

Высокопроизводительный адаптер USB-CAN-FD/V



Руководство пользователя

Содержание

1 Введение.....	3
2 Технические характеристики	3
3 Работа в Windows.....	4
3.1 Установка драйвера.....	4
4 Программное обеспечение USB-CAN-FD Tool	7
4.1 Загрузка и установка.....	7
4.2 Порядок использования.....	10
4.3 Тест обратной связи (Loopback Test)	15
5 Руководство для Linux	17
5.1 Raspberry Pi OS	17
5.2 Ubuntu (Jetson Nano).....	19
5.3 Linux для виртуальных машин VMware	19
Ресурсы	20
Часто задаваемые вопросы (FAQ)	21
.....	22

1 Введение

USB-CAN-FD — промышленный высокопроизводительный адаптер USB–CAN FD, интерфейсная плата для связи по шинам CAN/CAN-FD и анализатор данных протокола CAN/CAN-FD. Устройство оснащено двумя независимыми интерфейсами CAN FD с электрической изоляцией и несколькими схемами защиты. Поддерживаются Windows XP/7/8/10/11; в комплект входят драйверы, программное обеспечение CAN FD Tools, примеры для вторичной разработки и учебные материалы.

Устройство подключается к ПК или промышленному управляющему компьютеру через USB-порт и позволяет выполнять управление передачей и приемом, анализ и сбор данных, а также мониторинг сети CAN/CAN FD. Благодаря компактным размерам и простоте использования оно подходит для изучения и отладки шины CAN/CAN FD, а также для вторичной разработки и интеграции в различные промышленные, энергетические, коммуникационные и интеллектуальные системы управления, которым требуется обмен по CAN/CAN FD.

2 Технические характеристики

Модель		USB-CAN-FD-B	USB-CAN-FD
Тип продукта	Промышленный класс	Конвертер интерфейса USB в CAN FD, карта связи по шине CAN/CAN FD, анализатор данных протокола CAN/CAN FD	
Порт USB	Рабочее напряжение	5 В (питание напрямую через порт USB, дополнительный источник питания не требуется)	
	Форма интерфейса	Порт USB-B	
Интерфейс CAN/CAN FD	Каналы CAN/CAN FD	Два канала: CAN1 и CAN2 (независимы друг от друга, не влияют друг на друга, полная гальваническая развязка, напряжение изоляции 3000 В постоянного тока)	
	Форма интерфейса	Клеммный интерфейс (OPEN6/KF2EDG-6P, шаг 5,08 мм)	
	Согласующий резистор	Каждый канал CAN/CAN FD имеет два встроенных согласующих резистора 120 Ом, которые можно включать переключателем	
	Скорость передачи данных	100 Кбит/с ~ 5 Мбит/с	
	Поддержка протоколов	CAN2.0A и 2.0B, протокол ISO 11898-1 CAN FD версии 1.0	
	Скорость передачи	Скорость приёма и отправки каждого канала CAN/CAN FD может достигать 20000 кадров/с и 5000 кадров/с	
	Буфер передачи	2000 кадров буфера приёма и 1000 кадров буфера отправки на канал (автоматическая повторная передача при сбое передачи)	1500 кадров буфера приёма и 64 кадра буфера отправки на канал (автоматическая повторная передача при сбое передачи)

Модель		USB-CAN-FD-B	USB-CAN-FD
Индикаторы	PWR	Индикатор питания	
	SYS	Индикатор состояния системы, в норме выключен; горит постоянно при ошибке	
	CAN1	Индикатор канала CAN1 (мигает при отправке и приёме данных)	
	CAN2	Индикатор канала CAN2 (мигает при отправке и приёме данных)	
Поддержка систем	Windows	Windows XP/7/8/10/11 (32-разрядная и 64-разрядная)	
	Linux	Raspberry Pi OS, Ubuntu (Jetson Nano), виртуальные машины VMware и др.	Не поддерживается
Рабочая температура			-40~85°C
Материал корпуса	Корпус из алюминиевого сплава + 3D огнестойкие изоляционные листы с обеих сторон обеспечивают лучшую защиту от разряда с металлического острия, повышают безопасность изделия и продлевают срок службы		Корпус из алюминиевого сплава + 3D огнестойкие изоляционные листы с обеих сторон обеспечивают лучшую защиту от разряда с металлического острия, повышают безопасность изделия и продлевают срок службы
Габариты	104 × 70 × 25 (ед. изм.: мм)		104 × 70 × 25 (ед. изм.: мм)

3 Работа в Windows

Модели USB-CAN-FD и USB-CAN-FD-B поддерживают работу в Windows. Порядок работы в Windows для обеих моделей одинаков.

3.1 Установка драйвера

Подключите модуль USB-CAN-FD к ПК с помощью USB-кабеля. После обнаружения оборудования Windows автоматически запустит мастер установки нового оборудования. Загрузите драйвер Windows.

- Нажмите «Next» (Далее) для продолжения

Select Components

Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

典型安装

☒ MS VC++ 2008 Redistributable	4.3 MB
☒ driver	18.4 MB
☒ 驱动安装程序 (首次安装必选)	6.6 MB

Current selection requires at least 30.3 MB of disk space.

USB驱动安装工具1.40版

关于

< Back

Next >

Cancel

- Нажмите «Install» (Установить)

Ready to Install

Setup is now ready to begin installing USB驱动安装工具 on your computer.

Click Install to continue with the installation, or click Back if you want to review or change any settings.

Destination location:
C:\Program Files (x86)\USB_CAN TOOL\driver

Setup type:
典型安装

Selected components:
MS VC++ 2008 Redistributable
driver
驱动安装程序 (首次安装必选)

Additional tasks:
Additional icons:

