USB-CAN-A

From Waveshare Wiki
Jump to: navigation, search

Overview

USB-CAN-A

(https://www.waveshare.com/usb-can-a.htm)

USB To CAN
CAN

Features

- Supports CAN2.0A (standard frame) and CAN2.0B (extended frame).
- CAN baud rate is configurable in the range of 5Kbps-1Mbps.
- Supports 4 working modes: normal mode, loopback mode, silent mode, silent loopback mode.
- Supports multiple CAN data sending modes: single frame, multiple frames, manually, regular and cyclic sending.
- Supports multiple CAN data receiving modes: can be configured to only receive data from a certain ID, or specify an ID to automatically answer the configured data.
- Data can be saved as TXT or Excel.
- Supports CAN bus detection for checking status.
- Sending/receiving CAN data with a time scale, allows sequentially displaying.
- Baud rate of the USB virtual COM port is configurable in the range of 9600 ~ 2000000bps (2000000bps by default).
- Supports set working parameters by configuration software or serial command, can be saved after power off.
- Adopts STM32 chip solution, stable and reliable communication.
- Onboard TVS (Transient Voltage Suppressor), effectively suppresses surge voltage and transient spike voltage in the circuit.
- Comes with the host software for Windows system, easy to use.

- Easy secondary development, just need to modify the sending and receiving commands.

Primary Function

(/wiki/File:USB-CAN-A-details-5.jpg)

Using in Windows System

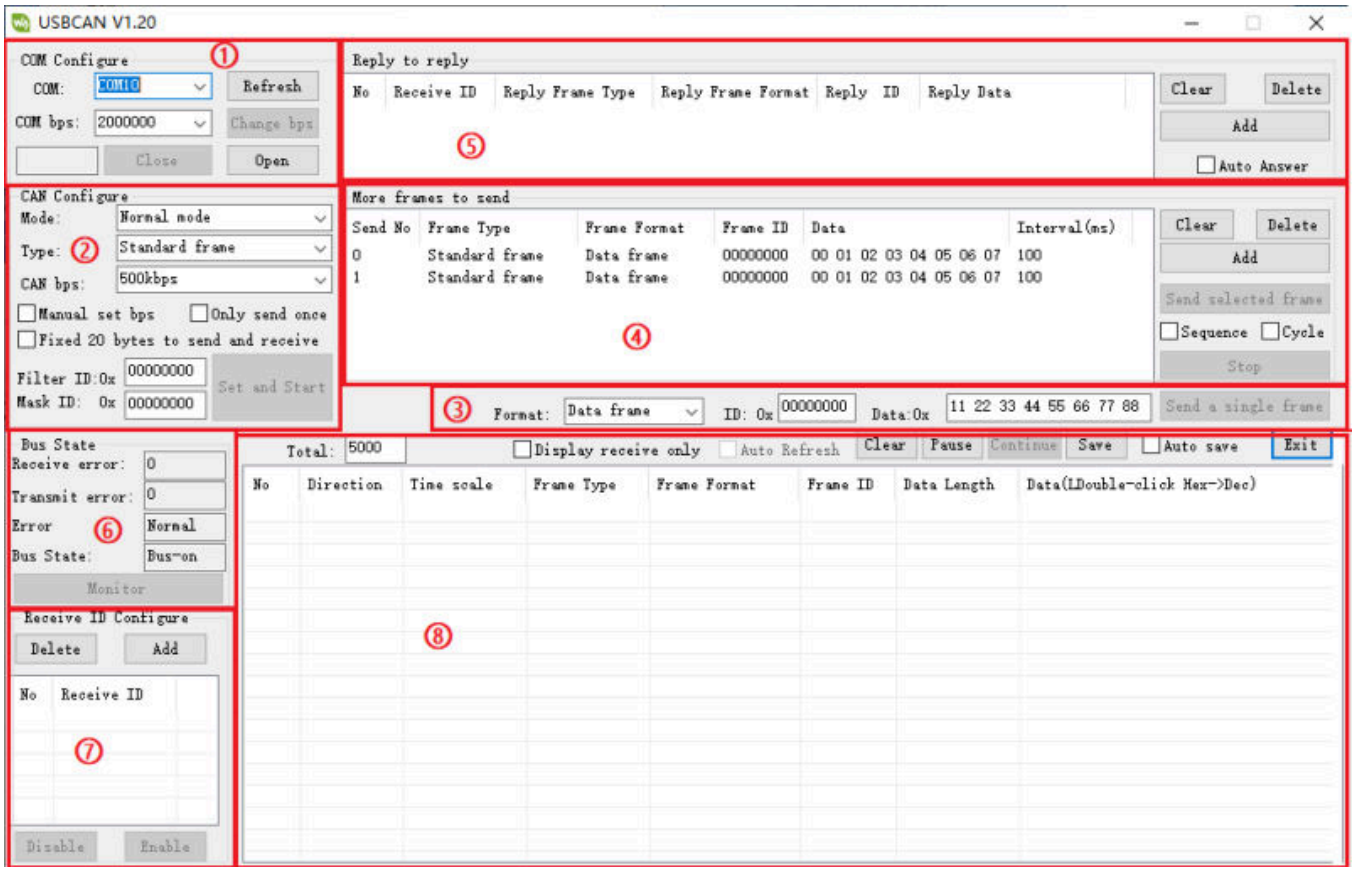
Install USB Driver

- Download CH341SER driver (https://files.waveshare.com/upload/c/c4/CH341SER.zip), double click "CH341SER.EXE" file, and click to install. Click "Finish" after installation is complete.

How to Use USB Config Tool

Open CAN config tool:

- Download USB-CAN-A_TOOL_1.2 tool (https://files.waveshare.com/upload/8/8a/USB-CAN-A_TOOL_EN.zip) and unzip, open "USB-CAN.exe".



(/wiki/File:USB-CAN-A_WIN01.png)

Config functions description:

① Port Setting:

After inserting the "USB To CAN" into the computer, the COM port of the computer can be found automatically. Choose the port, and it can be opened or closed (similar to the serial port assistant). The baud rate is 2,000,000bps by default.

> 2M of baud rate is compatible with most of CAN rate (the serial port baud rate almost is 2 times the CAN rate). Generally, a 2M baud rate is required by default. To revert to the 2M serial port baud rate, simply set the CAN rate back to the corresponding speed of the external CAN device (support up to 1M). If another baud rate was previously configured, it can be reset to restore the default 2M baud rate.

