

USB2CAN Series Python On Windows

Menu

USB2CAN Series Python On Windows	1
1. General Description:	3
2. Driver Install	4
3. Python Demo	6
3.1 install gs-usb	6
3.3 Modify the gs-usb code	6
3.2 Running Demo	8
3.3 Can Not Find gs_usb Device	9
4. Python Programme	9
5. User Manual Version Descriptions	10

1. General Description:

This document is for people want to use below product with Python programing on Windows platform.



USB2CAN



USB2CAN-C



USB2CAN-DEV



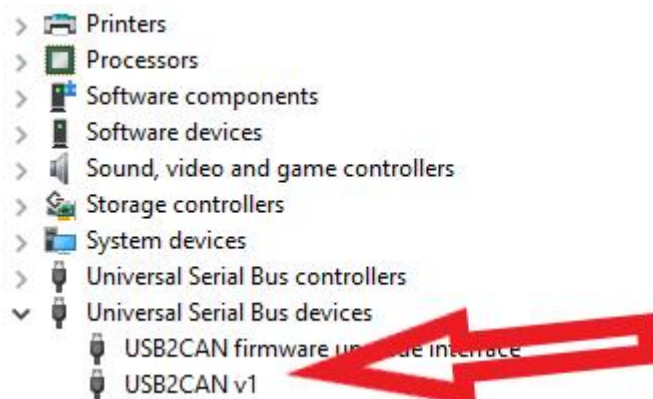
USB2CAN-X2

2. Driver Install

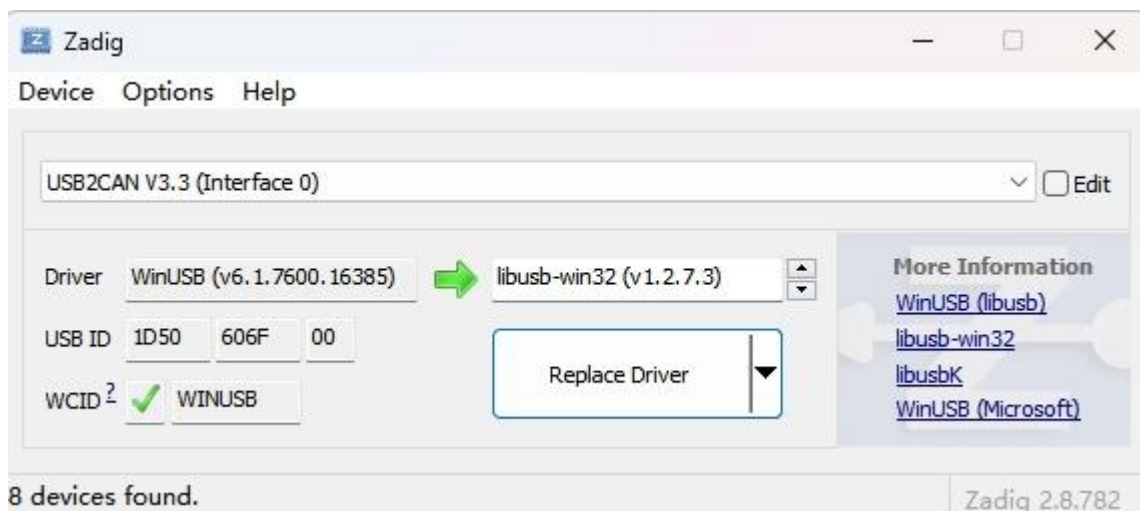
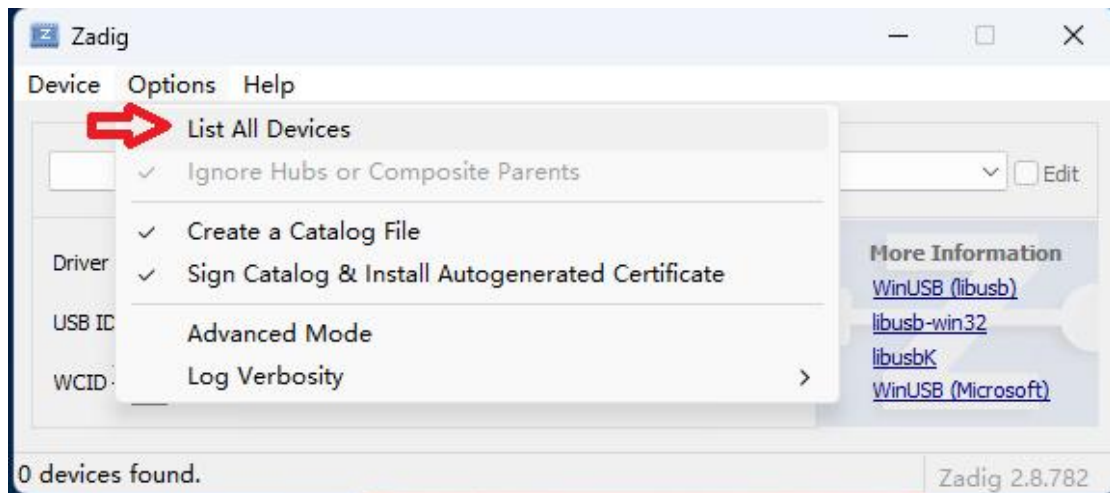
USB2CAN module is a WinUsb device, On Windows 10/8 have already auto-install the driver default. If your computer do not have the driver, please install the driver by Zadig tools.

