FMC 4-channel Gigabit Ethernet module FL2121 User manual

Rev 1.0

ALINX

Web-site:
Http://www.alinx.com.cn

Technical Forum:
http://www.heijin.org

Official flagship store:
http://alinx.jd.com

Email:
avic@alinx.com.cn

Tel:
021-67676997

Fax:
021-37737073

ALINX WeChat official account:

Document Revision Record:

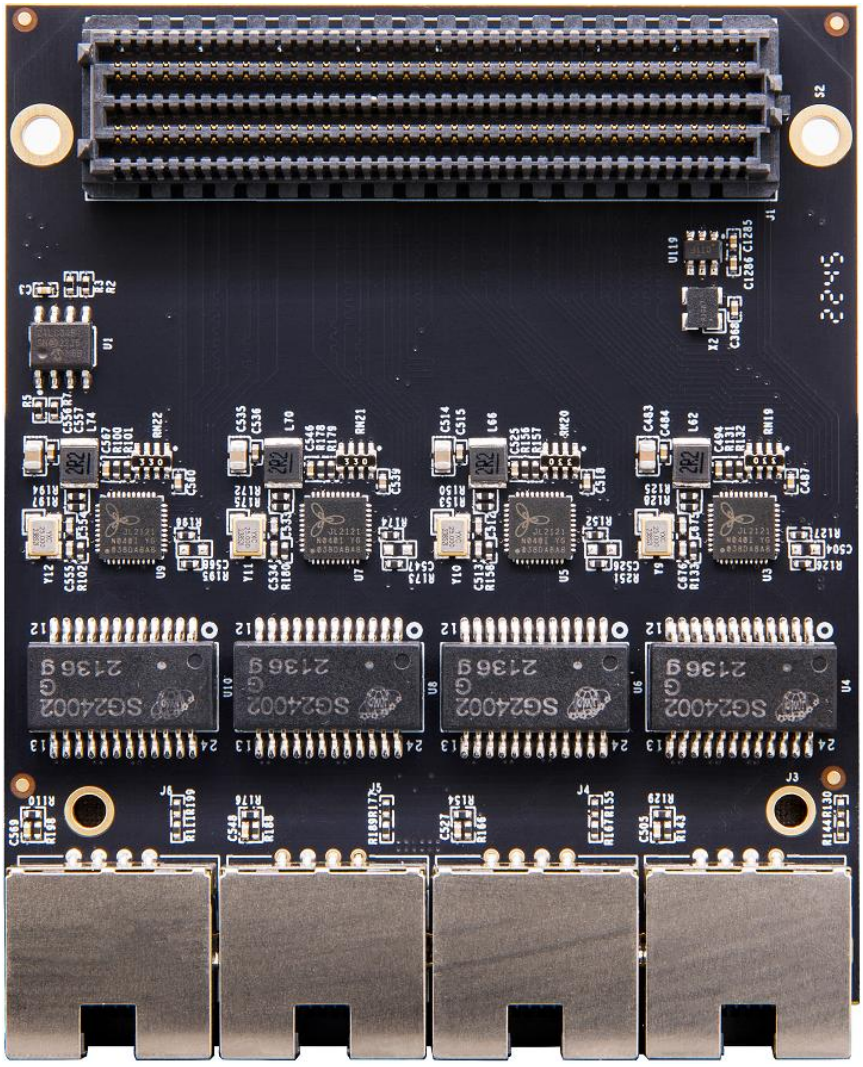
VERSION	TIME	Description
1.0	2022/11/20	First Release

Part 1: FMC High-Speed Ethernet Module Introduction

ALINX FMC Gigabit Ethernet module FL2121 is a 4-channel 10/100/1000Mbps self-adaptive Ethernet communication interface module. Its gigabit PHY chip uses four JL2121 Ethernet PHY chips from JLSemi company, supporting 10/100/1000 Mbps network transmission rates. Four-channel network interfaces use common RJ45 connectors to connect to and communicate with external networks.

