



HDLC-ATC

Air Traffic Data Communication Server

Rev. Dec 19, 2017



HDLC-ATC

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.....	5
2	Hardware Interface.....	6
2.1	Chassis and Panel.....	6
2.2	LED Indicators	6
2.3	Ethernet Interface	7
2.4	Serial Port.....	7
2.5	Power Supply	8
3	Building Configuration Environment	8
4	yacer-DMS Configuration Management Software.....	9
4.1	Get the Configuration Management Software yacer-DMS	9
4.1.1	Official website of yacer	9
4.1.2	Accompanied USB disk	9
4.2	Run yacer-DMS	10
4.2.1	Unzip the file.....	10
4.2.2	Run yacer-DMS.....	11
4.3	Main Window of DMS	12
4.4	Statistical Report	12
4.4.1	Receive/transmit indication.....	12
4.4.2	Information display	13
4.4.3	Control Widget	13
4.5	Configure Device	14
5	Function and Configuration	15
5.1	Ethernet Interface Configuration.....	15
5.1.1	Device alias	15
5.1.2	IP address configuration.....	15
5.1.3	Default Gateway.....	17
5.1.4	Announce Configuration	18
5.2	Serial Port Configuration	19
5.2.1	Select the working mode of the serial port.....	19
5.2.2	HDLC-NRZ parameter configuration	21
5.2.3	HDLC-NRZI/DBPL/MAN/MAN-Diff parameter configuration	24
5.2.4	Bit Stream parameter configuration	25
5.2.5	UART parameter configuration.....	26
5.2.6	UART-HDLC parameter configuration.....	27
5.3	UDP to Serial Conversion	28
5.3.1	Application model.....	28
5.3.2	Protocol Conversion.....	28

5.3.3	Parameter configuration	29
5.3.4	Receive multicast	30
5.4	Serial to UDP Conversion	30
5.4.1	Application model	30
5.4.2	Protocol Conversion	30
5.4.3	Parameter configuration	31
5.4.4	How does the UDP server identify the source serial	31
5.5	Serial to Serial Conversion	33
6	Firmware Version Upgrade	34
6.1	Start Upgrade	34
6.2	Locate Firmware Version	34
6.3	Upgrade Completed	35
6.4	Upgrade Confirmation	35
7	Device Reboot	36
8	PING	36



1 Overview

1.1 Introduction

The Yacer HDLC-ATC Air Traffic Data Communication Server provides dual 10/100M Ethernet interface and 2, 4, 8 or 16-channel synchronous HDLC serial ports (with two ports supporting asynchronous UART) to achieve the protocol conversion between the serial port and the Ethernet interface.

It is used to generate or receive the synchronous data of the ATC primary radar, secondary radar, ADS-B and other devices, with support for the protocol conversion of HDLC and TCP/IP.

With the 1U, 19-inch standard rack-mounted chassis, low power, fanless and high-reliability design, it supports the AC/DC power redundancy.

1.2 Features

- Dual 10/100M Ethernet interface, with support for the Ethernet switching, VLAN and dual-IP functions
- 2, 4, 8 and 16-channel synchronous HDLC serial ports, with two ports supporting asynchronous UART
- 1U, 19-inch standard rack-mounted chassis
- AC+DC power redundancy support
- Low power, fanless design



1.3 Applications

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

1.4 Technical Specifications

Serial Port	
Quantity	2, 4, 8, 16 x RJ-45
Working mode	Synchronous HDLC, with two ports supporting asynchronous UART
Interface type	RS-232 full-duplex
Encoding format	NRZ
Baud rate	≤ 250 Kbps
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	
Configuration tool	yacer-DMS configuration management software
Configuration interface	Ethernet interface
Power Requirements	
Power redundancy	AC+DC redundancy, AC priority
AC power supply	85 ~ 265 VAC
DC power supply	9 ~ 36 VDC
Power consumption	< 10 W
Mechanical Characteristics	
Dimensions	1U, 19-inch standard rack-mounted chassis fanless design
Weight	3.0 Kg
Operating Environment	
Operating temperature	-10 ~ +60°C
Storage temperature	-30 ~ +75°C
Operating humidity	30 ~ 90% RH (non condensing)
Storage humidity	10 ~ 90% RH (non condensing)

1.5 Order information

Basic product model:

Basic Model	Synchronous/ Asynchronous Serial Port	Synchronous Serial Port	Ethernet Interface
HDLC-ATC-200	2 x RS-232		2 x 10/100M
HDLC-ATC-400	2 x RS-232	2 x RS-232	2 x 10/100M
HDLC-ATC-800	2 x RS-232	6 x RS-232	2 x 10/100M
HDLC-ATC-1600	2 x RS-232	14 x RS-232	2 x 10/100M

Optional function:

Numbering Sequence	Suffix	Function
1	-AA	Dual-AC power redundancy

Product model example:

- HDLC-ATC-800: 8-serial port device, AC+DC redundant power, without the optional function.
- HDLC-ATC-800-AA: 8-serial port device with the dual-AC power redundancy.

2 Hardware Interface

2.1 Chassis and Panel

The chassis adopts the 19-inch standard chassis with the 1U height.

The front panel includes 2, 4, 8 or 16-channel serial interfaces (S1~S16), dual Ethernet interface (ETH1, ETH2) with the RJ-45 connector adopted for the serial port and Ethernet interface.

The rear panel includes:

- Power switch
- AC power socket: 110V/220V AC power input, with a fuse
- DC power socket: 9~36VDC input, with an on-board self-healing fuse
- Ground stud



2.2 LED Indicators

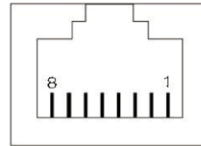
LED indicators on the front panel are as follows:



LED	Color	Description
ALARM	Red	Alarm indicator, flashing during initialization Light on during operation indicates the device failure
RUN	Green	Running indicator, flashing during normal operation
POWER	Green	Power indicator, constantly on after powered
MAIN	Green	Light on indicates that the main power supply is applied Light off indicates that the standby power supply is applied

2.3 Ethernet Interface

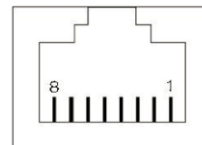
The dual 10/100M Ethernet RJ-45 interface supports for auto MDI/MDIX. Ethernet switching, VLAN and dual-IP support.



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

2.4 Serial Port

With the RJ-45 connector, it provides 2, 4, 8 or 16-channel 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.

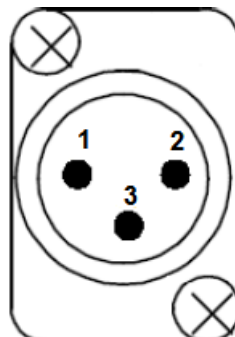


RJ45 Pin	RS-232	Description
1	GND	Ground
2	TxData	Transmit Data
3		
4	TxClock	Transmit Clock
5		
6	RxData	Receive Data
7		
8	RxClock	Receive Clock

2.5 Power Supply

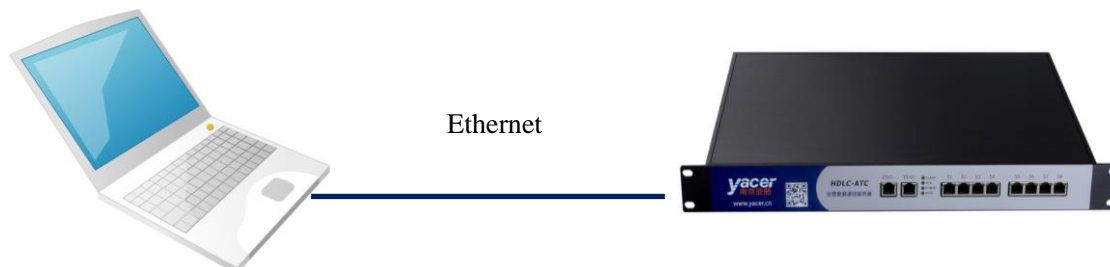
AC+DC redundancy, AC priority. AC supports the 110V/220V standard while DC adopts the 3-core XLR interface.

