Wireshark Dissector Support PCAN-Gateways IPEH-004010, IPEH-004011, IPEH-004012, IPEH-004020

🔏 *Eth	ernet					- 🗆 🗙
<u>D</u> atei	<u>B</u> earbeiten <u>A</u> nsicht	Navigation Aufzeichnen	Analyse <u>S</u> tatisti	ken Telephonie <u>W</u> ireless <u>T</u>	ools <u>H</u> ilfe	
	1 🐵 📘 📑 🗙 🛛	😋 🍳 👄 🔿 警 👰	🕹 📃 🗏 🔍 🤆	Q. Q. 👖		
ip.ad	dr == 192.168.1.111					× → +
No.	Time	Source	Destination	Protocol	Length Info	
1	27158 27.227632	192.168.1.111	192.168.1.62	PEAK-CAN over UDF	226 CAN Msg(s) in Frame:	2
	27159 27.229636	192.168.1.111	192.168.1.62	PEAK-CAN over UDF	234 CAN Msg(s) in Frame:	2
	27160 27.229636	192.168.1.111	192.168.1.62	PEAK-CAN over UDF	226 CAN Msg(s) in Frame:	2
	27161 27.231532	192.168.1.111	192.168.1.62	PEAK-CAN over UDF	234 CAN Msg(s) in Frame:	2
	27162 27.231532	192.168.1.111	192.168.1.62	PEAK-CAN over UDF	226 CAN Msg(s) in Frame:	2
	27163 27.233543	192.168.1.111	192.168.1.62	PEAK-CAN over UDF	234 CAN Msg(s) in Frame:	2
<						,
> Inte	rnet Protocol Vers	ion 4, Src: 192.168.	1.111, Dst: 192	168.1.62		^
Y CAN	Protocol Data - PF	AK-System Format	St POPt: 5050			
C	omplete CAN Data s	ize in Frame Length:	192			
c	AN FD Frames with	CRC: 2				
~ 0	AN Data Storage wi	th 2 entrys				
	CAN-Channel: 0					
1	/ internal Flags:					
	- Extended ID	2.00%				
	- Bit Rate Sw	itch				
	- Error State	Indicator				
	CAN Messagerype:	CAN FD WITH CKC FFa	ne			
	TimeStamp-LOW In	n uS+ 228762				
	CAN-TD: 0x000ffd	ff				
	Ext. 29Bit Msg.					
	DLC: 15					
	DB00:0x43 DB01:0	x24 DB02:0x32 DB03:0	x32 DB04:0x42 DB	05:0x03 DB06:0x00 DB07:0	00	
	DB08:0x00 DB09:0	x00 DB10:0x00 DB11:0	x00 DB12:0x00 DE	13:0x00 DB14:0x00 DB15:0	00	
	DB16:0x00 DB17:0	x00 DB18:0x00 DB19:0	x00 DB20:0x00 DE	21:0x00 DB22:0x00 DB23:0x	00 0	
	DB24:0x00 DB25:0	x00 DB26:0x00 DB27:0	x00 DB28:0x00 DE	29:0x00 DB30:0x00 DB31:0x	.00	
	DB32:0x00 DB33:0	x00 DB34:0x00 DB35:0	x00 DB36:0x00 DE	37:0x00 DB38:0x00 DB39:0x	00	
	DB40:0x00 DB41:0	x00 DB42:0x00 DB43:0	x00 DB44:0x00 DE	45:0x00 DB46:0x00 DB47:0x		ARK
	DB48:0x00 DB49:0	x00 DB50:0x00 DB51:0	x00 DB52:0x00 DB	53:0x00 DB54:0x00 DB55:0x		
	DB56:0x00 DB57:0	x00 DB58:0x00 DB59:0	x00 DB60:0x00 DE	61:0x00 DB62:0x00 DB63:0	(00	
	CRC: 3093460400					
	can chan la o					
	CAN-Channel: 0					
	CAN MassageType:	CAN ED with CRC Eca				
	TimeStamp-Low in	US: 40656773	iie.			
	TimeStamp-High i	n uS: 228762				
	CAN-ID: 0x000ffd	ff				
	Fyt 29Rit Mcg					•
<						>
0020	01 3e 9d cc 13 ba	00 c8 40 64 <mark>00 60</mark> 0	91 00 00 .>.	•••••• @d		
0030	00 00 00 00 00 00	02 6c 5f 80 00 03 7	d 9a 00 0f ····	····1 _···}···		
0040	00 52 80 0T TO TT	45 24 52 52 42 03 0 00 00 00 00 00 00 0	00 00 00 ·2·			
0060	00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00			
0 2	CAN /CAN-FD Ethernet	Gateway Protocol (peak-can)	2 Bytes		Pakete: 27837 · Angezeigt: 27195 (97.7%)· Verv	vorfen: 0 (0.0%) Profil: Default



