

PCAN-RS-232

User Manual



Relevant Product

Product Name	Model	Part Number
PCAN-RS-232	Plastic casing, screw terminal block	IPEH-002100

Imprint

PCAN is a registered trademark of PEAK-System Technik GmbH.

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 2.1.2 (2023-08-02)

Contents

Imprint	2
Relevant Product	2
Contents	3
1 Introduction	4
1.1 Properties at a Glance	5
1.2 Scope of Supply	6
1.3 Prerequisites for Operation	6
2 Connectors and Coding Solder Jumpers	7
2.1 Screw Terminal Block	9
2.2 J5 Connector Panel: JTAG Ports	10
2.3 Coding Solder Jumpers	13
3 Operation	15
4 Creating Own Firmware	16
4.1 Library	18
5 Firmware Upload	19
5.1 Uploading Firmware via CAN	19
5.2 Uploading Firmware via the Serial Connections	24
6 Technical Data	26
Appendix A CE Certificate	29
Appendix B UKCA Certificate	30
Appendix C Dimension Drawing	31
Appendix D Port Assignment of the Microcontroller	32
Appendix E Disposal	35

1 Introduction

The PCAN-RS-232 is a programmable module for the communication between RS-232 and CAN. The conversion of data traffic is done via a NXP LPC21 series microcontroller.

The behavior of the PCAN-RS-232 can be programmed freely for specific applications. The firmware is created using the included development package with GNU compiler for C and C++ and is then transferred to the module via CAN. Various programming examples facilitate the implementation of own solutions.

On delivery, the PCAN-RS-232 is provided with a standard firmware that routes from CAN to RS-232 and vice versa. It allows to configure the data transfer as well as the hardware with serial control commands. The corresponding source code is included as an example in the scope of supply.

1.1 Properties at a Glance

- NXP LPC21 series microcontroller (16/32-bi ARM CPU)
- 32 kbyte EEPROM
- High-speed CAN channel (ISO 11898-2) with bit rates from 40 kbit/s up to 1 Mbit/s
- Complies with CAN specifications 2.0 A/B
- Data transfer between CAN and RS-232 with a maximum bit rate of 115,200 bit/s
- One digital input and one digital output (low-active)
- 2-color LED for status signaling
- Connection via a 10-pole terminal strip (Phoenix)
- Voltage supply from 8 to 30 V
- Extended operating temperature range from -40 to +85 °C (-40 to +185 °F)
- New firmware can be loaded via CAN interface

1.2 Scope of Supply

- PCAN-RS-232
 - in plastic casing
 - with mating connector: Phoenix Contact MC 1.5/10-ST-3.5 - 1840447

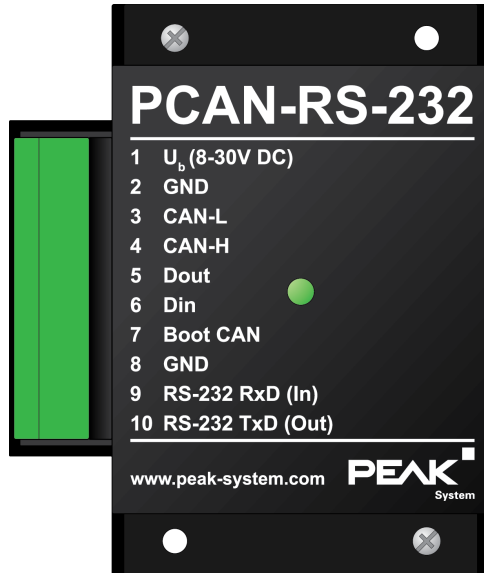
Download

- Windows development package with:
 - GCC ARM Embedded
 - Flash program
 - Programming examples
- Manual in PDF format

1.3 Prerequisites for Operation

- Power supply in the range of 8 to 30 V DC
- For uploading the firmware via CAN:
 - CAN interface of the PCAN series for the computer (e.g. PCAN-USB)
 - Operating system Windows 11 (x64/ARM64), 10 (x86/x64)

