



HDLC-ETH

Serial Ethernet Converter

Rev. Dec 20, 2017



HDLC-ETH

Datasheet

Email: yacer@yacer.cn

Website: www.yacer.cn

yacer 亚册
Building Blocks of Communication

1	Overview.....	3
1.1	Introduction.....	3
1.2	Features.....	3
1.3	Applications.....	3
1.4	Technical Specifications.....	4
1.5	Order Information.....	4
2	Hardware Interface.....	5
2.1	Appearance.....	5
2.2	LED Indicators.....	5
2.3	Ethernet Interface.....	5
2.4	Serial Port.....	6
2.5	DC: power interface.....	6
3	Building Configuration Environment.....	6
4	yacer-DMS Configuration Management Software.....	7
4.1	Get the Configuration Management Software yacer-DMS.....	7
4.1.1	Official website of yacer.....	7
4.1.2	Accompanied USB disk.....	7
4.2	Run yacer-DMS.....	8
4.2.1	Unzip the file.....	8
4.2.2	Run yacer-DMS.....	9
4.3	Main Window of DMS.....	10
4.4	Statistical Report.....	10
4.4.1	Receive/transmit indication.....	10
4.4.2	Information display.....	11
4.4.3	Control Widget.....	11
4.5	Configure Device.....	12
5	Function and Configuration.....	13
5.1	Ethernet Interface Configuration.....	13
5.1.1	Device alias.....	13
5.1.2	IP address configuration.....	13
5.1.3	Default gateway.....	15
5.1.4	Announce Configuration.....	16
5.2	Serial Port Configuration.....	17
5.2.1	Select the working mode of the serial port.....	17
5.2.2	HDLC-NRZ parameter configuration.....	18
5.2.3	Bit Stream parameters.....	21
5.2.4	UART parameter configuration.....	22
5.2.5	UART-HDLC parameter configuration.....	23
5.3	UDP to Serial Conversion.....	24
5.3.1	Application model.....	24
5.3.2	Protocol Conversion.....	24
5.3.3	Parameter configuration.....	25

5.3.4	Receive multicast	25
5.4	Serial to UDP Conversion	26
5.4.1	Application model	26
5.4.2	Protocol Conversion	26
5.4.3	Parameter configuration	27
5.4.4	How does the UDP server identify the source serial	27
5.5	Serial to Serial Conversion.....	29
6	Firmware Version Upgrade	30
6.1	Start Upgrade	30
6.2	Locate Firmware Version	30
6.3	Upgrade Completed	31
6.4	Upgrade Confirmation	31
7	Device Reboot.....	32
8	PING	32

1 Overview

1.1 Introduction

The Yacer HDLC-ETH protocol converter, provides dual or quad RS-232 synchronous & asynchronous serial ports, dual 10/100M Ethernet interface to achieve the protocol conversion between the serial port and the Ethernet interface.

Small footprint, +12V power supply, industrial wide temperature.

1.2 Features

- Dual 10/100M Ethernet interface, with support for the Ethernet switching, VLAN and dual-IP functions
- Dual or quad RS-232 synchronous & asynchronous serial ports
- Full-duplex, with the baud rate up to 250 Kbps
- Support for the synchronous HDLC protocol, asynchronous UART
- 15KV ESD protection
- +12V power supply
- Industrial wide temperature



1.3 Applications

- Secondary Surveillance Radar (SSR), ADS-B data communication
- Air Traffic Management (ATM)
- Air Traffic Control (ATC)
- Synchronous & Asynchronous serial communication and conversion
- Embedded application

1.4 Technical Specifications

Serial Port	
Quantity	2, 4 x RJ-45
Working mode	Synchronous HDLC, Asynchronous UART
Interface type	RS-232 full-duplex
Encoding format	NRZ
Baud rate	≤ 250 Kbps
HDLC frame length	≤ 1470 bytes
Synchronous clock	Normal, slave and master clock mode
ESD protection	± 15 KV
Ethernet Interface	
Quantity	2 x RJ-45
Function	Ethernet switching support VLAN and dual-IP support
Rate	10/100 Mbps, auto MDI/MDI-X
Network protocol	TCP/IP
Programming interface	UDP Server, UDP Client Unicast/Multicast/Broadcast
Configuration Management	
Software tool	yacer-DMS configuration management software
Management interface	Ethernet interface
Power Requirements	
Input voltage	+12V
Power consumption	< 3W
Power interface	DC-5.5-2.5mm interface
Mechanical Characteristics	
Dimensions	HxWxD: 32 mm x 193.55 mm x 124 mm
Weight	450g
Enclosure	Aluminum alloy fully-enclosed enclosure, IP30
Operating Environment	
Operating temperature	-40 ~ +85°C
Storage temperature	-40 ~ +85°C
Operating humidity	5~95% (non condensing)

1.5 Order Information

Model	RS-232 Serial Port	Ethernet Interface
HDLC-ETH-200	2 x synchronous/asynchronous serial ports	2 x 10/100M
HDLC-ETH-400	2 x synchronous/asynchronous serial ports + 2 x synchronous serial ports	2 x 10/100M

2 Hardware Interface

2.1 Appearance

Quad serial interfaces (S1~S4) and dual Ethernet interfaces (ETH1, ETH2) are located on the front side of the product while the DC power interface is located on the back side.

The serial port and Ethernet interface both adopt the RJ-45 connector.



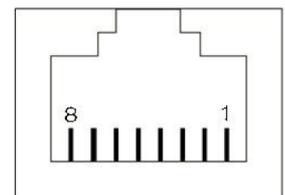
2.2 LED Indicators

LED	Description
ALARM	Alarm indicator, on when the device is not ready to start or in case of failure, and constantly off during normal operation
RUN	Running indicator, flashing during normal operation
POWER	Power indicator, constantly on after powered

2.3 Ethernet Interface

The dual 10/100M Ethernet RJ-45 interface supports for auto MDI/MDIX. Ethernet switching, VLAN and dual-IP support.

RJ45 Pin	Ethernet Signal
1	Tx+
2	Tx-
3	Rx+
6	Rx-



2.4 Serial Port

With the RJ-45 connector, it provides dual or quad RS-232 synchronous/asynchronous serial ports with support for the HDLC protocol. Each RJ45 is provided with two indicators; the yellow light flashing indicates data transmission while the green light flashing indicates data receiving.



RJ45 Pin	RS-232	Description
1	GND	Ground
2	TxDATA	Transmit Data
3		
4	TxClock	Transmit Clock
5		
6	RxDATA	Receive Data
7		
8	RxClock	Receive Clock

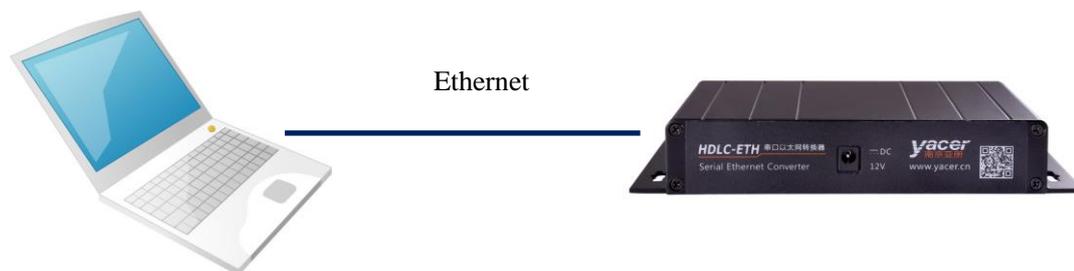
2.5 DC: power interface

HDLC-ETH operates with the +12V DC power supply.



3 Building Configuration Environment

Connect the management computer and HDLC-ETH over the network cable, as shown below.



4 yacer-DMS Configuration Management Software

4.1 Get the Configuration Management Software yacer-DMS

4.1.1 Official website of yacer

Visit the official website of yacer <http://en.yacer.cn> to enter the “Tools” channel, and open the yacer-DMS software page to download the latest version of the software.

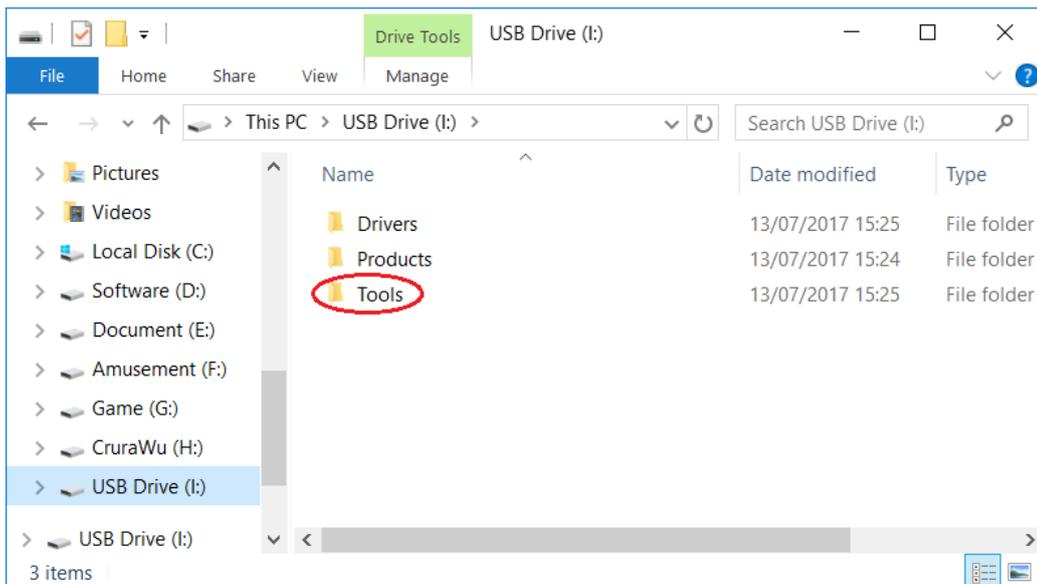
The screenshot shows the yacer website's navigation menu with options like Home, Products, Solutions, Tools, Applications, and Knowlag. The main content area displays the 'Yacer-DMS Configuration management software' page. A sidebar on the left has a 'Software' section with a 'Debugger' link. The central window shows a table with one device entry: Status 'OK', Model 'HDLC-USB-200', S/N '13164260', and IP Address '192.168.3.130'. Below the table is an 'HDLC-USB-200 Report' with a refresh period of 1 second. The report details include: Running time: 3947 seconds; Device S/N: 13164260; IP Address: 192.168.3.130; Hardware Version: 2.1; Firmware Version: 2016.0629; Serial Port S1: Clock = 9.6 KHz, Tx = 0, Rx = 0; S2: Clock = 9.6 KHz, Tx = 0, Rx = 0; UDP Transmit; UDP Receive; DMS Service Tx = 4842, Rx = 4833, Message Length: config = 252 bytes, report = 256 bytes.

