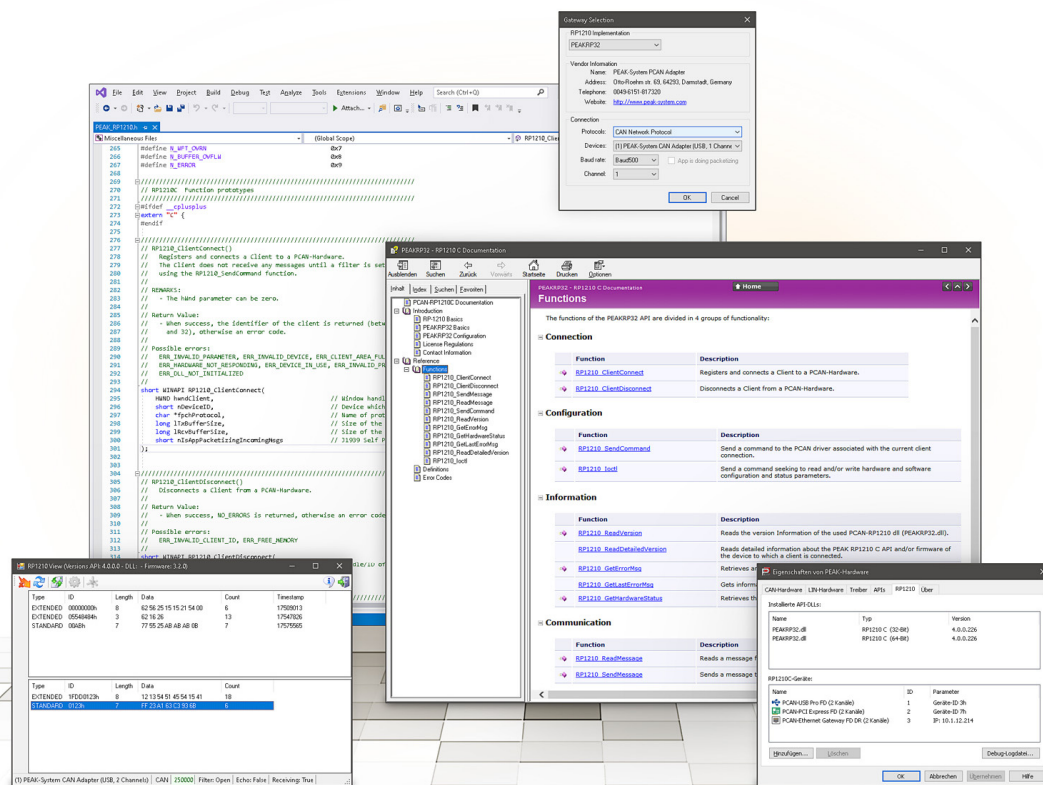


# PCAN-RP1210 API

API Implementation of the Recommended Practices 1210 (RP1210)

## Documentation



PCAN® is a registered trademark of PEAK-System Technik GmbH. All other product names mentioned in this document may be the trademarks or registered trademarks of their respective companies. They are not explicitly marked by "™" or "®".

Copyright © 2020 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-Roehm-Straße 69  
64293 Darmstadt  
Germany

Phone: +49 (0)6151 8173-20  
Fax: +49 (0)6151 8173-29

[www.peak-system.com](http://www.peak-system.com)  
[info@peak-system.com](mailto:info@peak-system.com)

Technical Support:  
E-mail: [support@peak-system.com](mailto:support@peak-system.com)  
Forum: [www.peak-system.com/forum/](http://www.peak-system.com/forum/)

Document version 2.0.1 (2020-11-06)

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>PCAN-RP1210A (PCANRP32) – DEPRECATED</b>	<b>5</b>
<b>3</b>	<b>PCAN-RP1210C (PEAKRP32)</b>	<b>6</b>
3.1	Implementation Name and Files	6
3.2	Scope of Supply	6
3.3	PCAN-RP1210C and the PCAN Environment	6
3.4	Configuration of PCAN Devices as RP1210C Devices	6
3.5	Supported Commands (RP1210_SendCommand function)	7
<b>4</b>	<b>Main Differences between PCAN-RP1210A and PCAN-RP1210C</b>	<b>8</b>
4.1	Blocking versus Non-Blocking Calls	8
4.2	Device Connection versus Channel Connection	8
4.3	Fixed Bit Rate versus Configurable Bit Rate	8
4.4	Support for ISO 15765	9
4.5	Redistribution for Third Parties	9

# 1 Introduction

The Recommended Practices 1210 (RP1210) is a standardized interface established by the Maintenance Council (TMC) to allow a communication between the on-vehicle data link and PC software applications developed for Windows operating systems, permitting in that way hardware adapters to be transparent to the application programmer. Until now, three versions of this recommended practices have been written: A, B, and C.