XLR	DC Power Supply
1	9 ~ 36V
2	GND
3	



3 Building Configuration Environment

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

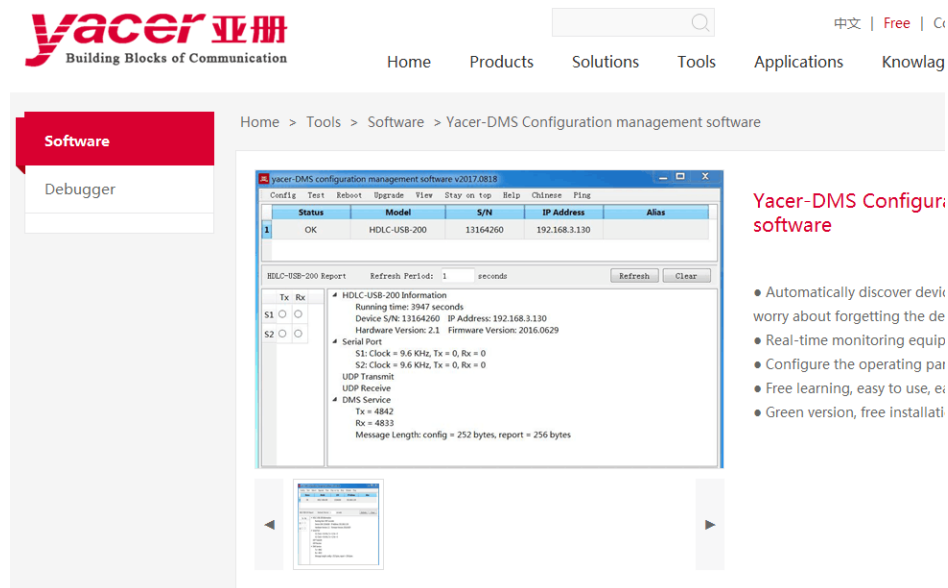


4 yacer-DMS Configuration Management Software

4.1 Get the Configuration Management Software yacer-DMS

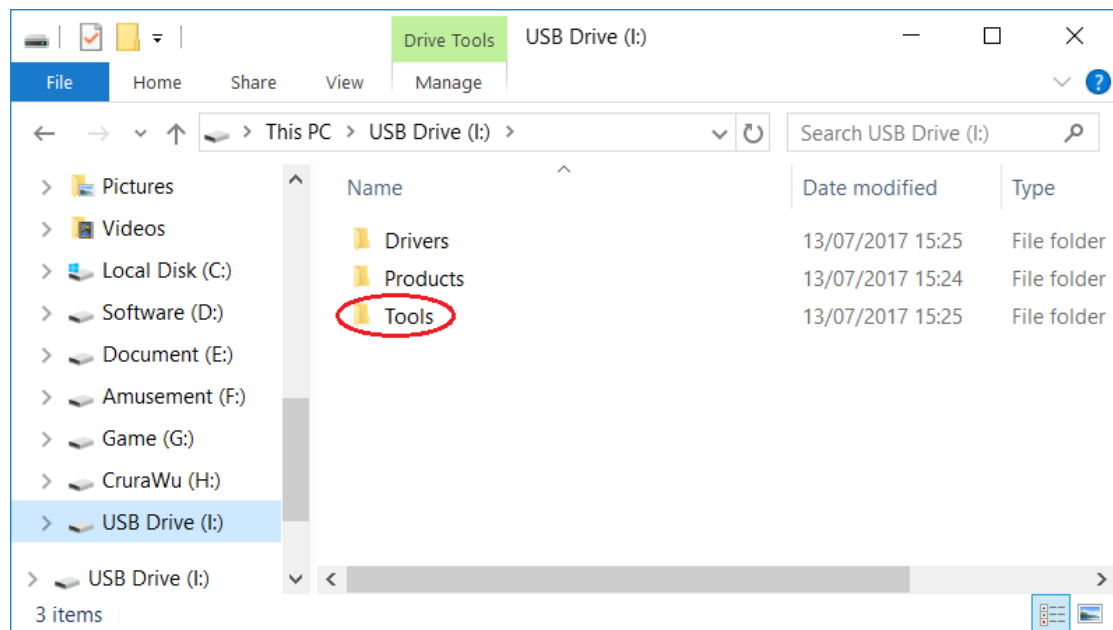
4.1.1 Official website of yacer

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.



4.1.2 Accompanied USB disk

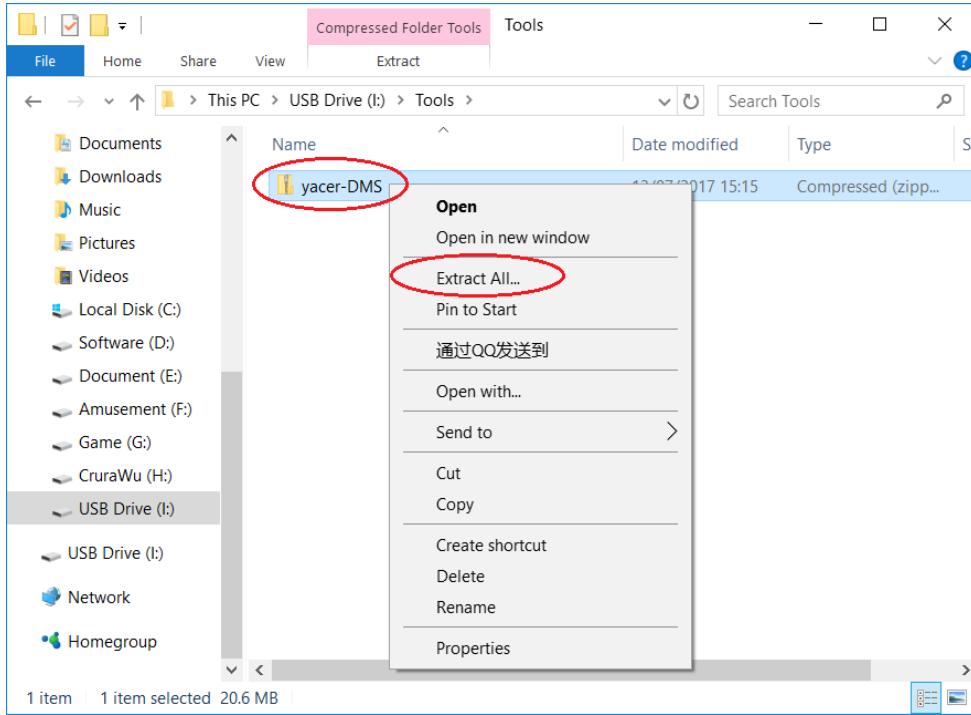
Insert the accompanied USB disk to the PC, open it and double-click to enter the “Tools” folder.