(/wiki/File:USB-CAN-A-faq.png)

If you want to modify the baud rate, you can refer to the following content:

1. Connect to the converter by a USB cable and pay attention to the flickering of TX and RX indicators at the same time:

```
Flash 1 time, the baud rate of the corresponding serial port is 2000000bps
Flash 2 times, the baud rate of the corresponding serial port is 1228800bps
Flash 3 times, the baud rate of the corresponding serial port is 115200bps
Flash 4 times, the baud rate of the corresponding serial port is 38400bps
Flash 5 times, the baud rate of the corresponding serial port is 19200bps
Flash 6 times, the baud rate of the corresponding serial port is 9600bps
```

2. Open the software and choose the corresponding COM port and baud rate. Click the "Open" button. (If the TX and RX indicators flash 1 time when the converter is connected to the USB cable and powered on, we should choose the baud rate of 2,000,000bps.)

3. Click the "Open" button.

4. Click the Change Baud Rate button, and a dialog box will pop up. After we select the baud rate we are going to set in the baud rate combo box, click Change; if you do not want to change, click the Cancel button.

5. For example, if we want to set the baud rate as 9600bps:

- Set "9600" in the baud rate combo box.
- Click "Change", and pay attention to the RX and TX indicators.
- The indicators will flash 6 times corresponding to the baud rate of 9600.
- As the baud rate is changed, you need to close the port and open it again after selecting the changed baud rate.

6. Click the "Close" button.
7. Set the baud rate as the changed one.
8. Click the "Open" button again.

② CAN Setting

The operating modes include: normal, loopback, silent, and silent loopback modes.

- Normal mode: the normal operating mode of CAN can send and receive the data to the bus.
- Loopback mode: the data can be sent to the CAN bus, give feedback to the internal reception area, and rejects the data from the external pin. This mode is for self-testing.
- Silent loopback mode: this mode is for "hot testing", which is self-testing. It can be tested like the loopback mode but does not affect the CAN bus system.
- Frame type: standard frame (CAN2.0A 11-bit ID), extended frame (CAN2.0B 29-bit ID)
- CAN baud rate: you can directly choose the common ones for CAN communication: 1M, 800K, 500K, 400K, 250K, 200K, 125K, 100K, 50K, 20K, 10K, 5K. *You can choose one if the above baud rate that can be directly set is different from your CAN device.
- Filter ID and Mask ID: Both are hexadecimal data, the lower 11 bits of filter ID and mask ID are valid in a standard frame (range: 0x00000000~0x000007ff), and the lower 29 bits of filter ID and mask ID in the extended frame are valid (range 0x00000000~0x1fffffff).
- Fixed 20-byte transceiver: There are 2 conversion protocols inside the CAN converter, one is a communication protocol that can be lengthened, and the other is a fixed 20-byte communication protocol. If selected, it will be a fixed 20-byte communication protocol. After selecting the variable protocol communication, a custom baud rate dialog box will pop up. The calculation formula of the CAN baud rate at the top, and set the phase buffer 1, phase buffer 2, and prescaler at the same time.
- Disable automatic message retransmission: CAN communication is generally automatically retransmitted if the transmission is unsuccessful. If the data is sent cyclically, you can set it to disable the automatic retransmission of the message. Click Set and start CAN communication.

③ Send single-frame data field:

The frame format includes data frame and remote frame, the frame ID is hexadecimal data, the frame ID range for the standard frame is 0x00000000~0x000007fff, and the frame ID range for the extended frame is 0x00000000~0x1fffffff. The data sent is also hexadecimal data, the data in the figure are 0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88. Click the send single frame button to send the frame data to the CAN bus.

④ Multi-frame sending data field:

- Add: you can add one sending frame data under the multi-frame sending data area.
- Delete: delete the chosen line.
- Clear: clear all the data in the multi-frame sending data area.
- Interval time: the interval sending time of two consecutive frames in the multi-frame sending area (milliseconds)
- Send the chosen frame: it will send the chosen row in this area.
- Sequence: it will turn to the next frame after clicking to send the chosen row.
- Cycle: it will send the chosen frame in the cycle after clicking to send the data.
- Stop: cancel the command of sending the multiple frames.

⑤ Auto Response field

- Add: add one auto-sending frame data in the Auto Response Area.
- Delete: delete the chosen row.
- Clear: clear all the data in this area.
- Auto response: it will automatically reply to the corresponding frame ID and data after receiving the ID.

⑥ CAN Bus Status

Mainly used to check the CAN communication status of the USB to CAN device.

⑦ Configure the Accept ID field

- Add button: an auto-response will be added under the selected row in the configuration acceptance area.
- Delete button: the selected row will be deleted.
- Enable button: After clicking this button, the converter will only upload the ID set in the configuration acceptance ID area, and other IDs will not be uploaded to the computer.
- Disable button: After clicking this button, the converter will upload all ID data to the computer.