Yacer-DMS Configur: software

- Automatically discover device
- Real-time monitoring equipment
- Configure the operating parameters
- Free learning, easy to use, easy to install
- Green version, free installation

4.1.2 Accompanied USB disk

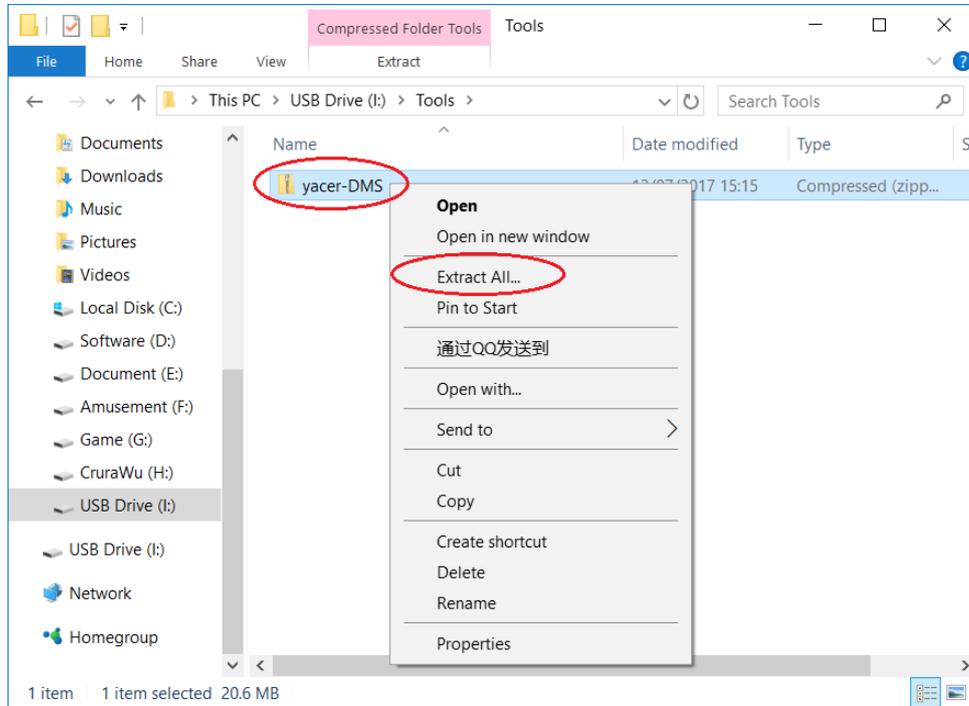
Insert the accompanied USB disk to the PC, open it and double-click to enter the “Tools” folder.



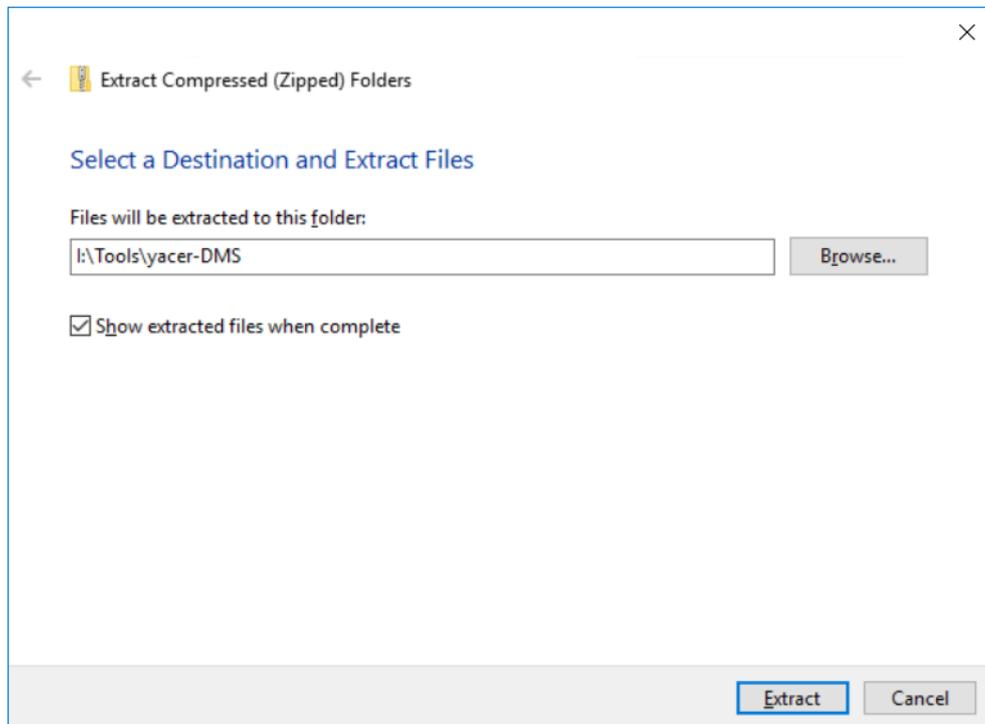
4.2 Run yacer-DMS

4.2.1 Unzip the file

After getting the yacer-DMS.zip compressed file, first unzip it.

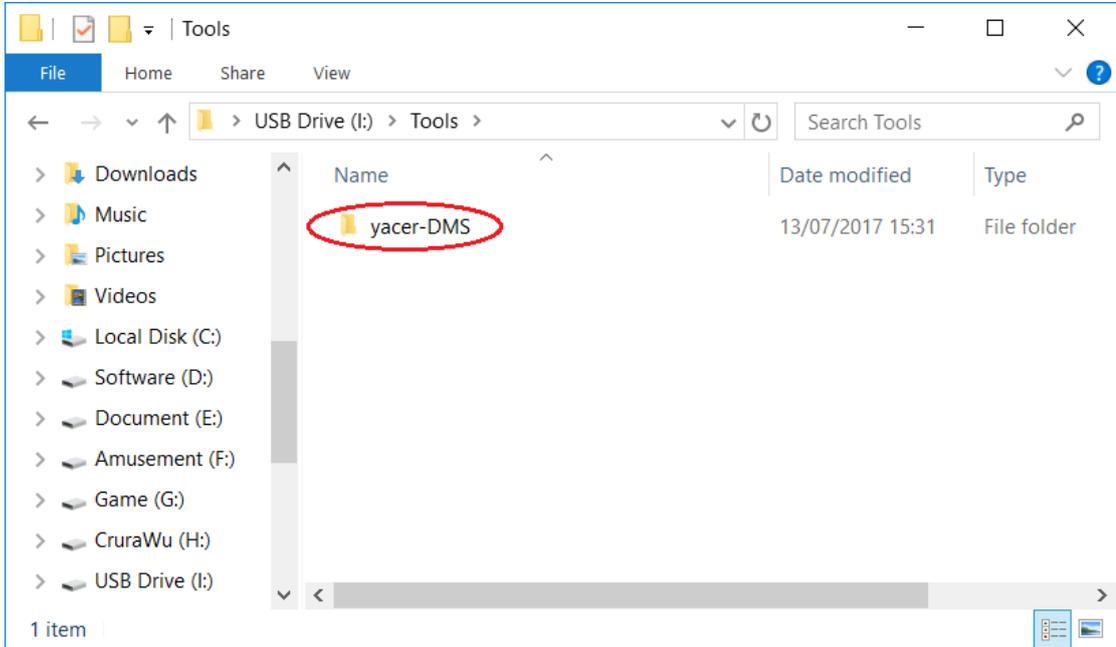


With the Extract Compressed Folder dialog box popping up, set the path for saving extracted files in the box and click on the “Extract” button once set up.

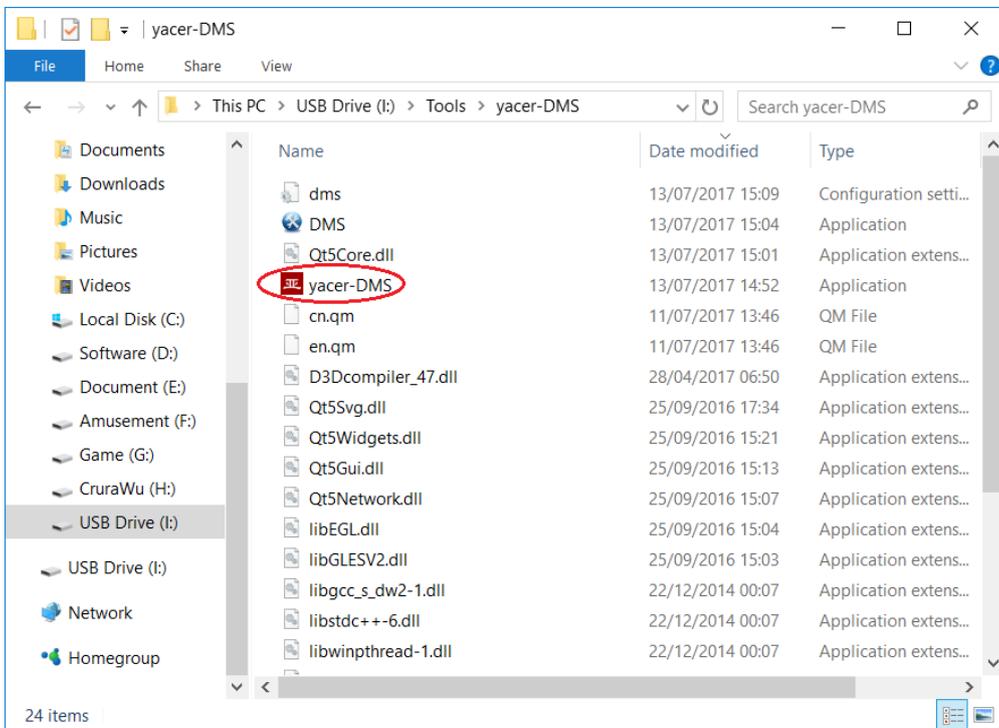


4.2.2 Run yacer-DMS

Locate the decompressed folder according to the path for extracting the compressed files and click on it.