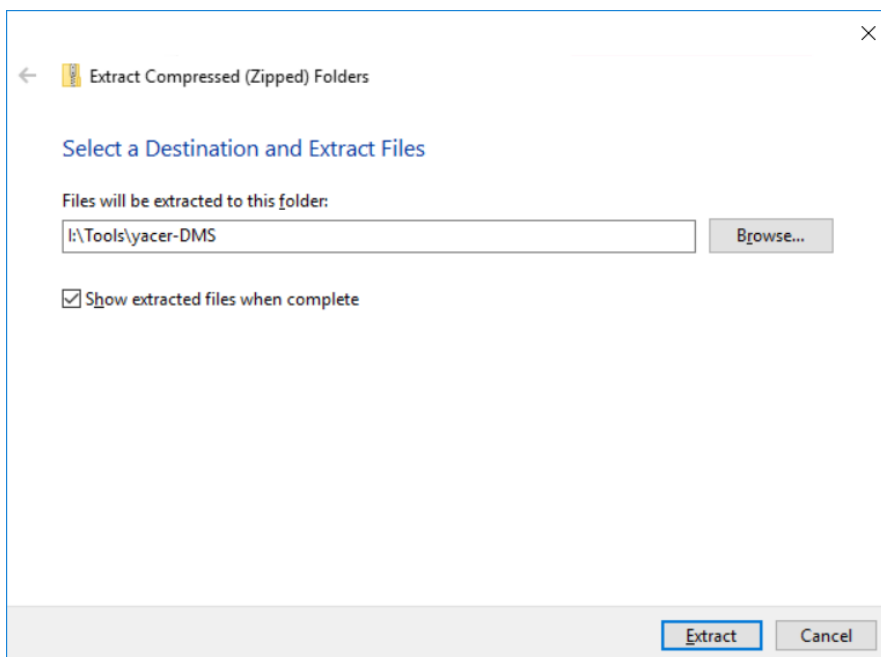
4.2 Run yacer-DMS

4.2.1 Unzip the file

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

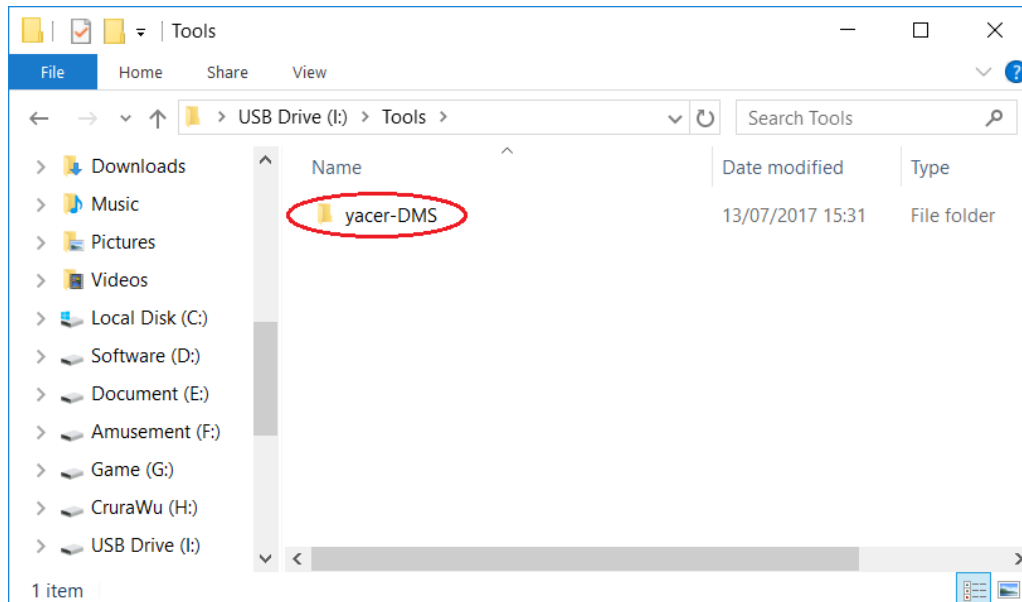


- With the Extract Compressed Folder dialog box popping up, set the path for saving extracted files in the box and click on the “Extract” button once set up.

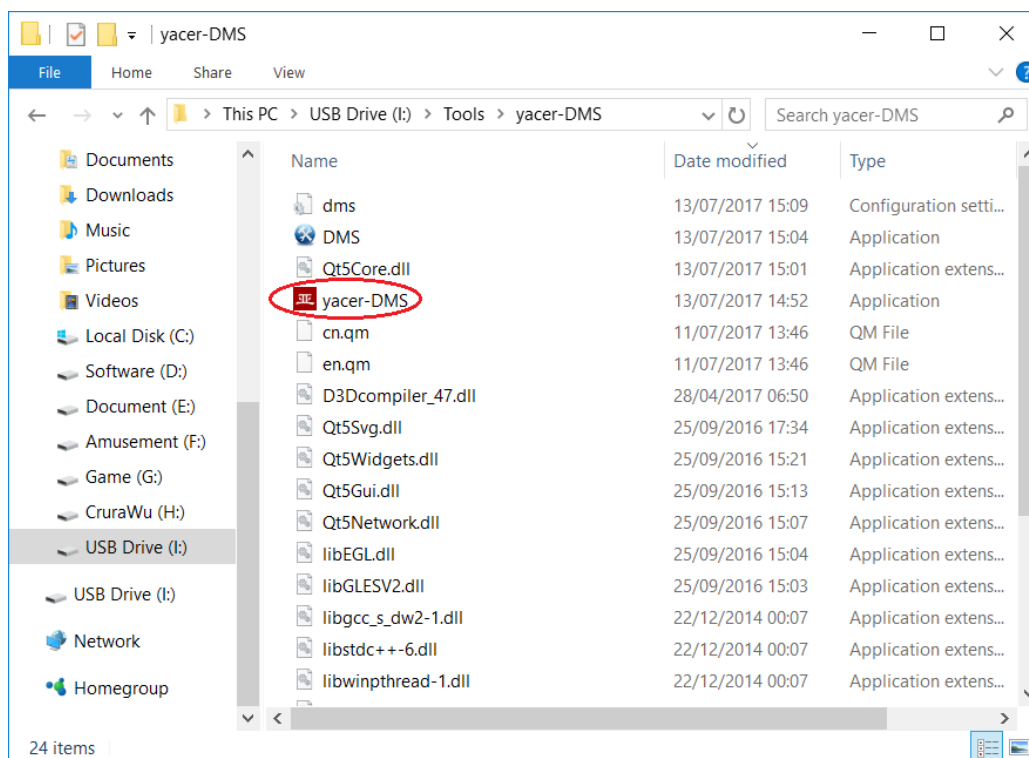


4.2.2 Run yacer-DMS

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



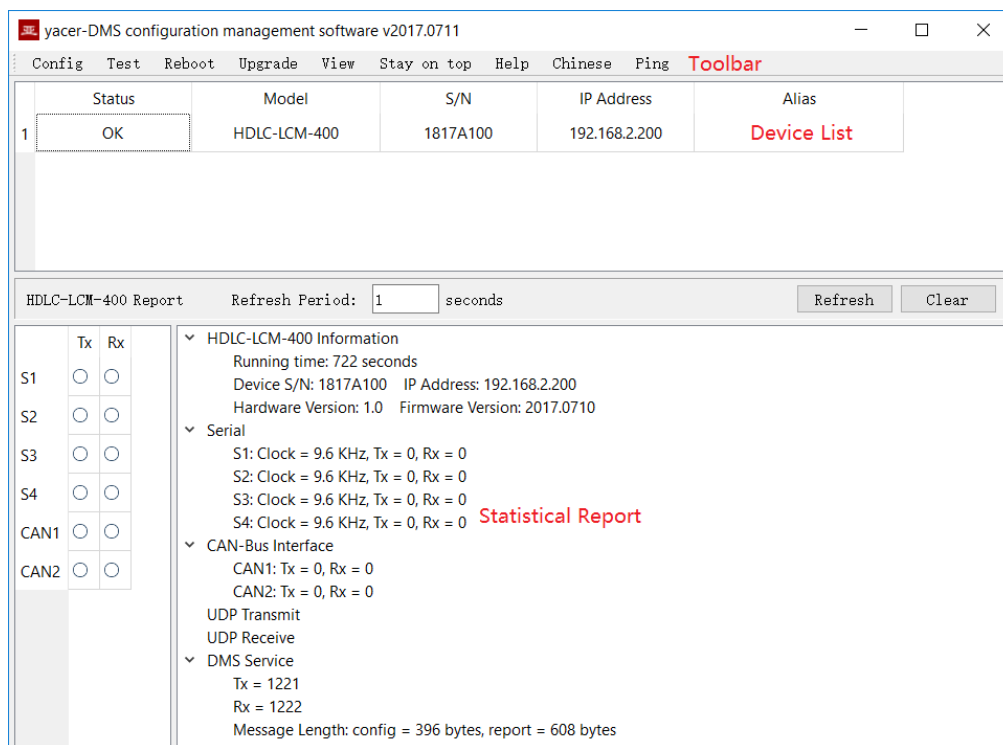
- Locate the yacer-DMS.exe file under directory and double-click on this file to run the configuration software.



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

4.4.1.1 HDLC-ATC-200, HDLC-ATC-400

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.

	Tx	Rx
S1	<input type="radio"/>	<input type="radio"/>
S2	<input type="radio"/>	<input type="radio"/>
S3	<input type="radio"/>	<input type="radio"/>
S4	<input type="radio"/>	<input type="radio"/>

4.4.1.2 HDLC-ATC-800, HDLC-ATC-1600

The transmit/receive status indication panel is located on the left side of the statistical report; the analog LED flashing indicates that waveform changes occur to the signal line of the corresponding serial port:

- TD: Transmit Data Line Signal Indication
- TC: Transmit Clock Line Signal Indication
- RD: Receive Data Line Signal Indication
- RC: Receive Clock Line Signal Indication

	TD	TC	RD	RC
S1				
S2				
S3				
S4				
S5				
S6				
S7				
S8				

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

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.

