Relevant Products

<table>
<thead>
<tr>
<th>Product name</th>
<th>Model</th>
<th>Part number</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCAN-M.2 Single Channel</td>
<td>One CAN channel, galvanic isolation for CAN connection</td>
<td>IPEH-004083</td>
</tr>
<tr>
<td>PCAN-M.2 Dual Channel</td>
<td>Two CAN channels, galvanic isolation for CAN connection</td>
<td>IPEH-004084</td>
</tr>
<tr>
<td>PCAN-M.2 Quad Channel</td>
<td>Four CAN channels, galvanic isolation for CAN connection</td>
<td>IPEH-004085</td>
</tr>
</tbody>
</table>

The cover picture shows the product PCAN-M.2 Dual Channel as format 2280 (left) and as format 2260 (right).

Imprint

PCAN is a registered trademark of PEAK-System Technik GmbH. CiA® is a registered community trademark of CAN in Automation e.V.

All other product names in this document may be the trademarks or registered trademarks of their respective companies. They are not explicitly marked by ™ or ®.

© 2023 PEAK-System Technik GmbH

Duplication (copying, printing, or other forms) and the electronic distribution of this document is only allowed with explicit permission of PEAK-System Technik GmbH. PEAK-System Technik GmbH reserves the right to change technical data without prior announcement. The general business conditions and the regulations of the license agreement apply. All rights are reserved.

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 2.1.0 (2023-09-01)
6 Technical Data ................................................................. 27
Appendix A CE Certificate .................................................. 29
Appendix B UKCA Certificate ............................................. 30
Appendix C Dimension Drawings ....................................... 31
Appendix D Quick Reference ............................................. 32
Appendix E Linux ............................................................. 33
Appendix F Disposal ......................................................... 34
1 Introduction

The PCAN-M.2 allows the connection of CAN and CAN FD networks via the M.2 interface (PCIe) of modern computer boards. With its small format, the plug-in card is ideal for embedded PCs, single-board computers (SBC), and compact embedded applications. There is a galvanic isolation between the computer and the CAN side up to 300 Volts. The card is available as a single, dual, or four-channel version.

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 A/B standard, thus CAN FD nodes can be used in existing CAN networks. However, in this case the CAN FD extensions are not applicable.

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.

This manual describes the use of the CAN interface with Windows. 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 CAN interface.
1.1 Properties at a Glance

- **CAN interface for the M.2 slot** (uses PCIe lane)
- **1, 2, or 4 High-speed CAN channels (ISO 11898-2)**
- **Form factor M.2 type: 2280/2260-B-M; Height: Single and Dual Channel 4.6 mm, Four Channel 10.2 mm; Component height of top side above form factor specification of 1.5 mm**
- **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 20 kbit/s up to 12 Mbit/s**
- **CAN bit rates from 20 kbit/s up to 1 Mbit/s**
- **CAN bus connection via connection cable and D-Sub, 9-pin (in accordance with CiA® 106)**
- **FPGA implementation of the CAN FD controller**
- **Microchip CAN transceiver MCP2558FD**
- **Galvanic isolation on the CAN connection up to 300 V, separate for each CAN channel**
- **CAN termination can be activated through solder jumpers, separately for each CAN channel**
- **PCIe data transfer via bus master DMA**
- **DMA memory access operations with 32- and 64-bit addresses**
- **Measurement of bus load including error frames and overload frames on the physical bus**
- **Induced error generation for incoming and outgoing CAN messages**
- **Extended operating temperature range from -40 to +85 °C (-40 to +185 °F)**

1.2 System Requirements

Computer with:
- **Operating system Windows 11 (x64), 10 (x64), or Linux**
- **M.2 2280 or M.2 2260 slot with PCIe lane**
1.3 Scope of Supply

- Plug-in card PCAN-M.2
- Connection cable, 20 cm, including D-Sub plug for each channel (40 cm or other cable lengths on request)

Downloads

- Device drivers for Windows 11 (x64), 10 (x64), and Linux
- CAN monitor PCAN-View for Windows
- Programming interface PCAN-Basic for developing applications with CAN connection
- Programming interfaces for standardized protocols from the automotive sector
- Manual in PDF format
2  Installation

This chapter covers the software setup for the CAN interface PCAN-M.2 in Windows and the installation of the CAN interface in the computer.