USB驱动安装工具1.40版

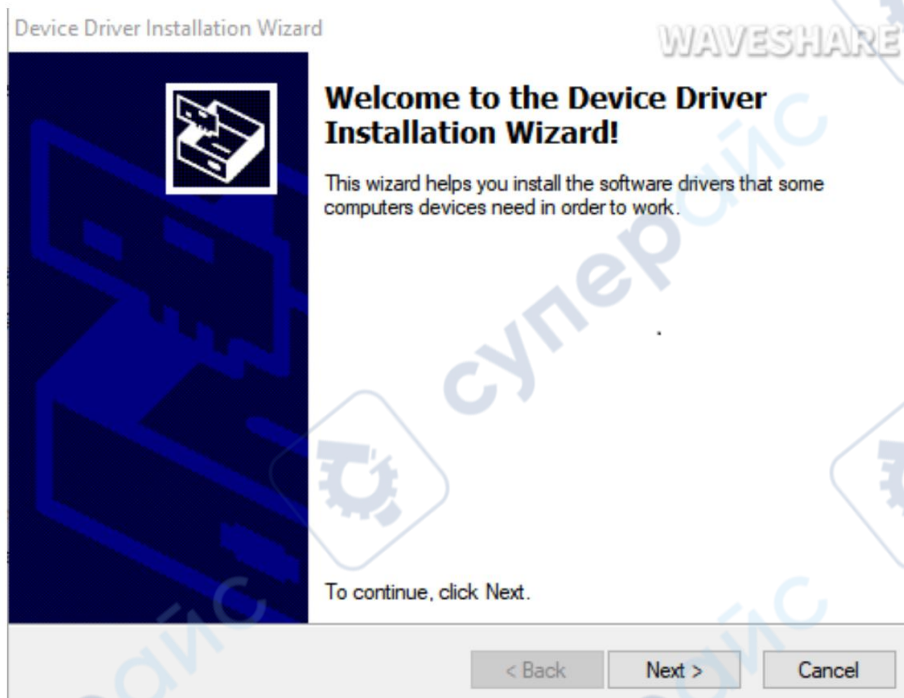
关于

< Back

Install

Cancel

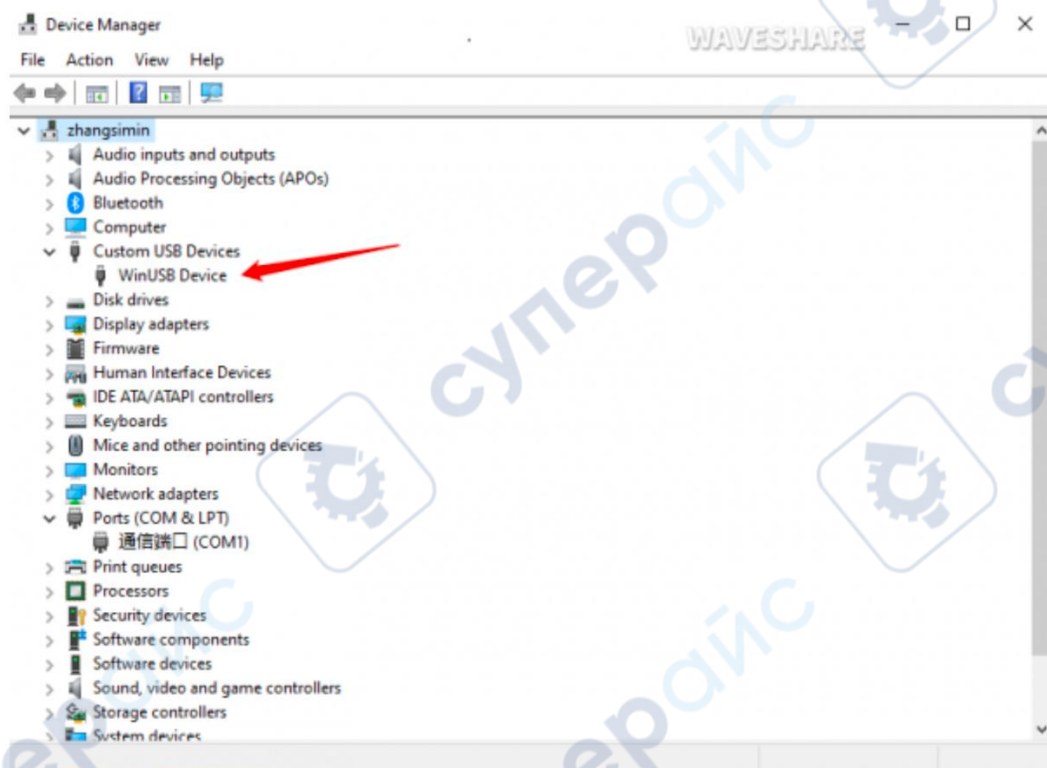
- Нажмите «Next» (Далее)



- Дождитесь завершения установки.



- После успешной установки вы увидите устройство WinUSB Device в Диспетчере устройств

