

TCP/IP communication based on Ethernet

User manual

WUXI XINJE ELECTRIC CO., LTD.

Data No. PD07 20240223EN 1.4.3

• Basic explanation

Thank you for purchasing Xinje Ethernet PLC.

This manual mainly introduces Ethernet function of PLC.

Please read this manual carefully before using and wire after understanding the content.

About software and programming instructions, please refer to related manuals.

Please hand this manual over to operation users.

• Notices for users

Only experienced operator can wire the plc. If any problem, please contact our technical department.

The listed examples are used to help users to understand, so it may not act.

Please confirm that PLC specifications and principles are suitable when connect PLC to other products. Please conform safety of PLC and machines by yourself when use the PLC. Machines may be damaged by PLC errors.

• Responsibility declaration

The manual content has been checked carefully, however, mistakes may happen.

We often check the manual and will correct the problems in subsequent version. Welcome to offer advices to us.

Excuse us that we will not inform you if manual is changed.

• Contact information

If you have any problem about products, please contact the agent or Xinje company.

Tel: 0086 510-85134136 85123803

Fax: 0086 510-85111290

Address: No. 816 Jianzhu West Road, Binhu District, Wuxi City, Jiangsu Province Code : 214072

WUXI XINJE ELECTRIC CO., LTD. copyrights

Do not copy or use manual without written permission. Offenders should be responsible for losses. Please keep all copyrights of our company including practical modules, designed patents and copyrights mentioned in register.

2018, 11, 12

CATALOG

1	ETHERNET COMMUNICATION OVERVIEW	3
	1-1. The basic concept of Ethernet	3
	1-1-1. IP allocation	3
	1-1-2. PC network address	3
	1-1-3. PING command	5
	1-2. TCP IP PROTOCOL	7
	1-2-1. Port number	7
	1-2-2. UDP protocol	7
	1-2-3. TCP protocol	7
2	ETHERNET PARAMETERS	10
	2-1. ETHERNET PARAMETERS	10
	2-1-1. IP address parameters	10
	2-1-2. Function specification	11
	2-2. CONFIGURE THE ETHERNET PARAMETERS IN THE SOFTWARE	12
	2-3. CONFIGURE ETHERNET PARAMETERS IN XINJECONFIG	13
3	WIRING AND COMMUNICATION PROTOCOL	16
	3-1. WIRING MODE	16
	3-2. MODBUS TCP PROTOCOL	16
	3-2-1. MODBUS TCP overview	16
	3-2-2. MODBUS address	17
	3-2-3. MODBUS function code	34
	3-3. MODBUS TCP GRAPHICS CONFIGURATION	34
	3-3-1. Overview	34
	3-3-2. Modbus TCP master station configuration	34
	3-3-3. Modbus TCP graphical application	38
	3-4. Free Format Protocol	39
4	ETHERNET COMMUNICATION INSTRUCTION	41
	4-1. ETHERNET COMMUNICATION INSTRUCTION OVERVIEW	41
	4-1-1. Create TCP connection/UDP port listening [S OPEN]	41
	4-1-2. Communication termination [S CLOSE]	45
	4-1-3. Free format communication send [S_SEND]	46
	4-1-4. Free format communication receive [S_RCV]	48
	4-1-5. MODBUS communication [M_TCP]	49
	4-1-6. Ethernet communication example	52
	4-2. READ WRITE COMMUNICATION PORT PARAMETERS	72
	4-2-1. Read serial port parameters [CFGCR]	72
	4-2-2. Write serial port parameters [CFGCW]	73
	4-2-3. Set the IP address [IPSET]	74
	4-2-4. Serial port parameter name and setting	76

4-2-5. Read write port parameter application	
4-3. ETHERNET COMMUNICATION FLAG AND REGISTER	79
4-4. ETHERNET COMMUNICATION ERROR LIST	80
5. ETHERNET/IP COMMUNICATION	82
5-1. Ethernet/IP overview	
5-2. Ethernet/IP nouns overview	83
5-3. ETHERNET/IP COMMUNICATION SPECIFICATION	83
5-3-1. Implicit function specification	83
5-3-2. Explicit function specification	
5-3-3. Client and server support variable types	
5-4. ETHERNET/IP EXPLICIT/IMPLICIT COMMUNICATION	
5-4-1. Implicit function	
5-4-1-1. Adapter configuration	
5-4-1-2. Scanner configuration	
5-4-1-3. Implicit communication application	93
Application 1: Implicit communication by using two XDH-60T-E PLCs with InputOnly conne	ction
type	93
Application 2: Implicit communication by using two XDH-60T-E PLCs with ExclusiveOwner	,
connection type	95
Application 3: Implicit communication by using three XDH-60T-E PLCs with ListenOnly con	nection
type	97
Application 4: Using Xinje PLC XDH-60T-E as an adapter and Omron PLC NJ501-1500 as a	
scanner for implicit communication.	100
Application 5: Use Xinje PLC XDH-60T-E as the scanner and Omron PLC NJ501-1500 as the	;
adapter for implicit communication	107
Application 6: Use Xinje PLC XDH-60T-E as the adapter and Keyence PLC KV-5500 as the s	scanner
for implicit communication.	111
Application 7: Use Xinje PLC XDH-60T-E as the scanner and Keyence PLC KV-5500 as the	adapter
for implicit communication.	116
Application 8: Implicit communication between Xinje PLC XDH-60T4-E as adapter and Xinje	e PLC
XSDH-60A32-E as scanner	120
Application 9: Using Xinje PLC XDH-60T4-E as a scanner and Xinje PLC XSDH-60A32-E a	s an
adapter for implicit communication	126
5-4-2. Explicit communication	130
5-4-2-1. Explicit server	130
5-4-2-2. Explicit client	131
5-4-2-3. Application	133
Application 1: Two Xinje XDH-60T-E for explicit label communication	133
Application 2: Use Xinje XDH-60T-E as the client and Omron NJ501-1500 as the server for e	xplicit
tag communication.	136
APPENDIX	139

1 Ethernet communication overview

1-1. The basic concept of Ethernet

Before the Ethernet communication, let's understand some Ethernet concepts such as IP address allocation, PC network address and settings.

1-1-1. IP allocation

If programmable devices (such as PC) using LAN network card to connect to the factory (or the Internet), the PLC and programming device must be in the same subnet. Combination of IP address and subnet mask can be specified subnet of the equipment.

Network ID is the IP address of the first part, the top three 8-bit groups (such as IP addresses for 211.154.184.16, 211.154.184 represents network ID) decided the user's IP network. The value of the subnet mask is usually 255.255.255.0. However, because of your computer is in the local area network (LAN), subnet mask (for example, 255.255.254.0) may have different values to set the unique subnet. Subnet mask and the equipment IP address will do logic AND operation to define the boundary of the IP subnet.

1-1-2. PC network address

Please check your programming device IP address as the following steps.

1. Open the network and sharing center:



eneral		Access type: Internet
Connection IPv4 Connectivity: IPv6 Connectivity:	Internet No Internet access	HomeGroup: Available to join Connections: Ethernet
Media State: Duration: Speed:	Enabled 03:06:57 100.0 Mbps	on; or set up a router or access point.
Details		t troubleshooting information.
Sent —	— Received	
Bytes: 37,397,155	106,854,928	

- 2. Click the Ethernet connections, choose properties:
- E control and rectrol and internet rectrol and ording control

3. Set the PC IP address, make it in the same subnet.

For example, the PLC IP is 192.168.2.1, the PC IP is set to 192.168.2.200, the subnet mask is 255.255.255.0. default gateway can be vacant. Then the PC can connect to the CPU.

Networking Sharin	9		General Alternate Configuration	n
Connect using:	e GBE Family Controller #	2	You can get IP settings assigned this capability. Otherwise, you for the appropriate IP settings	ed automatically if your network supports need to ask your network administrator
This connection us	es the following items:	Configure	Obtain an IP address auto	omatically
Gos Pack Microsoft Microsoft Microsoft Link-Laye Link-Laye Internet P	xet Scheduler Network Adapter Multipleo LLDP Protocol Driver r Topology Discovery Map r Topology Discovery Res rotocol Version 6 (TCP/IP	vor Protocol oper I/O Driver ponder v6)	D Use the following IP address: D Subnet mask: Default gateway:	
<	rotocol Version 4 (TCP/IP	v4)	Obtain DNS server addres Obtain DNS server addres Obtain DNS server addres	ss automatically rver addresses:
Install	Uninstall	Properties	Preferred DNS server:	221.228.255.1
Description Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.			Alternate DNS server:	8.8.8.8
			Validate settings upon exit	

1-1-3. PING command

Through the PING command, you can check the local TCP/IP protocol, and whether it can be normal connection to other computer local area network (LAN).



2. input "ping 127.0.0.1" to check the local TCP/IP protocol, it is normal when the receiving and



1. open the command prompt

4. input 'ping network device ip'' command to check whether the PC can connect to other PC in the LAN.

(1) input the command "ping 192.168.40.146", if the result shows "0% loss", this PC can connect the PC with IP 192.168.40.146.

	Command Prompt	_ C	×	
C:\Users\TXB>ping 127.0	.0.1			^
Pinging 127.0.0.1 with	32 bytes of data:			
Reply from 127.0.0.1: b	ytes=32 time<1ms TTL=128			
Reply from 127.0.0.1: b Reply from 127.0.0.1: b	ytes=32 time<1ms TTL=128			
Reply from 127.0.0.1. D Reply from 127.0.0.1: h	utes=32 time<1ms TTL=128			
	,			
Ping statistics for 127	.0.0.1:			
Packets: Sent = 4,	Received = 4, Lost = 0 (0% loss),			
Minimum = Oms. Maxi	num = Oms. Average = Oms			
C:\Users\TXB>ping 192.1	68.40.146			
Pinging 192.168.40.146	with 32 butes of data:			
Reply from 192.168.40.1	49: Destination host unreachable.			
Reply from 192.168.40.1	49: Destination host unreachable.			
Reply from 192.168.40.1	49: Destination host unreachable.			
Reply from 192.168.40.1	49: Destination host unreachable.			
Ping statistics for 192	.168.40.146:			
Packets: Sent = 4,	Received = 4, Lost = 0 (0% loss), 🔵			
C+\II.como\TYP\				
C. USEPS (IAB)				

(2) input the command "ping 192.168.40.127", it shows "100% loss", which means cannot connect to the PC with IP 192.168.40.127.

Note: in the ping statistics information, only 0% loss means communication normal.

The "ping network device IP" command can only ping four times. If you want to ping continuously, you can use the "ping network device IP -t" command, as shown in the following figure:



1-2. TCP IP protocol

TCP/IP protocol is a popular Ethernet communication protocol, compared with ISO open interconnection model, adopts a more open way, it has been recognized by the U.S. department of defense, and is widely used in practical engineering. TCP/IP protocol can be used in a variety of channels and the underlying protocol (such as T1, X.25 and RS232 serial interface). Specifically, TCP/IP protocol is including TCP protocol, IP protocol, the UDP protocol, ICMP protocol and some other groups.

1-2-1. Port number

In Ethernet, the communication based on TCP or UDP must use the port number to communicate with the upper application, port range is from 0 to 65535, some port numbers have default functions, such as port 80 for browsing the web service, port 21 for FTP service, port 502 for MODBUS TCP communications, and so on.

1-2-2. UDP protocol

UDP is the user data protocol, which is a simple connectionless transmission model with min protocol . UDP protocol doesn't have handshake mechanism, so the reliability of protocol is only equal to the underlying network. It cannot provide protection for receiving and sending message. UDP also provides checksum to ensure the integrity of data, and addresses different functions via different port numbers.

1-2-3. TCP protocol

1. The basic principle of TCP

TCP is transport control protocol, a connection-oriented, reliable transport layer protocol. Connection-oriented means a normal TCP transporting needs to build special virtual circuit between the TCP client and TCP server. To transmit data via TCP, a connection between the ends of the host must be established.

TCP provides reliable, orderly and error checking message function for application program running in the host machine which communicates through Ethernet. TCP can guarantee all the receiving and sending bytes have the same content and sequence. TCP protocol creates connections between active devices (i.e., a building connection device) and passive devices (i.e., receiving connection device). Once the connection is established, either party may initiate data transmission.

TCP protocol is a kind of "flow", which means that the message does not exist end flag, all received message is considered to be part of the data stream. For example, the client device sends three pieces of message to the server, each one is 20 bytes. Server only received a 60-byte "flow"

(assuming the server performs a receive operation after received three pieces of message).

2. The basic principle of socket

Socket (Socket) is the foundation of communication and basic operation unit to support the TCP/IP network communication. It is the abstract representations of the endpoint in the network communication process, contains five kinds of information for network communication: connection protocol, the IP address of the local host, port of the local process, the IP address of the remote host, the port of remote process.

When the application layer communicates through the transport layer, TCP will meet the problem of providing concurrent service for multiple application processes. Multiple TCP connections or more application processes may need pass through the same TCP port to transmit data. To distinguish different application processes and connection, many computer operating system provides a socket interface for the application and the TCP/IP protocol interaction. Application layer and transport layer can distinguish communication from different application processes or network connections through the socket interface, realize the data transmission of concurrent service.

3. Establish a socket connection

To establish a socket connection needs a pair of sockets at least, one runs on the client (also called the TCP client), called ClientSocket, another run on the server (also called the TCP server), called ServerSocket.

Socket connection process is divided into three steps: the server monitoring, the client request, connection confirmation.

Server monitoring: the server socket does not locate specific client socket, but is in a state of waiting for the connection, and real-time monitors network state, waits for the client's connection request.

Client requests: the client socket connection requests are put forward, the target is a server socket. For this reason, the client socket must first describe the server socket, and point out the server socket address and port number, and then the server socket connection requests are put forward.

Connection confirm: when the server socket receives the client socket connection request, it will response to the request of the client socket, set up a new thread, send a description of the server socket to the client, once the client confirms the description, the two sides have established connection. The server socket is in the listening state, continues to receive other client socket connection requests.

When creating a socket connection, you can specify the transport layer protocol, the socket can support different transport layer protocol (TCP or UDP), when using TCP protocol to connect the socket, the connection is a TCP connection.

TCP communication diagram:



In above diagram, the server socket is in the listening state, client connection requests to the server, the server receives a connection request and sends the reply to confirm the information to the client, after the client received message, it sends confirmation information to the server. After completion of the allocation of resources, a TCP connection is established successfully, this process is called "three-way handshake".

After the connection is established, the client and the server can send and receive data, after data transceiver is completed, the client or the server can request to close the connection, after the fourth "handshake", TCP connection is closed, all data transceiver interrupts.

2 Ethernet parameters

2-1. Ethernet parameters

2-1-1. IP address parameters

It needs to set the IP address in the Ethernet communication as the unique identification of each device. There are four parameters, the following charts are the IP setting interface of programming software.

	PLC1 - ethernet Set	×.
PLC Config I/O Password PLC Serial Port ethemet Module BD ED ED ED 4GBOX WBOX	general remote communication ethemet port: 8 Automatically obtain IP address IP: . subnet mask: . Default gateway: .	
	Read From PLC Write To PLC OK	Cancel

Obtain the IP

Support obtain the IP address automatically, static setting function, PLC initial setting is automatical obtain.

Automatic obtain mode: when there is a DHCP server in the subnet, IP, subnet mask, default gateway are assigned by the DHCP server. Without a DHCP server, network parameters use the default values:

IP address: 192.168.6.6

Subnet mask: 255.255.255.0

The default gateway: 192.168.6.1

Static specified mode: users assigned IP, subnet mask, default gateway information. Only supports private IP address information.

IP address type	IP address range	IP device quantity
Class A private address	10.0.0.0-10.255.255.255	16777216
Class B private address	172.16.0.0-172.31.255.255	1048576
Class C private address	192.168.0.0-192.168.255.255	65535

UDP multicast address

IP address type	IP address range	IP address
Type D address	224.0.0.0~224.0.0.255	Reserved multicast address (permanent
		group address)
	224.0.1.0~224.0.1.255	Public multicast address
	224.0.2.0~238.255.255.255	Available multicast addresses for users
		(temporary group addresses)
	239.0.0.~239.255.255.255	administratively scoped addresses

Note: It is recommended that users use IP addresses between $224.0.2.0 \sim 238.255.255.255$.

2-1-2. Function specification

Item	Parameter		
Number of a second section of a second	Ethernet series: 2 channels (same IP)		
Number of communication channels	XDH/XLH/XG2/XL5H series: 1 channel		
Communication speed	100Mbps		
Maximum space between stations	100m		
Network topology	Linear, star shape		

Communication type Maximum number of network nod				
Free format TCP	32			
UDP unicast	32			
UDP multicast	32			
Modbus TCP Client	XDH/XLH, Ethernet model: 32			
	XL5H: 4			
	XD3E: 8			
	Number of supported clients:			
Madhua TCD Samuar	XL5H: 4			
Widdbus ICF Server	XDH/XLH: 16			
	Ethernet model: 8			

Note:

There are a maximum of 32 TCP protocols, including free format TCP and Modbus TCP; Up to 32 UDP protocols, including UDP unicast and UDP multicast;

XDH and XLH series firmware versions 3.7.3 and above support UDP multicast functionality;

Ethernet PLC 3.7.2 and above firmware versions support UDP multicast functionality.

The UDP multicast function is only supported for Ethernet based PLCs with firmware versions

3.7.2 and above.

When using PLC as a server, Ethernet models of firmware version 3.7.2 and above support 8 clients; Ethernet models below version 3.7.2 support four clients. XDH/XLH models with firmware versions 3.7.2 and above support 16 clients, while XDH/XLH models with versions below 3.7.2 support 4 clients.

Ethernet models include: XD3E, XD5E, XDME, XL5E, XL5N, XLME.

2-2. Configure the Ethernet parameters in the software

Open the XINJE PLC programming software, click the ethernet in the left side, refer to below figure. This function is only available for Ethernet model.



Select remote communication in the above figure, you can configure the remote parameter, it no needs to set these parameters when communicating in the local area network (LAN), after completion of all the parameters, please restart the PLC to make the settings effective.

	PLC1 - ethernet Set
PLC Config	general remote communication enable remote:
	Read From PLC Write To PLC OK Cancel

2-3. Configure ethernet parameters in XINJEConfig

When configuring in XINJEConfig for Ethernet models, use a programming cable to connect the PLC and computer. Open the XINJEConfig configuration tool (using v2.3.0.9 version as an example), and select PLC from the configuration tool.

📑 Welcome	to use config tool					×	1
File(F) Too	ol(<u>1</u>) Environment(<u>E</u>)	Help(<u>H</u>)					L
🔌 PC 🎯 WBox	PLCCOBox	TouchWin	3	4GBox			100
							_
On line						3	

Method 1: Connect in Modbus TCP mode

In the pop-up dialog box, select Ethernet for the communication interface and Modbus for the communication protocol. At this time, connect the PLC using the Modbus TCP protocol, as shown in the following figure.

						Help
Cor	nnection m	node				
Inte	erface:	Ethemet		~		
Pro	otocol:	Modbus		~		
Cor	mm config	parameters				1
	IP:	192 <u>168</u>	66	<	Scan	
			0	Conn	ecting	

Method 2: Connect in Xnet mode

Select Ethernet for the communication interface, XNet for the communication protocol, and the specified address for the connection method. At this time, connect the PLC using the XNet protocol, set the device IP and corresponding network card, and click to connect the device, as shown in the following figure.

.CLinkForm	×— [×
Connection m	iode	Help
Interface:	Ethemet ~	
Protocol:	XNet ~	
connection:	SpecifiedAddress \lor	
Comm config	parameters	
IP:	192 168 6 6 <sca< td=""><td>in</td></sca<>	in
Adapter:	Ethernet 2 ~	
	Connectir	ng
	Connectir	ng
	CLinkForm Connection m Interface: Protocol: connection: Comm config IP: Adapter:	CLinkForm — Connection mode Interface: Ethemet · · Protocol: XNet · · connection: SpecifiedAddress · Comm config parameters IP: 192_168_6_6 < <sca Adapter: Ethemet 2 · · Connectir</sca

When configuring Ethernet parameters, select Ethernet configuration. Please refer to Section 2-1-1 for the description of configuration items, and the functions are the same as those of XDPPro configuration.

General Re	mote Advanced				Help
Ethen You c Port.	net Port 9 🗳	each Ethernet			
	 Get IP By DHCP Use Static IP 				
	IP address	192 168	66	3 1	
	Mask	255 255 2	255 0	V (
	Gate	li÷			

3 Wiring and communication protocol

3-1. Wiring mode

The physical interface of Ethernet model is RJ45, the wiring cable is recommended to use UTP and STP cable, single length cannot be more than 100 meters. Switch type is recommended to use MB/GB adaptive switch.

3-2. MODBUS TCP protocol

3-2-1. MODBUS TCP overview

MODBUS TCP combined standard TCP/IP, Ethernet physical network and MODBUS as the data representation method of data application protocol. MODBUS TCP communication message is encapsulated in Ethernet TCP/IP packets, MODBUS protocol one frame maximum length is 256 bytes.

MODBUS TCP/IP has two type of devices: Modbus TCP/IP clinet and server.

MODBUS client:

Client (TCP Client) launched a connection request to the Server (TCP Server), the connection is established successfully, it only allows the Client to initiate communication request.

When the Ethernet model is the MODBUS TCP client, it establishes a TCP connection through S_OPEN instruction, initiates MODBUS request by M_TCP instruction.

MODBUS server:

The server listened to port 502, waited for the client connection request, after the connection was established successfully, it responsed to the data communication request in accordance with the Modbus TCP protocol specification.

Ethernet devices defaulted open this service when power on, the maximum response is shown in the table below.

Firmwore version	PLC model	Number of
Filliware version		supported clients
Below 3.7.2	XD5E/XL5E/XDME/XLME/XDH/XLH	4
272	XL5H	
3.7.2 and up	XD3E/XD5E/XL5E/XL5N/XDME/XLME	8

Note: The number of clients supported by PLC is as follows:

Firmware version	PLC model	Number of supported clients
	XDH/XLH	16

3-2-2. MODBUS address

When the programmable controller serves as a Modbus server, the internal software component numbers and corresponding Modbus address numbers are as follows.

(1) Modbus address and internal software component comparison table for XD3E series PLC

Note: For the calculation of Modbus addresses for X and Y, please refer to the bottom of the table.