**Note:** For installation on Linux, see Appendix E *Linux.*

Install the driver before you install the CAN interface.

2.1  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. Follow the program instructions.
2.2 Connect the CAN interface

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

CAN ports: on the left Dual Channel, on the right Quad Channel.

Dual Channel with connection cables.

1. Shut down the computer.
2. Disconnect the computer’s power supply.
3. Open the computer case.
4. Insert the CAN interface into an empty M.2 slot.
   - If you do not have an M.2-**2280** slot, you can shorten the CAN interface to **2260** format. To do this, carefully break the back of the CAN interface along the predetermined breaking point.
5. For each CAN channel, mount a D-Sub connector of the connection cable in a suitable recess of the computer housing.
6. For each CAN channel interconnect the SUR socket connector of the connection cable and the SUR header on the CAN interface.
7. Close the computer case.
8. Reconnect the computer power supply.
9. Turn on the computer and start Windows.
   - Windows detects the new hardware and completes the driver installation.
2.3 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.
3 Connecting the CAN Bus

3.1 Connection over D-Sub Connector

After the connection cable has been connected to the CAN interface a CAN bus can be connected to the D-Sub connector. The pin assignment of the D-Sub connector corresponds to the specification CiA® 106:

<table>
<thead>
<tr>
<th>CAN interface</th>
<th>Connection cable</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUR header</td>
<td>SUR socket</td>
<td>D-Sub plug, 9-pin</td>
</tr>
<tr>
<td>Type: SM05B-SURS-TF</td>
<td>Type: 05SUR-32S</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SUR header</th>
<th>SUR socket</th>
<th>D-Sub plug, 9-pin</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 2 3 4 5</td>
<td>5 4 3 2 1</td>
<td>1 2 3 4 5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3, 6</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>1, 2</td>
<td>1, 2</td>
<td>1, 4, 5, 8, 9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CAN_GND</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CAN_High</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CAN_Low</td>
</tr>
<tr>
<td></td>
<td></td>
<td>None</td>
</tr>
</tbody>
</table>

3.2 Daisy Chain

The daisy chain can be activated via solder jumpers on the board to connect a CAN-FD-optimized connection to an existing CAN bus. This makes interference-free operation possible at higher CAN FD bit rates, because stubs and Y distributions are avoided as far as possible.
3.2.1  Connection example and assignment

For cabling, the cables and connectors must be assembled accordingly.

<table>
<thead>
<tr>
<th>D-Sub plug on CAN interface</th>
<th>D-Sub socket on cable</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 2, 3, 4, 5</td>
<td>5, 4, 3, 2, 1</td>
<td>1: CAN_Low Daisy Chain</td>
</tr>
<tr>
<td>6, 7, 8, 9</td>
<td>9, 8, 7, 6</td>
<td>2: CAN_Low</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3, 6: GND</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7: CAN_High</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8: CAN_High Daisy Chain</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4, 5, 9: None</td>
</tr>
</tbody>
</table>

**Note:** The activation of the daisy chain is optional. The manual is available on request from our customer support. Please contact support@peak-system.com.
3.3 Internal Termination

For each CAN channel a termination can be activated via solder bridges on the board. This switches a resistor of 120 Ω between CAN_High and CAN_Low. The conversion is done for an additional improvement of the CAN signal as split termination.

**Note:** The activation of the internal termination is optional. The manual is available on request from our customer support. Please contact support@peak-system.com.

**Tip:** 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.

3.4 Cabling

3.4.1 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).

3.4.2 Example of a Connection

![Diagram of CAN cabling with PCAN-M.2 interfaces and termination resistors IPEK-003001 and IPEK-003002/-mini]
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.