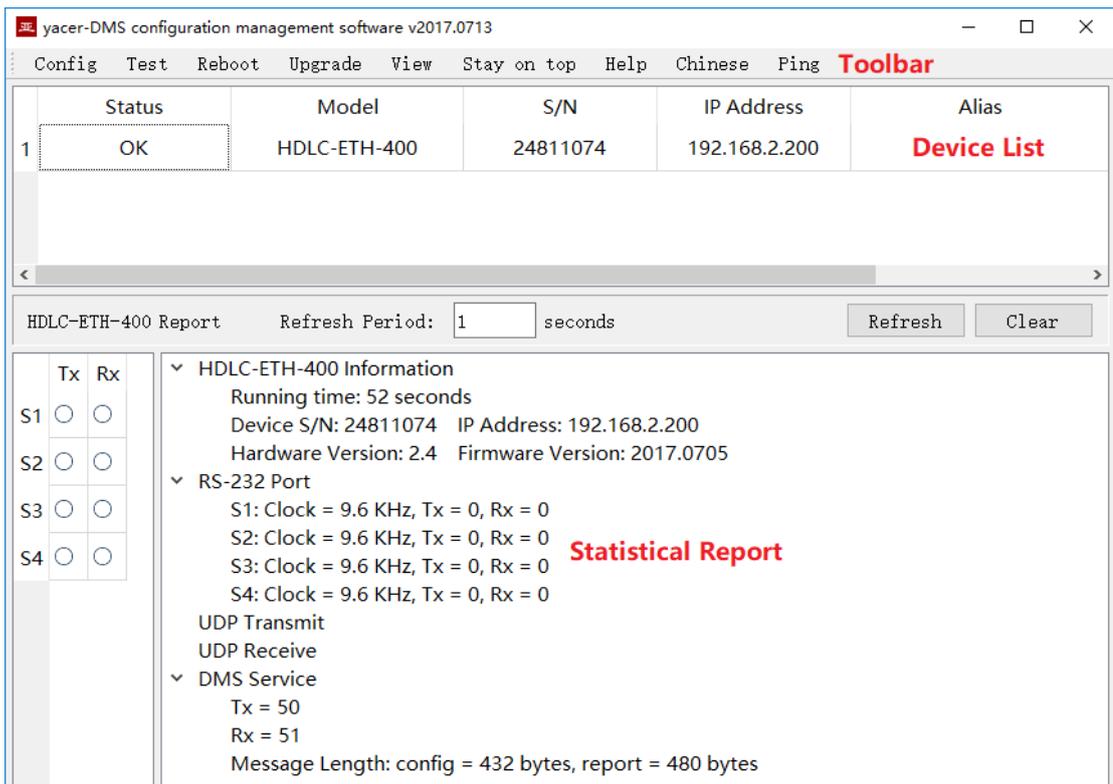
Locate the yacer-DMS.exe file under directory and double-click on this file to run the configuration software.



4.3 Main Window of DMS

Below is the main window of the configuration management software yacer-DMS, including three parts:

- Toolbar: Function operation buttons
- Device List: Displaying the basic information and running status of the on-line device
- Statistical Report: Displaying the receive/transmit indication & statistics, and device details of the specified device

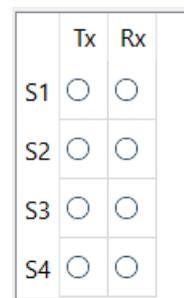


4.4 Statistical Report

4.4.1 Receive/transmit indication

Transmit: Each time the corresponding serial port transmits one frame data, the transmit indicator flashes once.

Receive: Each time the corresponding serial port receives one frame data, the receive indicator flashes once.



4.4.2 Information display

The information display area is located on the right side of the statistical report, showing the following contents:

- Device Information: Running time, serial number, IP address, version number
- Serial Port: Receive/transmit statistics of all serial ports
- UDP Transmit: Displaying the relevant transmitted packets of the UDP Client for each enabled serial port to UDP entry
- UDP Receive: Displaying the relevant received packets of the UDP Server for each enabled UDP to serial port entry
- DMS Service: Displaying the information receive/transmit statistics of the configuration management between the device and the configured management computer

▼ HDLC-ETH-400 Information
 Running time: 52 seconds
 Device S/N: 24811074 IP Address: 192.168.2.200
 Hardware Version: 2.4 Firmware Version: 2017.0705

▼ RS-232 Port
 S1: Clock = 9.6 KHz, Tx = 0, Rx = 0
 S2: Clock = 9.6 KHz, Tx = 0, Rx = 0
 S3: Clock = 9.6 KHz, Tx = 0, Rx = 0
 S4: Clock = 9.6 KHz, Tx = 0, Rx = 0

UDP Transmit
 UDP Receive

▼ DMS Service
 Tx = 50
 Rx = 51
 Message Length: config = 432 bytes, report = 480 bytes

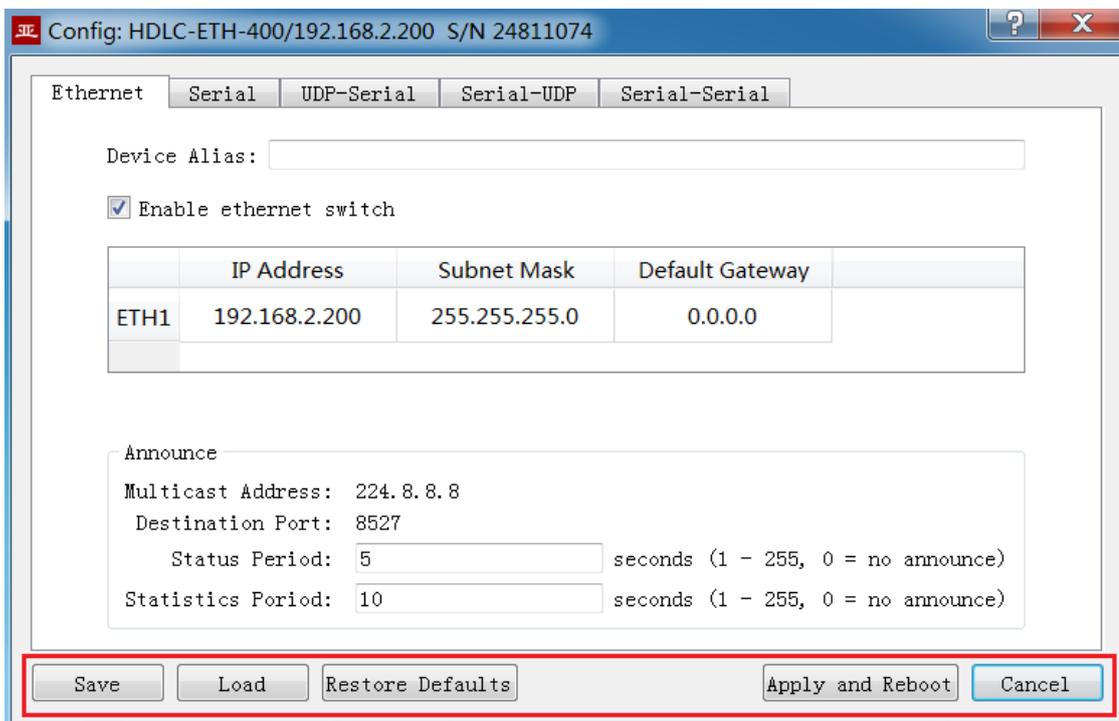
4.4.3 Control Widget

The statistical report includes the following control widgets:

Control Widget	Function
Refresh Period: <input type="text" value="1"/> seconds	Set the refresh period of report
<input type="button" value="Refresh"/>	Manually refresh the statistical report
<input type="button" value="Clear"/>	Clear the statistical report

4.5 Configure Device

Click on the **Config** button on the toolbar or double-click on the specified device in the device list; yacer-DMS pops up the configuration dialog. According to the interface and function, the dialog divides the configuration item into several configuration pages.



The following operation buttons are located at the bottom of the dialog:

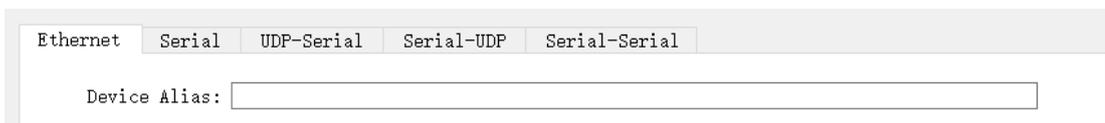
Button	Function
Save	Save the configuration content of the current dialog to the file
Load	Open the configuration file and read the refreshed configuration dialog content
Restore Defaults	Refresh the dialog content with the device's default factory configuration
Apply and Reboot	Write the configuration content of the dialog into the device and restart the device to bring the configuration into effect
Cancel	Cancel the current configuration operation

5 Function and Configuration

5.1 Ethernet Interface Configuration

5.1.1 Device alias

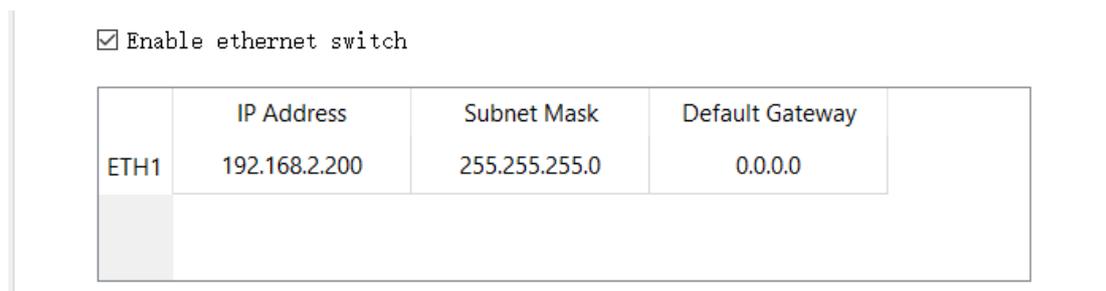
It allows users to set an alias for the HDLC-ETH, thus adding description to the device or helping to remember the identification.