2 Connectors and Coding Solder Jumpers



PCAN-RS-232 with 10 connector poles and one status LED

The PCAN-RS-232 converter has a 10-pole screw terminal block for connecting the following components:

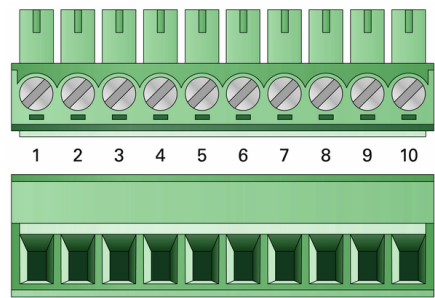
- Supply voltage
- CAN
- RS-232
- Digital input and digital output
- CAN bootloader activation

For direct access to the debugging ports (JTAG) of the microcontroller, an additional – yet not equipped – connector panel is available on the circuit board of the converter.

Furthermore the board has four coding solder jumpers in order to assign a fixed status to the corresponding input bits of the microcontroller. A concrete application is to identify a PCAN-RS-232 converter on the CAN bus for a firmware upload, especially if there are several converters connected and in operation.

The following subsections describe each connector assignment.

2.1 Screw Terminal Block



Screw terminal block with 3.5 mm pitch
(Mating connector Phoenix Contact MC 1.5/10-ST-3.5 - 1840447)

Terminal	Identifier	Function
1	+V _b	Power supply 8 to 30 V DC
2	GND	Ground
3	CAN_L	Differential CAN signal
4	CAN_H	
5	DOut	Digital output, Low-side switch
6	DIn	Digital input, Low-active
7	Boot CAN	CAN bootloader activation, High-active
8	GND	Ground
9	RS-232 RxD	RS-232 interface
10	RS-232 TxD	

For further connection details that are not needed for programming of the PCAN-RS-232 converter because of implementation in a library, see also Appendix D *Port Assignment of the Microcontroller*.

2.2 J5 Connector Panel: JTAG Ports

The unpopulated connector panel J5 on the PCAN-RS-232 converter's circuit board provides an access option to the JTAG ports of the LPC2194/01 microcontroller (μC) for hardware debugging.

Gain access to the unpopulated J5 connector panel:



Risk of short circuit! Soldering on the PCAN-RS-232 may only be performed by qualified electrical engineering personnel.

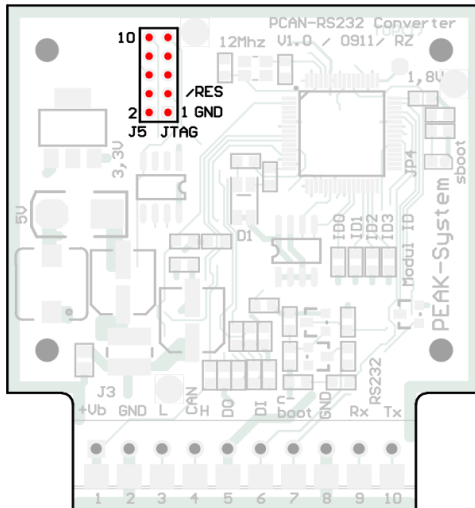


Attention! Electrostatic discharge (ESD) can damage or destroy components on the card. Take precautions to avoid ESD.

1. Disconnect the PCAN-RS-232 from the power supply.
2. Unscrew the two screws.
3. Remove the housing cover.
4. Remove the circuit board.

5. Make your desired settings.

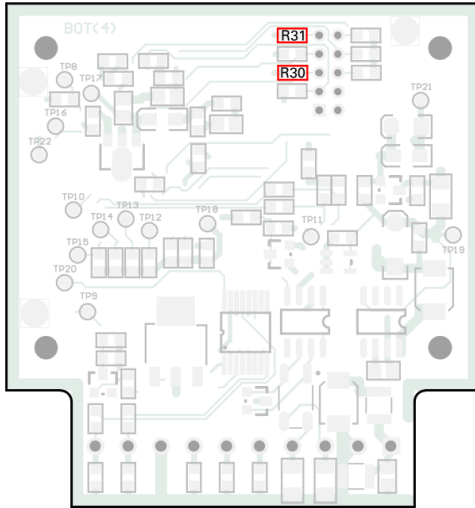
The following figure shows the positions of the JTAG panel (not equipped) on the top of board. The table below the figure contains information of the microcontroller and internal wiring.



