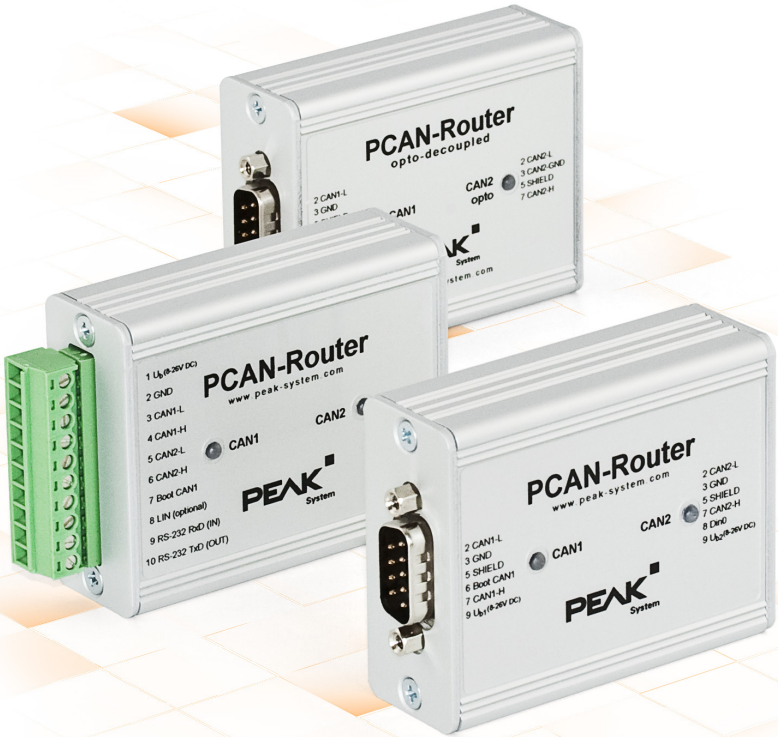
PCAN-Router

User Manual



Relevant Products

Product Name	Model	Part Number
PCAN-Router	2 D-Sub connectors, additional digital input	IPEH-002210 from SN 01000
PCAN-Router	Screw terminal block (Phoenix), additional serial interface	IPEH-002210-P from SN 01000
PCAN-Router opto-decoupled	2 D-Sub connectors, galvanic isolation for connector CAN2, additional digital input	IPEH-002211 from SN 00020

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 3.0.0 (2023-02-07)

Contents

1 Introduction

The PCAN-Router is a dual-channel CAN module whose NXP LPC21 series programmable microcontroller provides the option of using the CAN messages on both channels on a flexible basis. This gives a whole range of options for manipulation, evaluation, filtering, and routing of CAN messages.

Using the programming library and the GNU compiler for C and C++, a firmware is created and then transferred to the module via CAN. At delivery, the PCAN-Router is equipped with a demo firmware that forwards CAN messages 1:1 between both channels at 500 kbit/s. The corresponding source code is included as example in the scope of supply.

The module is installed in an aluminum profile casing, and is shipped in versions with two D-Sub connectors or a screw-terminal strip.

1.1 Properties at a Glance

- NXP LPC21 series microcontroller (16/32-bit ARM CPU)
- 32 kbyte EEPROM
- Two High-speed CAN channels (ISO 11898-2) with bit rates from 40 kbit/s up to 1 Mbit/s
- Complies with CAN specifications 2.0 A/B
- Galvanic isolation of the D-Sub connector CAN 2 (only applies to the opto-decoupled version)
- Status signaling with two 2-color LEDs
- Connections via two 9-pin D-Sub connectors or one 10-pole screw-terminal strip (Phoenix)
- Additional digital input (only applies to models with D-Sub connectors)
- Aluminum casing, optional with DIN rail fixing option available
- 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-Router in aluminum casing
- IPEH-002210-P: Mating connector Phoenix Contact MC 1.5/10-ST-3.81

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 a new firmware via CAN:
  - CAN interface of the PCAN series for the computer (e.g. PCAN-USB)
  - Operating system Windows 11 (x64), 10 (x86/x64)