The module has a standard LPC FMC interface for connecting to the FPGA development board, and the FMC connector model is ASP_134604_01.

The picture of module FL2121 is as follows:



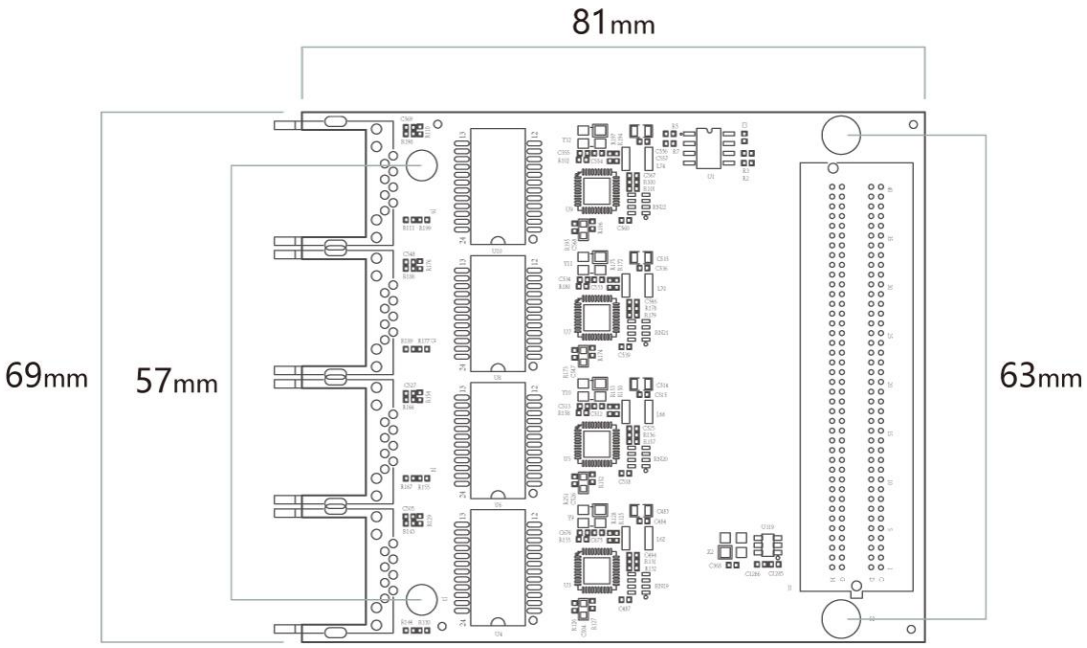
Picture of Module FL2121

1.1 Parameters of Module FL2121

The following are the detailed parameters of the module FL2121:

- Gigabit Ethernet chip: four JL2121
- Network interface: 4 RJ45;
- Ethernet communication rate: 10/100/1000 Mbps;
- Configuration interface: MDIO interface;
- Operating temperature: -40°~85°;