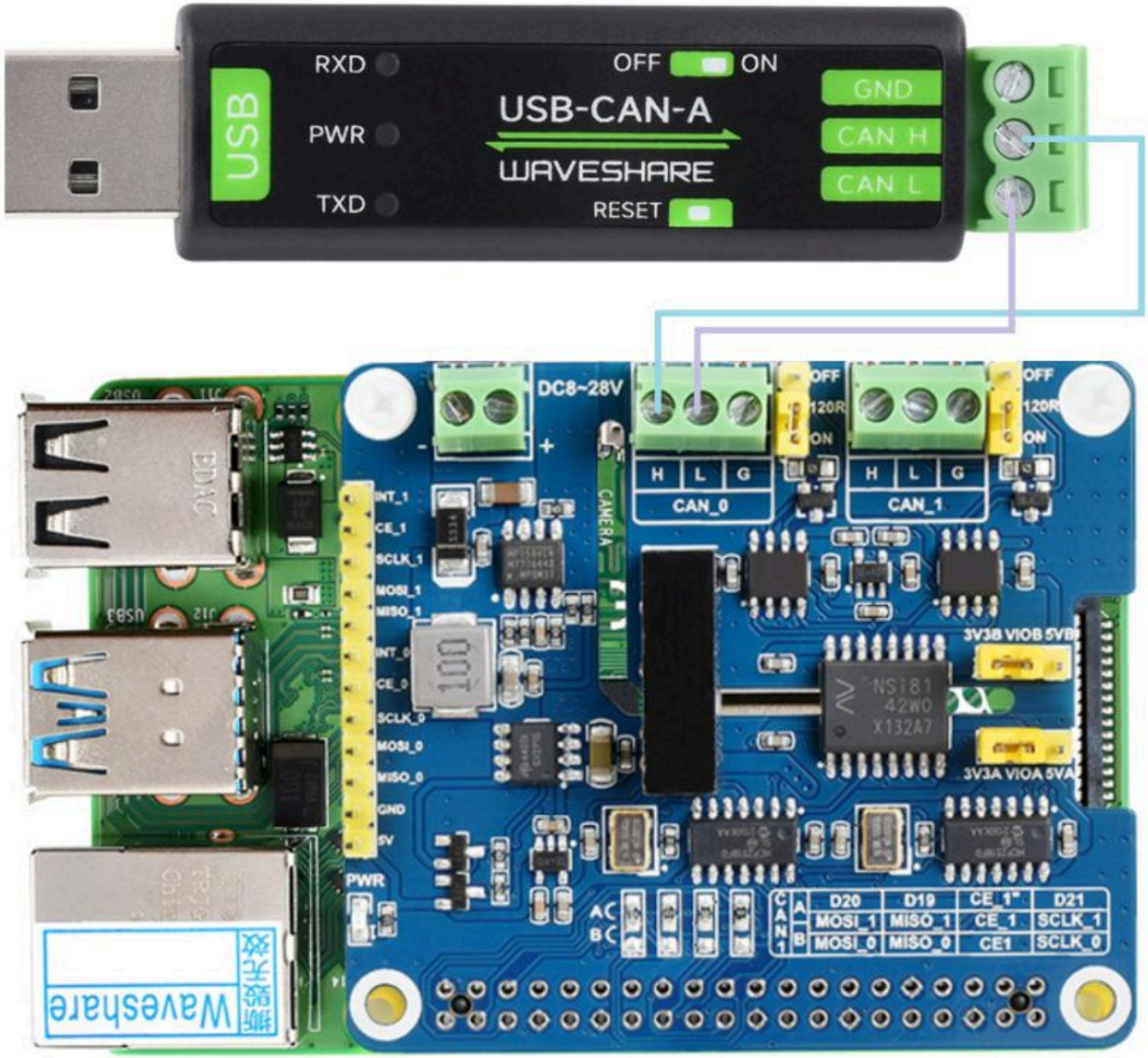
⑧ Send and receive data display field:

- Clear: clear all data on the display for sending and receiving data.
- Pause: pause to display other data in the send and receive display area.
- Continue: continue to display sending and receiving data.
- Save: you can save the data in the sending and receiving data buffer in two formats, excel or text.
- Do not display the sending frame: only the received data can be displayed in the sending and receiving data area.
- Automatic refresh: when only the accepted data is displayed, you can choose to automatically refresh. At this time, the data is refreshed in real-time instead of continuously increasing the column display. This function can summarize the ID data summarized by CAN and select any row.

Hardware Test

2-CH CAN FD HAT Connection Test

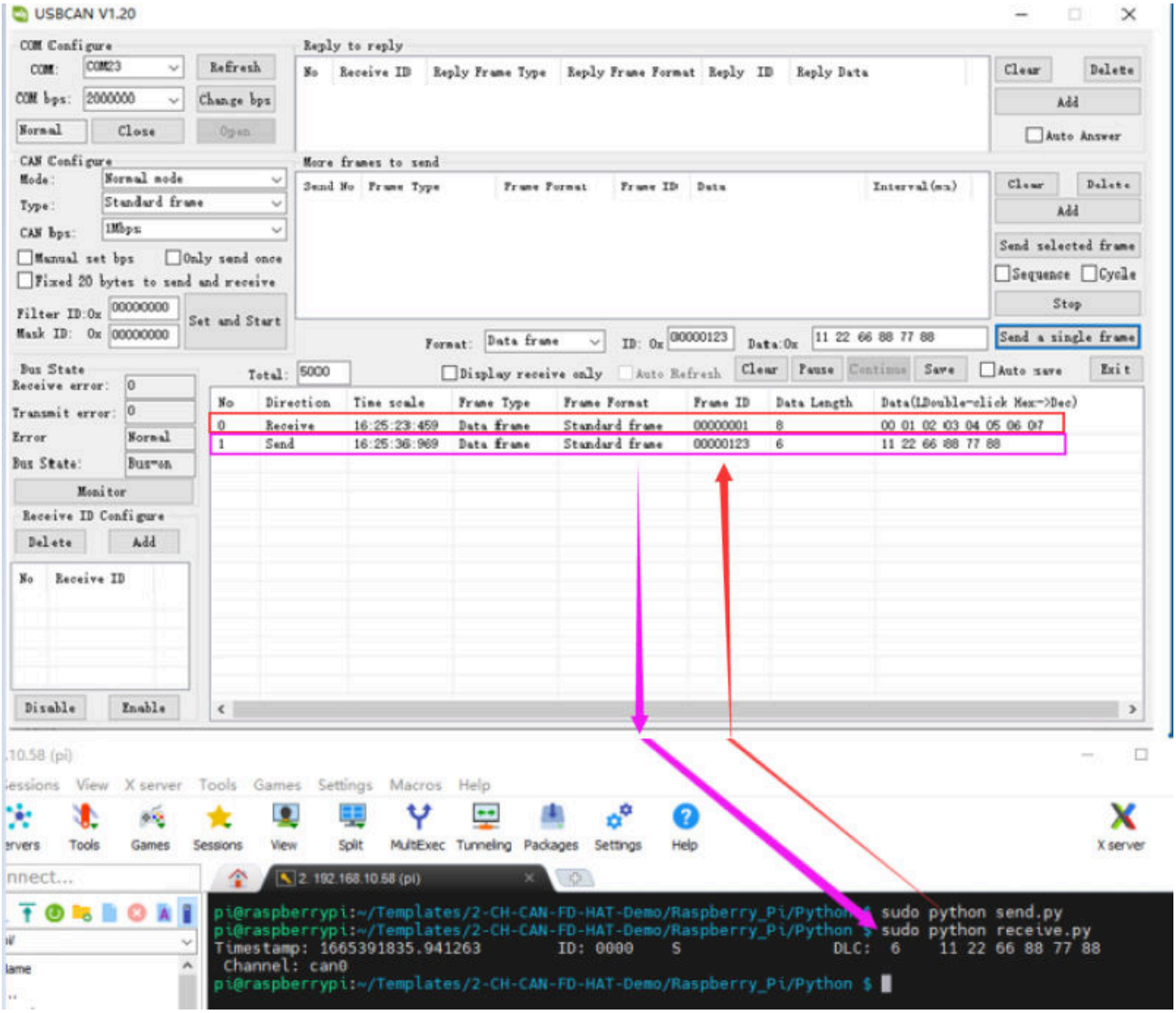
- Connect USB-CAN-A and the 2-CH CAN FD HAT (https://www.waveshare.com/2-ch-can-fd-hat.htm) with Dupont cables. Connect A-->A and B-->B, and connect the USB terminal of USB-CAN-A to the computer USB port:



(/wiki/File:USB-CAN-A-USB.jpg)

- Open USB-CAN debug tool on the computer and its default baud rate is 2M. Open the corresponding port, select the normal mode, the standard frame and the corresponding rate of the demo is 1Mbps (USB-CAN-A supports 1M). Set and boot it. Then, you can see the bidirectional data transmission between 2-CH CAN FD HAT (https://www.waveshare.com/2-ch-can-fd-hat.htm) and USB-CAN-A.

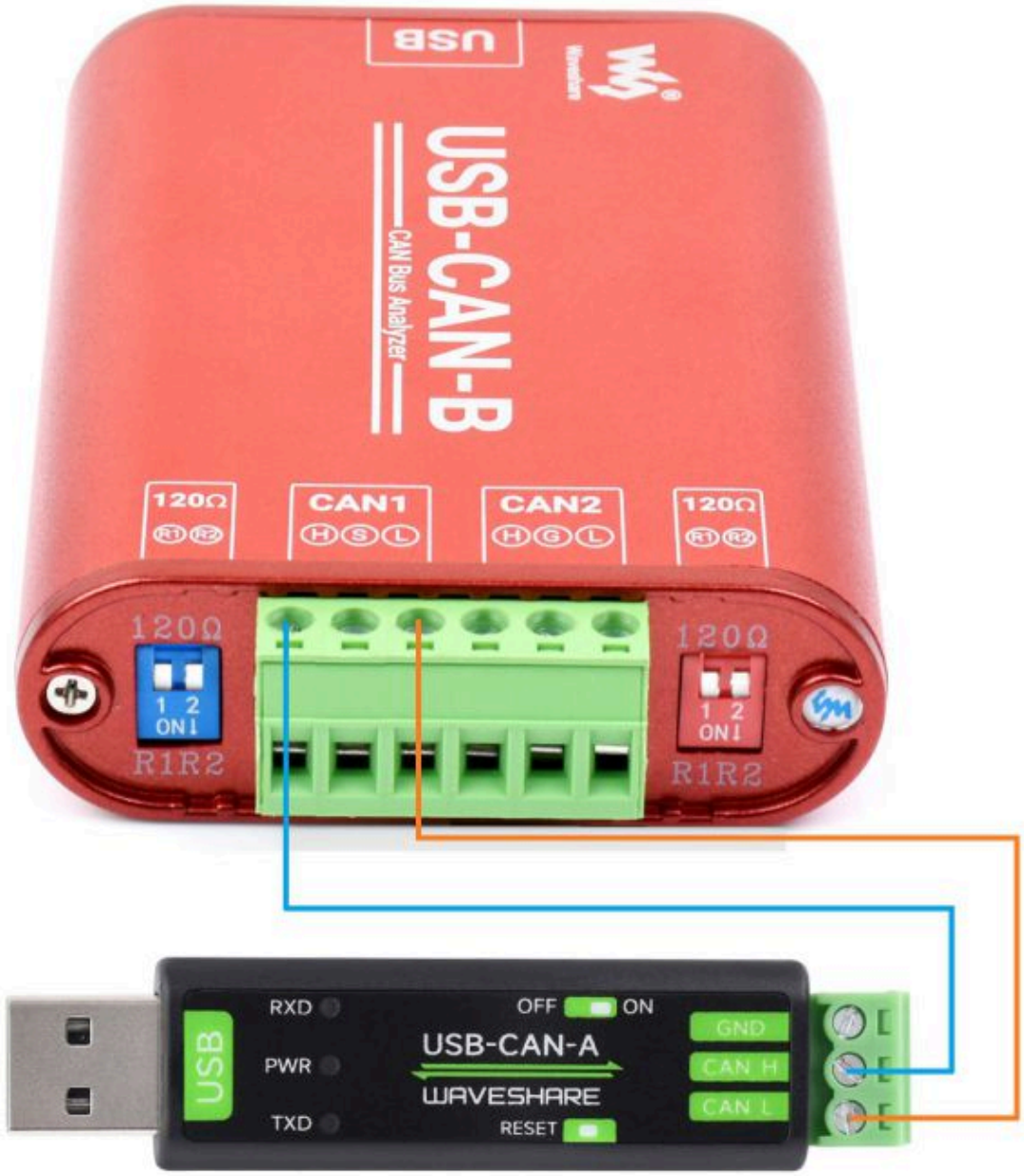
As shown below:



(/wiki/File:USB-CAN-A_RS485-CAN-HAT.png)