2 Connectors and Operating Elements

PCAN-Router with 10 connection poles or
2 x 9-pin D-Sub connectors and 2 status LEDs

The following connections can be used depending on the version:

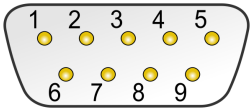
- 10-pole screw terminal block (IPEH-002210-P)
- 2 x 9-pin D-Sub connectors (m) (IPEH-002210/11)
- I/O pin: digital input on CAN channel 2 (IPEH-002210)
- I/O pin: digital input on CAN channel 1 (IPEH-002211)
- RS-232 interface for serial data transmission (IPEH-002210-P)
- Boot input for activation of the CAN bootloader for firmware upload (see section 6.2 *Preparing Hardware*)
- Serial and debugging ports of the microcontroller (see chapter 3 *Hardware Configuration*)

2.1 D-Sub Connectors

The PCAN-Router (IPEH-002210/11) each have two 9-pin D-Sub connectors, which are used for the CAN channels CAN1 and CAN2. The CAN lines (CAN-High, CAN-Low) are laid out corresponding to the CiA® 106 specification.

The power supply of the PCAN-Router can be provided via both D-Sub connectors. The opto-decoupled version (IPEH-002211) can only be supplied via CAN1. The supply connections U_{b1} and U_{b2} are connected internally without feedback. Thus, different voltage sources can be connected.

With the D-Sub connector CAN1, the bootloader for the firmware upload can be additionally activated via pin 6. The channel CAN1 (IPEH-002210) or CAN2 (IPEH-002211) additionally contains a digital input named Din0, which can be evaluated by the microcontroller.



Pin assignment D-Sub connector

Pin assignment IPEH-002210:

Pin	Function at CAN channel 1	Function at CAN channel 2
1	+5 V supply for external devices (optional)	+5 V supply for external devices (optional)
2	CAN1-Low	CAN2-Low
3	GND	GND
4	Reserved	Not used
5	SHIELD	SHIELD
6	Boot (High-active) Activation CAN bootloader	Not used
7	CAN1-High	CAN2-High
8	Not used	Din0 (Low-active)
9	Power supply $+U_{b1}$	Power supply $+U_{b2}$

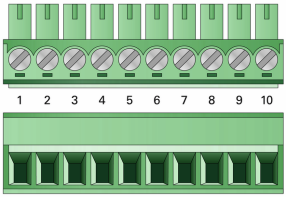
Pin assignment IPEH-002211 (opto-decoupled):

Pin	Function at CAN channel 1	Function at CAN channel 2
1	+5 V supply for external devices (optional)	+5 V supply for external devices (optionall, via DC/DC converter)
2	CAN1-Low	CAN2-Low
3	GND	GND
4	Reserved	Not used
5	SHIELD	SHIELD
6	Boot (High-active) Activation CAN bootloader	Not used
7	CAN1-High	CAN2-High
8	Din0 (Low-active)	Not used
9	Power supply $+U_{b1}$	Not used

2.2 Screw Terminal Block

The PCAN-Router (IPEH-002210-P) has a 10-pin screw terminal block for connecting the following components:

- Power supply
- CAN channels 1 and 2
- RS-232
- CAN bootloader activation (via pin 7)



Screw terminal block (Phoenix)
Mating connector Phoenix Contact MC 1.5/10-ST-3.81

Terminal assignment Screw terminal block:

Terminal	Identifier	Function
1	U_b	Power supply 8 to 30 V DC
2	GND	Ground
3	CAN1-Low	CAN channel 1 Low
4	CAN1-High	CAN channel 1 High
5	CAN2-Low	CAN channel 2 Low
6	CAN2-High	CAN channel 2 High
7	Boot CAN	Activation CAN bootloader (High-active)
8	Reserved	Reserved
9	RS-232 RxD	RS-232 interface
10	RS-232 TxD	

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

3 Hardware Configuration

For special applications, several settings can be done on the circuit board of the PCAN-Router by using solder bridges.

