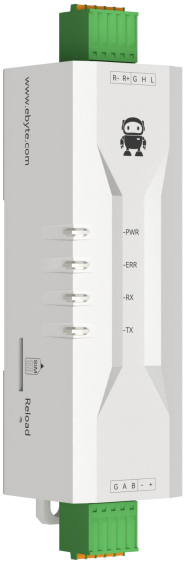
ECAN-101 Intelligent Protocol Converter



Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

contents

1. OVERVIEW 3

1. 1 Introduction 4

1. 2 Features and functions 4

1. 3 Application Scenario 4

2. PRODUCT SPECIFICATIONS AND CHARACTERISTICS 4

2.1 Basic parameters 5

2.2 Factory default parameters 5

3. QUICK START 6

Software preparation: You can download "Serial Port Assistant (X-COM)" in the "Related Downloads" of the official product website and configure the host computer 6

3.1 Connecting devices 6

3.1.1 RS485 connection method 8

3.1.2 CAN connection method 8

3.2 Device transparent debugging 9

3.3 Upper computer configuration 10

4. INTRODUCTION TO HARDWARE PARAMETER DESIGN 15

4.1 Design Introduction 16

4.2 Dimensional drawing 17

5. MODE DESCRIPTION 17

5.1 Data conversion method 17

5.1.1 Transparent conversion mode 17

5.1.2 Transparent tape identification mode 20

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.1.3 Protocol mode 23

5.1.4 Modbus mode 25

5.1.5 Custom Protocol Mode 28

6. AT INSTRUCTION 30

6.1 Enter AT command 31

6.2 Exit AT command 31

6.3 Query version 31

6.4 Restore default parameters 32

6.5 Echo Settings 32

6.6 Serial port parameters 32

6.7 Setting/querying CAN information 33

6.8 Set/Query Module Conversion Method 33

6.9 Set/query the filtering method of the CAN bus 34

6.10 Set/query frame header and footer data 34

6.11 Set/Query Identity Parameters 34

6.12 Set/Query Identity Parameters 35

6.13 Set/query transmission direction 36

6.14 Set/Query Filter Parameters 36

6.15 Delete filter parameters that have already been set 36

REVISION HISTORY 38

ABOUT US 38

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1. Overview

1.1 Introduction

ECAN-101 is a small intelligent protocol conversion product independently developed by Chengdu Ebyte Electronic Technology Co., Ltd. The product uses a wide voltage power supply of 8V ~ 28V and integrates 1 CAN-BUS interface and 1 RS485 interface, which can realize bidirectional conversion between CAN and RS485 different protocol data. This product supports serial port AT command configuration and host computer configuration of device parameters and working modes. It supports five data conversion modes: transparent conversion, transparent conversion with logo, protocol conversion, Modbus RTU conversion, and customization (user). In MODBUS mode, It can play the role of a single CAN interface replacing multiple 485 lines. At the same time, the ECAN-101 intelligent protocol converter has the characteristics of compact size and easy installation. It has extremely high cost performance in CAN-BUS product development and data analysis applications. It is a reliable assistant for engineering applications, project debugging and product development.

1.2 Features and functions

- Bidirectional conversion between CAN and RS485 protocol data;
- Support transparent conversion, transparent tape identification conversion, protocol conversion, Modbus RTU conversion, and custom protocol conversion;
- Support RS485 interface parameter configuration;
- Support the configuration of AT command parameters;
- Support the configuration of upper computer parameters;
- Support AT commands, upper computer, and button recovery to factory settings;
- Equipped with various status indicators such as power indicator light and status indicator light;
- Multi master and multi slave function, replacing multi-channel 485 with a single CAN interface;
- Equipped with a 120 ohm terminal resistor;

1.3 Application Scenario

- Industrial control and other CAN-BUS networks
- Networking of Automobile and Railway Equipment
- Security and fire protection network
- Underground remote communication
- Public Address System
- Parking lot equipment control
- Smart home and smart building

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2. Product specifications and characteristics

2.1 Basic parameters

main parameter	specifications
supply voltage	DC 8V~28V, recommended for 12V or 24V power supply
Working current	9.5mA@12V (standby)
Interface type	Serial interface: RS485:2.5 spring terminal, crimping method CAN interface: 2.5 spring terminals, crimping method
Interface characteristics	Serial interface: standard RS485 level interface, baud rate range: 600bps~460800bps CAN interface: follows ISO 11898 standard, supports CAN2.0A/B, CAN baud rate range: 6Kbps~1000Kbps
CAN terminal resistance	Integrated, 2.5 spring terminals, crimping method, short circuiting R120+and R120- for use
Communication indicator light	On board power supply (PWR), error reporting (ERR), receiving (RX), sending (TX) indicator lights
working temperature	-40 ℃~85 ℃, industrial grade
Working humidity	15% to 90% RH, without condensation

2.2 Factory default parameters

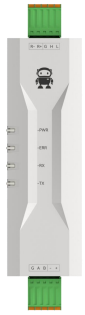
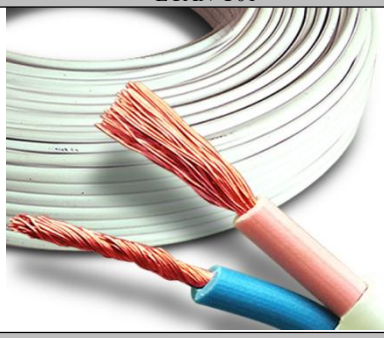
RS485	Serial Port Baud Rate	115200 bps
	Parity check	none
	Data bits	8
	Stop bit	1
	Flow control	close
CAN	CAN Baud Rate	100K bps
	CAN ID	0x0000000
Default working mode	Transparent conversion mode	Receive all data types

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3. Quick start

Hardware preparation: USB to RS485 converter * 1, power supply * 1 (recommended DC 12V power supply), computer * 1, USB to CAN device * 1 ECAN-U01(recommended)

		
ECAN-101	computer	ECAN-U01
		
DC 12V switching power supply	USB to RS485	Several cables

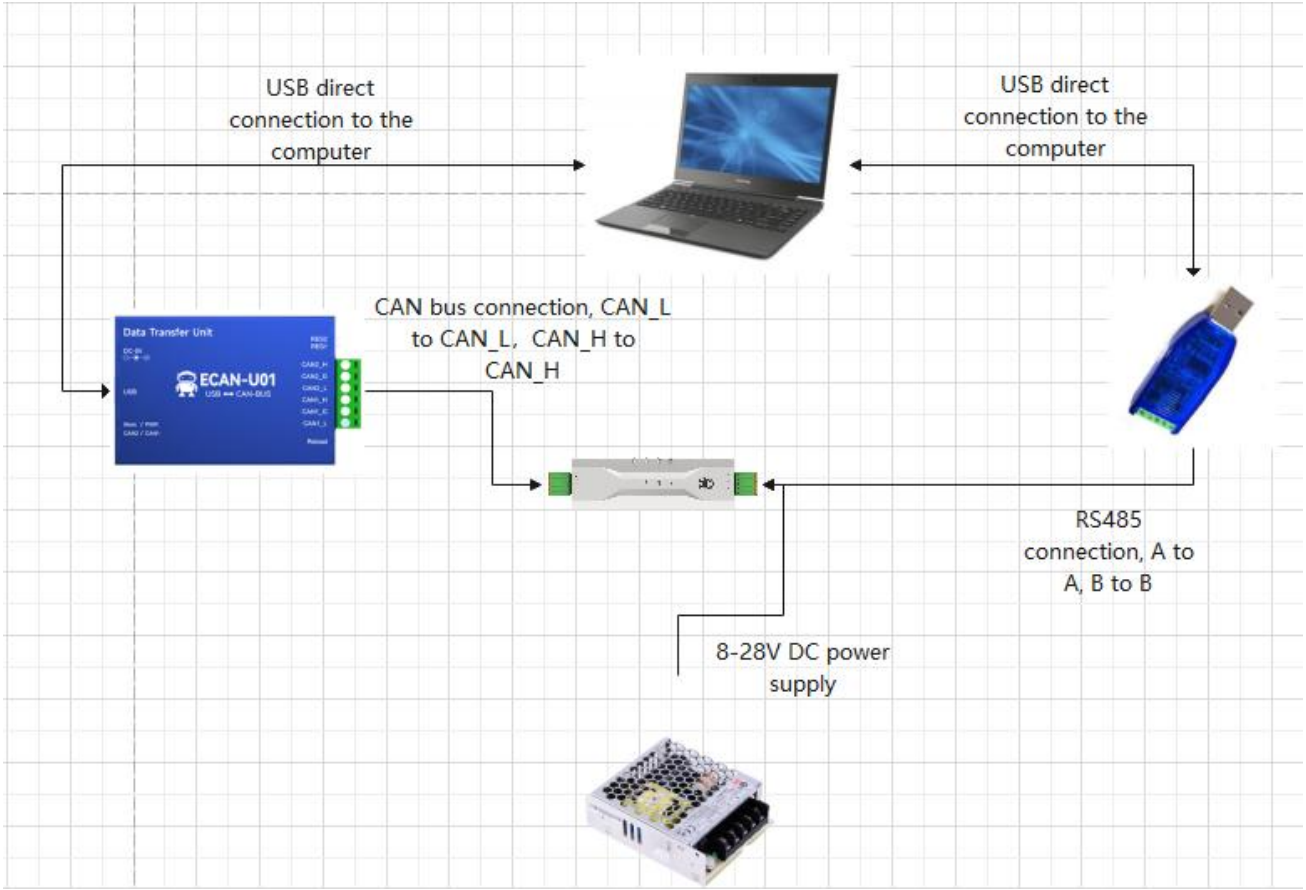
Software preparation: You can download "Serial Port Assistant (X-COM)" in the "Related Downloads" of the official product website and configure the host computer

3.1 Connecting devices

Here, using this device as ECAN-U01an example to demonstrate how to get started quickly.

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



First, use a USB to RS485 converter to connect the device and the computer. The RS485_A and RS485_B of the converter correspond to the RS485_A and RS485_B of the connected device; the USB end of the converter is directly connected to the computer;

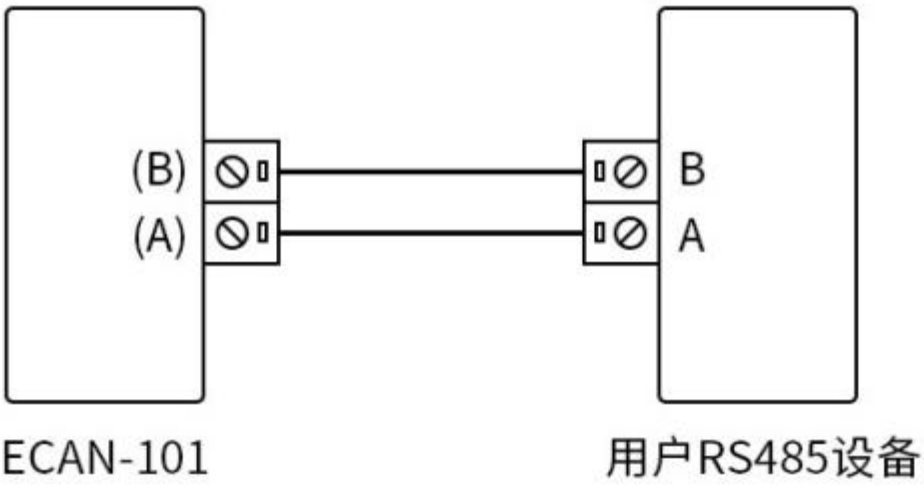
Secondly, use ECAN-U01 to connect the device and the computer; the CAN end of ECAN-U01 is connected to the CAN end of the device, CAN-L is connected to CAN-L, and CAN-H is connected to CAN-H. The USB end of ECAN-U01 is directly connected to the computer.

Note: When using the USB to RS485 converter for the first time, you need to install software and drivers. Please consult the relevant manufacturer for detailed usage..

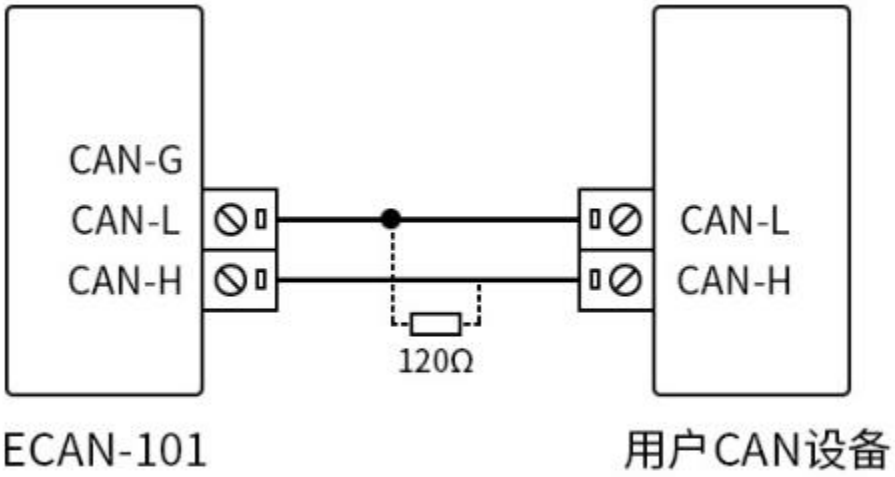
Copyright ©2012 – 2023，Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.1.1 RS485 connection method



3.1.2 CAN connection method



The linear topology is the most commonly used in CAN bus wiring specifications. The two main lines branch out to each node. Configure appropriate terminal resistors at both ends of the backbone to achieve impedance matching (typically 120 ohms within 2km).