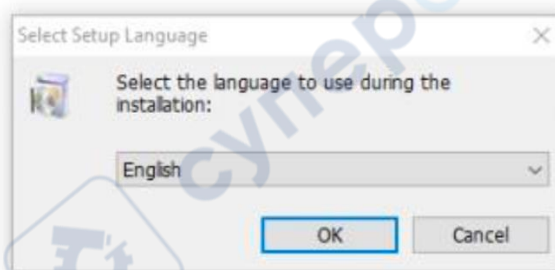


4 Программное обеспечение USB-CAN-FD Tool

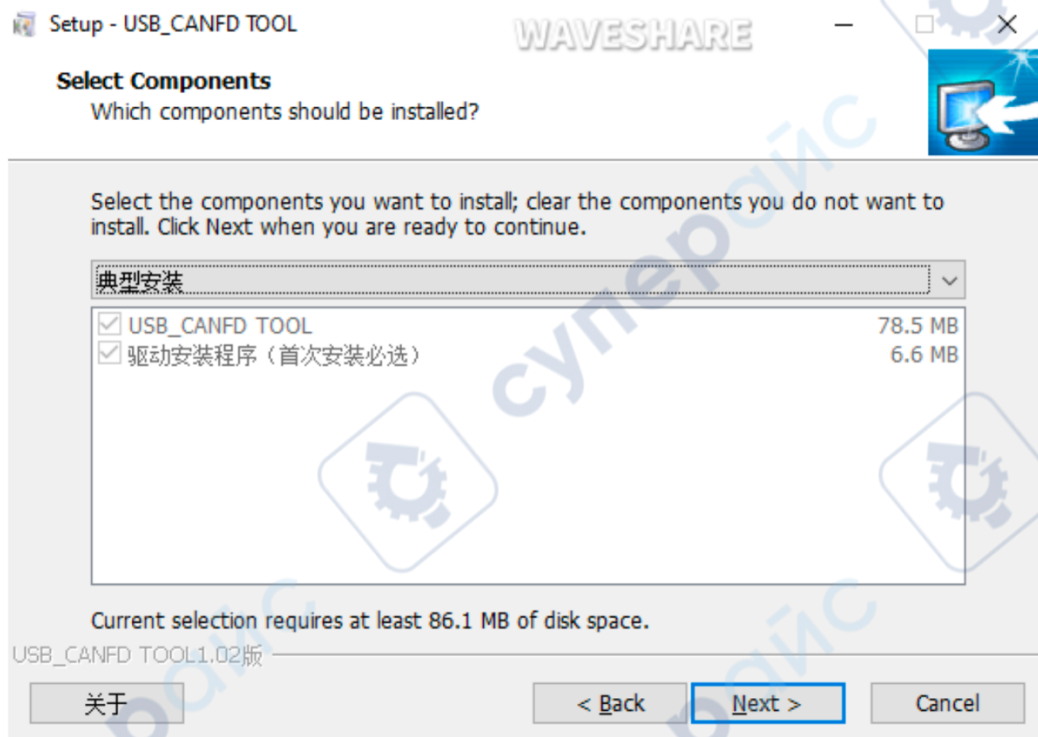
4.1 Загрузка и установка

- Выберите язык по своему усмотрению, здесь приведён пример с английским языком

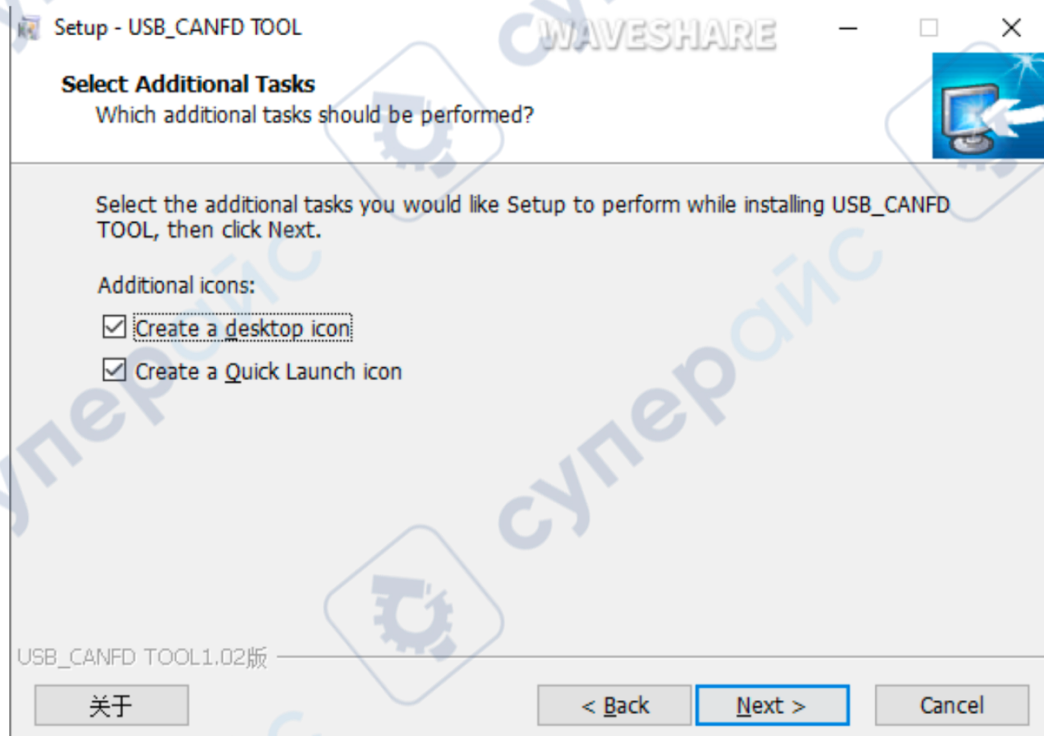
Name	Date modified	Type	Size
USB_CANFD TOOLSetup(V1.02).exe	2023/6/21 12:19	Application	29,690 KB



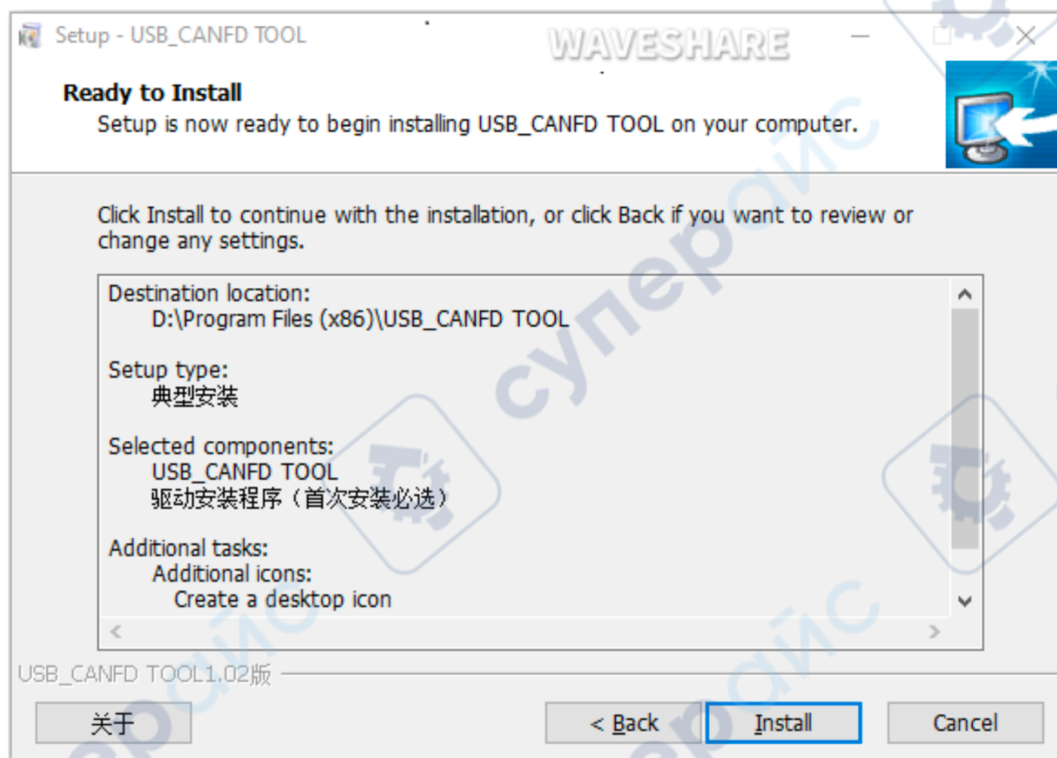
- Выберите вариант по умолчанию и нажмите «Next» (Далее)



