CP-CANFD USB to CAN FD Adapter

Datasheet

Version 1.0

Connective Peripherals Pte Ltd
1 Tai Seng Avenue, Tower A, #03-07 Singapore 536464
Tel.: +65 67430980

E-Mail (Support): support@connectiveperipherals.com Web: www.connectiveperipherals.com/products

1 Introduction

1.1 Functional Description

The CP-CANFD adapter is an industrial-grade device that adds one or two high-speed CAN FD (Controller Area Network Flexible Data Rate) interfaces to PCs or notebooks via a USB connection. It provides a 10-pin terminal block connector for connecting to CAN/CAN FD interfaces. It complies with CAN specifications ISO 11898-1 for CAN/CAN FD and meets the requirements of the ISO11898-2:2016 high-speed CAN specification. It also supports both ISO-11898-1:2015 and Bosch M_CAN revision 3.2.1.1.

The CP-CANFD adapter supports CAN 2.0A and CAN 2.0B protocols and has three supported modes which are Normal mode, Listen mode, and Loopback mode. It supports standard CAN data rates up to 1 Mbps and a CAN FD data rate up to 8 Mbps. The adapter also can support CAN bus message filter configuration and each CAN FD port can output DC +5V 100mA power through the terminal block for external CAN devices power usage.

The adapter has built-in selectable 120 Ohm termination resistors at each port for the CAN bus termination and is firmware upgradable for future firmware revisions. CAN bus signal lines are protected with +/-12 KV ESD protection. The CAN/CAN FD interface can be controlled over the Virtual COM port using simple ASCII commands and has LEDs to indicate the status of the CAN bus. The adapter is powered by the USB port and a USB A to USB B cable of 90cm length is provided. There is no external power adapter required.

A driver is provided for Windows 11 down to Windows 7, Windows Server, and Linux OS, and it supports Linux SocketCAN (SLCAN).

The CP-CANFD adapter is compact and robust. It comes with an ABS plastic enclosure and with a ventilated design suitable for harsh environments. The adapter is designed with a pair of built-in mounting brackets at the side and two T-type slots at the bottom for easy wall mounting.

The USB to CAN FD interface provides simple and quick connection to CAN and CAN FD networks. It supports CAN FD frames for both ISO and Bosch's Non-ISO standards. Also, the CP-CANFD adapter transmits and receives both CAN frames in the classical CAN 2.0 A/B format as well as CAN FD format. Being capable of switching to a faster or slower data rate dynamically, the CP-CANFD adapter contains more data capacity in a frame up to 64 bytes.

CP-CANFD adapter can be added flexibly to existing CAN networks. Since CAN FD is an extension to the original CAN bus protocol, the adapter is backward compatible with the CAN 2.0 A/B standards. This cost-effective USB to CAN FD adapter solution makes it easier than ever to enable CAN network connection to CAN FD with data rate up to 8 Mbps to your system.

1.2 LED Description

The CP-CANFD adapter uses two LEDs to indicate firmware initialization and CAN bus status. The green LED indicates CAN bus data activity while the red LED indicates a CAN bus error. Following is the definition of different LED combinations:

A: Power up (device initialized)

After CP-CANFD adapter powers up (device initialized), green LED and red LED will flash four times to indicate that the CP-CANFD adapter has been initialized.

B: CAN bus channel open/close

When the CAN bus channel opens, the green LED will turn on to indicate that the CAN bus channel is open. When the CAN bus channel closes, the green LED will turn off to indicate that the CAN bus channel is closed.

C: CAN bus Data Activity

When CAN data frame is transmitting or receiving, the green LED flashes continuously to indicate CAN bus data I/O activity.

D: CAN bus Error

When an error occurs on the CAN bus, the red LED flashes continuously to indicate CAN bus error.

LED Label	LED Color	Description		
		LED status	**State**	**Description**
Data	Green	Flashes 4 times	Initialization	Flashes four times together with red LED when device initialized
		ON	Operational	The device is in state OPERATIONAL (Channel is Open)
		OFF	Stopped	The device is in state STOPPED (Channel is Closed)
		Flashing continuously	Transmitting / Receiving	The device is transmitting or receiving data (Channel is Open)
Err	Red	Flashes 4 times	Initialization	Flashes four times together with green LED when device initialized
		Flashing continuously	Error	Error occurs on the device

Table 1 – LED Description

1.3 Features

- Adds one or two high speed CAN FD interfaces via a USB connection
- 10-pin terminal block connector for connecting to CAN bus interface
- Meets the requirements of the ISO 11898-2:2016 high speed CAN specification
- Complies to CAN specifications ISO 11898-1 for CAN and CAN FD
- CAN FD support both ISO-11898-1:2015 and Bosch M_CAN revision 3.2.1.1
- Supports CAN FD data rate up to 8 Mbps
- Supports CAN bus speed up to 1 Mbps
- Non-standard baud rates supported
- Supports CAN bus message filter configuration
- Provides +5V DC output at 100mA through pin-4 (port 1) and pin-9 (port 2) of terminal block
- Supports CAN 2.0A and CAN 2.0B protocols
- Supported CAN modes
    - Normal mode: normal operation on CAN bus
    - Listen mode: passive receiving of CAN frames
    - Loopback mode: transmitter also receives sent frames (for testing purposes)
- Built-in 120 Ohm termination resistors for CAN bus termination with a selectable jumper
- Firmware upgradable for future firmware revisions
- Galvanic isolation between PC and CAN bus connection up to 5700 Vrms (available only on CP-CANFD-1P-ISO and CP-CANFD-2P-ISO Adapter)
- CAN/CAN FD interface can be controlled over COM port using simple ASCII commands
- LEDs indicates CAN initialization and CAN bus status for monitoring port status
- Powered by USB port. No external power adapter required
- Includes one USB-A to USB-B cable. Cable length: 0.9meter
- Drivers provided for Windows 7 up to Windows 11, Windows Server, and Linux OS
- Supports Linux SocketCAN (SLCAN)
- Supported by CANHacker to test and analyze CAN frames
- Compact and robust ABS plastic enclosure with ventilation design suitable for harsh environment
- FCC, UKCA and CE compliant

2 Part Numbers/Ordering Information

Part Number	Description
CP-CANFD-1P	USB to single CAN FD Adapter; Non-Isolation version
CP-CANFD-1P-ISO	USB to single CAN FD Adapter; Isolation version
CP-CANFD-2P	USB to dual CAN FD Adapter; Non-Isolation version
CP-CANFD-2P-ISO	USB to dual CAN FD Adapter; Isolation version

Table 2 – Part Numbers / Ordering Information

Note: USB-A to USB-B cable (length: 90cm) will be provided.

Table of Contents

3 Specifications

The table below show the specifications of CP-CANFD adapter:

General	
USB Bus	USB 2.0 high speed 480 Mbps USB 1.1, USB 2.0, USB 3.0 compatible
CAN Bus	Supports CAN 2.0A and CAN 2.0B and CAN FD
CAN Controller	TCAN4551-Q1
CAN Signals	CAN_H, CAN_L, CAN_GND, CAN_V+
CAN Bus Speed	Up to 1 Mbps for CAN data transmit & receive Up to 8 Mbps for CAN FD transmit & receive Non-standard baud rate supported
CAN Bus Mode	Normal mode: normal operation on CAN bus Listen Only mode: passive receiving of CAN frames Loopback mode: transmitter also receives sent frames (for testing purposes)
Specification	ISO 11898-2 high speed CAN specification CAN FD support ISO-11898-1:2015 and Bosch M_CAN revision 3.2.1.1
Plug & Play	Supported
IRQ &IO Address	Assigned by system
CP-CANFD-1P	
Number of Ports	One
Connector	10-pin terminal block connector
Termination	120 Ohm terminator resistor on board
LED	Power, CAN bus data activity, CAN bus error
Protection	+/-12 KV ESD(HBM) protection for CAN signals
CP-CANFD-1P-ISO	
Number of Ports	One
Connector	10-pin terminal block connector
Termination	120 Ohm terminator resistor on board
LED	Power, CAN bus data activity, CAN bus error
Protection	+/-4 KV ESD(HBM) protection for CAN signals Isolation Protection Power: 3000 VDC CAN bus: 5700 Vrms for CAN signals
CP-CANFD-2P	
Number of Ports	Two
Connector	10-pin terminal block connector
Termination	120 Ohm terminator resistor on board (per port)
LED	Power, CAN bus data activity (per port), CAN bus error (per port)
Protection	+/-12 KV ESD(HBM) protection for CAN signals
CP-CANFD-2P-ISO	
Number of Ports	Two
Connector	10-pin terminal block connector
Termination	120 Ohm terminator resistor on board (per port)
LED	Power, CAN bus data activity (per port), CAN bus error (per port)
Isolation Protection	+/-4 KV ESD(HBM) protection for CAN signals Isolation Protection Power: 3000 VDC CAN bus: 5700 Vrms for CAN signals
Software Features	
O.S Driver Support	Windows 7 to Windows 11 OS Windows Server 2003 to 2012 R2 Linux
Power Requirement	
Power Input	Power supplied via USB (5V) connector