Document version 1.2 (2022-01-24)



Relevant products

Product name	Model	Part number
PCAN-Ethernet Gateway DR	CAN to LAN Gateway in DIN Rail Plastic Housing	IPEH-004010
PCAN-Wireless Gateway DR	CAN to WLAN Gateway in DIN Rail Plastic Housing	IPEH-004011
PCAN-Wireless Gateway	CAN to WLAN-Gateway	IPEH-004020 IPEH-004020-A
PCAN-Ethernet Gateway FD DR	CAN FD to LAN Gateway in DIN Rail Plastic Housing	IPEH-004012

PCAN® is a registered trademark of PEAK-System Technik GmbH. Wireshark and the "fin" logo are registered trademarks of the Wireshark Foundation

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 "TM" or "@".

Copyright © 2022 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 info@peak-system.com

Document version 1.2 (2022-01-24)

Contents

1 Introduction	4
1.1 Properties at a Glance	5
2 Installing the Software	6
3 Capture some traffic	7
3.1 Setup Wireshark™ for Decode CAN Data	7
3.2 Start Analyzing Network Traffic	9
3.3 Fragmented CAN Packages	11
3.5 IP Frame Format Description	12
3.5.1 Optional CRC32 Checksum	17
3.6 Notes about the License	18

PEAK



1 Introduction

The PEAK-System Wireshark[™] Dissector is a simple LUA AddIn for the free available Network Analyzer Wireshark[™].

Wireshark[™] is the world's foremost network protocol analyser. It lets you see what's happening on your network at a microscopic level. It is the de facto (and often de jure) standard across many industries and educational institutions.

Wireshark development thrives thanks to the contributions of networking experts across the globe. It is the continuation of a project that started in 1998.

See https://www.wireshark.org

PEAK-System offer a free of charge LUA AddIn that could be easily used to decode Network Traffic from/to the PEAK-System Network to/from CAN Interfaces via Wireshark™



1.1 Properties at a Glance

- Easy to integrate into an existing Wireshark[™] installation
- Open-Source LUA code
- Basic Setup could be done with the Wireshark[™] preference menu
- Will work with all Wireshark Implementations, at all available Operating Systems. See https://www.wireshark.org for more details.
- Note: This manual describes how to use of the Wireshark Dissector with Wireshark™ Version 3.6.1 on a Windows™ System.



2 Installing the Software

This chapter covers the installation of the LUA Dissector. Up in front you must install the Wireshark Software. https://www.wireshark.org/#download. For Windows™ is also a Portable Version available.

The Wireshark installer contains also the latest WinPcap installer. If you don't have WinPcap installed, you won't be able to capture live network traffic, but you will still be able to open saved capture files. By default, the latest version of WinPcap will be installed. If you don't wish to do this or if you wish to reinstall WinPcap you can check the Install WinPcap box as needed.

For more information about WinPcap see https://www.winpcap.org/ and https://wiki.wireshark.org/WinPcap.

Do the following to install the Dissector.:

- Copy the Start pcan_gateway.lua from the supplied Mediao or Download to the root of your Wireshark™ installation.
- Goto the x:\WiresharkPortable\App\Wireshark\ Directory and search for a file called init.lua (depending on your selected Wireshark™ Installation Path).
- 3. Open the init.lua file with a Text Editor (for Example Notepad)

```
Copy this Text block at the end of the File:
-- Add PEAK Network CAN Protocol at startup
PEAK_PROTO_SCRIPT_PATH="x:\\WiresharkPortable\\"
dofile(PEAK_PROTO_SCRIPT_PATH.."pcan_gateway.lua")
```

Change the PATH for your need.