JTAG panel on the circuit board (not equipped)

Pin	Signal	Port μ C	Internal Wiring
1, 2	GND		
3	/Reset	/Reset	Pull-up
4	3.3 V		
5	TCK	P1.29	Pull-down (R30)
6	TMS	P1.30	Pull-up
7	TDO	P1.27	Pull-up
8	TDI	P1.28	Pull-up
9	RTCK	P1.26	Pull-down (R31)
10	TRST	P1.31	Pull-up

The following figure shows the positions for soldering out the pull-down resistors on the bottom of the board. You can do that if the constant internal pull-down wiring of TCK or RTCK signals is not suitable for your purposes.



Pull-down resistors on the circuit board bottom:
R30 for pin 5 TCK, R31 for pin 9 RTCK

6. Insert the circuit board and place the housing cover on top.
7. Screw the two screws back into their original positions.

2.3 Coding Solder Jumpers

The board has four coding solder jumpers to assign a permanent state to the corresponding input bits of the microcontroller. The four positions for coding solder bridges (ID 0 - 3) are each assigned to one port of the microcontroller LPC2194/01 (µC). A bit is set (1) if the corresponding solder field is open.

A concrete application is the identification of a PCAN-RS-232 on the CAN bus during a firmware upload, especially if several devices are connected and in operation.

The status of the ports is relevant in the following cases:

- The loaded firmware is programmed so that it reads the status at the corresponding ports of the microcontroller. For example, the activation of certain functions of the firmware or the coding of an ID is conceivable here.
- For a firmware upload via CAN the PCAN-RS-232 converter is identified by a 4-bit ID which is determined by solder jumpers. A bit is set (1) when the corresponding solder jumper position is open (default setting: ID 15, all positions open).

Solder field	ID0	ID1	ID2	ID3
Binary Digit	0001	0010	0100	1000
Decimal Equivalent	1	2	4	8

See section 5.1 *Uploading Firmware via CAN* for more information.

Activate coding solder bridges:

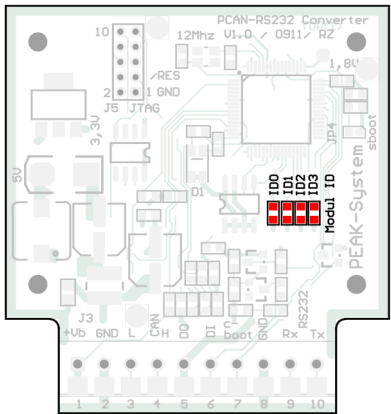


Risk of short circuit! Soldering on the PCAN-RS-232 may only be performed by qualified electrical engineering personnel.





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

1. Disconnect the PCAN-RS-232 from the power supply.
2. Unscrew the two screws.
3. Remove the housing cover.
4. Remove the circuit board.
5. Solder the solder bridge(s) on the board according to the desired setting.



Coding solder jumpers on the circuit board

Position	0	1	2	3
Port µC	P0.4	P0.5	P0.6	P0.7

Solder field status	Port status
	Low
	High

6. Insert the circuit board and place the housing cover on top.
7. Screw the two screws back into their original positions.

3 Operation

The PCAN-RS-232 converter is activated by applying the supply voltage to the respective input pins. More information can be found in chapter 2 *Connectors and Coding Solder Jumpers*. The firmware in the flash memory is subsequently run.

The status indication of the LED depends on the used firmware.

On delivery the PCAN-RS-232 is supplied with a standard firmware that routes from CAN to RS-232 and vice versa. It allows to configure the data transfer as well as the hardware with serial control commands.

The LED is green when the converter is switched on, and blinks during an established connection to a serial host.