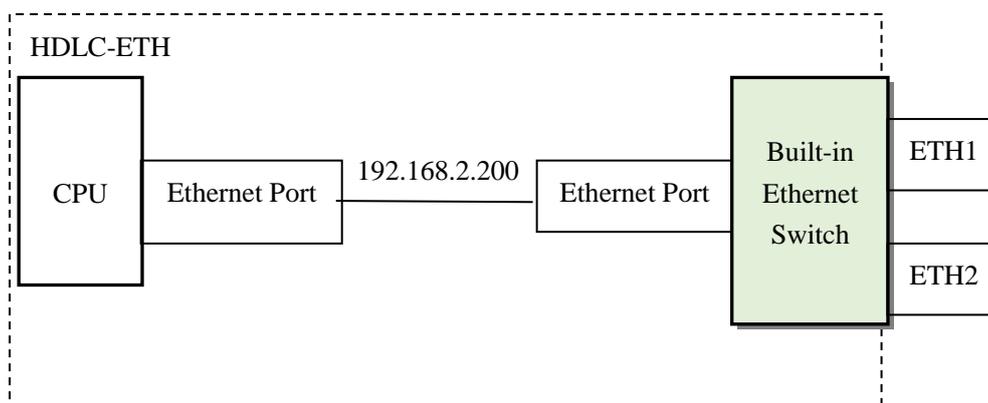
5.1.2 IP address configuration

5.1.2.1 Ethernet switching enable

By default, check the Enable ethernet switch checkbox to enable the built-in Ethernet switch, providing the Ethernet switching function between ETH1 and ETH2.



While the Ethernet switching function has been enabled, HDLC-ETH only has one IP address. The network function figure as follows:



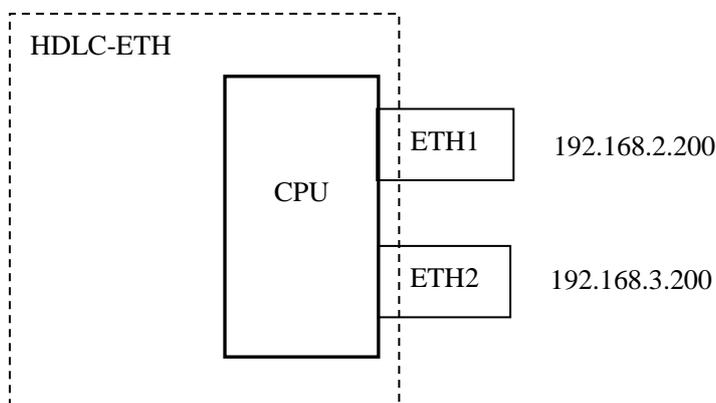
5.1.2.2 Dual IP configuration

When the Enable ethernet switch checkbox is unchecked, ensure ETH1 and ETH2 are not on the same subnet for configuration as they have an independent IP address.

Enable ethernet switch

	IP Address	Subnet Mask	Default Gateway
ETH1	192.168.2.200	255.255.255.0	0.0.0.0
ETH2	192.168.3.200	255.255.255.0	0.0.0.0

With the dual-IP function figure as follows, the HDLC-ETH is equivalent to a PC equipped with two network cards.

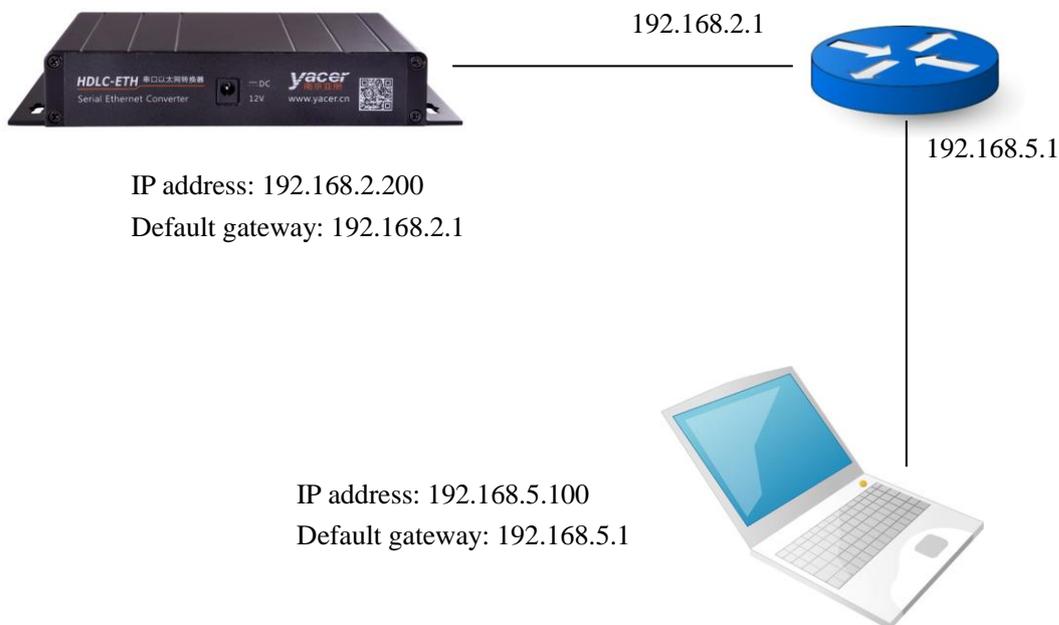


5.1.3 Default gateway

By default, the default gateway is 0.0.0.0, representing that there is no gateway configuration.

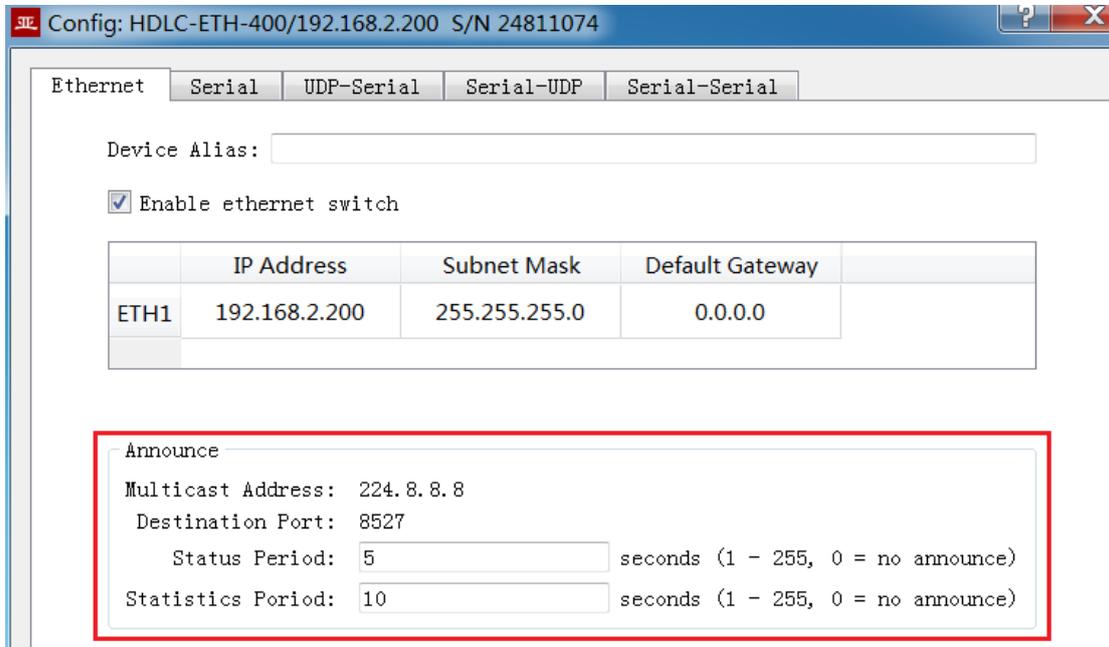
If HDLC-ETH needs to communicate with the host on other subnet, it must rely on an external router. At this time, the HDLC-ETH's IP address must be on the same subnet with the IP address of the connected router port. Meanwhile, the IP address of router is set to the default gateway.

As shown below, the IP address of HDLC-ETH and remote PC is 192.168.2.200 and 192.168.5.100 respectively. As they do not belong to the same subnet, they must rely on the router for communication. HDLC-ETH and PC need to set the IP address of the connected router port to the default gateway of this device.



5.1.4 Announce Configuration

HDLC-ETH can announce the notification data and provides the host computer with the serial card operation information and receive/transmit statistics of the data communication.



The announce setting includes the following parameters:

Parameters	Description
Multicast address	The notification is sent via multicast with the multicast address of 224.8.8.8 that can't be changed
Destination port	Notification is sent via the UDP message with the destination port number of 8527 that can't be changed
Announce period of the device information	Transmit period of device running the information notification, 1~255 seconds
Announce period of the statistical report	Transmit period of receive/transmit statistics, 1~255 seconds

5.2 Serial Port Configuration

5.2.1 Select the working mode of the serial port

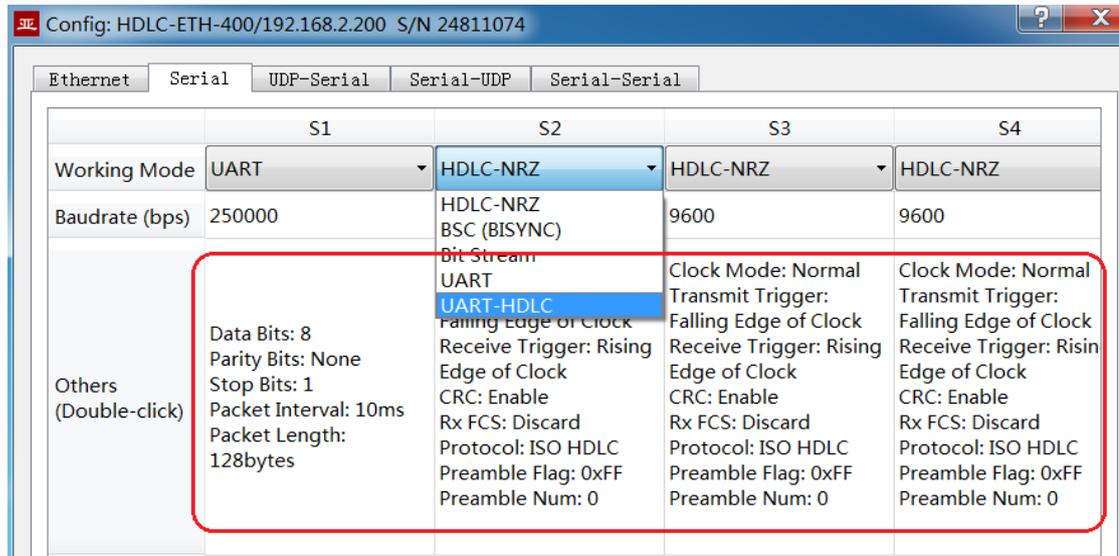
Serial S1 and S2 are synchronous and asynchronous serial ports, with support for the synchronous and asynchronous working modes.