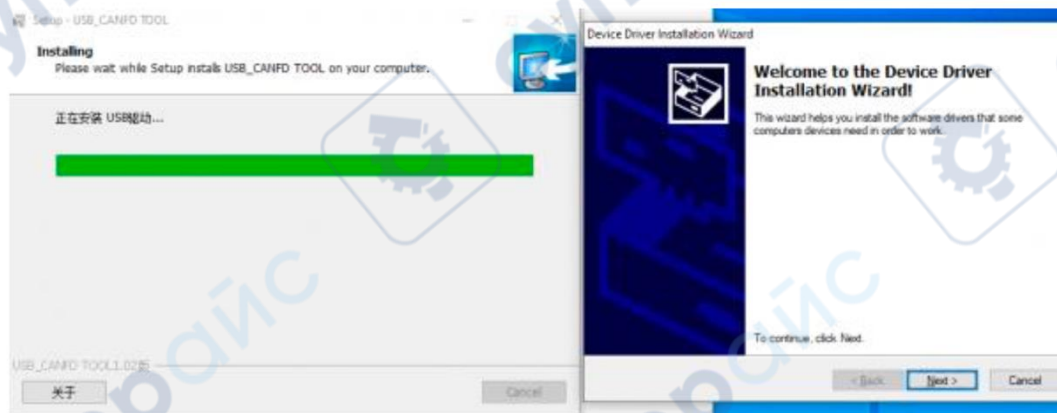
- Рекомендуется установить флажок для создания значка на рабочем столе и продолжить, нажав «Next» (Далее)



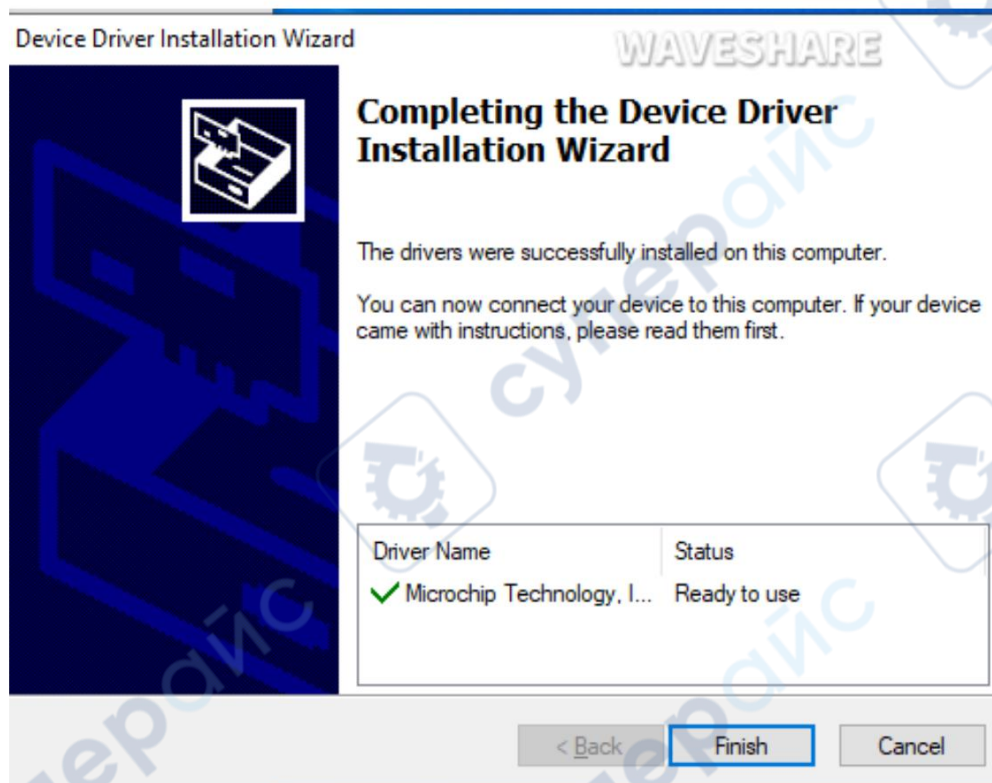
- Нажмите «Install» (Установить)



- После завершения продолжите, нажав «Next» (Далее)



- После завершения установки на рабочем столе и в списке приложений появится значок CANFDToolPro



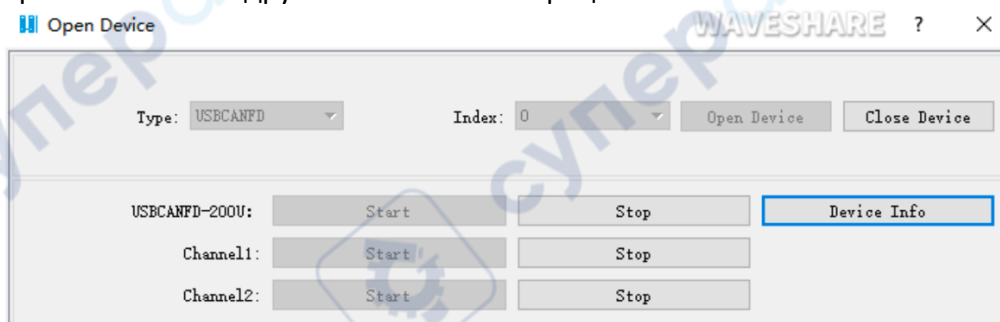
4.2 Порядок использования

Если связь работает нестабильно, установите оба терминирующих резистора 120 Ом каналов CAN1/CAN2 в положение ON.

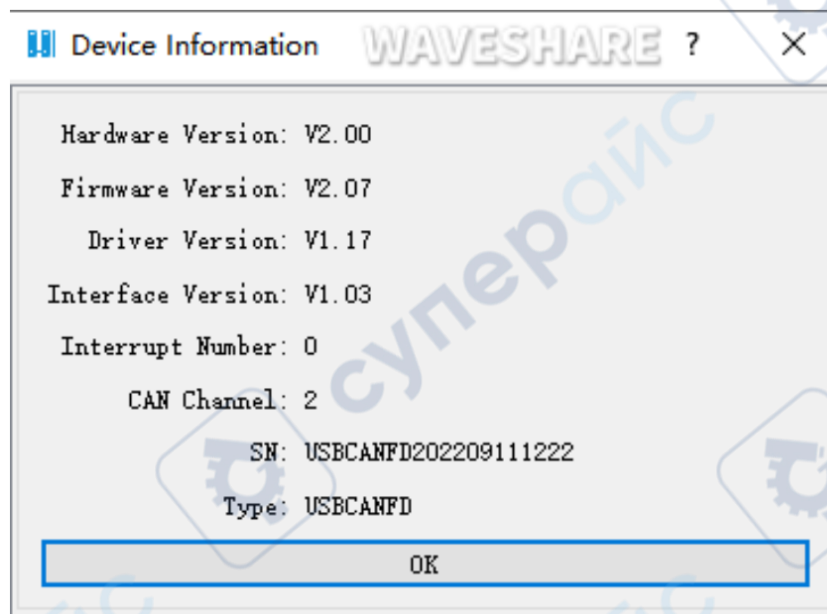
Подключите USB-порт устройства к USB-порту ПК с помощью USB-кабеля. Запустите USB-CAN-FD_TOOL.exe. Ниже приведено описание основных функций.