Config: HDLC-ATC-800/192.168.2.200 S/N 29175130

Ethernet Serial UDP-Serial Serial-UDP Serial-Serial

Device Alias:

☒ Enable ethernet switch

	IP Address	Subnet Mask	Default Gateway
ETH1	192.168.2.200	255.255.255.0	0.0.0.0

Save Load Restore Defaults Apply and Reboot Cancel

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

Button	Function
	Save the configuration content of the current dialog to the file
	Open the configuration file and read the refreshed configuration dialog content
	Refresh the dialog content with the device's default factory configuration
	Write the configuration content of the dialog into the device and restart the device to bring the configuration into effect
	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-ATC, thus adding description to the device or helping to remember the identification.

Ethernet
Serial
UDP-Serial
Serial-UDP
Serial-Serial

Device Alias:

5.1.2 IP address configuration

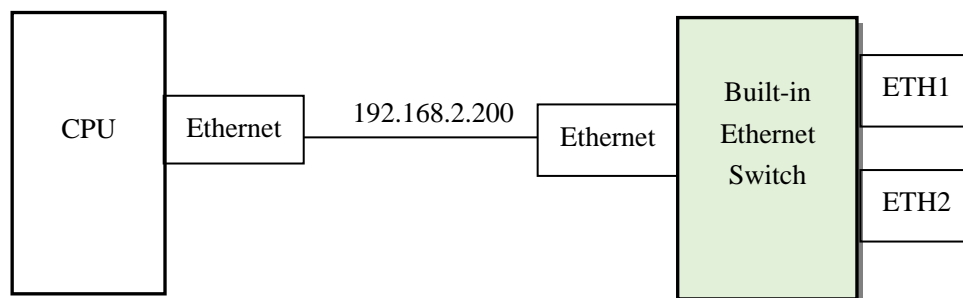
5.1.2.1 Ethernet switching enable

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

☒ Enable ethernet switch

	IP Address	Subnet Mask	Default Gateway
ETH1	192.168.2.200	255.255.255.0	0.0.0.0

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



5.1.2.2 Dual IP configuration

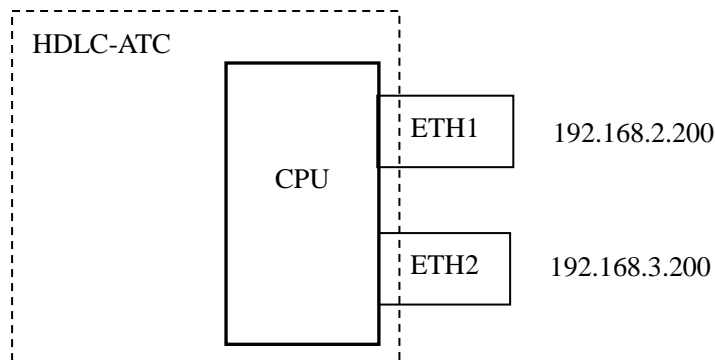
When the ☐ Enable ethernet switch checkbox is unchecked, ensure ETH1 and ETH2 are not on the same subnet for configuration as they have an independent IP address.

Device Alias:

☐ Enable ethernet switch ☐ IP Forward

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

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



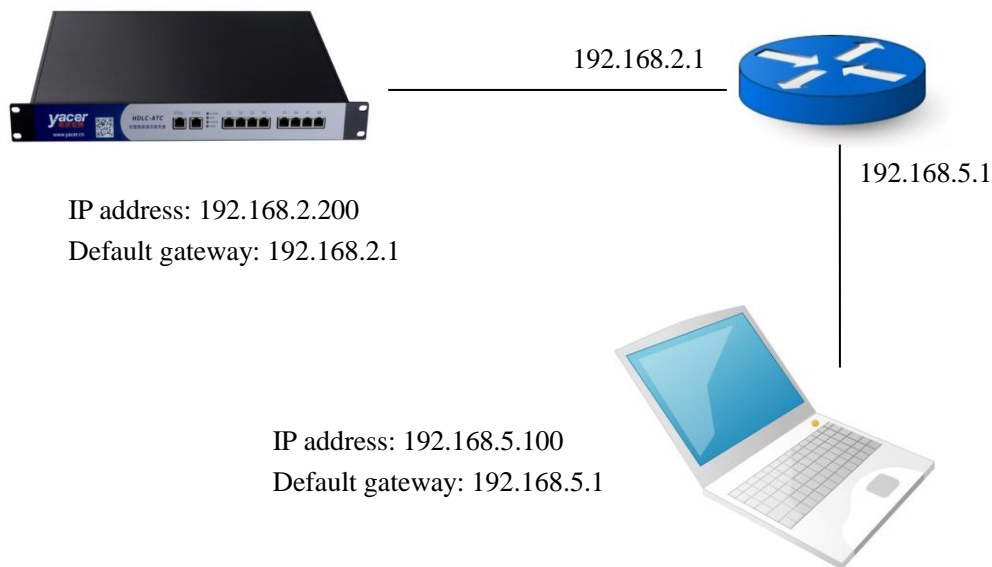
IP forwarding function is also provided for HDLC-ATC-800 and the higher models. After the checkbox ☒ IP Forward is selected, HDLC-ATC is able to enable the router function, providing the intersubnet IP forwarding function for the two LANs where ETH1 and ETH2 are located.

5.1.3 Default Gateway

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

If HDLC-ATC needs to communicate with the host on other subnet, it must rely on an external router. At this time, the HDLC-ATC'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-ATC 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-ATC 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-ATC 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 screenshot shows a configuration window titled "Config: HDLC-ATC-800/192.168.2.200 S/N 3617A186". It has tabs for "Ethernet", "Serial", "UDP-Serial", "Serial-UDP", and "Serial-Serial". The "Ethernet" tab is selected. Below the tabs, there is a "Device Alias:" field. A checkbox labeled "Enable ethernet switch" is checked. Below this is a table with columns "IP Address", "Subnet Mask", and "Default Gateway". The table has one row for "ETH1" with values "192.168.2.200", "255.255.255.0", and "0.0.0.0". Below the table is a red-bordered box containing the "Announce" settings: "Multicast Address: 224.8.8.8", "Destination Port: 8527", "Status Period: 0 seconds (1 - 255, 0 = no announce)", and "Statistics Period: 0 seconds (1 - 255, 0 = no announce)". At the bottom of the window are buttons for "Save", "Load", "Restore Defaults", "Apply and Reboot", and "Cancel".

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	Models Supported	Serial Ports Supported
Asynchronous	UART	Universal asynchronous serial, equivalent to the serial port on the common computer	All models	S1, S2
	UART-HDLC	UART-based similar HDLC communication protocol	All models	S1, S2
Synchronous	HDLC-NRZ	Synchronous HDLC protocol based on the NRZ encoding	All models	All serial ports
	Bit Stream	Serial Bit data based on the receive clock sampling	All models	All serial ports
	HDLC-NRZI	Synchronous HDLC protocol based on the NRZI encoding	HDLC-ATC-800 HDLC-ATC-1600	All serial ports
	HDLC-DBPL	Synchronous HDLC protocol based on the DBPL (Differential Bi-Phase-Level) encoding	HDLC-ATC-800 HDLC-ATC-1600	All serial ports
	HDLC-MAN	Synchronous HDLC protocol based on the Manchester encoding	HDLC-ATC-800 HDLC-ATC-1600	All serial ports
	HDLC-DiffMAN	Synchronous HDLC protocol based on the differential Manchester encoding	HDLC-ATC-800 HDLC-ATC-1600	All serial ports

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.