- Coding solder bridges for polling by the firmware
  (see section 3.1 *Coding Solder Bridges*)
- D-Sub versions IPEH-002210/-11 only: Supply of external devices with 5 Volts via the D-Sub connector
  (see section 3.2 *Power supply for external devices*)
- Debugging ports of the microcontroller
  (see section 3.3 *J5 Connector Panel: JTAG Ports*)
- Serial ports
  (see section 3.4 *J4 Connector Panel: Serial Ports*)

3.1 Coding Solder Bridges

The board has four coding solder bridges 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. A bit is set (1) if the corresponding solder field is open.

A concrete application is the identification of a PCAN-Router on the CAN bus during a firmware upload, especially if several routers 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-Router is identified by a 4-bit ID which is determined by the solder bridges (default setting: ID15, all solder fields open).

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

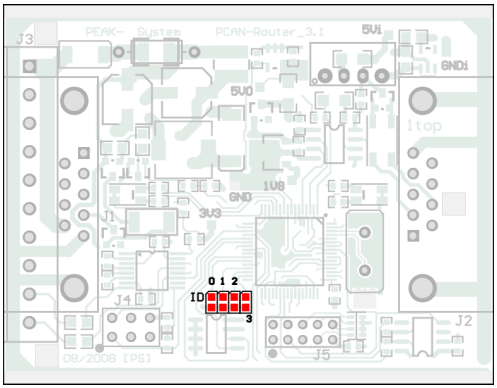
Activate coding solder bridges:

Risk of short circuit! Soldering on the PCAN-Router 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-Router from the power supply.
2. Unscrew the housing on both sides.
   For D-Sub version:
   Remove the screws next to one of the two D-Sub connectors.
3. Pull out the board.

4. Solder the solder bridge(s) on the board according to the desired setting.



Position of the coding solder fields

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

Solder Field Status	Port Status
	Low
	High

5. Carefully put the board back into the housing.
6. For D-Sub version:
   Reinsert the two screws at the D-Sub connector.
7. Screw the two sides of the housing back together.

3.2 Power supply for external devices

D-Sub versions IPEH-002210/-11 only

External devices with low power consumption (e.g. bus converters) can be supplied via the CAN connection CAN1 and CAN2. With a solder bridge for each CAN channel on the board of the PCAN-Router a voltage of 5 Volt can be applied to pin 1 of the D-Sub connector. The current consumption may not exceed 100 mA per CAN connection.

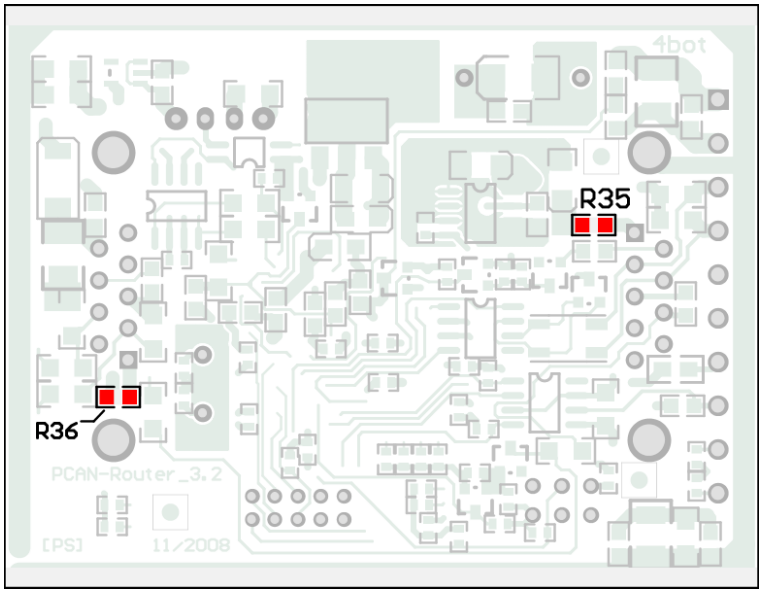
Activate 5 V supply:

Risk of short circuit! Soldering on the PCAN-Router 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-Router from the power supply.
2. Unscrew the housing on both sides.
3. Remove the screws next to one of the two D-Sub connectors.
4. Pull out the board.