	No external power needed
Power Consumption	Max. 150mA@5VDC (CP-CANFD-1P & CP-CANFD-2P) Max. 250mA@5VDC (CP-CANFD-1P-ISO & CP-CANFD-2P-ISO)
Mechanical	
Casing	ABS Plastic enclosure with built-in mounting brackets Two T-shaped slots on the bottom for wall mounting
Dimensions	102 mm X 73 mm X 30.4 mm (L x W x H) 102 mm X 96.5 mm X 30.4 mm with ears (L x W x H)
Weight	100g

Table 3 – CP-CANFD Adapter Specifications

4 Installation

4.1 Hardware Installation

4.1.1 Jumpers for Termination Resistors

A CAN bus network requires a 120ohm termination resistor at each end of the CAN network. If additional termination resistors are present through the CANbus network or if one of the termination resistors is missing at one of the ends, the CANbus network will not communicate.

CP-CANFD adapter has provided a built-in termination resistor. Users may enable or disable termination by switching the jumper on the PCB board.

Note:

- JP1 is for USB to single port CP-CANFD adapter
  - CP-CANFD-1P
  - CP-CANFD-1P-ISO
- JP1 and JP2 are for USB to dual port CP-CANFD adapter
  - CP-CANFD-2P
  - CP-CANFD-2P-ISO

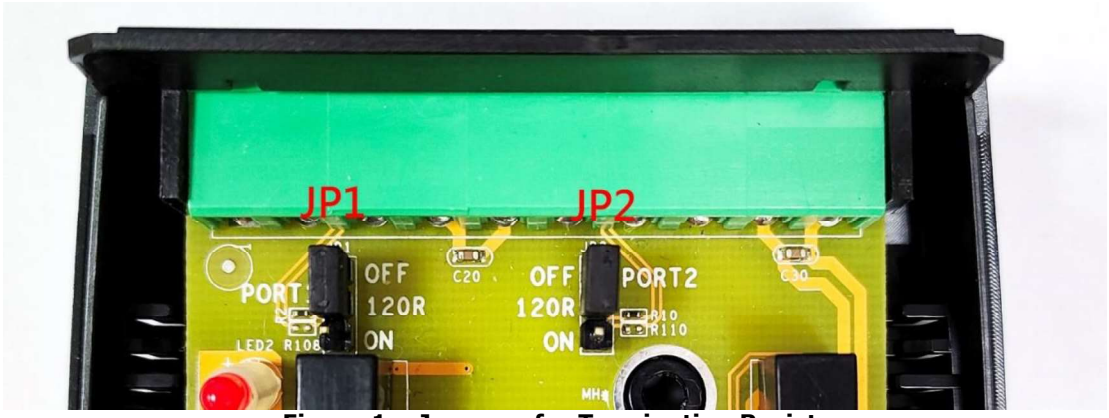

Figure 1 – Jumpers for Termination Resistors

JP1	JP2	Function
PIN 1-2 short		Disable 120Ω termination resistors
PIN 2-3 short		Enable 120Ω termination resistors

Table 4 – Jumpers for Enabling / Disabling Termination Resistors

4.1.2 Mounting

The CP-CANFD adapter comes with an ABS plastic enclosure. The adapter is designed with a pair of built-in mounting bracket and two T-type slots at the bottom for easy wall mounting.

4.1.2.1 USB to Single Port CAN FD Adapter

102mm X 73mm X30.4mm (L x W x H)
102mm X 96.5mm X30.4mm with brackets (L x W x H)

Figure 2 - CP-CANFD-1P/CP-CANFD-1P-ISO Adapter

4.1.2.2 USB to Dual Port CAN FD Adapter

Figure 3 - CP-CANFD-2P/CP-CANFD-2P-ISO Adapter

4.2 Device Driver Installation

4.2.1 Microsoft Windows

4.2.1.1 Installing Driver

The CP-CANFD adapter driver are available to download from
https://www.connectiveperipherals.com

Download the latest version of CP CDC Driver from the above link.

Figure 4 - CP CDC Driver

Double click the CP CDC Driver application file and click 'Install' to proceed for the installation.

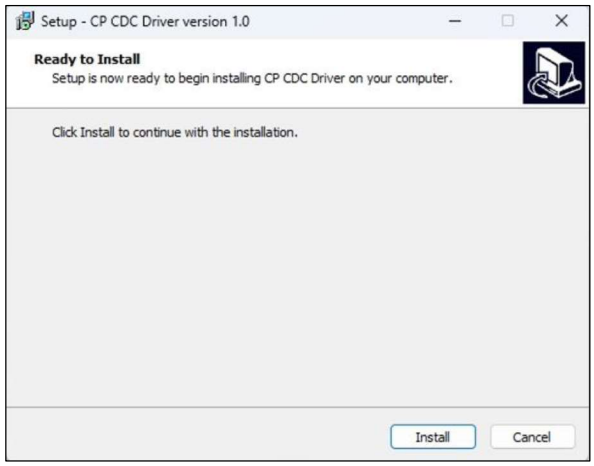

Figure 5 - Install CP CDC Driver

Follow the instructions until the installation finishes.

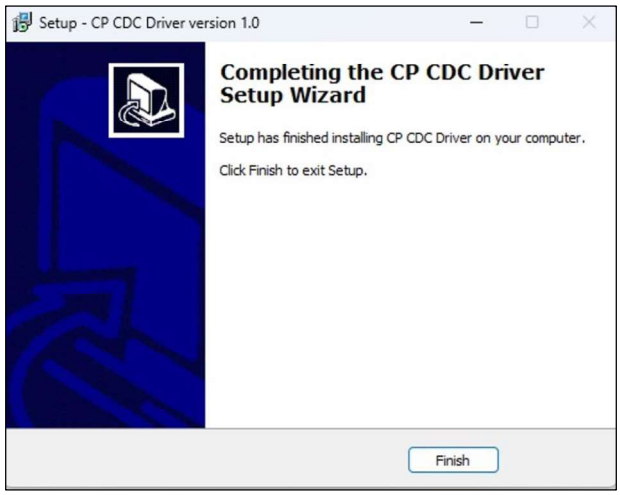

Figure 6 - Installation Complete

4.2.1.2 Changing COM Port Properties & COM Port Number

This feature is particularly useful for programs such as HyperTerminal, which only works with COM1 through COM4. Please ensure that you do not change to a COM port number that is already in use.

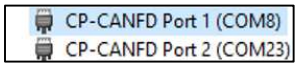

Figure 7 - COM Port Number

To change the virtual COM port properties:

1. Select the "CP-CANFD Port X (COMX)".
2. Click "Properties".
3. Select "Port Setting" and "Advanced".
4. Click the drop-down arrow on COM Port Number and scroll to the required COM port.
5. Select "OK".
6. Return to the Device Manager Screen. You will see that the CP-CANFD Port installation has been changed to the new COM Port Number.

4.2.2 Linux Driver (SocketCAN)

4.2.2.1 Installing Driver

SocketCAN is a set of open-source CAN drivers and a networking stack contributed by Volkswagen Research to the Linux kernel. Formerly known as Low Level CAN Framework (LLCF).

Installing SocketCAN is easy. In fact, it just needs to install can-utils packages. To proceed, uses the following command:

sudo apt-get install can-utils

By default, the SocketCAN device drivers are not automatically loaded by Linux at boot time on all systems: you may need to enable the relevant Linux kernel modules. You can manually load these modules with the following command:

sudo modprobe can
sudo modprobe vcan
sudo modprobe slcan

4.2.2.2 Using SocketCAN (SLCAN) with CP-CANFD Adapter

To use CP-CANFD adapter with SocketCAN, you must establish a "link" between the drivers and the hardware. This is done with the following command:

sudo slcand -o -c -s8 -S3000000 /dev/ttyUSB0 slcan0

The -s parameter allows to configure different interface speed as indicated below.