The documentation for the standard firmware can be found in the development package in the following directory branch:

Hardware\PCAN-RS-232\Examples\06_CAN_TO_SER_BY_COMMAND\help

The development package can be downloaded via the following link:

www.peak-system.com/quick/DLP-DevPack

4 Creating Own Firmware

With the help of the PEAK-DevPack development package, you can program your own application-specific firmware for PEAK-System programmable hardware products. For each supported product, examples are included.

On delivery the PCAN-RS-232 converter is supplied with the standard firmware `6_CAN_TO_SER_BY_COMMAND` that routes from CAN to RS-232 and vice versa. It allows to configure the data transfer as well as the hardware with serial control commands.

System requirements:

- Computer with operating system Windows 11 (x64), 10 (x86/x64)
- CAN interface of the PCAN series to upload the firmware to your hardware via CAN

Download of the development package:

www.peak-system.com/quick/DLP-DevPack

Content of the package:

- `Build Tools Win32\`
Tools for automating the build process for Windows 32-bit
- `Build Tools Win64\`
Tools for automating the build process for Windows 64-bit
- `Compiler\`
Compilers for the supported programmable products
- `Debug\`
 - OpenOCD and configuration files for hardware which supports debugging
 - VBScript `SetDebug_for_VSCode.vbs` to modify the example directories for the Visual Studio Code IDE with Cortex-debug
 - Detailed information about debugging in the enclosed documentation of the PEAK-DevPack Debug Adapter

- `Hardware\`
Sub directories with firmware examples for supported hardware. Use the examples for starting your own firmware development.
- `PEAK-Flash\`
Windows software for uploading the firmware to your hardware via CAN
- `LiesMich.txt` and `ReadMe.txt`
Short documentation how to work with the development package in German and English
- `SetPath_for_VSCode.vbs`
VBScript to modify the example directories for the Visual Studio Code IDE

Creating your own firmware:

1. Create a folder on your computer. We recommend using a local drive.
2. Unzip the development package `PEAK-DevPack.zip` completely into the folder. No installation is required.
3. Run the script `SetPath_for_VSCode.vbs`. This script will modify the example directories for the Visual Studio Code IDE. Afterwards, each example directory has a folder called `.vscode` containing the needed files with your local path information.
4. Launch Visual Studio Code. The IDE is available free of charge from Microsoft: <https://code.visualstudio.com>.
5. Select the folder of your project and open it. For example:
`d:\PEAK-DevPack\Hardware\PCAN-RS-232\Examples\01_CAN_ECHO`
 You can edit the C code and use the the menu *Terminal > Run Task* to call *make clean*, *make all*, or to compile a single file.
6. Create your firmware with *make all*. The firmware is the `*.bin` in the `out` subdirectory of your project folder.
7. Prepare your hardware for firmware upload like described in section 5.1.2 *Preparing Hardware*.

8. Use the PEAK-Flash tool to upload your firmware to the PCAN-RS-232 via CAN. The tool is either started via the menu *Terminal > Run Task > Flash Device* or from the subdirectory of the development package. Chapter 5.1.3 *Firmware Transfer* describes the process. A CAN interface of the PCAN series is required.

4.1 Library

The development of applications for the PCAN-RS-232 is supported by the library `libPCAN-RS-232-GNU*s.a` (* stands for version number), a binary file. You can access all resources of the PCAN-RS-232 by means of this library. The library is documented in the header files (*.h) which are located in the `inc` subdirectory of each example directory.

5 Firmware Upload

The microcontroller in the PCAN-RS-232 can be equipped with new firmware in two different ways:

- **Via CAN (recommended):**

Via a CAN channel and with the Windows software PEAK-Flash, the firmware can be transferred from the computer to the PCAN-RS-232.

More information can be found in section 5.1 *Uploading Firmware via CAN*.

- **Via serial connection:**

This requires access to the circuit board of the PCAN-RS-232.

More information can be found in section 5.2 *Uploading Firmware via the Serial Connections*.