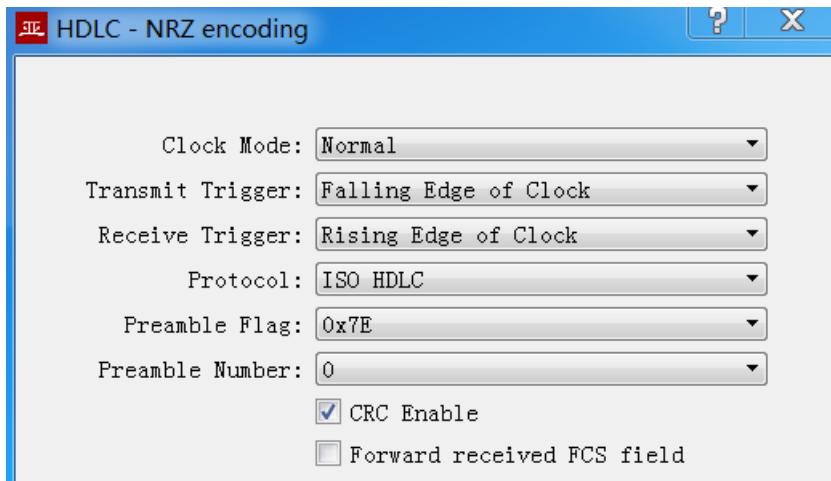
If further configuration of working parameters of the selected working mode is required, mouse double-click on the “Others” cell to pop up the parameter configuration dialog.

Config: HDLC-ATC-800/192.168.2.200 S/N 3617A186

	S1	S2	S3	S4	S5
Working Mode	UART	HDLC-NRZ	HDLC-NRZ	Bit Stream	HDLC-NRZ
Duplex	Full-Duplex	Full-Duplex	HDLC-NRZ BSC (BISYNC)	Full-Duplex	Full-Duplex
Baudrate (bps)	128000	115200	Bit Stream HDLC-NRZI	38400	19200
Others (Double-click)	Data Bits: 8 Parity Bits: None Stop Bits: 1 Packet Interval: 10ms Packet Length: 128bytes	Clock Mode: Normal Transmit Trigger: Falling Edge of Clock Receive Trigger: Rising Edge of Clock CRC: Enable Rx FCS: Discard Protocol: ISO HDLC Preamble Flag: 0x...	HDLC-DBPL HDLC-MAN HDLC-DiffMAN Falling Edge of Clock Receive Trigger: Rising Edge of Clock CRC: Enable Rx FCS: Discard Protocol: ISO HDLC Preamble Flag: 0x...	Clock Mode: Normal Transmit Trigger: Falling Edge of Clock Receive Trigger: Rising Edge of Clock Bit order: LSB First Packet Length: 128bytes	Clock Mode: Normal Transmit Trigger: Falling Edge of Clock Receive Trigger: Rising Edge of Clock CRC: Enable Rx FCS: Discard Protocol: ISO HDLC Preamble Flag: 0x...

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.



HDLC - NRZ encoding

Clock Mode: Normal

Transmit Trigger: Falling Edge of Clock

Receive Trigger: Rising Edge of Clock

Protocol: ISO HDLC

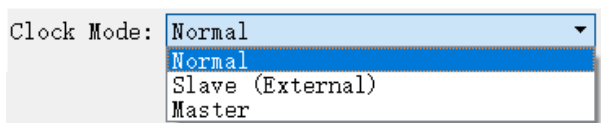
Preamble Flag: 0x7E

Preamble Number: 0

☒ CRC Enable

☐ Forward received FCS field

5.2.2.1 Clock mode



Clock Mode: Normal

Normal

Slave (External)

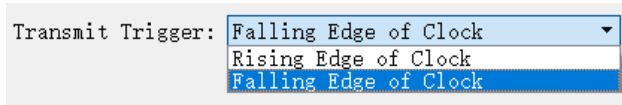
Master

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-ATC 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

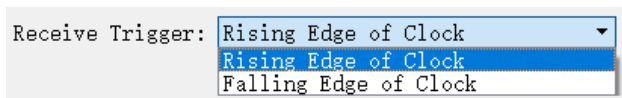


Transmit trigger defines the generation clock edge of the new data bit:

- 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:

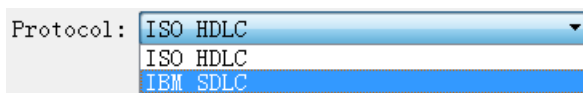
- Rising edge of clock: Data on the RxD line is read on the rising edge of the RxC signal
- Falling edge of clock: Data on the RxD line is read on the falling edge of the RxC signal

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

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-ATC 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-ATC 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.

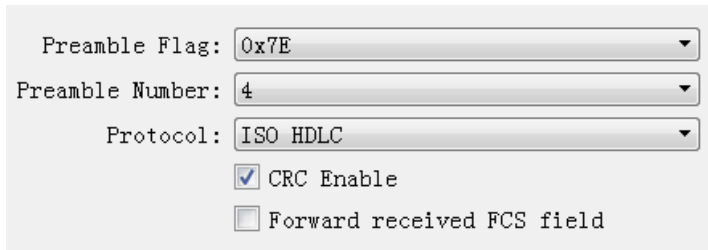
If this option is checked, then forward the user data and FCS field.

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 HDLC-NRZI/DBPL/MAN/MAN-Diff parameter configuration

Unlike the NRZ encoding format, the NRZI, DBPL, MAN and MAN-Diff encoding format data contains clock information, which only requires that the baud rate of the both communication sides should be the same, instead of the clock mode, transmit trigger, receive trigger and other parameters.

See the figure below for the configuration parameters of the working modes HDLC-NRZI, HDLC-DBPL, HDLC-MAN and HDLC-Diff with the parameter role and configuration mode identical to the the HDLC-NRZ working mode.



The image shows a configuration window for HDLC parameters. It includes three dropdown menus: 'Preamble Flag' set to '0x7E', 'Preamble Number' set to '4', and 'Protocol' set to 'ISO HDLC'. Below these are two checkboxes: 'CRC Enable' (checked) and 'Forward received FCS field' (unchecked).

Preamble Flag:	0x7E
Preamble Number:	4
Protocol:	ISO HDLC
<input checked="" type="checkbox"/> CRC Enable	
<input type="checkbox"/> Forward received FCS field	

5.2.4 Bit Stream parameter configuration

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.

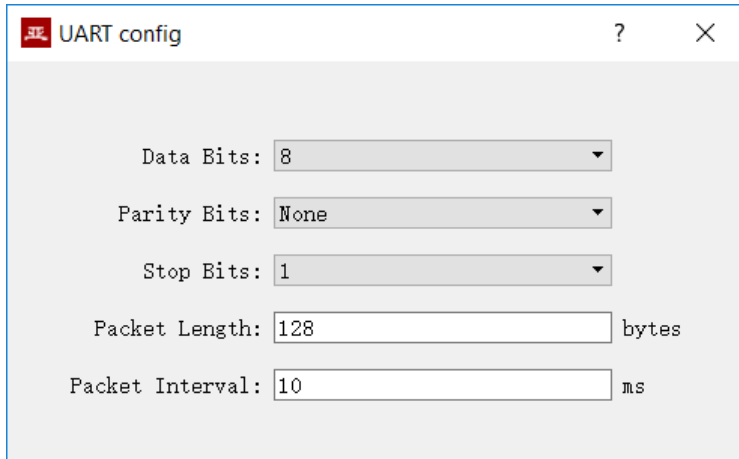
The online bit stream is stored in the computer or system memory in the form of byte. The receive/transmit sequence determines the conversion mode of byte and bit.

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

5.2.5 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 screenshot shows a window titled "UART config" with a red icon and standard window controls. It contains five configuration fields: "Data Bits" is a dropdown menu set to "8"; "Parity Bits" is a dropdown menu set to "None"; "Stop Bits" is a dropdown menu set to "1"; "Packet Length" is a text input field containing "128" followed by a "bytes" label; and "Packet Interval" is a text input field containing "10" followed by a "ms" label.

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 HDLC-ATC 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.

5.2.5.1 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.

5.2.5.2 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.