ASCII Command	CAN Bitrate
s0	10 Kbit/s
s1	20 Kbit/s
s2	50 Kbit/s
s3	100 Kbit/s
s4	125 Kbit/s
s5	250 Kbit/s
s6	500 Kbit/s
s7	800 Kbit/s
s8	1000 Kbit/s

Table 5 – ASCII Command for CAN Bitrate

If everything worked fine, you should now see the can0 network device. To vary it, just type the following command:

sudo ip link set slcan0 up

5 Pinout Information

The following are the pinouts of 10-pin terminal block connector for CP-CANFD adapter.

5.1.1 Single Port CP-CANFD Adapter

10	9	8	7	6	5	4	3	2	1

Figure 8 - CP-CANFD-1P/CP-CANFD-1P ISO Adapter Terminal Block

The pinout of the terminal block connector for single port CP-CANFD adapter is shown in Table 6.

Note: Pin 6 to 10 is NC pin for single port CP-CANFD adapter

Pin Number	Port Number	Signals	Description
1	PORT 1	CAN_GND	Signal ground
2	PORT 1	CAN_H	CAN_H bus line (dominant level is high)
3	PORT 1	CAN_L	CAN_L bus line (dominant level is low)
4	PORT 1	CAN_V+	Outputs DC+5V at 100mA
5	PORT 1	CAN_GND	Signal ground

Table 6 – Pin-Out for CP-CANFD-1P/CP-CANFD-1P-ISO Terminal Block

5.1.2 Dual Port CP-CANFD Adapter

10	9	8	7	6	5	4	3	2	1

Figure 9 - CP-CANFD-2P/cP-CANFD-2P-ISO Adapter Terminal Block

The pinout of the terminal block connector for dual port CP-CANFD adapter is shown in Table 7.

Pin Number	Port Number	Signals	Description
1	PORT 1	CAN_GND	Signal ground
2	PORT 1	CAN_H	CAN_H bus line (dominant level is high)
3	PORT 1	CAN_L	CAN_L bus line (dominant level is low)
4	PORT 1	CAN_V+	Outputs DC+5V at 100mA
5	PORT 1	CAN_GND	Signal ground
6	PORT 2	CAN_GND	Signal ground
7	PORT 2	CAN_H	CAN_H bus line (dominant level is high)
8	PORT 2	CAN_L	CAN_L bus line (dominant level is low)
9	PORT 2	CAN_V+	Outputs DC+5V at 100mA
10	PORT 2	CAN_GND	Signal ground

Table 7 – Pin-Out for CP-CANFD-2p/CP-CANFD-2P-ISO Terminal Block

6 Functions

6.1 ASCII Commands

The following ASCII commands provide a means of utilizing an interactive terminal program, or communication through a standard COM or TTY port. The CP-CANFD adapter can be registered as a virtual serial port on the host computer. With simple ASCII commands the CP-CANFD adapter can be controlled over this serial port. Users can send/receive commands from any simple serial terminal program.

Example: Set bitrate to 500Kbps, open CAN channel, send CAN frame (ID = 002h, DLC = 3, Data = 11 22 33), close CAN:

Command	Response	Function
S6[CR]	[CR]	Set bitrate of USB CAN adapter to 500Kbps
O[CR]	[CR]	Open CAN channel
t0023112233[CR]	z[CR]	Send CAN message (ID = 002h, DLC = 3, Data = 11 22 33)
C[CR]	[CR]	Close CAN channel

Table 8 –Example of sending CAN Command

6.2 Command list

The commands are line based and terminated with newline character CR (0xD). On error the response will be 0x7 (BELL).

The "help" command (**'H', 'h' or '?')** will list supported commands.

Command	Response	Function
H[CR]	[CR]	List all supported commands
h[CR]	[CR]	
?[CR]	[CR]	

Table 9 – "Help" Command

Example: H[CR]

Return Code

List of Supported Commands:
'O' – Open the channel in Normal mode
'L' – Open the channel in Listen Only mode
'Y' – Open the channel in Loopback mode
'C' – Close CAN Channel
'S' – Set standard CAN bitrate
's' – Set non-standard CAN bitrate
't' – Transmit a standard frame
'T' – Transmit an extended frame
'r' – Transmit a standard remote request frame
'R' – Transmit an extended remote request frame
'Z' – Set timestamp on/off
'm' – Set acceptance mask
'M' – Set acceptance filter
'F' – Read status flag
'V' – Check software version
'N' – Check serial number
'm' – Set acceptance mask
'M' – Set acceptance filter
'RST' – Reset USB CAN Adapter
'H', 'h' or '?' – List supported commands

6.2.1 Opening the CAN Bus Channel

The CAN bus channel will be opened with the command O[CR], L[CR], or Y[CR]. The command O[CR] will open the CAN bus channel in normal operation mode, the command L[CR] will open the CAN bus channel in listen only mode, in which no bus interaction will be done from the controller. the command Y[CR] will open the CAN bus channel in a loop-back mode, in which the USB to CAN adapter will also receive the frames that it sends. Before you use one of the commands, you should set a bitrate with the commands S or s.

Command	Response	Function
O[CR]	[CR]	Open the channel in Normal mode
L[CR]	[CR]	Open the channel in Listen Only mode
Y[CR]	[CR]	Open the channel in Loopback mode

Table 10 – ASCII Command for Open CAN Bus Channel

6.2.2 Closing the CAN Bus Channel

The CAN bus channel will be closed with the command C[CR]. The command can only be used if the CAN bus channel is open.

Command	Response	Function
C[CR]	[CR]	Close the CAN channel if it is opened

Table 11 – ASCII Command to Close CAN Bus Channel

6.2.3 Setting CAN Bitrate (Standard)

The CAN bus bitrate can be set with the command SX[CR]. The command can only be used if the CAN bus channel is closed.

Command	Response	Function
S00[CR]	[CR]	Set the CAN bus bitrate to 5K
S0[CR]	[CR]	Set the CAN bus bitrate to 10K
S1[CR]	[CR]	Set the CAN bus bitrate to 20K
S2[CR]	[CR]	Set the CAN bus bitrate to 50K
S3[CR]	[CR]	Set the CAN bus bitrate to 100K
S4[CR]	[CR]	Set the CAN bus bitrate to 125K
S5[CR]	[CR]	Set the CAN bus bitrate to 250K
S6[CR]	[CR]	Set the CAN bus bitrate to 500K
S7[CR]	[CR]	Set the CAN bus bitrate to 800K
S8[CR]	[CR]	Set the CAN bus bitrate to 1M

Table 12 – ASCII Command for CAN Bus Bitrate

Example: S6[CR] will be set USB CAN adapter to 500K bps CAN Bitrates.

6.2.4 Transmitting a Standard CAN Frame

Transmitting a standard CAN frame (ID: 11 bit) over a CAN bus can be done with the command tiiildddd…dd[CR]. The return value will be z[CR] or the normal error byte (BELL). The command is only available when the CAN bus channel is open.

Command	Response	Function
tiiildddd…dd[CR]	z[CR]	Transmits a standard CAN message (11 bit) over the CAN bus

Table 13 – ASCII Command for Transmitting Standard CAN Frame

iii: Standard CAN frame (11 bit) identifier in hexadecimal format (000-7FF).

l: CAN data length (0-8) DLC, with the maximum value being 8 (8 bytes).

dd: Data byte value in hexadecimal format (00-FF). The number of bytes must be equal to the data length field.

Example: t00231199FF[CR] will send a standard CAN frame with ID = 002h, DLC = 3, Data = 11 99 FF.

6.2.5 Transmitting a Standard Remote Request CAN Frame

Transmitting a standard remote request CAN frame (ID: 11 bit) over a CAN bus can be done with the command riiil[CR]. The return value will be z[CR] or the normal error byte (BELL). The command is only available when the CAN bus channel is open.

Command	Response	Function
riiil[CR]	z[CR]	Transmits a standard remote request (11 bit) over the CAN bus

Table 14 – ASCII Command for Transmitting Standard Remote Request CAN Frame

iii: Standard remote request CAN frame (11 bit) identifier in hexadecimal format (000-7FF).

l: CAN data length to request (0-8) DLC, with the maximum value being 8 (8 bytes).

Example: r0023[CR] will send a standard remote request CAN frame with ID = 002h, DLC = 3 and request 3 data bytes.