1.2 Structure diagram of Module FL2121

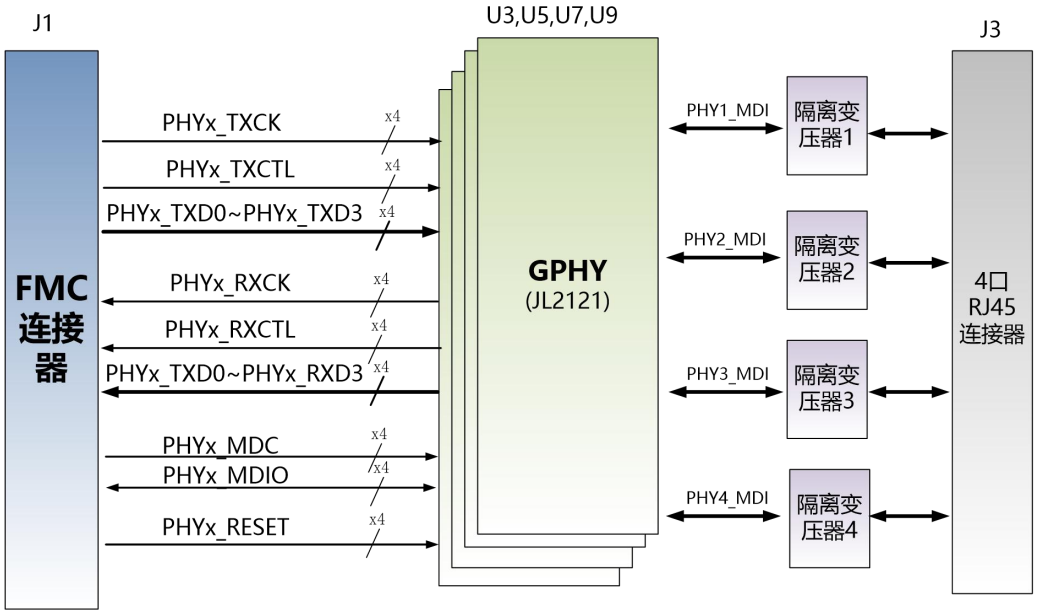


Structure diagram of Gigabit Ethernet module FL2121

Part 2: Function Description of Module FL2121

2.1 Schematic Diagram of Module FL2121

The schematic diagram of module FL2121 is as follows:



2.2 Ethernet Chip

Ethernet chip adopts industrial Ethernet GPHY chip (JL2121-N040I) from JLSemi to provide users with network communication services. The Ethernet PHY chip on the PS side is connected to the MIO interface of the BANK502 on the PS side on ZYNQ, while the Ethernet PHY chip on the PL side is connected to the IO of BANK66. The JL2121 chip supports 10/100/1000 Mbps network transmission rate and communicates with the MAC layer of MPSOC system through the RGMII interface. JL2121 supports MDI/MDX self-adaptive, various speed self-adaptive, Master/Slave self-adaptive, and supports MDIO bus for PHY register management.

When the JL2121 is powered on, it will detect some specific IO level states to determine its own working mode. The following table describes the default settings of the GPHY chip after it is powered on.

Configure Pin	Description	Configuration Value
RXD3_ADR0 RXC_ADR1 RXCTL_ADR2	PHY address in MDIO/MDC mode	PHY Address is 001
RXD1_TXDLY	TX Clock 2ns delay	Delay
RXD0_RXDLY	RX Clock 2ns delay	Delay

Default configuration value of GPHY chip

When the network is connected to Gigabit Ethernet, the MPSOC and

PHY chip JL2121 communicate data transmission over the RGMII bus with a transmission clock of 125Mhz, and the data is sampled on the rising edge and falling edge of the clock.

When the network is connected to 100 Gigabit Ethernet, the MPSOC and PHY chip JL2121 communicate data transmission over the RMII bus with a transmission clock of 25Mhz. Data is sampled on the rising edge and falling edge of the clock.

2.3 Pin Assignment of Module FMC LPC:

Only the signals of the power supply and network chip interfaces are listed below, while the signals of GND are not listed. Users can refer to the schematic diagram for details.