According to the ISO 11898 specification, in order to enhance the reliability of CAN-bus communication, terminal matching resistors (120 Ω) are usually added to the two endpoints of the CAN-bus network. The size of the terminal matching resistance is determined by the characteristic impedance of the transmission cable. For example, if the characteristic impedance of a twisted pair is 120 Ω, the two endpoints on the bus should also integrate a 120 Ω terminal resistance

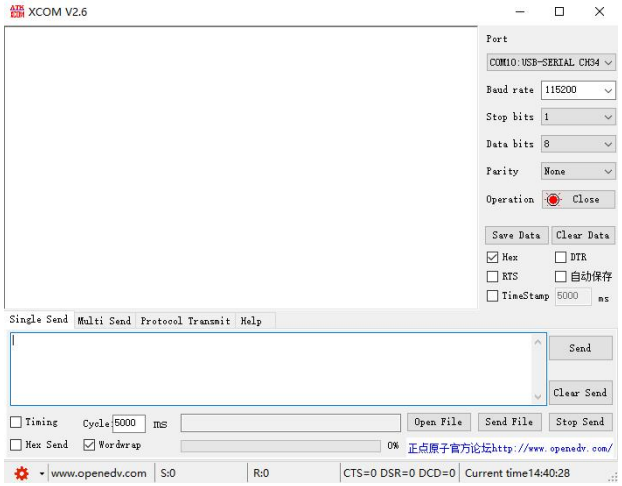
Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

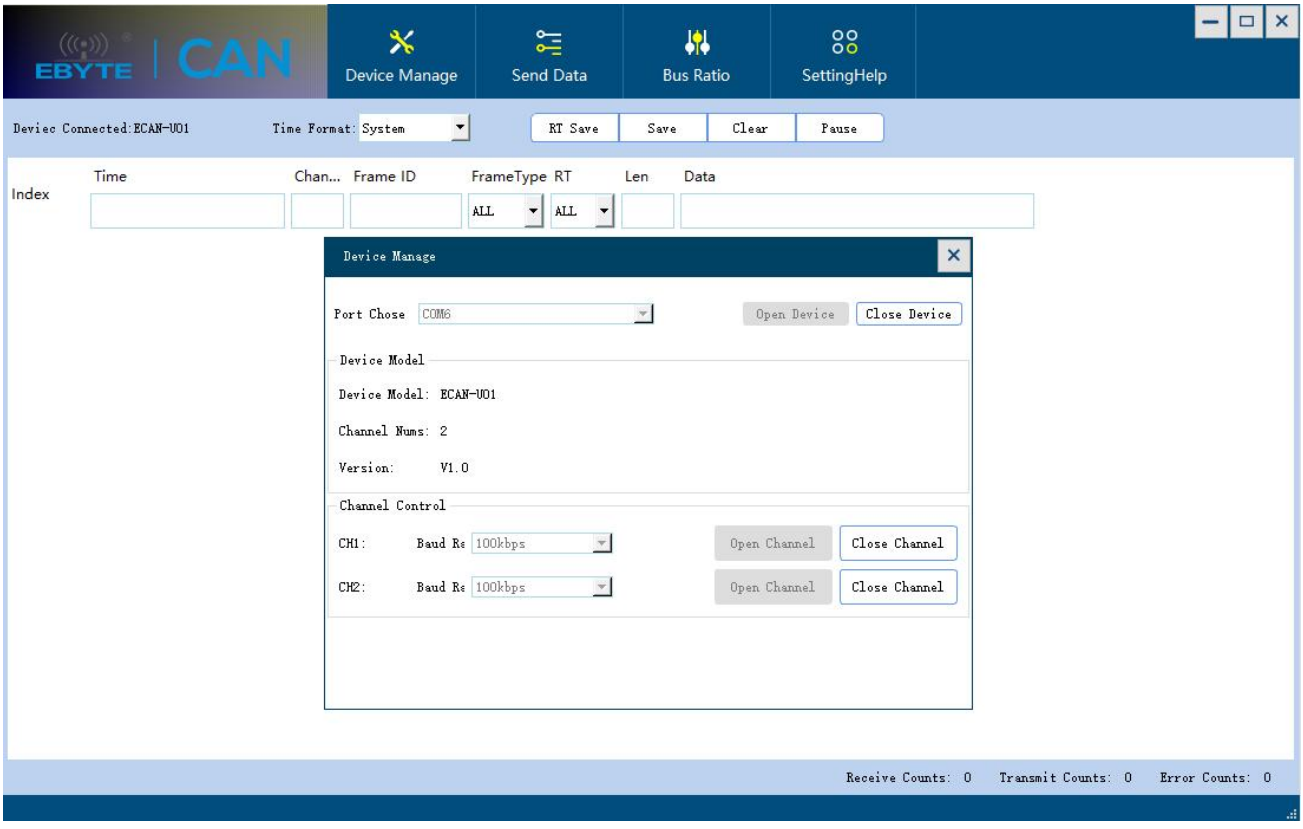
3.2 Device transparent debugging

After connecting the wires as described above, power on all equipment.

Open the serial port debugging assistant on the computer, select the serial port number used for the device's serial port (after entering the software, select the serial port number connected to ECAN-101, and click "Connect Device" to establish the connection). If the user does not know the serial port number, they can view it by entering the device manager of the PC And set the serial port parameters as the device default parameters (baud rate 115200, 8 data bits, no parity bit, 1 stop bit);



Open ECAN-U01 to debug the upper computer and set the CAN baud rate to 100K (ECAN-101 defaults to the factory CAN baud rate);



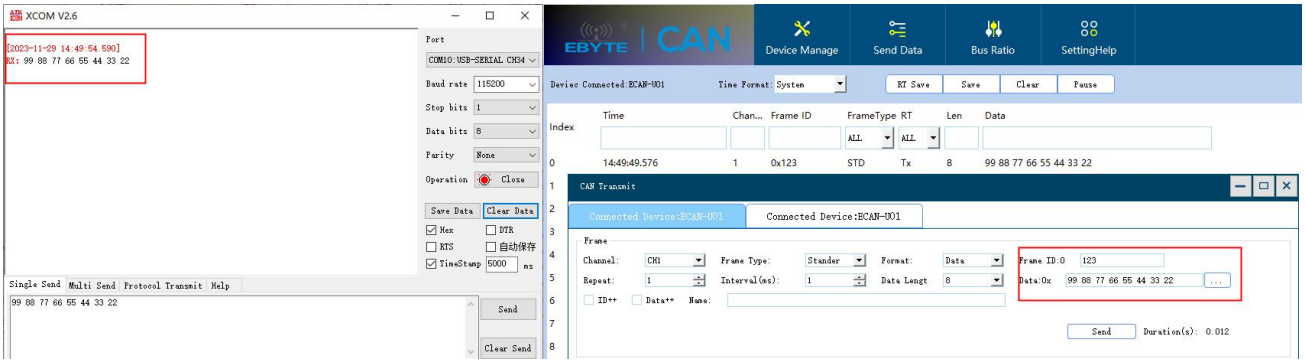
After the above steps, the transparent conversion debugging can begin:

Firstly, convert the CAN frame to a serial frame:

The CAN debugger (ECAN-U01) sends data: frame ID: 0x0123, data domain: 99 88 77 66 55 44 33 22. The device CAN receives the data sent by the CAN debugger (ECAN-U01) below, and the serial assistant displays the CAN data forwarded by the RS485 interface (hexadecimal display): 99 88 77 66 55 44 33 22.

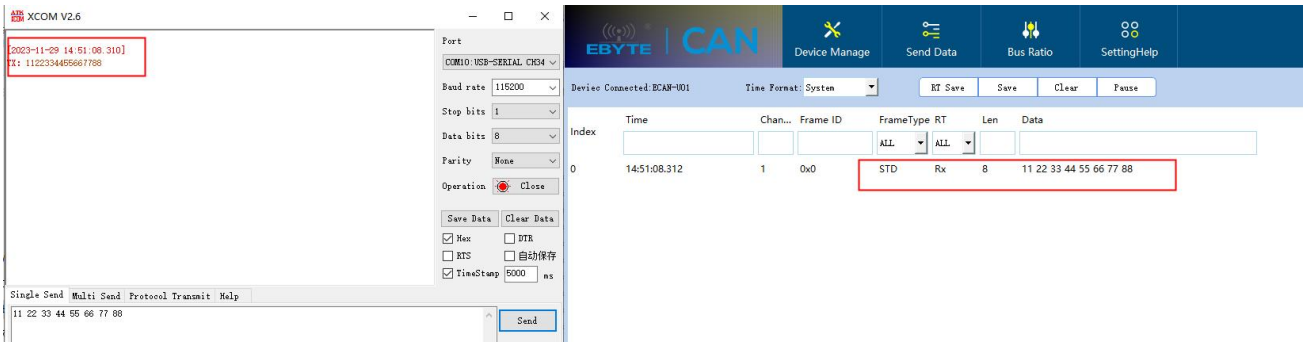
Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



Convert serial frames to CAN frames:

Send data in hexadecimal from the serial debugging assistant: 11 22 33 44 55 66 77 88. The data converted from the device CAN end can be displayed in the CAN debugger (ECAN-U01) below. The data is: 11 22 33 44 55 66 77 88, frame ID: 0x0000, standard frame.



From the above debugging operation, it can be seen that the device has achieved transparent conversion between serial frames and CAN frames.

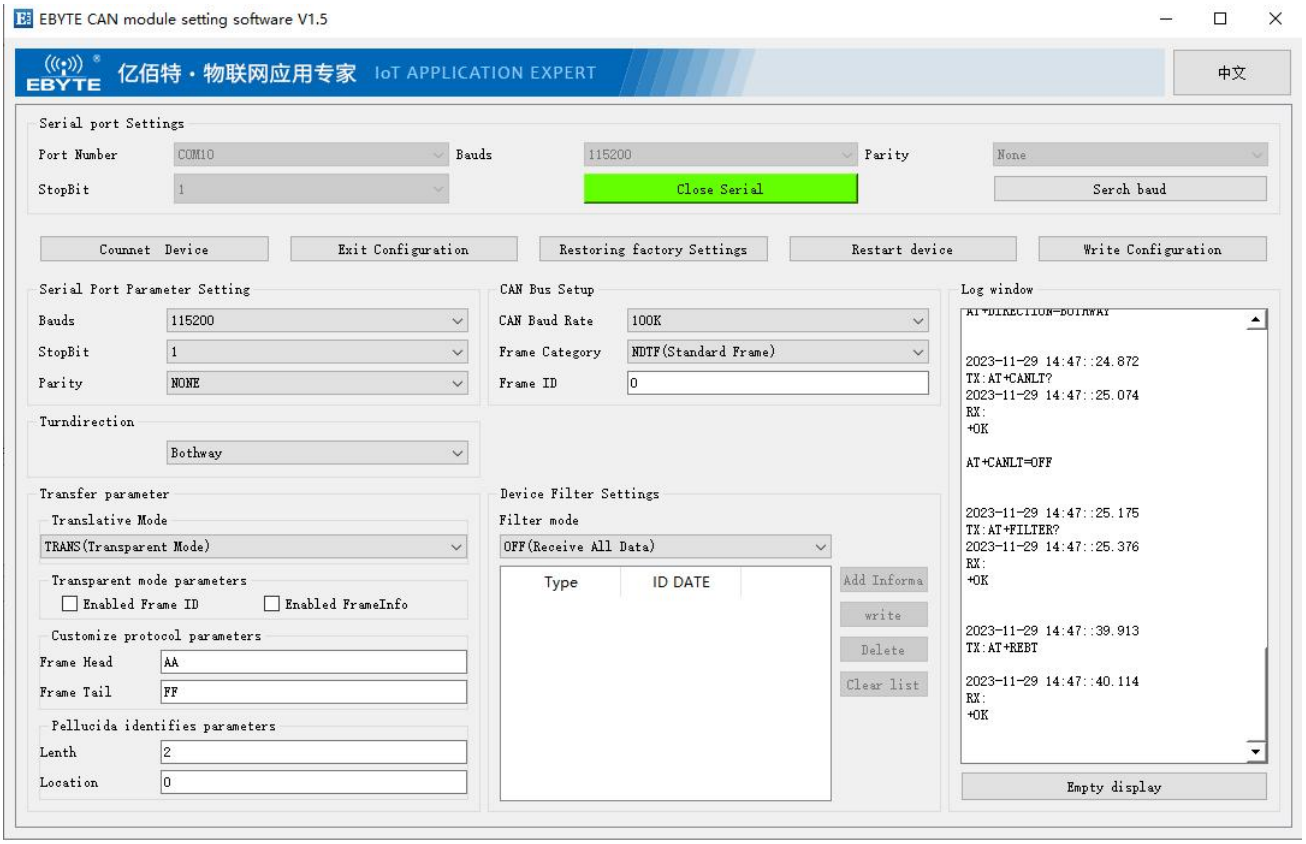
3.3 Upper computer configuration

The device can be configured with device parameters through AT commands or upper computer. Here is a brief introduction to the usage of upper computer configuration.