6.2.6 Transmitting an Extended CAN Frame

Transmitting an extended CAN frame (ID: 29 bit) over a CAN bus can be done with the command Tiiiiiiiildddd...dd[CR]. The return value will be Z[CR] or the normal error byte (BELL). The command is only available when the CAN bus channel is open.

Command	Response	Function
Tiiiiiiiildddd...dd[CR]	Z[CR]	Transmits an extended CAN frame (11 bit) over the CAN bus

Table 15 – ASCII Command for Transmitting Extended CAN Frame

iiiiiiii: Extended CAN frame (29 bit) identifier in hexadecimal format (00000000-1FFFFFFF).

l: CAN data length (0-8) DLC, with the maximum value being 8 (8 bytes).

dd: Data byte value in hexadecimal format (00-FF). The number of bytes must be equal to the data length field.

Example: T1FFFFFFF3112233[CR] will send an extended CAN frame with ID = 1FFFFFFFh, DLC = 3, data = 11 22 33.

6.2.7 Transmitting an Extended Remote Request CAN Frame

Transmitting an extended remote request CAN frame (ID: 29 bit) over a CAN bus can be done with the command Riiiiiiiil[CR]. The return value will be Z[CR] or the normal error byte (BELL). The command is only available when the CAN bus channel is open.

Command	Response	Function
Riiiiiiiil[CR]	Z[CR]	Transmits an extended remote request (29 bit) over the CAN bus

Table 16 – ASCII Command for Transmitting Extended Remote Request CAN Frame

iiiiiiii: Extended remote request CAN frame (29 bit) identifier in hexadecimal format (00000000-1FFFFFFF).

l: CAN data length to request (0-8) DLC, with the DLC maximum value being 8 (8 bytes).

Example: R100000023[CR] will send an extended remote request CAN frame with ID = 10000002h, DLC = 3 and request 3 data bytes.

6.2.8 Setting Timestamps ON/OFF

The timestamp command will set the timestamp functionality of received frames ON or OFF. This command is only available when the CAN channel is closed.

Command	Response	Function
Z1[CR]	[CR]	Set the timestamp functionality on received frames ON
Z0[CR]	[CR]	Set the timestamp functionality on received frames OFF

Table 17 – ASCII Command for Setting Timestamps

6.2.9 Setting Acceptance Mask

The acceptance mask, in conjunction with the acceptance code (M), defines which CAN message frames (i.e., of a specific ID or range of CAN IDs) will be passed to the serial interface. The acceptance mask value corresponds to bits within a range of valid CAN IDs for either standard or extended CAN frames. This command is only active if the CAN channel is initiated and not opened. Set Acceptance Mask (m) command should be executed *prior* to Set Acceptance Code (M).

Note:

The CAN channel will revert to its prior state after execution. For example, if the channel is open when this command is executed, the channel will update the setting and return to the open state.

Command	Response	Function
miii[CR]	[CR]	Set acceptance mask for standard CAN frame (11 bit) identifier
miiiiiiii[CR]	[CR]	Set acceptance mask for extended CAN frame (29 bit) identifier

Table 18 – ASCII Command for Setting Acceptance Mask

iii = standard 11-bit CAN mask (0x000 through 0x7FF)

iiiiiii = extended 29-bit CAN mask (0x00000000 through 0x1FFFFFFF)

A value of "0" in a bit location indicates that the bit location ID value is to be *ignored* when filtering messages.

Default is to pass all frames (acceptance mask = 0x000 for standard messages and 0x00000000 for extended messages)

Example: m700[CR] set acceptance mask to check bits 10, 9 and 8 against the filter. Bits 7 through 0 are ignored as "don't care". Use the acceptance mask in conjunction with the acceptance code, which is explained next.

6.2.10 Setting Acceptance Code

The acceptance code/filter, in conjunction with the acceptance mask (m), defines which CAN message frames (i.e., of a specific ID or range of CAN IDs) will be passed to the serial interface. The acceptance code value corresponds to a valid CAN IDs for either standard or extended CAN frames. This command is only active if the CAN channel is initiated and not opened.

The Set Acceptance Mask (m) command should be executed *prior* to the Set Acceptance Code (M) command.

Note:

The CAN channel will revert to its prior state after execution. For example, if the channel is open when this command is executed, the channel will update the setting and return to the open state.

Command	Response	Function
Miii[CR]	[CR]	Set acceptance code for standard CAN frame (11 bit) identifier
Miiiiiii[CR]	[CR]	Set acceptance code for extended CAN frame (29 bit) identifier

Table 19 – ASCII Command for Setting Acceptance Code

iii = standard 11-bit CAN mask (0x000 through 0x7FF)
iiiiiii = extended 29-bit CAN mask (0x00000000 through 0x1FFFFFFF)

Default is to pass all frames (acceptance code = 0x7FF for standard messages and 0x1FFFFFF for extended messages)

Example: m1FF[CR] sets acceptance code to receive standard messages with the CAN ID of 0x1FF. If used in conjunction with the acceptance mask example above, frames of the range 0x100 through 0x1FF will be passed, and all other CAN IDs will be blocked.

6.2.11 Getting Status Flags

Users can use the command F[CR] to get the status bits when an error occurs. A two-byte BCD number is returned to correspond to the 8-bits of the internal register of the CAN controller.

Command	Response	Function
F[CR]	XX[CR]	Get CAN bus status

Table 20 – ASCII Command for getting Status Flags

Return Codes

XX[CR]

XX = CAN bus status (A bit set to "1" indicates a true condition):

Bits 2, 1, 0**:** Last Error Code (LEC), The LEC field holds a code, which indicates the type of the last error to occur on the CAN bus.

LEC Bits 2, 1, 0	Meaning
Error Code 0 **0, 0, 0**	No error.
Error Code 1 **0, 0, 1**	Stuff error: more than 5 equal bits in a sequence have occurred in a part of a received message where this is not allowed.
Error Code 2 **0, 1, 0**	Form error: a fixed format part of a received frame has the wrong format.
Error Code 3 **0, 1, 1**	ACK Error: the message this CAN core transmitted was not acknowledged by another node.
Error Code 4 **1, 0, 0**	Bit 1 error: during the transmission of a message (Apart from the arbitration field), the device wanted to send a recessive level (bit of logical value "1"), but the monitored bus value was dominant.
Error Code 5 **1, 0, 1**	Bit 0 error: Bit 1 error: during the transmission of a message (or acknowledged bit, or active error flag, or overload flag), the device wanted to send a dominant level (bit of logical value "0"), but the monitored bus value was recessive. During the bus-off recovery, this status is set each time a sequence of 11 recessive bits have been monitored. This enables the CPU to monitor the proceedings of the bus-off recovery sequence (indicating the bus is not stuck at dominant or continuously disturbed).

Error Code 6 1, 1, 0	CRC error: the CRC checksum was incorrect in the message received; the CRC received for an incoming message does not match with the calculated CRC for the received data.
Error Code 7 1, 1, 1	Unused: no CAN bus event was detected since the CPU wrote this value to the LEC.

Table 21 – Return Code Description

Bit 3: Transmitted a message successfully

1 = Since this bit was last reset by CPU, a message has been successfully (error-free and acknowledged by at least one other node) transmitted.

0 = Since this bit was last reset by CPU, no message has been transmitted.

Bit 4: Received a message successfully

1 = A message has been successfully received since this bit was last reset by CPU (independent of the result of acceptance filtering).

0 = No message has been successfully received since this bit was last reset by CPU

Bit 5: Error Passive (Read only)

1 = The CAN core is in the error passive state as defined in the CAN specification.

0 = The CAN core is in the error active.

Bit 6: Error Warning Status (Read only)

1 = At least one of the error counters in the EML (Error Management Logic) has reached the error warning limit of 96.

0 = Both error counters are below the error warning limit of 96.

Bit 7: Bus-off Status (Read only)

1 = The CAN Module is in bus-off state.

0 = The CAN Module is not in bus-off state.

<BELL> = ERROR

Bit 0 ~ Bit 7 returned to correspond to the 8-bits of the internal register of the CAN controller.

6.2.12 Get Version Information

The command V[CR] to retrieve the current firmware version of the CP-CANFD adapter.

Command	Response	Function
V[CR]	VXXXX[CR]	Get the current firmware version of the CP-CANFD adapter

Table 22 – ASCII Command for get Version Information

This command is always available and will return the version information formatted like this: VXXXX[CR].