<https://github.com/INNO-MAKER/usb2can/tree/master/For%20Windows/Windows%20Driver%20Tool>

USB2CAN can be find in the device manager on Windows after Install completes.



If you want to use the python demo, the driver need to set as
libusb-win32



3. Python Demo

I use the python 3.11.7-64bit and Windows 10/11 PowerShell(terminal)
to show you how to run the demo.

3.1 install gs-usb

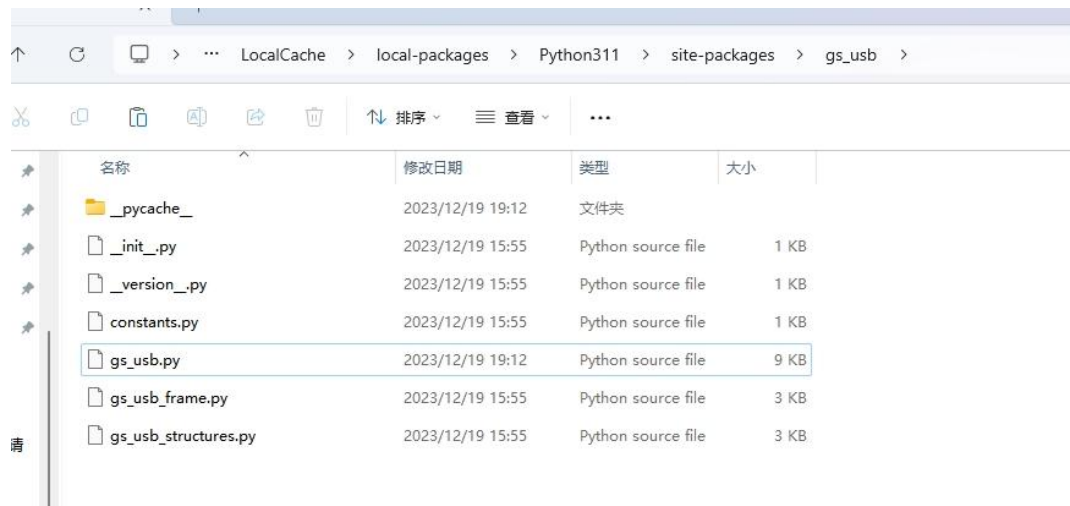
Use below command install gs-usb 0.3.0. Do not use other version.

```
pip install gs-usb==0.3.0
```

3.3 Modify the gs-usb code

On some windows system version, the usb restart function will bring some error. So we'd better modify gs-usb codes firstly.

(1)Open the file named **gs_usb.py** in the local-packages\Python311\site-packages\gs_usb which you have downloaded.



(2) Comment out the self.gs_usb.reset() And save the files.

```

test.py | gs_usb.py
22 GS_USB_ABE_CANDEBUGGER_FD_VENDOR_ID = 0x16D0
23 GS_USB_ABE_CANDEBUGGER_FD_PRODUCT_ID = 0x10B8
24
25 #gs_usb mode
26 GS_CAN_MODE_RESET = 0
27 GS_CAN_MODE_START = 1
28
29 # gs_usb control request
30 _GS_USB_BREQ_HOST_FORMAT = 0
31 _GS_USB_BREQ_BITTIMING = 1
32 _GS_USB_BREQ_MODE = 2
33 _GS_USB_BREQ_BERR = 3
34 _GS_USB_BREQ_BT_CONST = 4
35 _GS_USB_BREQ_DEVICE_CONFIG = 5
36
37
38 class GsUsb:
39     def __init__(self, gs_usb):
40         self.gs_usb = gs_usb
41         self.capability = None
42         self.device_flags = None
43
44     def start(self, flags=(GS_CAN_MODE_NORMAL | GS_CAN_MODE_HW_TIMESTAMP)):
45         """
46         Start gs_usb device
47         :param flags: GS_CAN_MODE_LISTEN_ONLY, GS_CAN_MODE_HW_TIMESTAMP, etc.
48         """
49         # Reset to support restart multiple times
50         self.gs_usb.reset()
51
52         # Detach usb from kernel driver in Linux/Unix system to perform IO
53         if "windows" not in platform.system().lower() and self.gs_usb.is_kernel_drive
54             0
55         \:

```

3.2 Running Demo

Download the demo usb2can.py from below link. And use PowerShell running it. You could the USB2CAN device send and receive frame.

<https://github.com/INNO-MAKER/usb2can/tree/master/For%20Windows/Python>

python usb2can.py

PS C:\test> python .\usb2can.py

```
PS C:\test> python usb2can.py
[<gs_usb.gs_usb.GsUsb object at 0x0000023A4C717F08>]
TX      7FF [8] 12 34 56 78 9A BC DE F0
RX      7FF [8] 12 34 56 78 9A BC DE F0
RX      7FF [8] 12 34 56 78 9A BC DE F0
TX      7FF [0]
RX      7FF [0]
RX      7FF [0]
TX      7FF [8] 12 34 56 78 9A BC DE F0
RX      7FF [8] 12 34 56 78 9A BC DE F0
TX 12345678 [8] 12 34 56 78 9A BC DE F0
RX 12345678 [8] 12 34 56 78 9A BC DE F0
RX 12345678 [8] 12 34 56 78 9A BC DE F0
TX 12345678 [0]
RX 12345678 [0]
RX 12345678 [0]
TX      7FF [0] remote request
RX      7FF [0] remote request
RX      7FF [0] remote request
TX 12345678 [0] remote request
RX 12345678 [0] remote request
RX 12345678 [0] remote request
TX      7FF [8] remote request
RX      7FF [8] remote request
RX      7FF [8] remote request
TX      7FF [8] 12 34 56 78 9A BC DE F0
RX      7FF [8] 12 34 56 78 9A BC DE F0
RX      7FF [8] 12 34 56 78 9A BC DE F0
TX      7FF [0]
RX      7FF [0]
RX      7FF [0]
TX      7FF [8] 12 34 56 78 9A BC DE F0
RX      7FF [8] 12 34 56 78 9A BC DE F0
TX 12345678 [8] 12 34 56 78 9A BC DE F0
RX 12345678 [8] 12 34 56 78 9A BC DE F0
RX 12345678 [8] 12 34 56 78 9A BC DE F0
TX 12345678 [0]
RX 12345678 [0]
RX 12345678 [0]
TX      7FF [0] remote request
RX      7FF [0] remote request
RX      7FF [0] remote request
TX 12345678 [0] remote request
RX 12345678 [0] remote request
RX 12345678 [0] remote request
```


The code default is loop-back mode with 1M baud rate for test. If you want communicate with othe CAN device. Change the work mode to GS_USB_MODE_NORMAL and reset baud rate.

```
# Configuration
if not dev.set_bitrate(1000000):
    print("Can not set bitrate for gs_usb")
    return

# Start device, If you have only one device for test, pls use the loop-back mode,
dev.start(GS_USB_MODE_LOOP_BACK)
#dev.start(GS_USB_MODE_NORMAL)
```

3.3 Can Not Find gs_usb Device

If you could see the usb2can device in the device manager, But PowerShell print 'Can not find gs_usb device' when you running the demo. The libusb driver may install failed.

Download the libusb packet from below link:

<https://sourceforge.net/projects/libusb/>

Copy the MS64\dll\libusb-1.0.dll to your computer C:\Windows\System32. Try run the demo again.

4. Python Programme

For more Programming detail , please refer to Comments in demo code and the source code for python gs_usb library:

https://github.com/jxltom/gs_usb

<https://pypi.org/project/gs-usb/0.3.0/>

5. User Manual Version Descriptions

Version	Description	Date	E-mail
V1.0		2022.07.02	support@inno-maker.com calvin@inno-maker.com
V1.1		2023.12.10	support@inno-maker.com calvin@inno-maker.com