Туре	Component	Range	Quantity	Modbus address	Modbus address
				(hex)	(decimal)
	M	M0~M7999	8000	0~1F3F	0~7999
		X0~X77 (main body)	64	5000~503F	20480~20543
		X10000~X10077 (#1	64	5100~513F	20736~20799
		module)	04		
		X10100~X10177 (#2	64	5140~517F	20800~20863
		module)	04		
		X10200~X10277 (#3	64	5180~51BF	20864~20927
		module)			
		X10300~X10377 (#4	64	51C0~51FF	20928~20991
		module)			
		X10400~X10477 (#5	64	5200~523F	20992~21055
		module)			
	v	X10500~X10577 (#6	64	5240~527F	21056~21119
Coil, bit	Λ	module)			
object		X10600~X10677 (#7	64	5280~52BF	21120~21183
		module)			
		X10700~X10777 (#8	64	52C0~52FF	21184~21247
		module)			
		X11000~X11077 (#9	64	5300~533F	21248~21311
		module)			
		X11100~X11177 (#10	64	5340~537F	21312~21375
		module)			
		X20000~X20077 (#1 BD)	64	58D0~590F	22736~22799
		X20100~X20177 (#2 BD)	64	5910~594F	22800~22863
		X30000~X30077 (#1 ED)	64	5BF0~5C2F	23536~23599
		Y0~Y77 (main body)	64	6000~603F	24576~24639
	Y	Y10000~Y10077 (#1	(A	6100~613F	24832~24895
		module)	04		

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		Y10100~Y10177 (#2	()	6140~617F	24896~24959
		module)	64		
		Y10200~Y10277 (#3	64	6180~61BF	24960~25023
		module)			
		Y10300~Y10377 (#4	64	61C0~61FF	25024~25087
		module)			
		Y10400~Y10477 (#5	64	6200~623F	25088~25151
		module)			
		Y10500~Y10577 (#6	64	6240~627F	25152~25215
		module)			
		Y10600~Y10677 (#7	64	6280~62BF	25216~25279
		module)			
		Y10700~Y10777 (#8	64	62C0~62FF	25280~25343
		module)			
		Y11000~Y11077 (#9	64	6300~633F	25344~25407
		module)			
		Y11100~Y11177 (#10	64	6340~637F	25408~25471
		module)			
		Y20000~Y20077 (#1 BD)	64	68D0~690F	26832~26895
		Y20100~Y20177 (#2 BD)	64	6910~694F	26896~26956
		Y30000~Y30077 (#1 ED)	64	6BF0~6C2F	27632~27695
	S	S0~S1023	1024	7000~73FF	28672~29695
	SM	SM0~SM2047	2048	9000~97FF	36864~38911
	Т	T0~T575	576	A000~A23F	40960~41535
	С	C0~C575	576	B000~B23F	45056~45631
	ET	ET0~ET31	32	C000~C01F	49152~49183
	SEM	SEM0~SEM31	32	C080~C09F	49280~49311
	HM ^{*1}	HM0~HM959	960	C100~C4BF	49408~50367
	HS ^{*1}	HS0~HS127	128	D900~D97F	55552~55679
	HT ^{*1}	HT0~HT95	96	E100~E15F	57600~57695
	HC ^{*1}	HC0~HC95	96	E500~E55F	58624~58719
	HSC ^{*1}	HSC0~HSC31	32	E900~E91F	59648~59679
	D	D0~D7999	8000	0~1F3F	0~7999
		ID0~ID99 (main body)	100	5000~5063	20480~20579
Docistor		ID10000~ID10099 (#1	100	5100 5162	20726 20025
word		module)	100	5100~5105	20730~20833
	ID	ID10100~ID10199 (#2	100	5164-5107	20836, 20025
		module)	100	5104~5107	20030~20933
		ID10200~ID10299 (#3	100	51C8~522B	20036-21025
		module)			20930~21035

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		ID10300~ID10399 (#4	100	522C~528F	21026 21125
		module)			21036~21135
		ID10400~ID10499 (#5	100	5290~52F3	21126 21225
		module)			21130~21233
		ID10500~ID10599 (#6	100	52F4~5357	21236-21335
		module)			21250-21555
		ID10600~ID10699 (#7	100	5358~53BB	21336~21435
		module)			21550-21455
		ID10700~ID10799 (#8	100	53BC~541F	21/136-21535
		module)			21430-21333
		ID10800~ID10899 (#9	100	5420~5483	21536~21635
		module)			21550-21055
		ID10900~ID10999 (#10	100	5484~54E7	21626 21725
		module)			21030~21733
		ID20000~ID20099 (#1 BD)	100	58D0~5933	22736~22835
		ID20100~ID20199 (#2 BD)	100	5934~5997	22836~22935
		ID30000~ID30099 (#1 ED)	100	5BF0~5C53	23536~23635
		QD0~QD99 (main body)	100	6000~6063	24576~24675
		QD10000~QD10099 (#1	100	(100 (1(2	24822 24021
		module)	100	0100~0103	24832~24931
		QD10100~QD10199 (#2	100	(1(4)(107	24932~25031
		module)		0104~01C/	
		QD10200~QD10299 (#3	100	61C9 622D	25032~25131
		module)		01C8~022B	
		QD10300~QD10399 (#4	100	622C, 628E	25132~25231
		module)		022C~028F	
		QD10400~QD10499 (#5	100	6200, 62E2	25232~25331
Register,		module)		0290~02F3	
word	QD	QD10500~QD10599 (#6	100	62E4, 6257	25332~25431
object		module)		02F4~0337	
		QD10600~QD10699 (#7	100	6259 62DD	25432~25531
		module)		0338~03BB	
		QD10700~QD10799 (#8	100	62PC 641F	25532~25631
		module)		03BC~0411	
		QD10800~QD10899 (#9	100	6420, 6483	25632~25731
		module)		0420~0485	
		QD10900~QD10999 (#10	100	6484~6457	25732~25831
		module)		0404~04E/	
		QD20000~QD20099 (#1	100	68D0, 6022	26832, 26021
		BD)	100	0000~0933	20032~20931

Туре	Component	Range	Quantity	Modbus address (hex)	Modbus address (decimal)
		QD20100~QD20199 (#2 BD)	100	6934~6997	26932~27031
		QD30000~QD30099 (#1 ED)	100	6BF0~6C53	27632~27731
	SD	SD0~SD2047	2048	7000~77FF	28672~30719
	TD	TD0~TD575	576	8000~823F	32768~33343
	CD	CD0~CD575	576	9000~923F	36864~37439
	ETD	ETD0~ETD31	32	A000~A01F	40960~40991
	HD ^{*1}	HD0~HD999	1000	A080~A467	41088~42087
	HSD ^{*1}	HSD0~HSD499	500	B880~BA73	47232~47731
	HTD ^{*1}	HTD0~HTD95	96	BC80~BCDF	48256~48351
	HCD ^{*1}	HCD0~HCD95	96	C080~C0DF	49280~49375
	HSCD ^{*1}	HSCD0~HSCD31	32	C480~C49F	50304~50335
	FD ^{*2}	FD0~FD5119	5120	C4C0~D8BF	50368~55487
	SFD ^{*2}	SFD0~SFD1999	2000	E4C0~EC8F	58560~60559
	FS ^{*2}	FS0~FS47	48	F4C0~F4EF	62656~62703

(2) XD5E, XDME, XL5E, XL5N, XL5H, XLME series Modbus address and internal software component comparison table.

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
	М	M0~M20479	20480	0~4FFF	0~20479
		X0~X77 (main body)	64	5000~503F	20480~20543
		X10000~X10077 (#1	64	5100~513F	20736~20799
		module)	04		
		X10100~X10177 (#2	64	5140~517F	20800~20863
	Х	module)	04		
		X10200~X10277 (#3	64	5180~51BF	20864~20927
Coil hit		module)			
con, on		X10300~X10377 (#4	64	51C0~51FF	20928~20991
object		module)			
		X10400~X10477 (#5	64	5200~523F	20992~21055
		module)			
		X10500~X10577 (#6	64	5240~527F	21056~21119
		module)			
		X10600~X10677 (#7	64	5280~52BF	21120~21183
		module)			
		X10700~X10777 (#8	64	52C0~52FF	21184~21247

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		module)			
		X11000~X11077 (#9	64	5300~533F	21248~21311
		module)			
		X11100~X11177 (#10	64	5340~537F	21312~21375
		module)			
		X11200~X11277 (#11	64	5380~53BF	21376~21439
		module)			
		X11300~X11377 (#12	64	53C0~53FF	21440~21503
		module)			
		X11400~X11477 (#13	64	5400~543F	21504~21567
		module)			
		X11500~X11577 (#14	64	5440~547F	21568~21631
		module)			
		X11600~X11677 (#15	64	5480~54BF	21632~21695
		module)			
		X11700~X11777 (#16	64	54C0~54FF	21696~21759
		module)			
		X20000~X20077 (#1 BD)	64	58D0~590F	22736~22799
		X20100~X20177 (#2 BD)	64	5910~594F	22800~22863
		X30000~X30077 (#1 ED)	64	5BF0~5C2F	23536~23599
		Y0~Y77 (main body)	64	6000~603F	24576~24639
		Y10000~Y10077 (#1	64	6100~613F	24832~24895
		module)			
		Y10100~Y10177 (#2	64	6140~617F	24896~24959
		module)	0.		
		Y10200~Y10277 (#3	64	6180~61BF	24960~25023
		module)			
		Y10300~Y10377 (#4	64	61C0~61FF	25024~25087
		module)			
	Y	Y10400~Y10477 (#5	64	6200~623F	25088~25151
		module)			
		Y10500~Y10577 (#6	64	6240~627F	25152~25215
		module)		(000 (000	
		Y10600~Y10677 (#7	64	6280~62BF	25216~25279
		module)		(0.00 (0.00	
		Y 10700~Y 10777 (#8	64	62C0~62FF	25280~25343
		module)		(200 (225	
		Y 11000~Y 11077 (#9	64	6300~633F	25344~25407
		module)	<i>C</i> A	(240 (275	05400 05451
		Y11100~Y11177 (#10	64	6340~637F	25408~25471

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		module)			
		Y11200~Y11277 (#11	64	6380~63BF	25472~25535
		module)			
		Y11300~Y11377 (#12	64	63C0~63FF	25536~25599
		module)			
		Y11400~Y11477 (#13	64	6400~643F	25600~25663
		module)			
		Y11500~Y11577 (#14	64	6440~647F	25664~25727
		module)			
		Y11600~Y11677 (#15	64	6480~64BF	25728~25791
		module)			
		Y11700~Y11777 (#16	64	64C0~64FF	25792~25855
		module)			
		Y20000~Y20077 (#1 BD)	64	68D0~690F	26832~26895
		Y20100~Y20177 (#2 BD)	64	6910~694F	26896~26956
		Y30000~Y30077 (#1 ED)	64	6BF0~6C2F	27632~27695
	S	S0~S7999	8000	7000~8F3F	28672~36671
	SM	SM0~SM4095	4096	9000~9FFF	36864~40959
	Т	T0~T4095	4096	A000~AFFF	40960~45055
	С	C0~C4095	4096	B000~BFFF	45056~49151
	ET	ET0~ET39	40	C000~C027	49152~49191
	SEM	SEM0~SEM127	128	C080~C0FF	49280~49407
	HM ^{*1}	HM0~HM6143	6144	C100~D8FF	49408~55551
	HS ^{*1}	HS0~HS999	1000	D900~DCEF	55552~56551
	HT ^{*1}	HT0~HT1023	1024	E100~E4FF	57600~58623
	HC ^{*1}	HC0~HC1023	1024	E500~E8FF	58624~59647
	HSC ^{*1}	HSC0~HSC39	40	E900~E927	59648~59687
	D	D0~D20479	20480	0~4FFF	0~20479
		ID0~ID99 (main body)	100	5000~5063	20480~20579
		ID10000~ID10099 (#1	100	5100 5162	20726 20825
		module)	100	5100~5105	20730~20833
		ID10100~ID10199 (#2	100	5164, 5107	20826, 20025
Register,		module)	100	5104~5107	20830~20933
word	ID	ID10200~ID10299 (#3	100	51C8~522B	20936-21035
object	ID	module)			20750-21055
		ID10300~ID10399 (#4	100	522C~528F	21036~21135
		module)			21030-21133
		ID10400~ID10499 (#5	100	5290~52F3	21136~21235
		module)			21130-21233
		ID10500~ID10599 (#6	100	52F4~5357	21236~21335

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		module)			
		ID10600~ID10699 (#7	100	5358~53BB	
		module)			21336~21435
		ID10700~ID10799 (#8	100	53BC~541F	
		module)			21436~21535
		ID10800~ID10899 (#9	100	5420~5483	
		module)			21536~21635
		ID10900~ID10999 (#10	100	5484~54E7	01/02/ 01/20/
		module)			21636~21/35
		ID11000~ID11099 (#11	100	54E8~554B	01706 01005
		module)			21/36~21835
		ID11100~ID11199 (#12	100	554C~55AF	21926 21025
		module)			21830~21933
		ID11200~ID11299 (#13	100	55B0~5613	21026 22025
		module)			21936~22033
		ID11300~ID11399 (#14	100	5614~5677	22036-22135
		module)			22030~22133
		ID11400~ID11499 (#15	100	5678~56DB	22136-22235
		module)			22150-22255
		ID11500~ID11599 (#16	100	56DC~573F	22236~22335
		module)			22230 22333
		ID20000~ID20099 (#1 BD)	100	58D0~5933	22736~22835
		ID20100~ID20199 (#2 BD)	100	5934~5997	22836~22935
		ID30000~ID30099 (#1 ED)	100	5BF0~5C53	23536~23635
		QD0~QD99 (main body)	100	6000~6063	24576~24675
		QD10000~QD10099 (#1	100	6100~6163	24832~24931
		module)	100	0100 0100	
		QD10100~QD10199 (#2	100	6164~61C7	24932~25031
		module)			
		QD10200~QD10299 (#3	100	61C8~622B	25032~25131
		module)			
	QD	QD10300~QD10399 (#4	100	622C~628F	25132~25231
		module)			
		QD10400~QD10499 (#5	100	6290~62F3	25232~25331
		module)			
		QD10500~QD10599 (#6	100	62F4~6357	25332~25431
		module)			
		QD10600~QD10699 (#7	100	6358~63BB	25432~25531
		module)	100		
		QD10700~QD10799 (#8	100	63BC~641F	25532~25631

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		module)			
		QD10800~QD10899 (#9	100	6420 6482	25632~25731
		module)		0420~0483	
		QD10900~QD10999 (#10	100	6484 6457	25732~25831
		module)		0484~0417	
		QD11000~QD11099 (#11	100	64E8- 654B	25832~25931
		module)		0428~054D	
		QD11100~QD11199 (#12	100	654C. 65AE	25932~26031
		module)		034C~03AI	
		QD11200~QD11299 (#13	100	65B0-6613	26032~26131
		module)		0500~0015	
		QD11300~QD11399 (#14	100	6614~6677	26132~26231
		module)		001+*0077	
		QD11400~QD11499 (#15	100	6678~66DB	26232~26331
		module)		0070-00DD	
		QD11500~QD11599 (#16	100	66DC~673F	26332~26431
		module)		0020 0751	
		QD20000~QD20099 (#1	100	68D0~6933	26832~26931
		BD)	100	0020 0755	20032 20331
		QD20100~QD20199 (#2	100	6934~6997	26932~27031
	OD	BD)	100		
		QD30000~QD30099 (#1	100	6BF0~6C53	27632~27731
		ED)	100		
	SD	SD0~SD4095	4096	7000~7FFF	28672~32767
	TD	TD0~TD4095	4096	8000~8FFF	32768~36863
	CD	CD0~CD4095	4096	9000~9FFF	36864~40959
	ETD	ETD0~ETD39	40	A000~A027	40960~40999
	HD ^{*1}	HD0~HD6143	6144	A080~B87F	41088~47231
	HSD ^{*1}	HSD0~HSD1023	1024	B880~BC7F	47232~48255
	HTD ^{*1}	HTD0~HTD1023	1024	BC80~C07F	48256~49279
	HCD ^{*1}	HCD0~HCD1023	1024	C080~C47F	49280~50303
	HSCD ^{*1}	HSCD0~HSCD39	40	C480~C4A7	50304~50343
	FD ^{*2}	FD0~FD8191	8192	C4C0~E4BF	50368~58559
	SFD ^{*2}	SFD0~SFD4095	4096	E4C0~F4BF	58560~62655
	FS ^{⋇2}	FS0~FS47	48	F4C0~F4EF	62656~62703

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
	М	M0~M20479	20480	0~4FFF	0~20479
		X0~X77 (main body)	64	5000~503F	20480~20543
		X10000~X10077 (#1	64	5100~513F	20736~20799
		module)	04		
		X10100~X10177 (#2	64	5140~517F	20800~20863
		module)	04		
		X10200~X10277 (#3	64	5180~51BF	20864~20927
		module)			
		X10300~X10377 (#4	64	51C0~51FF	20928~20991
		module)			
		X10400~X10477 (#5	64	5200~523F	20992~21055
		module)			
		X10500~X10577 (#6	64	5240~527F	21056~21119
		module)			
		X10600~X10677 (#7	64	5280~52BF	21120~21183
	Х	module)			
		X10700~X10777 (#8	64	52C0~52FF	21184~21247
		module)			
Coil bit		X11000~X11077 (#9	64	5300~533F	21248~21311
object		module)			
00jeet		X11100~X11177 (#10	64	5340~537F	21312~21375
		module)			
		X11200~X11277 (#11	64	5380~53BF	21376~21439
		module)			
		X11300~X11377 (#12	64	53C0~53FF	21440~21503
		module)			
		X11400~X11477 (#13	64	5400~543F	21504~21567
		module)			
		X11500~X11577 (#14	64	5440~547F	21568~21631
		module)			
		X11600~X11677 (#15	64	5480~54BF	21632~21695
		module)			
		X11700~X11777 (#16	64	54C0~54FF	21696~21759
		module)			
		X20000~X20077 (#1 BD)	64	58D0~590F	22736~22799
		X20100~X20177 (#2 BD)	64	5910~594F	22800~22863
		X30000~X30077 (#1 ED)	64	5BF0~5C2F	23536~23599
	Y	Y0~Y77 (main body)	64	6000~603F	24576~24639
	1	Y10000~Y10077 (#1	64	6100~613F	24832~24895

(3) X	DH and XLH	Series Modbus Addresses	and Inter	rnal Softv	ware Compone	nts.

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		module)			
		Y10100~Y10177 (#2	()	6140~617F	24896~24959
		module)	64		
		Y10200~Y10277 (#3	64	6180~61BF	24960~25023
		module)			
		Y10300~Y10377 (#4	64	61C0~61FF	25024~25087
		module)			
		Y10400~Y10477 (#5	64	6200~623F	25088~25151
		module)			
		Y10500~Y10577 (#6	64	6240~627F	25152~25215
		module)			
		Y10600~Y10677 (#7	64	6280~62BF	25216~25279
		module)			
		Y10700~Y10777 (#8	64	62C0~62FF	25280~25343
		module)			
		Y11000~Y11077 (#9	64	6300~633F	25344~25407
		module)			
		Y11100~Y11177 (#10	64	6340~637F	25408~25471
		module)			
		Y11200~Y11277 (#11	64	6380~63BF	25472~25535
		module)			
		Y11300~Y11377 (#12	64	63C0~63FF	25536~25599
		module)			
		Y11400~Y11477 (#13	64	6400~643F	25600~25663
	Y	module)			
		Y11500~Y11577 (#14	64	6440~647F	25664~25727
		module)			
		Y11600~Y11677 (#15	64	6480~64BF	25728~25791
		module)			
		Y11700~Y11777 (#16	64	64C0~64FF	25792~25855
		module)			
		Y20000~Y20077 (#1 BD)	64	68D0~690F	26832~26895
		Y20100~Y20177 (#2 BD)	64	6910~694F	26896~26956
		Y30000~Y30077 (#1 ED)	64	6BF0~6C2F	27632~27695
	S	S0~S7999	8000	7000~8F3F	28672~36671
	SM	SM0~SM4095	4096	9000~9FFF	36864~40959
	Т	T0~T4095	4096	A000~AFFF	40960~45055
	С	C0~C4095	4096	B000~BFFF	45056~49151
	ET	ET0~ET39	40	C000~C027	49152~49191
	SEM	SEM0~SEM127	128	C080~C0FF	49280~49407

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
	HM ^{*1}	HM0~HM6143	6144	C100~D8FF	49408~55551
	HS ^{∦1}	HS0~HS999	1000	D900~DCEF	55552~56551
	HT ^{×1}	HT0~HT1023	1024	E100~E4FF	57600~58623
	HC ^{*1}	HC0~HC1023	1024	E500~E8FF	58624~59647
	HSC ^{*1}	HSC0~HSC39	40	E900~E927	59648~59687
	D	D0~D20479	20480	0~4FFF	0~20479
		ID0~ID99 (main body)	100	5000~5063	20480~20579
		ID10000~ID10099 (#1	100	5100 5162	20726 20825
		module)	100	5100~5165	20730~20833
		ID10100~ID10199 (#2	100	5164.5107	20826, 20025
		module)	100	5104~5107	20830~20933
		ID10200~ID10299 (#3	100	51C8~522B	20936-21035
		module)			20750-21055
		ID10300~ID10399 (#4	100	522C~528F	21036~21135
		module)			21050-21155
		ID10400~ID10499 (#5	100	5290~52F3	21136~21235
		module)			21130 21233
		ID10500~ID10599 (#6	100	52F4~5357	21236~21335
		module)			21230 21333
		ID10600~ID10699 (#7	100	5358~53BB	21336~21435
		module)			
Register,		ID10700~ID10799 (#8	100	53BC~541F	21436~21535
word	ID	module)			
object		ID10800~ID10899 (#9	100	5420~5483	21536~21635
		module)	100	5404 5455	
		ID10900~ID10999 (#10	100	5484~54E7	21636~21735
		module)	100	54E9 554D	
		ID11000~ID11099 (#11	100	54E8~554B	21736~21835
		module)	100	554C 55AE	
		ID11100~ID11199 (#12	100	554C~55AF	21836~21935
		ID11200. ID11200 (#12	100	55D0.5612	
		module)	100	5500~5015	21936~22035
		ID11300~ID11300 (#1/	100	5614~5677	
		module)	100	5017-50//	22036~22135
		ID11400~ID11499 (#15	100	5678~56DB	
		module)			22136~22235
		ID11500~ID11599 (#16	100	56DC~573F	
		module)		0.020 0701	22236~22335
		ID20000~ID20099 (#1 BD)	100	58D0~5933	22736~22835

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		ID20100~ID20199 (#2 BD)	100	5934~5997	22836~22935
		ID30000~ID30099 (#1 ED)	100	5BF0~5C53	23536~23635
		QD0~QD99 (main body)	100	6000~6063	24576~24675
		QD10000~QD10099 (#1	100	(100 (1(2	24822 24021
		module)	100	6100~6163	24832~24931
		QD10100~QD10199 (#2	100	(1(4)(107	24932~25031
		module)		0104~01C/	
		QD10200~QD10299 (#3	100	61C9 622D	25032~25131
		module)		01C8~022B	
		QD10300~QD10399 (#4	100	(220 (295	25132~25231
	QD	module)		022C~028F	
		QD10400~QD10499 (#5	100	6200 6252	25232~25331
		module)		0290~0215	
		QD10500~QD10599 (#6	100	6254 6257	25332~25431
		module)		0214~0337	
		QD10600~QD10699 (#7	100	6358-63BB	25432~25531
		module)		0338~03BB	
		QD10700~QD10799 (#8	100	63BC~641F	25532~25631
		module)		03BC~0411	
		QD10800~QD10899 (#9	100	6420~6483	25632~25731
		module)		0420*0405	
		QD10900~QD10999 (#10	100	6484~64F7	25732~25831
		module)			
		QD11000~QD11099 (#11	100	64E8~654B	25832~25931
		module)			
		QD11100~QD11199 (#12	100	654C~65AF	25932~26031
		module)		0010 0011	
		QD11200~QD11299 (#13	100	65B0~6613	26032~26131
		module)			
	QD	QD11300~QD11399 (#14	100	6614~6677	26132~26231
		module)			
		QD11400~QD11499 (#15	100	6678~66DB	26232~26331
		module)			
		QD11500~QD11599 (#16	100	66DC~673F	26332~26431
		module)			
		QD20000~QD20099 (#1	100	68D0~6933	26832~26931
		BD)			
		QD20100~QD20199 (#2	100	6934~6997	26932~27031
		BD)			
		QD30000~QD30099 (#1	100	6BF0~6C53	27632~27731