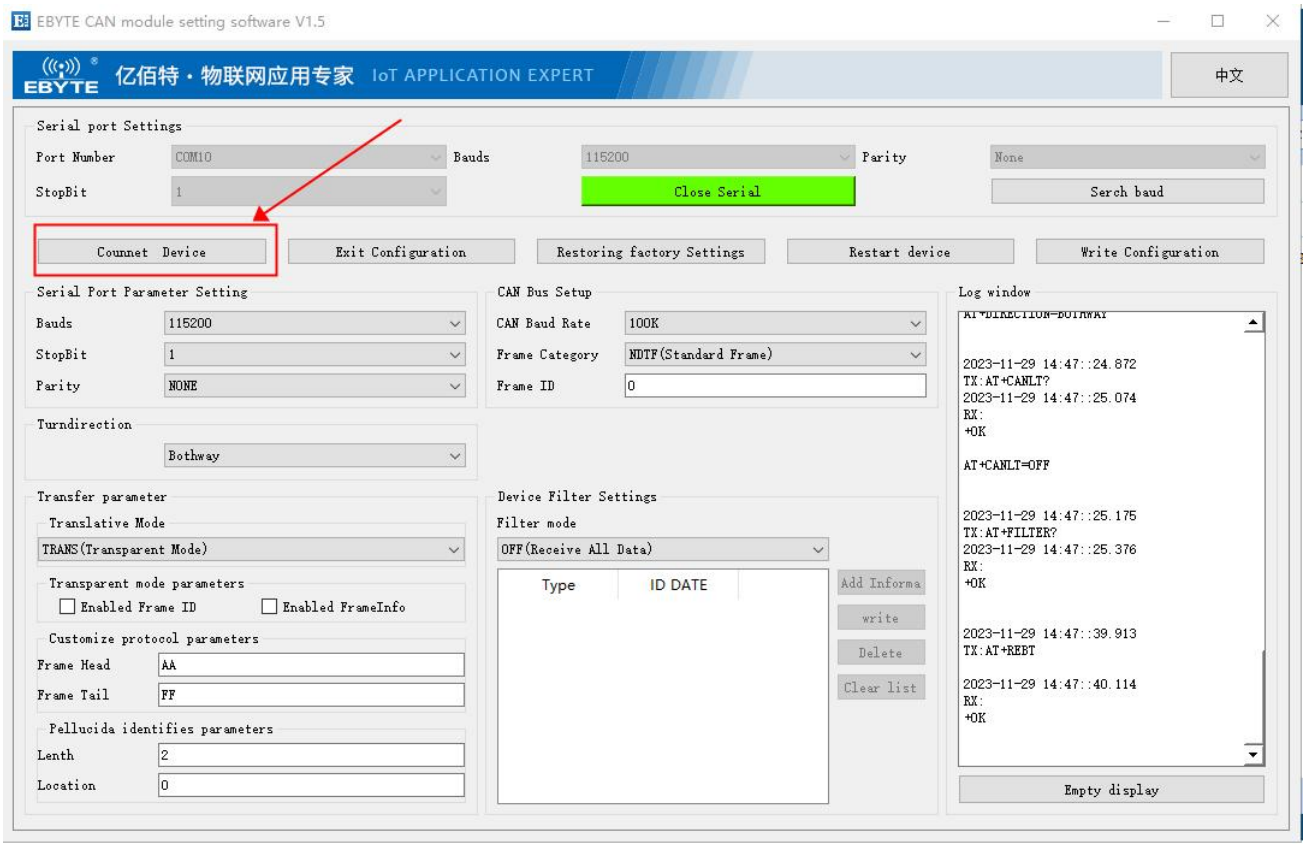
1. To change the device parameters, open ECAN-101 to configure the upper computer, select the default serial port parameters (baud rate 115200, 8 data bits, no parity bit, 1 stop bit), and click to open the serial port.

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



2. Click on "Connect Device" to enter configuration mode and read parameters

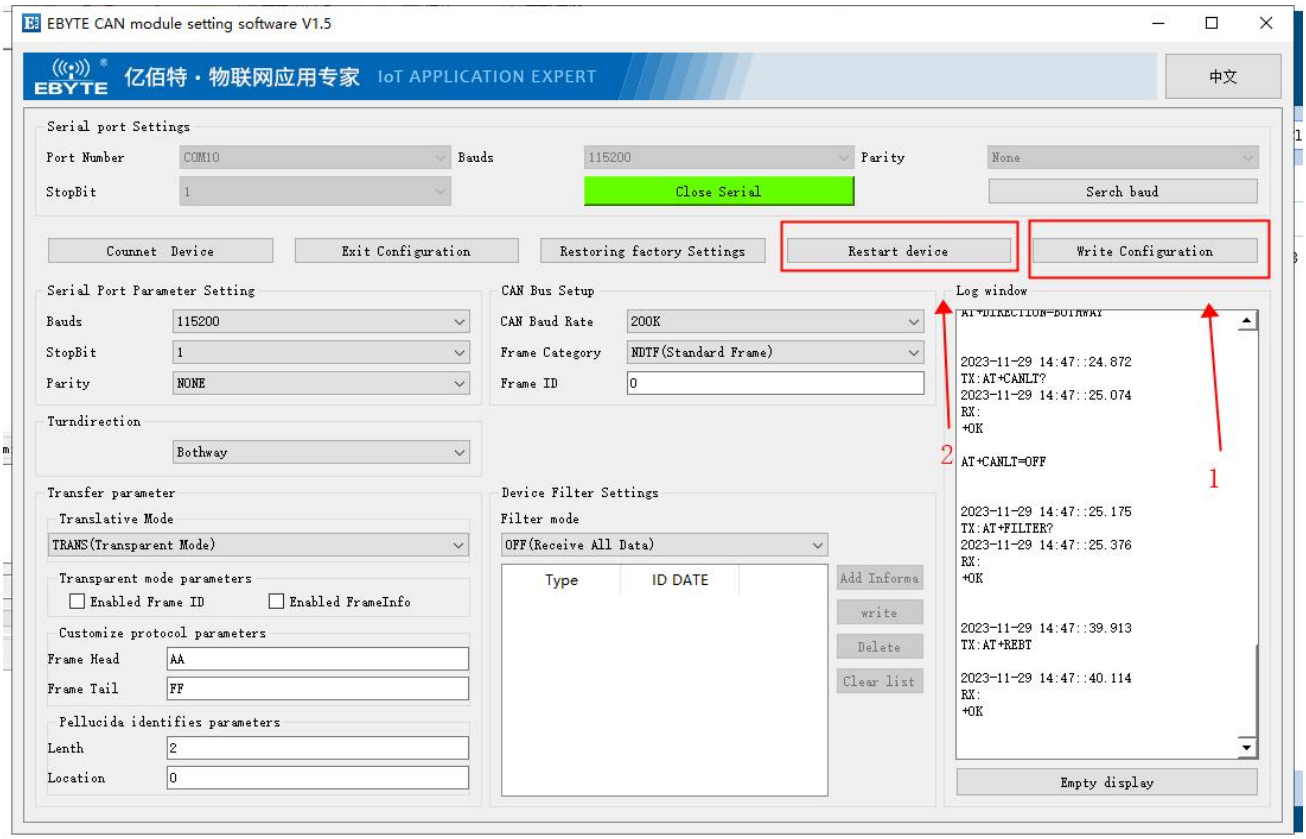


3. The device's "serial port parameter settings", "CAN bus settings", "conversion direction", "conversion mode", "device filtering settings" and other parameters can be changed according to different application scenarios. After modifying the parameters, click

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

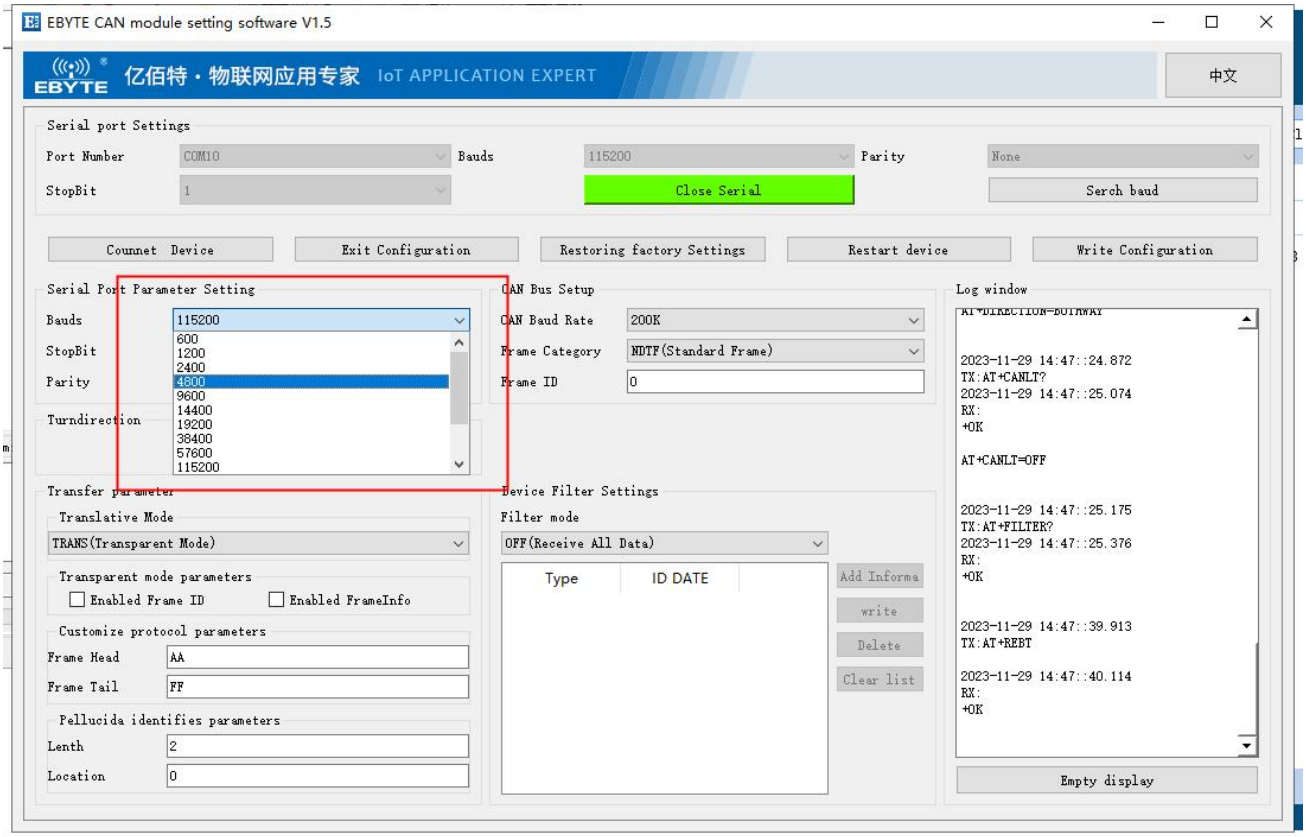
"write configuration" and "restart the device" to take effect.



Serial port parameter settings: This section is for serial port parameter settings, which are divided into "Baud rate", "Check bit", and "Stop bit". The baud rate range is: 600120024004800960144001920038400430005760 0, 76800, 115200, 128000, 230400, 256000, 460800 units: bps; Inspection bits: odd parity (ODD), even parity (EVEN), and no parity (NONE); Stop bits: 1, 2.