6.2.13 Get Serial Number

The command N[CR] will retrieve the serial number of the CP-CANFD adapter.

Command	Response	Function
N[CR]	TXXXXXXXX[CR]	Get the serial number of the CP-CANFD adapter

Table 23 – ASCII Command for get Serial Number

This command is always available and will return the decimal serial number like this: TXXXXXXX[CR].

6.2.14 Reset CP-CANFD Adapter

The command RST[CR] will reset the CP-CANFD adapter.

Command	Response	Function
RST[CR]	-	Reset the CP-CANFD adapter

Table 24 – ASCII Command for Reset CP-CANFD Adapter

This command is always available.

7 Tools

7.1 Firmware Upgrade

The CP-CANFD adapter firmware can be updated for bug fixes and enhanced features. You can use our tool program to upgrade the firmware contents via serial port.

The firmware upgrade program (USB-CAN-FD_update.exe) can be downloaded from https://www.connectiveperipherals.com .

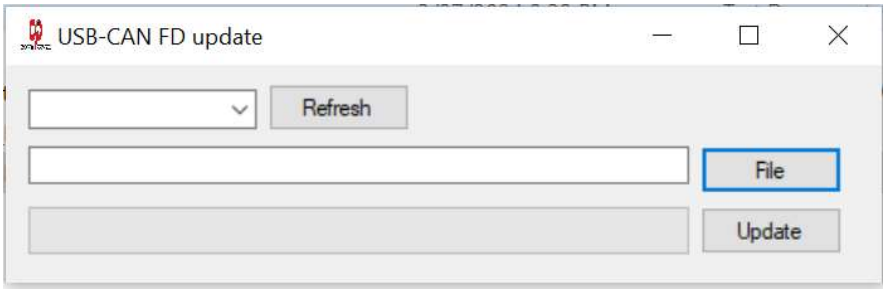

Figure 10 - USB-CAN FD Update Tool

Step 1: Click Refresh button to get the COM port number with in used.

Step 2: Select the COM port number which the CP-CANFD adapter used.

Step 3: Click File button to select the firmware file and click Update button to upgrade CP-CANFD adapter.

7.2 CANHacker

CANHacker is a Windows application software for analyzing and transmitting/receiving CAN frames. The CANHacker software has a friendly interface and is easy to use. Through the software, users can easily test and analyze the CAN frames.

The following shows its main panel:

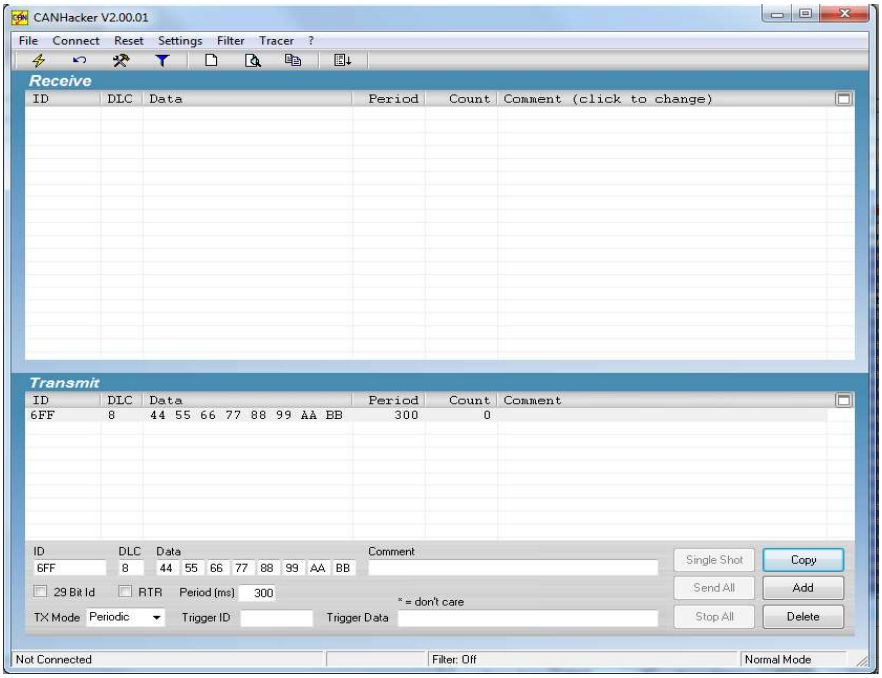

Figure 11 - CANHACKER Main Panel

The following sections will briefly introduce the necessary steps on how to use the software.

7.2.1 Settings Procedure for Selecting and Configuring the USB to CAN Adapter

1. Open CANHacker and click "Settings" under the menu.

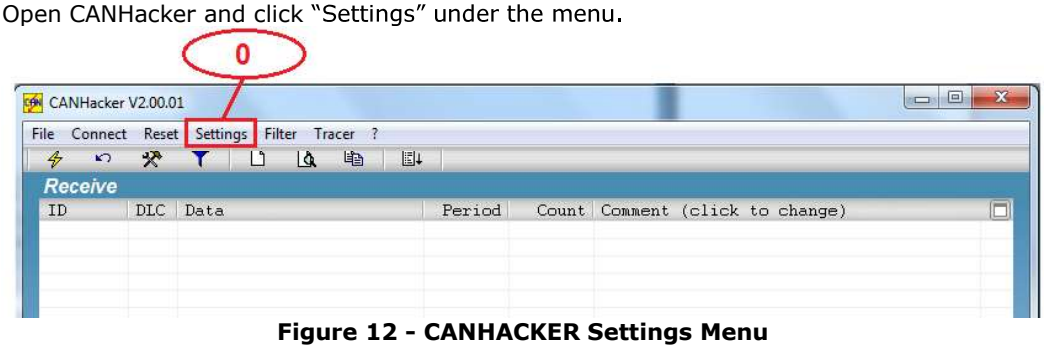

Figure 12 - CANHACKER Settings Menu

2. Select COM port of the USB to CAN adapter.
3. Check "RTS HS" to enable RTS handshake function.
4. Check "Time Stamp" to enable timestamp function.
5. Select CAN Baudrate for the CAN bus operating speed.
6. Finally, click "OK" to finish the settings and return to the main panel.

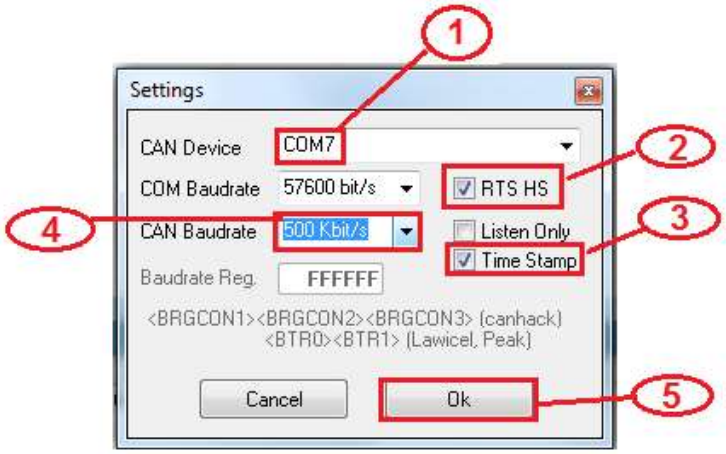

Figure 13 - CANHACKER Settings Tab

You may connect the CP-CANFD adapter after configuration. Click "Connect", as shown in the figure, to start the CANHacker software operation.

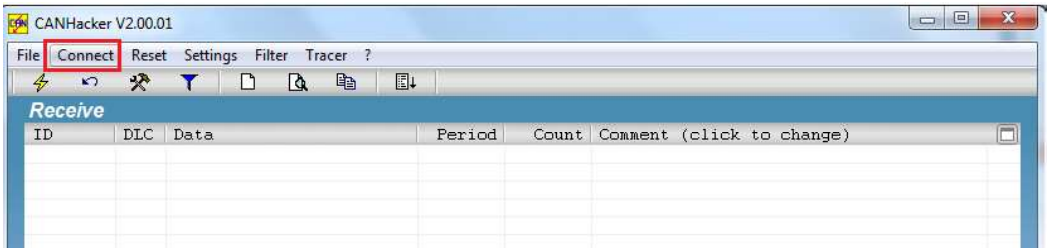

Figure 14 - Start CANHACKER Software Operation

When CP-CANFD adapter successfully connects, you will find the message "Connected to XXX kbits/s", firmware version VXXXX and operation mode at the bottom of the main panel.