5. Solder the solder bridge(s) on the board according to the desired setting.



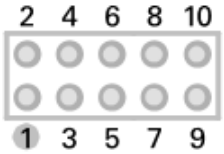
Position of the solder fields for the supply of external devices
(D-Sub connector R35 for CAN1, R36 for CAN2)

D-Sub Connector	5 Volt Supply None (Default)	 Pin 1
R35 (CAN1)		
R36 (CAN2)		

6. Carefully put the board back into the housing.
7. Reinsert the two screws at the D-Sub connector.
8. Screw the two sides of the housing back together.

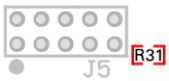
3.3 J5 Connector Panel: JTAG Ports

The unequipped connector panel J5 on the board of the PCAN-Router can be used as an access possibility to the JTAG ports of the microcontroller LPC2194/01 (µC) for hardware debugging.



Pin	Signal	Port µC	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

If constant internal pull-down wiring of the TCK or RTCK signals is not suitable for your purposes, you can remove the respective pull-down resistor on the circuit board of the PCAN-Router by soldering it out. Both resistors (each 10 kΩ) are located next to the connector panel J5 (see figures).

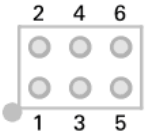


Board top side

Board bottom side

3.4 J4 Connector Panel: Serial Ports

The unequipped connector panel J4 on the board of the PCAN-Router can be used as an access possibility to the serial ports of the microcontroller LPC2194/01 (µC).



Pin	Signal	Port µC
1	RxD0	P0.1
2	TxD0	P0.0
3	Not used	
4	/Boot_ser	P0.14
5	GND	
6	+5.0 V	

The RxD0 and TxD0 signals are forwarded to a level converter for the RS-232 standard. The PCAN-Router model with screw terminal block provides access to the adjusted signals at terminals 9 (RS-232 RxD) and 10 (RS-232 TxD).

Attention! The RxD0 (pin 1) and TxD0 (pin 2) signals in connector panel J4 are designed for TTL levels only. Using RS-232 levels at these connections can cause damage to the electronics of the PCAN-Router.

4 Operation

The PCAN-Router is activated by applying the supply voltage to the respective connectors. More information about the connections can be found in chapter 2 *Connectors and Operating Elements*. The firmware in the flash memory is subsequently run.

On delivery, the PCAN-Router is supplied with an example firmware which forwards CAN messages 1:1 between the two channels at 500 kbit/s. An incoming CAN message causes a change between green and orange of the LED status indication for the respective CAN channel.

The source code for the example firmware `01_ROUTING` and further examples can be downloaded from the following link:

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

More about firmware can be found in chapter 5 *Creating Own Firmware*.

4.1 Status LEDs

Routing firmware is pre-installed on delivery. Therefore, the LEDs CAN1 and CAN2 light up between green and orange on incoming CAN messages.

Additionally, they light up orange when the bootloader is activated. Additional LED functions can be programmed with your own firmware. More details can be found in the programming examples supplied.

5 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-Router is supplied with the example firmware `01_ROUTING` which forwards CAN messages 1:1 between both CAN channels.

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-Router\Examples\01_ROUTING`

   You can edit the C code and use 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 6.2 *Preparing Hardware*.

8. Use the PEAK-Flash tool to upload your firmware to the PCAN-Router via CAN. The tool is either started via the menu *Terminal* > *Run Task* > *Flash Device* or from the subdirectory of the development package. Chapter 6.2 *Preparing Hardware* describes the process. A CAN interface of the PCAN series is required.

5.1 Library

The development of applications for the PCAN-Router is supported by the library `libPCAN-RouterGNU*ys.a` (`*` stands for version number), a binary file. You can access all resources of the PCAN-Router 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.

6 Firmware Upload

The microcontroller in the PCAN-Router 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-Router.
  See section 6.3 *Firmware Transfer via CAN*.
- **Via serial ports:**
  This requires access to the circuit board of the PCAN-Router.
  See section 6.4 *Uploading Firmware via the Serial Connections*.