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

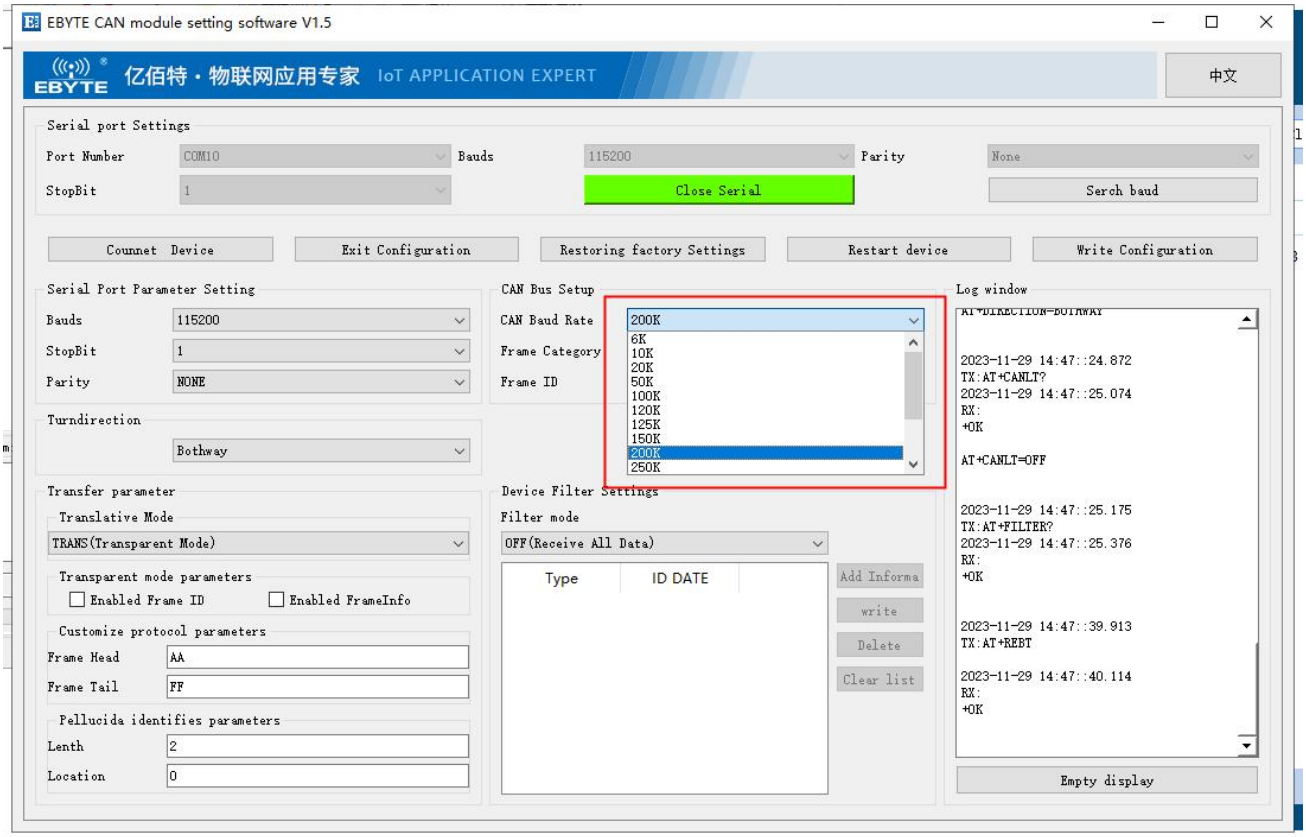
Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



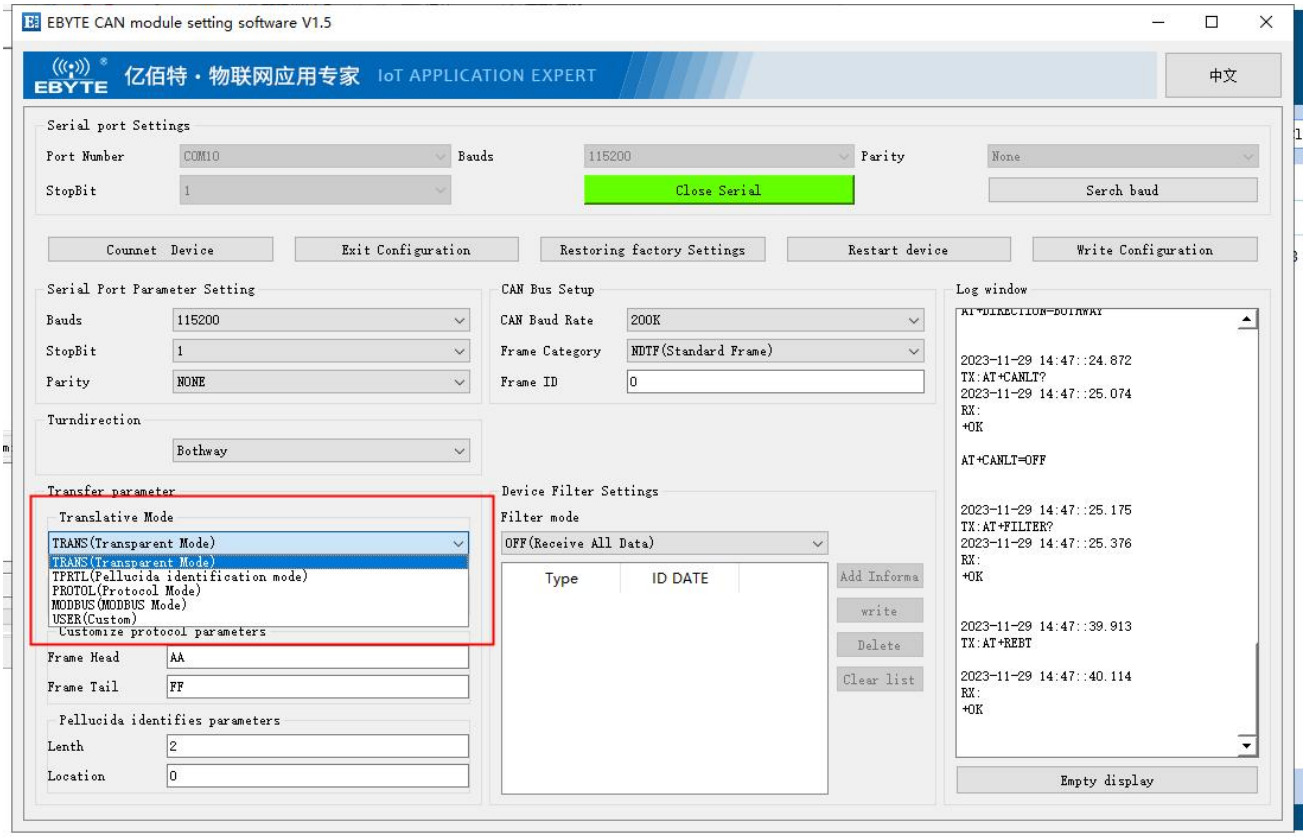
CAN bus settings: This section is the CAN section settings, divided into "CAN baud rate", "frame category", and "frame ID". The CAN baud rate range is 6K, 10K, 20K, 50K, 100K, 120K, 125K, 150K, 200K, 250K, 400K, 500K, 600K, 750K, and 1000K. The unit is bps. If the baud rate set is inconsistent with the bus baud rate of the access module, normal communication conversion will not be possible. Frame category: NDTF (standard frame), EDTF (extended frame), Frame ID: 0-7FF (standard frame), 0-1FFFFFFF (extended frame)

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



Conversion modes: There are five types: TRANS (transparent), TPRTL (transparent band identification), PROTOL (protocol mode), USER (custom protocol), MODBUS (MODBUS). The protocol rules for each mode are different, as described in Chapter 5.



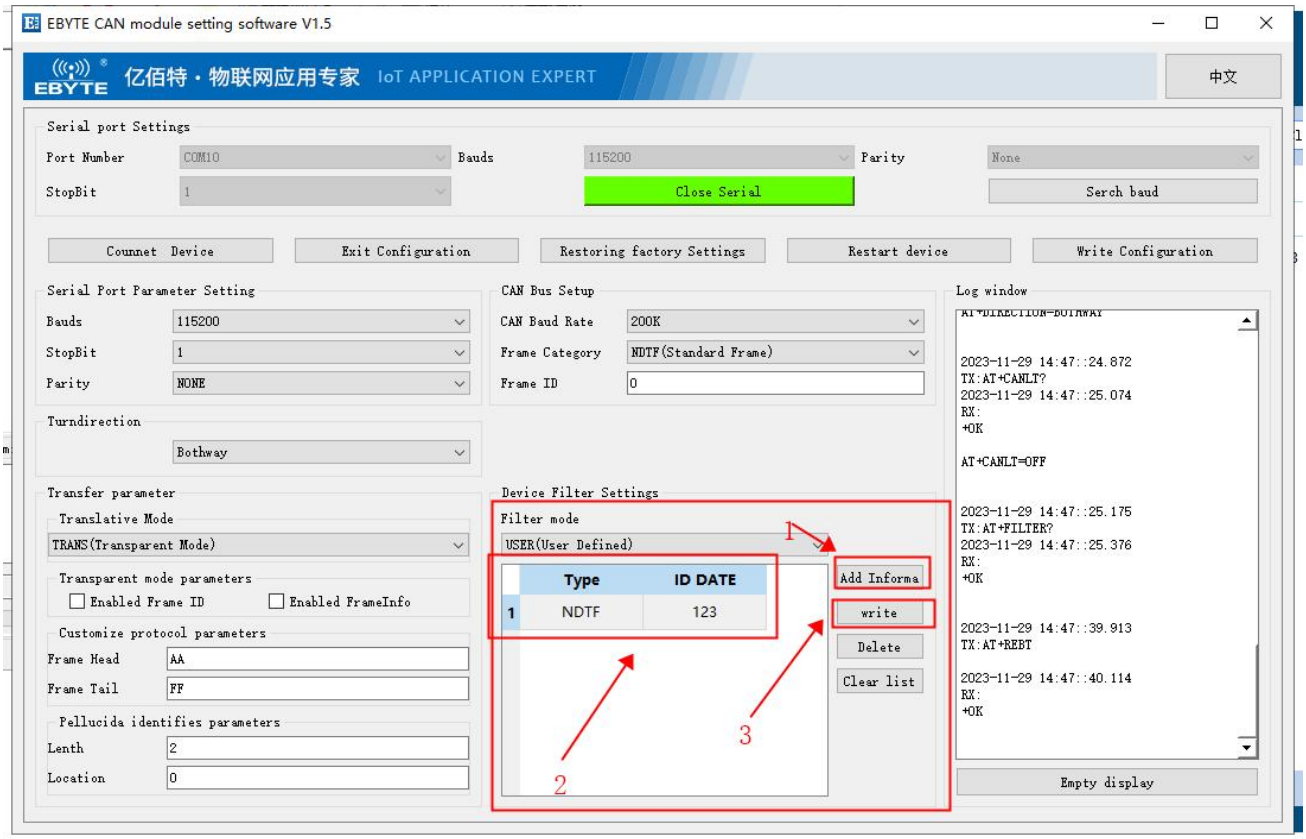
Device filtering settings: There are four filtering methods: 1. Receive all data 2. Only receive extended frames 3. Only

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

receive standard frames 4. User defined.

The first three options only require configuration selection, while the last user defined filter can add 14 sets of custom ID filters. Click "Add Information", add the desired filtering ID in the left information box, and then click "Write" to complete.

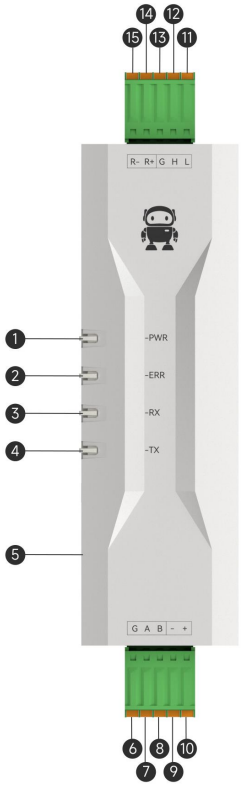


Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

4. Introduction to Hardware Parameter Design

4.1 Design Introduction

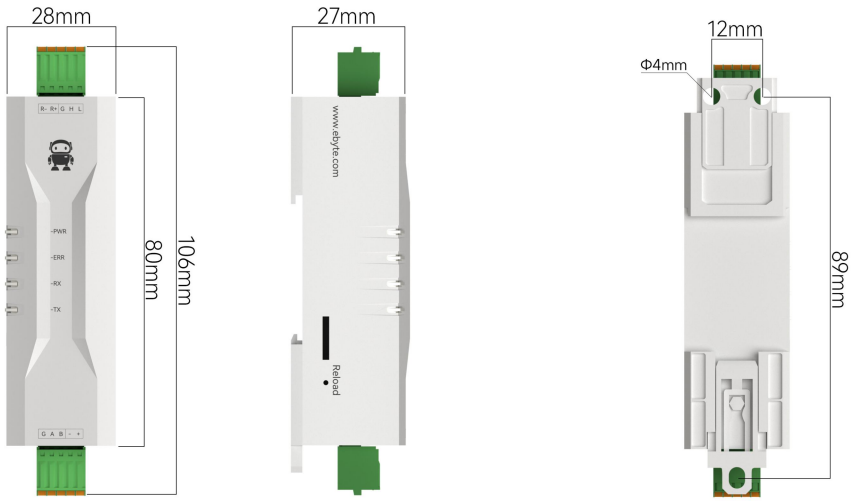


Pin number	Pin Name	Pin Usage
1	PWR	Power indicator light, always on when powered on (red)
2	ERR	Error indicator light, bus error always on (red), bus recovery off
3	RX	Data indicator light, flashing (green) when receiving data through the serial port
4	TX	Data indicator light, serial port sending data, flashing (green)
5	Restore	Restore button, press and hold for more than 5 seconds to restore factory settings
6	GND	Signal ground
7	485-A	RS485 bus data A
8	485-B	RS485 bus data B
9	V-	Negative pole of power supply, DC 8V~28V, recommended 12V or 24V power supply
10	V+	Positive pole of power supply, DC 8V~28V, recommended 12V or 24V power supply
11	CAN_ L	CAN communication interface
12	CAN_ H	CAN communication interface
13	CAN_ G	CAN grounding
14	R+	120 ohm matching resistor positive, short circuited R120- for use
15	R-	120 ohms matching resistance negative, short circuited to R120+for use

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

4.2 Dimensional drawing



5. Mode Description

During "transparent conversion" and "format conversion", a byte of frame information is used to identify some information of the CAN frame, such as type, format, length, etc. The frame information format is as follows.