 Start the Wireshark Tool and check if the Dissector was loaded successfully. For that simply open the Wireshark™ preferences Dialog by going to the main menu and select "*Edit – Preferences*" (Ctrl+Shift+P). In the section *Protocols* you should now see a entry "PEAK-CAN"



3 Capture some traffic

3.1 Setup Wireshark™ for Decode CAN Data

When you have startup the Wireshark Tool, open the Preferences and go to Protocols. Search for *PEAK-CAN*.

PCP PDC PDCP-LTE PDCP-NR PDU Transport PEAK-CAN PEEKREMOTE PER PER PECP	 CAN /CAN-FD Ethernet Gateway Protocol A PEAK-System Ethernet and WLAN Gateway Dissector for WireShark Ports 50001-50002 Show all available Info (Expert mode) Select the used Protocol UDP ∨
--	---

In the first Edit field, you can set a Port Range, or a single Port Value where the Dissector should listen to.



The CheckBox allow you to set/unset a enhanced decoding. When the Check Box is set, the Dissector also decode the not used BitFields in the Data Package.

Show all available Info (Expert mode)

Here a samples of a CAN-FD Frame transmittet with UDP, with Experet mode off and on.



Expert mode off:

```
CAN Protocol Data - PEAK-System Format
     Complete CAN Data size in Frame Length: 192
     CAN FD Frames with CRC: 2
  ✓ CAN Data Storage with 2 entrys
        TimeStamp-Low in µS: 40656768
        TimeStamp-High in µS: 228762
        CAN-ID: 0x000ffdff
        Ext. 29Bit Msg.
        DLC: 15
        DB00:0x43 DB01:0x24 DB02:0x32 DB03:0x32 DB04:0x42 DB05:0x03 DB06:0x00 DB07:0x00
        DB08:0x00 DB09:0x00 DB10:0x00 DB11:0x00 DB12:0x00 DB13:0x00 DB14:0x00 DB15:0x00
        DB16:0x00 DB17:0x00 DB18:0x00 DB19:0x00 DB20:0x00 DB21:0x00 DB22:0x00 DB23:0x00
        DB24:0x00 DB25:0x00 DB26:0x00 DB27:0x00 DB28:0x00 DB29:0x00 DB30:0x00 DB31:0x00
        DB32:0x00 DB33:0x00 DB34:0x00 DB35:0x00 DB36:0x00 DB37:0x00 DB38:0x00 DB39:0x00
        DB40:0x00 DB41:0x00 DB42:0x00 DB43:0x00 DB44:0x00 DB45:0x00 DB46:0x00 DB47:0x00
        DB48:0x00 DB49:0x00 DB50:0x00 DB51:0x00 DB52:0x00 DB53:0x00 DB54:0x00 DB55:0x00
        DB56:0x00 DB57:0x00 DB58:0x00 DB59:0x00 DB60:0x00 DB61:0x00 DB62:0x00 DB63:0x00
        CRC: 3093460400
```

Expert mode on:

```
CAN Protocol Data - PEAK-System Format
     Complete CAN Data size in Frame Length: 192
     CAN FD Frames with CRC: 2
  CAN Data Storage with 2 entrys
        CAN-Channel: 0
     internal Flags:
           - Extended ID
           - Bit Rate Switch
           - Error State Indicator
        CAN MessageType: CAN FD with CRC Frame
        TimeStamp-Low in µS: 40656768
        TimeStamp-High in µS: 228762
        CAN-ID: 0x000ffdff
        Ext. 29Bit Msg.
        DLC: 15
        DB00:0x43 DB01:0x24 DB02:0x32 DB03:0x32 DB04:0x42 DB05:0x03 DB06:0x00 DB07:0x00
        DB08:0x00 DB09:0x00 DB10:0x00 DB11:0x00 DB12:0x00 DB13:0x00 DB14:0x00 DB15:0x00
        DB16:0x00 DB17:0x00 DB18:0x00 DB19:0x00 DB20:0x00 DB21:0x00 DB22:0x00 DB23:0x00
        DB24:0x00 DB25:0x00 DB26:0x00 DB27:0x00 DB28:0x00 DB29:0x00 DB30:0x00 DB31:0x00
        DB32:0x00 DB33:0x00 DB34:0x00 DB35:0x00 DB36:0x00 DB37:0x00 DB38:0x00 DB39:0x00
        DB40:0x00 DB41:0x00 DB42:0x00 DB43:0x00 DB44:0x00 DB45:0x00 DB46:0x00 DB47:0x00
        DB48:0x00 DB49:0x00 DB50:0x00 DB51:0x00 DB52:0x00 DB53:0x00 DB54:0x00 DB55:0x00
        DB56:0x00 DB57:0x00 DB58:0x00 DB59:0x00 DB60:0x00 DB61:0x00 DB62:0x00 DB63:0x00
        CRC: 3093460400
```