5.2.6 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 screenshot shows a window titled "UART-HDLC config" with the following settings:

- Data Bits: 8
- Parity Bits: None
- Stop Bits: 1
- ☒ CRC Enable
- ☐ Forward received FCS field
- Frame Flag: 0x7E
- 0x7E escape: 0x7D 0x5E
- 0x7D escape: 0x7D 0x5D

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:

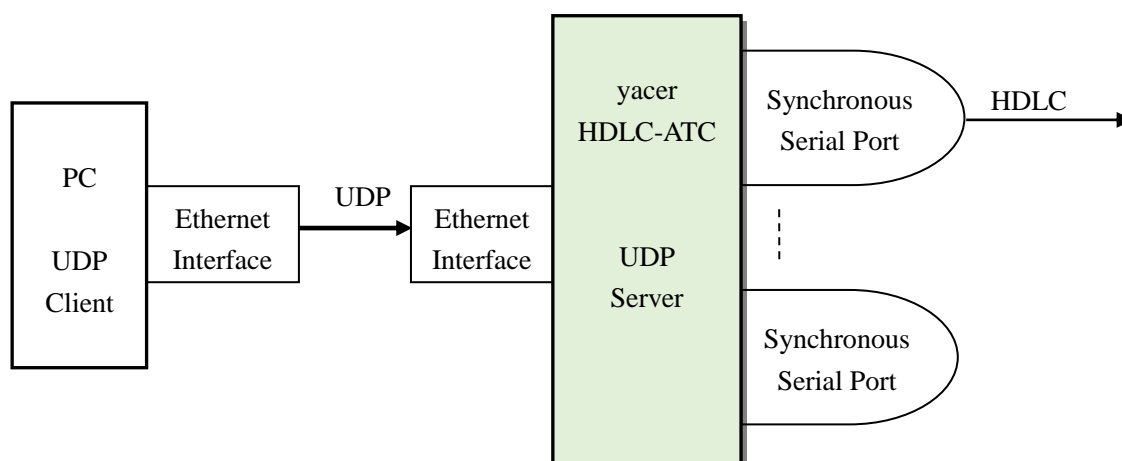
Actual Receive Data	Data
0x7D 0x5E	0x7E
0x7D 0x5D	0x7D
Others	No change

5.3 UDP to Serial Conversion

5.3.1 Application model

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

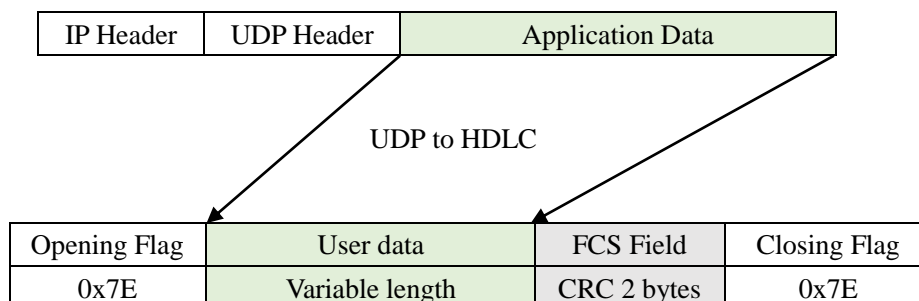
The typical application is shown as follows. PC sends a UDP message over the Ethernet interface as the UDP Client, and HDLC-ATC sends it out from the synchronous serial port after converting the received UDP message into the HDLC frame.



5.3.2 Protocol Conversion

The most typical UDP-to-HDLC application is shown below. HDLC-ATC 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-ATC 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.

Config: HDLC-ATC-800/192.168.2.200 S/N 29175130

Ethernet Serial **UDP-Serial** Serial-UDP Serial-Serial

	Ingress UDP Port	Forward to	Egress Serial
2	8002	enable	Serial-S2
3	8003	enable	Serial-S3
4	8004	enable	Serial-S4
5	8005	enable	Serial-S5
6	8006	enable	Serial-S6
7	8007	enable	Serial-S7
8	8008	enable	Serial-S8
9	0	disable	Serial-S1

Rx Multicast Address:

Group-1	0.0.0.0
Group-2	0.0.0.0
Group-3	0.0.0.0
Group-4	0.0.0.0
Group-5	0.0.0.0
Group-6	0.0.0.0
Group-7	0.0.0.0

(224. 0. 0. 0 - 239. 255. 255. 255)

The following configuration realizes the application, where data received from a UDP and distributed to 8 serial ports:

Config: HDLC-ATC-800/192.168.2.200 S/N 29175130

Ethernet Serial **UDP-Serial** Serial-UDP Serial-Serial

	Ingress UDP Port	Forward to	Egress Serial
2	8000	enable	Serial-S2
3	8000	enable	Serial-S3
4	8000	enable	Serial-S4
5	8000	enable	Serial-S5
6	8000	enable	Serial-S6
7	8000	enable	Serial-S7
8	8000	enable	Serial-S8
9	0	disable	Serial-S1

Rx Multicast Address:

Group-1	0.0.0.0
Group-2	0.0.0.0
Group-3	0.0.0.0
Group-4	0.0.0.0
Group-5	0.0.0.0
Group-6	0.0.0.0
Group-7	0.0.0.0

(224. 0. 0. 0 - 239. 255. 255. 255)

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.

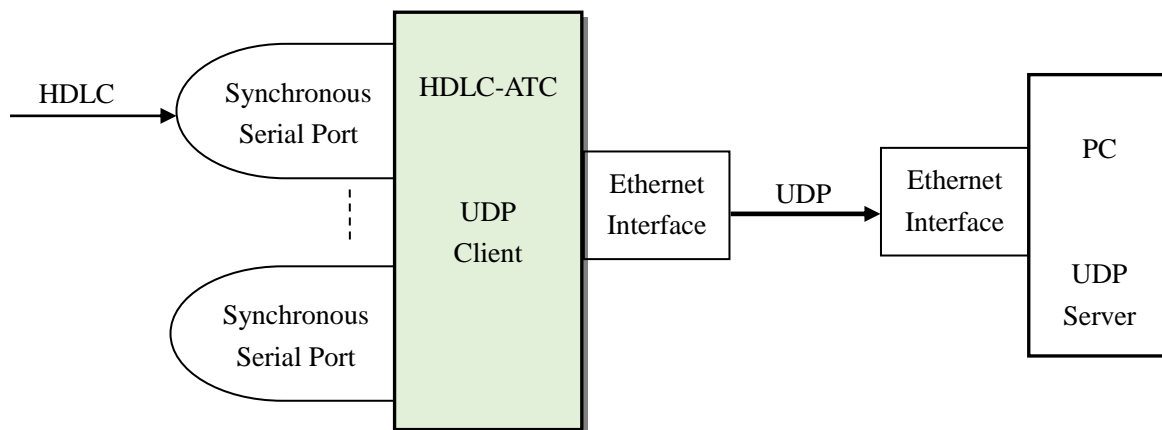
Range of the multicast address is 224.0.0.0 ~ 239.255.255.255, 224.8.8.8 is the configuration management address of the HDLC-ATC and users can't use this address.

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

5.4 Serial to UDP Conversion

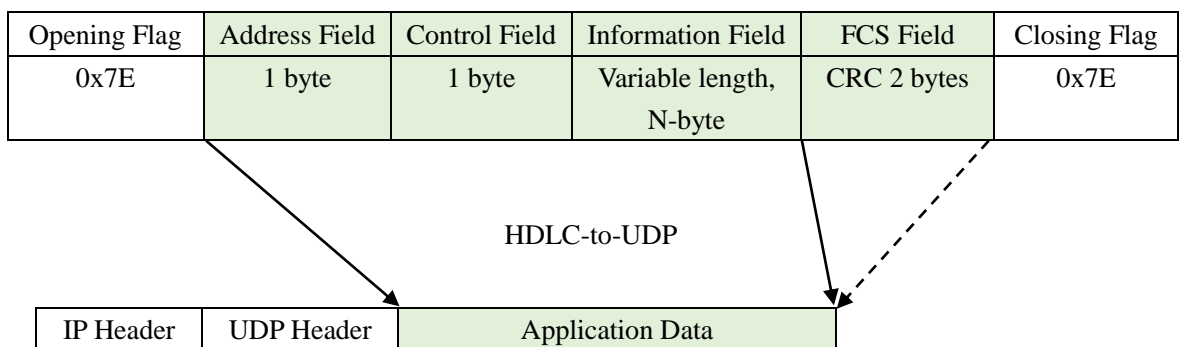
5.4.1 Application model

The HDLC-to-UDP function figure is shown below. HDLC-ATC 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-ATC 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	8000
2	Serial-S2	enable	255.255.255.255	9000
3	Serial-S3	enable	224.10.10.10	10000
4	Serial-S4	disable	0.0.0.0	0
5	Serial-S1	disable	0.0.0.0	0
6	Serial-S2	disable	0.0.0.0	0
7	Serial-S3	disable	0.0.0.0	0
8	Serial-S4	disable	0.0.0.0	0
9	Serial-S5	disable	0.0.0.0	0
10	Serial-S6	disable	0.0.0.0	0
11	Serial-S7	disable	0.0.0.0	0
12	Serial-S8	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.