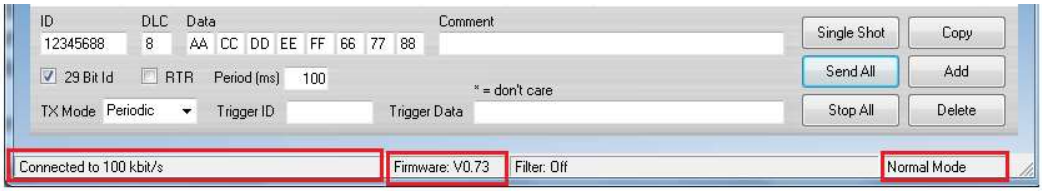

Figure 15 – Connection Successful

7.2.2 Receiving CAN Frames

When CANHacker receives CAN frames from another CAN node, it will show all CAN frame messages in the middle of main panel. The CAN frame messages includes ID, DLC, Data, Period, Count.

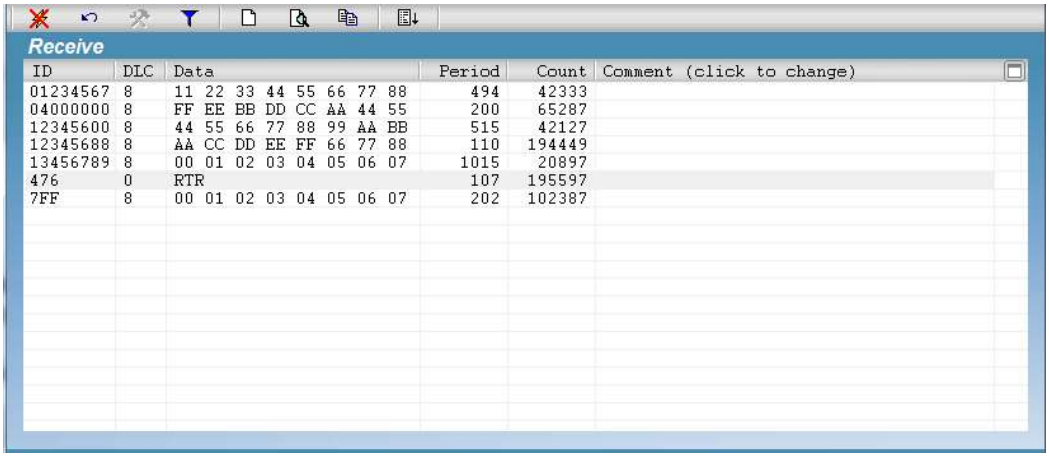

Figure 16 – CANHACKER Receiving CAN Frame

7.2.3 Sending CAN Frames

CANHacker provides many parameters for sending CAN frames to another CAN node, you can set the following parameters on the bottom of the main panel for CAN data transmission:

Figure 17 – CANHACKER Set CAN Data Transmission Parameter

Select transmit an extended CAN Frame (29 bits ID) or a standard CAN frame (11 bits ID).

Check "29 Bit Id" 

to transmit an extended CAN Frame (29 bits ID) and uncheck "29 Bit Id" 

to transmit a standard CAN frame (11 bits ID).

Select remote request frame mode or transmit CAN frame mode.

Check "RTR" 

for a remote request frame mode or uncheck "RTR" 

for transmit CAN frame mode.

Enter CAN frame messages in the respective fields, including ID, DLC, Data.

Figure 18 – Example of CAN Frame Message

In "TX Mode" dialog box, you can select "off", "Periodic", "RTR", "Trigger" modes.

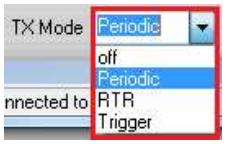

Figure 19 – TX Mode

When "Periodic" mode is selected, you can enter "Period (ms)" to send CAN frames message repeatedly (enter "500" to send CAN messages every 500ms).

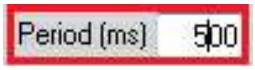

Figure 20 – Periodic Mode Setting

To send a single CAN frame message, click "Single Shot". Click "Send All" to send CAN frames message repeatedly.

To stop sending CAN frame messages, click "Stop All".

Figure 21 – Send / Stop CAN Frame Message

7.2.4 Assistant Features

There are many assistant features included in CANHacker, as shown in the figure below:

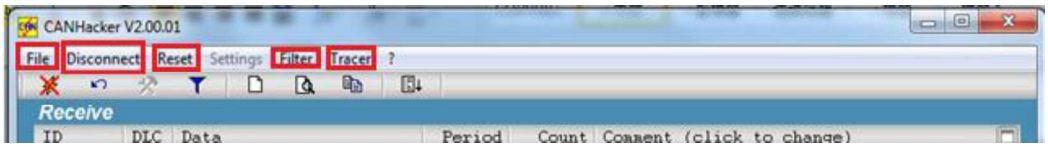

Figure 22 – Other Assistant Features in CANHACKER

Saving data to file or loading data from file:

Select **"File"** option to save Rx List, Trace, Tx List, Command List and Load Trace, Tx List, Command List.

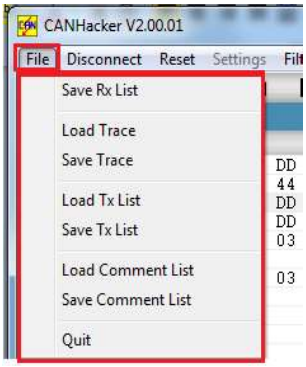

Figure 23 – CANHACKER File Menu

Click **"Disconnect"** to stop CANHacker.

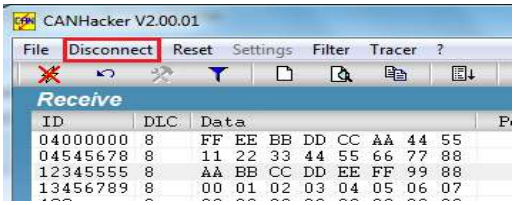

Figure 24 – CANHACKER for Disconnect

Click "Reset" to renew the received CAN frame messages and reset the transmission (received) count.

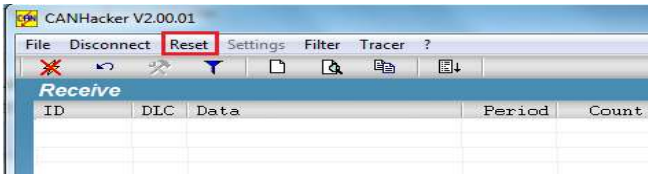

Figure 25 – CANHACKER for Reset

Select **"Filter"** to set mask filter and range filter.

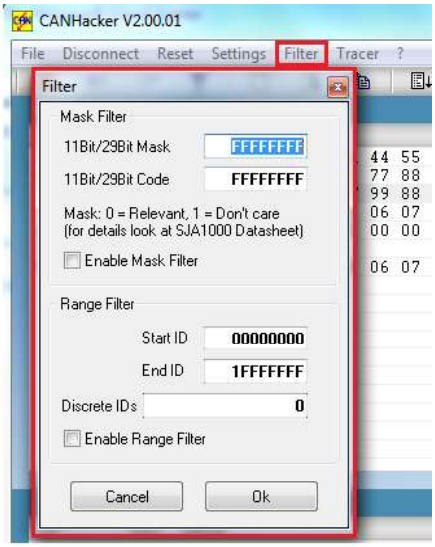

Figure 26 – CANHACKER for Filter Setting

Select **"Tracer"** or **"Monitor"** to trace or monitor the CAN frame messages.

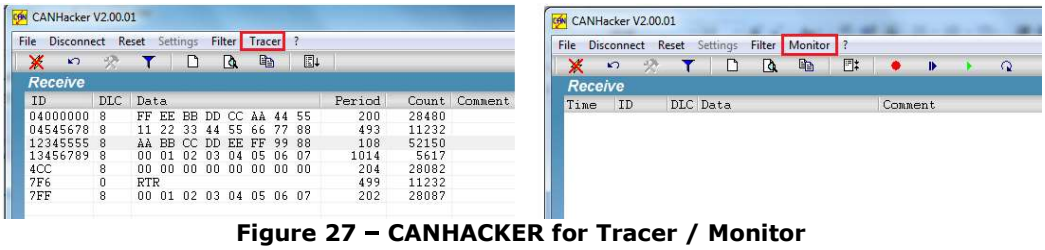

Figure 27 – CANHACKER for Tracer / Monitor

7.3 CP CAN FD Utility

CP CAN FD Utility is a Windows application software for testing and transmitting/receiving CAN FD frames. CP CAN FD Utility is an easy-to-use software, through the software users can easily test and analyze the CAN FD frames.