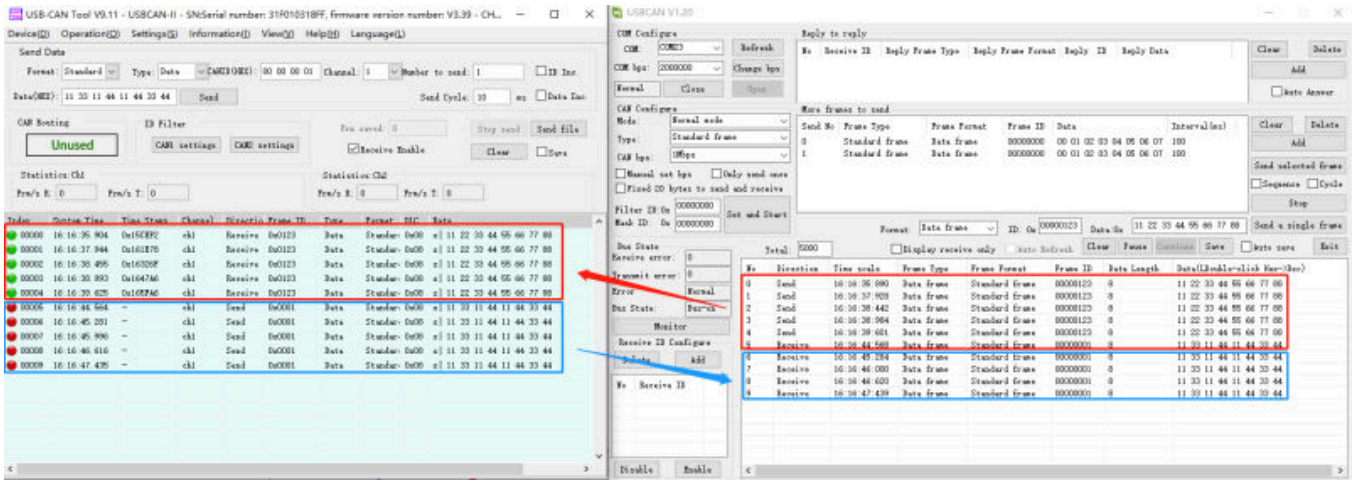
USB-CAN-B Connection Test

- Connect USB-CAN-A and 2-CH CAN FD HAT (https://www.waveshare.com/2-ch-can-fd-hat.htm) by H-L and L-L with Dupont cables. Then, turn on their 120 Ohm resistive switches.



(/wiki/File:USB-CAN-A-B.jpg)

- Open the corresponding config tool, USB-CAN-A sends data, and USB-CAN-B can receive it; USB-CAN-B sends data, and USB-CAN-A can receive it too. That means the test is successful.

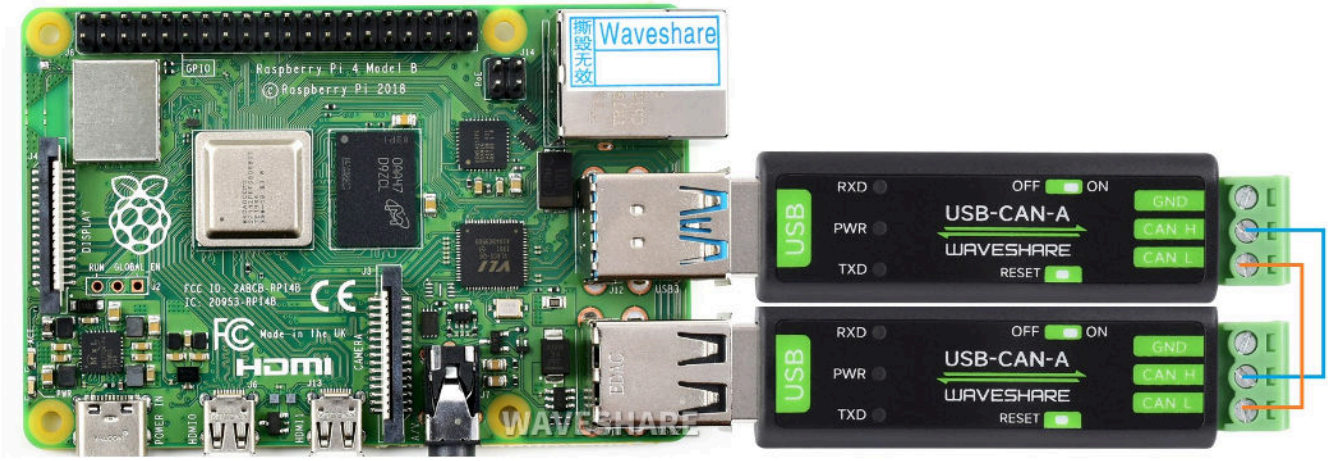


(/wiki/File:USB-CAN-A_USB-CAN-B.png)

Using in Linux System

Hardware Test

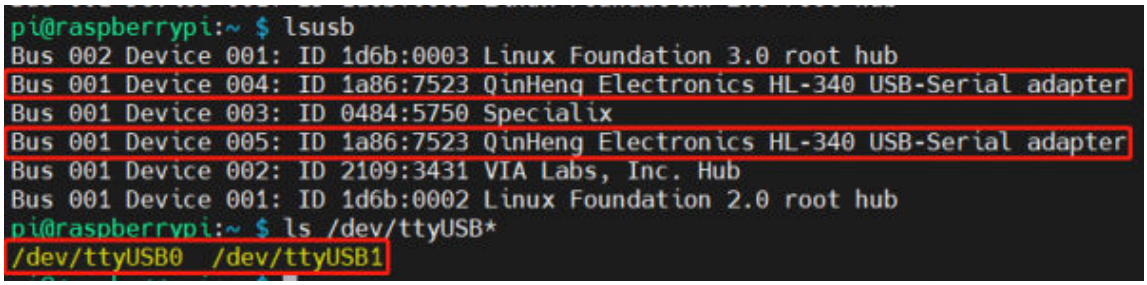
- The L and H of the two USB-CAN-A are connected correspondingly, and the 120-ohm resistance switch of the two is turned on (the switch is moved to the side near the green terminal), and the USB ends are respectively connected to the USB ports of the Raspberry Pi. The hardware connection is shown below:



(/wiki/File:2-USB-CAN-A.jpg)

- The following commands query and confirm that the Raspberry Pi recognizes the USB-CAN-A and loads the related driver:

```
lsusb
ls /dev
```



(/wiki/File:USB-CAN-A_Linux_Use.png)

- For some Linux systems that do not automatically load driver descriptors, you can use the following command to load the driver:

```
sudo su
sudo modprobe option
sudo sh -c 'echo "1a86 7523" > /sys/bus/usb-serial/drivers/option1/new_id'
```

Demo Test

- Copy the following commands to the terminal of the Raspberry Pi.

```
wget https://files.waveshare.com/upload/7/72/USB-CAN-A.zip
sudo apt-get install unzip
unzip USB-CAN-A.zip
cd USB-CAN-A
sudo make clean
sudo make
```