5.1 Uploading Firmware via CAN

5.1.1 System Requirements

- CAN interface of the PCAN series for the computer, for example PCAN-USB
- CAN cabling between the CAN interface and the PCAN-RS-232 converter with correct termination at both ends of the CAN bus with 120 Ohm each.
- Operating system Windows 11 (x64/ARM64), 10 (x86/x64)
- If you want to update several PCAN-RS-232 converters on the same CAN bus with new firmware, you must assign an ID to each converter. See section 2.3 *Coding Solder Jumpers*.

5.1.2 Preparing Hardware

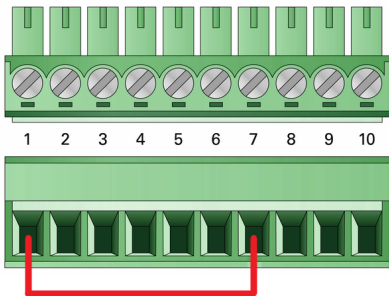
For an upload of new firmware via CAN, the CAN bootloader must be activated in the PCAN-RS-232.

Activating CAN Bootloader:



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

1. Disconnect the PCAN-RS-232 from the power supply.
2. Establish a connection between **Boot** and the power supply U_b .



Connection at the screw terminal block between terminals 1 and 7

Because of that, a High level is later applied to the **Boot** connection.

3. Connect the CAN bus of the converter with a CAN interface connected to the computer. Pay attention to the proper termination of the CAN cabling (2 x 120 Ohm).
4. Reconnect the power supply.

Due to the High level at the **Boot** connection, the PCAN-RS-232 starts the CAN bootloader. This can be determined by orange quickly blinking of the status LED.

5.1.3 Firmware Transfer

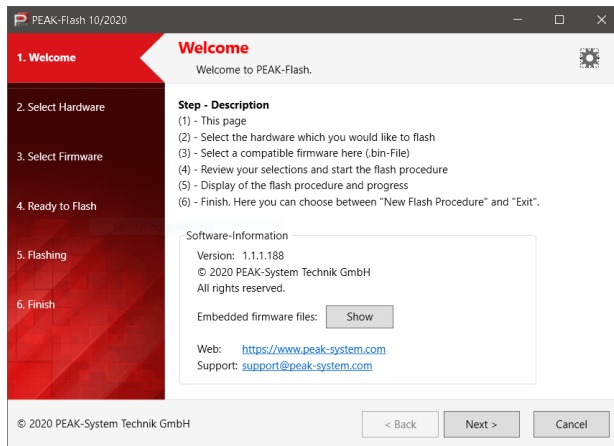
A new firmware version can be transferred to the PCAN-RS-232. The firmware is uploaded via a CAN bus using the Windows software PEAK-Flash.

Transfer firmware with PEAK-Flash:

The software PEAK-Flash is included in the development package, which can be downloaded via the following link: www.peak-system.com/quick/DLP-DevPack

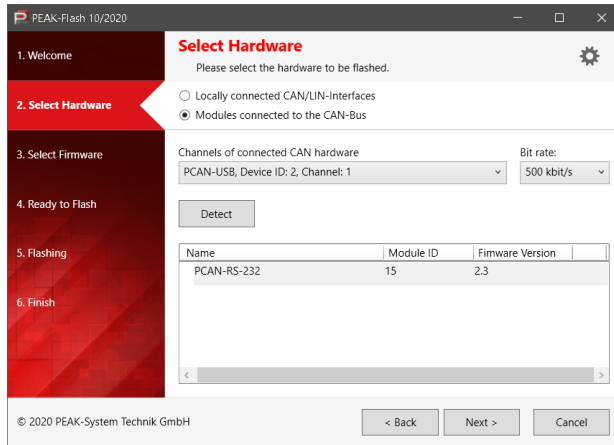
1. Open the zip file and extract it to your local storage medium.
2. Run the `PEAK-Flash.exe`.

The main window of PEAK-Flash appears.