### 3.4.3 Maximum Bus Length

The maximum bus length depends primarily on the bit rate:

<table>
<thead>
<tr>
<th>Nominal bit rate</th>
<th>Buslength</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Mbit/s</td>
<td>40 m</td>
</tr>
<tr>
<td>500 kbit/s</td>
<td>110 m</td>
</tr>
<tr>
<td>250 kbit/s</td>
<td>240 m</td>
</tr>
<tr>
<td>125 kbit/s</td>
<td>500 m</td>
</tr>
<tr>
<td>50 kbit/s</td>
<td>1.3 km</td>
</tr>
<tr>
<td>25 kbit/s</td>
<td>2.5 km</td>
</tr>
<tr>
<td>20 kbit/s</td>
<td>3.3 km</td>
</tr>
</tbody>
</table>

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 apply 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.

### 3.5 Example Application under Windows

As an example application for accessing the CAN interface, run the CAN monitor PCAN-View from the Windows Start menu.
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.
4.1 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.

<table>
<thead>
<tr>
<th>CAN interface</th>
<th>List entry in Available Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>USB Interface, 1-channel</td>
<td>see example above</td>
</tr>
<tr>
<td>USB Interface, 2-channel</td>
<td><img src="#" alt="PCAN-USB Pro FD: Device ID 251181Fh, Channel 1" /></td>
</tr>
<tr>
<td>PCIe Interface, 2-channel</td>
<td><img src="#" alt="PCAN-PCI Express at PCI Bus 1, Device 0, Channel 1" /></td>
</tr>
<tr>
<td></td>
<td><img src="#" alt="PCAN-PCI Express at PCI Bus 1, Device 0, Channel 2" /></td>
</tr>
</tbody>
</table>
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.
4.2 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.

<table>
<thead>
<tr>
<th>CAN</th>
<th>CAN FD</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="New Transmit Message" /></td>
<td><img src="image2.png" alt="New Transmit Message" /></td>
</tr>
</tbody>
</table>

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.
4.3 Additional Tabs

Depending on the CAN interface, additional tabs are available.

4.3.1 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.
4.3.2 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.
4.3.3 Bus Load Tab

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

5.1 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
5.2 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.
## Technical Data

### Connections

<table>
<thead>
<tr>
<th>CAN connection cable</th>
<th>Connection side</th>
<th>Connection type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAN connection cable</td>
<td>CAN bus</td>
<td>D-Sub (m), 9-pin, pin assignment according to CiA® 106</td>
</tr>
<tr>
<td></td>
<td>CAN interface</td>
<td>05SUR-32S (<a href="http://www.jst-mfg.com">www.jst-mfg.com</a>)</td>
</tr>
<tr>
<td>CAN socket on CAN interface</td>
<td>Type: SM05B-SURS-TF</td>
<td></td>
</tr>
<tr>
<td>Computer</td>
<td>M.2 2280(60), B+M key; 59-pin; electromechanical specifications 1.1; uses PCIe lane</td>
<td></td>
</tr>
</tbody>
</table>

### CAN

<table>
<thead>
<tr>
<th>Protocols</th>
<th>CAN and CAN FD according to ISO 11898-1; non-ISO CAN FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical transmission</td>
<td>ISO 11898-2 (High-speed CAN)</td>
</tr>
<tr>
<td>Transceiver</td>
<td>NXP TJA1044GT</td>
</tr>
<tr>
<td>CAN bit rates</td>
<td>Nominal: 20 kbit/s to 1 Mbit/s</td>
</tr>
<tr>
<td>CAN FD bit rates</td>
<td>Nominal: 20 kbit/s to 1 Mbit/s; Data: 20 kbit/s to 12 Mbit/s</td>
</tr>
<tr>
<td>Controller</td>
<td>FPGA implementation</td>
</tr>
<tr>
<td>Time stamp resolution</td>
<td>1 µs</td>
</tr>
<tr>
<td>Galvanic isolation</td>
<td>up to 300 V, separate for each CAN connector</td>
</tr>
<tr>
<td>Internal Termination</td>
<td>Not activated at delivery. Soldering manual on request: <a href="mailto:support@peak-system.com">support@peak-system.com</a></td>
</tr>
<tr>
<td>Daisy Chain</td>
<td>Not activated at delivery. Soldering manual on request: <a href="mailto:support@peak-system.com">support@peak-system.com</a></td>
</tr>
</tbody>
</table>