The Last Option is for selecting the Protocoll which should be used by the Dissector. TCP or UDS. Please be sure that you select the correct Type!

Select the used Protocol UDP	~
	set the UDP or TCP as protocol - TCP need a listener to get work

3.2 Start Analyzing Network Traffic

After selected the correct Network Interface (be sure you know on which Network you Gateways are running) you could now start capturing the Network traffic. If you run on the same Network Interface other Application, or you have only one physical Network Interface on your PC, you better setup a filter. Otherwise, you also see all the other Network Traffic in the Wireshark Tool.

Simply add "peak-can" in the Filter line to be sure that you only see Ethernet Packages that are send over the ports you have selected in the PORT settings. Or simply filter by the IP Address of the Gateway. Both is possible, also in combination.



Sample: ip.addr == 192.168.1.111 && peak-can

1	Ethernet							
File	Edit View Go Ca	pture Analyze Statistic	s Telephony Wireless	Tools Help				
41	1 🖉 💿 📘 🛤 🗙	🔁 🍳 👄 🔿 🕾 🛉	J 📜 🔲 Q Q 🖲	N				
ip.	addr == 192.168.1.111 &8	k peak-can		•				
No.	Time	Source	Destination	Protocol	Length	Info		
	8349 7.523352	192.168.1.111	192.168.1.62	PEAK-CAN over UDP	234	CAN Msg(s)	in Frame:	2
	8350 7.523352	192.168.1.111	192.168.1.62	PEAK-CAN over UDP	226	CAN Msg(s)	in Frame:	2
1	8351 7.525266	192.168.1.111	192.168.1.62	PEAK-CAN over UDP	466	CAN Msg(s)	in Frame:	5
10	8352 7.525266	192.168.1.111	192.168.1.62	PEAK-CAN over UDP	446	CAN Msg(s)	in Frame:	5



Also check if you have selected the correct protocol type.