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		ED)			
	SD	SD0~SD4095	4096	7000~7FFF	28672~32767
	TD	TD0~TD4095	4096	8000~8FFF	32768~36863
	CD	CD0~CD4095	4096	9000~9FFF	36864~40959
	ETD	ETD0~ETD39	40	A000~A027	40960~40999
	HD ^{*1}	HD0~HD6143	6144	A080~B87F	41088~47231
	HSD ^{*1}	HSD0~HSD1023	1024	B880~BC7F	47232~48255
	HTD ^{*1}	HTD0~HTD1023	1024	BC80~C07F	48256~49279
	HCD ^{*1}	HCD0~HCD1023	1024	C080~C47F	49280~50303
	HSCD ^{*1}	HSCD0~HSCD39	40	C480~C4A7	50304~50343
	FD ^{*2}	FD0~FD8191	8192	C4C0~E4BF	50368~58559
	SFD ^{*2}	SFD0~SFD4095	4096	E4C0~FC2F	58560~64559
	FS ^{×2}	FS0~FS47	256	F4C0~F4EF	62656~62911

(4) XG Series Modbus Address and Internal Software Components:

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
	М	M0~M20479	20480	0~4FFF	0~20479
		X0~X77 (main body)	64	5000~503F	20480~20543
		X10000~X10077 (#1	64	5100~513F	20736~20799
		module)	04		
		X10100~X10177 (#2	64	5140~517F	20800~20863
		module)	04		
		X10200~X10277 (#3	64	5180~51BF	20864~20927
	X	module)			
		X10300~X10377 (#4	64	51C0~51FF	20928~20991
		module)			
Coil, bit		X10400~X10477 (#5	64	5200~523F	20992~21055
object		module)			
		X10500~X10577 (#6	64	5240~527F	21056~21119
		module)			
		X10600~X10677 (#7	64	5280~52BF	21120~21183
		module)			
		X10700~X10777 (#8	64	52C0~52FF	21184~21247
		module)			
		X11000~X11077 (#9	64	5300~533F	21248~21311
		module)			
		X11100~X11177 (#10	64	5340~537F	21312~21375
		module)			

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		X11200~X11277 (#11	64	5380~53BF	21376~21439
		module)			
		X11300~X11377 (#12	64	53C0~53FF	21440~21503
		module)			
		X11400~X11477 (#13	64	5400~543F	21504~21567
		module)			
		X11500~X11577 (#14	64	5440~547F	21568~21631
		module)			
		X11600~X11677 (#15	64	5480~54BF	21632~21695
		module)			
		X11700~X11777 (#16	64	54C0~54FF	21696~21759
		module)			
		Y0~Y77 (main body)	64	6000~603F	24576~24639
		Y10000~Y10077 (#1	64	6100~613F	24832~24895
		module)	04		
		Y10100~Y10177 (#2	64	6140~617F	24896~24959
		module)	04		
		Y10200~Y10277 (#3	64	6180~61BF	24960~25023
		module)			
		Y10300~Y10377 (#4	64	61C0~61FF	25024~25087
		module)			
		Y10400~Y10477 (#5	64	6200~623F	25088~25151
		module)			
		Y10500~Y10577 (#6	64	6240~627F	25152~25215
		module)			
		Y10600~Y10677 (#7	64	6280~62BF	25216~25279
	Y	module)			
		Y10700~Y10777 (#8	64	62C0~62FF	25280~25343
		module)			
		Y11000~Y11077 (#9	64	6300~633F	25344~25407
		module)			
		Y11100~Y11177 (#10	64	6340~637F	25408~25471
		module)			
		Y11200~Y11277 (#11	64	6380~63BF	25472~25535
		module)			
		Y11300~Y11377 (#12	64	63C0~63FF	25536~25599
		module)			
		Y11400~Y11477 (#13	64	6400~643F	25600~25663
		module)			
		Y11500~Y11577 (#14	64	6440~647F	25664~25727
		module)			

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
		-		(hex)	(decimal)
		Y11600~Y11677 (#15	64	6480~64BF	25728~25791
		module)			
		Y11700~Y11777 (#16	64	64C0~64FF	25792~25855
		module)			
	S	S0~S7999	8000	7000~8F3F	28672~36671
	SM	SM0~SM4095	4096	9000~9FFF	36864~40959
	Т	T0~T4095	4096	A000~AFFF	40960~45055
	С	C0~C4095	4096	B000~BFFF	45056~49151
	ET	ET0~ET39	40	C000~C027	49152~49191
	SEM	SEM0~SEM127	128	C080~C0FF	49280~49407
	HM ^{*1}	HM0~HM6143	6144	C100~D8FF	49408~55551
	HS^{*1}	HS0~HS999	1000	D900~DCEF	55552~56551
	HT ^{×1}	HT0~HT1023	1024	E100~E4FF	57600~58623
	HC ^{*1}	HC0~HC1023	1024	E500~E8FF	58624~59647
	HSC ^{*1}	HSC0~HSC39	40	E900~E927	59648~59687
	D	D0~D20479	20480	0~4FFF	0~20479
		ID0~ID99 (main body)	100	5000~5063	20480~20579
		ID10000~ID10099 (#1	100	5100~5163	20736~20835
		module)	100	5100 5105	
		ID10100~ID10199 (#2	100	5164~51C7	20836~20935
		module)	100		20030 20030
		ID10200~ID10299 (#3	100	51C8~522B	20936~21035
		module)			
		ID10300~ID10399 (#4	100	522C~528F	21036~21135
		module)	100		
		ID10400~ID10499 (#5	100	5290~52F3	21136~21235
Register,		module)	100	5054 5057	
word	ID	ID10500~ID10599 (#6	100	52F4~5357	21236~21335
object		module)	100	5259 52DD	
		ID10600~ID10699 (#/	100	2328~23BB	21336~21435
		ID10700, ID10700 (#8	100	52DC 5/1E	
		module)	100	5560~5411	21436~21535
		ID10800~ID10899 (#9	100	5420~5483	
		module)	100	5120 5105	21536~21635
		ID10900~ID10999 (#10	100	5484~54E7	
		module)	100		21636~21735
		ID11000~ID11099 (#11	100	54E8~554B	
		module)			21736~21835
		ID11100~ID11199 (#12	100	554C~55AF	21836~21935

				Modbus	Modbus
Туре	Component	Range	Quantity	address	address
				(hex)	(decimal)
		module)			
		ID11200~ID11299 (#13	100	55B0~5613	21026 22025
		module)			21930~22033
		ID11300~ID11399 (#14	100	5614~5677	22026 22125
		module)			22030~22133
		ID11400~ID11499 (#15	100	5678~56DB	22126 22225
		module)			22130~22233
		ID11500~ID11599 (#16	100	56DC~573F	22226 22225
		module)			22230~22555
		QD0~QD99 (main body)	100	6000~6063	24576~24675
		QD10000~QD10099 (#1	100	6100 6162	24822 24021
		module)	100	0100~0103	24832~24931
		QD10100~QD10199 (#2	100	6164 6107	24932~25031
		module)		0104~01C7	
		QD10200~QD10299 (#3	100	61C9 622D	25032~25131
		module)		01C8~022B	
		QD10300~QD10399 (#4	100	622C 628E	25132~25231
		module)		0220~0281	
		QD10400~QD10499 (#5	100	6290~62F3	25232~25331
		module)		0270*021*3	
		QD10500~QD10599 (#6	100	62F4~6357	25332~25431
		module)		021 + 00007	
		QD10600~QD10699 (#7	100	6358~63BB	25432~25531
		module)		0550 0500	
	QD	QD10700~QD10799 (#8	100	63BC~641F	25532~25631
		module)		0520 0111	
		QD10800~QD10899 (#9	100	6420~6483	25632~25731
		module)		0.20 0.00	
		QD10900~QD10999 (#10	100	6484~64E7	25732~25831
		module)			
		QD11000~QD11099 (#11	100	64E8~654B	25832~25931
		module)			
		QD11100~QD11199 (#12	100	654C~65AF	25932~26031
		module)			
		QD11200~QD11299 (#13	100	65B0~6613	26032~26131
		module)			
		QD11300~QD11399 (#14	100	6614~6677	26132~26231
		module)			
		QD11400~QD11499 (#15	100	6678~66DB	26232~26331
		module)			

	Component		Quantity	Modbus	Modbus
Туре		Range		address	address
				(hex)	(decimal)
		QD11500~QD11599 (#16	100	66DC~673F	26332~26431
		module)		00DC*0751	
	SD	SD0~SD4095	4096	7000~7FFF	28672~32767
	TD	TD0~TD4095	4096	8000~8FFF	32768~36863
	CD	CD0~CD4095	4096	9000~9FFF	36864~40959
	ETD	ETD0~ETD39	40	A000~A027	40960~40999
	HD ^{*1}	HD0~HD6143	6144	A080~B87F	41088~47231
	HSD ^{*1}	HSD0~HSD1023	1024	B880~BC7F	47232~48255
	HTD ^{*1}	HTD0~HTD1023	1024	BC80~C07F	48256~49279
	HCD ^{*1}	HCD0~HCD1023	1024	C080~C47F	49280~50303
	HSCD ^{*1} HSCD0~HSCD39		40	C480~C4A7	50304~50343
	FD ^{*2}	FD0~FD8191	8192	C4C0~E4BF	50368~58559
	SFD ^{*2})**2 SFD0~SFD4095		E4C0~FC2F	58560~64559
	FS ^{∞2} FS0~FS47		48	F4C0~F4EF	62656~62911

Note :

*1: The area marked with *1 is the power failure retention area; Flash area marked with *2.

*2: The addresses in the above table are used when the PLC is used as the lower computer and Modbus RTU or Modbus ASCII protocol communication is used. Generally, the upper computer is: configuration/touch screen/PLC.

*3: If the upper computer is a PLC, write the program according to the Modbus-RTU or Modbus-ASCII protocol.

%4: If the upper computer is SCADA or HMI, there are two situations: the first one has a Xinje driver, such as Xinje HMI, which can be directly written using the PLC internal software components (Y0/M0); The second type does not have a Xinje driver, so choose Modbus-RTU or Modbus-ASCII protocol, and then use the addresses in the above table to define data variables.

※5: The input and output points are in octal. Please calculate the corresponding input and output point Modbus address according to octal. For example, the Modbus address corresponding to Y0 is H6000, the Modbus address corresponding to Y10 is H6008 (not H6010), and the Modbus address corresponding to Y20 is H6010 (not H6020).

%6: When the Modbus address exceeds K32767, it needs to be represented in hexadecimal and the address needs to be preceded by "0". For example, the Modbus address of HD0 is 41088 in decimal (beyond K32767), and K41088 cannot be written to the software, so it needs to be represented as H0A080 in hexadecimal.

%7: Modbus address calculation for X and Y, taking X as an example, the Modbus address calculation for Y is the same as for X.

X0: 20480 X10: 20480+8 X20: 20480+16 X30: 16384+24.... X10000: 20736 X10010: 20736+8 X10020: 20736+16....

X1020: 20800 X10210: 20800+8 X10220: 20800+16....

3-2-3. MODBUS function code

Function	Function	Descriptions
code		
01H	Read coil	Read 0X address, max quantity is 2000
02H	Read input coil	Read 1X address, max quantity is 2000
03H	Read holding register	Read 4X address, max quantity is 125
04H	Read input register	Read 3X address, max quantity is 125
05H	Write single coil	Write single 0X address
06H	Write single register	Write single 4X address
0FH	Write multiple coils	Write 0X address, max quantity is 1976
10H	Write multiple registers	Write 4X address, max quantity is 123

Ethernet model PLC supports the following Modbus communication function codes:

3-3. Modbus TCP graphics configuration

3-3-1. Overview

ModbusTCP, as a standard protocol for industrial communication, is widely used on site. Ethernet models integrate the MODBUS-TCP protocol, including servers and clients. In order to easily achieve communication data interaction with MODBUS-TCP devices, for XDH/XLH models, V3.7.3 and above firmware can support MODBUS-TCP graphical configuration. If users need to flexibly use MODBUS-TCP to achieve specific requirements, or if the device to be connected does not support MODBUS-TCP and only supports free format TCP/IP, communication data interaction can be achieved by establishing sockets. The establishment of sockets does not conflict with the establishment of connection resources through graphical configuration.

3-3-2. Modbus TCP master station configuration

Modbus TCP master (client) configuration for XDH/XLH models, supports establishing connections with 32 Modbus TCP slaves (server) simultaneously, and establishes a maximum of 3000 connection instructions for sharing with 32 Modbus TCP slaves (server). The process of establishing connection configuration is as follows:

Open the XDPpro software, click Modbus TCP in the left project bar to enter the configuration interface.
💷 Xinje PLC Program Tool			- 🗆 ×
File Edit Search View Ogli	e <u>Configure</u> <u>Option Window</u> <u>Help</u> 1	 日合塩酸酸酸+m 1、米 To To To To To To To 	
Project 4	PLC1 - Ladder		• ×
Module			
BD	ModbusTCP Config		
4GBOX WBOX BystemConfig LC Communication EpcAdapter EpcAdapter EpcAdapter	⊡ Master station ⊥PLC Master		
Motion control(H movement)	Add Delete Copy Attribute		
Axis debug			
PLC Status PLC Status PLC Status PLC Status PLC Project Message BD Details BD Details Details Cock Details Cock Details Fror Details Fror Details Fror Details			
A Struction Class Project	Number of connections built: 0/32 Number of instructions built: 0/3	000	Read PLC Write PLC

1. Modbus TCP Graphical Configuration Table

ModbusTCP Config	New built Incent	Delete Meyelle Meye	our Class Invest OutPast				×
PIC Master	Rumber Name S	Derete Move Dy Move D	Trigger condition Function co.	de Slave address	Slave offset	Count	Map address
Add Delete Copy Attribute 2 Slave configuration -[0]192.168.6.1502 -[1]192.168.6.3502 -[2]192.168.6.3502 3			0				
Number of connections built: 3/32	Number of instruction	ns buit: 0/3000 5		6	ead PLC Write PL	сок	Cancel

【Area 1】: Display the configuration information of the master station;

[Area 2]:

٠	Support the addition, deletion, replication, and attribute functions of a slave node
Add	Add a default slave node at the bottom and position the cursor to the added slave
	node.
Delete	Delete the selected slave node by the user. Clicking this function is invalid when the
	current tree node is empty.
Сору	The user clicks the copy button to copy a selected slave information
	(attribute+instruction configuration information) and automatically paste it to the
	bottom of the tree node. At the same time, the IP address is changed to the default IP

Support the addition, deletion, replication, and attribute functions of a slave node

tribute Open the Modbus TCP s	ettings interface of the selected slave node
ModbusTCP Set	×
Device selection:	Xinje PLC Equipment \checkmark
IP Address:	192.168.6.3
Port number:	502
Timeout time(ms):	500
Number of retransmissions:	1
Enable control software:	
Connection flag bit :	
	Ok Cancel

• The following contents can be set in the Modbus TCP configuration interface

Device selection	Xinje PLC and other Modbus equipment; Default Xinje PLC device
IP address	The IP address of the target PLC; Default 192.168.6.1, starting from 1, the next
	item defaults to the previous address +1
Port number	Fill in 502 by default
Time out	Default setting 500ms, range: 10-65535
Enable control	By default, it is not enabled. Enabling can set the coil control of the PLC.
software	When not enabled: PLC automatically establishes a TCP connection to the
component	target IP after running;
	When enabled: Only bit registers are supported, and TCP connections are only
	established to the target IP when the coil is set ON. When the conditions are
	not met, close the TCP connection
Connection flag	Store the result of the successful connection of this device in the corresponding
bit	connection flag bit register

[Area 3] : Display slave configuration information;

[Area 4]:

- Support users to select relevant instruction configuration functions for slave nodes, including create, insert, delete, move up, down, clear, import, and export.
- Display the command information of the selected slave node.

Number	Name	Slave number	Trigger mode	Trigger condition	Function code	Slave address	Slave offset	Count	Map address
0	slave	1	Circulate(ms)	1000	Read register	D	0	1	DO
1	slave	1	Circulate(ms)	1000	Read register	D	0	1	DO
2	slave	1	Circulate(ms)	1000	Read register	D	0	1	DO
3	slave	1	Circulate(ms)	1000	Read register	D	0	1	DO

[Area 5]: Monitor the number of connections currently established and the number of instructions established

[Area 6]: Supports functions such as read PLC, writ PLC, and save data (confirm, cancel).

2. Add instructions

Number	Name	Slave number	Trigger mode	Trigger condition	Function code	Slave address	Slave offset	Count	Map address
0	slave	1	Circulate(ms)	1000	Read register	Ю	0	1	DO
1	slave	1	Circulate(ms)	1000	Write register	D	200	1	D100
2	slave	1	Circulate(ms)	1000	Read register	D	100	1	D200

- Name: The name of the current mapping instruction, which can be modified by the user;
- Slave station number: default 1, range 0-247;
- Triggering method: cycle (ms) and conditional triggering
 - Cycle (ms): When the triggering method is cycle, the value in the triggering condition is the cycle period, in ms; Range: 0-2 ^ 32-1;
 - Conditional trigger: When the trigger method is conditional trigger, the trigger condition is SM/M/HM coil or bit of word. Default to edge triggering, implemented by the PLC.
- Triggering conditions: Depending on the triggering method, check the validity of this item when "confirmed";
- Function code
 - When the user selects a Xinje PLC as the device

Read coil	The maximum number of reading coils supports
	2000
Write coil	The maximum number of write coils is 1960
Read register	The maximum number of read registers
	supported 125
Write register	The maximum number of write registers supports
	122

•	When the use	er selects other	MODBUS	devices
---	--------------	------------------	--------	---------

Read coil (01H)	Read 0X type addresses, maximum quantity 2000
Read input coil (02H)	Read 1X type addresses, maximum quantity 2000
Read register (03H)	Read 4X type addresses, maximum quantity 125
Read input register (04H)	Read 3X type addresses, maximum quantity 125
Write single coil (05H)	Write a single 0X type address
Write single register	Write a single 4X type address
(06H)	
Write multiple coils	Write 0X type addresses, with a maximum number
(0FH)	of 1960
Write multiple registers	Write 4X type addresses, with a maximum number
(10H)	of 122

Slave station address space

If the current slave station is a Xinje PLC, this is the register type corresponding to the function code. The reference settings are as follows:

- Read and write coils, pull-down options: M, X, Y, HM, S, SM, T, C, ET, SEM, HS, HT, HC, HSC;
- Read and write registers, pull-down options: D, HD, ID, QD, SD, TD, CD, ETD, HSD, HTD, HCD, HSCD, FD, SFD, FS.
- Quantity: The length of data that can be read or written, with a default of 1. The

maximum length of data that can be read or written depends on the above function code.

• Mapping address: coil status, cache address in the master station. Default is D0.

3-3-3. Modbus TCP graphical application

By using Modbus TCP graphical configuration function, automatic connection and data exchange between two PLCs can be established upon power on. Taking the communication between two XDH-60T4 as an example, the IP address of PLC 1 (client) is 192.168.6.10, and that of PLC 2 (server) is 192.168.6.6.

The operation of this case is as follows:

1. The client performs a register write operation and writes the data from the 10 registers of the client D0-D9 to the 10 registers of the server HD0-HD9 in a loop of 500ms as a trigger;

2. The client performs a register reading operation and reads the data from the 10 registers of D100-D109 on the server into the 10 registers of HD100-HD109 on the client using the trigger method (M600);

3. The client performs a write coil operation and writes the status of the 10 coils of the client M0-M9 to the ten coils of the server HM0-HM9 in a loop of 500ms as the trigger method;

4. The client performs a coil reading operation by triggering (M601) to read the status of the 10 coils of the server's M0-M9 into the 10 coils of the client's HM0-HM9.

(1) Configure the IP address and related configuration information of the slave station as follows:

⊟ Master station └ PLC Master		
	ModbusTCP Set	×
	Device selection:	Xinje PLC Equipment V
	IP Address:	192.168.6.6
Add Delete Copy Attribute	Port number:	502
- Slave configuration	Timeout time(ms):	500
	Number of retransmissions:	1
	Enable control software :	M200
	Connection flag bit :	M201
		Ok Cancel

(2) Create instructions for the four data interaction operations mentioned above, as follows:

Master station	New-	New-built Insert Delete Move Up Move Down Clear Import OutPort								
-FEC Master	Number	Name	Slave number	Trigger mode	Trigger condition	Function code	Slave address	Slave offset	Count	Map address
	0	slave	1	Circulate(ms)	500	Write register	Ю	0	10	DO
	1	slave	1	Trigger	M600	Read register	D	0	10	HDO
	2	slave	1	Circulate(ms)	500	Write coil	нм	0	10	MO
	3	slave	1	Trigger	M601	Read the coil	м	0	10	HMO

(3) Check the status of the connection between the client and server, as well as information on data interaction, as shown in the following figure:

监控窗口 • 添加	修改 删除	全部删	除 上移 下移	; 置顶 置底
名称	监控值	土田	B広急さ1位11-1	注释
- • M200	ON	BIT	位	(中台)空制校元件
- M200	ON	BIT	位	连接标志位
- 0 D0	1	INT	苗字	安户端宫撮作数据省栅址
- 0 D1	-2	INT	——	
- D2	3	INT	—— 单字	
• D3	4	INT	単字	
- • D4	5	INT	単字	
- • D5	6	INT	单字	
◆ D6	7	INT	単字	
- • D7	8	INT	単字	
- D8	9	INT	単字	
- • D9	10	INT	単字	
- HDO	12	INT	单字	客户端读操作数据首地址
- HD1	13	INT	单字	A TRANSPORT
- HD2	14	INT	单字	
- ID3	15	INT	单字	
- > HD4	0	INT	单字	
- • HD5	0	INT	单字	
- 🔷 HD6	0	INT	单字	
- VHD7	0	INT	单字	
- 🔷 HD8	0	INT	单字	
- 🔷 HD9	0	INT	单字	
- 🔷 M0	OFF	BIT	位	客户端写线圈状态首地址
-🔷 M1	OFF	BIT	位	
- 🔷 M2	OFF	BIT	位	
- 🔷 M3	OFF	BIT	位	
-🔷 M4	OFF	BIT	位	
- • M5	OFF	BIT	位	
-🔷 M6	OFF	BIT	位	
- 🔷 M7	OFF	BIT	位	
- 🔷 M8	OFF	BIT	位	
-🔷 M9	OFF	BIT	位	
-🔷 HMO	OFF	BIT	位	客户端读线圈状态首地址
- 🔷 HM1	OFF	BIT	位	
- 🔷 HM2	OFF	BIT	位	

3-4. Free format protocol

Freedom communication based on Ethernet is divided into two categories: TCP and UDP, Ethernet model using TCP communication can be used as a TCP client (TCP client), can also be used as a TCP server (TCP server).