The CP CAN FD Utility can be downloaded from https://www.connectiveperipherals.com.

The following figure shows its main panel:

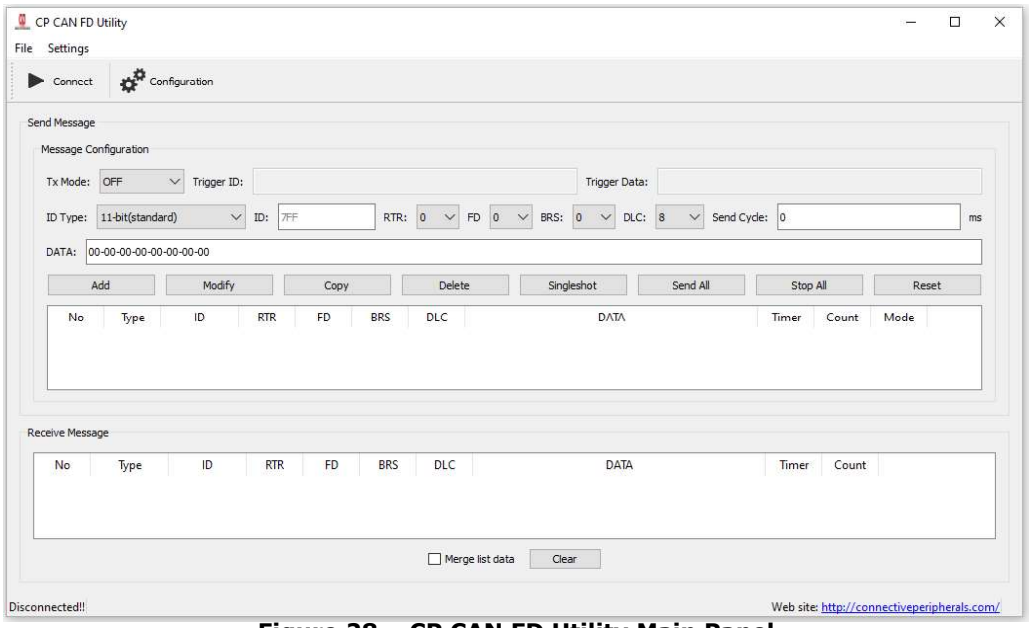

Figure 28 – CP CAN FD Utility Main Panel

7.3.1 Settings Procedure for Selecting and Configuring the USB to CAN FD Adapter

Figure 29 – CP CAN FD Utility Menu

1. Open CP CAN FD Utility and click "**Configuration**" under the menu.

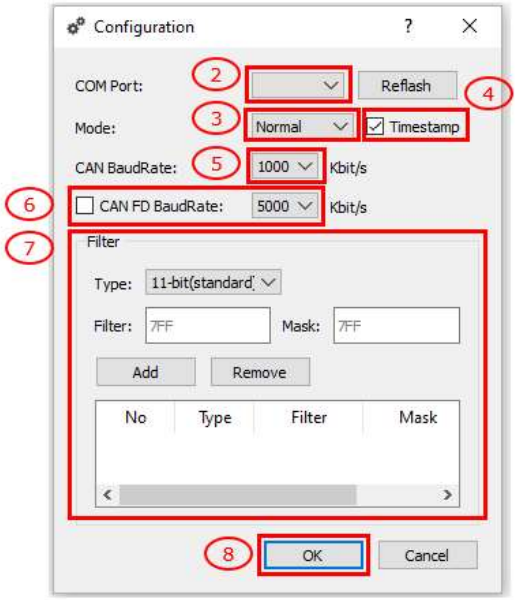

Figure 30 – CP CAN FD Utility Configuration

2. Select COM port of the USB to CAN FD adapter.
3. Select Mode to open the CAN bus adapter in loopback or listen only operation mode, otherwise the CAN bus adapter will open in normal operation mode.
4. Check "TimeStamp" to enable timestamp function.
5. Select CAN Baudrate for the CAN bus operating speed.
6. Check CAN FD Baudrate and select CAN FD Baudrate for the CAN FD operating speed.
7. Set the mask and filter to specify the CAN IDs that are passed or blocked.
8. Finally, click "OK" to finish the settings and return to the main panel.

Note:

Before you set the **"Mask Filter"** function, you need to unplug the USB adapter. After setting the value of "Mask" and "Filter", connect the USB adapter again to enable the "Mask Filter" function.

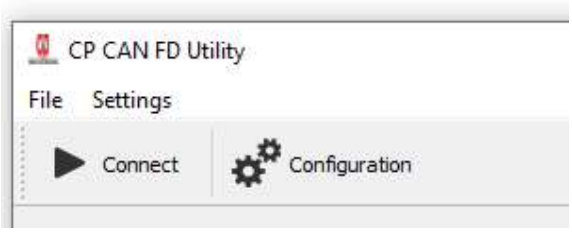

Figure 31 – Connect to CP CAN FD Utility

Users may connect the CP CAN FD adapter after configuration. Click "**Connect**" to start the CP CAN FD Utility.

Figure 32 – Connection Status

When CP CAN FD adapter successfully connects, user will find the message "Connected|COMxx|BR:xxxkbps" at the bottom of the main panel.

7.3.2 Receiving CAN Frames

When CP CAN FD utility receives CAN frames from another CAN node, it will show all CAN frame messages in Receive Message panel of the CP CAN FD Utility. Users may merge the data from the same ID by checking the 'Merge list data' box and clear the data in the Receive Message panel by clicking "Clear".

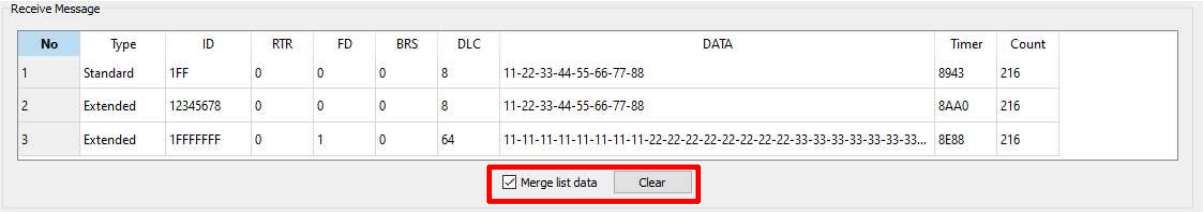

Figure 33 – CP CAN FD Utility - Receive Message

7.3.3 Sending CAN Frames

CP CAN FD Utility provides many parameters for sending CAN frames to another CAN node, you can set the following parameters in the Send Message panel for CAN data transmission:

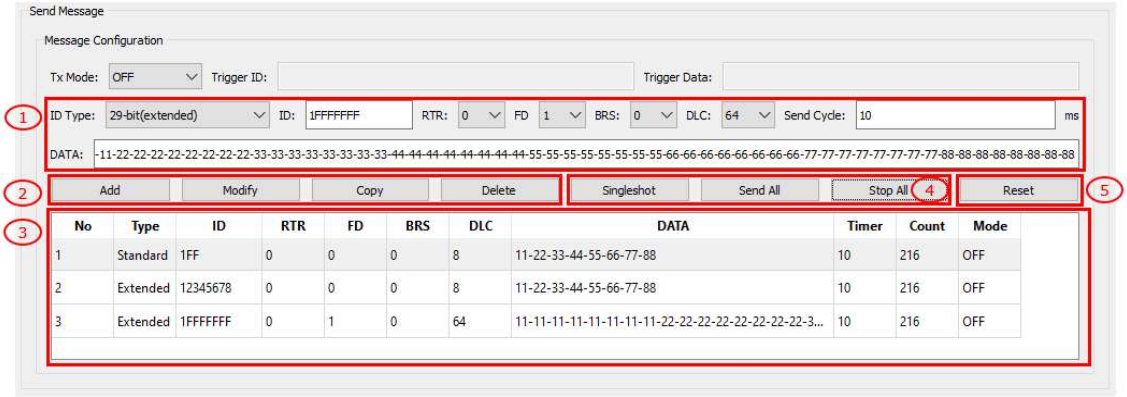

Figure 34 – CP CAN FD Utility - Send Message

1. Select transmit an extended CAN frame (29 bits ID) or a standard CAN frame (11 bits ID). Enter CAN frame messages in the respective fields.
2. Click "Add" to add a new send CAN frame message.
   Click "Modify" to modify a send CAN frame message.
   Click "Copy" to copy a send CAN frame message repeatedly.
   Click "Delete" to delete send CAN frame message.
3. CAN frame message to transmit will show in the data list.
4. Click "Singleshot" to send a single CAN frame message.
   Click "Send All" to send CAN frames message repeatedly.
   Click "Stop All" to stop sending CAN frame messages.
5. Click "Reset" to reset the "Count" column in the data list.