4 *Ethern	et							-		×
File Edit	View Go Ca	pture Analyze	Statistics Telephony Wi	eless Tools Hel	р					
	🖲 📘 🛤 🗙	9 9 00	2 T 4 T = Q	Q Q II						8
eak-can								×	Expression	. +
No.	Time	Source	Destination	Protocol		Length Info				^
50575	106.811236	10.1.12.201	10.1.12.91	PEAK-CAN	over TCP	342 CAN Msg(s) in Frame: 8	68 - C			
50595	106.852285	10.1.12.201	10.1.12.91	PEAK-CAN	over TCP	378 CAN Msg(s) in Frame: 9	18			
50613	106.894023	10.1.12.201	10.1.12.91	PEAK-CAN	over TCP	342 CAN Msg(s) in Frame: 8				
50631	106.935767	10.1.12.201	10.1.12.91	PEAK-CAN	over TCP	342 CAN Msg(s) in Frame: 8	19			
50651	106.976646	10.1.12.201	10.1.12.91	PEAK-CAN	over TCP	378 CAN Msg(s) in Frame: 9	5			-
50671	107.017667	10.1.12.201	10.1.12.91	PEAK-CAN	over TCP	342 CAN Msg(s) in Frame: 8				
50690	107.059020	10.1.12.201	10.1.12.91	PFAK-CAN	nver TCP	342 CAN Mep(s) in Frame: A				~
> Frame S > Etherne > Interne	50945: 414 byte et II, Src: Pe et Protocol Ve	es on wire (33 ak-Sys_00:22:0 rsion 4, Src:	12 bits), 414 bytes ca 1 (f0:73:ae:00:22:01), 10.1.12.201, Dst: 10.1	ptured (3312 bi Dst: LcfcHefe_ .12.91	ts) on ir 7c:3c:d5	terface 0 (50:7b:9d:7c:3c:d5)				î
> Transmi	ission Control	Protocol, Src	Port: 60768, Dst Port	: 50001, Seq: 7	74541, Ad	k: 1, Len: 360				
Y CAN 2.0	Ba/b Protocol I	Data - DEAN C.	the Encest		1					
Comp	olete CAN Data	size : Exp	nd Subtrees	Shift+Right						
CAN	CAN Frames: 10 Y CAN Data Storage with 10 TimeStamp-Low in µS:		apse Subtrees	Shift+Left						
Y CAN			nd All	Ctd+ Right						
т				curringin						
T	imeStamp-High	in µS Coll	apse All	Ctrl+Left						
S	AN-10: 0x122 Std. 11Bit Msg.	. App	ly as Column	Ctrl+Shift+I						
0	DLC: 8	Арр	ly as Filter							
	D0:0X25 D01:05	Pres	are a Filter		10X05					
	lastran los	In use Com	unration Citor							
	imeStamp-Low 1	in uS	versation miter							
	AN-TD: 0v122	Lin por Coli	rize with Filter							
S	td. 11Bit Msg.	Foll	w	•						
0	DLC: 8	C	6							
0	B0:0x23 DB1:0x	x21 DB: Cop	y		:0x03					
. ÷.		Sho	w Packet Bytes	Ctrl+Shift+O						¥
0030 0b	68 57 20 00 00	0 00 2/ Exp	nt Packet Bytes	Ctrl+Shift+X						^
• •		Wik	Protocol Page		-	II				~
v z c	AN Ethernet Gatewa	Filte	r Field Reference		Packets: 50946 · Displayed: 2602 (5.1%) · Dropped: 0 (0.0%) Profile: Defaul			tault 🚲		
		Prot	ocol Preferences	•	Oper	CAN Ethernet Gateway Protocol prefer	ences			
		Dec	ode As		Porte	50001-50002				
		Got	o Linked Packet		Show	all available info (Expert mode)				
		Sho	w Linked Packet in New Wind	low	Selec	t the used Protocol	•	UDP		
		L			Disab	le PEAK-CAN		тср		

If all is setup correct, you now could see the decoded CAN Data inside the Ethernet Packages.

~	CA	I Protocol Data - PEAK-System Format						
	Complete CAN Data size in Frame Length: 424							
	CAN Frames with CRC: 1							
		CAN FD Frames with CRC: 4						
	\sim	CAN Data Storage with 5 entrys						
		CAN-Channel: 0						
		✓ internal Flags:						
		- Extended ID						
		- Bit Rate Switch						
		- Error State Indicator						
		CAN MessageType: CAN FD with CRC Frame						
		TimeStamp-Low in µS: 1246898458						
		TimeStamp-High in μS: 228763						
		CAN-ID: 0x00001233						
		Ext. 29Bit Msg.						
		DLC: 15						
		DB00:0x27 DB01:0x43 DB02:0x12 DB03:0x67 DB04:0x41 DB05:0x24 DB06:0x12 DB07:0x89						
		DB08:0x46 DB09:0x18 DB10:0x97 DB11:0x21 DB12:0x89 DB13:0x26 DB14:0x01 DB15:0x65						
		DB16:0x01 DB17:0x64 DB18:0x35 DB19:0x81 DB20:0x45 DB21:0x26 DB22:0x43 DB23:0x95						
		DB24:0x86 DB25:0x23 DB26:0x49 DB27:0x79 DB28:0x81 DB29:0x97 DB30:0x23 DB31:0x87						
		DB32:0x71 DB33:0x32 DB34:0x35 DB35:0x74 DB36:0x72 DB37:0x73 DB38:0x37 DB39:0x36						
		DB40:0x31 DB41:0x32 DB42:0x66 DB43:0x34 DB44:0x35 DB45:0x76 DB46:0x31 DB47:0x32						
		DB48:0x36 DB49:0x72 DB50:0x73 DB51:0x33 DB52:0x65 DB53:0x37 DB54:0x34 DB55:0x35						
		DB56:0x72 DB57:0x31 DB58:0x38 DB59:0x36 DB60:0x35 DB61:0x33 DB62:0x34 DB63:0x34						