6.1 System Requirements

- CAN interface of the PCAN series for the computer, for example PCAN-USB
- CAN cabling between the CAN interface and the PCAN-Router with correct termination at both ends of the CAN bus with 120 Ohm each.
- Operating system Windows 11 (x64), 10 (x86/x64)
- If you want to update several PCAN-Router on the same CAN bus with new firmware, you must assign an ID to each device. See section 3.1 *Coding Solder Bridges*.

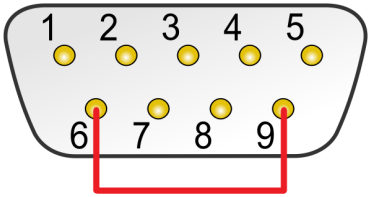
6.2 Preparing Hardware

For an upload of new firmware via CAN, the CAN bootloader must be activated in the PCAN-Router. The firmware transfer can only be done via CAN channel 1.

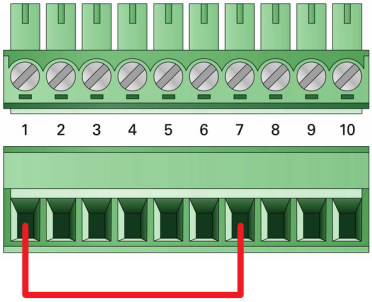
Activating CAN Bootloader:

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

1. Disconnect the PCAN-Router from the power supply.
2. Establish a connection between **Boot** and the power supply **U_{b1}, U_{b2}, or U_b**.



Connection at D-Sub connector CAN1 between the pins 6 and 9



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 a CAN bus of the PCAN-Router with a CAN interface connected to the computer. Pay attention to the proper termination of the CAN cabling (2 x 120 Ohm).

Risk of short circuit! A CAN cable with D-Sub connectors must not have a connection on pin 6, as it can be seen on 1:1 cables, for example. At other CAN nodes (e.g. a CAN interface of the PCAN series) this line may be applied to the mass. Damage or destruction of the electronics is a possible consequence.

4. Reconnect the power supply.
   Due to the high level at the **Boot** connection, the PCAN-Router starts the CAN bootloader. This can be determined by the status LEDs:

LED	State	Color
CAN1	quickly blinking	orange
CAN2	on	orange

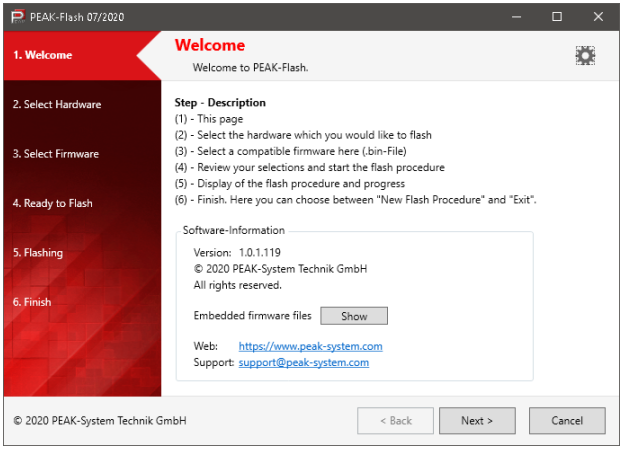
6.3 Firmware Transfer via CAN

A new firmware version can be transferred to the PCAN-Router with CAN channel 1. The firmware is uploaded via a CAN bus using the Windows software PEAK-Flash.

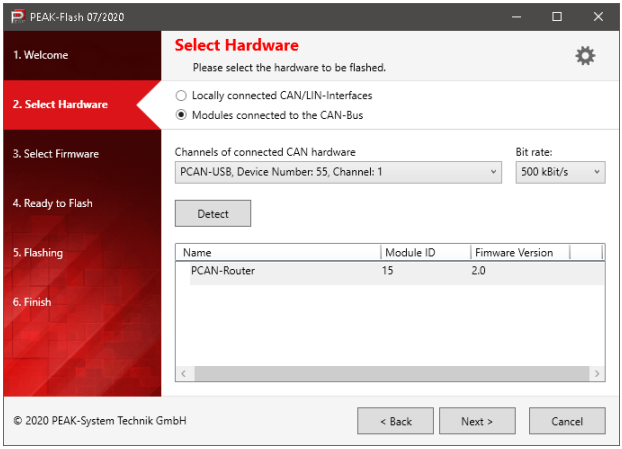
Transfer firmware with PEAK-Flash:

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.