Frame information description

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
FF	RTR	NO	NO	DLC3	DLC2	DLC1	DLC0

Table 1.1 Frame Information

FF: Identification of standard and extended frames, with 0 being the standard frame and 1 being the extended frame;
RTR: Identification of remote frames and data frames, with 0 being the data frame and 1 being the remote frame;
NO: Not used;
NO: Not used;
DLC3~DLC0: Identify the data length of the CAN message;

5.1 Data conversion method

The ECAN-101 device supports five data conversion methods: transparent conversion, transparent tape identification conversion, protocol conversion, MODBUS conversion, and custom protocol conversion. Supports bidirectional conversion between CAN and RS485.

Data conversion method	change direction
Transparent conversion	CAN and RS485 bidirectional conversion
Transparent tape identification conversion	CAN and RS485 bidirectional conversion
protocol conversion	CAN and RS485 bidirectional conversion
MODBUS conversion	CAN and RS485 bidirectional conversion
Custom Protocol Conversion	CAN and RS485 bidirectional conversion

5.1.1 Transparent conversion mode

Transparent conversion: The converter converts bus data from one format to another format without attaching or modifying data. This not only achieves the exchange of data formats without changing the data content, but also makes the converter "transparent" to the buses at both ends, hence it is called transparent conversion.

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

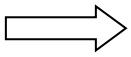
Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

The ECAN-101 device can convert valid data received from the CAN bus intact to the RS485 serial bus output. Similarly, the device can also convert the valid data received from the RS485 serial bus to the CAN bus output intact. Realize transparent conversion between RS485 and CAN.

1. Serial frame to CAN message

All data of the serial frame is sequentially filled into the data domain of the CAN message frame. Once the module detects data on the serial bus, it immediately receives and converts it. The converted CAN message frame information (frame type part) and frame ID come from the user's prior configuration, and the frame type and frame ID remain unchanged during the conversion process.

Serial frame	
0	Data 1
1	Data 2
2	Data 3
3	Data 4
4	Data 5
5	Data 6
6	Data 7
7	Data 8



CAN message	
Frame information	User Configuration
Frame ID	User Configuration
	User Configuration
Data Domain	Data 1
	Data 2
	Data 3
	Data 4
	Data 5
	Data 6
	Data 7
	Data 8

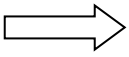
Convert serial frames into CAN messages (transparent)

Conversion example:

Convert serial frames into CAN messages (transparently).

Assuming that the configuration CAN frame information is' standard frame ', frame ID is' 0x0213', and serial frame data is from 0x01 to 0x0C, the conversion format is as follows. The frame ID of the CAN message is 0x0213 (user configured), and the frame category is standard frame (user configured). The data part in the serial frame will be converted into the CAN message without any modification.

Serial frame
0x01
0x02
0x03
0x04
0x05
0x06
0x07
0x08
0x09
0x0A
0x0B
0x0C



	CAN message 1	CAN message 2
Frame information	0x08	0x04
Frame ID	0x02	0x02
	0x13	0x13
Data Domain	0x01	0x09
	0x02	0x0A
	0x03	0x0B
	0x04	0x0C
	0x05	
	0x06	
	0x07	
	0x08	

Convert CAN serial frames into messages (transparent)

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2. CAN message to serial frame

During conversion, all data in the CAN message data domain is sequentially converted into serial frames. If "Enable Frame Information" is checked during configuration, the module will directly fill the "Frame Information" byte of the CAN message into the serial frame. If 'Enable Frame ID' is checked, all 'Frame ID' bytes of the CAN message will also be filled in to the serial frame.

Note: If you want to receive CAN frame information or frame ID on the serial interface, you need to enable the corresponding function. Only then can the corresponding information be received.

CAN message			Serial frame	
Frame information	User Configuration		Frame information	User Configuration
Frame ID	User Configuration		Frame ID	User Configuration
	User Configuration			User Configuration
Data Domain	Data 1		0	Data 1
	Data 2		1	Data 2
	Data 3		2	Data 3
	Data 4		3	Data 4
	Data 5		4	Data 5
	Data 6		5	Data 6
	Data 7		6	Data 7
	Data 8		7	Data 8

Converting CAN messages into serial frames (transparent mode)

Conversion example:

The CAN message "frame information" enabled and "frame ID" enabled in this example configuration. Frame ID 1: 0x123, Frame Category: Standard Frame, Frame Type: Data Frame. Conversion direction: bidirectional. The data is 0x12, 0x34, 0x56, 0x78, 0xAB, 0xCD, 0xEF, 0xFF. The data before and after conversion is as follows:

CAN message			Serial frame
Frame information	0x08		0x08
Frame ID	0x01		0x01
	0x23		0x23
Data Domain	0x12		0x12
	0x34		0x34
	0x56		0x56
	0x78		0x78
	0xAB		0xAB
	0xCD		0xCD
	0xEF		0xEF
	0xFF		0xFF

Converting CAN messages into serial frames (transparent mode)

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.1.2 Transparent tape identification mode

Transparent band identification conversion is a special use of transparent conversion. In the serial frame, the ID information of the CAN message can be sent with different IDs as needed. It is beneficial for users to establish their own network more conveniently through modules and use customized application protocols. This method automatically converts the ID information in the serial frame into the frame ID of the CAN bus. As long as the module is told in the configuration that the ID information is at the starting position and length of the serial frame, the module extracts this frame ID during conversion and fills it in the frame ID field of the CAN message, which is used as the ID of the CAN message when forwarding the serial frame. When converting CAN messages into serial frames, the ID of the CAN message is also converted to the corresponding position in the serial frame.

Conversion method:

1. **Serial frame to CAN message**

The starting address and length of the "frame ID" of the CAN message carried in the serial frame can be set by configuration. The starting address range is 0-7, and the length range is 1-2 (standard frame) or 1-4 (extended frame). During conversion, all CAN message "frame IDs" in the serial frame are converted into the frame ID domain of the CAN message according to the previous configuration. (If the number of frame IDs carried is less than the number of frame IDs in the CAN message, the high byte of the frame ID in the CAN message is filled with 0.) Other data is sequentially converted. If a CAN message does not fully convert the serial frame data, the same ID is still used as the frame ID of the CAN message to continue conversion until the serial frame conversion is completed.

Note: If the ID length is greater than 2, the frame category sent by the device will be set to extended frames. At this point, the frame ID and frame category configured by the user are invalid, determined by the data inside the serial frame. The frame ID range of standard frames is: 0x000-0x7FF, represented as frame ID1 and frame ID0 respectively, where frame ID1 is a high byte, and the frame ID range of extended frames is: 0x00000000-0x1FFFFFFF, represented as frame ID3, frame ID2, frame ID1, and frame ID0 respectively, where frame ID3 is a high byte.

Serial frame	
0	Data 1 (ID1)
1	Data 2 (ID0)
2	Data 3
3	Data 4
4	Data 5
5	Data 6
6	Data 7
7	Data 8
…	…
N-2	Data N-1
N-1	Data N



	CAN message	CAN message x
Frame information	User Configuration	User Configuration
Frame ID	Data 1 (ID1)	Data 1 (ID1)
	Data 2 (ID0)	Data 2 (ID0)
Data Domain	Data 3	data
	Data 4	Data N-1
	Data 5	Data N
	Data 6	
	Data 7	
	Data 8	
	Data 9	
	Data 10	

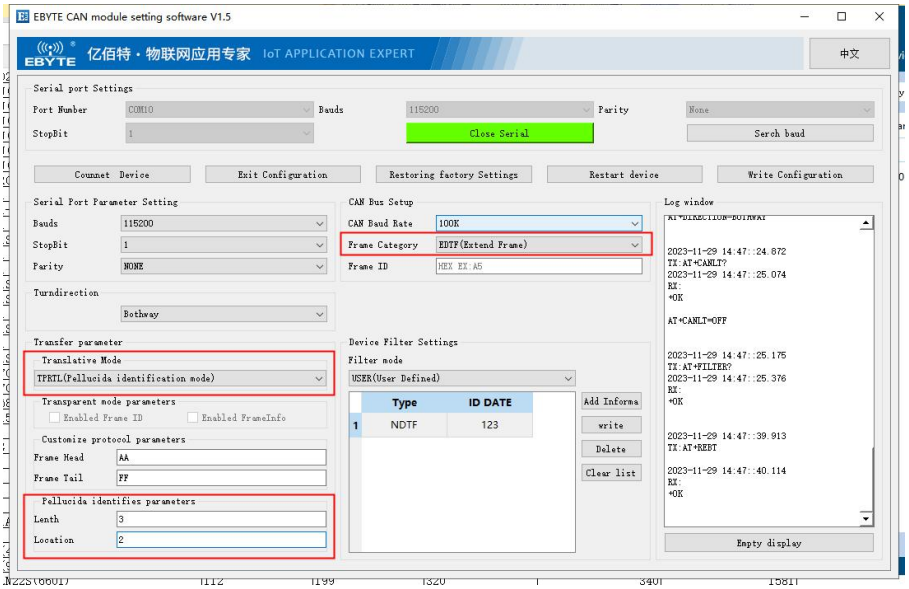
Convert serial frames into CAN messages (transparent with identification)

Conversion example:

Serial frame to CAN message (transparent tape identification).

The CAN configuration parameters for this example configuration. Conversion mode: Transparent tape identification conversion, starting address 2, length 3. Frame type: Extended frame, Frame ID: No configuration required, Conversion direction: Bidirectional. The upper computer configuration interface is as follows:

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



The schematic diagram of the data before and after conversion is as follows:

Serial frame
01
02
03
04
05
06
07
08
09
0A
0B
0C
0D
0E
0F
10

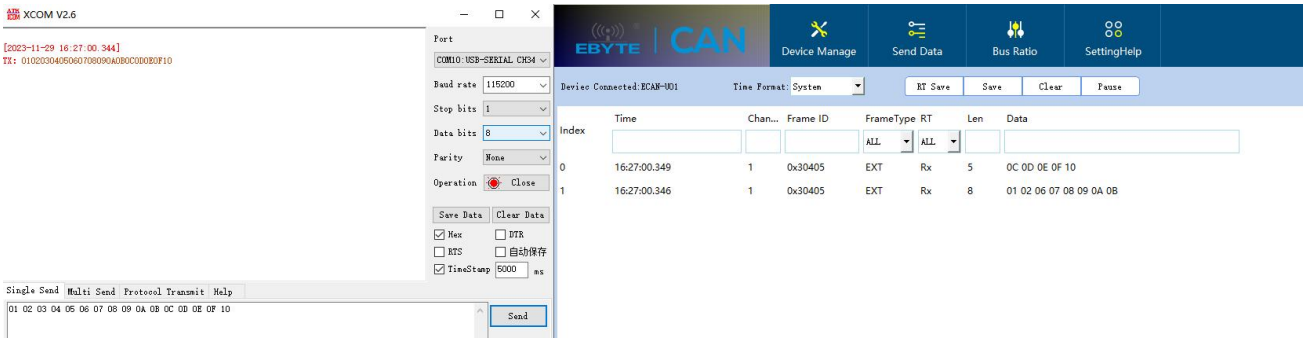
	CAN message 2		CAN message 2
Frame information	88	Frame information	85
Frame ID	00	Frame ID	00
	03		03
	04		04
	05		05
Data Domain	01	Data Domain	0C
	02		0D
	06		0E
	07		0F
	08		10
	09		
	0A		
	0B		

Example of serial frame to CAN message conversion (transparent transmission with identification conversion)

The serial debugging assistant sends a serial frame: 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10, and then receives the converted CAN frame on the CAN debugger: frame ID 0x00030405, the first frame data field is 01 02 06 07 08 09 0A 0B, and the second frame data field is 0C 0D 0E 0F 10. From this, it can be seen that the device has already extracted the second bit in the serial frame, with the last three bits 03 04 05 as frame IDs, The remaining data is repackaged as CAN frame data domain.