1. Управление устройством

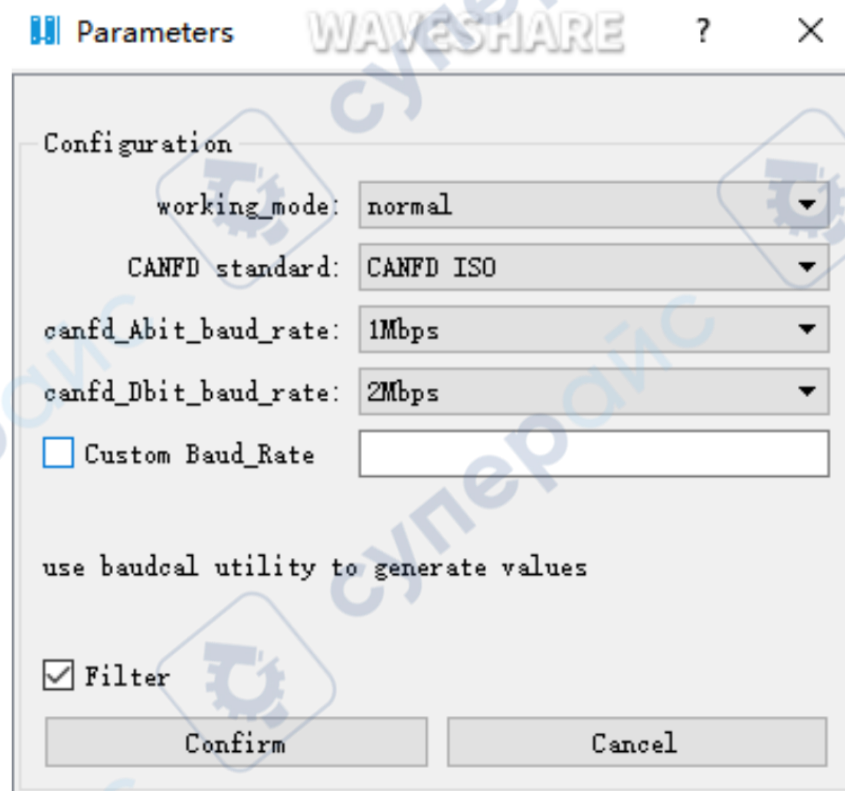
Выберите «Device Manage(O)» -> «Open Device», чтобы открыть интерфейс управления устройством. Здесь можно открыть устройство, запустить и остановить канал, настроить параметры и выполнить другие связанные операции.



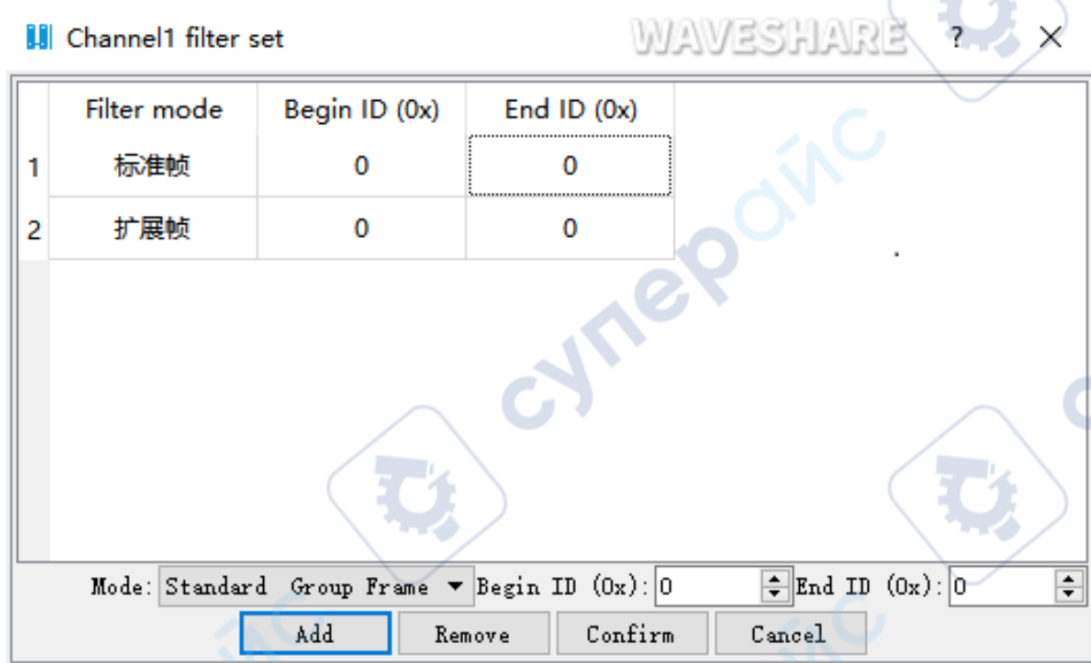
- Нажмите «Device Information», чтобы просмотреть информацию о версии программного и аппаратного обеспечения текущего устройства.



- При запуске канала открывается окно настройки параметров. В нем можно выбрать режим работы, стандарт CANFD, скорость арбитражной фазы, скорость фазы данных и другие параметры.



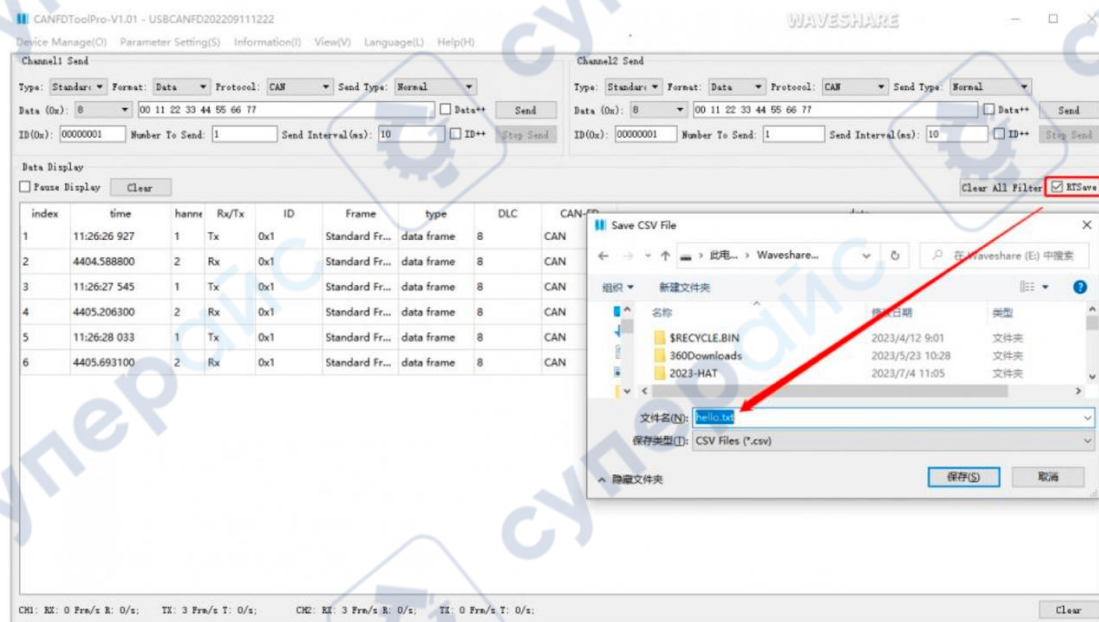
- Параметры фильтров задаются в соответствующем окне. Для стандартных кадров можно настроить до 64 групп фильтров, для расширенных кадров — до 32 групп.



