

Introduction

USB-CAN-B is an industrial high-performance USB to CAN adapter, CAN-bus communication interface card and CAN protocol data analyzer. Integrating dual independent channels with electrical isolation and multiple protection circuits. Supports Windows and Linux systems, comes with drivers, CANTools related software, secondary development examples and tutorials. It can be connected to the PC or industrial control host via USB port to realize transceiver control, data analysis, collection and monitoring of CAN bus network. Compact in size and easy to use, which can be used for learning and debugging of CAN bus, and secondary development and integration into various industrial, power communication and intelligent control applications that require CAN bus communication.

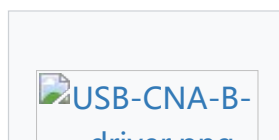
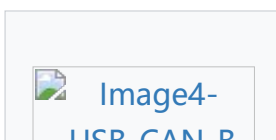
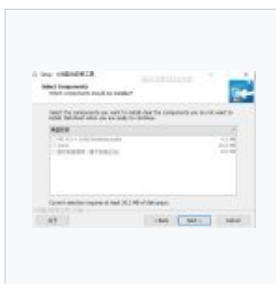


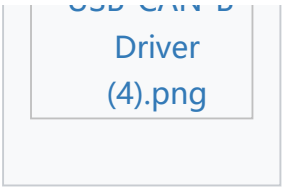
Working with Windows PC

Driver Installation

Connect USB-CAN-B intelligent interface module to the PC with a USB cable correctly. The Windows will then auto-run an installation wizard called "new hardware is found" after the hardware is detected. Select "Install the software automatically (Recommended)".

Click "Next" to continue;

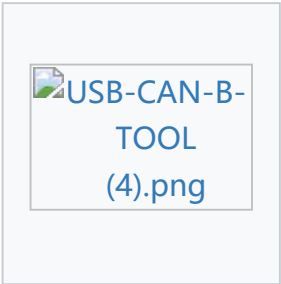
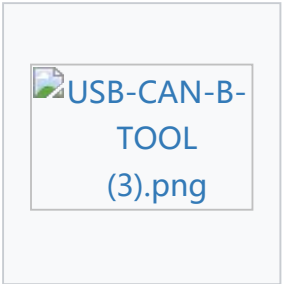
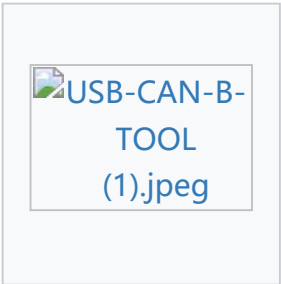
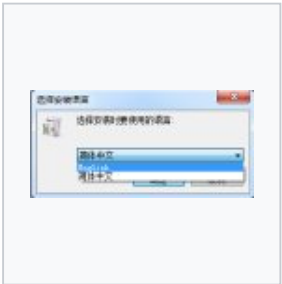




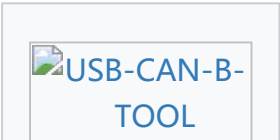
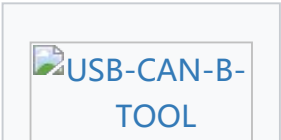
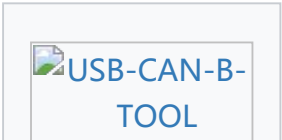
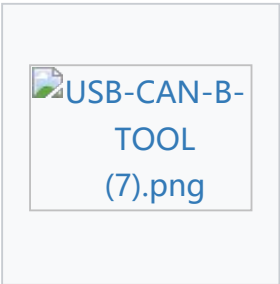
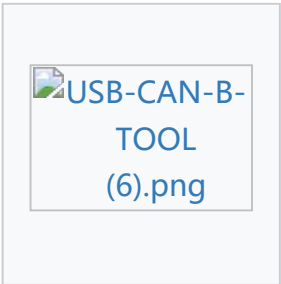
USB-CAN-B Tool Software

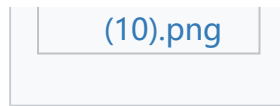
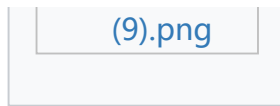
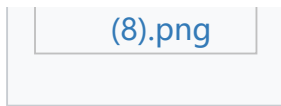
Download and Install

Download and unpack [USB-CAN-B TOOL_EN](#), then run and install USB_CAN_Tool.exe.



The first time to install the USB-CAN-B please make sure to check "run Engine".this installation package is needed for the application developed from the LABVIEW environment. The first time to install the USB-CAN-B please make sure to check "run Engine".this installation package is needed for the application developed from the LABVIEW environment.