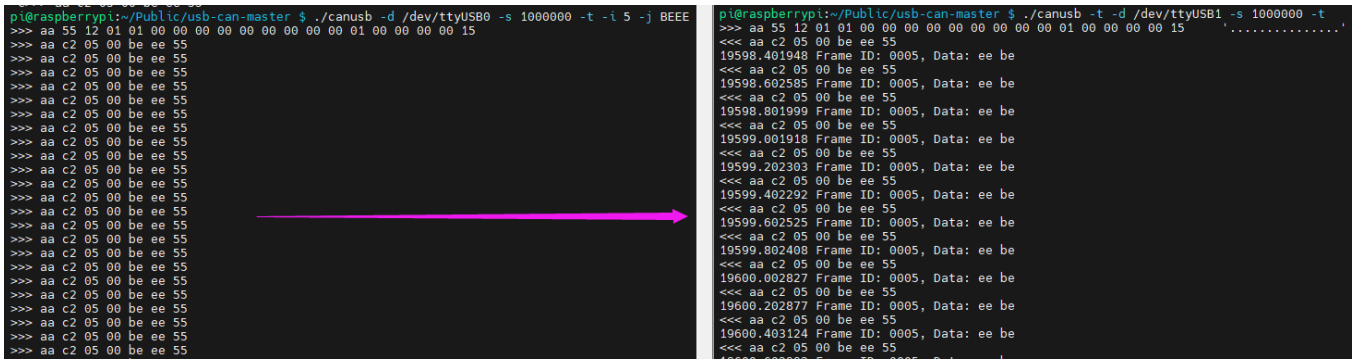
- Enter the following commands in the terminal interface.

```
./canusb -t -d /dev/ttyUSB0 -s 1000000 -t
```

- Open another terminal interface and run:

```
./canusb -d /dev/ttyUSB1 -s 1000000 -t -i 5 -j BEEE
```

- One USB-CAN-A (ttyUSB0) sends data, and the other USB-CAN-A (ttyUSB1) receives the corresponding data. The effect is as follows:



(/wiki/File:Usb-can-a_linux10.png)

Resource

Demo

- C Demo (/wiki/File:USB-CAN-A.zip)

Software

- CH341SER Driver (https://files.waveshare.com/upload/c/c4/CH341SER.zip)
- USB-CAN-A_TOOL_2.10 (the latest version, recommended) (https://files.waveshare.com/wiki/USB-CAN-A/Tool/USBCANV2.10.zip)
- USB-CAN-A_TOOL_1.2 (https://files.waveshare.com/upload/8/8a/USB-CAN-A_TOOL_EN.zip)

Documentation

- Serial port CAN protocol secondary development document (/wiki/File:USB_(Serial_port)_to_CAN_protocol_defines.pdf)
- CAN Configuration Command && Secondary Development Instruction Manual (/wiki/File:Can_config.pdf)

FAQ

Question:What do the parameters of the Linux commands represent?

```
./canusb -d /dev/ttyUSB1 -s 1000000 -t  -i  601   -j  00000601000000EA    BEEE1
```

Answer:

The following is a description of some of the parameters:

```
-t Print TTY/serial traffic debug information on stderr.
 -d device Use TTY device.
 -s SPEED Sets the CAN speed in bps.
 -b BAUDRATE Sets the TTY/serial baud rate (default: 2000000).
 -i ID Use ID (specified as a hexadecimal string) to inject.
 -j DATA CAN DATA To be injected (specified as a hexadecimal string).
 -n COUNT To terminate after a COUNT frame (default: infinite).
 -g MS To inject sleep gap in MS milliseconds (default: 200 milliseconds).
 -m MODE To inject load mode (0 = random, 1 = incremental, 2 = fixed).
```

Question:Why does the data lose packets?

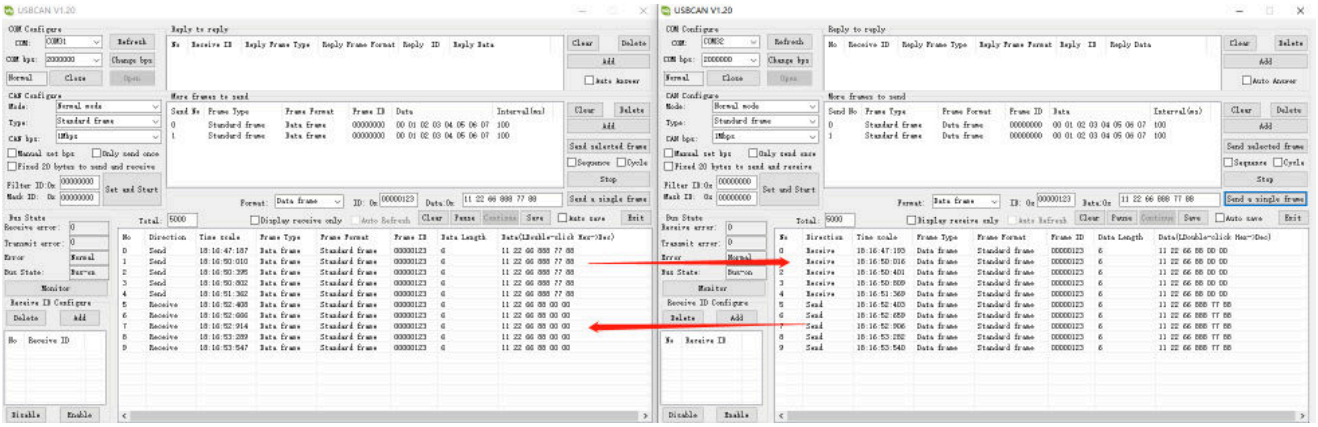
Answer:

Sending data too fast, and starting to send the next one before the first one is successfully sent, can also lead to packet loss; sending data just when there is data on the bus can lead to packet loss. It is recommended that packets be sent at certain intervals or with appropriate delays so that other devices have enough time to process the packets.

Question:How can I test two USB-CAN-A data transmission and reception?

Answer:

The sender's frame ID should match the receiver's filter ID; both parties must have matching CAN bus rates, and the default baud rate for the serial port is 2 Mbps.