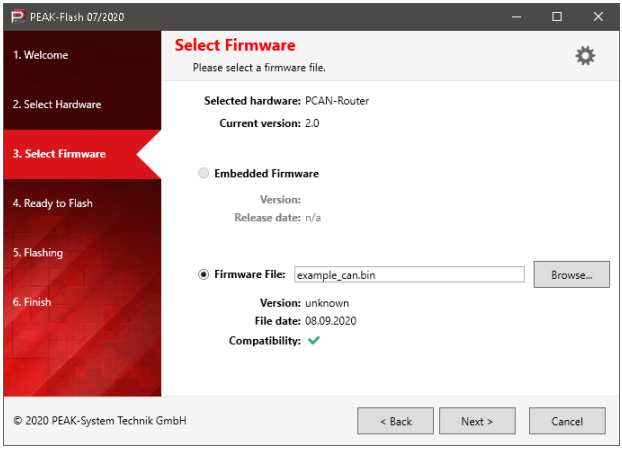


3. Click the button *Next*.
   The *Select Hardware* window appears.



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-Router 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-Router.
    The *Flashing* dialog appears.
13. After the process is complete, click *Next*.
14. You can exit the program.
15. Disconnect the PCAN-Router from the power supply.
16. Remove the connection between **Boot** and the power supply $\mathbf{U_{b1}}$, $\mathbf{U_{b2}}$, or $\mathbf{U_b}$.
17. Connect the PCAN-Router to the power supply.

You can now use the PCAN-Router with the new firmware.

6.4 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.

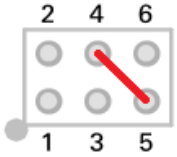
Important: When uploading a firmware via the RS-232 interface, the CAN bootloader may be overwritten. Afterwards, a firmware upload via CAN is not possible anymore.

Activate the microcontroller's bootloader:

Risk of short circuit! Soldering on the PCAN-Router 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-Router from the power supply.
2. Unscrew the housing on both sides.
   For D-Sub version:
   Remove the screws next to one of the two D-Sub connectors.
3. Pull out the board.
4. Establish a connection on the J4 connector panel between pin 4 (\Boot_ser) and pin 5 (GND).



5. Establish a serial connection to the computer or program adapter.
   This is carried out either via the RS-232 interface (IPEH002210-P only) or via the serial ports of the microcontroller (TTL levels).

6. 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.
7. You can now start the firmware upload.

After completing the firmware upload

1. Disconnect the PCAN-Router from the power supply.
2. Remove the connection between pin 4 (\Boot_ser) and pin 5 (GND) on the J4 connector panel.
3. Carefully put the board back into the housing.
4. For D-Sub version:
   Reinsert the two screws at the D-Sub connector.
5. Screw the two sides of the housing back together.

You can now use the PCAN-Router with the new firmware.

7 Technical Data

Connectors IPEH-002210/-11	
Power	2 x D-Sub (m), pin 9 (U_b)
CAN	2 x D-Sub (m), pins 2 and 7, assignment according to specification CiA® 106

Connectors IPEH-002210-P	
Screw terminal block	10 connection poles, pitch 3.81 mm Phoenix Contact MC 1.5/10-ST-3.81
Power	Pin 1 (U_b)
CAN channel 1	Pins 3 and 4
CAN channel 2	Pins 5 and 6
RS-232	Pins 9 and 10

RS-232 (IPEH-002210-P)	
Connectors	RxD and TxD
Bitrate maximum	230400 Baud
Signal level maximum	±9 V
Dielectric strength	V_{ESD} ±4 kV
Galvanic isolation	None