Directions for Use

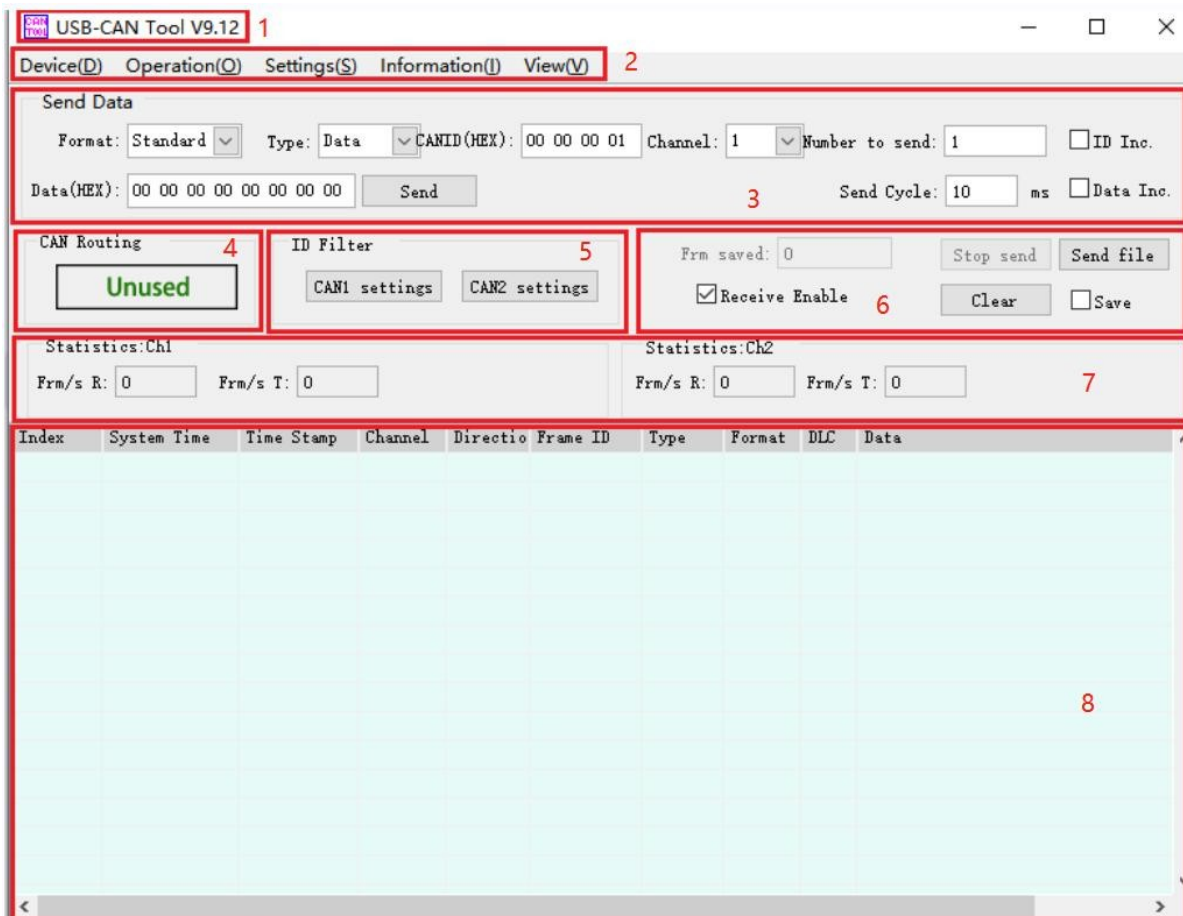
If you need a termination resistor, connect R+ with R-.



Must ensure that there are two 120Ω termination resistors connected to the bus. Otherwise, it will cause CAN-bus work abnormality.

Connect the USB interface of the device with the PC via USB.

Run "USB-CAN-B Tool.exe" as shown in the following figure:



1. Title Bar

When the device is turned off, USB-CAN Tool V9.xx will be displayed, where V9.xx represents the current program version number; when a USB-CAN adapter is successfully opened, the model, serial number, firmware version number, and brand name will be appended to the above name.

2. Menu Bar

Most of the functions of this tool are in the menu, and the functions can be realized by clicking the corresponding menu.

3. Sending Setting Section

This section contains the setting information related to CAN information transmission, among which the "ID increment" function is that when sending multi-frame data, the ID value of the next frame of data is 1 greater than the ID value of the previous frame; The sent 8 bytes of data from a 64-bit number according to the byte order from low to high, and each time the data is sent, it is automatically incremented by 1. This function cooperates with the "receive incremental check" function to test the frame loss rate during high-speed transmission. Note: This function is generally used for internal testing and is not recommended for users. When the data length is set to be less than 8 bytes, it will be automatically filled with 0 and filled to 8 bytes.

4.Can Relay

The CAN relay function is configured in the menu "Device Operation" -> "Relay Mode Options". When configured in relay mode, the relay status will be displayed here. If you want to turn off this function, you can close in the menu "Device Operation" -> "Relay Mode Options". Note: After the relay function is enabled, the baud rate and other parameter settings are invalid, but the sending and receiving, and intelligent filtering functions can be used normally.

5.Intelligent Filtering Function

The two channels can be set separately, and the filter is only for reception. After enabling the filtering function, the ID in the filter list or the ID in the ID segment will be displayed and received, and the data outside the filter list will be masked and discarded. With flexible settings for intelligent filtering, any ID or ID segment can be allowed or blocked from being received.

6.Control Button

"Stop sending" - stop the current sending operation; "Send file" - store the sent frame information in a file in a specific format, and send it in the order of the frames stored in the file; "Open CAN reception" - the CAN receive is enabled, otherwise the CAN receiving function is in the "suspended" state, at this time the host computer will no longer display the CAN bus data received by the USBCAN device, but the USBCAN device is still receiving the data on the CAN bus; "Clear" - clear The contents of the current data list; "Real-time storage" - saves the real-time data to a file.

Note: Sending a file may be consistent with the file format saved by the real-time storage function, and the saved file can be sent directly in the form of a file.

7.Statistics Area

Includes one-way rate and receives incrementing parity error counters for channel 1 and channel 2. "Frame rate R" - receiving rate, unit: fps (frames per second);

"frame rate T" - sending rate, unit: fps (frames per second); "Parity error" (internal functions are hidden by default) - —Receive data increment parity error counter, this value is the total number of dropped frames when receiving increment parity.