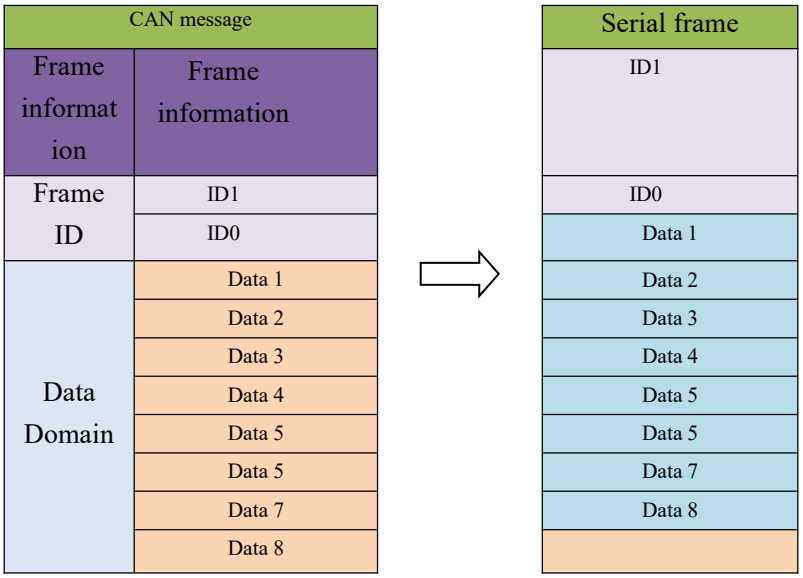
Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



2. CAN message to serial frame

For CAN messages, once a frame is received, it is immediately forwarded, and each time it is forwarded, the ID in the received CAN message is converted accordingly based on the pre configured position and length of the CAN frame ID in the serial frame. Forward other data in order. It is worth noting that both serial frames and CAN messages should have a frame format (standard frame or extended frame) that meets the pre configured frame format requirements when applied, otherwise communication may not be successful.



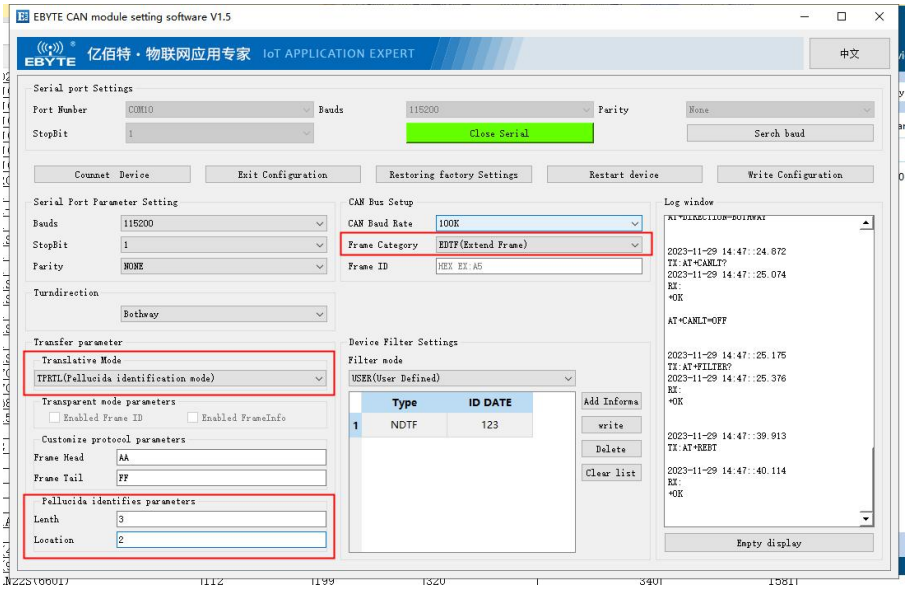
Converting CAN messages into serial frames

Conversion example:

The CAN configuration parameters for this example configuration. Conversion mode: Transparent tape identification conversion, starting address 2, length 3. Frame type: extended frame, frame type: data frame. Conversion direction: bidirectional. The upper computer configuration interface is as follows:

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



Send identifier: 0x0000123, then the data before and after conversion is as follows.

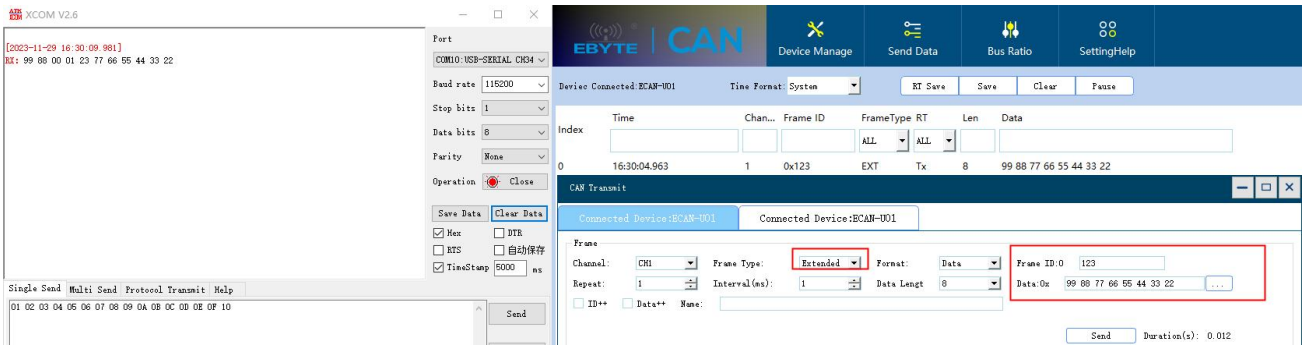
CAN message	
Frame information	88
Frame ID	00
	00
	01
	23
Data Domain	99
	88
	77
	66
	55
	44
	33
	22

⇨

Serial frame
99
88
00
01
23
77
66
55
44
33
22

Example of converting CAN messages to serial frames (transparent band information conversion)

From ECAN-U01 (CAN debugger) sending data: frame ID: 0x00000123, data domain: 99 88 77 66 55 44 33 22, serial port debugging assistant receiving serial frame: 99 88 00 01 23 77 66 55 44 33 22, it can be seen that when converting CAN frames into serial frames, data is filled in according to our set rules, and the CAN frame ID is filled in the last three digits starting from the second digit of the data domain.



5.1.3 Protocol mode

The fixed 13 bytes of CAN format conversion represent a CAN frame data, and the 13 byte content includes CAN frame information+frame ID+frame data. In this conversion mode, the CANID set is invalid because the identifier (frame ID) sent at this time is filled with frame ID data from the format serial frame mentioned above. The configured frame type is also invalid, and the frame type is determined by the frame information in the format serial frame. The format is as follows:

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

CAN fixed format serial frame (13 bytes)		
Frame information	Frame ID	Frame data
1 Byte	4Byte	8Byte

The frame information is shown in the table below:

The length of the frame ID is 4 bytes, the standard frame significant bit is 11 bits, and the extended frame significant bit is 29 bits.

Extended frame ID number 0x12345678			
0x12	0x34	0x56	0x78

Standard frame ID number 0x3FF			
0x00	0x00	0x03	0xFF

1. Serial frame to CAN message

During the process of converting serial frames to CAN messages, in a fixed byte (13 byte) aligned serial data frame, if the data format of a certain segment of fixed bytes is not standardized, the fixed byte length will not be converted, and the subsequent data will be converted. If some CAN messages are missing after conversion, please check if the fixed byte length serial data format of the corresponding message does not comply with the standard format.

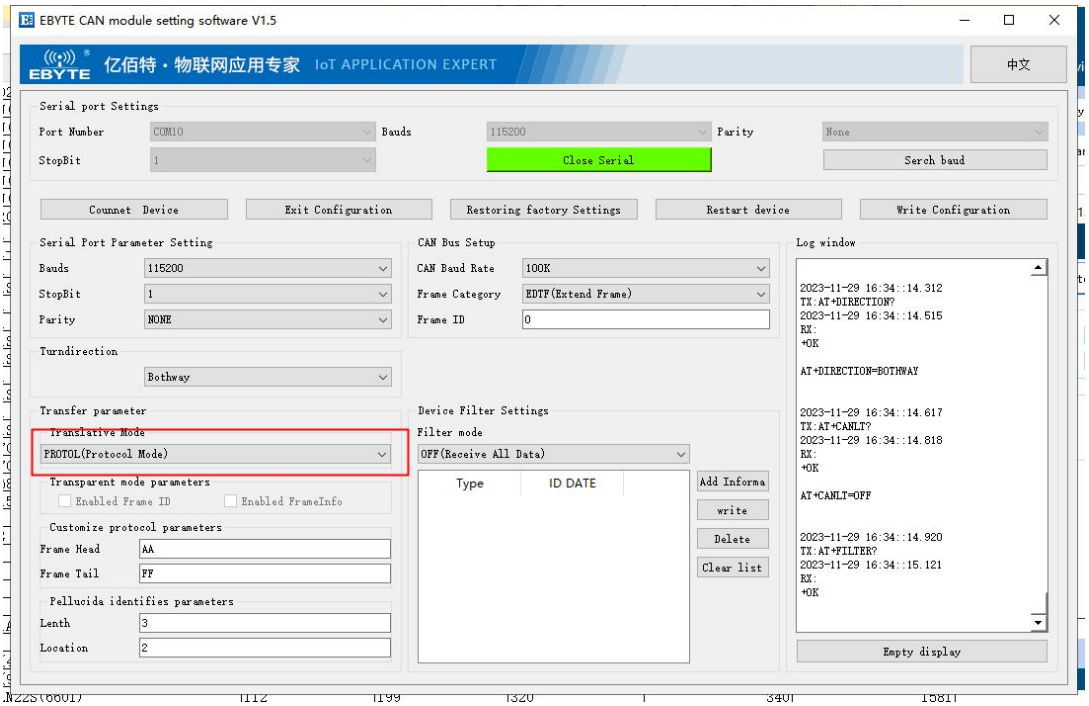
When converting frame data to CAN format, the length is fixed at 8 bytes. The effective length is determined by the values of DLC3~DLC0. When the effective data is insufficient for a fixed length, it is necessary to supplement 0 to a fixed length.

In this mode, it is important to strictly follow the fixed byte format serial data format in order to successfully convert. The CAN mode conversion can refer to the example (CAN format conversion standard frame example). When converting, it is first necessary to ensure that the frame information is correct and the data length representation is correct, otherwise the conversion will not be carried out.

Conversion example:

Serial frame to CAN message (protocol mode).

The CAN configuration parameters for this example configuration. Conversion mode: Protocol mode; Frame category: extended frames; Conversion direction: bidirectional. Frame ID: No configuration required; The configuration of the upper computer is as follows:



The schematic diagram of the data before and after conversion is as follows:

Serial frame
88

⇨

CAN message	
Frame information	88

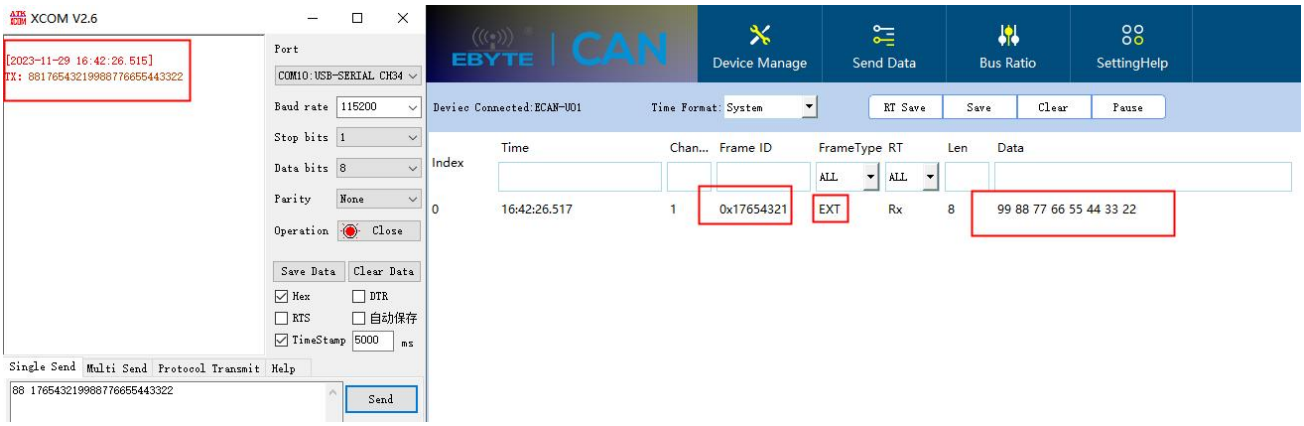
Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

17		Frame ID	17
65			65
43			43
21			21
99		Data Domain	99
88			88
77			77
66			66
55			55
44			44
33			33
22			22

Serial frame to CAN message (protocol mode)

The serial debugging assistant sends data: 88 17 65 43 21 99 88 77 66 55 44 33 22, and receives the converted CAN frame on the CAN debugger: an extended frame with frame ID 0x17654321 and data domain: 99 88 77 66 55 44 33 22. From here, it can be seen that the first bit of data "88" in the serial frame indicates the frame type and frame data domain length: extended frame, data domain length 8.