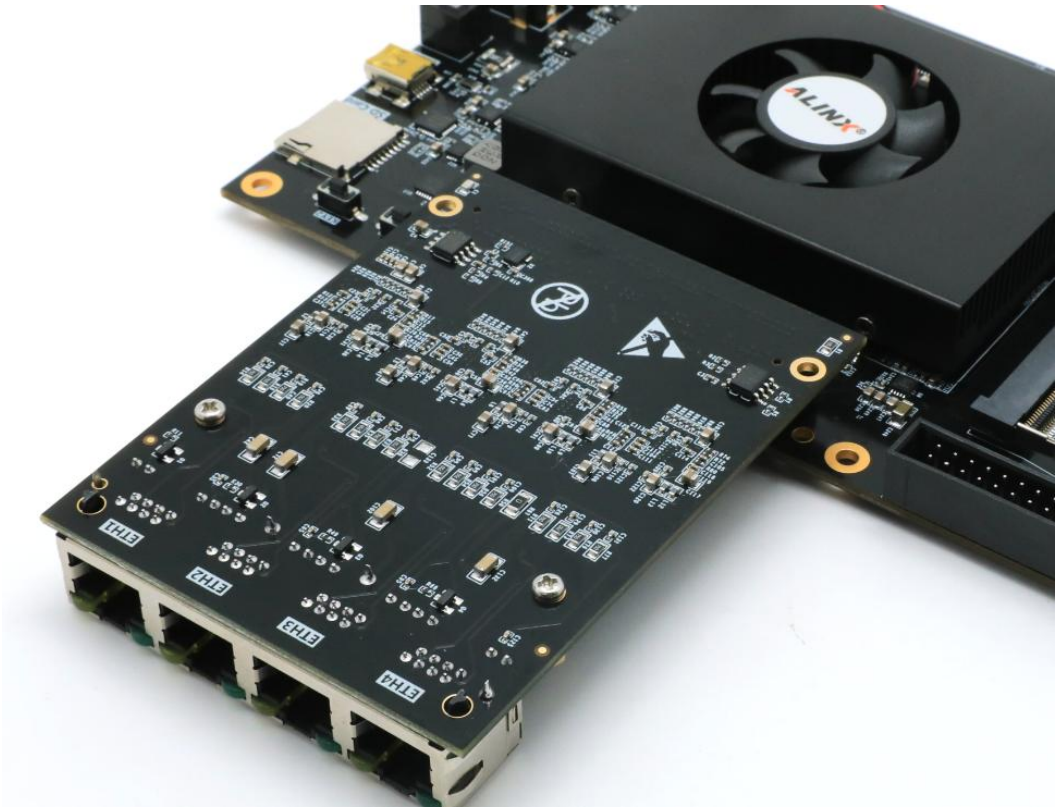
Pin Number	Signal Name	Description
C35	+12V	12V power input
C37	+12V	12V power input
D32	+3.3V	3.3V power input
C34	GA0	EEPROM address bit 0 bit
D35	GA1	EEPROM address bit 1 bit
D11	PHY1_MDC	Ethernet 1st-channel MDIO management clock
C11	PHY1_MDIO	Ethernet 1st-channel MDIO management data
D12	PHY1_RESET	Ethernet 1st-channel reset signal
G6	PHY1_RXCK	Ethernet 1st-channel RGMII receiving clock
G7	PHY1_RXCTL	Ethernet 1st-channel receives data valid signal
H4	REFCLK	50MHz reference clock
H7	PHY1_RXD0	Ethernet 1st-channel receives data Bit 0
H8	PHY1_RXD1	Ethernet 1st-channel receives data Bit 1
G9	PHY1_RXD2	Ethernet 1st-channel receives data Bit 2
G10	PHY1_RXD3	Ethernet 1st-channel receives data Bit 3
H11	PHY1_TXCK	Ethernet 1st-channel RGMII transmitting clock
H14	PHY1_TXCTL	Ethernet 1st-channel transmits data valid signal
H10	PHY1_TXD0	Ethernet 1st-channel transmits data Bit 0
G12	PHY1_TXD1	Ethernet 1st-channel transmits data Bit 1
G13	PHY1_TXD2	Ethernet 1st-channel transmits data Bit 2
H13	PHY1_TXD3	Ethernet 1st-channel transmits data Bit 3
D18	PHY2_MDC	Ethernet 2nd-channel MDIO management clock
C19	PHY2_MDIO	Ethernet 2nd-channel MDIO management data
H20	PHY2_RESET	Ethernet 2nd-channel reset signal
D8	PHY2_RXCK	Ethernet 2nd-channel RGMII receiving clock