3.3 Fragmented CAN Packages

When using UDS, you could configure the CAN Data count (max.) inside an Ethernet Package on the Gateways. With TCP this is not possible. The size is fixed defined by the MTU of 1500 which the Ethernet Gateway(s) use. So, it could be possible that the Package will be send meanwhile the CAN Frame is not completely.

The TCP Protocol itself reassemble these packages.

4	*Ether	net					
Ei	e <u>E</u> dit	<u>V</u> iew <u>G</u> o	Capture Analyze Statis	tics Telephony <u>W</u> ire	eless <u>I</u> ools <u>H</u> elp		
4		i 💿 📘 🛅	🗙 🖸 🤇 🗢 🗢 😫	T 🛓 🚍 📃 🔍	Q, Q, II		
	peak-ca	n				Expr	ession +
No		Time	Source	Destination	Protocol	Length Info	^
	56752	22.113436	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	106 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 1	
	56808	22.135474	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	70 CAN Msg(s) in Frame: 41	
	56860	22.155842	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	106 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 1	
	56919	22.177007	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	142 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 2	
÷	56977	22.200861	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	178 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 3	
	57035	22.223298	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	142 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 2	
	57090	22.243835	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	70 CAN Msg(s) in Frame: 41	
	57136	22.263754	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	142 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 2	
	57195	22.287592	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	106 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 1	
	57246	22.309262	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	142 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 2	
	57303	22.331272	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	106 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 1	
	57357	22.351903	10.1.12.201	10.1.12.91	PEAK-CAN over TCP	106 CAN Msg(s) in Frame: 41 CAN Msg(s) in Frame: 1	
<							>
>	Frame	56977: 178 b	ytes on wire (1424 b	its), 178 bytes cap	otured (1424 bits) on in	nterface 0	^
>	Ether	net II, Src:	Peak-Sys_00:22:01 (ff	0:73:ae:00:22:01),	Dst: LcfcHefe_7c:3c:d5	(50:7b:9d:7c:3c:d5)	
>	Inter	net Protocol	Version 4, Src: 10.1	.12.201, Dst: 10.1.	12.91		
>	Transi	mission Contr	ol Protocol, Src Port	t: 60768, Dst Port:	50001, Seq: 494949, Ad	:k: 1, Len: 124	
×	[2 Rei	assembled TCP	Segments (1476 byte:	s): #56971(1460), #	56977(16)]		
	[Fr	rame: 56971,	payload: 0-1459 (1466	bytes)]			
	[Fr	name: 56977,	payload: 1460-1475 (1	L6 bytes)]			
	[Se	egment count:	2]				
	[Re	eassembled TC	P length: 1476]				
	[Re	eassembled TC	P Data: 002400800000	000000000000f7a824e	40000139600030000]		
Y	CAN 2	.0a/b Protoco	1 Data - PEAK-System	Format			
	Cor	mplete CAN Da	ta size in Frame Leng	th: 1476			
	CAI	N Frames: 41					
	> CAI	N Data Storag	e with 41 entrys				
Y	CAN 2	.0a/b Protoco	l Data - PEAK-System	Format			
	Cor	mplete CAN Da	ta size in Frame Leng	gth: 108			
	CAI	N Frames: 3					
	Y CAP	N Data Storag	e with 3 entrys				
		TimeStamp-Low	w in µS: 4155014758				
		TimeStamp-Hip	gh in μS: 5014				
		CAN-ID: 0x03	2				
		Std. 11Bit M	sg.				
		DLC: 3					
		DB0:0x32 DB1	:0x12 DB2:0x01				
		71	to use assessment				
		TimeStamp-Low	w in µ5: 4155015055				~
		TimeStamp-Hi	Ku 10 H2: 2014				•
00	150 00	0 00 t7 a8 7e	65 00 00 13 95 00 0	3 00 00 00 00	·····		\$
F	rame (17	8 bytes) Reas	sembled TCP (1476 bytes)				
0	7	CAN Ethernet Gate	eway Protocol (peak-can), 96	bytes		Packets: 58343 · Displayed: 436 (0.7%) · Dropped: 0 (0.0%) Prof	ile: Default