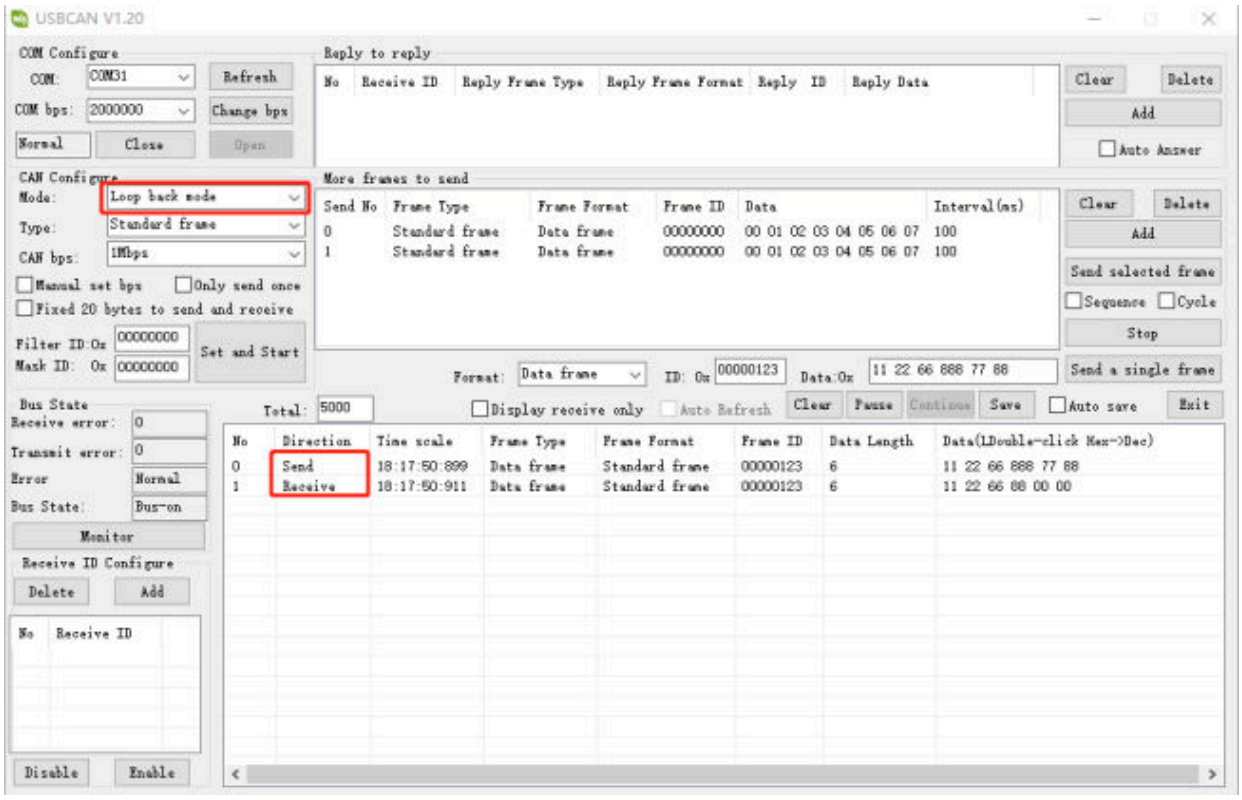


(/wiki/File:USB-CAN-A_FAQ01.png)

Question:I only have a USB-CAN-A on hand, how can I tell that its function is OK?

Answer:

- Please check the baud rate first. If the TX and RX indicators blink once when it is powered, the default setting is right (the serial port baudrate is 2M).
- Please set the module to loopback mode, set the 2M serial port baudrate and the CAN rate by yourself. Boots it, send the data and receive them, and you can see that the function is OK.



(/wiki/File:USB-CAN-A_FAQ02.png)

If the above method can not work, you can restore the module to the default factory

setting and test:

1	TX indicator:	Blinks when CAN is sending data
2	PWR indicator:	Power indicator, light up red when the USB port is connected
3	RX indicator:	Blinks when CAN is receiving data
4	Reset button:	Press before power on then release after power on to restore factory settings
5	120Ω resistor switch:	Switch to CAN terminal side to enable 120Ω resistor, switch to the other side to disable

(/wiki/File:USB-CAN-A-details-inter.jpg)

Question:Which protocols does USB-CAN-A support?

Answer:

Currently, it supports CAN 2.0A and CAN 2.0B, which are the most basic and common CAN standards. It supports a maximum data transfer rate of 1 Mbps.

Question:Why is the USB-CAN-A serial port baud rate set to 2M?

Answer:

The 2M baud rate can accommodate most CAN speeds. In general, the default 2M baud rate suffices, and adjusting the CAN rate to match the external CAN device (maximum support 1M) is usually all that's needed.

Question:Does the USB-CAN-A have message buffering? If so, how many messages can it buffer?

Answer:

Yes, it does. Data is placed in the buffer to prevent errors. If data were forwarded immediately upon reception, errors could occur when new data is received. Hence, buffering is required. Internally, it can buffer up to 20 messages. The converter operates on a mostly real-time conversion basis.

Question:Why wasn't the baud rate change successful?

Answer:

It's not recommended to alter the baud rate. In 99% of cases, 2M is universally applicable. If you belong to the 1% that needs alteration, opt for the default 2M setting. Since the default is 2M, selecting this should take effect. If it still doesn't work, reset to the default

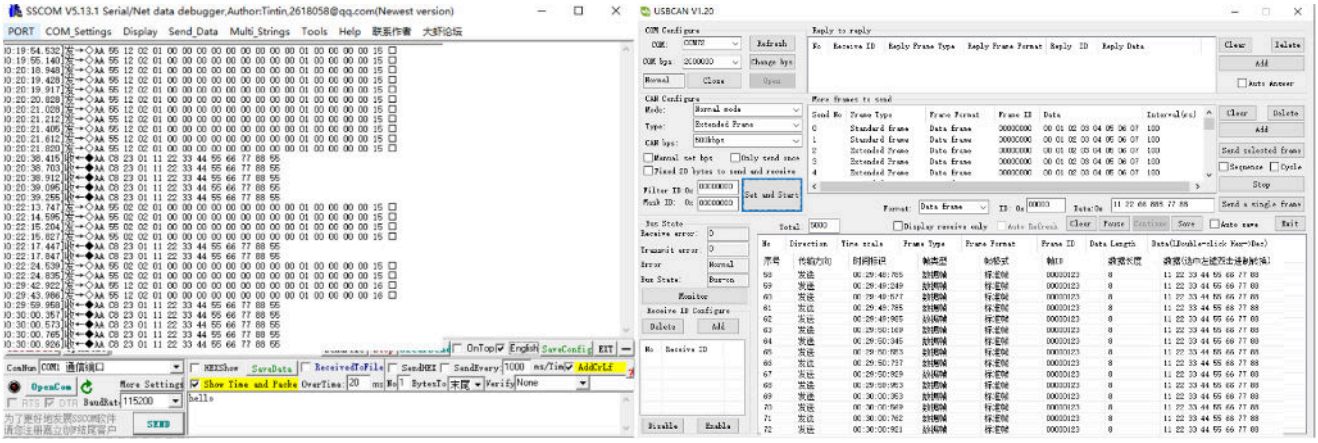
2M before making further changes.