Digital Inputs (IPEH-002210/-11)	
Count	1
Connectors	Din0
Input voltage	-30 to +30 V
Input current range	0 to -1.44 mA
Input current	0.2 mA
Input impedance	22 kΩ
Input circuitry	Pull-down: 10 kΩ to ground
Switching thresholds	Low-High: > 0.6 V High-Low: < 0.8 V
Low pass	7.23 kHz
Protection	V_{ESD} ±4 kV
Galvanic isolation	IPEH-002211: D-Sub connector CAN 2 up to 500 V

Power Supply		
Supply voltage (U_b)	12 V DC, 8 to 30 V DC	
Current consumption	Idle:	60 mA at 12 V
	Maximum:	124 mA at 8 V
		90 mA at 12 V
		50 mA at 30 V
Power saving modes	None	
Protection	±1 kV surge protection -50 V reverse polarity protection ±4 kV ESD protection	

CAN		
Protocols	CAN 2.0 A/B	
Physical transmission	ISO 11898-2, High-speed CAN	
Transceiver	NXP TJA1040T	
CAN bit rates	Nominal: 40 kbit/s to 1 Mbit/s (lower bit rates on request)	
Controller	Internal CAN controller (NXP LPC2194/01)	
Supported clock frequencies	60 MHz	
Supported bit timing values		Nominal
	Prescaler (BRP)	1 to 1024
	Time Segment 1 (TSEG1)	1 to 16
	Time Segment 2 (TSEG2)	1 to 8
	Synch. Jump Width (SJW)	1 to 4
Galvanic isolation	IPEH-002211: D-Sub connector CAN 2 up to 500 V	
Internal termination	None	
Dielectric strength	V_{ESD} ±4 kV V_{CAN} ±22 V per pin	

Microcontroller	
CPU	NXP LPC2194/01
Clock frequency	60 MHz
RAM	16 kByte SRAM
Memory	240 kByte MCU Flash 32 kByte On-Chip-EEPROM
Firmware upload	Via CAN (PCAN interface required) or serial ports of the microcontroller IPEH-002210-P: additionally possible via RS-232

Measures		
Size	IPEH-002210/-11: IPEH-002210-P: Casing: Circuit board:	78.7 x 55 x 24 mm (W x H x D) 82.3 x 55 x 24 mm (W x H x D) 70.2 x 50 x 24 mm (W x H x D) 51 x 65 mm (W x H)
Weight	IPEH-002210/-11: IPEH-002210-P:	95 g 89 g

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 2	EU directive 2011/65/EU (RoHS 2) + 2015/863/EU DIN EN IEC 63000:2019-05
EMC	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-Router**
Item number(s):	**IPEH-002210, IPEH-002210-P, IPEH-002211**
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 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

Uwe Wilhelm, Managing Director

Appendix B UKCA-Certificate

UK Declaration of Conformity

This declaration applies to the following product:

Product name: **PCAN-Router**

Item number(s): **IPEH-002210, IPEH-002210-P, IPEH-002211**

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

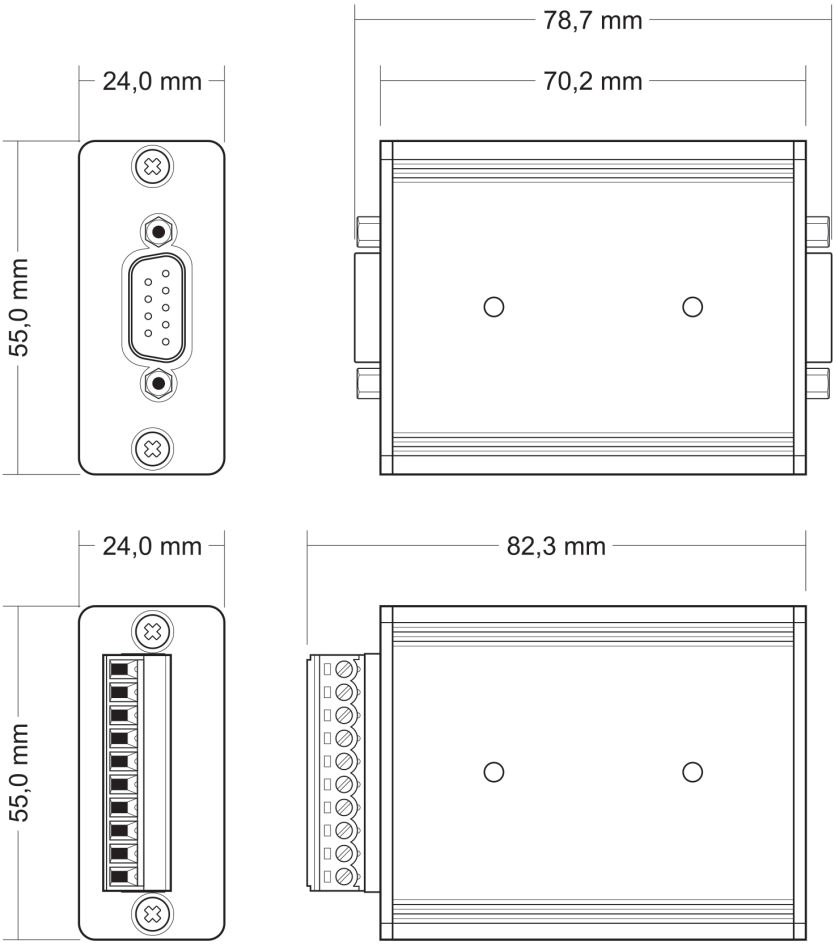
Electromagnetic Compatibility Regulations 2016

