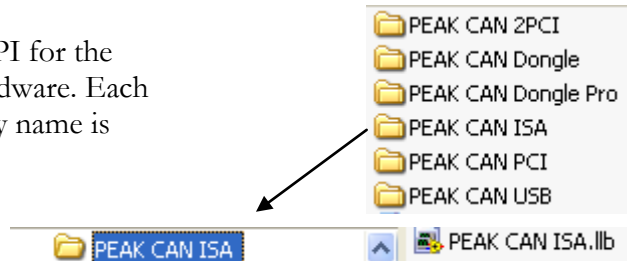


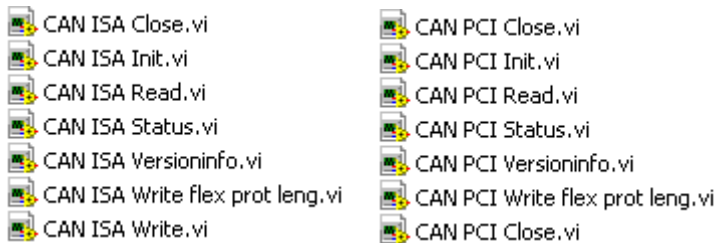
Easy to use LabView API for the PEAK CAN Light dev. driver.

The PEAK CAN Light driver interface for [LabView](#) makes it easy to write programs for CAN communication in LabView. The LabView API is developed for fast CAN bus usage while it is easy to use with a complete context help in English and German.

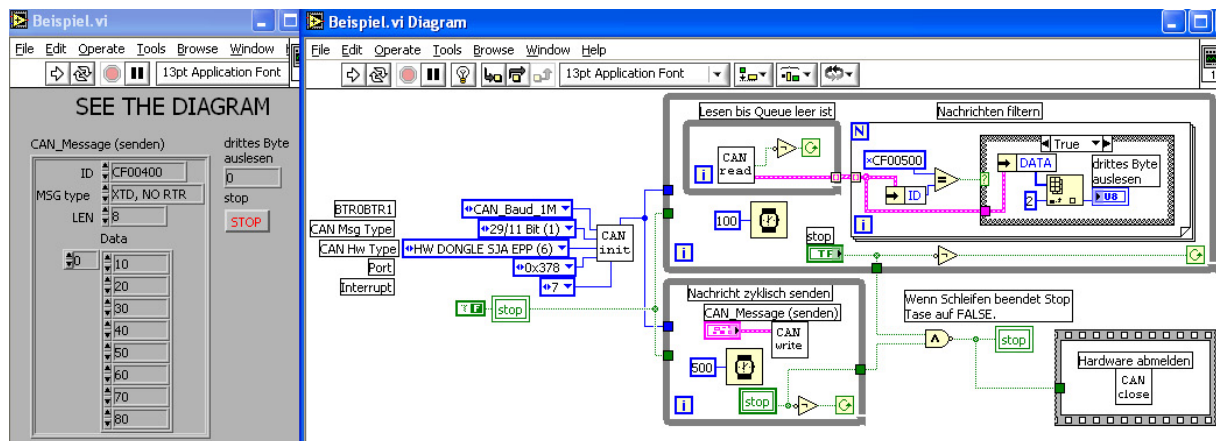
Included in the package is the [LabView](#) API for the [PEAK](#) CAN Light driver for different hardware. Each hardware has it's own folder. The directory name is the name of the physical interface.



All the libs contain VIs with the same functionalities and the same connectors except that they use different dll calls. Using this feature the customer will be able to write applications that offer the chance to use different hardware with the same program.



Every folder holds an example for the specific hardware. This example illustrates how to use the VIs and may give an easy intro.



Example: Send and receive of CAN data with LabView

All VIs are documented in the context help of LabView.

Context Help

BTR0BTR1(User) — Error Number
 BTR0BTR1 — BTR0BTR1 out
 CAN Msg Type — CANMsgType out
 CAN Hw Type — CANHwType out
 Port — IO_Port out
 Interrupt — Interrupt out

CAN_Init.vi

Dieses VI initialisiert die CAN Hardware.
 Alle Eingabe sind durch ENUM Variablen realisiert. Diese lassen sich in aufrufenden Diagrammen über die Funktion 'Create Constant' oder 'Create Control' erzeugen.

All input variables are realized as enum. use create control or create constant except 'BTR0BTR1 (User)' this is a hex value and will only be used if the BTR0BTR1 enum has the User value.

Eingabe:
 Input:
 BTR0BTR1: Baudratenregister. Einige Baudraten sind bereits vorgegeben. Eigene Einstellungen sind über den Eintrag 'User' in der ENUM und die Belegung der Variablen 'BTR0BTR1 (User)' einstellbar.
 Baud rate control

CAN Msg Type Einstellung ob 11 oder 29/11 bitt Identifier
 select 11 or 29/11 bit IDs

CAN Hw Type Auswahl der Hardware
 Select Hardware

Port E/A Adresse
 I/O address

Interrupt IRQ der Hardware
 Hardware IRQ

Ausgabe:
 Output:
 Error Number Die Bedeutung steht im headerfile des Treibers. Leider steht in der Light Version des Treibers die in der Vollversion enthaltene Funktion zur Ausgabe des Fehlertextes nicht zur Verfügung.
 Error number as discribed in the PEAK CAN .h file.
 The light version of the driver does not support error number to string translation.

