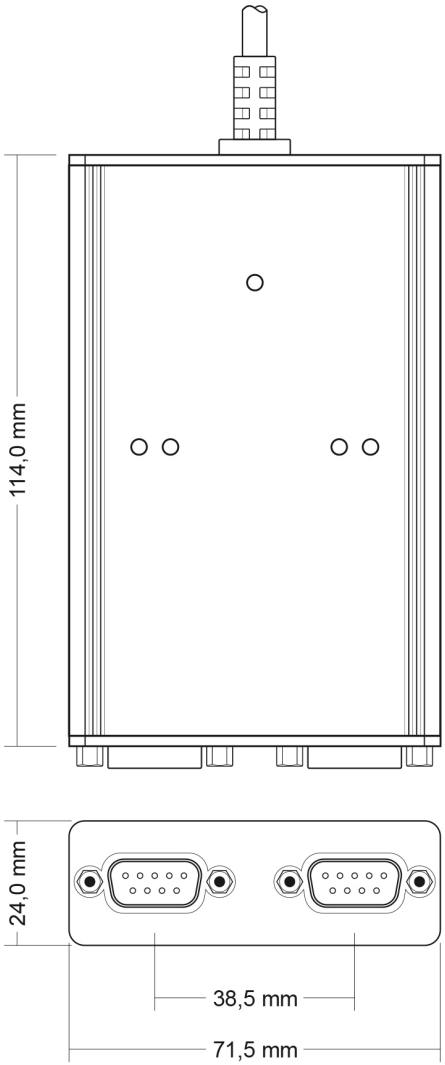
DIN EN 55032:2022-08
Electromagnetic compatibility of multimedia equipment - Emission requirements (CISPR 32:2015);
German version of EN 55032:2015 + AC:2016 + A11:2020 + A1:2020

DIN EN 55035:2018-04
Electromagnetic compatibility of multimedia equipment - Immunity requirements (CISPR 35:2016, modified);
German version of EN 55035:2017

Darmstadt, 19 January 2023

Uwe Wilhelm, Managing Director

Appendix C Dimension Drawings



Appendix D Quick Reference

Software/Hardware Installation under Windows

Download the device drivers installation package from our website www.peak-system.com/quick/DL-Driver-E. Install the driver before you connect the interface.

After driver installation connect the interface to a USB port of the computer. The new hardware is recognized by Windows and the driver is initialized. The LED „USB“ on the interface then lights up green.

Check the operational readiness. Open the Windows Start menu. Type `Peak Settings` and press Enter. The window *PEAK Settings* appears. The connected CAN interface is displayed under *CAN Hardware*. The connected LIN interface is displayed under *LIN Hardware*.

Getting Started under Windows

Run the CAN monitor PCAN-View from the Windows Start menu as a sample application for accessing the CAN interface. For initialization of the CAN interface select the desired CAN channel and CAN bit rate.

Run the LIN monitor PLIN-View Pro from the Windows Start menu as a sample application for accessing the LIN interface. For initialization of the LIN interface select the desired LIN channel, the mode, and LIN bit rate.

USB

Status	Meaning
Green on	High-speed USB is connected (≥USB 2.0)
Green blinking	Communication via High-speed USB
Orange on	Full-speed USB is connected (USB 1.1)
	The PCAN-USB Pro FD interface is in suspend mode (only voltage supply via USB cable, e.g. during computer standby).
Green on	Communication via Full-speed USB

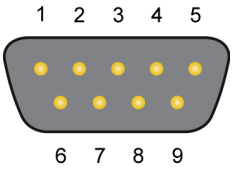
CAN1 and CAN2

Status	Meaning
Green on	The CAN interface is initialized. There's a connection to a driver of the operating system.
Green slow blinking	A software application is connected to the CAN channel.
Green quick blinking	Data is transmitted via the connected CAN bus.
Red quick blinking	An error is occurring during the transmission of CAN data.

LIN1 and LIN2

Status	Meaning
Green on	There's a connection to a driver of the operating system.
Green slow blinking	The LIN channel is initialized with a valid bit rate. A software application is connected to the LIN channel.
Green quick blinking	Data is transmitted via the connected LIN bus.

D-Sub assignment

Pin	Assignment	D-Sub socket
1	CAN_V+ (external, optional)	
2	CAN_Low	
3	CAN_GND	
4	LIN	
5	LIN_GND	
6	LIN_GND	
7	CAN_High	
8	Not assigned	
9	LIN_V_{Bat}	

Appendix E Linux

Depending on the Kernel version, device drivers for the CAN interfaces from PEAK-System are already included in the operating system. The PCAN interfaces are handled as network devices (SocketCAN, netdev). You can find the documentation for SocketCAN under:
https://www.kernel.org/doc/Documentation/networking/can.txt

The following command lists the available drivers:

```
grep PEAK_ /boot/config-`uname -r`
```

Whether the required driver for the PCAN interface is present and loaded can be checked with the following command:

```
lsmod | grep ^peak check
```

If the initialization was successful, the response line starts with `peak_usb` or `peak_pci`.

You can find a recent list of which PCAN interface is supported from which kernel version onwards on our Linux website.

If the required drivers are not listed, install the "Driver Package for Proprietary Purposes". The download and documentation for the driver can also be found on:
www.peak-system.com/linux

This driver package is also needed to use the APIs based on the chardev driver, for example PCAN-Basic, libpcan, or libpcanfd.

Appendix F Disposal

The PCAN-USB Pro FD must not be disposed of in household waste. Dispose of the PCAN-USB Pro FD properly in accordance with local regulations.