- 1. as a TCP client, take the initiative to establish a TCP connection with the TCP server, and bind socket ID.
- 2. as the TCP server, waiting for the TCP client and establish a TCP connection, and bind socket ID.
- 3. using UDP, listening to the specified local port, and bind socket ID.

Based on the above three forms, which can realize the freedom of Ethernet communication. Freeform communication in the form of a block of data to transmit data, restricted by PLC cache, a single to send and receive data volume of 1000 bytes.

Based on the above three forms, it can realize the free communication of Ethernet. Free format communication transfers the data in the form of data block, be restricted by PLC cache, single-time sending and receiving data volume is 1000 bytes.

Free format communication parameters:

Data buffer mode: 8-bit, 16-bit

- 1. 8-bit buffer communication: the high byte of the register is invalid, PLC only uses the low byte of the register to send and receive data.
- 2. 16-bit buffer communication: for the received data, PLC saves the low byte first, then saves the high byte; for the sending data, PLC sends the low byte first, then sends the high byte.
- 3. When the received data package length is larger than setting length, data will be stored as 16-bit buffer mode.

4 Ethernet communication instruction

4-1. Ethernet communication instruction overview

Ethernet communication instructions include: communication task opening and closing, send/receive data, MODBUS TCP. When using Ethernet instruction, please follow the following steps:

(1) open communications task: confirm the communication protocols and communication type, configure communication parameters, to create a TCP connection/UDP port listening, and bind socket ID.

(2) to realize the data communication: open successful communications task, achieve free Ethernet communication or MODBUS TCP data communications.

(3) close communications task: after communicating with target device, or TCP connection is abnormal, it needs to close communication tasks.

4-1-1. Create TCP connection/UDP port listening [S_OPEN]

1. Overview

Communication task creates the instruction, use together with abort communication task instruction S_CLOSE.

Create TCP connection /UDP port listening [S_OPEN]								
16-bit	S_OPEN	32-bit	-					
instruction		instruction						
Execution	Edge triggered	Suitable	XD3E, XD5E, XDME, XDH, XG,					
condition		model	XL5E, XL5N, XLME, XL5H,					
			XLH					
Firmware	V3.5.3 and up	Software	V3.5.3 and up					

2. Operand

Operand	Function	Туре
S1	Socket ID	16-bit, BIN
S2	Communication type	16-bit, BIN
S3	Local device communication mode	16-bit, BIN
S4	Parameter block start address	16-bit, BIN
S5	Flag start position	Bit

3. Suitable soft component

word	operand	System con					constant	Mo	dule					
-		D	FD	E	D	TD	CD	DX	DY	DM	DS	K/H	D	QD
	S1	•										•		
	S2	•										•		
	S3	•										•		
	S4	•												
Bit	operand				Svs	tem								
Bit operand	operana	v	v	M*	Sy5	т*	<i>C</i> *	Dom						
	S5	Λ	1	•	5	1	C	Dimi						

*Note: D means D HD ; TD means TD HTD ; CD means CD HCD HSCD HSD; DM means DM DHM;

DS means DS DHS. M means M HM SM; S means S HS; T means T HT; C means C HC.

Function and action



- Create the communication task, when M0 rising edge is coming, the instruction will create one TCP connection or open UDP port listening once.
- S1: socket ID, range: K0~K63. Note: the socket quantity cannot be more than 64, TCP quantity cannot over 32, UDP quantity cannot be more than 32.
- S2: communication type, range: K0, K1. K0 is UDP, K1 is TCP.
- S3: communication mode. Range: K0, K1. K0 is server, K1 is client.
- S4: parameter block start address, occupy 9 registers from S4 to S4+8.
- S5: flag start position, occupy 10 coils from S5 to S5+9.
- This instruction can be set through the following window

Note: 1. The server needs to first open the socket and wait for the client to connect,

otherwise the socket may not be established successfully.

2. The UDP multicast function is only supported for Ethernet PLC firmware version 3.7.2 and above, and XDH/XLH series firmware 3.7.3 and above versions.

= 1	PID Config Pulse Config	۲					
-	High Speed Co	unt Config					
1	Ethernet Conne	ction Config					
	Modbus Tcp Co	onfig					
			Ethernet Co	nnectio	on Config		
		S_OPEN	Parameter S	letting			3
Basic Setting							
Basic Setting Socket ID	KD	Communication	TCP(K1)	~	Mode Selection	Client(K1)	~
Basic Setting Socket ID Reg Start Position	K0 HD0	Communication type Rag Start Position	TCP(K1) M0	•	Mode Selection The "Basic Settin after downloading	Client(K1) ngs" program wil g!	▼ take effect
Basic Setting Socket ID Reg Start Position	K0 HD0	Communication type Rag Start Position	TCP(K1) M0	•	Mode Selection The "Basic Settin after downloading	Client(K1) Igs" program wil I	¥ Itake effect
Basic Setting Socket ID Reg Start Position	K0 HD0	Communication type Rag Start Position Buffer type	TCP(K1) M0 8位	~	Mode Selection The "Basic Settin after downloading Timeout(10ms)	Client(K1) ngs" program wil 1	✓ take effect
Basic Setting Socket ID Reg Start Position	K0 HD0 0	Communication type Rag Start Position Buffer type . 0 Destination Port	TCP(K1) M0 8f立	×	Mode Selection The "Basic Settin after downloading Timeout(10ms)	Client(K1) ngs" program wil ! 0	V I take effect

Note: the parameters in the red frame will be effective after power on the PLC again.

• Ethernet error flag SM1921 is ON when communication is abnormal, the error information will be stored in SD1920 and SD1921, please refer to chapter 4-3.

Take above image as an example, the address starting from HD0 and flag address starting from M0 are shown as below:

SOpen configuration instruction help interface							
Local Port	HD0	Connection start mark	MO				
Target IP Sec 1 (e.g. :192)	HD1 High Byte	Linked mark	M1				
Target IP Sec 2 (e.g. :168)	HD1 Low Byte	Sending mark	M2				
Target IP Sec 3 (e.g. :0)	HD2 High Byte	Passed mark	M3				
Target IP Sec 4 (e.g. :1)	HD2 Low Byte	Receiving mark	M4				
Destination Port	HD3	Received mark	M5				
The data buffering	HD4	Closing mark	M6				
Receiving Timeout	HD5	Modbus TCP communication mark	M7				
The reserved	HD6	TCP exception mark	M8				
Actual number of bytes received(Byte)	HD7	Error mark	M9				
Error Type	HD8						

Parameter explanation:

,	1		71		71		
Communication	Local	destination	Destination	Buffer	Buffer Timeout		Error
type	port	IP	port	type	Timeout	bytes	code
TCP client	-	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
TCP server	\checkmark	-	-	\checkmark	\checkmark	\checkmark	\checkmark
UDP	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark

The communication task created by S_OPEN is divided into three categories: TCP client, TCP server, UDP. The parameters used by the three types are different, please refer to below table.

1. Local port

The range is 1 to 60000, port 502 and 531 is special port which can not be used. Local port only can be used by one communication task.

2. Destination IP

The target IP refers to the IP address of the target communication device, with a value range of 0-254, which is in the same subnet as the local machine.

3. Destination port

The net port no. of target device. The range is 1 to 65535. The port must be 502 for modbus tcp communication.

4. Data Buffer mode

When the value is 0, it is 8-bit mode. When the value is non-zero, it is 16-bit mode. The actual received data packet length is received based on the corresponding set buffer length.

5. Timeout

The time from PLC requests data receiving to the receiving data ends. The range is 0 to 65536. The unit is 10ms. 0 means the timeout is disabled, it will continue receiving data. Non-zero means the timeout function is enabled. The receiving timeout is effective for S_RCV and M_TCP.

If the timeout is set to 300ms, it will wait for 300ms when the request begins, and terminate at once when the data is received successfully. If it hasn't received data over 300ms, the present instruction will end and report the receiving timeout error.

Note: When the receive timeout time is set to 0 in versions V3.7.3 and above, M_TCP will default to a receive timeout time of 10 seconds, and S_RCV will default to no receive timeout time.

6. TCP keep alive

(1) the value is 0, TCP keep alive function is not enabled.

(2) the value is non-zero, TCP keep alive function is enabled.

Connection is in the inactive state over a period of time, when the keep alive function is enabled, it will send keep alive detection to the object, if the sender did not receive the response message, then the other host will be confirmed as unreachable. Triggering time is $1 \sim 5$ min, when it is abnormal, TCP abnormal flag is set on.

Note: The TCP keep alive function is only supported for Ethernet based PLCs with firmware versions 3.7.2 and above.

7. Data receiving mode

Automatic reception: If the other party sends too quickly during reception, the data that is not received will be automatically discarded; Not receiving or receiving timeout will also discard the data sent by the other party.

8. Receiving data length

Execute S_RCV instruction, the actual length of received data, in bytes.

9. Error code

The error message when Ethernet free format communication and Modbus TCP communication are abnormal, please refer to chapter 4-4.

10. Flag bit

The functional description of communication related flag bits is shown in the table below: (Address description starts with Mn)

Bit address	Flag bit	Function
Mn	Connecting	Creating the connection, M (n) is ON
M (n+1)	Connected	Creating connection completed, M (n+1) is ON
M (n+1)	Sending	Data is sending, M (n+2) is ON
M (n+3)	Sent	Sending data completed, M (n+3) is ON
M (n+4)	Receiving	Data is receiving, M (n+4)is ON
M (n+5)	Received	Data receiving completed, M (n+5) is ON
M (n+6)	Closing	The present connection is closing, M (n+6) is ON
M (n+7)	MODBUS TCP	When executing M_TCP instruction, M (n+7) is ON
	communicating	
M (n+8)	TCP abnormal	TCP connection is abnormal, M (n+8) is ON
M (n+9)	Error flag	Communication is error, M (n+9) is ON

4-1-2. Communication termination [S_CLOSE]

1. Instruction overview

Communication termination instruction, please use together with S_OPEN.

Communica	tion termination [S_CLOSE]		
16-bit	S_CLOSE	32-bit	-
Execution	Edge triggering	Suitable	XD3E, XD5E, XDME, XDH, XG,
condition		model	XL5E, XL5N, XLME, XL5H,
			XLH
Firmware	V3.5.3 and up	Software	V3.5.3 and up

2. Operand

Operand	Function	Туре
S1	Close socket ID	16-bit, BIN

3. Suitable soft component

word	operand					Syste	m				Constant	Мс	dule
		D	FD	ED	TD	CD	DX	DY	DM	DS	K/H	ID	QD
	S1	٠									•		

*Note: D means D HD ; TD means TD HTD ; CD means CD HCD HSCD HSD; DM means DM DHM; DS means DS DHS.



- Terminate the communication task when the rising edge of M0 is coming. Note: this instruction must be used together with S_OPEN.
- S1: the socket ID which needs to close, the operand can be register or constant, the range is K0~K63.
- After this instruction is executed, the instruction M_TCP, S_SEND, S_RCV based on this socket ID cannot run anymore.

4-1-3. Free format communication send [S_SEND]

1. Instruction overview

Free format	communication	send instruction	needs to us	se together	with S	OPEN	and S	CLOSE.
				0		_		

Free format	communication send [S_SEND]		
16-bit	S_SEND	32-bit	-
Execution	Edge triggering	Suitable	XD3E, XD5E, XDME, XDH, XG,
condition		model	XL5E, XL5N, XLME, XL5H,
			XLH
Firmware	V3.5.3 and up	Software	V3.5.3 and up

2. Operand

Operand	Function	Туре
S1	Socket ID	16-bit, BIN
S2	Send data local register head address	16-bit, BIN
S3	Send data quantity	16-bit, BIN

3. Suitable soft component

word	operand					Syste	m				Constant	Mo	dule
		D	FD	ED	TD	CD	DX	DY	DM	DS	K/H	ID	QD
	S1	•									•		
	S2	•											
	S3	٠									•		

*Note: D means D HD ; TD means TD HTD ; CD means CD HCD HSCD HSD; DM means DM DHM; DS means DS DHS.



• Free format communication send instruction, it will send data when the M0 rising edge is coming.

Note: this instruction must be used together with S_OPEN and S_CLOSE.

- S1: socket ID, the operand can be register or constant, the range is K0~K63
- S2: local register sending head address
- S3: send data quantity, the operand can be register or constant
- Please input this instruction in the ladder chart
- When using, pay attention to the data buffer type in the S_OPEN instruction in the socket ID (16 bits/8 bits).
- When the buffer bit is 8 bits, only the low byte data of the register should be sent. For example, to send the low byte data in registers D100 to D107, S3 should be set to 8.
- When the buffer bit is 16 bits, both high and low byte data of the register will be sent. For example, to send high and low byte data from D100 to D107, S3 should be set to 16, and when sending, the low byte should be in front of the high byte.

4-1-4. Free format communication receive [S_RCV]

1. Instruction overview

Free format communication receive instruction needs to use together with S_OPEN and S CLOSE.

Free format	communication receive [S_RCV]		
16-bit	S_RCV	32-bit	-
Execution	Normally ON/OFF, edge	Suitable	XD3E, XD5E, XDME, XDH, XG,
condition	triggering	model	XL5E, XL5N, XLME, XL5H,
			XLH
Firmware	V3.5.3 and up	Software	V3.5.3 and up

2. Operand

Operand	Function	Туре
S1	Socket ID	16-bit, BIN
S2	Receive data local register head address	16-bit, BIN
S3	Receive data quantity	16-bit, BIN

3. Suitable soft component

word	operand					Syste	m				Constant	Mo	dule
		D	FD	ED	TD	CD	DX	DY	DM	DS	K/H	ID	QD
	S1	•									•		
	S2	•											
	S3	•									•		

*Note: D means D HD ; TD means TD HTD ; CD means CD HCD HSCD HSD; DM means DM DHM; DS means DS DHS.

Function and action



- Free format communication receive instruction, it will receive data when the M0 rising edge is coming.
 Note: this instruction must be used together with S OPEN and S CLOSE.
- _ _
- S1: socket ID, the operand can be register or constant, the range is K0~K63

- S2: local register receiving head address
- S3: receive data quantity, the operand can be register or constant
- Please input this instruction in the ladder chart
- When using, pay attention to the data buffer type in the S_OPEN instruction in the socket ID (16 bits/8 bits).
- When the buffer bit is 8 bits, only the low byte data of the register should be sent. For example, to send the low byte data in registers D100 to D107, S3 should be set to 8.
- When the buffer bit is 16 bits, both high and low byte data of the register will be sent. For example, to send high and low byte data from D100 to D107, S3 should be set to 16, and when sending, the low byte should be in front of the high byte.

4-1-5. MODBUS communication [M_TCP]

1. Instruction overview

When PLC is client, receive and send data in modbus tcp protocol. It can be used together with S OPEN and S CLOSE.

MODBUS 7	CCP communication [M_TCP]		
16-bit	M_TCP	32-bit	-
Execution	Edge triggering	Suitable	XD3E, XD5E, XDME, XDH, XG,
condition		model	XL5E, XL5N, XLME, XL5H,
			XLH
Firmware	V3.5.3 and up	Software	V3.5.3 and up

2. Operand

Operand	Function	Model
S1	Remote station no.	16-bit, BIN
S2	Modbus communication function code	16-bit, BIN
S3	Target head address	16-bit, BIN
S4	Register or coil quantity	16-bit, BIN
S5	Local head address	16-bit, BIN
S6	Socket ID	16-bit, BIN

3. Suitable soft component

Word	operand					Syste	m				Constant	Mo	dule
		D	FD	ED	TD	CD	DX	DY	DM	DS	K/H	D	QD
	S1	•									•		
	S2	•									•		
	S3	•									•		
	S4	•									•		
	S5	•											
	S6	•									•		

*Note: D means D HD ; TD means TD HTD ; CD means CD HCD HSCD HSD; DM means DM DHM; DS means DS DHS.

Function and action



- MODBUS TCP communication instruction, it will Modbus TCP communicate once when M0 rising edge is coming.
- S1: remote communication station no., the range is K0~K247
- S2: MODBUS communication function code
- S3: target head address, it is Modbus communication address.
- S4: communication data quantity
- S5: local head address
- S6: socket ID, specify the TCP connection, the target port must be 502.
- This instruction must be used together with S_OPEN and S_CLOSE.
- M_TCP is only effective when PLC is client, and receives and sends the data of Modbus TCP protocol.

Note: As a server, ModbusTCP has a port number of 502 and does not require writing communication instructions. The client can establish a socket and write communication instructions.

• This instruction needs to set through the following window



		MO	ubus rep (lonngaration	
Socket ID	K1	∽ <mark>S6</mark>	Local Strat Address	M100	S5
Mo <mark>d</mark> bus TCP					
Station No.	K1	✓ S1	Function Code	0x01 Read the coil	✓ S3
Data Address	К0	S2	Count	К1	S4
					OK Cancel

Function code:

Value	Function code	Value	Function code
K1	Read the coil	K3	Read the register
K2	Read the input discrete magnitude	K4	Read input register
K5	Write single coil	K6	Write single register
K15	Write multiple coil	K16	Write multiple register

4-1-6. Ethernet communication example

Example 1:

By using the following program, PLC can automatically create three forms of communication tasks: TCP client, TCP server, and UDP after power on, and achieve data transmission and reception based on each communication task. The IP address of PLC 1 is 192.168.1.12, and the IP address of PLC 2 is 192.168.1.6.

Note: The server needs to first open the socket and wait for the client's connection, otherwise the socket may not be established successfully.

Program operation:

(1) After PLC 1 is powered on, it actively establishes a TCP connection to the TCP server service port 1111 of PLC 2 as a TCP client and binds a socket ID of 1. After the connection is successfully established, it sends the low 8-bit of D1000~D1549 to PLC 2 D2600~D3149, while continuously receiving data from PLC 2 D2000~D2399 and storing it in the low 8-bit of registers D1600~D1999. When a TCP connection encounters an exception or the sender does not receive a response message within the set live time (where the live time is set to 2 seconds), the TCP connection is actively closed and rebuilt.

Due to the varying number of Ethernet ports in different series of PLCs, please distinguish which Ethernet port the network cable is connected to when using communication related coils SM1902 or SM1903. (SM1902 is the symbol for connecting network devices, used in the first network port of a dual port model or in a single port model to connect to a switch/router/other network device. SM1903 is the symbol for connecting network devices, used in the second network port of a dual port model to connect to a switch/router/other network port of a dual port model to connect to a switch/router/other network port of a dual port model to connect to a switch/router/other network port of a dual port model to connect to a switch/router/other network device).

PLC1 program:

	PLS M0
Nor mally ON coil	
SM1901 SM1902 M101 SM12 HM0	S_OPEN K1 K1 K1 HD100 M100
network device connected coil coil	
M101 M1	S_SEND K1 D1000 K550
Socket 1 Socket 1 send data	
M2	S_RCV K1 D1600 K400
Socket 1 receive data	
	S CLOSE K1
Close	5_02002 M
M109	
Socket	
M108	
Socket 1 TCP	
error SM1902	
-W	
Con nect device	
НМО	

Build connection

Basic Setting					
Socket ID	K1 ~	Communication type	TCP(K1) ~	Mode Selection	Client(K1) ~
Reg Start Position	HD100	Flag Start Position	M100	The "Basic Setting after downloading!	ps" program will take effect
Local Port	0	Buffer type	8 bit ~] Timeout(10ms)	0
Destination IP	192.168.1.6	Destination Port	1111	AcceptMode	AutoMode 🗸 🗸
Kana Aliva (C)	0	Used Space:	HD100-HD109 M10	0-M109	

The configuration information for the client socket **S_OPEN** is as follows:

PLC 2 program:



Build connection

PEN Parameter	Setting				ſ
Basic Setting					
Socket ID	K1 ~	Communication type	TCP(K1)	✓ Mode Selection	Server(K0) V
Reg Start Position	HD100	Flag Start Position	M100	The "Basic Settin after downloading	gs" program will take effect !
Local Port	1111	Buffer type	8 bit	✓ Timeout(10ms)	0
Destination IP	0.0.0	 Destination Port	0	AcceptMode	AutoMode 🗸 🗸
Keep-Alive(S)	Þ E	Used Space:	HD100-HD109,	M100-M109	

The configuration information for server socket S_OPEN is as follows

(2) After PLC 1 is powered on, it actively listens to port 1001 as a TCP server and waits for the TCP client device of PLC 2 to establish a TCP connection and bind a socket ID of 2. After the connection is successfully established, it sends the low 8-bit of D3000-D3549 to the connected device PLC 2, while continuously receiving data from the connected device PLC 2. The data is stored in the low 8-bit of registers D3600-D3999. When a TCP connection encounters an exception or the sender does not receive a response message within the set live time (where the live time is set to 2 seconds), the TCP connection is actively closed and rebuilt.

PLC 1 program:

SMO	
	PLS M10
Normally	
ON coil	
SM1001 SM1002 M201 SM12 UM20	
	S OPEN K2 K1 K0 HD200 M200
Turkini	
network Line Socket 2 100ms Build	
device connected coll connection	
M201 M20	S SEND K2 D2000 K550
	3_SEND K2 D3000 K350
Socket 2 Socket 2	
connected data	
M21	
	S_RCV K2 D3600 K400
data	
	S CLOSE K2
Close socket	5_02002 112
when power on	
M209	
Socket	
2 error	
M208	
Socket 2	
ТСР	
error	
SM1902	
Connect	
uevice	
HM30	
Build	

connection

Basic Setting					
Socket ID	К2 ~	Communication type	TCP(K1) ~	Mode Selection	Server(K0) 🗸
Reg Start Position	HD200	Flag Start Position	M200	The "Basic Setting after downloading!	s" program will take effect
Local Port	1001	Buffer type	8 bit 🗸	/ Timeout(10ms)	0
Destination IP	0.0.0.0	Destination Port	0	AcceptMode	AutoMode 🗸 🗸
	2			00.0000	

The configuration information for server socket **S_OPEN** is as follows

PLC 2 program:



connection

PEN Parameter	Setting				?
Basic Setting					
Socket ID	К2 ~	Communication type	TCP(K1)	Mode Selection	Client(K1) ~
Reg Start Position	HD200	Flag Start Position	M200	The "Basic Setting äfter downloading	gs" program will take effect
Local Port	1001	Buffer type	8 bit	✓ Timeout(10ms)	0
Destination IP	192 . 168 . 1 . 12	Destination Port	1001	AcceptMode	AutoMode 🗸 🗸
Keep-Alive(S)	2	Used Space:	HD200-HD209,M	200-M209	
			Read From PLC	Write To PLC	OK Cancel

The configuration information for client socket S_OPEN is as follows

(3) After powering on PLC 1, UDP communication is used with an IP address of 192.168.1.12. The local port is set to 1002, the target IP is 192.168.1.6, and the target port is 3000. The socket ID is bound to 3. After the connection is successfully established, the low 8-bit of D4000-D4549 are sent to PLC 2, and the data from PLC2 is continuously received and stored in registers D4600~D4999. When UDP unicast encounters abnormal connections, it actively closes the UDP unicast connection and reconstructs the connection.