8. Data list

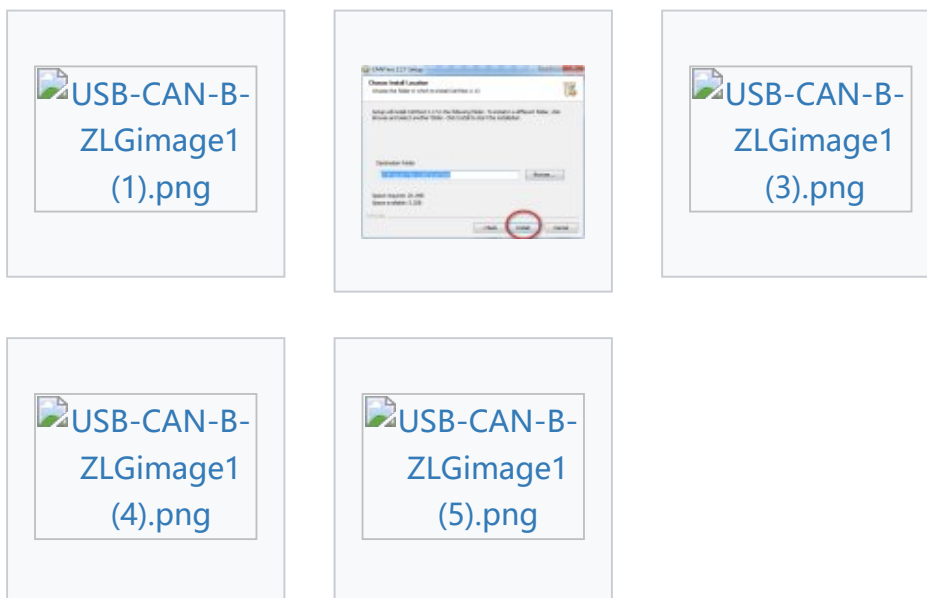
The list for displaying the data sent and received.

Please refer to the [USB-CAN TOOL Debugging Software User Manual](#) documentation for details.

Compatible with Mainstream CANTest Software

Installation Steps

1. After downloading [CANTest_Setup_V2.68](#), run CANTest_Setup.exe to start the installation.



2. Change the name of the file in CANTest installation directory from "[ControlCAN.dll](#)" to "[ControlCAN.dll.bak](#)", as a backup.

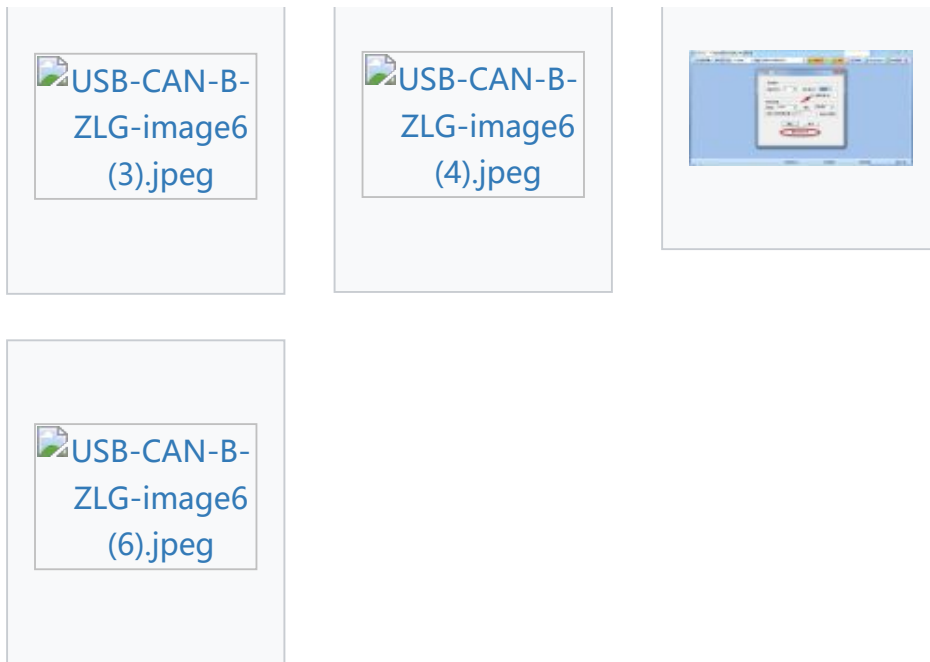


3. Copy "[ControlCAN.dll](#)" file to CANTest installation directory.



At this point, you have completed the software installation process.

4. Open and run CANTest.exe, after the correct installation of USB-CAN adapter\CANalyst-II analyzer driver of our company, the operation can be performed by CANTest software.



5. If you need back to use mainstream USB-CAN-B, you can change the backup file "ControlCAN.dll.bak" to "ControlCAN.dll" and cover to replace the existing file with the same name.

Compatibility Manual

Using our USB-CAN TOOL software can play the best performance of the equipment), when you use mainstream CANTest software to operate our USB-CAN-B, there may be a few functions that can hardly be used cannot compatible with mainstream CANTest software temporarily, but will not affect the use of normal functions.