3. Click the button *Next*.

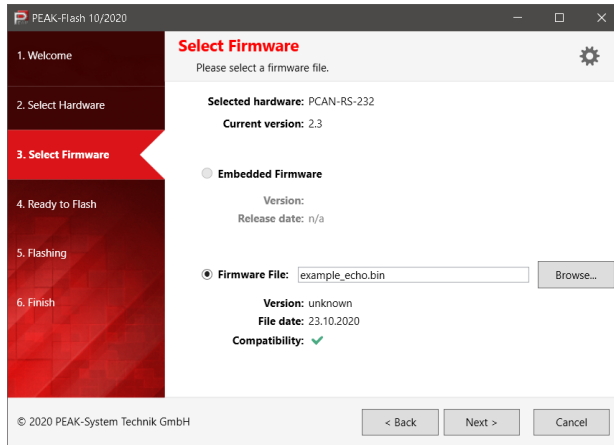
The *Select Hardware* window appears.



4. Click on the *Modules connected to the CAN bus* radio button.
 5. In the drop-down menu *Channels of connected CAN hardware*, select a CAN interface connected to the computer.
 6. In the drop-down menu *Bit rate*, select the nominal bit rate 500 kbit/s.
 7. Click on *Detect*.
- In the list, the PCAN-RS-232 appears together with the Module ID and Firmware version. If not, check whether a proper connection to the CAN bus with the appropriate nominal bit rate exists.

8. Click *Next*.

The *Select Firmware* window appears.



9. Select the *Firmware File* radio button and click *Browse*.

10. Select the corresponding file (*.bin).

11. Click *Next*.

The *Ready to Flash* dialog appears.

12. Click *Start* to transfer the new firmware to the PCAN-RS-232.

The *Flashing* dialog appears.

13. After the process is complete, click *Next*.

14. You can exit the program.

15. Disconnect the PCAN-RS-232 from the power supply.

16. Remove the connection between **Boot** and the power supply **U_b**.

17. Connect the PCAN-RS-232 to the power supply.

You can now use the PCAN-RS-232 with the new firmware.

5.2 Uploading Firmware via the Serial Connections

This section shows how to activate the microcontroller's bootloader. The actual upload process depends on the upload software used which is supplied by a third party and is not described here.



Important: When uploading a firmware via the RS-232 interface, the CAN bootloader may be overwritten. Afterwards, a firmware upload via CAN is not possible anymore.

Activate the microcontroller's bootloader:



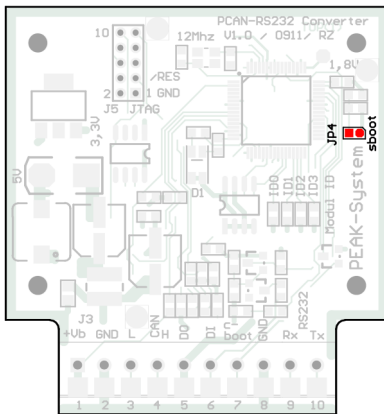
Risk of short circuit! Soldering on the PCAN-RS-232 may only be performed by qualified electrical engineering personnel.





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

1. Disconnect the PCAN-RS-232 from the power supply.
2. Unscrew the two screws.
3. Remove the housing cover.
4. Remove the circuit board.

5. Solder the solder bridge JP4 on the board according to the desired setting.



Activation of the bootloader by jumper JP4 on the circuit board

Solder Field Status	Port Status
	Default: Bootloader is not activated.
	Bootloader is activated.

6. Establish a serial connection to the computer or the programming adapter via the RS-232 terminals **RxD (9)** and **TxD (10)**.
7. Reconnect the power supply.
Due to the low level at port **P0.14** of the microcontroller the converter starts the bootloader for the serial transmission. The LED remains off.
8. Insert the circuit board and place the housing cover on top.
9. Screw the two screws back into their original positions.