Other serial ports are synchronous, with support only for the synchronous working mode.

Working Mode		Description	Ports Supported
Asynchronous	UART	universal asynchronous serial, equivalent to the serial port on the common computer	S1, S2
	UART- HDLC	UART-based similar HDLC communication protocol	S1, S2
Synchronous	HDLC-NRZ	Synchronous HDLC protocol based on the NRZ encoding	S1, S2, S3, S4
	Bit Stream	Serial Bit data based on the receive clock sampling	S1, S2, S3, S4

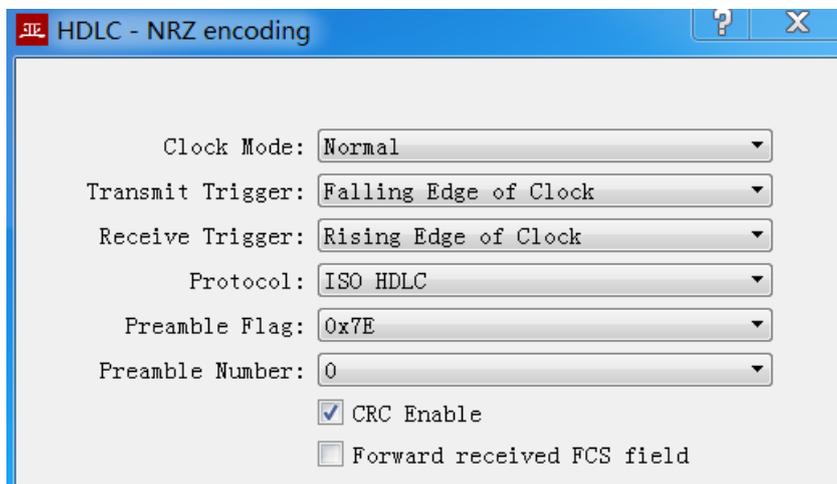
Users can select the desired working mode from the “working mode” combobox. Due to different parameter configuration of each working mode, contents of the “Others” cell will be adjusted automatically according to the determined working mode.

If further configuration of working parameters of the selected working mode is required, mouse double-click on the “Others” cell to pop up the parameter configuration dialog.

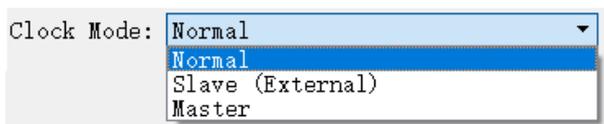


5.2.2 HDLC-NRZ parameter configuration

HDLC-NRZ is the common synchronous working mode, which is mainly used for the secondary surveillance radar and ADS-B data communication in the air traffic control (ATC) and air traffic management (ATM) fields.



5.2.2.1 Clock mode

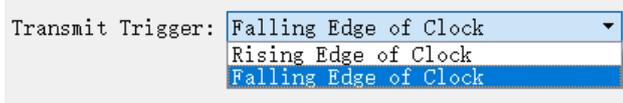


There are three clock modes for the synchronous serial port, normal, slave clock and master clock.

Clock Mode	Transmit Clock	Receive Clock
Normal	Generation from the local device, TxC output	Generation from the opposite device, RxC input
Slave Clock (External)	Generation from the opposite device, RxC input TxC output synchronizes with RxC automatically	Generation of the opposite device, RxC input
Master Clock	Generation of the local device, TxC output	Generation of the local device, ignoring the RxC clock

The slave clock mode is also called as the external clock working mode. When the opposite device is the DCE, HDLC-ETH is often configured as the slave clock mode and transmits data with the clock provided by the DCE, ensuring the data transmission across the whole network based a clock and avoiding packet loss concerns caused due to different clock sources.

5.2.2.2 Transmit trigger

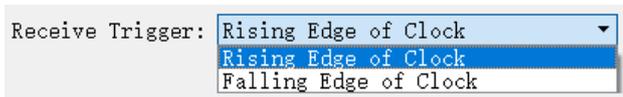


Transmit trigger defines the generation clock edge of the new data bit:

- Falling edge of clock: A new data bit is generated on the falling edge of clock
- Rising edge of clock: A new data bit is generated on the rising edge of clock

During the ATC communication, falling edge of clock is generally chosen to trigger the new data transmission. In some custom development applications, sometimes users will adopt the rising edge to trigger the new data transmission.

5.2.2.3 Receive trigger



Receive trigger defines the sampling clock edge of the serial port receive data:

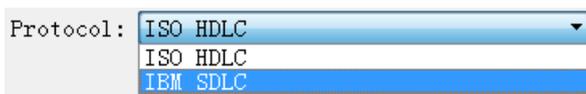
- Rising edge of clock: Data on the RxD line is read on the rising edge of the RxC signal
- Falling edge of clock: Data on the RxD line is read on the falling edge of the RxC signal

During the ATC communication, receive trigger must be configured as the rising edge of clock to ensure correct reading of data as it takes time to stabilize the new data that is triggered with the falling edge.

The local receive trigger configuration is determined according to the transmit trigger of the remote terminal device:

Remote Transmit Trigger	Local Receive Trigger
Falling edge of clock	Rising edge of clock
Rising edge of clock	Falling edge of clock

5.2.2.4 Protocol



- ISO HDLC: ISO HDLC protocol standard
- IBM SDLC: SDLC protocol standard

5.2.2.5 Leading Sign and Number

Preamble Flag: 0x7E

Preamble Number: 0
 1
 2
 3
 4
 5
 6
 7

In some specific applications, especially the half-duplex applications, a leading sign is required in front of the frame. For example, 2~5 0x7E leadings are added for synchronization of the receiver.

For the full-duplex applications, leading is always not required with the number of leadings set to be zero (no leading).

5.2.2.6 CRC enable

CRC Enable

By default, the HDLC protocol communication should enable the CRC function.

CRC Enable	Data Transmit	Data Receive
<input checked="" type="checkbox"/> CRC Enable	HDLC-ETH automatically calculates the CRC data, and adds a 2-byte FCS field at the end of data	Check CRC of the receive frame and discard the data frame with the check failed
<input type="checkbox"/> CRC Enable	No CRC calculation or FCS field adding	No CRC check and receive all frames

5.2.2.7 Forward received FCS field

Forward received FCS field

This configuration is only effective with CRC enable. The HDLC frame structure is shown in the following table, where FCS is the frame check sequence field.

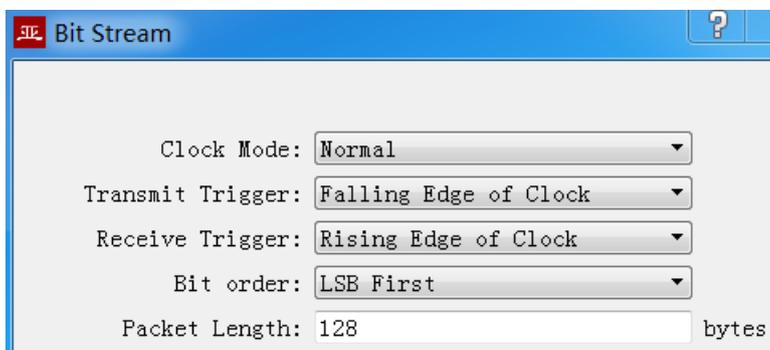
By default, this option is not checked; HDLC-ETH will discard the 2-byte FCS field at the end of data and only forward the user data after the receive HDLC frame check is passed.

If this option is checked, then forward the user data and FCS field.

Opening Flag	Address Field	Control Field	Information Field	FCS Field	Closing Flag
0x7E	1 byte	1 byte	Variable length	CRC 2 bytes	0x7E
0x7E	User data			CRC 2 bytes	0x7E

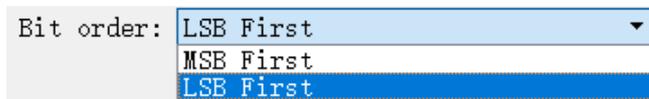
5.2.3 Bit Stream parameters

The rising or falling edge of each clock cycle samples the 1bit data on the data line, which forms a UDP message and sends to the destination IP after receiving a byte with the packet length by forming a byte with each 8bit.



Refer to the HDLC-NRZ parameter configuration for configuration of clock mode, transmit trigger, receive trigger and idle flag.

The online bit stream is stored in the computer or system memory in the form of byte. The receive/transmit sequence determines the conversion mode of byte and bit.