D9	PHY2_RXCTL	Ethernet 2nd-channel receives data valid signal
C10	PHY2_RXD0	Ethernet 2nd-channel receives data Bit 0
D14	PHY2_RXD1	Ethernet 2nd-channel receives data Bit 1
C15	PHY2_RXD2	Ethernet 2nd-channel receives data Bit 2
D15	PHY2_RXD3	Ethernet 2nd-channel receives data Bit 3
H17	PHY2_TXCK	Ethernet 2nd-channel RGMII transmitting clock
H19	PHY2_TXCTL	Ethernet 2nd-channel transmits data valid signal
G16	PHY2_TXD0	Ethernet 2nd-channel transmits data Bit 0
H16	PHY2_TXD1	Ethernet 2nd-channel transmits data Bit 1
G18	PHY2_TXD2	Ethernet 2nd-channel transmits data Bit 2
G19	PHY2_TXD3	Ethernet 2nd-channel transmits data Bit 3
H28	PHY3_MDC	Ethernet 3rd-channel MDIO management clock
G28	PHY3_MDIO	Ethernet 3rd-channel MDIO management data
H29	PHY3_RESET	Ethernet 3rd-channel reset signal
D20	PHY3_RXCK	Ethernet 3rd-channel RGMII receiving clock
G21	PHY3_RXCTL	Ethernet 3rd-channel data valid signal
G22	PHY3_RXD0	Ethernet 3rd-channel receives data Bit 0
H22	PHY3_RXD1	Ethernet 3rd-channel receives data Bit 1
D23	PHY3_RXD2	Ethernet 3rd-channel receives data Bit 2
D24	PHY3_RXD3	Ethernet 3rd-channel receives data Bit 3
H25	PHY3_TXD3	Ethernet 3rd-channel transmits data Bit 3
G27	PHY3_TXCTL	Ethernet 3rd-channel transmits data valid signal
H23	PHY3_TXD0	Ethernet 3rd-channel transmits data Bit 0
G24	PHY3_TXD1	Ethernet 3rd-channel transmits data Bit 1
G25	PHY3_TXD2	Ethernet 3rd-channel transmits data Bit 2
H26	PHY3_TXCK	Ethernet 3rd-channel RGMII transmitting clock
H35	PHY4_MDC	Ethernet 4th-channel MDIO management clock
H37	PHY4_MDIO	Ethernet 4th -channel MDIO management data
H38	PHY4_RESET	Ethernet 4th-channel reset signal
C22	PHY4_RXCK	Ethernet 4th-channel RGMII receiving clock
C23	PHY4_RXCTL	Ethernet 4th-channel receives data valid signal
D26	PHY4_RXD0	Ethernet 4th-channel receives data Bit 0
C26	PHY4_RXD1	Ethernet 4th-channel receives data Bit 1
D27	PHY4_RXD2	Ethernet 4th-channel receives data Bit 2
C27	PHY4_RXD3	Ethernet 4th-channel receives data Bit 3
H32	PHY4_TXCK	Ethernet 4th-channel RGMII transmitting clock
H34	PHY4_TXCTL	Ethernet 4th-channel transmits data valid signal
G31	PHY4_TXD0	Ethernet 4th-channel transmits data Bit 0
H31	PHY4_TXD1	Ethernet 4th-channel transmits data Bit 1
G33	PHY4_TXD2	Ethernet 4th-channel transmits data Bit 2
G34	PHY4_TXD3	Ethernet 4th-channel transmits data Bit 3
C30	SCL	EEPROM's I2C clock
C31	SDA	EEPROM's I2C data