PLC 1 program:

SM0	
	PLS M30
Nor mally ON coil	
SM1901 SM1902 M401 SM12 HM30	S OPEN K3 K0 K1 HD400 M400
Initial Connect Socket 3 100ms Build device connected coil connection	
M401 M32	S_SEND K3 D4000 K550
Socket 3 Socket 3 connected send data	
	S_RCV K3 D4600 K400
Socket 3 receive data M30	
Class see list	S_CLOSE K3
when power on M409	
Socket 3 error	
M408	
Socket 3 TCP error	
SM1902	
Connect device	
HM30	
Build	

connection

Basic Setting					
Socket ID	К3 ~	Communication type	UDP(K0)	Mode Selection	Client(K1) ~
Reg Start Position	HD400	Flag Start Position	M400	The "Basic Settings after downloading!	" program will take effect
Local Port	1002	Buffer type	8 bit	✓ Timeout(10ms)	0
Destination IP	192.168.1.6	Destination Port	3000	AcceptMode	AutoMode 🗸 🗸
	0		and a second	Language .	

The configuration information for UDP socket S_OPEN is as follows

PLC 2 program:

SMO	
	PLS M30
Normally	
ON coil	
SM1901 SM1902 M401 SM12 HM30	
	S_OPEN K3 K0 K1 HD400 M400
Initial Connect Socket 3 100ms Build network device connected coil connection	
M401 M32	S SEND K3 D4000 K400
Socket 3	_
Socket 3 send	
data	
M33	
<u>├</u> ─-┤├ <i>──</i> ────	S_RCV K3 D4600 K550
Socket 3	
receive	
data	
M30	
<u>├</u> ── <u> </u>	S_CLOSE K3
Close socket	
when power on	
M409	
Socket	
3 error	
M408	
Socket 3	
ТСР	
error	
31 (1002	
SM 1902	
Connect	
device	

connection

Basic Setting					
Socket ID	КЗ 🗸	Communication type	UDP(K0)	Mode Selection	Client(K1) 🗸
Reg Start Position	HD400	Flag Start Position	M400	The "Basic Settings after downloading!	" program will take effect
Local Port	3000	Buffer type	8 bit	✓ Timeout(10ms)	0
Destination IP	192.168.1.12	Destination Port	1002	AcceptMode	AutoMode 🗸 🗸

The configuration information for UDP socket S_OPEN is as follows

Example 2:

Through the following program, the PLC can automatically communicate with MODBUS-TCP server devices A and B after power on. The IP address of the PLC is 192.168.1.12, the IP address of device A is 192.168.1.6, the Modbus station number is 1, the IP address of device B is 192.168.1.14, and the Modbus station number is 1.

Note: As a server, ModbusTCP does not require writing communication instructions.

Due to the varying number of Ethernet ports in different series of PLCs, when using communication related coils SM1902 or SM1903, please pay attention to distinguishing which Ethernet port is connected to the PLC by the network cable (SM1902 is the symbol for connecting network devices, used in the first Ethernet port of a dual port model or in a single port model to connect to switches/routers/other network devices). SM1903 is a symbol for connecting network devices, used in dual port models where the second network port is connected to a switch/router/other network devices.

Program operation:

(1) After the PLC is powered on, it actively establishes a TCP connection to the TCP server service port 502 of device A as a TCP client and binds the socket ID to 1. After the connection is successfully established, the value of D1000-D1019 is written to device A's 4x100-4x119 every 1 second. When a TCP connection encounters an exception or the sender does not receive a response message within the set live time (where the live time is set to 2 seconds), the TCP connection is actively closed and rebuilt.

(2) After the PLC is powered on, it actively establishes a TCP connection to the TCP server service port 502 of device B as a TCP client and binds a socket ID of 2. After the connection is successfully established, the value of D1000-D1019 is written to device B's 4x200-4x219 every 1 second. When a TCP connection encounters an exception or the sender does not receive a response message within the set live time (where the live time is set to 2 seconds), the TCP connection is actively closed and rebuilt.

Program:

Socket 1	
	PLS M1
Normally	
ON coil	
SM1901 SM1902 M101 SM12 HM0	S_OPEN K1 K1 K1 HD100 M100
Connect Socket 1 100ms Build	
Initial device connected coil connection	
network	
M101 M2	
	M_TCP K1 K16 K100 K20 D1000 K1
Seclert 1 Socket 1	
connected send	
data	
M1	
	S_CLOSE K1
Close socket	
when power on	
M109	
├ ─── │	
Socket	
1 error	
M108	
II	
Socket 1	
ТСР	
error	
SM1902	
₩	
Connect	
device	
HM1	
┝───₩────┘	
Build	
connection	

Socket 2	
	PLS M10
Nor mally ON coil	
SM1901 SM1902 M201 SM12 HM10	S_OPEN K2 K1 K1 HD200 M200
Initial Connect Socket 2 100ms Build device connected coil connection	1
M201 M11	M_TCP K1 K16 K200 K20 D1000 K2
Socket 2 connected data	
M10 The Close	S_CLOSE K2
sock et when	
power on M209	
Socket 2 error	
M208	
Socket 2 TCP error	
SM 1902	
Connect device	
HM10	
Build connection	

The co	onfigura	tion	info	ormation	for	socket	1	S	OPEN	is	as	follows:
								_	-			

Basic Setting					
Socket ID	K1 ~	Communication type	TCP(K1) ~	Mode Selection	Client(K1) ~
Reg Start Position	HD100	Flag Start Position	M100	The "Basic Setting after downloading!	s" program will take effect
Local Port	3000	Buffertype	8 bit	 Timeout(10ms) 	0
Destination IP	192 . 168 . 1 . 6	Destination Port	502	AcceptMode	AutoMode 🗸 🗸

The configuration information for socket 1 M_TCP is as follows

Modbus Tcp conf	iguration			×
Socket ID	K1 ~	Local Strat Address	D1000	
Modbus TCP				
Station No.	К1 ~	Function Code	0x10 Write multiple registers \sim	
Data Address	К100	Count	К20	
			ОК	Cancel

The configuration information for socket 2 S_OPEN is as follows

Basic Setting		6	-		
Socket ID	K2 ~	Communication type	TCP(K1)	Mode Selection	Client(K1) ~
Reg Start Position	HD200	Flag Start Position	M200	The "Basic Settings after downloading!	" program will take effect
1	2000	D.4	0 64		
Local Port	3000	Buffer type	o Dic	✓ Timeout(Tums)	0
Destination IP	192 . 168 . 1 . 🚺	Destination Port	502	AcceptMode	AutoMode 🗸 🗸
ter anticites	2	Lined Server	UD200 UD200 M2	P0CM.000	

The configuration information for socket 2 M_TCP is as follows

dbus Tcp conf	iguration			
Socket ID	K2 ~	Local Strat Address	D1000	
Modbus TCP				
Station No.	K1 ~	Function Code	0x10 Write multiple registers \sim	
Data Address	К2ро	Count	К20	

Example 3: By using the following program, the PLC can automatically create a UDP multicast communication task after being powered on. When there is an abnormality in the connection, it can actively close the UDP multicast connection and rebuild it. Implement one send multiple receive. The IP address of PLC 1 is 192.168.1.6, PLC 2 is 192.168.1.12, and PLC 3 is 192.168.1.14.

Due to the varying number of Ethernet ports in different series of PLCs, please distinguish which Ethernet port the network cable is connected to when using communication related coils SM1902 or SM1903. (SM1902 is the symbol for connecting network devices, used in the first network port of a dual port model or in a single port model to connect to a switch/router/other network device. SM1903 is the symbol for connecting network devices, used in the second network port of a dual port model to connect to a switch/router/other network port of a dual port model to connect to a switch/router/other network port of a dual port model to connect to a switch/router/other network port of a dual port model to connect to a switch/router/other network device)

Program operation:

(1) After PLC1 is powered on, UDP multicast communication is used. The target IP is set to 230.0.0.0, the target port is 7000, and the socket ID is bound to 1. After establishing a successful connection, PLC 1 sends the low 8-bit of D1000-D1499 at a frequency of 1 second. PLCs 2 and 3 continuously receive data from PLC1 and store it in registers D1000-D1499.

(2) After PLC2 is powered on, it uses UDP multicast communication, sets the target IP to 230.0.0, the target port to 7000, and binds the socket ID to 1. After establishing a successful connection, PLC 2 continuously receives data from PLC1 and stores it in the lower eight bits of registers D1000~D1499.

(3) After PLC3 is powered on, it uses UDP multicast communication, sets the target IP to 230.0.0, the target port to 7000, and binds the socket ID to 1. After establishing a successful connection, PLC 3 continuously receives data from PLC1 and stores it in the lower eight bits of registers D1000~D1499.

PLC1 program:

SMO	
	PLS M0
Nor mally ON coil	
SM 1901 SM 1902 M101 SM 12 HM0	S_OPEN K1 K2 K1 HD100 M100
Initial Connect Socket 1 100ms Build device connected coil connection	
	S_SEND K1 D1000 K500
Socket 1 send data	
M0	S_CLOSE K1
Close socket when power on M109	
Socket 1 error	
Socket 1 TCP error	
SM1902 _₩	
Connect device	
connection	

The UDP multicast S_OPEN parameter configuration is as follows:

basic setting					
Socket ID	К1 🗸	Communication type)P multicast (K2)	Mode Selection	Client(K1) 🗸 🗸
Reg Start Position	HD100	Flag Start Position	M100	The "Basic Setting after downloading!	s" program will take effec
Local Port	0	Buffer type	8 bit	✓ Timeout(10ms)	0
Destination IP	230.0.0.0	Destination Port	7000	AcceptMode	AutoMode 🗸 🗸
	1				

PLC2 program:



The UDP multicast S_OPEN parameter configuration is as follows:

-					
Socket ID	K1 ~	Communication type)P multicast (K2) 🗸	Mode Selection	Client(K1) 🗸 🗸
Reg Start Position	HD100	Flag Start Position	M100	The "Basic Settings" after downloading!	" program will take effec
Local Port	0	Buffer type	8 bit 🗸	/ Timeout(10ms)	0
	230.0.0.0	Destination Port	7000	AcceptMode	AutoMode 🗸 🗸
Destination IP	1				
PLC3 program:





Basic Setting					
Socket ID	К1 ~	Communication type)P multicast (K2) 🗸	Mode Selection	Client(K1) ~
Reg Start Position	HD100	Flag Start Position	M100	The "Basic Setting after downloading!	ps" program will take effect
		D.#	Rh#	Times t(10me)	0
Local Port	0	Buffer type	U DIL V	nineout(roms)	U
Local Port Destination IP	0	Destination Port	7000	AcceptMode	Auto Mode V

4-2. Read write communication port parameters

To ensure the normal implementation of Ethernet communication, it is recommended to use communication port parameter read/write instructions when writing communication programs. Firstly, by calling the communication parameter read instruction, the corresponding parameters on the communication port are read into the specified register group. The user then modifies the corresponding values in the register group as needed, and then writes the modified values of the register group to the corresponding communication port configuration through the communication parameter write instruction.

4-2-1. Read serial port parameters [CFGCR]

(1) Overview

Read the serial port parameters into the specified registers in the local machine.

Read the ser	al port parameters [CFGCR]		
16-bit	CFGCR	32-bit	-
instruction		instruction	
Execution	Normally ON/OFF coil, edge	Suitable	XD, XL, XG
condition	triggering	model	
Firmware	-	Software	V3.4 and up

(2) Operand

Operand	Function	Туре
D	Specify the first address of the local register	16 bits, BIN
S1	Specify the number of serial port parameters to be	16 bits, BIN
	read	
S2	Specify the serial port number to be read	16 bits, BIN

(3) Suitable soft component

operand		Word									Bit							
	System					Constant	Mo	dule	System									
	D	FD	TD	CD	DX	DY	DM	DS	K/H	ID	QD	X	Y	М	S	Т	C	Dn.m
D	•																	
S1	•	•							•									
S2	•								K									

Note: D represents D and HD; TD represents TD and HTD; CD represents CD, HCD, HSCD, HSD; DM stands for DM and DHM; DS stands for DS and DHS.

Tor DW and DHW, DS stands for DS and DHS.

M represents M, HM, SM; S represents S and HS; T represents T and HT; C represents C and HC.

(4) Function and action



• Operand S1: The number of registers occupied by reading serial port parameters, usually 8

(Ethernet port parameters are 9).

- Operand S2: Serial port number range: K0~K5. K0: COM0, K1: COM1, K2: COM2, or COM2-RS232 or COM2-RS485, K3: COM3, K4: COM4, K5: COM5, K9: Ethernet port.
- Read the 8 parameters of serial port 2 into HD0~HD7. The specific parameter names and definitions can be found in sections 4-2-4.

4-2-2. Write serial port parameters [CFGCW]

(1) Instruction overview

Write the values from the specified registers in the local machine to the specified serial port.

Write serial port parameters [CFGCW]									
16-bit	CFGCW	32-bit	-						
instruction		instruction							
Execution	Normally ON/OFF coil, edge	Suitable	XD, XL, XG						
condition	triggering	model							
Firmware	-	Software	V3.4 and up						

(2) Operand

Operand	Function	Туре
S1	Specify the first address of the local register	16 bits, BIN
S2	Specify the number of serial port parameters to be	16 bits, BIN
	written	
S3	Specify the serial port number for writing	16 bits, BIN

(3) Suitable soft component

Operand		Word									Bit							
	System					constant	mo	dule	le system									
	D	FD	TD	CD	DX	DY	DM	DS	K/H	ID	QD	X	Y	Μ	S	Т	C	Dn.m
S1	•																	
S2	•	•							•									
S3	•								K									

Note: D represents D and HD; TD represents TD and HTD; CD represents CD, HCD, HSCD, HSD; DM stands for DM and DHM; DS stands for DS and DHS.

M represents M, HM, SM; S represents S and HS; T represents T and HT; C represents C and HC.

(4) Function and action



• Operand S2: The number of registers occupied by writing serial port parameters, usually 8 (Ethernet port parameters are 9).

- Operand S3: Serial port number range: K0~K5. K0: COM0, K1: COM1, K2: COM2, or COM2-RS232 or COM2-RS485, K3: COM3, K4: COM4, K5: COM5, K9: Ethernet port.
- Write the values from HD0 to HD7 into the parameters of serial port 2. The specific parameter names and definitions can be found in sections 4-2-4.
- After writing, the PLC needs to power on again to make the parameters take effect.

4-2-3. Set the IP address [IPSET]

(1) Instruction overview

Set the IP address of the local device.

Set IP address [IPSET]										
16-bit	IPSET	32-bit	-							
instruction		instruction								
Execution	Edge triggering	Suitable	XD3E, XD5E, XDME, XDH, XG,							
condition		model	XL5E, XL5N, XLME, XL5H, XLH							
Firmware	V3.5.3b and up	Software	V3.5.3 and up							

(2) Operand

Operand	Function	Туре
S0	Specify local register address	16-bit integer
S1	Specify the register numbers (K4, K12)	16-bit integer
S2	Specify the local serial port no. (K9)	16-bit integer

(3) Suitable soft component

Operand		Word soft component									Bit soft component							
	System					Constant	Mo	System										
	D	FD	TD	CD	DX	DY	DM	DS	K/H	D	QD	X	Y	М	S	Т	C	Dn.m
D0	•								•									
D1	•								•									
D2	•								•									

Note: D stands for D, HD; TD indicates TD and HTD. CD indicates CD, HCD, HSCD, and HSD. DM indicates DM and DHM. DS indicates DS and DHS.

M stands for M, HM, SM; S stands for S and HS; T stands for T and HT; C stands for C, HC.

(4) Function and action

Instruction format



• Write the network parameters in HD0-HD11 to the Ethernet port of PLC

Address	Function	Example	Data format			
HD0		192	Decimal			
HD1	ID	168	Decimal			
HD2	IP	51	Decimal			
HD3		103	Decimal			
HD4		255	Decimal			
HD5	Subnot most	255	Decimal			
HD6	Subhet mask	255	Decimal			
HD7		0	Decimal			
HD8		192	Decimal			
HD9	Default actoryou	168	Decimal			
HD10	Default galeway	51	Decimal			
HD11		1	Decimal			

- S0: Specifies the first address of the local register.
- S1: The value is K4 or K12.
 - K4: Write only the IP address, for example, IP address: 192.168.51.103.

K12: Write the IP address, subnet mask, default gateway to the Ethernet port of the PLC;

For example, IP address: 192.168.51.103

Subnet mask: 255.255.255.0

Default gateway: 192.168.51.1

• S2: the Ethernet port parameters of the PLC are fixed to K9.

Note:

(1) After the parameters are written, the PLC needs to be powered on again to take effect;

(2) When the current IP address is automatically obtained, executing the IPSET command will change the IP address to a fixed IP address;

(3) Set the IP to 0, you can change the fixed IP to automatically obtain IP.

Address	Function	Туре	Data format		
SD1930		Read only	Decimal		
SD1931	ID	Read only	Decimal		
SD1932	IP	Read only	Decimal		
SD1933		Read only	Decimal		
SD1934		Decimal			
SD1935	auhn at maale	Read only	Decimal		
SD1936	subnet mask	Read only	Decimal		
SD1937		Read only	Decimal		
SD1938		Read only	Decimal		
SD1939	default actorier	Read only	Decimal		
SD1940	default gateway	Read only	Decimal		
SD1941		Read only	Decimal		

• Register address of Ethernet port parameters

Note: The Ethernet parameter registers are read-only. To change the IP address, you must use the IPSET instruction.

4-2-4. Serial port parameter name and setting

Assuming that HD0~HD14 correspond to serial port parameters, the parameter names and settings represented by each register are shown in the table below:

Parameter	Parameter name and setting							
address	MODBUS	Free format	X-NET com	munication	Ethernet			
	communication	communication	OMMS	TBN	communication			
	(HD0=1)	(HD0=2)	(HD0=3)	(HD0=3)	(HD0=3)			
HD0	Network type:							
	1: MODBUS	2: free format 3	3: X-NET 4: MODBU-TCP					
HD1	MODBUS	Baud rate	Network	Network	Network number			
	station no.	Refer to table 1	number	number	IP high two			
	1~254		0~32767	0~32767	bytes			
HD2	Transmission	Frame format	Station no.	Station no.	Station no.			
	mode	Refer to table 2	0~100	0~100	IP low two bytes			
	0: RTU							
	128: ASCII							
HD3	Baud rate	Free property	Physical layer ty	ре				
	Refer to table 1	bit7:	0: PHY_RS485					
		1: Has a starting	1: PHY_SOF (U	nidirectional fiber	ring network)			
		character	2: PHY_OFPP (Fiber optic dot network)					
		0: no starting	3: PHY_RS232					
		character	4: PHY_RS422					
		bit6:	5: PHY_TTL (TTL level network)					
		1: There is a						
		terminator						
		0: No terminator						
HD4	frame format	Starting character	Link layer type					
	refer to table 2		0: TBN					
			1: HDN					
			2: CCN					
			3: PPFD					
			4: PPU					
110.5		•	5: Ethernet	D				
HD5	retry count 0~5	terminator	OMMS	Baud rate	Subnet Mask			
			Properties	Refer to table	High Iwo Bytes			
			128: Supports	1				
			periodic					
			communication					
			, otherwise not					
	Descent	English (*)	supported	4-1	Seehaart 1			
HD6	Kesponse	time 0 255	UNINS baud	token rotation	Subnet mask			
	overtime	ume 0~255		$1 \sim 60000$	low two bytes			
	0~63333		Keter to table 1	(ms)				

HD7	Delay	before	Response	e	OMMS	Slave	Maximum		Gateway address
	sending		timeout ()~65535	List		number	of	two bytes higher
	0~255		(0 is	infinite	Each bit	of	stations 1~1	00	
			waiting)		each byte	e in the			
					array				
					represen	ts			
					whether	the			
					slave sta	tion			
					can be a	ccessed			
					(valid fo	r the			
					master st	tation,			
					i.e. the st	tation			
					number i	is 1)			
HD8	-		-		-		-		Gateway address
									two bytes lower

[Note]: The table does not include "buffer bits" in free format communication mode, so "buffer bits" cannot be read and written using CFGCR and CFGCW instructions, but can be read and written using MOV instructions. The address of "buffer bits" is shown in Appendix 3.

Table 1: baud rate

Value	Baud rate	Value	Baud rate	Value	Baud rate	Value	Baud rate
1	300 bps	7	19200 bps	13	256000 bps	19	1000000 bps
2	600 bps	8	28800 bps	14	288000 bps	20	1200000 bps
3	1200 bps	9	38400 bps	15	384000 bps	21	1500000 bps
4	2400 bps	10	57600 bps	16	512000 bps	22	2400000 bps
5	4800 bps	11	115200 bps	17	576000 bps	23	3000000 bps
6	9600 bps	12	192000 bps	18	768000 bps		

Table 2: frame format

Stop bit		Parity bit			Data bit length		
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
00: 1 bit		000: none			000: 5 bits		
01: 1.5 bit		001: odd			001: 6 bits		
10: 2 bits		010: even			010: 7 bits		
		011: vacant	;		011: 8 bits		
		100: Mask			100: 9 bits		

4-2-5. Read write port parameter application

Example 1: By using the parameter read instruction [CFGCR] and write instruction [CFGCW], the network parameters of the PLC are read into 9 consecutive registers D10~D18. After modification, the network parameters of the 9 consecutive registers D10~D18 are written into the serial port settings of the PLC.

PLC program:

	- CFGCR D10 K9 K9	_
M2	- CFGCW D10 K9 K9	

PLC1-自由监	SH2			P >
监控 添	加修改册	除删除到	注部 上移	下移置顶置底
寄存器	监控值	字长	进制	注释
D10	0003	单字	16进制	-
D11	COA8	单字	16进制	IP地址前两位,CO对应K192,A8对应K168
D12	003C	单字	16进制	IP地址后两位,00对应KD,3C对应K60
D13	0000	单字	16进制	
D14	0005	单字	16进制	
D15	FFFF	单字	16进制	子网掩码前两位,分别对应255.255
D16	FFOO	单字	16进制	子网掩码后两位,分别对应255.0
D17	COA8	单字	16进制	默认网关前两位,分别对应192.168
D18	0001	单字	16进制	默认网关后两位,分别对应0.1

D11: IP address first two bits, C0 corresponds to K192, A8 corresponds to K168

D12: IP address last two bits, 00 corresponds to K0, 3C corresponds to K60

D15: subnet mask first two bits, correspond to 255.255

D16: subnet mask last two bits, correspond to 255.0

D17: default gateway first two bits, correspond to 192.168

D18: default gateway last two bits, correspond to 0.1

When M1 is set, it triggers the network parameter reading of the PLC. After modifying the network parameters, set ON M2 to write the modified network parameters into the PLC. After writing, the PLC will power off and then power on to make the serial port parameters effective.