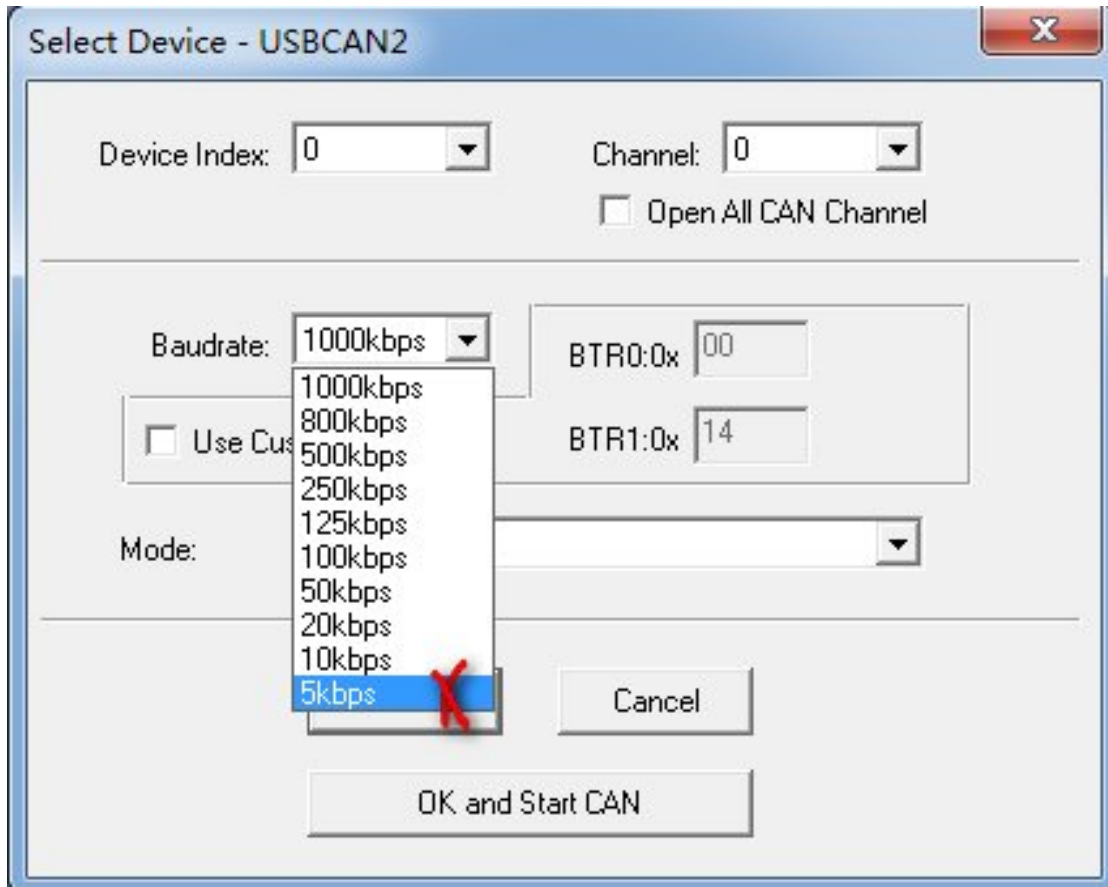
When using mainstream CANTest Software to operate our USB-CAN-B, Compatible part includes: Baud rate setting, Filter settings (partially compatible), Standard Frame \Extern Frame \ Remote Frame \ Data Frame transmission and reception, and so on.

Those temporary incompatible functions include:

1. When starting the software, in the "Select Device" dialog box, select the baud rate (The baud rate under 10kbps does not support) when starting up mainstream CANTest software, the "Select Device" dialog box filled in the initial parameters will pop up, which has a drop-down list of baud rates, including 10 kinds of common baud rates ranging from 5kbps to 1000kbps and others, As shown in the figure below:



Our USB-CAN-B does not support the baud rate under 10kbps, so the list 5kbps baud rate parameter is invalid:



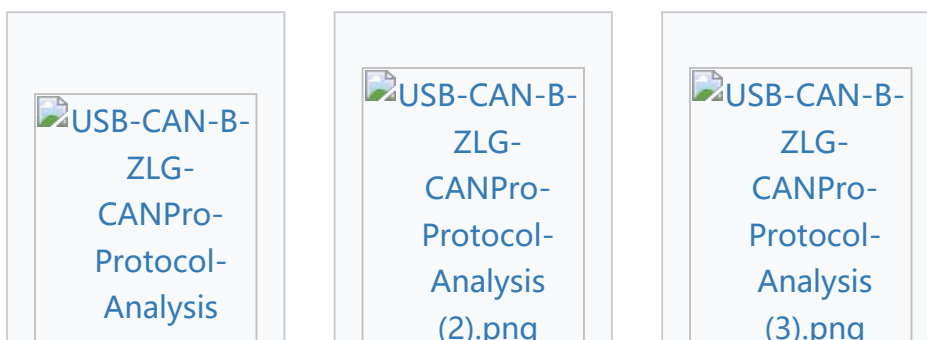
Note: if you carry on secondary development based on our USB-CAN-B, for the use of the interface function as well as the compatibility of the mainstream function library, please refer to < Bus Interface Adapter Interface Function Library User Instruction.pdf >