2. Функция отображения

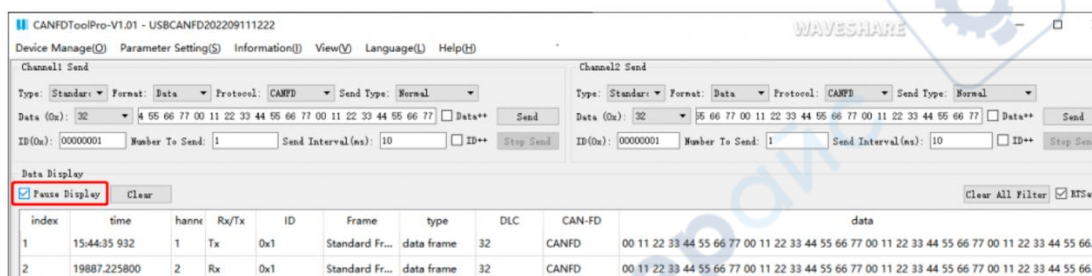
2.1 Сохранение в реальном времени

- Формат сохранения по умолчанию — csv; если требуется сохранить в формате txt, просто введите расширение .txt непосредственно в имени файла.



2.2 Пауза отображения

- Во время приема данных можно установить флажок Pause display, чтобы временно приостановить вывод данных на экран. Прием данных при этом продолжается; данные лишь не отображаются в текущем окне. После снятия флажка отображение продолжится.



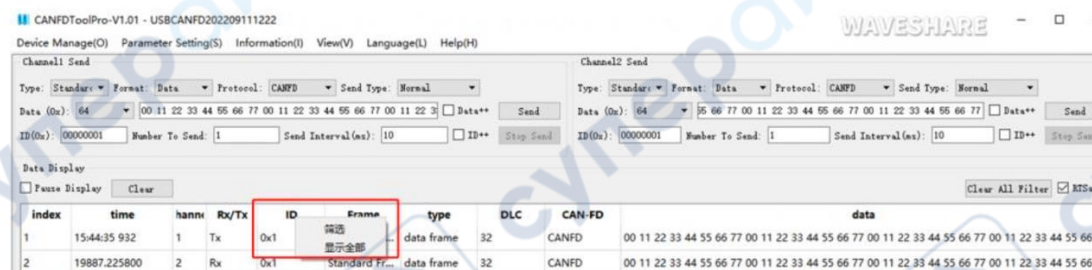
2.3 Счётчики передачи

В нижней части программы подсчитывается общее количество приёмов и передач, а также частота кадров приёма и передачи в реальном времени для обоих каналов. Нажмите кнопку «Clear», чтобы сбросить счётчик.



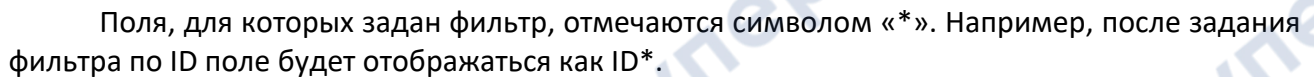
2.4 Буфер отображения

Буфер отображения указывает максимальное количество кадров, которое может быть отображено в текущем интерфейсе; значение по умолчанию — 1000.

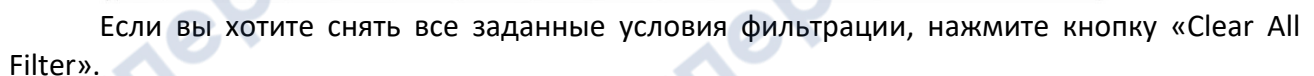


2.5 Просмотр данных

Если кадр слишком длинный (например, 64 байта) и не помещается в окне полностью, наведите указатель мыши на границу столбца. Когда указатель примет вид двунаправленной стрелки, дважды щелкните мышью, чтобы развернуть содержимое. Для просмотра данных используйте горизонтальную полосу прокрутки.



Все условия фильтрации связаны отношением «И». Например, чтобы просмотреть кадры данных с ID 0x55 для канала 1, необходимо задать следующие условия фильтрации: Channel -> выбрать 1, Rx/Tx -> выбрать Rx, ID -> выбрать 0x55, type -> выбрать data frame. Как показано на рисунке ниже:

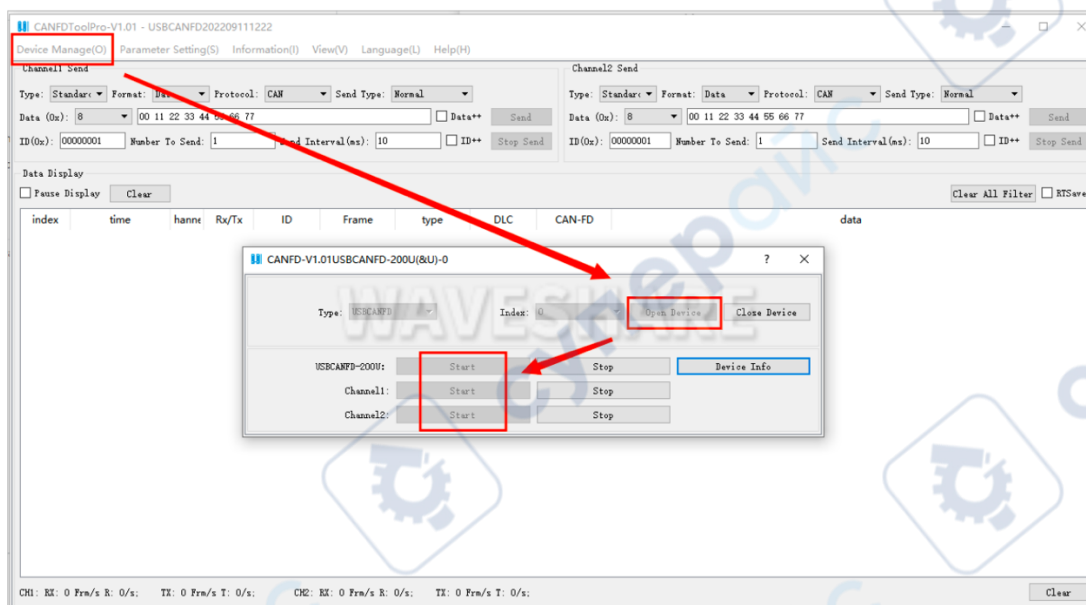


Тест обратной связи — надёжный способ проверки работы оборудования, и он является предпочтительным методом для первоначального тестирования. Основные шаги следующие:

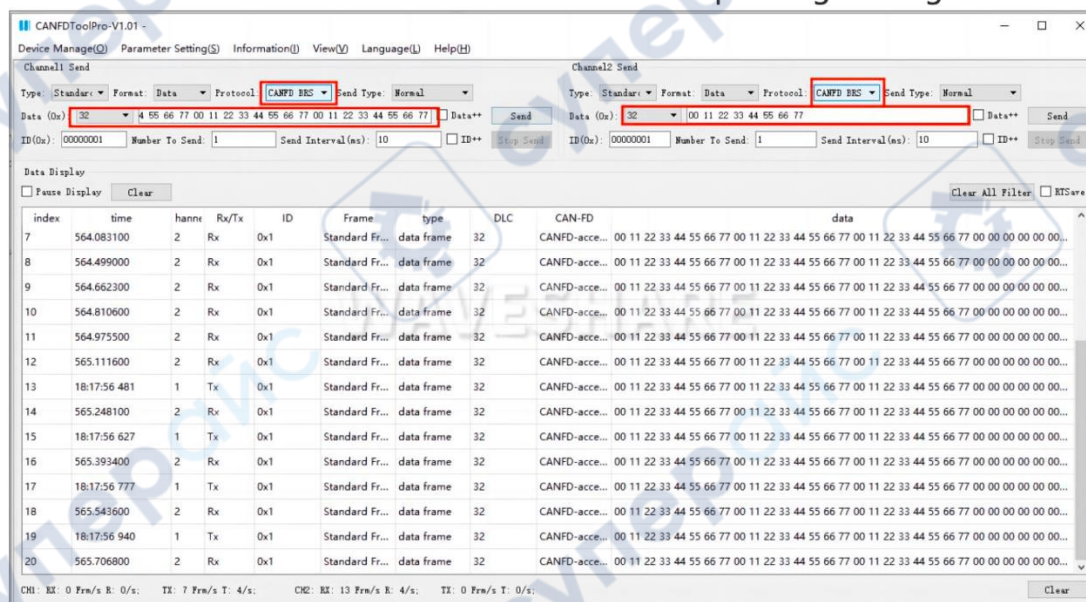
- Руководство пользователя



2. Откройте CANFDTool, нажмите «Device Manage(O)» -> «Open Device» -> «Start».



3. Установите Data (0x) в значение 32, Protocol — в CANFD BRS, отправьте сообщение 00 11 22 33 44 55 66 77 00 11 22 33 44 55 66 77 -> примите соответствующее сообщение.



5 Руководство для Linux

Раздел Linux относится только к USB-CAN-FD-B. Модель USB-CAN-FD Linux не поддерживает; для нее этот раздел можно пропустить.

5.1 Raspberry Pi OS

USB-CAN-FD-B можно использовать на распространённых 32-разрядных материнских платах Linux; в качестве примера взята Raspberry Pi 4B.

- Подключите USB-CAN-FD-B и Raspberry Pi через USB и соедините интерфейсы H и L канала 1 USB-CAN-FD-B с интерфейсами H и L канала 2 соответственно, как показано на рисунке ниже:



Скопируйте в терминал Raspberry Pi строки команд следующего демо-примера на c или python и выполните их:

c

```
cd ~
wget https://files.waveshare.com/wiki/USB-CAN-FD/USB-CAN-FD-B-Linux.zip
unzip USB-CAN-FD-B-Linux.zip
sudo chmod 777 -R USB-CAN-FD-B-Linux
cd /USB-CAN-FD-B-Linux/Raspberry/c
sudo make clean
sudo make
sudo ./hello_cpp
```

После запуска демо-примера на C результаты будут следующими:

```
Index:0000 CANFD1 TX ID:0x00000000 Standard Data DLC:0x40 data:(0x) 00 01 02 03 04 05 06 07
08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F
20 21 22 23 24 25 26 27
28 29 2A 2B 2C 2D 2E 2FIndex:0001
CANFD2 RX ID:0x00000000 Standard Data DLC:0x40 data:(0x) 00 01 02 03 04 05 06 07
08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F
20 21 22 23 24 25 26 27
28 29 2A 2B 2C 2D 2E 2F
30 31 32 33 34 35 36 37
38 39 3A 3B 3C 3D 3E 3F
TimeStamp:0x002D4128
30 31 32 33 34 35 36 37
38 39 3A 3B 3C 3D 3E 3F
Index:0002 CANFD2 TX ID:0x00000001 Standard Data DLC:0x40 data:(0x) 00 01 02 03 04 05 06 07
08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F
20 21 22 23 24 25 26 27
28 29 2A 2B 2C 2D 2E 2F
30 31 32 33 34 35 36 37
38 39 3A 3B 3C 3D 3E 3F
Index:0003 CANFD1 RX ID:0x00000001 Standard Data DLC:0x40 data:(0x) 00 01 02 03 04 05 06 07
```

python

```
cd ~
cd /USB-CAN-FD-B-Linux/Raspberry/python3/
sudo python3 python3.8.0.py
```