5.1.4 Modbus mode

Modbus protocol is a standard application layer protocol widely used in various industrial control situations. This protocol is open, has strong real-time performance, and good communication verification mechanism, making it very suitable for occasions with high communication reliability requirements. The module uses the standard Modbus RTU protocol format on the serial port side, so the module not only supports users to use the Modbus RTU protocol, but also can directly interface with other devices that support the Modbus RTU protocol. On the CAN side, a simple and easy-to-use segmented communication format has been developed to achieve Modbus communication. A method for segmenting and reassembling information with a length greater than the maximum data length of a CAN message is used. "Data 1" is used to segment and identify data. The transmitted Modbus protocol content can start from "Data 2" bytes. If the protocol content is greater than 7 bytes, the remaining protocol content will continue to be converted according to this segmented format, Until the conversion is completed. When there is no other data on the CAN bus, frame filters may not be set. To complete the communication. When there is other data on the bus, a filter needs to be set. Distinguish the source of data received by the device. According to this method. It is possible to achieve communication between multiple hosts on a bus. The data transmitted on the CAN bus does not require CRC validation. The data validation on the CAN bus has a more comprehensive validation method.In this mode, the device supports Modbus authentication and forwarding, rather than the master or slave of Modbus. Users can communicate according to the Modbus protocol.

Segmented transmission protocol:

Bit number	7	6	5	4	3	2	1	0
Frame information	FF	RTR	EDL	BRS	DLC (Data Length)			
Frame ID3	X	X	X	ID.28-ID.24				
Frame ID2	ID.23-ID.16							
Frame ID1	ID.15-ID.8							

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Frame ID0	ID.7-ID.0		
Data 1	Segmentation markers	Segment Type	Segmented counter
Data 2	DATE		
Data 3	DATE		
Data 4	DATE		
Data 5	DATE		
Data 6	DATE		
Data 7	DATE		
Data 8	DATE		

The method of segmenting and reassembling information with a length greater than the maximum data length of a CAN message. When using a CAN message, "Data 1" is used to segment and identify the data. The segmented message format is as follows, and the Modbus protocol content transmitted can start from "Data 2" bytes. If the protocol content is greater than 7 bytes, the remaining protocol content will continue to be converted according to this segmented format until the conversion is completed.

Segmentation markers	Segment Type		Segmented counter				
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

- Segmentation mark: indicates whether the message is a segmented message. A bit of 0 indicates a separate message, and a bit of 1 indicates a frame belonging to the segmented message.
- Segment type: Indicates whether it is the first, middle, or last segment.
-

Bit value	meaning	explain
0	First segment	If the segment counter contains a value of 0, then this is the first segment in the segment series
1	Middle segment	Indicates that this is an intermediate segment
2	Final segmentation	Flag last segment

- Segment counter: The flag of each segment, indicating the sequence number of that segment in the entire message. If it is the segment, the value of the counter is. This enables verification of whether any segments have been lost upon receipt. A total of 5Bit is used, with a range of 0-31.

1. Serial frame to CAN message

The serial interface adopts the standard Modbus RTU protocol, so user frames can comply with this protocol. If the transmitted frame does not comply with the Modbus RTU format, the module will discard the received frame and not convert it.

2. Convert CAN message to serial frame

For the Modbus protocol data of the CAN bus, there is no need to perform cyclic redundancy check (CRC16). The module receives it according to the segmented protocol, automatically adds cyclic redundancy check (CRC16) after receiving a frame for analysis, converts it into a Modbus RTU frame, and sends it to the serial bus. If the received data does not comply with the segmentation protocol, the set of data will be discarded and not converted.

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

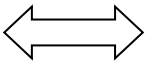
Modbus RTU frame
Address Domain
Function code
Data Domain
CRC Domain



CAN message	CAN message 1	CAN message x
Frame information	Frame information	Frame information
Frame ID3	0x00	0x00
Frame ID2	0x00	0x00
Frame ID1	0x00	0x00
Frame ID0	Frame ID (user set)	Frame ID (user set)
Data 1	[Use of segmented protocols]	[Use of segmented protocols]
Data 2	Address Domain	Data Domain
Data 3	Data Domain	
Data 4		
Data 5		
Data 6		
Data 7		
Data 8		

Conversion Example 1:

Modbus RTU frame	Serial frame
Address Domain	55
Function code	0F
Data Domain	00
	00
	00
	04
	02
	00
	00
CRC Domain	29
	13



CAN message	CAN message 1	CAN message 2
Frame information	08	02
Frame ID3	00	00
Frame ID3	00	00
Frame ID1	00	00
Frame ID0	Frame ID	Frame ID
Data 1	81	C2
Data 2	55	00
Data 3	0F	00
Data 4	00	
Data 5	00	
Data 6	00	
Data 7	04	
Data 8	02	

Conversion Example 2:

Serial port side Modbus RTU data (hexadecimal):

01 03 14 00 0A 00 00 00 00 00 14 00 00 00 00 00 17 00 2C 00 37 00 C8 4E 35

The first byte 01 is the address code of the Modbus RTU, which is converted to ID.7-ID.0 of CAN: The last two bytes (4E 35) are CRC checksums for Modbus RTUs, and are discarded without conversion.

The final converted CAN data message is as follows:

Frame 1 CAN message: 81 01 03 14 00 0A 00 00

Frame 2 CAN message: a2 00 00 00 14 00 00 00

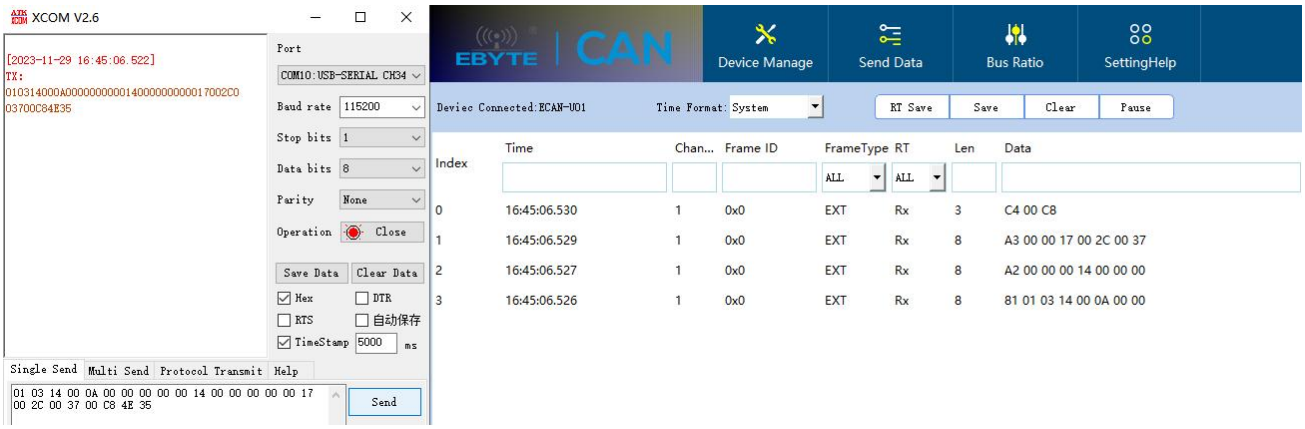
Frame 3 CAN message: a3 00 00 17 00 2C 00 37

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Frame 4 CAN message: The frame type (standard frame or extended frame) of c4 00 c8 CAN message is set through configuration software;

The first data of each CAN message is filled with segmented information (81, a2, a3, and c4), which is not converted into Modbus RTU frames and is only used as confirmation control information for the message.



5.1.5 Custom Protocol Mode

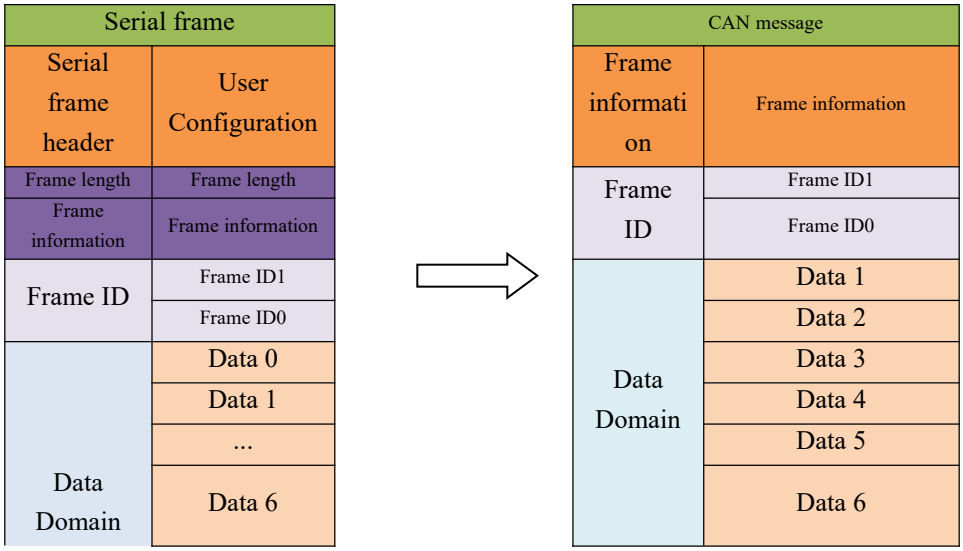
It must be a complete serial frame format that complies with custom protocol regulations, and must include all the contents of the serial frame in the user configured mode. If there is any error in the byte content except for the data domain, this frame will not be successfully sent. The content of a serial frame includes: frame header, frame length, frame information, frame ID, data domain, and frame tail.

Note: The frame ID and frame category configured by the user in this mode are invalid, and the data will be forwarded based on the format inside the serial frame.

1. Serial frame to CAN message

The serial frame format must comply with the specified frame format, as the CAN frame format is based on messages, the serial frame format is based on byte transmission. Therefore, in order to facilitate the use of CAN-bus by users, the serial frame format is moved closer to the CAN frame format, and the start and end of a frame are specified in the serial frame, which is the "frame header" and "frame tail" in the AT command. Users can configure it themselves. Frame length refers to the length from the beginning of frame information to the end of the last data, excluding the serial frame tail. Frame information is divided into extended frames and standard frames. The fixed representation of standard frames is 0x00, while the fixed representation of extended frames is 0x80. Unlike transparent conversion and transparent band identification conversion, in custom protocol conversion, the frame information content remains fixed regardless of the length of data contained in each frame data domain. When the frame type is a standard frame (0x00), the last two bytes of the frame type represent the frame ID, with the high bit leading; When the frame information is an extended frame (0x80), the last 4 bytes of the frame type represent the frame ID, with the high bit in front.

Note: In custom protocol conversion, regardless of the length of data contained in each frame data field, the frame information content is fixed and unchanged. Fixed to standard frame (0x00) or extended frame (0x80). The frame ID needs to match the ID range, otherwise the ID may be incorrect.



Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Data 7			Data 7
	Data 8			Data 8
Serial frame tail	User Configuration			

2. Convert CAN message to serial frame

When a CAN bus message receives a frame, it is forwarded to another frame. The module will convert the data in the CAN message data domain in sequence, and add data such as frame header, frame length, and frame information to the serial frame. In fact, it is the reverse form of converting a serial frame to a CAN message.

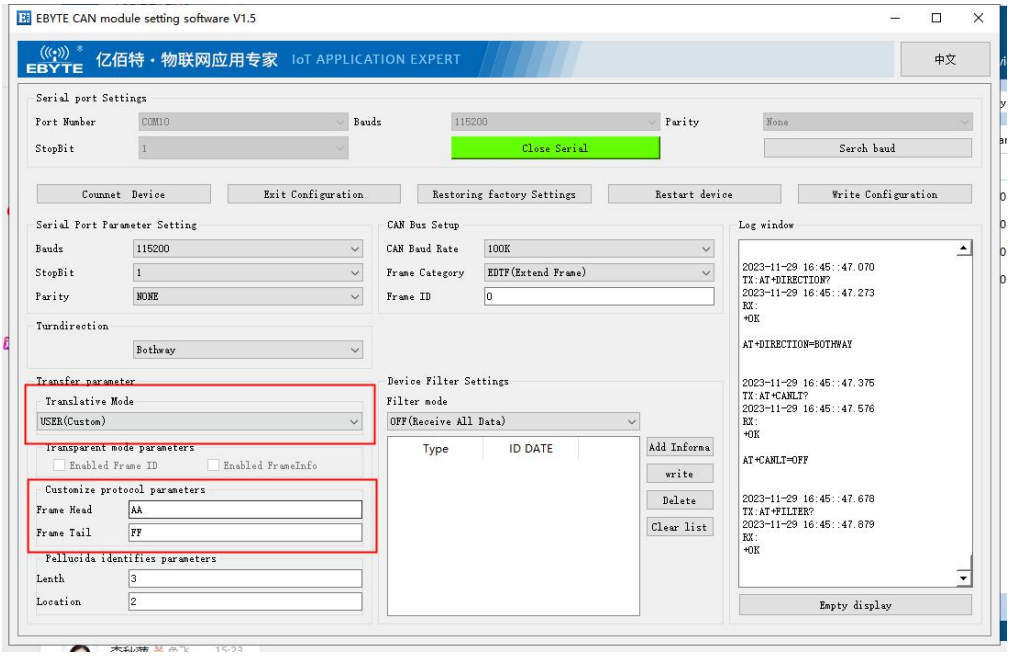
CAN message			Serial frame	
Frame information	Frame information		Serial frame header	User Configuration
Frame ID	Frame ID1		Frame length	Frame length
	Frame ID0		Frame information	Frame information
Data Domain	Data 1		Frame ID	Frame ID1
	Data 2			Frame ID0
	Data 3		Data Domain	Data 1
	Data 4			Data 2
	Data 5			...
	Data 6			Data 7
	Data 7			Data 8
	Data 8		Serial frame tail	User Configuration

Converting CAN messages into serial frames

Conversion example:

Serial frame to CAN **message (custom** protocol).