G39	VADJ	VADJ power input
H40	VADJ	VADJ power input

Part 3: Hardware Connection and Testing

The hardware connection between the module FL2121 and the FPGA development board is very simple, as long as the FMC interface and the FMC interface of the development board can be inserted, and then you can fix them with screws. The following is the hardware connection diagram of the ALINX AX7325 development board and the module FL2121:



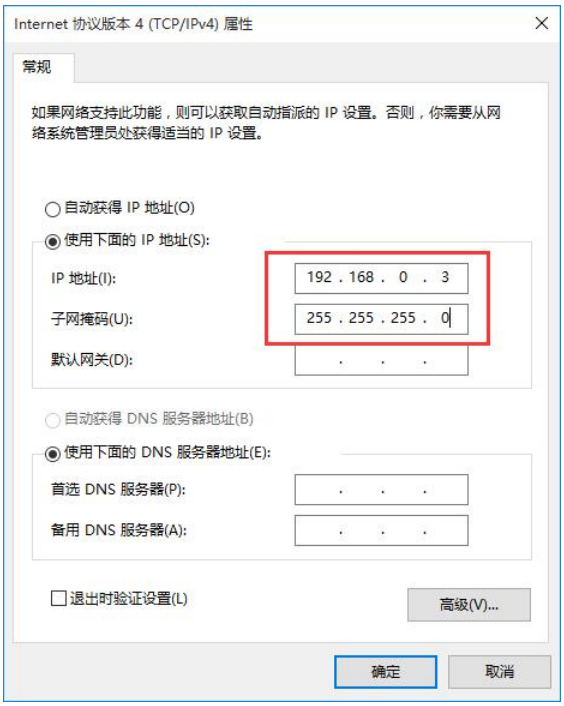
➢ Preparations

First: Confirm whether the network card of your PC is a gigabit network card. Users can click the local connection to check it, and then connect the network port of the development board and the network port of the PC with category 5+ or category 6 network

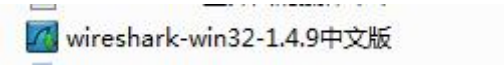
cable.

Second: Change the IP address of the PC to 192.168.0.3. The IP Address of the PC must be the same as that set in mac_test.v in the program, otherwise the network debugging assistant will not receive the UDP packets sent by the development board.

```
.source_mac_addr          (48'h00_0a_35_01_fe_c0)
.TTL                      (8'h80),
.source_ip_addr           (32'hc0a80002),
.destination_ip_addr      (32'hc0a80003),
.udp_send_source_port     (16'h1f90),
.udp_send_destination_port (16'h1f90),
```



Third (selectable): The Wireshark is installed to facilitate network communication debugging. Users can install Wireshark, a network packet capture tool in the TOOL directory of the CD, which can be used to view details about data sent and received through the network port of a PC during experiments.