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

Uwe Wilhelm, Managing Director

Appendix C Dimension Drawings



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-Router. It is meant as supplemental information. The converter's functionality is implemented by the supplied library.

Get additional information about the LPC2194/01 microcontrollers 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, J4:2 or STB:10 (RS-232 levels)
P0.1	I	RxD UART0	RxD0		Serial communication, Receive, J4:1 or STB:9 (RS-232 levels)
P0.2	I, O	SCL	SCL		I^2C bus to the EEPROM Microchip 24LC02B or Atmel AT24C256B
P0.3	I, O	SDA	SDA		
P0.4	I	Port pin	ID0	High	Coding solder bridges 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	
P0.8	O	TxD UART1			
P0.9	I	RxD UART1			
P0.10	O	Port pin			
P0.11	I	Hardware capture with timer			

[1] CAN1/2:n Pin n of the respective D-Sub connector
SKL:n Terminal n on the screw terminal block
J4/5:n Pin n of the respective connector panel on the board

Port	I/O	µC Function	Signal	Active (µC)	Function / Connection[1]
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, J4:4
P0.15	I	Port pin	/Boot_CAN	Low	Activate flashing via CAN1 with 500 kbit/s, CAN1:9 and STB:7 (due to wiring High-active)
P0.17	O	Port pin	V24_en	High	Deactivate the RS-232 converter by Low level (activated by default); possibility for energy saving
P0.18	I	Hardware capture with timer			
P0.19	I	Port pin	Switch	High	Digital input Din0, CAN2:8 (due to wiring Low-active)
P0.20		Hardware capture with timer			
P0.21	O	Port pin	CAN_en_2	Low	Activate the respective CAN transceiver[2]
P0.22	O	Port pin	CAN_en_1	Low	
P0.23	I	RD2	CAN2_RxD		CAN2 Receive
P0.24	O	TD2	CAN2_TxD		CAN2 Transmit
P0.25	I	RD1	CAN1_RxD		CAN1 Receive
TD1	O	TD1	CAN1_TxD		CAN1 Transmit
P0.27	I	Analog input	V-Power2		Measure voltage $+U_{b2}$, maximum value (0x03FF) corresponds to approx. 16.5 V

Port	I/O	µC Function	Signal	Active (µC)	Function / Connection[1]
P0.28	I	Analog input	V-Power1		Measure voltage $+U_{b1}$ or $+U_b$, 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 CAN1 red
P1.17	O[3]	Port pin		Low	LED CAN1 green
P1.18	O	Port pin		Low	LED CAN2 red
P1.19	O	Port pin		Low	LED CAN2 green
P1.25	O				
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

[2] After resetting the microcontroller, the CAN transceivers are deactivated and must be reactivated to use them.

[3] It may occur that an LED glows slightly when the respective 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).

Appendix E Disposal

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