The CAN configuration parameters for this example configuration. Conversion mode: **custom protocol,** frame header AA, frame footer FF, conversion direction: bidirectional. Frame ID: No configuration required, frame category: No configuration required, upper computer configuration is as follows:

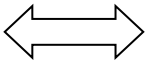


Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

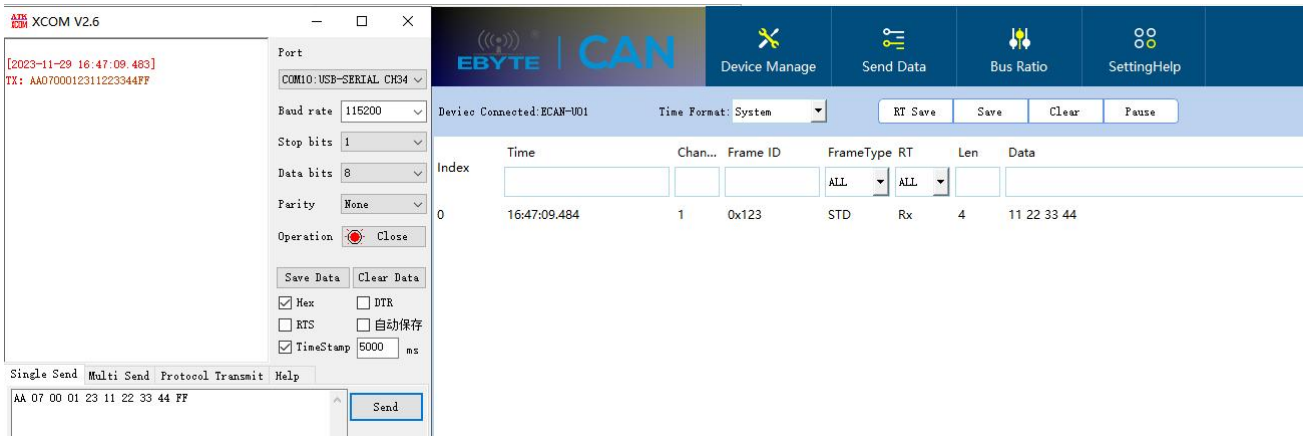
The data before and after conversion is as follows:

Serial frame	
Serial frame header	AA
Frame length	07
Frame information	00
Frame ID	01
	23
Data Domain	11
	22
	33
	44
Serial frame tail	FF



CAN message	
Frame information	00
Frame ID	01
	23
Data Domain	11
	22
	33
	44

Because the frame header is set to AA and the frame tail is set to FF, to send standard frame data with a data domain of 11 22 33 44 and a frame ID of 0x123, the serial debugging assistant will send it according to the protocol: AA 07 00 01 23 11 22 33 44 FF. The data converted from the device CAN end is displayed on the CANT debugger (ECAN-U01) below as the standard frame data with a data domain of 11 22 33 44 and a frame ID of 0x123. Similarly, the conversion of CAN frames to serial frames is also encapsulated according to this logic.



6. AT instruction

1. Enter AT command mode: Send++through the serial port, send AT again within 3 seconds, and the device will display AT MODE to enter AT command mode.

2. Unless otherwise specified, all subsequent AT instruction operations need to add "/r/n".

3. All examples were conducted with the command echo function turned off.

4. After setting the parameters, it is necessary to restart the device to ensure that the set parameters take effect.

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Error code table:

Error code	explain
-1	Invalid command format
-2	Invalid command
-3	Not defined yet
-4	Invalid parameter
-5	Not defined yet

Default parameters

Parameter category	Parameter Name	Parameter value	Related instructions
Serial port	Baud rate	115200	AT+UART
	Digits	8	
	Stop bit	1	
	Parity check	无	

6.1 Enter AT command

instructions	AT
function	Entering AT command mode
send	AT
return	<CR><LF>+OK<CR><LF>

[Example]

Send:+++//No line breaks

Send: AT//No line breaks

Response:<CR><LF>AT MODE<CR><LF>

6.2 Exit AT command

instructions	EXAT
function	Exit AT instruction mode
set up	AT+EXAT<CR><LF>
return	<CR><LF>+OK<CR><LF>

[Example]

Sending: AT+EXAT/r/n

Response:<CR><LF>+OK<CR><LF>

6.3 Query version

instructions	VER?
function	Query firmware version
query	AT+VER< CR><LF>
return	<CR><LF>VER=x.x<CR><LF>
Remarks	x. X version number

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

[Example]

Send: AT+VER/ R/n

Response:<CR><LF>VER=x.x<CR><LF>

6.4 Restore default parameters

instructions	RESTORE
function	Restore the default parameters of the device (factory parameters)
set up	AT+RESTORE<CR><LF>
return	<CR><LF>+OK<CR><LF>
Remarks	The parameter takes effect when the device needs to be restarted

[Example]

Send: AT+RESTORE/r/n

Response:<CR><LF>+OK<CR><LF>

6.5 Echo Settings

instructions	E
function	User Command Echo Settings/Queries
set up	AT+E=ON<CR><LF><CR><LF>
return	<CR><LF>+OK<CR><LF>
Remarks	ON OFF

[Example]

Settings:

Send: AT+E=OFF/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+E/ R/n

Response:<CR><LF>+OK<CR><LF>

6.6 Serial port parameters

instructions	UART
function	Set parameters for module serial communication
set up	AT+UART=baud, date, stop, party, flowcontrol
return	<CR><LF>+OK=<snString><CR><LF>
query	AT+UART?
parameter	Baud (Serial Port Baud Rate): 600120024004800960144001920038400430057600, 76800, 115200, 128000, 230400, 256000, 460800 Unit: bps Date: 8 Stop: 1,2 Party: NONE, EVEN, ODD

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Flowcontrol: NFC (no flow control), FC (flow control),

[Example]

Settings:

Sending: AT+UART=115200,8,1, EVEN, NFC/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+UART/ R/n

Response:<CR><LF>+OK<CR><LF>AT+UART=115200,8,1, EVEN, NFC<CR><LF>

6.7 Setting/querying CAN information

instructions	CAN
function	Set CAN interface communication parameters
set up	AT+CAN=baud, id, mode<CR><LF>
return	<CR><LF>+OK<CR><LF>
query	AT+CAN?
parameter	Baud (CAN Baud Rate): 6K, 10K, 20K, 50K, 100K, 120K, 125K, 150K, 200K, 250K, 400K, 500K, 600K, 750K, 1000K Unit: bps ID (frame ID): 0-7FF (standard frame), 0-1FFFFFFFFF (extended frame) Mode: (frame category): NDTF (standard frame), EDTF (extended frame)

[Example]

Settings:

Sending: AT+CAN=100,70, NDTF/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+CAN/ R/n

Response:<CR><LF>+OK<CR><LF>AT+CAN=100,70, NDTF<CR><LF>

6.8 Set/Query Module Conversion Method

instructions	MODE
function	Set/Query Module Conversion Method
set up	AT+MODE=mode<CR><LF>
return	<CR><LF>+OK<CR><LF>
query	AT+MODE?
parameter	Mode: TRANS (transparent), TPRTL (transparent tape identification), PROTOL (protocol mode), USER (custom protocol), MODBUS (MODBUS),

[Example]

Settings:

Send: AT+CANLT=ETF/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+CANLT/ R/n

Response:<CR><LF>+OK<CR><LF>AT+CANLT=ETF<CR><LF>

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

6.9 Set/query the filtering method of the CAN bus

instructions	CANLT
function	Set/query the filtering method of the CAN bus
set up	AT+CANLT=mode<CR><LF>
return	<CR><LF>+OK<CR><LF>
query	AT+CANLT?
parameter	Mode (filtering mode): OFF (receive all functions), ETF (only receive extended frames), NTF (only receive standard frames), USER (custom)

[Example]

Settings:

Send: AT+MODE=MODBUS/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+MODE/ R/n

Response:<CR><LF>+OK<CR><LF>AT+MODE=MODBUS<CR><LF>

6.10 Set/query frame header and footer data

instructions	UDMHT
function	Set/query frame header and footer data in custom mode
set up	AT+UDMHT=head, tail<CR><LF>
return	<CR><LF>+OK<CR><LF>
query	AT+UDMHT?
parameter	Head (frame header data), tail (frame tail data). Data range 0~0xFF

[Example]

Settings: Set the frame header data to FF and the frame footer data to 55

Sending: AT+UDMHT=FF, 55/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+UDMHT/ R/n

Response:<CR><LF>+OK<CR><LF>AT+UDMHT=FF, 55<CR><LF>

6.11 Set/Query Identity Parameters

instructions	RANDOM
function	Set/Query Query Identification Parameters
set up	AT+RANDOM=idLength, idLocation<CR><LF>
return	<CR><LF>+OK<CR><LF>

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

query	AT+RANDOM?
parameter	IdLength (frame header ID length), idLocation (frame ID position). Data range: Length range 0-4 Position 0-7

[Example]

Settings: Set frame ID length 4, position 2

Sending: AT+RANDOM=4,2/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+RANDOM/ R/n

Response:<CR><LF>+OK<CR><LF>AT+RANDOM=4,2<CR><LF>

6.12 Set/Query Identity Parameters

instructions	MSG
function	Set/Query Frame ID Frame Information Enable
set up	AT+MSG=flag_ ID, flag_ Type<CR><LF>
return	<CR><LF>+OK<CR><LF>
query	AT+MSG?
parameter	Flag_ ID (frame header data), tail (frame tail data). Data range 0~0xFF

[Example]

Settings: Enable frame ID, frame information

Sending: AT+MSG=1,1/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+MSG/ R/n

Response:<CR><LF>+OK<CR><LF>AT+MSG=1,1<CR><LF>

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

6.13 Set/query transmission direction

instructions	DIRECTION
function	Set/Query Frame ID Frame Information Enable
set up	AT+DIRECTION=parameter<CR><LF>
return	<CR><LF>+OK<CR><LF>
query	AT+DIRECTION?
parameter	Parameter (direction parameter), UART CAN (serial port to can). CAN-UART BOTHWAY

[Example]

Setting: Only convert serial port data to the can bus

Send: AT+DIRECTION=UART CAN/r/n

Response:<CR><LF>+OK<CR><LF>

Query:

Send: AT+DIRECTION/ R/n

Response:<CR><LF>+OK<CR><LF>AT+DIRECTION=UART CAN<CR><LF>

6.14 Set/Query Filter Parameters

instructions	FILTER
function	Set/Query Filter Frame Information
set up	AT+FILTER=id_ Type, date<CR><LF>
return	<CR><LF>+OK<CR><LF>
query	AT+FILTER?
parameter	Type (frame category), date (frame data) Type: NDTF represents that this command is a standard ID, and EDTF represents that this command is an extended frame ID Date: ID data.

[Example]

Settings: Set frame filtering parameters: Standard frame ID, 719

Sending: AT+LFILTER=NDTF, 719/r/n

Response:<CR><LF>+OK<CR><LF>

Query: will return all IDs that have been set

Sent to: AT+FILTER/ R/n

Response:<CR><LF>+OK<CR><LF>AT+LFILTER=NDTF, 719<CR><LF>

6.15 Delete filter parameters that have already been set

instructions	DELFILTER
function	Set/Query Filter Frame Information

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

set up	AT+DELFILTER=id_ Type, date<CR><LF>
return	<CR><LF>+OK<CR><LF>
parameter	Type (frame category), date (frame data) NDTF: Represents that this command is a standard ID, and EDTF represents that this command is an extended frame ID. Date: ID data.

[Example]

Settings: Delete filtering parameters: Standard frame 719

Sending: AT+DELFILTER=NDTF, 719/r/n

Response:<CR><LF>+OK<CR><LF>

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Revision History

version	Revision date	Revision Description	Maintainer
1.0	2023-09-28	Initial version	LM

About us

Technical support: support@cdebyte.com
Documents and RF Setting download link: www.ebyte.com
Thank you for using Ebyte products! Please contact us with any questions or suggestions: info@cdebyte.com
---
Phone: +86 028-61399028
Web: www.ebyte.com
Address: B5 Mould Park, 199# Xiqu Ave, High-tech District, Sichuan, China

Chengdu Ebyte Electronic Technology Co.,Ltd.

Copyright ©2012 – 2023, Chengdu Ebyte Electronic Technology Co., Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000