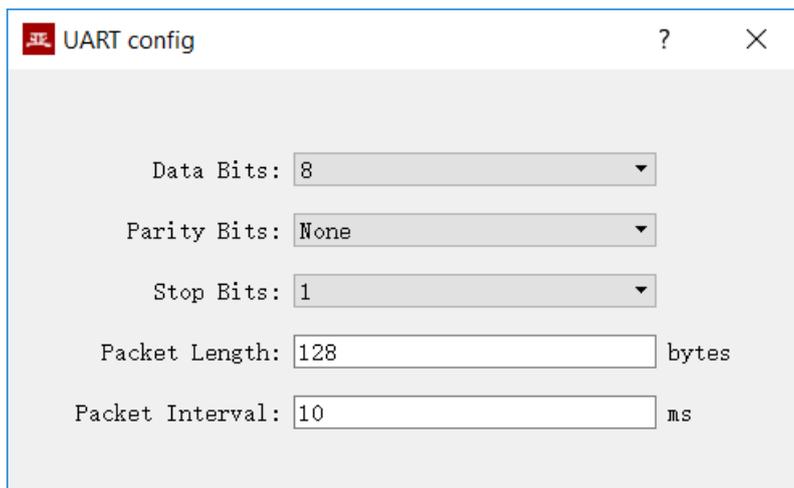


Receive/Transmit Sequence	Transmit Operation	Receive Operation
MSB first	First transmit the high-bit byte	Data received first is placed on the byte high bit
LSB first	First transmit the low-bit byte	Data received first is placed on the byte low bit

5.2.4 UART parameter configuration

Asynchronous UART is a character stream communication; data bits, parity bits and stop bits define the basic working parameters of the asynchronous serial port, which must be configured identically to the opposite terminal device.

Generally, data bits are defined as 8, i.e. one byte, and UART corresponds to the byte stream communication.



The image shows a software window titled "UART config" with a red icon on the left and a question mark and close button on the right. The window contains five configuration fields:

- Data Bits: 8 (dropdown menu)
- Parity Bits: None (dropdown menu)
- Stop Bits: 1 (dropdown menu)
- Packet Length: 128 bytes (text input)
- Packet Interval: 10 ms (text input)

When the UART byte stream is converted into a UDP message or HDLC frame, it is too costly and inefficient if each byte is converted to a UDP message for transmission.

To improve efficiency, the protocol converter buffers the received byte stream and forms a number of buffered bytes into a UDP message to send, of which this process is called as packet.

Packets are controlled with two parameters, namely the packet length and the packet interval.

5.2.4.1 Packet length

For example, if the packet length is set to 128 bytes, then it will form a UDP message to send after UART receives the full 128 bytes.

5.2.4.2 Packet interval

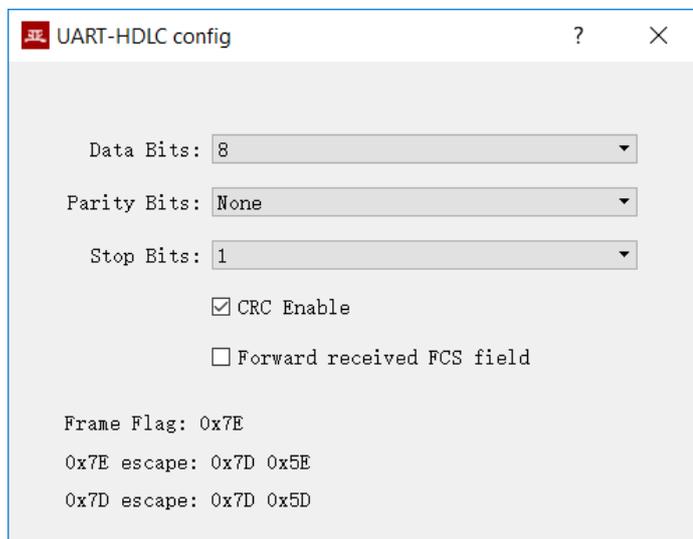
Packet improves the UDP transmission efficiency, but also introduces new problems.

As shown in the above example again, if the transmitter sends 100 bytes and stops sending, the receiver will always be waiting as the packet length is less than 128 bytes.

In order to solve the problem of waiting, users need to configure the packet interval parameter, such as set to 10ms; then data received by UART will wait for a maximum of 10ms. In case of 10ms timeout, it will convert the buffer data into a UDP message to send no matter whether it has received the full 128 bytes.

5.2.5 UART-HDLC parameter configuration

The UART-HDLC working mode is a custom protocol by yacer which form the asynchronous HDLC frame on the basis of the normal UART communication by packaging the byte stream. Therefore, the asynchronous serial port can perform the packet-based communication with the UDP message and synchronous HDLC frame.



The UART-HDLC frame format adds 0x7E before and after the packet as the opening flag and closing flag with the frame structure as follows:

opening Flag	Information Field	FCS Field	closing Flag
0x7E	2-1470 bytes of data	2-byte CRC data	0x7E

As the information field and FCS field may appear 0x7E, perform the character escape on such fields before transmission with the escape rules as follows:

- 0x7E: Escaped to two characters, 0x7D 0x5E
- 0x7D: Escaped to two characters, 0x7D 0x5D
- Other characters: No escape

The escape operation of data transmit is as follows:

Original Data	Actual Transmit Data
0x7E	0x7D 0x5E
0x7D	0x7D 0x5D
Others	No change

The escape operation of data receive is as follows:

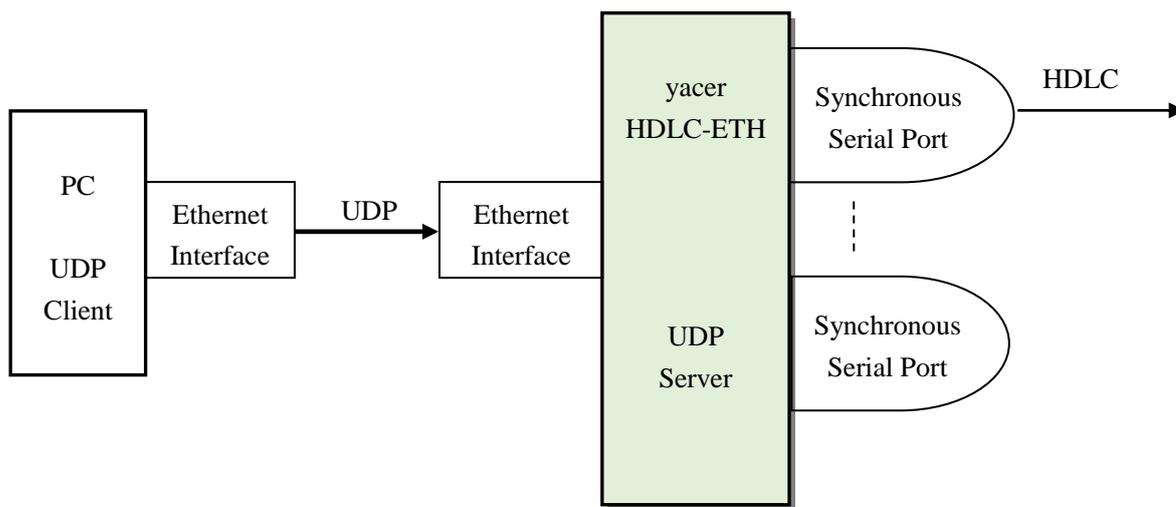
Actual Receive Data	Data
0x7D 0x5E	0x7E
0x7D 0x5D	0x7D
Others	No change

5.3 UDP to Serial Conversion

5.3.1 Application model

With HDLC-ETH, PC or server can realize the transmit function of the synchronous HDLC serial port.

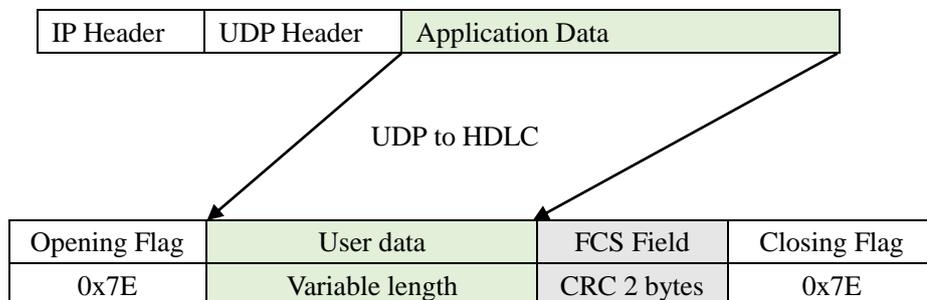
The typical application is shown as follows. PC sends a UDP message over the Ethernet interface as the UDP Client, and HDLC-ETH sends it out from the synchronous serial port after converting the received UDP message into the HDLC frame.



5.3.2 Protocol Conversion

The most typical UDP-to-HDLC application is shown below. HDLC-ETH loads the UDP application data into the user data area of the HDLC frame, and then calculates CRC and populates the FCS field to form a complete HDLC frame to send.

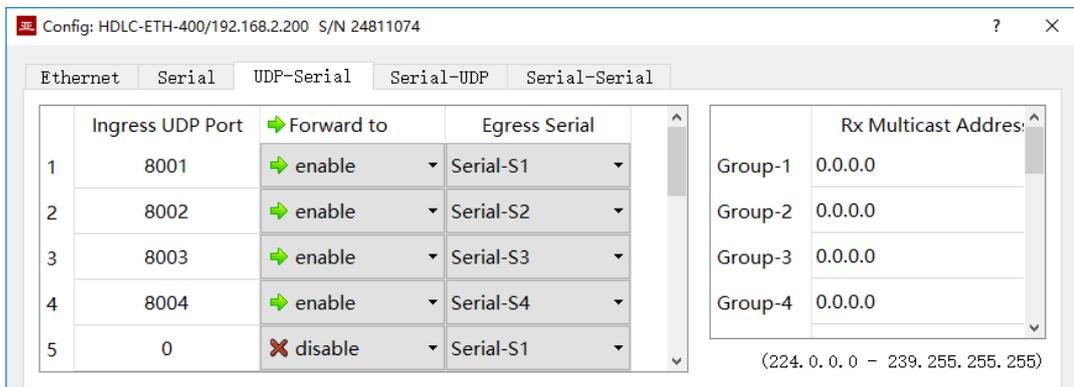
In order to reduce the calculation load of PC and the user programming complexity, normally, the FCS field of HDLC is not included in the UDP message, which is populated through HDLC-ETH calculation.