6 Technical Data

Connectors

Screw terminal block	10-pin, pitch 3.5 mm Phoenix Contact MC 1.5/10-ST-3.5 - 1840447
----------------------	--

Digital Input (Din)

Quantity	1
Type	Low-active
Switching thresholds	Low → High: $U > 1.2 \text{ V}$ High → Low: $U < 0.8 \text{ V}$
Maximum level	U_b

Digital Output (Dout)

Quantity	1
Type	Low-side driver
Maximum voltage at load	60 V
Maximum output current	0.7 A

Power Supply

Supply voltage (U_b)	8 to 30 V DC
Current consumption	max. 70 mA at 12 V

CAN

Specification	ISO 11898-2, High-Speed-CAN 2.0A/B
Bit rates	40 kbit/s to 1 Mbit/s (lower bit rates on request)
Transceiver	NXP TJA1040T
Internal termination	None
Listen-only mode	Programmable; not activated at delivery

Microcontroller

CPU	NXP LPC2194/01
Clock frequency	60 MHz
Add-on memory	32 kByte, EEPROM Atmel AT24C256B (via I ² C)
Firmware upload	via CAN (PCAN interface required) via serial connection

Measures

Size	Casing: 68 x 57 x 25.5 mm (W x D x H) Circuit board: 51 x 54 mm (W x D)
Weight	36 g

Environment

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

Conformity

RoHS 2	EU Directive 2011/65/EU (RoHS 2) + 2015/863/EU DIN EN IEC 63000:2019-05
EMV	EU Directive 2014/30/EU DIN EN 61326-1:2022-11

Appendix A CE Certificate

EU Declaration of Conformity



This declaration applies to the following product:

Product name: **PCAN-RS-232**
Item number(s): **IPEH-002100**
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 61326-1:2022-11

Electrical equipment for measurement, control and laboratory use - EMC requirements - Part 1:
General requirements (IEC 61326-1:2020);
German version of EN IEC 61326-1:2021

Darmstadt, 19 January 2023

A handwritten signature in black ink, appearing to read "Uwe Wilhelm".

Uwe Wilhelm, Managing Director

Appendix B UKCA Certificate

UK Declaration of Conformity



This declaration applies to the following product:

Product name: **PCAN-RS-232**

Item number(s): **IPEH-002100**

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 61326-1:2022-11

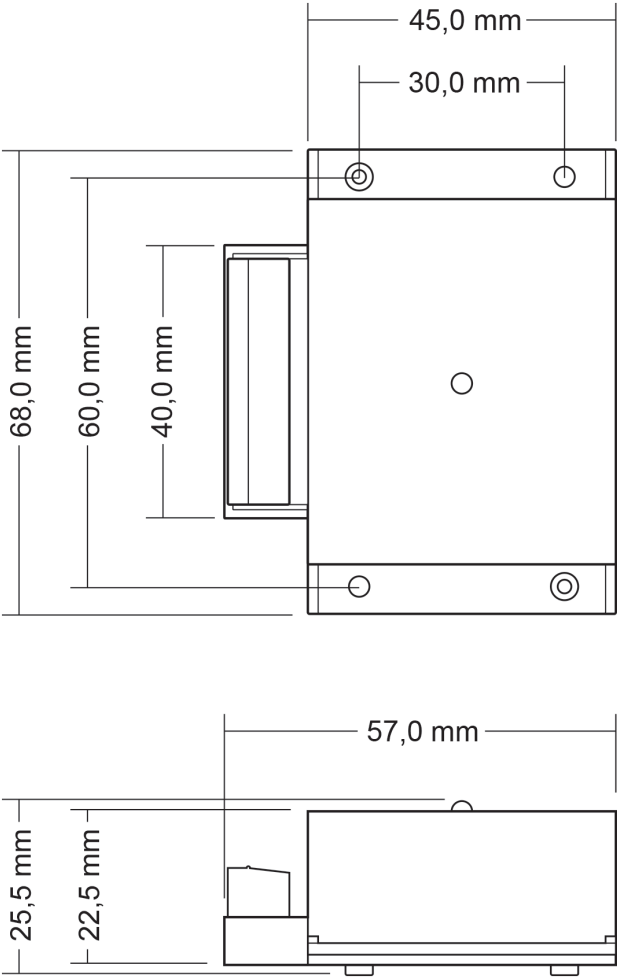
Electrical equipment for measurement, control and laboratory use - EMC requirements - Part 1: General requirements (IEC 61326-1:2020);
German version of EN IEC 61326-1:2021

Darmstadt, 19 January 2023

A handwritten signature in black ink, appearing to read "Uwe Wilhelm".