(/wiki/File:USB-CAN-A-faq.png)

Question:Can you use a different CAN tool as the host computer?

Answer:

Different CAN configuration tools have different underlying commands. Apart from using our host computer tool, you can also utilize a serial port debugging assistant tool to send underlying commands, as shown in the image:



(/wiki/File:USB-CAN-A-faq2.png)

Question:How to confirm the baud rate?

Answer:

Blink once: Corresponding serial port baud rate is 2,000,000 bps.

Blink twice: Corresponding serial port baud rate is 1,228,800 bps.

Blink three times: Corresponding serial port baud rate is 115,200 bps.

Blink four times: Corresponding serial port baud rate is 38,400 bps.

Blink five times: Corresponding serial port baud rate is 19,200 bps.

Blink six times: Corresponding serial port baud rate is 9,600 bps.

Question:Can USB-CAN-A receive CAN messages from vehicles?

Answer:

Vehicles meeting the following conditions can be supported:

- Having a standard CAN interface, supporting CAN 2.0 A/B protocol.
- As vehicle interfaces may vary and the protocol can be complex, USB-CAN-A might not be optimally utilized. When communicating, it may be necessary to remove the 120-ohm jumper cap on the USB-CAN-A device. Apart from this consideration, communication is similar to normal devices. As long as the baud rate is consistent, data can be sent and received normally.
- If industrial-grade or vehicle-specific SAE J1939 protocol is required, please use USB-CAN-B (https://www.waveshare.com/usb-can-b.htm) instead.

Question:Can USB-CAN-A function as a node for data transmission and reception?

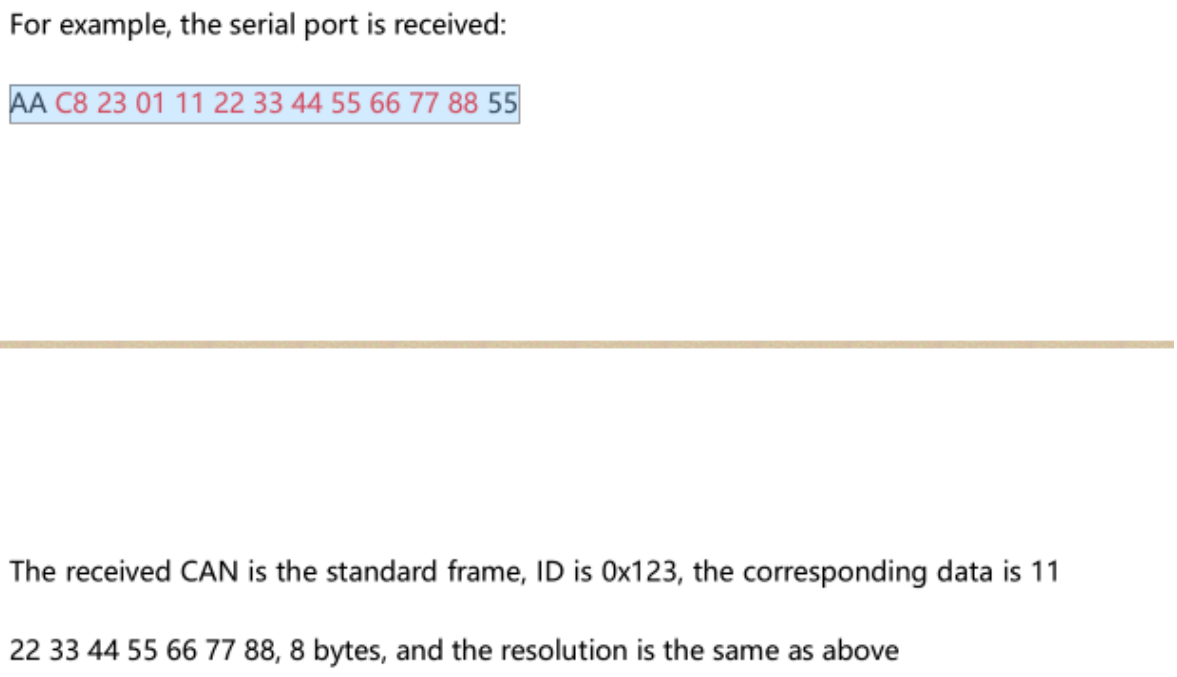
Answer:

Yes, it can function as a node on the CAN bus for data transmission and reception. By connecting to a PC or other host system through a USB interface, it uses its integrated CAN controller to communicate with the CAN bus network. This setup allows the host system to access the CAN network via the USB interface, enabling data transmission and reception.

Question:I want to know does USB to CAN Adapter Model A has a Hardware interface for python-can or with one is support for it?

Answer:

We haven't developed a demo for python yet. You can try to send it with tools such as our conversion definition and serial port debugging assistant. Basically, anyone who has done programming can quickly understand it;
You can directly send the message inside. Try AA C8 23 01 11 22 33 44 55 66 77 88 55



(/wiki/File:USB-can-a-faq02.png)
For more details, you can refer to this link (https://files.waveshare.com/wiki/USB-CAN-A/Demo/USB%20(Serial%20port)%20to%20CAN%20protocol%20defines.pdf).

Question:Does USB-CAN-A support opencan?

Answer:

Not fully support. You can check it from CAN OPEN data, and it does not analyze the CAN OPEN protocol. It is recommended to use USB-CAN-B (https://www.waveshare.com/usb-can-b.htm).

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 PM GMT+8 (Monday to Friday)

Submit Now (https://service.waveshare.com/)