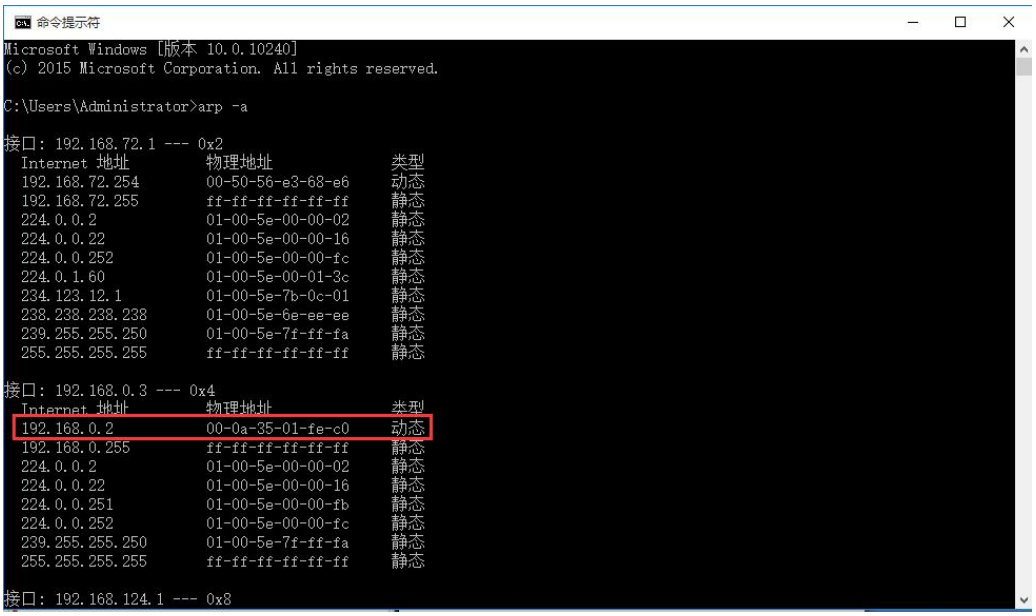


➢ Ethernet Communication Test

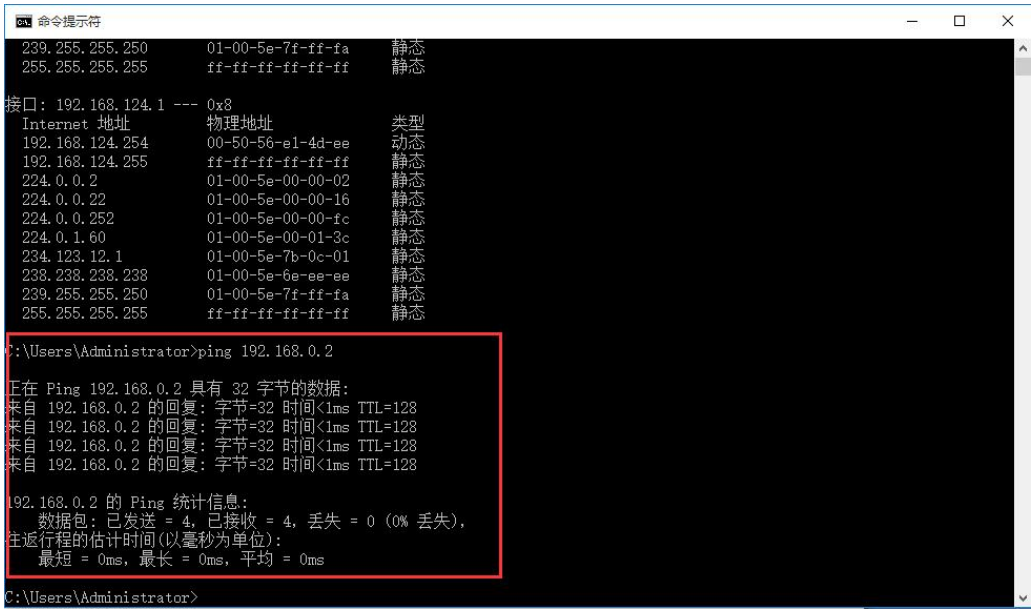
First: Program the bit file to the FPGA chip.

Second: Press KEY2 on the development board, open the CMD window,

and enter arp -a to view the ARP binding result. You can see that the IP and MAC addresses of the development board are cached.