7.3.4 Assistant Features

The below figures illustrate some of the wide range of features included in CP CAN FD Utility

The **"File"** menu has features like *"Save Receive"*, *"Load Send"* and *"Save Send"*.

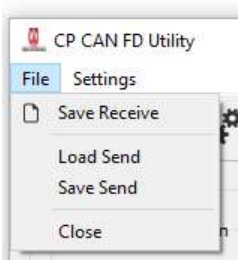

Figure 35 – CP CAN FD Utility File Menu

The **"Settings"** menu offers the *"Auto reconnect"* feature. The CP CAN FD adapters will auto reconnect after unplugging and replugging if this feature is enabled.

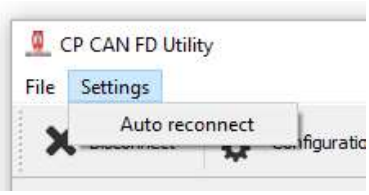

Figure 36 – Settings – Auto reconnect CP CAN FD Utility

Click **"Disconnect"** to close the open channel of CP CAN FD adapter.

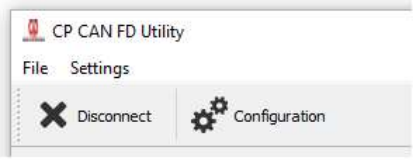

Figure 37 – Settings – Disconnect CP CAN FD Utility

7.4 BUSMASTER

BUSMASTER is an open-source software tool for simulating, analyzing and testing data bus systems such as CAN, LIN.

A guide to using CP CAN FD adapter with BUSMASTER tool can be found below.

Step1: Install busmaster-installers-3.2.2

Step2: Download the CP_BUSMASTER patch from https://www.connectiveperipherals.com.

Step3: Backup BusmasterDriverInterface.dll file. Copy and replace all files in CP_BUSMASTER patch to BUSMASTER installed folder

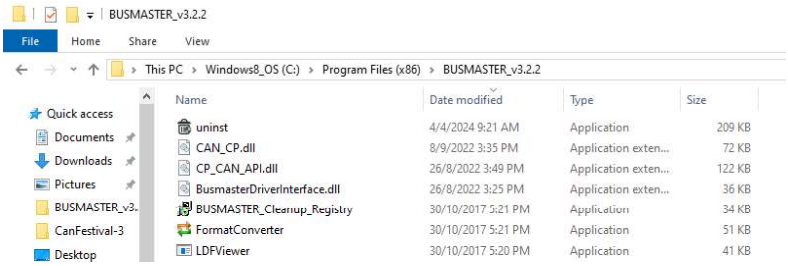

Figure 38 – Adding File in CP_BUSMASTER Patch to BUSMASTER Installed Folder

Step4: Open BUSMASTER software and select CP CAN-API for CP CAN FD adapter.

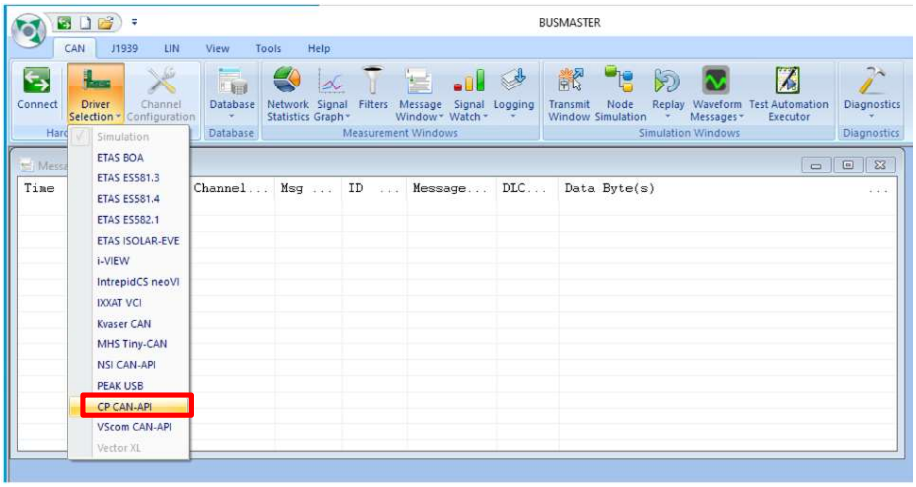

Figure 39 – Selecting CP CAN-API

Step5: Click **"Advanced"** button to show the CP CAN Device Config, Click Search for Devices on COM-ports, then select the COM port you want to use.

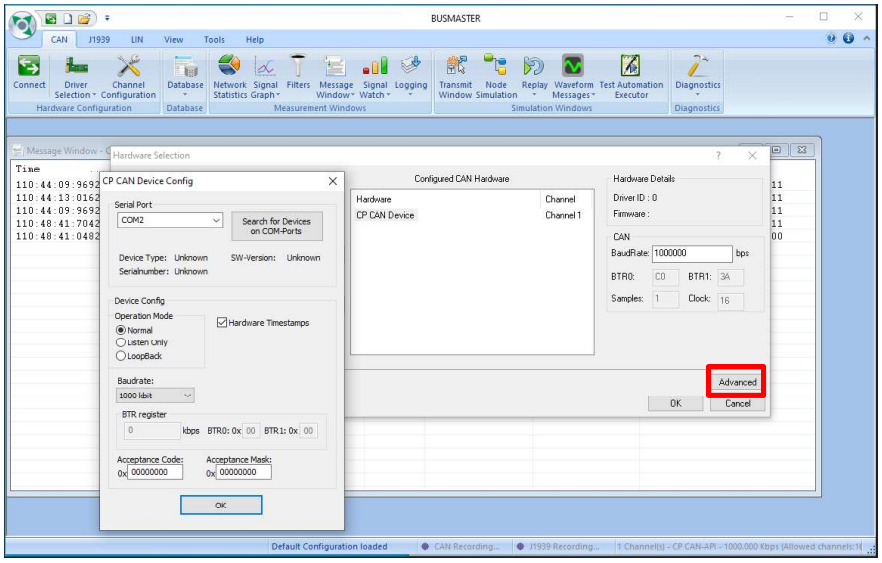

Figure 40 – CP CAN Device Configuration

8 Contact Information

Global Headquarters – Singapore

Connective Peripherals Pte Ltd
1 Tai Seng Avenue
Tower A
#03-07
Singapore 536464

Tel: +65 67430980

E-Mail (Sales) sales@connectiveperipherals.com
E-Mail (Support) support@connectiveperipherals.com
Web Site URL http://www.connectiveperipherals.com
Web Shop URL http://www.connectiveperipherals.com

Appendix A – List of Tables & Figures

List of Tables

List of Figures

Appendix C – Revision History

Document Title: CP-CANFD USB to CAN FD Adapter Datasheet
Document Reference No.: CP_000091
Clearance No.: CP#078
Product Page: https://connectiveperipherals.com/
Document Feedback: Send Feedback

Revision	Changes	Date
1.0	Initial Release	16-05-2024

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Interface Modules ***category:***

Click to view products by Connective Peripherals ***manufacturer:***

Other Similar products are found below :

FAB118 ATX6022/14GP7 ATX6022/8 9441560000 73-546-001 60016-023 PCE-DP10-00A1E 60016-002 60006-004 SNAP-PAC-SB2 60011-093 ADAM-4541-C ADAM-4510S-F USB-CARD-READER USB-RS485 USB-SATA RSM232D TD502D232H RS232 TO RS485 TD302D232H TD501D232H RSM232 RSM232D TD301D232H TD302D232H TD502D232H 20270 60567 30255 50260 20295 40961 20277 50261 ZHB5105J EUR0001H ZHB2301G 50985 USO9ML4 TTL/CAN-USB 4000-68000-4170000 4000-68000-0220000 4000-68512-0940003 4000-68000-0090000 6ES73317KF020AB0 6ES71556AA010BN0 6ES71936AR000AA0 GPS50M FAB8020785 1020882