3.5 IP Frame Format Description

PCAN-Gateways allow the connection of various CAN busses over IP networks. For this CAN frames are wrapped in TCP or UDP messages packets and transmitted over the IP network from one device to another. If all the precautions are taken, you can also use a socket to send and receive CAN data via UDP or TCP.

Depending on the type of the CAN message and if the CRC feature is used, the CAN data is transmitted with a different structure in the IP frame. The possible data structures are shown in the following tables.

The values are stored in Network Byte order. The CAN data is stored as single bytes in ascending order. Whether you send or receive, the structure remains the same. With simple TCP/UDP implementations such as in embedded applications, it is possible to receive the TCP or UDP header in addition.

We recommend to study the extra document "*PCAN-Gateways - Developer Documentation*" which is available for download from the Download Section of the Product pages of the Ethernet Gateway.



Data structure for CAN 2.0 A/B frames

Length	Field Name	Meaning			
2 Byte	Length	This field specifies the total length of the packet including this Length field in bytes. The maximum length of a classic CAN frame is 0x24, decimal 36.			
2 Byte	Message Type	This field specifies the type of the message. The value 0x80 represents a classic CAN frame.			
8 Byte	Tag	Not used in the current version.			
4 Byte	Timestamp Low	Timestamp of CAN messages in μ s. The value has no effect on the transmission of frames. This information is			
4 Byte	Timestamp High	purely informative.			
1 Byte	Channel	Not used in the current version. Note: The CAN channel is determined by the route configuration.			
1 Byte	DLC	The Data Length Count (DLC) indicates the length of the CAN data in bytes.			
2 Byte	Flags	 The following flags are defined for this frame type: 0x01 - Message is a Remote Transmission Request (corresponds to bit 30 in the ID field) 0x02 - Message has an Extended ID (corresponds to bit 31 in the ID field) 			
4 Byte	CAN ID	Bit 0 - 28 ID Bit 29 Fixed value 0 Bit 30 RTR Bit 31 1 for Extended frame, 0 for Standard frame.			
8 Byte	CAN Data	This field always contains 8 x 8 data bits. Note: Use only as many bytes as the DLC indicates. All the following bytes are available but invalid.			



Data structure for CAN FD frames

Length	Field Name	Meaning			
2 Byte	Length	This field specifies the total length of the packet including this Length field in bytes. The length of the packet for CAN FD frames is variable in contrast to th classic CAN frame. Only as many bytes as necessary transmitted. The maximum length of a CAN FD frame 0x60, decimal 96.			
2 Byte	Message Type	This field specifies the type of the message. The value 0x90 represents a CAN FD frame.			
8 Byte	Тад	Not used in the current version.			
4 Byte	Timestamp Low	Timestamp of CAN messages in µs. The value has no effect on the transmission of frames. This information			
4 Byte	Timestamp High	purely informative.			
1 Byte	Channel	Not used in the current version. Note: The CAN channel is determined by the route configuration.			
1 Byte	DLC	The Data Length Count (DLC) indicates the length of t CAN data in bytes.	he		
2 Byte	Flags	 The following flags are defined for this message type: 0x02 - Message has an Extended ID (corresponds to bit 31 in the ID field) 0x10 - Message with Extended Data Length 0x20 - Message with activated Bit Rate Switch 0x40 - Message with activated Stream State Ladicates bit 			
4 Byte	CAN ID	Bit 0 - 28 ID Bit 29 Fixed value 0 Bit 31 1 for Extended frame, 0 for Standard frame.			
N Byte	CAN Data	This field contains the CAN data bytes. The number of bytes transmitted corresponds to the length specified in the DLC field.			