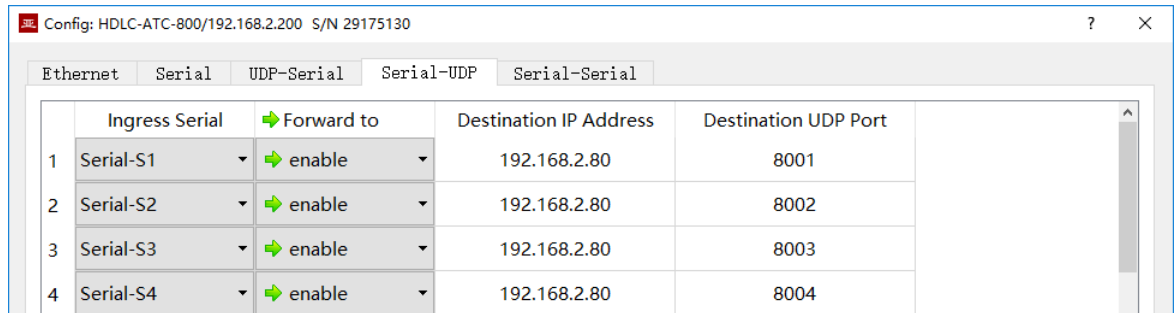
5.4.4 How does the UDP server identify the source serial

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 8004 is from the serial port S4.



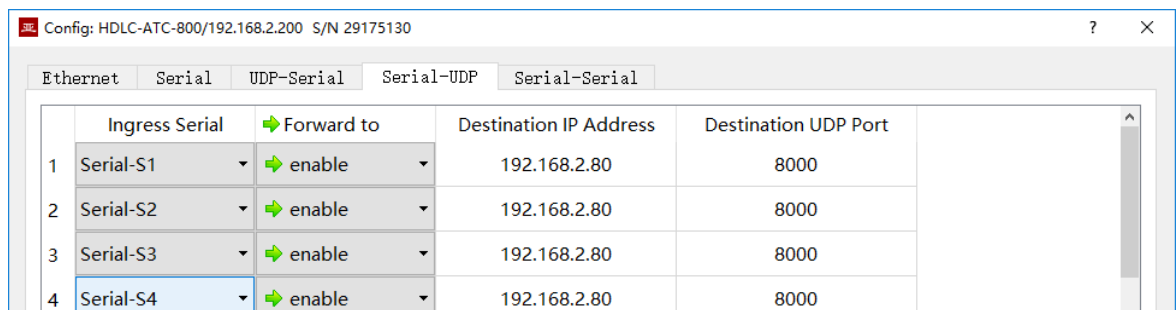
	Ingress Serial	Forward to	Destination IP Address	Destination UDP Port
1	Serial-S1	enable	192.168.2.80	8001
2	Serial-S2	enable	192.168.2.80	8002
3	Serial-S3	enable	192.168.2.80	8003
4	Serial-S4	enable	192.168.2.80	8004

5.4.4.2 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-ATC 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-ATCs are provided, UDP Server can distinguish the source device via the source IP.



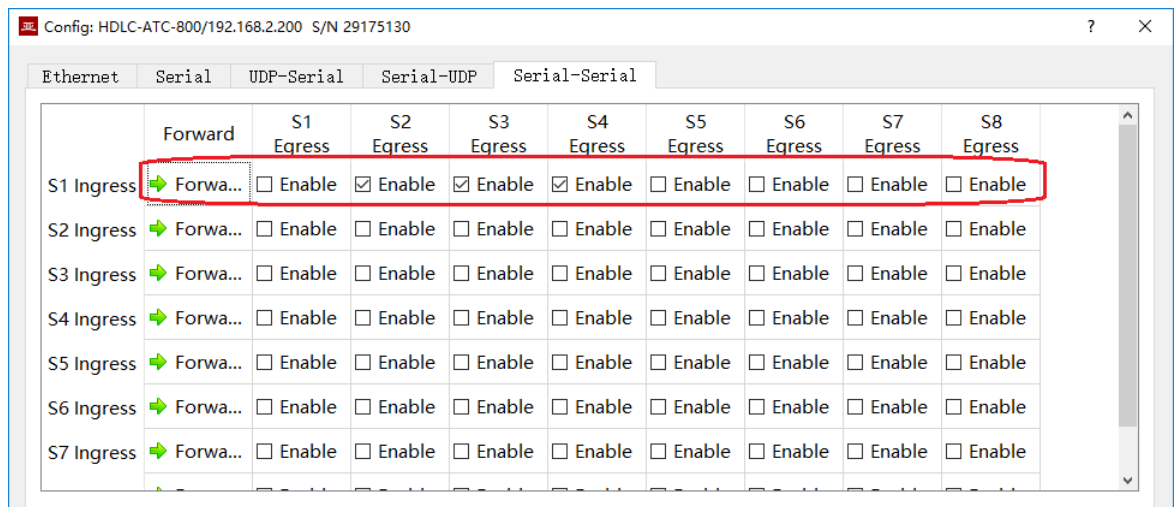
	Ingress Serial	Forward to	Destination IP Address	Destination UDP Port
1	Serial-S1	enable	192.168.2.80	8000
2	Serial-S2	enable	192.168.2.80	8000
3	Serial-S3	enable	192.168.2.80	8000
4	Serial-S4	enable	192.168.2.80	8000

5.5 Serial to Serial Conversion

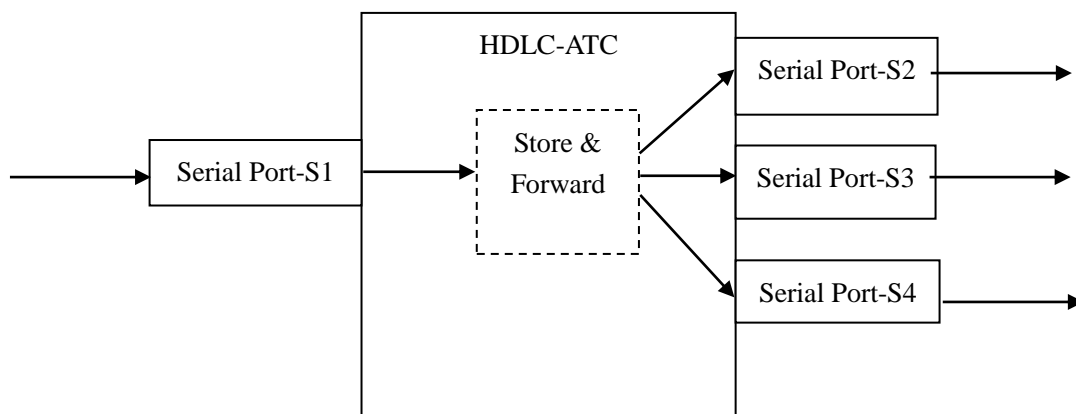
Serial-serial can forward the input data of the specified serial port to other serial port outputs, which is mainly used for:

- 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-ATC, the different baud rates and clock modes can be set for each-channel serial port, avoiding the packet loss caused by clock inconsistency.