4-3. Ethernet communication flag and register

Address	Format	Function	Explanation
SD1905	Hex	IP net number	The first two bytes of IP address
SD1906	Hex	IP station no.	The last two bytes of IP address
SD1907	Hex		The first two bytes of subnet mask
SD1908	Hex	Subnet mask	The last two bytes of subnet mask
SD1909	Hex		The first two bytes of defaulted gateway
SD1910	Hex	Defaulted gateway	The last two bytes of defaulted gateway
CD1020			Abnormal socket ID, only be effective
SD1920	Decimal	Abnormal socket ID	when the connection is not created
			1: the socket ID is over the range
			2: not registered socket ID sends a
			communication request
			3: communication type error, out of the
			range
			0TCP 1UDP
			4: TCP connection quantity out of the
	Decimal		range, max is 32
		Error code	5: UDP connection quantity out of the
SD1921			range, max is 32
			6: communication mode error, out of the
			range, 0Server 1Client
			7: Abnormal flag bit (usually abnormal
			flag bit in XDPPRO software)
			8: Target port error (check if the target port
			setting is 0)
			9: Local port error (check if the local port
			setting is 0)
			10: Communication busy
SD1930	Decimal	IP address	IP address first byte
SD1931	Decimal		IP address second byte
SD1932	Decimal		IP address third byte
SD1933	Decimal		IP address fourth byte
SD1934	Decimal	Subnet mask	Subnet mask first byte
SD1935	Decimal		Subnet mask second byte
SD1936	Decimal		Subnet mask third byte
SD1937	Decimal		Subnet mask fourth byte
SD1938	Decimal	Default gateway	Default gateway first byte
SD1939	Decimal		Default gateway second byte
SD1940	Decimal		Default gateway third byte
SD1941	Decimal		Default gateway fourth byte

Communication registers:

Communication coils:

Address	Function	Explanation
SM1900	Log in remote server successfully flag	Set on when the remote connection
		succeeded
SM1901	Ethernet initialization completed flag	MODBUS TCP Server/TCP IP/ XNET
SM1902	Connect net device flag	First network port of dual network port
		models or single network port model
		connect to swither/router/ other net devices
SM1903	Connect net device flag	Second network port of dual network port
		models connect to swither/router/ other net
		devices
SM1921	Ethernet error flag	Set on when the error in any of the SD1921
		generated

4-4. Ethernet communication error list

Error code	Explaination
0	Communication normal
1	The socket which is needed to OPEN already created connection
2	Return error when creating the socket
3	Bind appointed port failed
4	TCPServerAccept failed
5	TCPClientConnect failed
6	When calling Send, Recv, Clos, the specified socket hasn't created
	connection
7	Call Send return failed
8	Call Recv return failed
10	The specified sending data length is out of the range
11	The specified receiving data length is out of the range
20	When UDP communicating, received data is not from specified IP
21	When UDP communicating, received data is not from specified port
30	Actual received data length is larger than specified length
31	Actual received data length is less than specified length
32	Received data length error (non specified length)
33	Sending data length error
40	Receive timeout
50	Specified target port error, MODBUS TCP is not port 502,
	The using port is out of range (1~60000)
51	Port reuse (indicating that the port is used for both TCP and Modbus TCP)
60	Socket communication busy
61	No receiving task when receiving data (usually when the PLC receives
	data without calling S_RCV)
62	Parameter setting error

63	Remote shutdown
64	Socket type error
65	Insufficient memory for task request (task request too fast)
66	Incorrect use of IP address
67	Port usage error
68	Sending blocking error
70	Socket index error
71	Socket connection status error
100	Receive error
101	Receive timeout
182	Station no. error
183	Send buffer overflow
400	Function code error
401	Address error
402	Length error
403	Data error
404	Slave station busy
405	Memory error (Erase Flash)

5. EtherNet/IP communication

EtherNet/IP using requirements				
Suitable model	XDH, XLH, XG2			
Firmware	V3.7.4 and higher	Software	V3.7.17a and up	

5-1. Ethernet/IP overview

EtherNet/IP (Ethernet/Industrial Protocol) uses standard IEEE 802.3 technology and is an industrial automation communication protocol based on Ethernet. EtherNet/IP uses standard Ethernet and TCP/IP technology to transmit CIP communication packets.

The EtherNet/IP protocol mainly has the following characteristics: based on Ethernet technology, supporting TCP/IP and UDP/IP protocols, providing explicit and implicit messages, supporting multiple data formats, and supporting device description files.

Based on Ethernet technology

The EtherNet/IP protocol uses Ethernet technology for communication, which has the advantages of high speed, wide area, and low cost of Ethernet, and can achieve real-time control and data communication in the field of industrial automation.

Support TCP/IP and UDP/IP protocols

The EtherNet/IP protocol supports TCP/IP and UDP/IP protocols, and different protocols can be selected for communication according to the needs of the application. The TCP/IP protocol ensures the reliability and integrity of data transmission, and is suitable for control and communication situations that require high reliability; The UDP/IP protocol is suitable for broadcasting and multicast communication scenarios, with the advantages of low latency and high efficiency.

Provide explicit and implicit messages

The EtherNet/IP protocol provides two communication methods: explicit and implicit messaging. Explicit messaging uses TCP/IP protocol for communication, with flexible message formats and expandable functionality; Implicit messaging uses UDP/IP protocol for communication, suitable for broadcasting and multicast scenarios, with the advantages of simple message format and low latency.

Supports multiple data formats

The EtherNet/IP protocol supports multiple data formats, including bits, bytes, integers, floating-point numbers, etc., and can meet the data transmission needs of different application scenarios.

Supporting device description files

The EtherNet/IP protocol supports device description files, which can identify and configure devices connected to the network, improving device interoperability and manageability.

In summary, EtherNet/IP protocol is a modern industrial automation communication protocol with flexible, efficient, and reliable characteristics, widely used in industrial control, intelligent manufacturing, robotics and other fields.

5-2. Ethernet/IP nouns overview

Abbreviation	Explanation
IEEE 802.3	A standard specification in the field of communication technology, also known as Ethernet
	protocol; This standard specification defines the transmission method and format of data in
	Ethernet networks
EIP	Ethernet/IP, Industrial Ethernet
CIP	Common Industrial Protocol. Used to describe various industrial automation protocols
EipScanner	EIP master station, referred to as scanner in Etehrnet/IP
EipAdapter	EIP slave station, referred to as adapter in Ethernet/IP
EDS	Electronic Data Sheets, used to describe Ethernet/IP device
RPI	Request/Response Interval, also known as communication cycle
PPS	Packet Per Second, the number of data packets transmitted per second

5-3. Ethernet/IP communication specification

5-3-1. Implicit function specification

Scanner (Main Station)	Communication specification parameters
Suitable model	XDH, XLH, XG2
Slave station connection numbers	≤128
Number of shared connections	Adapter+Scanner≤256 pieces
Ethernet Maximum communication volume	4000pps
Data length	1~724 words (Note: 1 word=2 bytes)
RPI	1ms~65535ms
Adapter (Slave Station)	Communication specification parameters
Suitable model	XDH, XLH, XG2
Label name	≤64 bytes
Instance ID	100~199
Mapping first address	Support D/HD registers
Data length	1~724 word (Note: 1 word=2 bytes)
Number of shared connections	Adapter+Scanner ≤256 pieces
Allow configuration items	Input $(O>T) + Output (T>O) = 256 pieces$
RPI	1ms~65535ms

5-3-2. Explicit function specification

Client parameter	Communication specification parameters
Suitable model	XDH, XLH, XG2
Name	≤64 bytes
Allow configuration items	32 slave stations share 3000 instructions
Maximum byte length of communication	504 bytes (CIP packet head+CIP packet data)
data packet	
Timeout time	10~65535ms
Number of retransmissions	1~15
Enable control	Only support M0~M199999, HM0~HM19999
Connection flag bit	Only support M0~M199999, HM0~HM19999

Server parameter	Communication specification parameters				
Suitable model	XDH, XLH, XG2				
Number of connected clients	≤16				
Number of configurable labels	≤5000				
Label name	≤64 bytes				
Maximum byte length of communication	504 bytes (CIP packet head+CIP packet data)				
data packet					

5-3-3. Client and server support variable types

Client variable type	Server variable types	Data length
-	BIT	Bit (8-bit)
BOOL	BOOL	Bool (8-bit)
SINT	SINT	Short integer (8-bit)
USINT	USINT	Unsigned short integer (8-bit)
INT	INT	Integer (16-bit)
UINT	UINT	Unsigned double integer (16-bit)
DINT	DINT	Double integer (32-bit)
UDINT	UDINT	Unsigned long integer (32-bit)
LINT	LINT	Long integer (64-bit)
ULINT	ULINT	Unsigned long integer (64-bit)
REAL	REAL	Real (32-bit)
LREAL	LREAL	Long real (64-bit)
BYTE	BYTE	A bit string with a length of 8 (8-bit)
WORD	WORD	A bit string with a length of 16 (16-bit)
DWORD	DWORD	A bit string with a length of 32 (32-bit)
LWORD	LWORD	A bit string with a length of 64 (64-bit)

5-4. Ethernet/IP explicit/implicit communication

In the Ethernet/IP protocol, there are two different data transfer methods between devices or between devices and multiple devices, namely implicit and explicit functions. Their functions and usage methods are not exactly the same. Below, we will provide corresponding functional introductions for these two communication methods

5-4-1. Implicit function

The implicit function in the Ethernet/IP protocol refers to the method of data transmission through I/O data tables, which is usually used in real-time control and monitoring applications. In implicit functionality, two important components need to be used: a scanner and an adapter.

5-4-1-1. Adapter configuration

Before conducting network data exchange, the device needs to configure the address and length of the implicit message to be transmitted in the corresponding adapter configuration interface. The label setting and signature must be configured, and the instance ID is an optional configuration. The operation method is as follows:

Free Monitor Master Config	Adapter->S	canner (T->0)				Scanner->A	dapter(0->T)			
Data Monitor EtherNet/IP Adapter		an 2000 200 viet is 10 viet.					~39778538414326948			
Set Reg Init Value	Number	Tag name	Living	Map first	Enter data	Munber	Tag name	Living	Map first	Enter dat
Function Version Switch		-	example	address	length		-	example	address	length
PLC Config	0	tag_l		ро	1	0	tag_2	-	DO	1
NO VO										
Password						11				
PLC Serial Port						11				
ethernet						11				
Pulse						11				
Module						11				
BD						11				
FD						11				
4GB0X										
4GBOX WBOX										
4GBOX WBOX SystemConfig				bbd	Delete				bba	Delete
4GB0X 4GB0X WB0X SystemConfig C Communication				Add	Delete				Add	Delete
GBOX GOX WBOX SystemConfig Communication Gthereto				Add	Delete				Add	Delete
43BDX 44BDX 3 VB0X 5 SystemConfig 1 Communication 3 Ethernetip W EpScanner				Add	Delete				Add	Delete
dGBOX d	Label set	ing		Add	Delete	Label set	ting		Add	Delete
AGBOX WBOX SystemConfig C.Communication Communication Ethernetp Fig.Scnner Fi	Label sett	ing L name	tar 1	Add	Delete	Label set	ting 1 name	tag 2	Add	Delete
AGBOX WBOX SystemConfig C Communication Ethernetip EtheScanner EtheScanner EtheScanner EtheScanner EtheScanner EtheScanner EtheScanner EtheScanner EtheScanner	Label sett	ing L name	tag_1	Add	Delete	Label set Labe	ting 1 name	tag_2	bba	Delete
dGBOX dGBOX dGBOX dGBOX SystemConfig Communication theretp EpScience the Configuration theretp the configuration configuration	Label sett Labe Livi	ing 1 name ng example ID	tsg_1	Add	Delete	Label set Labe Livi	ting 1 name ng example ID	tag_2	bba	Delete
AGBOX AGBOX AGBOX AGBOX SystemConfig Communication Expending Explore ForAdget ModusTcp Explore Expront Expront Expront Expront Ava configuration	Label sett Labe Livir	ing 1 name ng example ID	tag_1	Add	Delete	Label set Labe Livi	ting 1 name ng example ID	tag_2	Add	Delete
AGBOX AGBOX AGBOX AGBOX AGBOX AGBOX AGBOX AGGOX A	Label sett	ing 1 name ng example ID fivet adducer	tag_1	Add	Delete	Label set Labe Livi	ting 1 name ng example ID fivet address	tag_2	Add	Delete
AGBOX WBOX GBOX	Label set Labe Livin Map	ing 1 name ng example ID first address	tag_1	Add	Delete 	Label set Labe Livi Map	ting 1 name ng example ID first address	tag_2	Add	Delete
ASBOX ASBOX ASBOX ASBOX ASBOX ASBOX ASBOX ASBOX ASSOCIATION Communication Cthemetip Cthemetip Cthemetic Ct	Label sett Labe Livin Map	ring 1 name ng example ID first address	tag_1	Add	Delete	Label set Labe Livi Map	ting 1 name ng example ID first address	tag_2	Add	Delete

(1) Double click to enter Ethernet/IP adapter configuration interface.

(2) Adapter—>Scanner(T->O) data configuration area:

The data configured in this configuration area is in the T ->O direction, and the data transmission direction is from the adapter to the scanner.

Add	Add one piece of Adapter—>Scanner(T->O) data configuration.
Delete	Delete the selected Adapter—>Scanner(T->O) data configuration.
Label name	As the target end, use the corresponding configured label name as the connection and respond to the
	connection establishment request. The label name can support a maximum of 64 bytes
Instance	As the target end, respond to the connection request by using the corresponding configured instance
ID^{*1}	ID as the connection path. The optional range of this instance ID is 100-199;
Map first	The starting address for data transmission in this connection;
address*2	
Data	The number of registers connected for data transmission.
length*3	

Note:

% 1: The label name must be set, and the instance ID is an optional configuration;

* 2: The starting address currently supports two register types, D and HD;

 \approx 3: When establishing a connection with the corresponding label or instance ID, it is important to note that the data length configured by the adapter should be consistent with the data length configured by the scanner to avoid communication anomalies when establishing the corresponding connection.

(3) Scanner—> Adapter(O->T) data configuration area.

The data configured in this configuration area is in the O \rightarrow T direction, and the data transmission direction is from the scanner to the adapter for data transmission.

Add	Add one piece of Scanner—> Adapter(O->T) data configuration
Delete	Delete the selected Scanner—> Adapter(O->T) data configuration
Label name	As the target end, use the corresponding configured label name as the connection and respond to the
	connection establishment request. The label name can support a maximum of 64 bytes
Instance	As the target end, respond to the connection request by using the corresponding configured instance
ID^{*1}	ID as the connection path. The optional range of this instance ID is 100-199;
Map first	The starting address for data transmission in this connection
address*2	

Data	The number of registers connected for data transmission
length*3	
Import	Import the configured information into the current configuration interface in the form of an XML
	file
Export	Export the configured information in the form of an XML file
Upload	Upload the configuration information downloaded to the PLC to the current configuration interface,
	and the uploaded configuration information will overwrite the existing configuration information
	on the current interface
Download	Download the configuration information of the current configuration interface to the PLC. The
	downloaded configuration information will overwrite the original configuration information in the
	PLC and take effect in real time with the new configuration information
Ok	Click OK to save the configuration information for the current page
Cancel	Click to cancel the configuration information for the current page and discard it

Note:

% 1: The label name must be configured, and the instance ID is an optional configuration;

* 2: The starting address currently supports two register types, D and HD;

 \times 3: When establishing a connection with the corresponding tag or instance ID, it is important to note that the data length configured by the adapter should be consistent with the data length configured by the scanner to avoid communication anomalies when establishing the corresponding connection.

5-4-1-2. Scanner configuration

PLC Config VO Password PLC Serial Port ethernet EtherNet/IP Scanner Config × Master Config Library × Add Device Puise Install EDS File Uninstall EDS File Module BD ED M ED WBOX Title Vendor Slave Config -XINJE ElectricCo., Ltd -XINJE EtherNet/IP XINJE ElectricCo., Ltd SystemConfig PLC Communicatio Ethernetip EipExplicit IbusTcp EthercatMaster Motion control(H movement) - Axis configuration < Vendor Axis group configuration Device Name Device ID CPU Detail PLC Project Message Expansion Details BD Details Version Description ED Details 6 Scan Cycle Add Close Re Error Details Slave Number: 0 Connection Number: 0/256 Theory throughput: 0 PPS Actual throughput 0 PPS Import Export Upload DownLoad Ok Cancel ord

1. Load EDS file, add slave device.

(1) Double click on EipScanner to enter the EtherNet/IP Scanner parameter configuration interface;

(2) Right click on EtherNet/IP Scanner to add devices;

(3) Load or unload third-party EDS files in the device library to prepare for the next communication configuration step;

(4) Double click or click to add the EDS file to be communicated, and add the specified slave to the slave configuration information bar.

2. Configure General Settings for Slave Stations

EtherNet/IP Scanner Config		×
Renter Config	Routine Connection IOMapping Connection Status	
master Config EtherNet/IP-Scanner	Address config IP Address: 192 168 6 1 3	
Slave Config -StationIdO:XINTE EtherNet/IP	Compatible check	
	Vendor ID: 1723	
	Device Type: 12	
	Product Code: 14	
	Major Revisions: 1	
	Minor Revisions: 1	
		6
9		<u> </u>

	1	Double click on the node corresponding to the slave station configuration bar to configure						
		relevant communication information						
	2	Click on Routine to configure the IP address and compatibility check accordingly						
	3	The IP address is the IP address of the slave device under the corresponding node						
	4	If compatibility check is conditional, check the checked conditions to determine whe						
		EDS file matches the slave device. If compatibility check is checked, all relevant information						
		will be matched with the information in the EDS file by default. It should be noted that						
		compatibility checks are conducted during the connection period. If the check fails, the						
		connection cannot be made						
	5	The slave number is used to count the number of slave stations connected under the current						
		master station, and the connection number is used to count the number of connections						
		established between the master station and all connected slave stations. The specific						
		specifications for the slave number or supported connections supported by Ethernet/IP						
		communication can be found in 5-3. Ethernet/IP communication specifications						
	6	Theoretical throughput *1 is used to display the network throughput of the current connection,						
		while actual throughput is used to display the throughput of the entire Ethernet network of the						
		current device;						
	Import	Import the configured information into the current configuration interface in the form of an						
		XML file						
	Export	Export configured information in the form of an XML file						
	Upload	Upload the configuration information downloaded to the PLC to the current configuration						
		interface, and the uploaded configuration information will overwrite the existing configuration						
7		information on the current interface						
	Download	Download the configuration information of the current configuration interface to the PLC. The						
		downloaded configuration information will overwrite the original configuration information in						
		the PLC and take effect in real time with the new configuration information						
	OK	Click OK to save the configuration information for the current page						
	Cancel	Click to discard the configuration information for the current page.						

× 1: pps is the unit of network throughput, also known as Packet Per Second, represents the total number of packet data packets that can be sent and received within 1 second.

■ When used as an InputOniy connection, the calculate formula for each connection:

When RPI<100ms, the theoretical throughput pps=1000ms/RPI+10;

When RPI>100ms, the theoretical throughput pps=1000ms/RPI * 2.

Example: Two PLCs establish implicit communication, and two InputOniy type connections are established in the Scanner connection configuration interface. One connection has an RPI communication cycle of 110ms, and the other connection has an RPI communication cycle of 10ms. So the total theoretical throughput pps=1000/110*2+(1000/10+10)=128pps.

pps=1000/110/2 + (1000/10+10)=120pps.

■ When used as an ExclusiverOwner connection, the calculate formula for each connection:

RPI_1: Communication cycle from adapter input (T ->O) direction;

RPI_2: Communication cycle in the direction of output to adapter (O ->T);

Theoretical throughput pps=1000ms/RPI_1+1000ms/RPI_2.

Example: Two PLCs establish implicit communication and establish an ExclusiverOwner connection on the Scanner connection configuration interface. The communication period from the input (T ->O) direction of the adapter RPI_1 is 100ms, and the communication period from the output to the adapter (O ->T) direction RPI_2 is 10ms. The total theoretical throughput pps is 1000/100+1000/10=110 pps.

therNet/IP Scanner Config									
Master Config	Routine	Connection IC	OMapping Conn	ection Statu	15				
EtherNet/IP Scanner	No	Connection	Input Connection Point	DataSize	IN Addres	OUT connectic Point	n DataSize	OVT Address	Connectio ID
	0	InputOnly(ID	. IN_100	1	DO				0
lave Config -StationIdO:XINJE EtherNet/IP					0				
	Connec	tion Name Inp e out(T <mark>4 RP</mark> I	utOnly(ID Type) *16 ~	(IN: 1600ms	3 0VT:1600ms)	Configure Instan	ce 1	2 Add	Delect
	IN(Inpu	it from the adapte	er)			OUT (Output to th	e adapter)		
	Conne	ction Type Poi	nt to point		~	Connection Typ	e Point to po	nt	\sim
	Connec	ction Point IN_	_100		~	Connection Poir	oUT_254		\sim
	Da	ta Size 1		(1-724Wor	d)	Data Size		(1-1Word	D
	Мар	Maddres DO			5	MapMaddres			
	Send	l trigger Cvc	:le		~	RPI (communicati	on 100	(1-65535	ims)
	RPI(cor	mmunication 100 cycle)	l	(1-65535m	s)	cycrey	Keep cons	istent	
lave Number: 1 Connection Number:	1/256 Theory t	throughput: 0 PPS	Actual throughput	t O PPS	Impor	t Export	Upload D	ownLoad Ok	Cance
Connection	The com	nection dis	play box	can show	v connec	tion types a	nd corresp	onding confi	guration
display area	informat	tion	1 5			51	1	0	0
Connection ID	Assign a	unique co	nnection	ID to the	added c	onnection, v	which will	not change v	vith the
addition or removal of the connection									
Add	Clicking	, on add wi	ill create a	new cor	nnection				
Delete	Select th	e correspo	nding esta	ablished	connecti	on. click de	lete to dele	te the select	ed
	connecti	on	8-34			,			
	201110001	~							

3. Add the connection

name

transmission is in the T ->O direction.

data from the adapter. The adapter can only send data to the scanner, that is, data



	 Multicast: data exchange between multiple devices in the network. In this way, multiple scanners can simultaneously obtain data in the T ->O direction sent from the same adapter, and only one corresponding data frame needs to be sent during data exchange, which can save adapter network resources to a certain extent. (Note: The implementation arrow represents the data frames that need to be sent, and the dashed arrow represents the fewer data frames sent compared to point-to-point transmission in multicast.) Scanner Scanner Scanner ListenOnly multicast Adapter
	When using ListenOnly (ID Type) multicast, it must be attached to an InputOnly (ID Type) or ExclusiveOwner (ID Type) connection, and the corresponding connection type attached to InputOnly or ExclusiveOwner must also be multicast. The configured data size and RPI communication cycle must be consistent with the attached connection type, otherwise the
	establishment will not be successful.
Connection point ^{*1}	The tag name or instance ID required to establish communication.
Data size	The number of registers connected for data transmission
Map first address *2	The starting address for data transmission in this connection
Send trigger	 Loop: Trigger the scanner periodically based on the set RPI; State change: When the status of the adapter changes, the scanner is triggered. If the status of the adapter changes periodically and is less than 1/4 of the RPI, the scanner is triggered periodically at 1/4 of the RPI; Application trigger: Trigger rules are consistent with state changes.
RPI	Used to set the communication time for the corresponding connection cycle, with a default of 100ms and a setting range of 1-65535ms. RPI (communication cycle) can be set according to the priority of data transmission and reception, so as to adjust the overall communication volume for data transmission and reception.
	Connection point ^{*1} Data size Map first address ^{*2} Send trigger RPI

Note:

 \times 1: When establishing a connection with the corresponding tag or instance ID, it is important to note that the data length configured by the adapter should be consistent with the data length configured by the scanner to avoid communication abnormalities when establishing the corresponding connection;

% 2: The starting address currently supports two register types, D and HD.