Data structure for CAN 2.0 A/B frames with CRC

Length Field Name Meaning				
2 Byte	Length	This field specifies the total length of the packet including this Length field in bytes. The maximum length of a classic CAN frame with additional CRC checksum is 0x28, decimal 40.		
2 Byte	Message Type	This field specifies the type of the message. The value 0x81 represents a classic CAN frame with additional CRC checksum.		
8 Byte	Tag	Not used in the	current version.	
4 Byte	Timestamp Low	Timestamp of CAN messages in μ s. The value has no effect on the transmission of frames. This information i		
4 Byte	Timestamp High	purely informative.		
1 Byte	Channel	Not used in the current version. Note: The CAN channel is determined by the route configuration.		
1 Byte	DLC	The Data Length CAN data in byt	n Count (DLC) indicates the length of the es.	
2 Byte	Flags	The following fl • 0x01 - Mess (correspondence) • 0x02 - Mess (correspondence)	ags are defined for this frame type: sage is a Remote Transmission Request ds to bit 30 in the ID field) sage has an Extended ID ds to bit 31 in the ID field)	
4 Byte	CAN ID	Bit 0 - 28 ID Bit 29 Fixed value 0 Bit 30 RTR Bit 31 1 for Extended frame, 0 for Standard fr		
8 Byte	CAN Data	This field always contains 8 x 8 data bits. Note: Use only as many bytes as the DLC indicates. All the following bytes are available but invalid.		
4 Byte	CRC32	CRC checksum. See the following chapter for details.		



Data structure for CAN FD frames with CRC

Length	Field Name	Meaning	
2 Byte	Length	This field specifies the total length of the packet including this Length field in bytes. The length of the packet for CAN FD frames is variable in contrast to the classic CAN frame. Only as many bytes as necessary are transmitted. The maximum length of a CAN FD frame with additional CRC checksum is 0x64, decimal 100.	
2 Byte	Message Type	This field specifies the type of the message. The value 0x91 represents a CAN FD frame with additional CRC checksum.	
8 Byte	Tag	Not used in the current version.	
4 Byte	Timestamp Low	Timestamp of CAN messages in μ s. The value has no effect on the transmission of frames. This information is purely informative.	
4 Byte	Timestamp High		
1 Byte	Channel	Not used in the current version. Note: The CAN channel is determined by the route configuration.	
1 Byte	DLC	The Data Length Count (DLC) indicates the length of the CAN data in bytes.	
2 Byte	Flags	 The following flags are defined for this message type: 0x02 - Message has an Extended ID (corresponds to bit 31 in the ID field) 0x10 - Message with Extended Data Length 0x20 - Message with activated Bit Rate Switch 0x40 - Message with set Error State Indicator bit 	
4 Byte	CAN ID	Bit 0 - 28 ID Bit 29 Fixed value 0 Bit 31 1 for Extende	d frame, 0 for Standard frame.
N Byte	CAN Data	This field contains the CAN data bytes. The number of bytes transmitted corresponds to the length specified in the DLC field.	
4 Byte	CRC32	CRC checksum. See the following chapter for details.	



3.5.1 Optional CRC32 Checksum

CAN frames can be transmitted over the IP network with an additional CRC32 checksum in the IP packet. It is created from DLC, flags, CAN ID, and data of the CAN frame together with a CRC start value and polynomial. This option was introduced with software version 2.8.1 for classic CAN frames and for CAN FD frames with software version 1.0.0 of the PCAN-Ethernet Gateway FD DR.

The CRC option for routes can be configured via the JSON interface or the INI file. For a valid connection with CRC option, the CRC settings of the Send and Receive route must match.

CRC32 example for an incoming message:



The colored values DLC, flags, CAN ID, and CAN data are used for the calculation.

There is no XOR with 0xFFFFFFF at the end of the calculation. The CRC32 result is added to the message in little endian byte order.



3.6 Notes about the License

Device drivers, the interface DLL, and further files needed for linking own applications are property of the PEAK-System Technik GmbH and should be used only in connection with a hardware component purchased from PEAK-System or one of its partners. If a CAN hardware component of third-party suppliers should be compatible to one of PEAK-System, then you are **not allowed** to use or to pass on the driver software of PEAK-System.

If a third-party supplier develops software based on the PCAN-Basic and problems occur during the use of this software, consult the software provider.