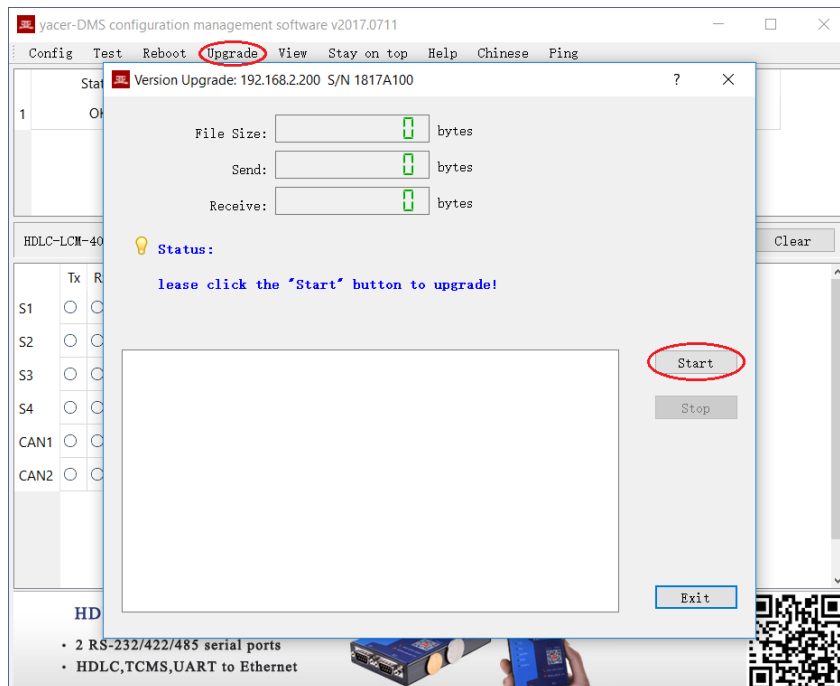
Configuration shown above realizes the application of the input demultiplexing of the serial port S1 to S2, S3 and S4 outputs. HDLC-ATC performs storing and forwarding on the received data. Even if the baud rate and clock mode of S1, S2, S3 and S4 are different, packet will not loss.



6 Firmware Version Upgrade

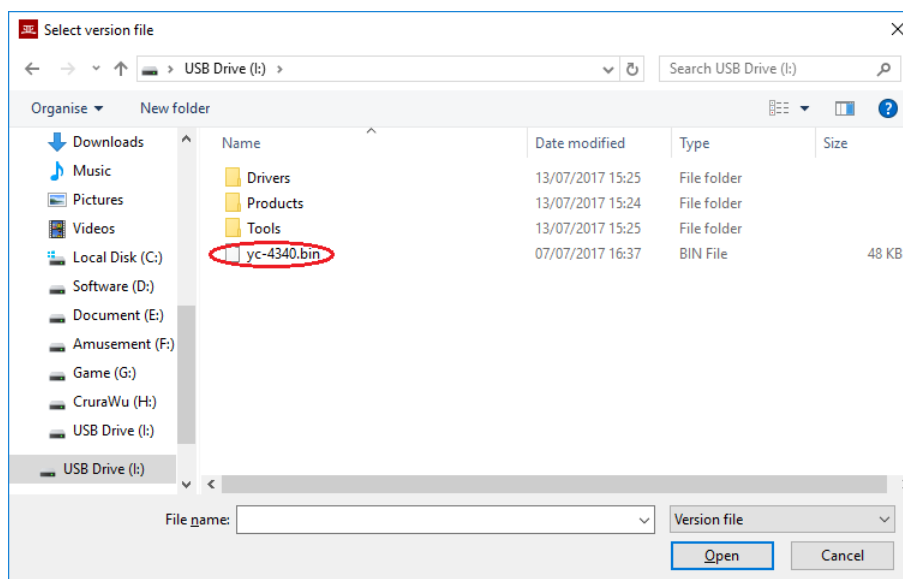
6.1 Start Upgrade

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



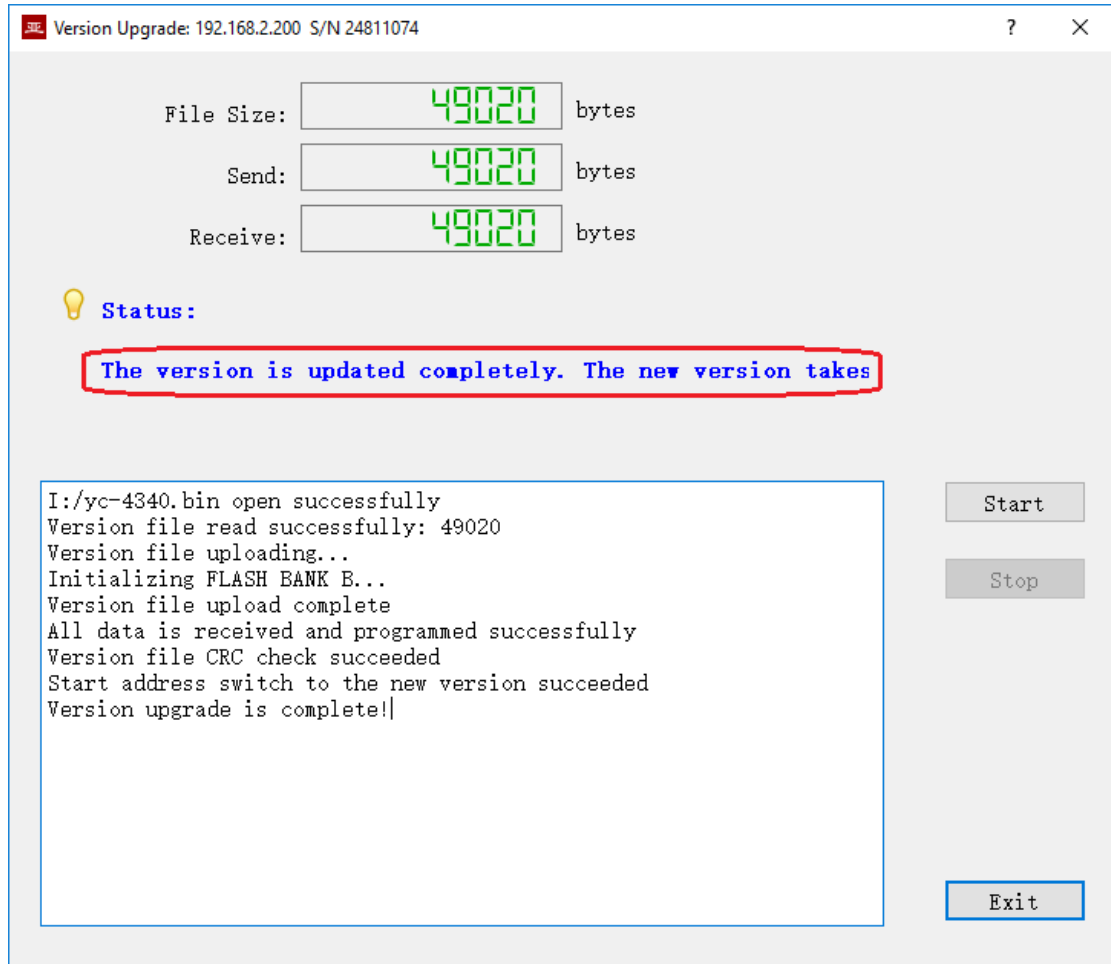
6.2 Locate Firmware Version

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



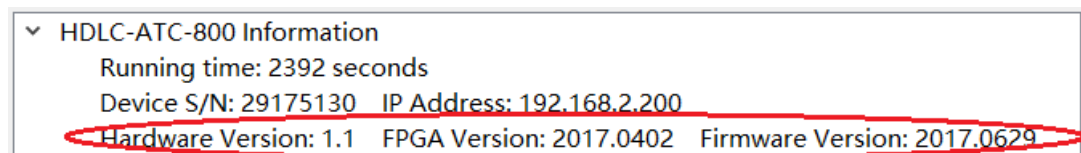
6.3 Upgrade Completed

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



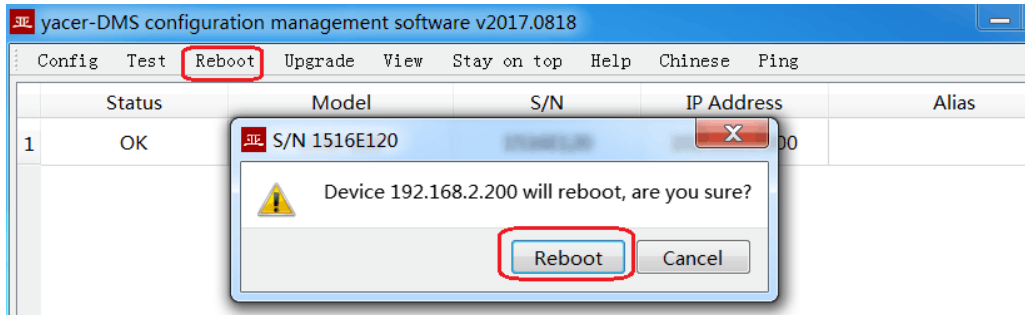
6.4 Upgrade Confirmation

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-ATC is normal.

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