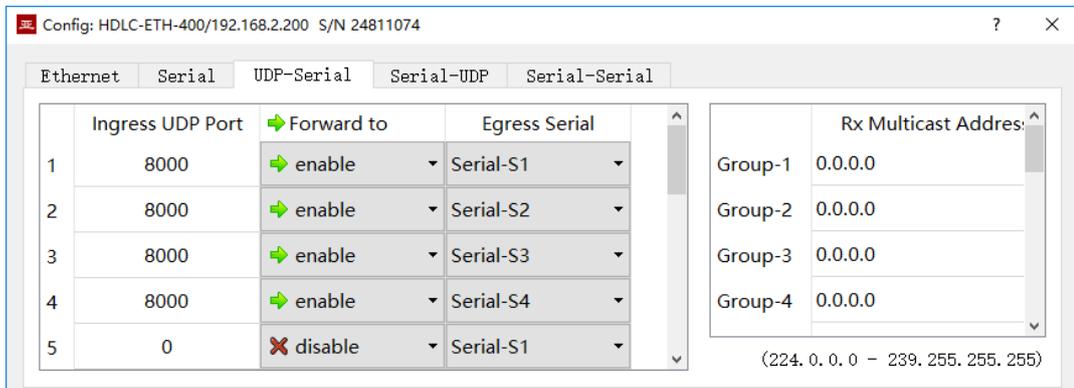
5.3.3 Parameter configuration

Set the UDP to serial port. Each row represents the forwarding entry from a UDP port to the serial port with three forwarding strategies to be achieved:

- Forwarding: Data received by the specified UDP port can be forwarded to the specified serial port.
- Multiplexer: Data received by several different UDP ports can be forwarded to the same serial port.
- Demultiplexer: Data received from the same UDP port can be forwarded to the different serial ports.



The following configuration realizes the application, where data received from a UDP and distributed to 4 serial ports:



5.3.4 Receive multicast

If users need to receive the multicast UDP message, add the required multicast address from the right “Rx Multicast Address” list.

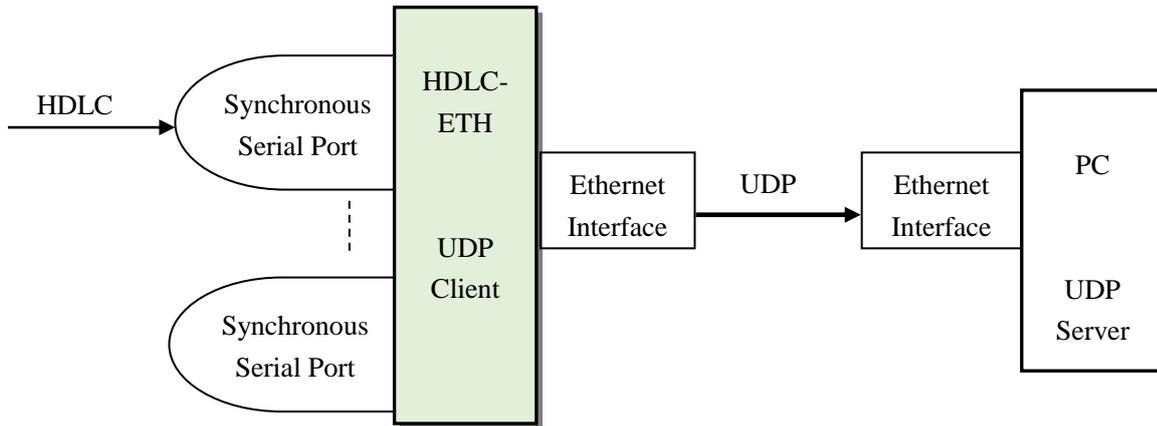
Range of the multicast address is 224.0.0.0 ~ 239.255.255.255, 224.8.8.8 is the configuration management address of the HDLC-ETH and users can’t use this address.

The multicast address configured as 0.0.0.0 indicates that the entry is not in effect.

5.4 Serial to UDP Conversion

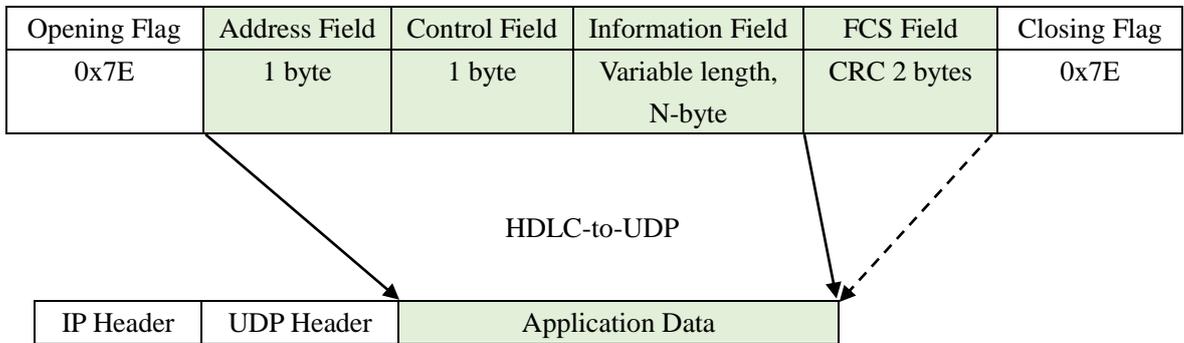
5.4.1 Application model

The HDLC-to-UDP function figure is shown below. HDLC-ETH receives the HDLC frame from other devices over the synchronous serial interface, converts it into the UDP message and sends to PC or server over Ethernet.



5.4.2 Protocol Conversion

To ensure the integrity of user data, HDLC-ETH places the complete HDLC frame in the UDP application data, and forwards to the UDP Server.



5.4.3 Parameter configuration

Set the serial port to UDP. Each row represents the forwarding entry from a serial port to the UDP port with three forwarding strategies to be achieved:

- Forwarding: Data received from the specified serial port can be forwarded to the specified destination UDP port.
- Multiplexer: Data received from several different serial ports can be forwarded to the same destination UDP port.
- Demultiplexer: Data received from the same serial port can be forwarded to the different destination UDP ports.

	Ingress Serial	Forward to	Destination IP Address	Destination UDP Port
1	Serial-S1	enable	192.168.2.80	8001
2	Serial-S2	enable	255.255.255.255	9000
3	Serial-S3	enable	224.10.10.10	10000
4	Serial-S1	disable	0.0.0.0	0
5	Serial-S1	disable	0.0.0.0	0

As shown above, three serial port to UDP entries are configured for achieving the following items:

- Serial port S1 to UDP unicast, with the destination IP address as 192.168.2.80 and destination UDP port as 8000
- Serial port S2 to UDP broadcast, all hosts on the subnet can receive data from S2 at the 9000 port
- Serial port S3 to UDP multicast, only the PC joining Group 224.10.10.10 on the network can receive data from S3.

5.4.4 How does the UDP server identify the source serial

In many applications, such as the ATC, the HDLC frame from several different serial ports needs to be forwarded to a server or PC.

In this case, a strategy enables PC to know which serial port the received UDP message data is from.

5.4.4.1 Distinguish the source serial port according to the destination UDP port

As shown below, set different forwarding destination UDP ports for each serial port. The UDP Server PC receives data at the different UDP ports. Message received at port 8001 is from the serial port S1 while message received at port 8002 is from the serial port S2.

	Ingress Serial	Forward to	Destination IP Address	Destination UDP Port
1	Serial-S1	enable	192.168.2.80	8001
2	Serial-S2	enable	192.168.2.80	8002
3	Serial-S3	enable	192.168.2.80	8003
4	Serial-S4	enable	192.168.2.80	8004
5	Serial-S1	disable	0.0.0.0	0

5.4.4.2 Distinguish the source serial port according to the source UDP port

When the source serial port is identified with the destination UDP port, UDP Server needs to listen and receive data on a plurality of UDP ports. In case there are many serial ports, not only the UDP Server port occupies too many resources, the configuration and programming complexity also increases significantly.

In order to simplify implementation of the UDP Server side, we can use the configuration example below, forwarding each conversion to the same port of the UDP Server. During yacer’s HDLC-ETH forwarding, it will automatically adjust the source port number of the UDP message according to the source serial port. The source ports of the UDP message forwarded by the serial ports S1 and S2 are 8001 and 8002 respectively; the following is gradually increasing.

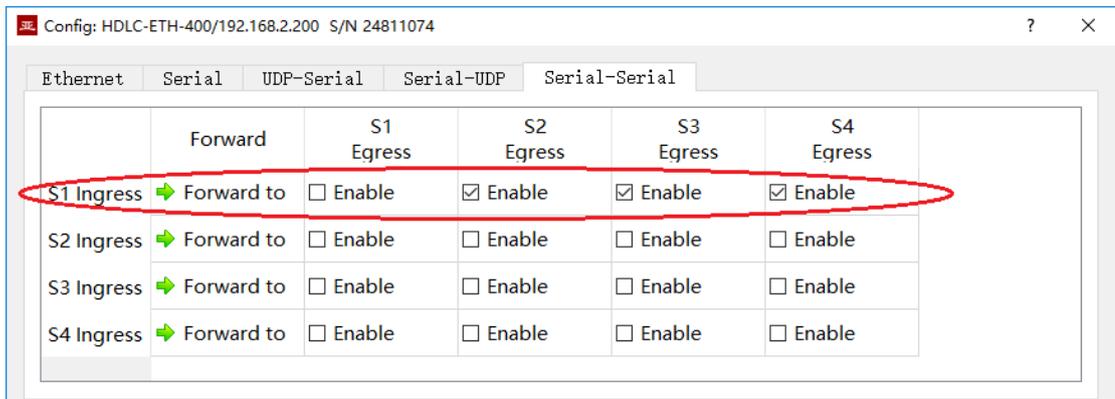
Thus, UDP Server only needs to listen and receive data at a port (8000 in the example below) and distinguishes the source serial port according to the source UDP port. If several HDLC-ETHs are provided, UDP Server can distinguish the source device via the source IP.