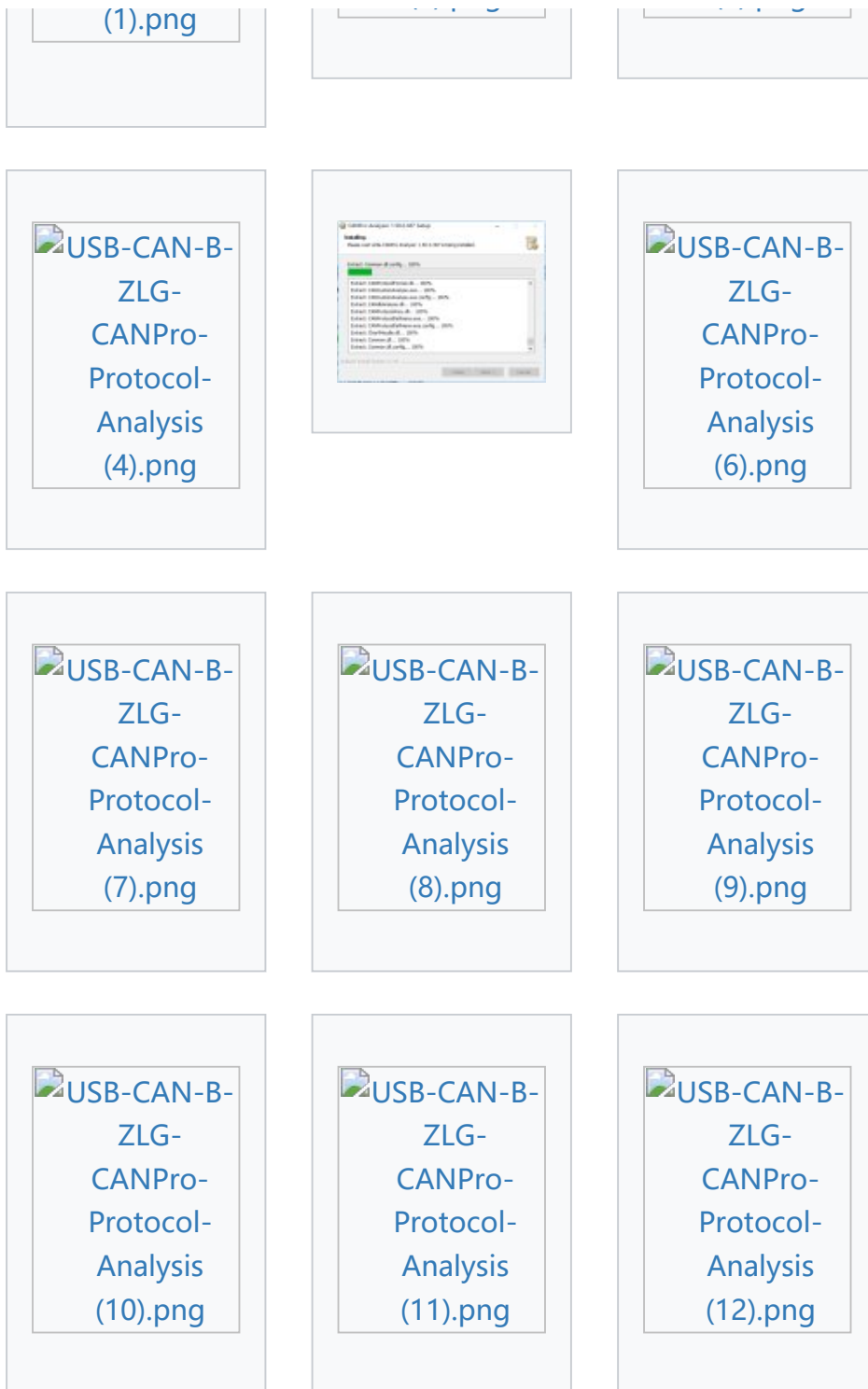
How to compatible with CANPro Protocol Analysis

The purpose of this document is to guide our clients on the first time buying and using the CANalyst-II analyzer to be compatible with PC software of CANalyst-II analyzer, namely CANPro Protocol Analysis Software.

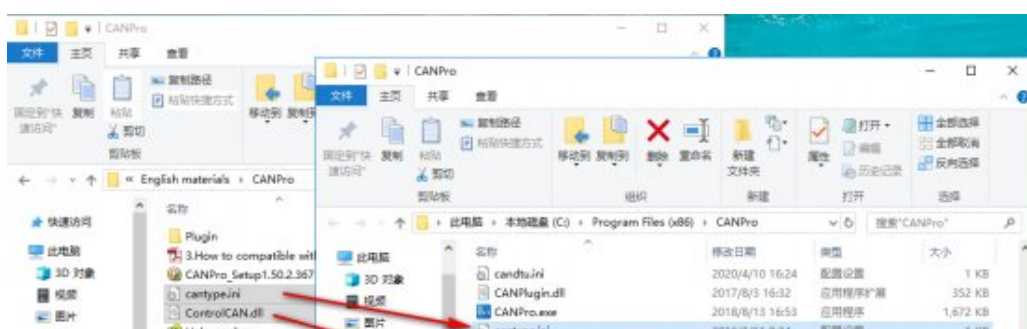
How to Compatible with CANPro Software

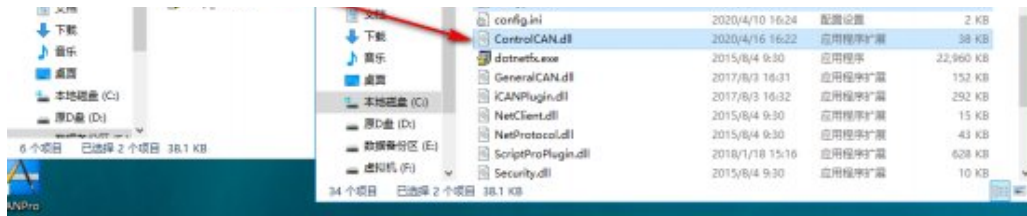
1. After downloading [CANPro_Setup1.50.2.367](#), unzip and install CANPro_Setup1.50.2.367.exe first, and then install three protocol analysis plug-in installation packages.