4. IO mapping

The IO mapping interface can display mapping addresses for different connection configurations and view the status of data in real-time.

F	Routine Connection	1 IOMapping	Connection Status		
Master Config EtherNet/IP Scanner	Tag ⊟-InputOnly(ID Type) ⊟-IN_100	Channel	MapAdress	Value	
5 M N N N N N N N N N N N N N N N N N N	É-IN_100[0]	InPut	DO	0	
Slave Config		bit0	DO. O	OFF	
-StationIdO:XINJE EtherNet/IP	-	bit1	DO. 1	OFF	
		bit2	DO. 2	OFF	
	-	bit3	DO. 3	OFF	
		bit4	DO.4	OFF	
		bit5	DO. 5	OFF	
	-	bit6	DO. 6	OFF	
	-	bit7	DO. 7	OFF	
		bit8	DO. 8	OFF	
	-	bit9	DO. 9	OFF	
		bit10	DO. 10	OFF	
		bit11	DO. 11	OFF	
	-	bit12	DO. 12	OFF	
		bit13	DO. 13	OFF	
	-	bit14	DO. 14	OFF	
		bit15	DO. 15	OFF	
					2.6

5. Connection status

You can view the status information of each connection in real-time, where the connection ID on the "Connection" configuration interface is consistent with that on the "Connection Status" configuration interface.

Master Config EtherNet/IP Scanner		Routine Connect Connection B	ion IOMappin Name Inpu	g Connection	0 v				
Slave Config —StationIdO:XINJE EtherNet/IF		Connection Conection statu Configure statu General status Extend status Status descri	ID						
	No	Connection	Input Connection Point	DataSize	IN Address	OUT Connection Point	DataSize	OVT Address	Connection ID
	0	InputOnly(ID	IN_100	1	DO		-		0

Connection	Select the various connections that have been added to the current slave station's
name	"connections".
Connection ID	The connection ID corresponding to the connection.
Connection	Display the current connection status in hexadecimal.

status code	
Configure status	Display the current configuration status in hexadecimal.
code	
General status	Display the current general status in hexadecimal.
code	
Extend status	Display the current extended status code in hexadecimal.
code	
Status	Used to display the status information of the current connection.
description	

Note: For detailed extension status codes, please refer to Appendix Ethernet/IP communication extension codes.

6. Rules for judging descriptive information

The combination of information that is not in the following three states is prompted in the "state description": Undefined Error!

- When the "Connection Status Code"=3, "General Status Code"=0, and "Extended Status Code"=0, the configuration status code does not need to be determined, and the status description prompts "Connection successful, communication normal";
- When "Connection Status Code"=1 and "Configure Status Code"=4, the status description prompts "Unable to find IP or the IP does not support EIP";
- When the "Connection Status Code"=1 and the "General Status Code"=1, there is no need to determine the configure status code. The specific information in the status description is prompted based on the "Extended Status Code".

7. Viewing Connection Status of Structural System Variables

In the ladder diagram, the system structure variables can be directly called to view the current connection status. When calling the ladder diagram or freely monitoring the connection status, the corresponding structure number is the "connection ID" of the corresponding connection.

Example: View the communication status with connection ID 0 through a ladder diagram to determine if the corresponding connection status code is 3. If the connection status code is 3, it indicates successful communication.

	SM12		0							
0						 	 	 DMOV	EIPScanner[0].ConnectedStatus	Connected1 3
		DMOV EIPScanner[0].								
1		ConfigationStatus	DINT	配置状态码						
		ConnectedStatus	DINT	连接状态码						
		ConnectionID	DINT	注接ID 扩展出太风						
2		GeneralStatus	DINT	通用状态码						
	1		1.000		_					



5-4-1-3. Implicit communication application

Application 1: Implicit communication by using two XDH-60T-E PLCs with InputOnly connection type.

Use PLC1: XDH-60T-E (IP 192.168.6.6) as the adapter and PLC2: XDH-60T-E (IP 192.168.6.7) as the scanner to achieve implicit communication between two PLCs. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with that of the adapter.



Step 1: Create and add a communication type with test_1 as the label in the adapter, mapping the first address to D0, and inputting a data length of 5. Use the communication type with test_2 as the label for instance ID100 (check the use of instance ID), the mapping first address is D10 and the data length is 5.

Master Config EtherNet/IP Adapter	Adapter->S	Scanner (T->O)				Scanner->Adapter (0->T)				
	Number	Tag name	Living example	Map first address	Enter data length	Mumber	Tag name	Living example	Map first address	Enter data length
	0	tag_1		DO	1					
	1	tag_2	100	D10	1					
		tting		Add	Delete	Label set	ting		Add	Delete
	Label Set	al name	Lance o			Labe	Inome			
	Label Set	el name	tag_2			Labe	l name			
	Label Set	el name ing example ID	tag_2	V u	se (100-199)	Labe	l name ng example ID		u	se (100-199)
	Label Set Lab	el name ing example ID first address	tag_2 100 D10	V u	se (100-199)	Labe Livi Map	l neme ng example ID first address		u	se (100-199)

Step 2: Add slave devices to the scanner and perform relevant configuration operations on the adapter's IP address and compatibility check:

EtherNet/IP Scanner Config		×
Master Config EtherNet/IP Scanner	Routine Connection IOMapping Connection Status Address config IP Address: 192 168 6 6	
Slave Config —StationIdO:XINJE EtherNet/IP	Compatible check	
	Vendor ID: 1723 🔽 Device Type: 12	
	Product Code: 14	
	Major Revisions:	
Slave Number: 1 Connection Number:	1/256 Theory throughput: 0 PPS Actual throughput: 0 PPS Import Export Upload DownLoad 0k Cance	1

Step 3: Add two types of connections, InputonIy (Tag Type) and InputonIy (ID Type), and establish the first connection of InputonIy (ID Type). Use the Adapter ->Scanner input connection point as IN_100 and the connection type as point-to-point to receive data into five registers with D0 as the starting address. Establish the second connection of InputonIy (Tag Type), and use the Adapter ->Scanner input connection point as test_1, The data with point-to-point connection type is received in 5 registers starting from D10.

94 C	Routine Connectio	n IOMapping Con	mection Stat	15				
Master Config EtherNet/IP Scanner	No Connection	Input Connection Point	DataSize	IN Addres	OUT ss Connection Point	DataSize	OVT Address	Connection ID
	0 InputOnly(ID IN_100	5	DO		1	-	0
Slave Config	1 InputOnly(Tag test_1	5	D10	<u></u>	-	-	1
	Connection Name	InputOnly(Tag Typ	e)	~			bbA	Delect
	Time out(T)	RPI*16	(IN:1600ms)	OVT:1600ms)	Configure Instance		~	
	Time out(T) IN(Input from the	RPI*16 ·	✓ (IN:1600ms)	OVT:1600ms)	Configure Instance 	adapter)	~	
	Time out(T) -IN(Input from the s Connection Type	RFI*16	(IN:1600ms)	00T:1600ms)	Configure Instance OVT (Output to the Connection Type	adapter) Point to po	~	~
	Time out(T) IN(Input from the s Connection Type Connection Point	RPI*16 adapter) Point to point test_1	<pre>(IN:1600ms.)</pre>	OUT:1600ms)	Configure Instance OUT (Output to the Connection Type Connection Point	adapter) Foint to po	v	×
	Time out(T) IN(Input from the Connection Type Connection Point Data Size	RFI*16	(IN:1600ms) (1-724Wor)	OUT:1600ms)	Configure Instance -OUT (Output to the Connection Type Connection Point Data Size	adapter) Point to po	✓	a)
	Time out(T) IN(Input from the s Connection Type Connection Point Data Size MagMaddres	RPI*16 v adapter) Point to point test_1 5 D10	(IN:1600ms) (IN:1600ms) (1-724Wor)	OUT : 1600m s)	Configure Instance -OUT(Output to the Connection Type Connection Point Data Size MapMaddres	adapter) Point to po	~	a)
	Time out(T) IN(Input from the Connection Type Connection Foint Data Size MapMaddres Send trigger	RPI*16 v adapter) Point to point test_1 5 D10 Cycle	(IN: 1600ms) (1-724Wox)	0UT : 1600ms)	Configure Instance -OUT(Output to the Connection Type Connection Point Data Size MagMaddres RPI(communication cycle)	adapter) Foint to point	<pre>v int (1-1Wor (1-6553))</pre>	

Step 4, enter the connection status to check the connection status of IN_100 and test_1. When the connection status shows successful connection and communication is normal, it indicates that the communication has been successfully established. You can also check whether the data is correct through the corresponding mapped register.

Application 2: Implicit communication by using two XDH-60T-E PLCs with ExclusiveOwner connection type

Use PLC1: XDH-60T-E (IP 192.168.6.6) as the adapter and PLC2: XDH-60T-E (IP 192.168.6.7) as the scanner to achieve implicit communication between two PLCs. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with that of the adapter.



Step 1:

---Create two connections in Adapter ->Scanner (T ->O) direction.

Connect1: communication type with the label as test_3 on the adapter. Map the first address as D0 and input data length as 10.

Connection 2: communication type with instance ID100, label name test_5 on the adapter. Map the first address as D20 and input data length as 10.

---Create two connections in Scanner->Adapter(O->T) direction.

Connection 1: communication type with the label as test_4 on the adapter. Map the first address as D100 and input data length as 10.

Connection 2: communication type with the instance ID101, label name test_6 on the adapter. Map the first address as D120 and input data length as 10.

EtherNet/IP Adapter Config										X	
Master Config EtherNet/IP Adapter	Adapter->S	Scanner (T->0)				Scanner->/	Scanner->Adapter(0->T)				
	Number	Tag name	Living example	Map first address	Enter data length	Mumber	Tag name	Living example	Map first address	Enter data length	
	0	test_3	100	DO	10	0	test_4	1000	D100	10	
	1	test_5	100	D20	10		test_6	770	D120	10	
	Label set	ting		Add	Delete	Label set	ting		Add	Delete	
	Labe	el name	test_5			Labe	el name	test_6			
	Livi	ing example ID	100		15e (100-199)	Livi	ing example ID		u	se (100–199)	
	Мар	first address	D20			Мар	first address	D120			
	Date	a length	10	(1-)	'24word)	Date	a length	10	(1-7	24word)	
]				Import	Outport	Upload	Downlo	ad Ok	Cancel	

Step 2: Add slave devices to the scanner and configure the IP address and compatibility check of the adapter accordingly.

EtherNet/IP Scanner Config		~
Master Config EtherNet/IP Scanner	Routine Connection IOMapping Connection Status	
Slave Config —StationIdO:XINJE EtherNet/IP	Compatible check	
	Vendor ID: 1723	
	Product Code: 14	
	Major Revisions: 1	
		16
Slave Number: 1 Connection Number:	2/256 Theory throughput: 0 PPS Actual throughput: 0 PPS Import Export Upload DownLoad 0k	Cancel

Step 3: Add two types of connections on the scanner: ExclusiveOwner (Tag Type) and ExclusiveOwner (ID Type); Establish the first connection for ExclusiveOwner (ID Type), and receive data from Adapter ->Scanner with input connection point IN_100 and connection type point-to-point into 10 registers starting from D20. Send out 10 data from Scanner ->Adapter with output connection point OUT_101 and connection type point-to-point, D30 as the starting address. Establish a second connection for ExclusiveOwner (Tag Type), and receive data from Adapter ->Scanner with input connection point test_3 and connection type point-to-point into 10 registers with D40 as the starting address. Send out 10 data from Scanner ->Adapter with output connection type point-to-point, connection point test 4 and D50 as the starting address.

therNet/IP Scanner Config	P (' Connecto	ion TOTE 1 C						
Master Config	Koutine Connect.	ion iomapping (onnection stat	15				
EtherNet/IP Scanner	No Connecti	on Input Point	on DataSize	IN Address	OUT Connection Point	DataSize	OUT Address	Connection ID
	0 Exclusive	eOwne IN_100	10	D20	OUT_101	10	D30	0
lave Config	1 Exclusiv	eOwne test_3	10	D40	test_4	10	050	1
								n.lt
	Connection Name	ExclusiveOwner(Tag Type)	×			Add	Delect
	Connection Name Time out(T)	ExclusiveOwner(Tag Type)	~ OVT:1600ms) 0	Configure Instance		bba ~	Delect
	Connection Name Time out(T) IN(Input from the	ExclusiveOwner(RFI*16 e adapter)	Tag Type)	✓ OVT:1600ms) (Configure Instance OUT(Output to the s	adapter)	bba ~	Delect
	Connection Name Time out(T) IN(Input from the Connection Type	ExclusiveOwner(RFI*16 adapter) Point to point	(Tag Type)	V OUT:1600ms) 0 V 1	Configure Instance OUT(Output to the s Connection Type	adapter) Point to po	Add ~	Delect
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point	ExclusiveOwner(RFI*16 e adapter) Point to point t test_3	(Tag Type)	 OUT:1600ms) (Configure Instance OUT(Output to the e Connection Type Connection Foint	adapter) Point to po test_4	Add ~	
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size	ExclusiveOwner(RFI*16 e adapter) Point to point t test_3 10	(Ing Type)	> 00T: 1600ms) 00 → 1<	Configure Instance OUT(Output to the e Connection Type Connection Point Data Size	adapter) Point to po test_4 10	Add ~ int (1-724W	Jelect
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size MapMaddres	ExclusiveOwner(RFI*16 e adapter) Point to point t test_3 10 D40	(Ing Type)	OUT: 1600ms) (4)	Configure Instance OUT(Output to the e Connection Type Connection Point Data Size MapMaddres	adapter) Point to po test_4 10 D50	Add ~ ~	Delect v ord)
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size MapMaddres Send trigger	ExclusiveOwner(RFI*16 e adapter) Point to point t test_3 10 D40 Cycle	(Iag Type)	OUT:1600ms) () d)	Configure Instance OUT(Output to the e Connection Type Connection Point Data Size MapMaddres RPI(communication ogyle)	Adapter) Point to po test_4 10 100	Add v int (1-724W (1-6653	Delect

Step 4: Enter the connection status to check the connection status of IN_101 and test_3. When the connection status shows successful connection and communication is normal, it indicates that the communication has been successfully established. The corresponding mapped register can also be used to check if the data is correct.

Application 3: Implicit communication by using three XDH-60T-E PLCs with ListenOnly connection type.

Using PLC1: XDH-60T-E (IP 192.168.6.6) as the adapter, PLC2: XDH-60T-E (IP 192.168.6.7) as the scanner, and PLC2: XDH-60T-E (IP 192.168.6.20) as the scanner, to achieve multicast transmission of the 60 register data of adapter D0-D59 to the other two scanners' HD0-HD59 registers. During the connection creation process, attention should be paid to the connection type used, connection point, data size set, and RPI (communication cycle). The configuration needs to be consistent with the adapter.



Step 1: Create a communication type with an instance ID100 labeled as test_aa in Adapter ->Scanner (T ->O) direction on the adapter, mapping the initial address to D0, and entering a data length of 60 (check the use of instance ID).

laster Config therNet/IP Adapter	Adapter->So	canner (T->0)				Scanner->Adapter(0->T)					
	Number	Tag name	Living example	Map first address	Enter data lensth	Mumber	Tag name	Living example	Map first address	Enter data length	
	0	test_aa	100	DO	60						
				Ya					()a		
				Add	Delete				Add	Delete	
		23				274 2	13				
	Label sett	ing				Label set	ting				
	Label	name	test_aa			Labe	l name				
	Livin	ng example ID	100		se (100-199)	Livi	ng example ID			se (100-199)	
	Map f	first address	DO			Map	first address				
		1 1	60		or 1)				4.7	N 1)	
	Data	length	00	(1-7	24word)	Date	i length		(1-0	24word)	

Step 2: Add slave devices to the scanner of PLC2: XDH-60T-E (IP 192.168.6.7) and perform relevant configuration operations on the adapter's IP address and compatibility check:

EtherNet/IP Scanner Config			×
Master Config EtherNet/IP Scanner	Routine Connection IOMapping Connection Status Address config IP Address: 192 168 6 6		
Slave Config StationIdO:XINJE EtherNet/IP	Compatible check		
	Vendor ID: 12		
	Major Revisions:		
Slave Number: 1 Connection Number:	2/256 Theory throughput: 0 PPS Actual throughput: 0 PPS Import Export Upload DownLoad	Ok	Cancel

Step 3: Add an InputOnly (ID) connection type on the scanner of PLC2: XDH-60T-E (IP 192.168.6.7), with multicast as the connection type, IN_100 as the connection point, 60 words in data size, HD0 as the mapping address, and 100ms as the RPI (communication cycle).

Master Config	Koutine Connecti	ion IOMapping Com	nection Statu	15				
EtherNet/IP Scanner	No Connecti	on Input Connection Point	DataSize	IN Address	OUT Connection Point	DataSize	OUT Address	Connection ID
and an	0 Input0nly	/(ID IN_100		100				
Slave Config —StationIdO:XINJE EtherNet/IP								
							Add	Delect
	Connection Name	InputOnly(ID Type)		~				
	Connection Name Time out(T)	InputOnly(ID Type) RPI*16 ~	(IN:1600ms	∨ 0VT:1600ms)	Configure Instance	1	~	
	Connection Name Time out(T) IN(Input from the	InputOnly(ID Type) RFI*16 ~	(IN:1600ms	∨ OVT:1600ms)	Configure Instance OVT(Output to the s	1 dapter)	~	
	Connection Name Time out(T) -IN(Input from the Connection Type	InputOnly(ID Type) RFI*16 ~ adapter) Multicast	(IN:1600ms	~ OVT:1600ms)	Configure Instance -OVT (Output to the s Connection Type	1 idapter) Foint to po	~	
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point	InputOnly(ID Type) RFI*16 ~ adapter) Multicast : IN_100	(IN:1600ms	✓ OUT:1600ms)	Configure Instance -OUT(Output to the e Connection Type Connection Point	1 dapter) Foint to po OUT_254	v	× ×
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size	InputOnly(ID Type) RFI*16 ~ adapter) Multicast : IN_100 60	(IN: 1600ms)	 ✓ OUT:1600ms) ✓ ✓ ✓ • 	Configure Instance -OUT (Output to the s Connection Type Connection Point Data Size	1 Roint to po OUT_254	✓ vint (1-1%or	
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size MagMaddres	InputOnly(ID Type) RFI*16 ~ adapter) Multicast : IN_100 60 HD0	(IN: 1600ms)	✓ OUT:1600ms) ✓ ✓ ✓ ✓ •	Configure Instance -OUT (Output to the a Connection Type Connection Point Data Size MagMaddres	1 Mapter) Point to po OUT_254	✓	~ ~ (b
	Connection Name Time out(T) -IN(Input from the Connection Type Connection Point Data Size MapMaddres Send trigger	InputOnly(ID Type) RFI*16 ~ multioast IN_100 60 KDD Cycle	(IN:1600ms)	<pre>></pre>	Configure Instance -OUT (Output to the a Connection Type Connection Point Data Size MapMaddres EPI (communication could)	1 dapter) Point to po 0UT_254 100	v	

Step 4: Enter the connection status to check the connection status of InputOnly---IN_100. When the connection status shows successful connection and communication is normal, it indicates that the communication has been successfully established. The corresponding mapped register can also be used to check if the data is correct.

Step 5: Add slave devices to the scanner of PLC3: XDH-60T-E (IP 192.168.6.20) and perform relevant configuration operations on the adapter's IP address and compatibility check:

EtherNet/IP Scanner Config			×
Master Config EtherNet/IP Scanner	Routine Connection IOMapping Connection Status Address config IP Address: 192 , 168 , 6 , 6		
Slave Config StationIdO:XINJE EtherWet/IP	Compatible check Vendor ID: 1723 Ø Device Type: 12 Ø Product Code: 14 Ø Major Revisions: 1 Ø Minor Revisions: 1		
Slave Number: 1 Connection Number:	: 1/256 Theory throughput: 0 PPS Actual throughput: 0 PPS Import Export Upload DownLoad	Ok	Cancel

Step 6: On the scanner of PLC3: XDH-60T-E (IP 192.168.6.20), add a ListenOnly (ID Type) connection type to listen for the T ->O direction data sent by PLC1 to PLC2. Select multicast as the connection type, IN_100 as the connection point, 60 words in data size, HD0 as the mapping address, and 100ms as the RPI (communication period).