PCAN-RP1210 stands for PEAK-CAN RP1210 and it is an implementation of the TMC's RP1210 Windows™ Communication Application Program Interface (API), which is intended for a datalink communication from a personal computer to a vehicle, under the Microsoft Windows™ family of operating systems (Windows™ 8.1, and 10).

Until 2019 PEAK-System offered two RP1210 implementations in one package. PCAN-RP1210A (internally called PCANRP32), the first official version of RP1210, and PCAN-RP1210C, which is the newest version (internally called PEAKRP32). Each of them is implemented as a separated and individual library.

PEAK-System discontinued the implementation of PCAN-RP1210A in July of 2020. The product model was also changed, PCAN-RP1210 is now available for free.



**Note:** RP1210 versions A and B were superseded by version C. For this reason, PEAK-System maintains only the implementation of version C in the PCAN-RP1210 package.

## 2 PCAN-RP1210A (PCANRP32) – DEPRECATED

IMPORTANT: This library is deprecated and will not be maintained anymore. Use PCAN-RP1210C, instead. PEAK-System discontinued the implementation of PCAN-RP1210A in July of 2020.

## 3 PCAN-RP1210C (PEAKRP32)

PCAN-RP1210C is the PEAK-System implementation of the Recommended Practice RP1210, version C, from TMC. PCAN-RP1210C works under Windows systems of 32-bit and 64-bit and supports the protocols CAN, J1939, and ISO 15765.

### 3.1 Implementation Name and Files

This implementation consists of a dynamic link library (DLL) and an initialization file (INI) with information about the supported PCAN devices and protocols. The names of the DLL and INI files are: `PEAKRP32.DLL` and `PEAKRP32.ini`. The name of the library is the same for both platforms (32-bit and 64-bit).

### 3.2 Scope of Supply

Added to the implementation files, the PCAN-RP1210 package also installs basic information about this API (this document), an online-help API documentation (chm document), a header file for the programming language C++, `PEAK_RP1210.h`, as API reference, and a small test application (binary) for a device connection check.

### 3.3 PCAN-RP1210C and the PCAN Environment


All PCAN devices offered by PEAK-System can be used with PCAN-RP1210C. Among others, the following devices are supported: PCAN-USB Classic, PCAN-USB Pro, PCAN-USB FD, PCAN-USB Pro FD, PCAN-USB X6, PCAN-PCI (1/2/4 channels), PCAN-Ethernet Gateway DR (using Virtual PCAN-Gateway), PCAN-PCI Express, and PCAN-PC Card.

The link between PCAN-RP1210C and PCAN devices is done using the PEAK Hardware Control Panel Applet. PCAN devices are registered to act as RP1210C devices, that can have one or more channels. PCAN-RP1210C (PEAKRP32) allows passing the desired bit rate as parameter as well as the channel to use on the connecting device. This happens by using the connection string of the function

`RP1210_ClientConnect`.

### 3.4 Configuration of PCAN Devices as RP1210C Devices

With PCAN-RP1210C the possibility to connect to different channels of a same device was introduced. This required a better identification of a device and its type. Due to this extension a RP1210C device must be registered over the PEAK Hardware Control Panel Applet. Since no default IDs are possible anymore, each device must explicitly be registered to be used.

 **Note:** During installation of the PCAN-RP1210 package, all connected hardware will automatically be registered. Hardware connected after installation must be registered as RP1210C device over the PEAK Hardware Control Panel Applet. Do the following to register a RP1210C device:

1. Make sure that the desired device is connected to the computer.
2. Go to the Windows control panel and locate/start the PEAK Hardware Control Panel Applet.



**Tip:** Enter **PEAKCPL** in the Windows search bar to open the applet.

3. Change to the **RP1210** tab.

The already registered devices are listed here. The list already shows the device IDs to be used with the `RP1210_ClientConnect` function. Furthermore, the IDs can be found in the `PEAKRP32.ini` file.

4. Click the **Add** button to get a list of available devices.
5. Select the desired device from the list and click **Ok**.