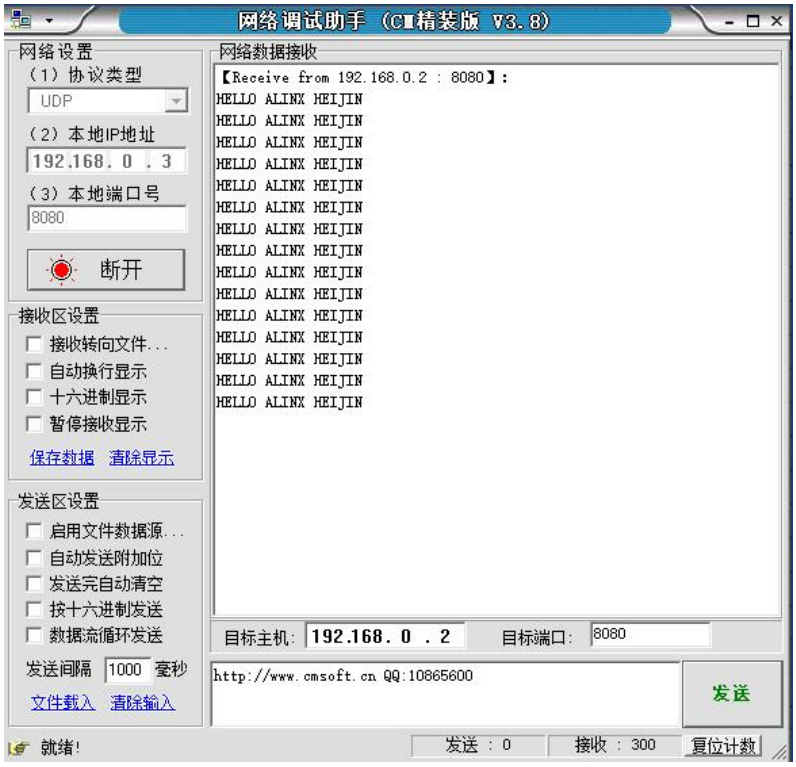


Third: In the CMD window, enter the ping 192.168.0.2 to check whether the PC can ping the development board.



Fourth: Open the network debugging assistant in the TOOL directory and set the following parameters, and then press the Connect button (the local IP address here is the IP Address of the PC, and the local port must be the same as that in the FPGA program, which is 8080).

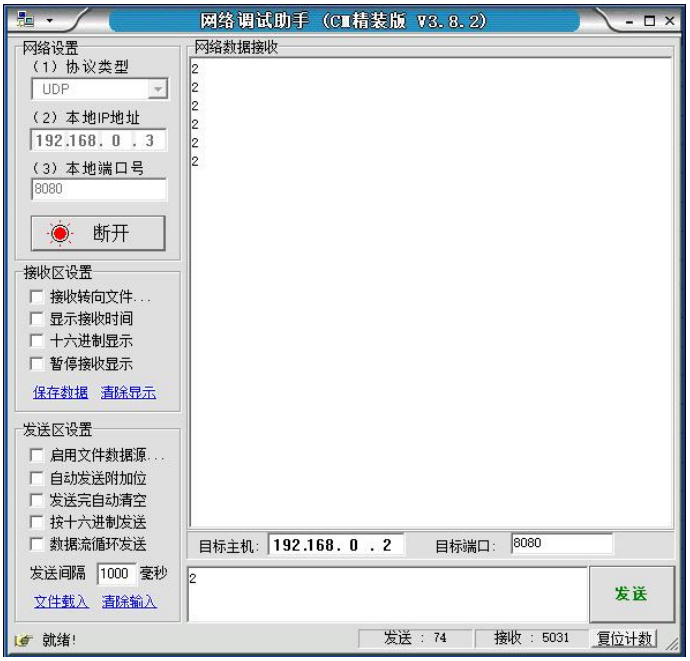
At this time, the network data receiving window will display the Ethernet packet "Hello ALINX HEIJIN" sent by the FPGA to the PC. The IP address of the target host must be consistent with the IP address in the FPGA program, and the target port number must be consistent with that in the FPGA program (8080). Network display as shown in the following figure:



Fifth: Then send a large string of characters in the send window of the network debugging assistant. In the data receiving window of the network,

we can see that the data returned from the FPGA has also become the string just sent.

It is also possible to send fewer characters, less than 46 bytes, and the FPGA program will automatically supplement to 46 bytes, as shown below:



Sixth: This step is optional for users. To view more information about data packets transmitted, use the Wireshark, a network packet capture tool, to view the network data received and sent by the PC network adapter.