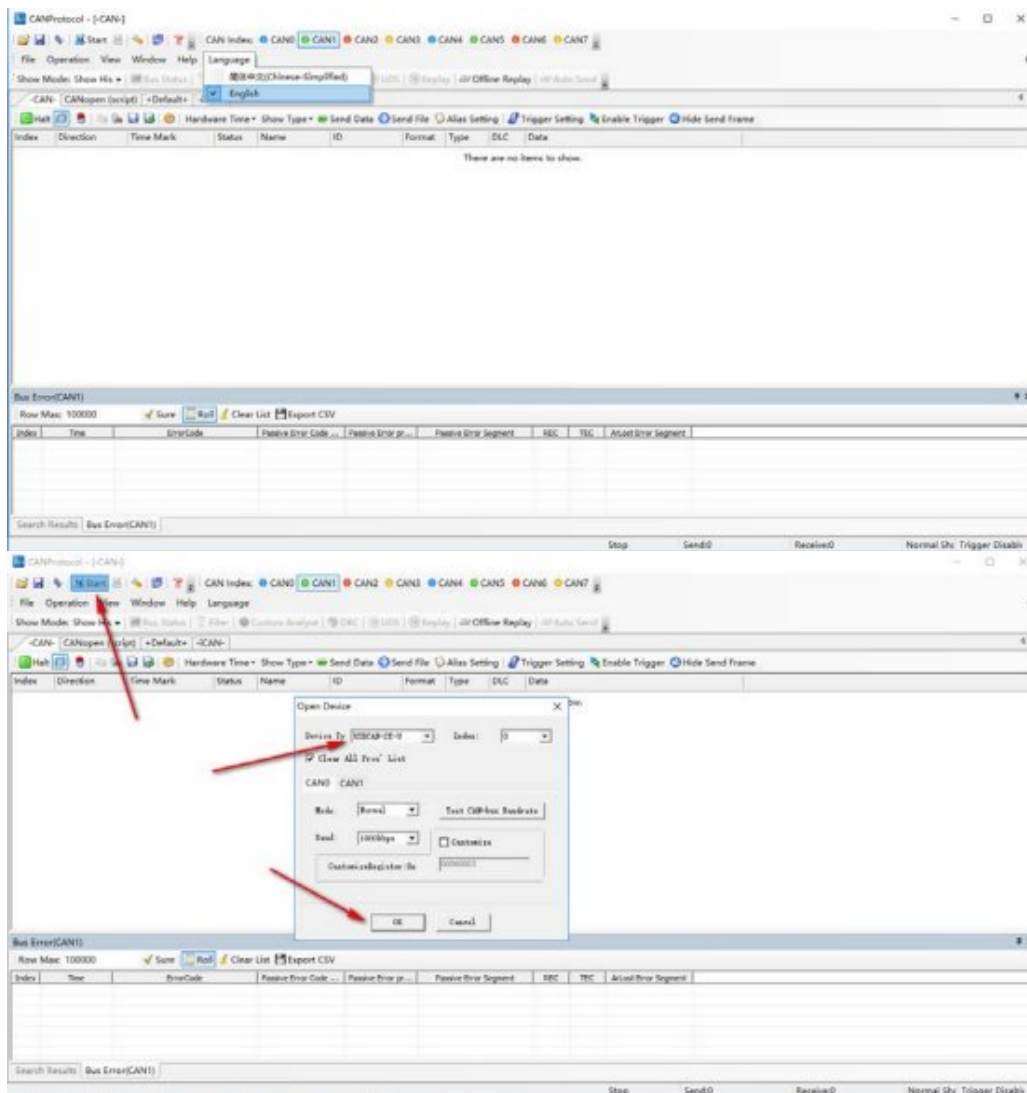
2. Copy "[ControlCAN.dll](#)" and "[cantype.ini](#)" files to CANPro installation directory, Overwrite file with the same name.