### Power supply

<table>
<thead>
<tr>
<th>Max. current consumption at 3.3 V pin</th>
<th>Single Channel</th>
<th>450 mA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dual Channel</td>
<td>550 mA</td>
</tr>
<tr>
<td></td>
<td>Quad Channel</td>
<td>750 mA</td>
</tr>
</tbody>
</table>
**Measures**

<table>
<thead>
<tr>
<th></th>
<th>Single Channel</th>
<th>Dual Channel</th>
<th>Quad Channel¹</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size (W x L x H)¹</td>
<td>30 x 51 x 4 mm</td>
<td>30 x 51 x 4 mm</td>
<td>30 x 51 x 12 mm</td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Length connection cable</td>
<td>20 cm (default length), other cable lengths on request</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Single Channel</th>
<th>Dual Channel</th>
<th>Quad Channel</th>
<th>Connection cable</th>
<th>Screw for D-Sub connector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight</td>
<td>6 g</td>
<td>6 g</td>
<td>10 g</td>
<td>7 g (each)</td>
<td>1 g (each)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

¹ Height of the top does not meet the form factor.

**Environment**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating temperature</td>
<td>-40 to +85 °C (-40 to +185 °F)</td>
</tr>
<tr>
<td>Temperature for storage and transport</td>
<td>-40 to +100 °C (-40 to +212 °F)</td>
</tr>
<tr>
<td>Relative humidity</td>
<td>15 to 90 %, not condensing</td>
</tr>
</tbody>
</table>

**Conformity**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RoHS</td>
<td>EU Directive 2011/65/EU (RoHS 2) + 2015/863/EU</td>
</tr>
<tr>
<td></td>
<td>DIN EN IEC 63000:2019-05</td>
</tr>
<tr>
<td>EMC</td>
<td>EU Directive 2014/30/EU</td>
</tr>
<tr>
<td></td>
<td>DIN EN 55032:2022-08</td>
</tr>
<tr>
<td></td>
<td>DIN EN 55035:2018-04</td>
</tr>
</tbody>
</table>
EU Declaration of Conformity

This declaration applies to the following product:
Product name: PCAN-M.2
Item number(s): IPEH-004083, IPEH-004084, IPEH-004085
Manufacturer: PEAK-System Technik GmbH
Otto-Röhm-Straße 69
64293 Darmstadt
Germany

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);

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

Darmstadt, 25 August 2023

Uwe Wilhelm, Managing Director
UK Declaration of Conformity

This declaration applies to the following product:

Product name: PCAN-M.2
Item number(s): IPEH-004083, IPEH-004084, IPEH-004085

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);

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

Darmstadt, 25 August 2023

Uwe Wilhelm, Managing Director
Appendix C  Dimension Drawings

Dimensions in mm of PCAN-M.2 Single Channel and Dual Channel.

Dimensions in mm of PCAN-M.2 Quad Channel.

* shortened length
** Formfactor not fulfilled on upper side
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 install the CAN interface.

Turn off the computer and insert the PCAN-M.2 into an available M.2 slot (using PCIe lane).

If you do not have an M.2-2280 slot, you can shorten the card to 2260 format. To do this, carefully break the back of the card along the predetermined breaking point.

The new hardware is detected at the next Windows start and the driver is initialized. 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.

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.

Pin assignment

<table>
<thead>
<tr>
<th>CAN interface</th>
<th>Connection cable</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUR header</td>
<td>SUR socket</td>
<td>D-Sub plug, 9-pin</td>
</tr>
<tr>
<td>Type: SM05B-SURS-TF</td>
<td>Type: 05SUR-32S</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CAN channel</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3, 6</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>1, 2</td>
<td>1, 4, 5, 8, 9</td>
</tr>
</tbody>
</table>

None
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_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 product must not be disposed of in household waste. Dispose of the product properly in accordance with local regulations.