Uwe Wilhelm, Managing Director

Appendix C Dimension Drawing



Appendix D Port Assignment of the Microcontroller

The following table lists the used inputs and outputs (ports) of the LPC2194/01 microcontroller (µC) and their function in the PCAN-RS-232 converter. It is meant as supplemental information. The converter's functionality is implemented by the supplied library.

Get additional information about the LPC2194/01 microcontroller on the homepage of NXP (www.nxp.com).

Port	I/O	µC Function	Signal	Active (µC)	Function / Connection ¹
P0.0	O	TxD UART0	TxD0		Serial communication, Transmit, STB:10 (RS-232 levels)
P0.1	I	RxD UART0	RxD0		Serial communication, Receive, STB:9 (RS-232 levels)
P0.2	I, O	SCL	SCL		I ² C bus to the Atmel AT24C256B EEPROM
P0.3	I, O	SDA	SDA		
P0.4	I	Port pin	ID0	High	Coding solder jumpers on board (ID 0 - 3), bridged = Low
P0.5	I	Port pin	ID1	High	
P0.6	I	Port pin	ID2	High	
P0.7	I	Port pin	ID4	High	
P0.12	O	Port pin			Reserved
P0.13	I, O	Port pin			
P0.14	I	Port pin	/Boot_ser	Low	Activate flashing via serial interface, JP4

¹ SKL:n Terminal n on the screw terminal block
 J4/5:n Pin n of the respective connector panel on the circuit board
 JPx Jumper position on the circuit board (set = active)

Port	I/O	µC Function	Signal	Active (µC)	Function / Connection ¹
P0.15	I	Port pin	/Boot_CAN	Low	Activate flashing via CAN with 500 kbit/s, STB:7 (due to wiring High-active)
P0.17	O	Port pin	V24_en	High	Deactivate the RS-232 component by Low level (activated by default); possibility for energy saving
P0.19	I	Port pin	Switch	High	Digital input Din, STB:6 (due to wiring Low-active)
P0.22	O	Port pin	CAN_en_1	Low	Activate the CAN transceiver ²
P0.25	I	RD1	CAN_RxD		CAN receive
TD1	O	TD1	CAN_TxD		CAN transmit
P0.28	I	Analog input	V-Power1		Measure voltage U _b , maximum value (0x03FF) corresponds to 33.1 V
P0.29	I	Analog input			Lies on GND
P0.30	I	Analog input			Lies on 1.8 V (microcontroller supply)
P1.16	O ³	Port pin		Low	LED red
P1.17	O ³	Port pin		Low	LED green
P1.21	O	Port pin		Low	Digital output Dout, STB:5 (Low-side switch)

² After resetting the microcontroller, the CAN transceiver is deactivated and must be reactivated to use it.

³ It may occur that the LED glows slightly when the output is inactive. If you would like to prevent this, your firmware must change the port type to input (I). Before switching on the LED again, the respective port type must be set to output (O).

Port	I/O	µC Function	Signal	Active (µC)	Function / Connection ¹
P1.26		JTAG Interface	RTCK		Debugging, J5:9
P1.27		JTAG Interface	TDO		Debugging, J5:7
P1.28		JTAG Interface	TDI		Debugging, J5:8
P1.29		JTAG Interface	TCK		Debugging, J5:5
P1.30		JTAG Interface	TMS		Debugging, J5:6
P1.31		JTAG Interface	TRST		Debugging, J5:10

Appendix E Disposal

The PCAN-RS-232 must not be disposed of in household waste. Dispose of the PCAN-RS-232 properly in accordance with local regulations.