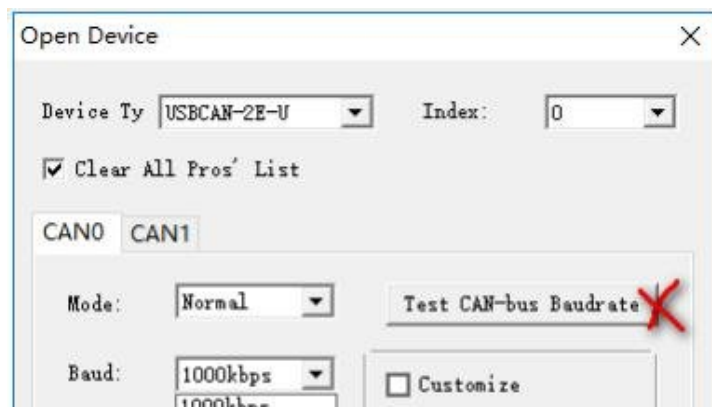


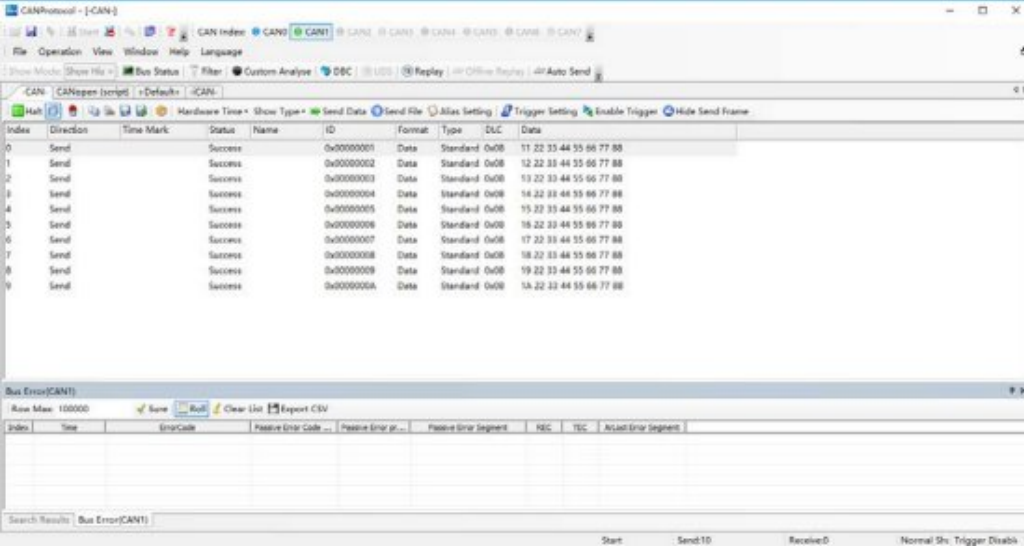
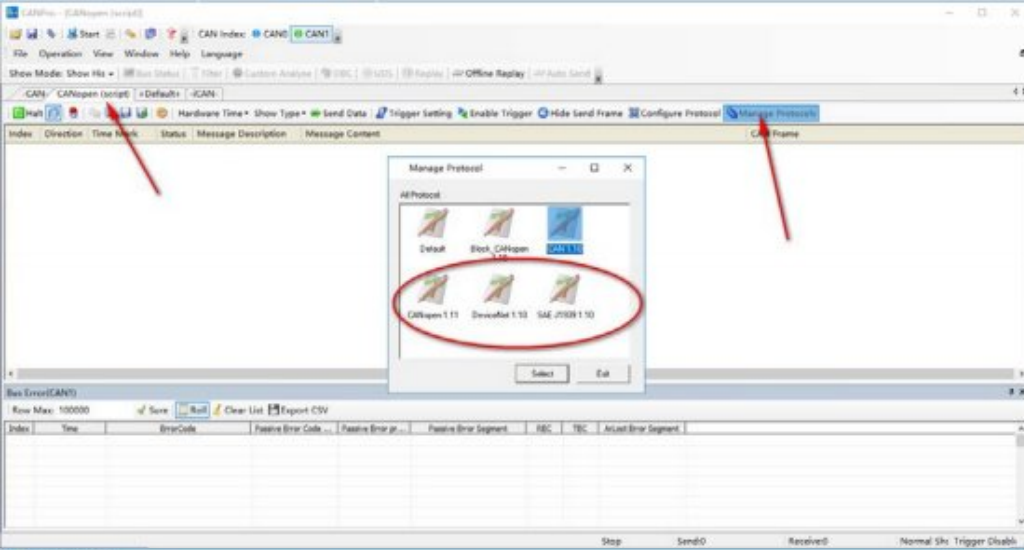
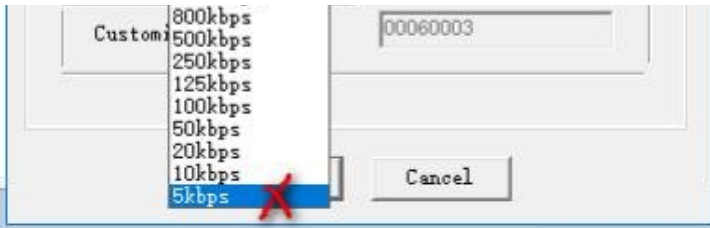
At this point, you have completed the software installation process.

3. Open and run CANPro.exe, after the correct installation of the CANalyst-II analyzer driver of our company, the operation can be performed by CANPro software.



⚠ 5K baud rate is not supported, Test CAN-bus Baudrate is not supported!





Working with Linux

Raspberry

c

Support access to mainstream Linux system motherboards, take Raspberry Pi 4B as an example:





```
cd ~
wget https://files.waveshare.com/upload/1/1c/USB-CAN-B_Code.zip
unzip USB-CAN-B_Code.zip
sudo chmod 777 -R USB-CAN-B_Code
cd USB-CAN-B_Code/Raspberry/c/
sudo make clean
sudo make
sudo ./hello_cpp
```

```
>>this is hello !
>>USBCAN DEVICE NUM:1 PCS
Device:0
>>Get VCI_ReadBoardInfo success!
>>Serial_Num:31F010318FF
>>hw_Type:CAN-Linux
>>Firmware Version:V3.39
>>
>>
>>
>>open deivce success!
>>Get VCI_ReadBoardInfo success!
>>Serial_Num:31F010318FF
>>hw_Type:CAN-Linux
>>Firmware Version:V3.39
Index:0000 CAN1 TX ID:0x00000000 Extend Data DLC:0x08 data:0x 00 01 02 03
04 05 06 07
Index:0001 CAN2 TX ID:0x00000001 Extend Data DLC:0x08 data:0x 00 01 02 03
04 05 06 07
Index:0002 CAN1 TX ID:0x00000002 Extend Data DLC:0x08 data:0x 00 01 02 03
04 05 06 07
Index:0003 CAN2 TX ID:0x00000003 Extend Data DLC:0x08 data:0x 00 01 02 03
04 05 06 07
Index:0004 CAN1 TX ID:0x00000004 Extend Data DLC:0x08 data:0x 00 01 02 03
04 05 06 07
Index:0005 CAN2 TX ID:0x00000005 Extend Data DLC:0x08 data:0x 00 01 02 03
04 05 06 07
Index:0006 CAN1 TX ID:0x00000006 Extend Data DLC:0x08 data:0x 00 01 02 03
04 05 06 07
```