Результаты запуска демо-примера на python следующие:

```

jetson@ubuntu:~/Music/python3.8.0$ sudo python3 python3.8.0.py
#####
## Chuang Xin USBCANFD python(x64) test program V2.0 ##
#####
./libcontrolcanfd.so
ZCAN_OpenDevice Enter
open_usb_device ret:0
Open Device OK, device handle:0xffffabdf6d18.
Set CAN0 abt:1M OK!
Set CAN0 dbt:5M OK!
Set CAN1 abt:1M OK!
Set CAN1 dbt:5M OK!
Set CAN0 ISO mode OK!
Set CAN1 ISO mode OK!
Init CAN0 OK!
ZCAN_StartCAN can_ind:0
begin pthread_create pthread_0
Start CAN0 OK!
my_rcv_thread dev_idx:0
my_rcv_thread can_idx:0
my_rcv_thread begin while!
Init CAN1 OK!
ZCAN_StartCAN can_ind:1
begin pthread_create pthread_1
Start CAN1 OK!
my_rcv_thread dev_idx:0
my_rcv_thread can_idx:1
my_rcv_thread begin while!

CAN0 Transmit CANFD Num: 10.
CAN1 Received CANFD NUM: 1.
[0]:ts:157016400, id:5, len:16, eff:1, rtr:0, esi:0, brs: 1, data:0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

CAN0 Transmit CAN Num: 10.
CAN1 Received CAN NUM: 2.
[0]:ts:157018700, id:5, len:8, eff:1, rtr:0, data:0 1 2 3 4 5 6 7
[1]:ts:157018800, id:6, len:8, eff:1, rtr:0, data:0 1 2 3 4 5 6 7

```

5.2 Ubuntu (Jetson Nano)

c

```

wget https://files.waveshare.com/wiki/USB-CAN-FD/USB-CAN-FD-B-Linux.zip
unzip USB-CAN-FD-B-Linux.zip
sudo chmod 777 -R USB-CAN-FD-B-Linux
cd /VMware/x86-c
sudo make clean
sudo make
sudo ./hello_cpp

```

python

```

cd ~
cd /USB-CAN-FD-B-Linux/VMware/x86-python3/
sudo python3 python3.8.0.py

```

5.3 Linux для виртуальных машин VMware

c

```

wget https://files.waveshare.com/wiki/USB-CAN-FD/USB-CAN-FD-B-Linux.zip
unzip USB-CAN-FD-B-Linux.zip
sudo chmod 777 -R USB-CAN-FD-B-Linux
cd USB-CAN-FD-B-Linux/VMware/x86-c/
sudo make clean
sudo make
sudo ./hello_cpp

```

```

Index:0000  CANFD1 TX ID:0x00000000 Standard  Data  DLC:0x40 data:(0x) 00 01 02 03 04 05 06 07
08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F
20 21 22 23 24 25 26 27
28 29 2A 2B 2C 2D 2E 2F
30 31 32 33 34 35 36 37
38 39 3A 3B 3C 3D 3E 3F

Index:0001  CANFD2 TX ID:0x00000001 Standard  Data  DLC:0x40 data:(0x) 00 01 02 03 04 05 06 07
08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F

```

python

```

cd ~
cd USB-CAN-FD-B-Linux/VMware/x86-python3/
sudo python3 python3.8.0.py

```

```

h@ubuntu:~/USB-CAN-FD-B-Linux/VMware/x86-python3$ sudo python3 python3.8.0.py
#####
## Chuang Xin USBCANFD python(x64) test program V2.0 ##
#####
./libcontrolcanfd.so
ZCAN_OpenDevice Enter
open_usb_device ret:0
Open Device OK, device handle:0x7fc99a423d18.
Set CAN0 abt:1M OK!
Set CAN0 dbit:5M OK!
Set CAN1 abt:1M OK!
Set CAN1 dbit:5M OK!
Set CAN0 ISO mode OK!
Set CAN1 ISO mode OK!
Init CAN0 OK!
ZCAN_StartCAN can_ind:0
begin pthread_create pthread_0
Start CAN channel OK!
my_recv_thread dev_idx:0
my_recv_thread can_idx:0
my_recv_thread begin while!
Init CAN1 OK!
ZCAN_StartCAN can_ind:1
begin pthread_create pthread_1
my_recv_thread dev_idx:0
my_recv_thread can_idx:1
Start CAN channel OK!
my_recv_thread begin while!

CAN0 Transmit CANFD Num: 10.
CAN1 Received CANFD NUM: 2.
[0]:ts:22328719000, id:5, len:16, eff:1, rtr:0, esi:0, brs:1, data:0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
[1]:ts:22328722200, id:6, len:16, eff:1, rtr:0, esi:0, brs:1, data:0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

CAN0 Transmit CAN Num: 10.
CAN1 Received CAN NUM: 2.
[0]:ts:22328740000, id:5, len:8, eff:1, rtr:0, data:0 1 2 3 4 5 6 7
[1]:ts:22328744000, id:6, len:8, eff:1, rtr:0, data:0 1 2 3 4 5 6 7
my_recv_thread end return!
Close CAN0 OK!
my_recv_thread end return!
Close CAN1 OK!
close_usb_device index:0
libusb_close return
Close Device OK!

```

Ресурсы

Драйвер и программное обеспечение Windows

- [Windows-драйвер USB-CAN-FD/USB-CAN-FD-B](#)
- [Программа отладки для Windows USB-CAN-FD/USB-CAN-FD-B](#)
- [Файлы библиотеки для вторичной разработки](#)

Демо-пример драйвера Linux (применимо только к USB-CAN-FD-B)

- [Демо-пример Linux для USB-CAN-FD-B](#)

Документация

- [Руководство по библиотеке API для вторичной разработки](#)

Часто задаваемые вопросы (FAQ)

Вопрос: Почему у резисторов 120 Ом каждого канала два переключателя и чем отличается включение одного переключателя от двух?

Ответ: При включении одного переключателя 120 Ом между H и L подключается один резистор 120 Ом. При включении второго переключателя параллельно подключается еще один резистор 120 Ом, поэтому результирующее сопротивление между H и L составляет 60 Ом.

Вопрос: Почему связь с внешним CAN-устройством работает некорректно, хотя двухканальный loopback-тест проходит успешно?

Ответ: Проверьте настройки скорости арбитражной фазы и фазы данных CAN. Если внешнее устройство работает по обычному CAN, например 500 кбит/с, установите 500 кбит/с. Для CAN FD может потребоваться отдельно задать скорость арбитражной фазы и скорость фазы данных. Разница в скоростях передачи между обычным CAN и CAN FD заключается в следующем:

Обычный CAN:

- Скорость передачи при арбитраже: при передаче данных узлы определяют приоритет посредством арбитража; обычно используется единая скорость, например 1 Мбит/с.
- Скорость передачи данных: в обычном CAN скорость арбитражной фазы и скорость фазы данных обычно совпадают.

CAN FD:

- Скорость передачи при арбитраже: ак и в обычном CAN, используется для определения приоритета; обычно применяется более низкая скорость, например 1 Мбит/с.
- Скорость передачи данных: может быть выше скорости арбитражной фазы и поддерживать более высокую скорость передачи данных (до 8 Мбит/с), что повышает эффективность обмена.