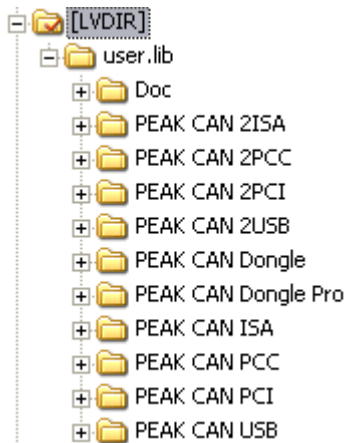
Die weiteren Ausgaben entsprechen den Eingaben. (Rückgabewerte der Treiber DLL)
 All other outputs are the same as the input values.

Example: LabView context help.

Before you can use the LabView driver API you must install the corresponding PCAN-Light interface driver. Remember to copy the hardware specific dll from the PEAK distribution to the /windows/system32 folder. Otherwise the needed dlls are not available. The VIs will work from [LabView](#) 8.5 and above for all MS Windows versions supported by PEAK-System.

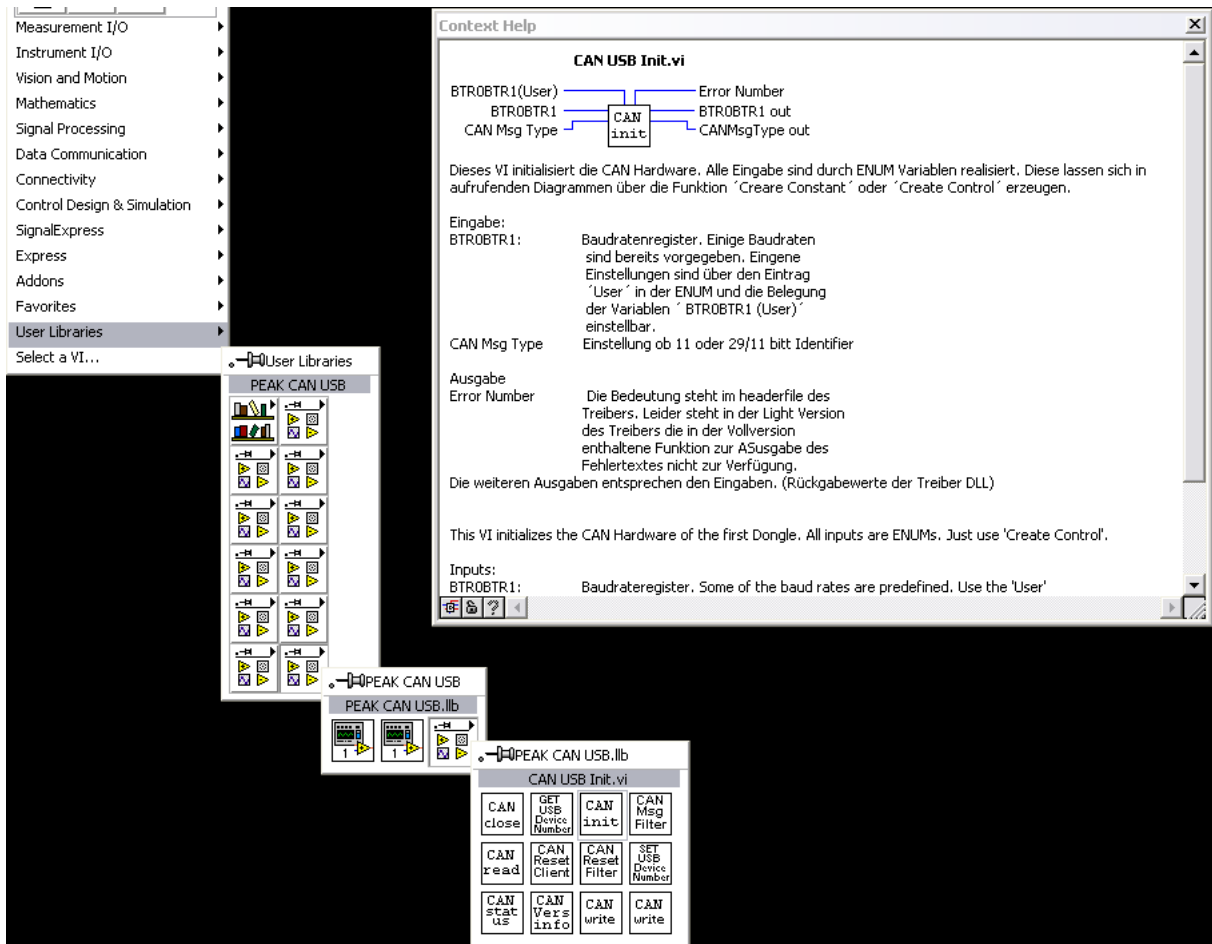
Installation of the LabView API:

Start the Setup.exe file. Setup will copy all the files to the LabView user.lib folder of the current LabView version, as shown in the next picture.



The API files are stored in the LabView user.lib folder

The VIs are access able from the LabView block diagram by choosing the user libraries and the specific hardware sub menu. If the context help is activated the VI documentation is displayed automatically.



Access the VIs from the block diagram

The [LabView](#) API is distributed by [PEAK-System Technik GmbH](#).

Contact:

Dipl.-Ing. (FH) Martin Kunze

labview@peak-system.com