python

```
cd ~
cd USB-CAN-B_Code/Raspberry/python/
sudo python3 python3-32bit.py
```

```
./ControlCAN.dll
Call VCI_OpenDevice successfully

Call VCI_InitCAN1 successfully

Call VCI_StartCAN1 successfully

Call VCI_InitCAN2 successfully

Call VCI_StartCAN2 successfully

CAN1 channel sent successfully

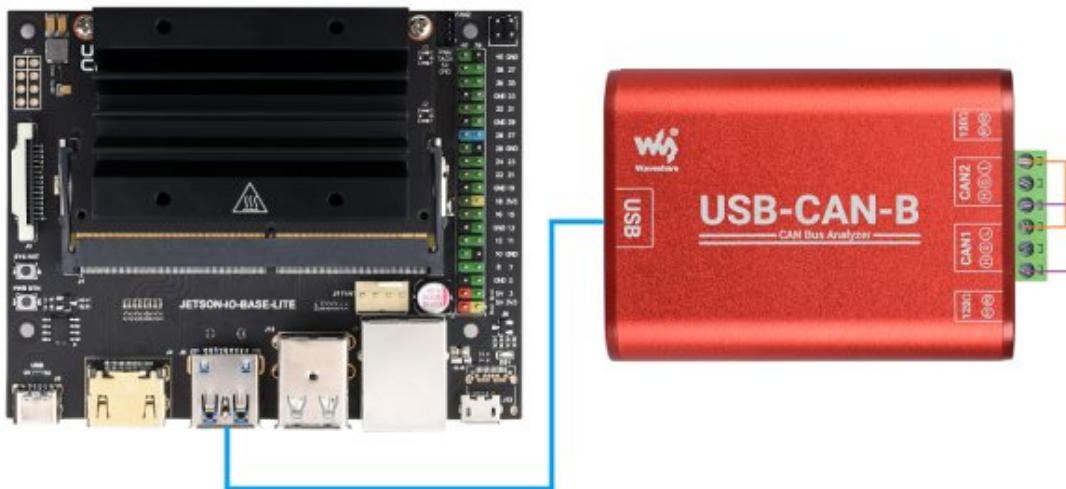
CAN2 channel received successfully

ID:
1
DataLen:
8
Data:
[1, 2, 3, 4, 5, 6, 7, 8]
```

Jetson Nano

c

Support access to mainstream Linux system motherboards, take JETSON-IO-BASE-A as an example:



```
wget https://files.waveshare.com/upload/1/1c/USB-CAN-B_Code.zip
unzip USB-CAN-B_Code.zip
sudo chmod 777 -R USB-CAN-B_Code
cd USB-CAN-B_Code/Jetson/c
sudo make clean
sudo make
```



```
sudo ./hello_cpp
```

```
jetbot@jetbot-desktop:~/USB-CAN-B_Jetson$ sudo make
g++ -o hello_cpp main.cpp /home/jetbot/USB-CAN-B_Jetson/libcontrolcan.so -lpthread
jetbot@jetbot-desktop:~/USB-CAN-B_Jetson$ sudo ./hello_cpp
>>this is hello !
>>USBCAN DEVICE NUM:1 PCS
Device:0
>>Get VCI_ReadBoardInfo success!
>>Serial_Num:31F010318FF
>>hw_Type:CAN-Linux
>>Firmware Version:V3.39
>>
>>
>>
>>open deivce success!
>>Get VCI_ReadBoardInfo success!
>>Serial_Num:31F010318FF
>>hw_Type:CAN-Linux
>>Firmware Version:V3.39
Index:0000 CAN1 TX ID:0x00000000 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0001 CAN2 TX ID:0x00000001 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0002 CAN1 TX ID:0x00000002 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0003 CAN2 TX ID:0x00000003 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0004 CAN1 TX ID:0x00000004 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0005 CAN2 RX ID:0x00000000 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF58C
Index:0006 CAN2 TX ID:0x00000005 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0007 CAN1 TX ID:0x00000006 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0008 CAN1 RX ID:0x00000001 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF598
Index:0009 CAN2 TX ID:0x00000007 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0010 CAN1 TX ID:0x00000008 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0011 CAN2 TX ID:0x00000009 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07
Index:0012 CAN2 RX ID:0x00000002 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5A3
Index:0013 CAN1 RX ID:0x00000003 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5AF
Index:0014 CAN2 RX ID:0x00000004 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5BB
Index:0015 CAN1 RX ID:0x00000005 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5C6
Index:0016 CAN2 RX ID:0x00000006 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5D2
Index:0017 CAN1 RX ID:0x00000007 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5DE
Index:0018 CAN2 RX ID:0x00000008 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5E9
Index:0019 CAN1 RX ID:0x00000009 Extend Data DLC:0x08 data:0x 00 01 02 03 04 05 06 07 TimeStam
p:0x006DF5F5
```

python

```
cd ~
cd USB-CAN-B_Code/Jetson/python/
sudo python3 python3-64bit.py
```

```
./ControlCAN.dll
Call VCI_OpenDevice successfully
Call VCI_InitCAN1 successfully
Call VCI_StartCAN1 successfully
Call VCI_InitCAN2 successfully
Call VCI_StartCAN2 successfully
CAN1 channel sent successfully
CAN2 channel received successfully
```

```
ID:
1
DataLen:
8
Data:
[1, 2, 3, 4, 5, 6, 7, 8]
```

Resource

Documentation

- [USB-CAN-B Manual](#)
- [Interface Function Library User Instruction](#)
- [USB-CAN TOOL Debugging Software User Manual](#)

Software

- [USB-CAN-B TOOL_EN\(V9.15\)](#)
- [USB-CAN-B Driver_installation_\(Tool_SetupV1.40\)](#)
- [USB-CAN-B Usb_Drivers\(Manual installation\)](#)
- [ControlCAN.DLL file](#)

Demo

- [USB-CAN-B_Code](#)
- [Linux_development_demo](#)

FAQ

Question:How to communicate with other CAN devices?

Answer:

Take [2-CH CAN FD HAT](#) as an example, the test steps are as follows:

- Hardware connection:





- Software setup

```
sudo apt-get install can-utils  
sudo ip link set can0 up type can bitrate 1000000  
candump can0  
cansend can0 000#11.22.33.44
```



Support

Technical Support

If you need technical support or have any feedback/review, please click the **Submit Now** button to submit a ticket, Our support team will check and reply to you within 1 to 2 working days. Please be patient as we make every effort to help you to resolve the issue.

Working Time: 9 AM - 6 AM GMT+8 (Monday to Friday)

[Submit Now](#)