	Ingress Serial	Forward to	Destination IP Address	Destination UDP Port
1	Serial-S1	enable	192.168.2.80	8000
2	Serial-S2	enable	192.168.2.80	8000
3	Serial-S3	enable	192.168.2.80	8000
4	Serial-S4	enable	192.168.2.80	8000
5	Serial-S1	disable	0.0.0.0	0

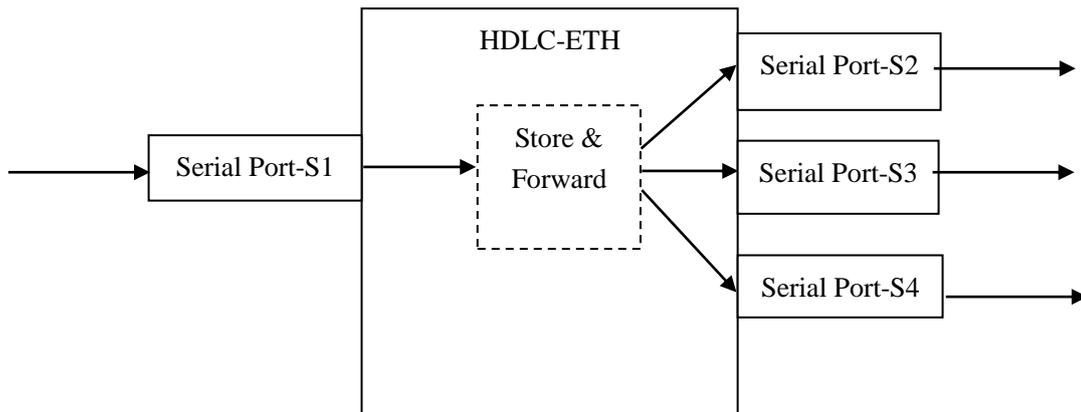
5.5 Serial to Serial Conversion

Serial-serial can forward the input data of the specified serial port to other serial port outputs, which is mainly used for:

- Conversion between synchronous and asynchronous serial ports
- Serial port Demultiplexer: Divide the single serial port data into multi-channel data. Unlike the common demultiplexer, demultiplexing is possible with HDLC-ETH, the different baud rates and clock modes can be set for each-channel serial port, avoiding the packet loss caused by clock inconsistency.



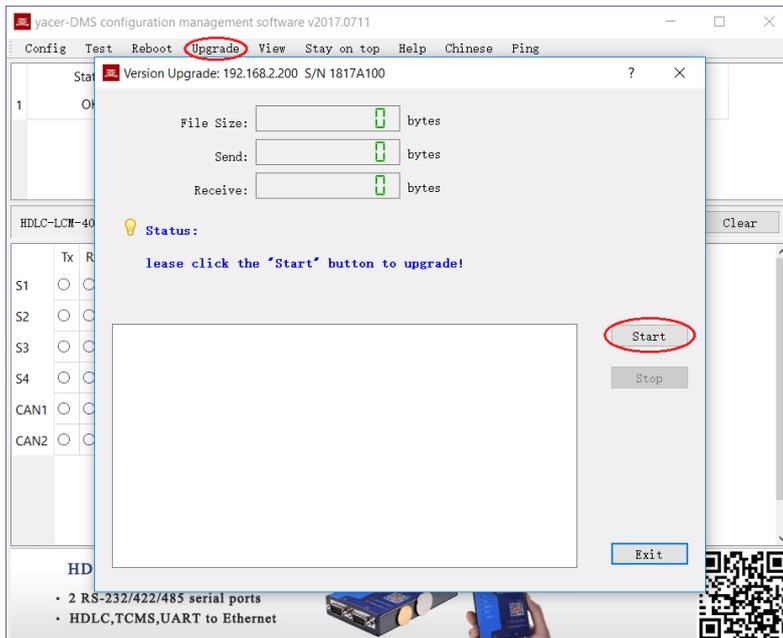
Configuration shown above realizes the application of the input demultiplexing of the serial port S1 to S2, S3 and S4 outputs. HDLC-ETH performs storing and forwarding on the received data. Even if the baud rate and clock mode of S1, S2, S3 and S4 are different, packet will not loss.



6 Firmware Version Upgrade

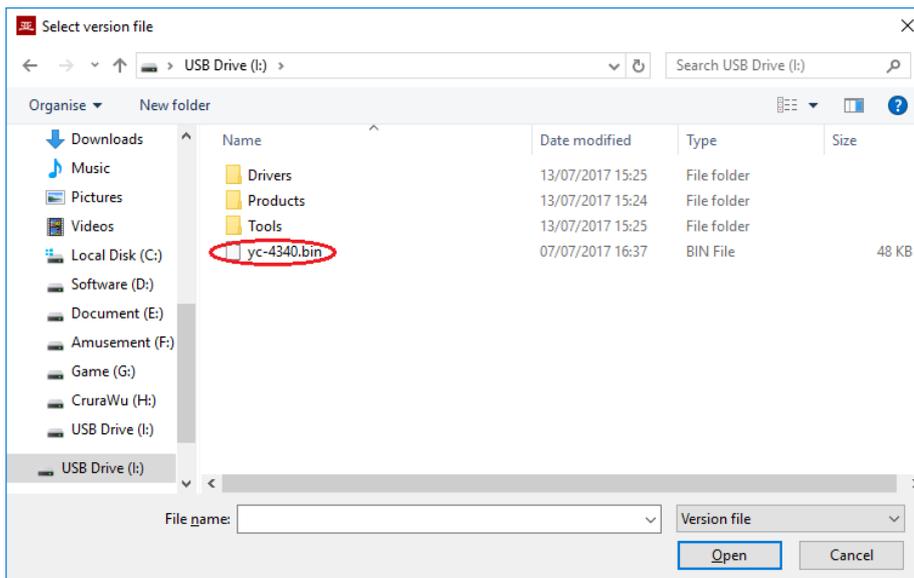
6.1 Start Upgrade

Click on the **Upgrade** button on the toolbar to pop up the version upgrade dialog, and then click on the **Start** button.



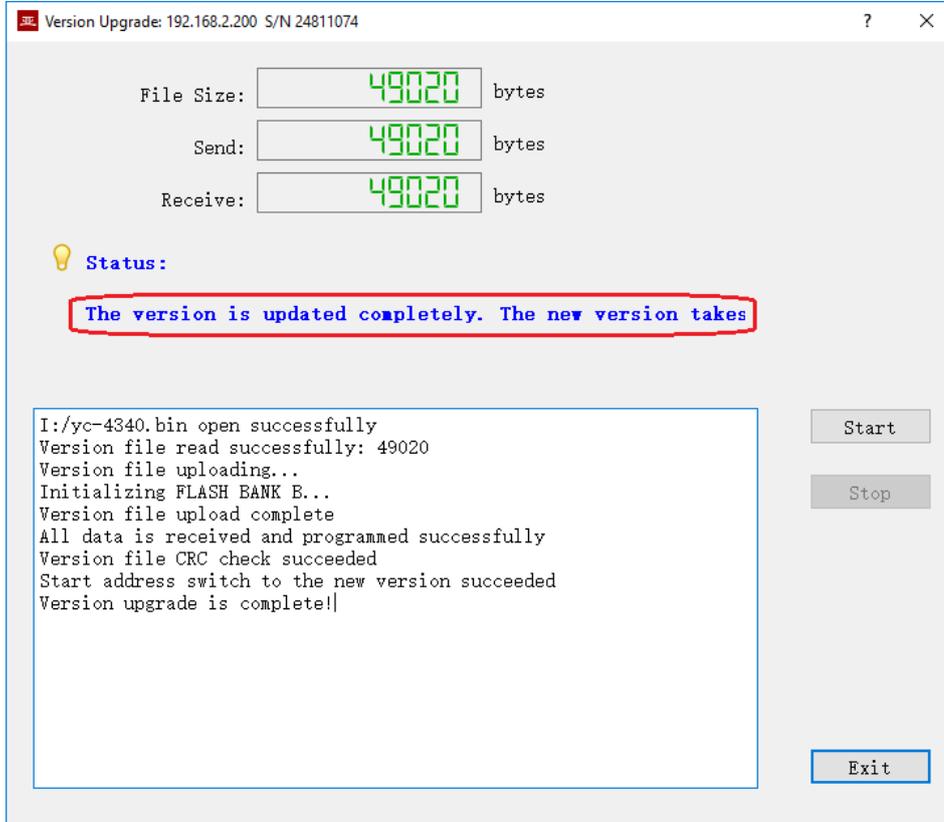
6.2 Locate Firmware Version

The “Selection Version File” dialog pops up. Locate the folder for storing the latest firmware version, select and click “Open” to start updating.



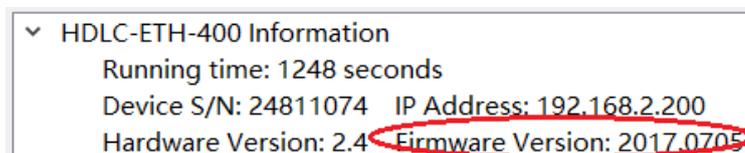
6.3 Upgrade Completed

After completion of upgrade, “Version Update Completed” displaying on the page indicates that the version update is completed.



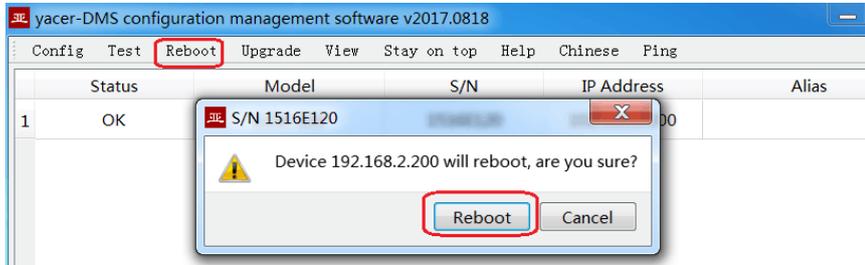
6.4 Upgrade Confirmation

After completion of update, re power-on the device, observe the version information in the statistical report and determine whether the new version is updated successfully via the version date.



7 Device Reboot

Click on the **Reboot** button on the toolbar to pop up the device reboot dialog, and then click on the **Reboot** button to reboot the device.



8 PING

By clicking on the **Ping** button on the toolbar, DMS will start the ping command automatically for the selected device so as to check whether the network connection between the configuration management computer and HDLC-ETH is normal.

Before performing the Ping command, first ensure that the IP address of PC and HDLC-ETH is on the same subnet.