**Note:** Only devices are listed, that are not yet in the list of RP1210C devices.

## 3.5 Supported Commands (`RP1210_SendCommand` function)

- └ Reset Device
- └ Set All Filter States to Pass
- └ Set Message Filtering for CAN
- └ Set Message Filtering for J1939
- └ Set Message Filtering for ISO15765
- └ Set Echo Transmitted Messages
- └ Set All Filter States to Discard
- └ Set Message Receive
- └ Protect J1939 Address
- └ Release J1939 address
- └ Set ISO15765 flow control
- └ Clear ISO15765 flow control
- └ Set broadcast list for CAN
- └ Set broadcast list for J1939
- └ Set broadcast list for ISO15765
- └ Set filter type for CAN
- └ Set filter type for J1939
- └ Set filter type for ISO15765
- └ Set J1939 broadcast interpacket timing
- └ Set max error message size
- └ Disallow further client connections
- └ Set CAN Baudrate
- └ Set J1939 Baudrate
- └ Set ISO15765 Baudrate
- └ Set blocking timeout
- └ Flush send/receive buffers

## 4 Main Differences between PCAN-RP1210A and PCAN-RP1210C

Earlier versions of the PCAN-RP1210 package came with two different implementations of RP1210. The deprecated version A called PCAN-RP1210A (PCANRP32) and the current version, version C, called PCAN-RP1210C (PEAKP32). For better understanding of this first approach, this chapter covers the main differences between these two libraries.

### 4.1 Blocking versus Non-Blocking calls

PCAN-RP1210A supports calling a function in blocking mode, this is, calling the function and waiting for a response or a timeout. This was removed in RP1210, version C, with exception of calls to `RP1210_ReadMessages`.

Some software relies on blocking procedures as a kind of assertion for different task, for example, to be sure a message was really sent on the CAN bus.

Most applications only need to read in blocking mode, as a form to react as soon as possible to received messages. For this, a thread is normally started and `RP1210_ReadMessage` is called in blocking mode, so that the function only returns when a message is received.

### 4.2 Device Connection versus Channel Connection


PCAN-RP1210C introduced the possibility to connect to channels of a same device. The device ID is used within the `RP1210_ClientConnect` function to connect a multi-channel device. This is the same for all channels of that device. The channel to use is set over the `fpchProtocol` string parameter. More information about this can be found within the companion PCAN-RP1210C Documentation (`PCAN-RP1210C_enu.chm`) or in the official RP1210 specification, version C, from TMC/ATA.

The difference regarding the connection to channels/devices also makes a difference on the tool used for channel/device configuration. For PCAN-RP1210A the PCAN-Net Configuration tool is used. For PCAN-RP1210C the PEAK Hardware Control Panel Applet is used. These configurations are independent from each other. This means, configuring the device for PCAN-RP1210C does not configure it for use with PCAN-RP1210A, and vice versa.

### 4.3 Fixed Bit Rate versus Configurable Bit Rate

PCAN-RP1210A has support for 250 kbit/s for CAN and J1939 protocols. This bit rate is fix and cannot be changed using any `RP1210_ function`.

For PCAN-RP1210C, the bit rate to be used can be set with the `fpchProtocol` string parameter. Moreover, the value **Auto** can be used, so that the bit rate is automatically determined.

 **Note:** Only 4 bit rates are supported: 125 kbit/s, 250 kbit/s, 500 kbit/s, and 1 Mbit/s. More information about this can be found within the companion PCAN-RP1210C documentation (`PCAN-RP1210C_enu.chm`) or in the official RP1210, version C, specification from TMC/ATA.

Added to this, PCAN-RP1210C makes it possible to change the bit rate “on the fly” by calling the function `RP1210_SendCommand` with the parameters `RP1210_SetCAN_Baud` and `RP1210_Set_J1939_Baud`, and also ask the current bit rate by using the parameter `RP1210_Get_Protocol_Connection_Speed`.



## 4.4 Support for ISO 15765

PCAN-RP1210C introduced support for ISO 15765-2, also known as ISO-TP. This protocol is widely used for diagnosis and allows transporting large data packages of up to 4095 bytes.

## 4.5 Redistribution for Third Parties

The PEAK-Drivers Setup contains the latest version of the device drivers and of PCAN-RP1210C. If desired, it is possible to deliver this package together with own developed RP1210 software. This package can be installed using the command line, so that drivers and APIs can be installed using scripts, which allows the integration of this setup in an end user solution. Please consult the help document of PEAK-Drivers Setup, [PEAK-Drivers\\_4.x\\_Command-Line-User-Manual\\_en-us.pdf](#), to obtain more information.