Master Config	Koutine Conne	action TOW	apping Conn	lection status	5				
EtherNet/IP Scanner	No Conne	ection	Input Connection Point	DataSize	IN Address	OUT Connection Point	DataSize	OUT Address	Connection ID
n dae de Marie	0 Listen	nOnly(ID	IN_100		HDO				
Slave Config —StationIdO:XINJE EtherNet/IP									
	-								
								* 1 3	D -1+
		-17						Add	Delect
	Connection Na	ame Liste	nOnly(ID Type))	~			Add	Delect
	Connection Na Time out(T)	ame Liste) RPI*1	nOnly(ID Type) 6 ~) (IN:1600ms 0		Configure Instance	1	Add ~	Delect
	Connection Na Time out(T) IN(Input from	ame Liste) RFI*1 the adapter	nOnly(ID Type) 6 ~)) (IN:1600ms 0	∨ 10T:1600ms)	Configure Instance OUT(Output to the s	1 idapter)	bba	Delect
	Connection Na Time out(T) IN(Input from Connection Ty	ame Liste) RFI*1 the adapter Type Multi	nOnly(ID Type) 6 ~) cast) (IN:1600ms 0	~ IVT:1600ms)	Configure Instance -OUT(Output to the e Connection Type	1 dapter) Foint to po	Add ~	Delect
	Connection Na Time out(T) -IN(Input from Connection Ty Connection Po	ame Lista) RPI*1 the adapter Type Multi oint IN_10	nOnly(ID Type) 6 ~) .cast 10)] (IN:1600ms 0 ,	∨ NUT:1600ms) ∨	Configure Instance -OVT (Output to the s Connection Type Connection Point	1 dapter) Foint to po OUT_255	Add ~	Delect
	Connection Na Time out(T) IN(Input from Connection Ty Connection Po Data Size	ame Liste) RPI*1 the adapter (ype Multi oint IN_10 = 60	nnOnly(ID Type) 6 ~ ~) :cast 10) (IN:1600ms 0	~ UT : 1600ms)	Configure Instance -OUT(Output to the e Connection Type Connection Point Data Size	1 dapter) Point to po OUT_255	Add ~	Delect
	Connection Na Time out(T) IN(Input from Connection Ty Connection Po Data Size MagMaddres	ame Liste) RPI*1 the adapter Type Multi oint IN_10 a 60 s HD0	nOnly(ID Type) 6 ~ ~) .cast 10) (IN:1600ms 0 , , , , , , , , , , ,	~ IVT:1600ms) ~))	Configure Instance -OUT(Output to the s Connection Type Connection Point Data Size MagMaddres	1 Point to pe OVI_255	Add ~	Delect
	Connection Ma Time out(T) -IN(Input from Connection Ty Connection Po Data Size MapMaddres Send trigge	ame Liste) RPI*1 the adapter (ype Multi oint IN_10 e 60 s HDO er Cycle	nOnly(ID Type) 6 ~ ~) .cast DO) (IN:1600ms 0 , , , , , , , , , , , , , , ,	<pre>></pre>	Configure Instance -OUT (Output to the s Connection Type Connection Point Data Size MagMaddres EFI (communication EFI (communication	1 Mapter) Point to po 0VT_255	Add ~ int (1-1%or (1-0653	d) Ems)

Step 7: Enter the connection status to check the connection status of ListenOnly---IN_100. When the connection status shows successful connection and communication is normal, it indicates that the communication has been successfully established. The corresponding mapped register can also be used to check if the data is correct.

Application 4: Using Xinje PLC XDH-60T-E as an adapter and Omron PLC NJ501-1500 as a scanner for implicit communication.

PLC1: XDH-60T-E (IP 192.168.250.20) as the adapter and PLC2: NJ501-1500 (IP 192.168.250.1) as the scanner to achieve implicit communication between two PLCs. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with the data size of the adapter.



Step 1: Add four connections on the adapter XDH-60T4-E in the direction of Adapter->Scanner(T->O).

Connection 1: Label name test a, mapping first address D0, data length 10.

Connection 2: Label name test_c, instance ID100, mapping first address is D30, data length 10.

Connection 3: Label name test e, mapping first address D60, data length 10.

Connection 4: Label name test_f, instance ID 102, mapping first address D80, data length 10.

Add two connections in the direction of Scanner->Adapter(O->T).

Connection 1: Label name test_b, mapping first address D100, data length 10.

Connection 2: Label name test_d, instance ID101, mapping first address D130, data length 10.

EtherNet/IP Adapter Config										
Master Config EtherNet/IP Adapter	Adapter->Sca	anner (T->O)			Scanner->/	Scanner->Adapter(0->T)				
	Number	Tag name	Living example	Map first address	Enter data length	Mumber	Tag name	Living example	Map first address	Enter data length
	0	test_a		B 0		0	test_b	1000	D100	10
	1	test_c	100	D30	10	1	test_d	101	D130	10
	2	test_e	<u>, 1997</u> ,	D60	10					
	3	test_f	102	D80	10					
	Label setti	ng			Label set	ting				
	Label	name	test_a			Labe	el name	test_d		
	Living	g example ID		u	se (100–199)	Livi	ing example ID	101	v 1	se (100–199)
	Map fi	irst address	DO			Map	first address	D130		
	Data 1	length	10	(1-7	24word)	Date	a length	10	(1-7	24word)
]L				Import	Outport	Upload	Downlo	ad Ok	Cancel

Step 2: Define the variables that require EIP communication in the Omron Sysmac Studio programming software. Global variables that require data reception and processing with the communication object during EIP communication are selected as inputs in the network public, while global variables that require data transmission and processing with the communication object are selected as outputs in the network public.

多视图浏览器 🚽 🕂	heat	自同变量 × EtherNe	t/IP设备列表 内置Ethe	erNet/IP端口设置	1连								
new_Controller_0	组第	読器 ▼ (没有组)	v										
▶ 配置和设置		名称	数据举题	2	初始值	分配到	保持	常量	网络公开		注释		
▼ 编程		test_1	ARRAY[110] OF INT				Z		输入	v			
V 🖉 POUs		test_2	ARRAY[110] OF INT				X		输出	v		1	
▼濡 程序	1000	test_3	ARRAY[110] OF INT				×		输入	*			
▼ 🖃 Program0		test_4	ARRAY[110] OF INT				×		输入	T			
L de Section0		test_5	ARRAY[110] OF INT				×		输入	T			
∟≋ 功能	1000	test_6	ARRAY[110] OF INT				Z		輸出				
∟≋ 功能块													
▼■ 数据													
2 影 教根类型													
● 全局变量													
▶ 雨 任务													
1													
1.27													
	1AU												
	181 au												• + ×
1 筛选器 ✔													

- Double click on the global variable to create a new one and add the variable type and data length to be transferred;
- Customize the name of the created variable;
- Define data types and lengths based on the length of input/output;
- Select the corresponding input and output types for the defined variable in the network public as needed.

Step 3: Enter the Ethernet/IP connection settings operation page, click on Tools in the function bar, select and click on Ethernet/IP connection settings, and finally double-click on the built-in Ethernet/IP port settings to enter the Ethernet/IP configuration interface.



Step 4: Enter the built-in Ethernet/IP port setting operation page, select the label group operation page, and register the global variables for input and output in the relevant network public. You can click on input/output to view the registered variable information.

	a 🛛 📼 🗗	人家创品教工	0 K A	a a k	· O 및	e c e	Q "2		
多视图浏览器 → 早	Mel 全局变量	EtherNet/IP设备列表 内置Ethe	erNet/IP端口设置 连	×					-
new_Controller_0 ▼	1	□- 标签组							
▶ 回画和IQ里 ▼编程 ▼ 値 POUs	o48	▼ 设备信息 型号名称 NJ501-1500 修订版 1.01							
▼ @ 程序 ▼ @ Program0 L & Section0		描述 节点地址 192.168.250.1 序列号 00000000		同步识别					
□圓 功能 □圖 功能块 ▼■ 数据		▼标签组 标签组/最大:6/32 标签/	偈大: 6 / 256				2	全部注册 导入	日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□		标签组名称	位选择	大小(字节)	大小(位)	I 实例ID	控制器状态		1
● 全局变量		▼ test_1		20		Auto	不包含		
► @ 1135		test_1		20	0				
		▼ test_4		20		Auto	不包含		
		test_4		20	0				
		▼ test_5		20		Auto	不包含		
		test_5		20	0		TEA		
		test_3		20 20	0	Auto	不包答		
		test_s		20	0				
		重启						L V	全部返回到默认值
							传送到控制	器 从控制器传送	比较
	输出								- I ×

Step 5: Go to the built-in Ethernet/IP port setting operation page and select the connection operation page. Right click on the blank space in the toolbox on the right side of the connection operation page to enter the displayed EDS library. Add the EDS file corresponding to Xinje Ethernet/IP to it.

文件(F) 编辑(E) 视图(V) 插入(I)	工程(P) 控制器(C) 模拟(S) 工具(T) 窗口(W) 帮助(H)		
	· ■ # < ₩ G R # A 0 K <u>A ≫</u> % # % * 0	4 P I Q Q R	
多視問刘范器 - 9	EtherNet/PI设备列表 内面EtherNet/P算口段音 连×		- I月初
new_Controller_0	□- □		目标设备
> 配置和设置			
 ● 監理 ● COLS ◆ IR 和P. ● Program0 C. IS Section0 C. IS Julith ● TAR ● TAR ▶ Th 任务 	Y 通報 諸規(長大 0 / 32 日時記録 1 施設名称 1 连接//公売型 1%A入和出1 日時受量 ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	Wendor Vaskaw Electric America, Inc. OMKON Corporation Omron Adept Technologies, Inc. Omron Microscan Systems, Inc. XINIE ElectricCo.Ltd	
	1938 <mark>≸ 100 mail ANU mail । । 1 3938 - 1 103≉ । 0238 1</mark>	安莱 3 英团	

Step 6: Click the add button + in the toolbox on the right side of the connection operation page, and three operation requirements for adding objects will appear: node address (IP address of the object to be connected), model name (matching EDS file of the object to be connected), revised version (select the version of the EDS file of the connected object), and the operation is shown in the figure. After the establishment is completed, click the add button, and the addition and configuration information will be completed as shown in the figure below:

EtherNet/IPi@4	资则表 内等FtherNet/IP端门设置 连 ¥			
-	•••••••••••••••••••••••••••••••••••••			目标设备
世	▼ 连接 连接/ 優大:0 / 32 日标设备 连接名称 连接//0类型	1翰入/翰出 目标变量 大小(字节] 起始变量	- (大小字节) 连接关型 (RPI(毫约 超时值)	支量名 大小(字节)
	★ 回 ○ □ □			
	重启		全部返回到默认(传送到控制器 从控制器传送 比较	
编译 <u> 第 1 </u> 1	します。 説明 I 程序 I	位置		1 ×

EtherNet/IP设	例表 内置EtherNet/P論目设置该×	工具箱 → ↓
0-	<mark>□</mark> 连接	节点地址 192.168.250.20 型号名称 XINJE EtherNet/IP ▼ 修订版 1 ▼
	▼连接 连连发展大:0 / 32 目标设备 连接名称 连接//0类型 H输入/输出 目标变量 大小/字节] 起始变量 大小/字节] 连接类型 RPI[零秒 超时值 [
编译 <mark> </mark>	→ 1 x 減用 程序 位置	
输出编译		添加取消

Step 7: Right click on the blank space in the toolbox or the blank space in the connection area to add a connection.

EtherNet/IP设计	的表内置EtherNet/IP端	111设置连×				•	
.							目标设备 192.168.250.20 XINIE Ethe
	▼ 连接						
n.f.B	连接/最大:0/32				法检察期 1000高级 2004度		4
- 40	日彻收留	建後有称 建後1/0米型 :			JEBGROUP INPIGERATING MEMORE		
	添加(A)	1					
	删除(D)						• + •
	更改节点地址(N)						
	. 史或目标反音(1) 参当(1)						●
	重做(R)						
	全部选择(S)						
	论体带来						
	56 B1020						
	重启					全部返回到默认值	
				JESA TRAVE		Heath .	
				141219111	2011年1月1日1日1日1日1日1日1日1日1日1日1日1日1日1日1日1日1日1	L HURX	
编译						- 1 X	
20 错误	0 ===						
1 1	说明	程序	位置			I	
							TO S SCHOOL
输出编译							有人的应用

Step 8: Add ExclusiveOwner (Tag Type), ExclusiveOwner (ID Type), and IputOnIy (Tag Type) and IputOnIy (ID Type) connections, and communicate with tag variables or instance IDs respectively. The configured variable types are shown in the following figure:

M 全局变量	EtherNet/IP设备列表	内置EtherNet,	1P蠲口设置连 ×									-	工具箱	, ą
•	••••• 连接												目标设备 192.168.250.20 XINJE E	the
	▼ 连接													
- CH														
L.C.	目标设备 🔺	」 连接名称	连接 /0类型	输入/输出	目标变量	大小[字节]] 起始变量	大小[字节] 连接类型	RPI[毫秒	超时值			
	192.168.250.20 XINJE Ether	default_001	ExclusiveOwner(lag lyp	输入	test_a	20	test_1	20	Point to Point connection	50.0 H	RPI x 4	-8		
				輸出	test_b	20	test_2	20	Point to Point connection			_		
	192.168.250.20 XINJE Ether	default_002	ExclusiveOwner(ID Type	输入	100	20	test_3	20	Point to Point connection	50.0 F	RPI x 4			
				输出	101	20	test_6	20	Point to Point connection					
	192.168.250.20 XINJE Ether	default_003	InputOnly(Tag Type)	输入	test_e	20	test_4	20	Point to Point connection	50.0 F	RPI x 4			
	192.168.250.20 XINJE Ether	default_004	InputOnly(ID Type)	输入	102	20	test_5	20	Point to Point connection	50.0 F	RPI x 4		杏县名 大小[字书	1
	+													
	设备带宽													
	重启								ļ	全部返回	到默认值			
							1	送到控制器	从控制器传送	Ж	较			
輸出											* 1	×		

Step 9: After completing the information configuration, click on "online", then click on "synchronize" to download the project information to the controller, and finally click on "transfer to controller" to transfer the connection configuration information to the controller.

Perec Controller ①		····	EtherNet/IP设备列表	内端EtherNet/	19月1日日日 15 ×							_	12	工具箱
 ・ 注注 ・ 注注 ・ 注注	Controller_0	Π۰	•••											目标设备 192.168.250.20 XIN
	Ŧ		▼這接											
◆ (# 1) (* 1)	d POUs	n-f-11	连接/最大:6/32	1 10-10-07-02	はた時にの発用	180.3 /8014	日協商業	1 de de Pile alto	1 404605.00	1 deals references	1. 次後米刑	IDDI(751	61 800+68 I I	
 ● Program0 Le SocianO Le SocianO Le SocianO Le SocianO 	▼無程序	- Lii	102 158 250 20 VINIE Ether	default-001	EvolusiveOwner/Tag Tvn	18人/1808 46 入	test a	20	test 1	20	Point to Point connection	50.0	RPI v 4	
・ (* Section) ・ (* 7)が能 ・ (* 7)が能決 ・ (* 7)が能決 ・ (* 2) ・ (*	▼ ⊟ Program0		THE POLED HERE EAC	, denouil_out	contracting typ	\$0.H	test b	20	test 2	20	Point to Point connection		IUTA 4	12 + B
Ling (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)	L∉ Section0		192,168,250,20 XINJE Ether	default 002	ExclusiveOwner(ID Type	输入	100	20	test 3	20	Point to Point connection	50.0	RPLx 4	
Log AllingCole State	LIE 初館		Mark Contractor South Contra			输出	101	20	test 6	20	Point to Point connection	-		受量名 │ 大小[
Comparison	- 新語		192.168.250.20 XINJE Ether	default_003	InputOnly(Tag Type)	输入	test_e	20	test 4	20	Point to Point connection	50.0	RPI x 4	
			192.168.250.20 XINJE Ether	default_004	InputOnly(ID Type)	输入	102	20	test_5	20	Point to Point connection	50.0	RPIx4	
			+ 0 2472 12									全部返	回到默认蛋	

Step 10: Control the input and output data, and check whether the sending and receiving data is normal through the monitoring window.

Verl 全局变量	EtherNet/IP设备列表	内置EtherNet/	IP簧口设置 连 ×									▼ 工具箱	- ù	
0-	□ □ 佳 注 注 注 ·										目标设备 192.168.250.20 XINJE Ethe			
	▼ 连接 连接/最大:6 / 32	连接 连接/最大 6 / 32												
ot:B	目标设备 🔺	」 连接名称	连接I/O类型	输入/输出	目标变量	大小 字	[节] 起始变量	大小 字	·节] 连接类型	RPI[毫秒	匀 超时值	ו		
	192.168.250.20 XINJE Ether	default_001	ExclusiveOwner(Tag Typ	输入	test_a	20	test_1	20	Point to Point connection	50.0	RPI x 4		-	
				输出	test_b	20	test_2	20	Point to Point connection		1		.	
	192.168.250.20 XINJE Ether	default_002	ExclusiveOwner(ID Type	输入	100	20	test_3	20	Point to Point connection	50.0	RPI x 4		大小(字节)	
				输出	101	20	test_6	20	Point to Point connection				,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	
	192.168.250.20 XINJE Ether	default_003	InputOnly(Tag Type)	输入	test_e	20	test_4	20	Point to Point connection	50.0	RPI x 4			
	192.168.250.20 XINJE Ether	default_004	InputOnly(ID Type)	输入	102	20	test_5	20	Point to Point connection	50.0	RPI x 4			
	+ 🗇													
	设备带宽													
										A #00mm	7708831.00	-		
	里后									王部返回	回到默认值			
	(6)X 51455428 X 1454428 H 455													
							14	Notification of the second		-	G4X			
收加/工程\1												×		
设备名	称	名称	在线值	修改		注释	数据类型	1	分配到 显示	格式	1			
new_Control	ler_0 test_1[1]		1212				INT	ĺ.	Decin	nal 🔽				
new_Control	ler_0 test_3[1]		1213	(INT		Decin	nal 🔻		导入标签组		
new_Control	ler_0 test_4[1]		1214				INT	ĺ	Decin	nal 🔻		-		
new_Control	ler_0 test_5[1]		1215				INT		Decin	nal 🔽		控制器状态	- 	
new_Control	ler_0 test_2[1]		1010	1010			INT		Decin	nal 🔻		1	×	
new_Control	ler_0 test_6[1]		1020	1020			INT		Decin	nal 🔻		在线 🧧	192.168.250.1	
new_Control	ler_0 题入名称											ERR/ALM	运行模式	
<u>11</u>														
rNet/IP Adapter	Adapter->S	canner (T->O)				Scanner->	dapter(0->T)							
-----------------	------------	---	-------------------	----------------------	----------------------	-----------	---------------	-------------------	----------------------	---------------------				
	Number	Tag name	Living example	Map first address	Enter data length	Mumber	Tag name	Living example	Map first address	Enter dat length				
	0	test_a	_	DO	10	0	test_b	555	D100	10				
	1	test_c	100	D30	10	1	test_d	101	D130	10				
	2	test_e	<u>, 2005</u>	D60	10									
	3	test_f	102	D80	10									
				Add	Delete				Add	Delete				
	Label set:	ing				Label set	ting							
						100		-						
	Labe.	l name	test_a			Labe	el name	test_d						
	Livi:	ng example ID		u	se (100–199)	Livi	ng example ID	101		se (100-199				
	Мар	first address	DO			Map	first address	D130						
	Dias	1	10	(1-7	(L	Dete	1	10	(1-7)	(L				
	Data	Tengtu	10	(1-)	24word)	Date	r rength	10	<u> </u>	24wor d)				
					Import	Outport	Upload	Downlo	ad Ok	Can				
	DI CI	that the state of	-				14.1			1.12				
	监控	· · 》》	15 4820	憲論 全部連	能 上移 下	¥8								
	2.87	and the second	112120	****	st êt i n									
		0	1010	THE	前空									
		30	1212	INI	単子									
		60	1214	TNT	9.52									
		80	1015	INT	単字									
	-0 1	100	1010	INT	単字									
			1010											

Application 5: Use Xinje PLC XDH-60T-E as the scanner and Omron PLC NJ501-1500 as the adapter for implicit communication.

Use PLC1: XDH-60T-E (IP 192.168.250.20) as the scanner and PLC2: NJ501-1500 (IP 192.168.250.1) as the adapter to achieve implicit communication between two PLCs. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with the data size of the adapter.

Step 1: Define the variables that need to be used for EIP communication in the Omron Sysmac Studio programming software. During the EIP communication process, the global variables that need to be sent and processed with the communication object are selected as output in the network public.

名柳图浏览器 _ 1		/ID治夕冽主 由罢CebaeNiat/ID浴》	7沿要 法	_	_	_	_	 2-1
		/irix=792 PyliculeInet/irjan	-102 <u>H</u> (±					
▼ 配置和设置	名称	数据类型	初始值	分配到 係	特常量	网络公开	注释	
	test_aa	ARRAY[110] OF INT			Y 🗆	₩出 ▼		
► © CPU/扩展机架	test_bb	ARRAY[110] OF INT		8		输出 🔻		
■ ** I/O 映射								
▶ ℝ 控制器设置								
▶ 👳 运动控制设置								
er Cam数据设置								
■ ▶ 事件设置								
■ 🖻 任务设置								
■ 図数据跟踪设置								
▼ 编程								
🔲 🔻 💼 POUs								
■ ▼鳳 程序								
🖉 🔤 Program0								
L 雪 Section0								
■ ∟罵 功能								
■ こ園 功能块								
▼■ 数据								
∟ 🖂 数据类型								
■ 1= 全局变量								
▶ 由 任务								

- Double click on the global variable to create a new one and add the variable type and data length to be transferred;
- Customize the name of the created variable;
- Define the data type and length based on the length of the output;
- Select the corresponding output type for the defined variable in the network public as needed.

Step 2: Enter the Ethernet/IP connection settings operation page, click on Tools in the function bar, select and click on Ethernet/IP connection settings, and finally double-click on the built-in Ethernet/IP port settings to enter the Ethernet/IP configuration interface.



Step 3: Go to the built-in Ethernet/IP port setting operation page, select the label group operation page, and register the output global variables in the relevant network public. You can click on the output to view the registered variable information.

₩ 全局交量	Et	herNet/IP设备列表 内	I置EtherNet/II	P端口设置 连 🗙							
0-	Ľ	- 标签组									
□ + 0 □	▼ 谈 型号 (と备信息 名称 NJ501-1500 印版 1.01									
	节点 月	地址 192.168.250.1 列号 0000000			同步识别						
	▼ 8	i签组 iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii	标签/最大:	2 / 256					全部注册	导入	导出
	输.	1 输出									
		标签组名称 test aa		位选择	1 大小(字节)	大小(位)	实例ID	控制器状态	输出严重错误	1	
	P	test aa		6	20	0	100	不包占	· 清除		
	-	test_bb		6	20		101	不包含	1312		
		test_bb		6	20	0			清除		
		重启								Ξ	部返回到默认值
								传送到控制	器 从控制	器传送	比较

Step 4: After completing the information configuration, click on "online", then click on "synchronize" to download the project information to the controller, and finally click on "transfer to controller" to transfer the connection configuration information to the controller.

2	30	-5ª <	. 24 53	Fin 188	H 🖲	民	A 🔉	63 6 3	% %	0	21 69	<u> </u>	Q, "Q									
	Vari 全局变	编 Ef	therNet/IP该	格列表 🔥]置EtherNet/	1P端口设计	≝连 ×				2								•	工具箱		÷ (
	0.		- 标签	翅															Ĩ	目标设备		
	0-10 0-10	▼ 型: 1	设备信息 号名称 NJ50 修订版 1.01 描述 点地址 192.	1-1500	-															1		
I		F	亨列号 0000	0000				同步	识别											变量名	i I	大小[字节]
l			标签组 标签组/最大	: 2 / 32	标签/最大:	2 / 25	6								全部注册	导入		导出				
I		输	入輸出	1- MAY 101 1-1-1			VIE AND	· L.I.m	-	1.1.76	** .	challen			401117777644	a .						
I			test aa	机应温着树	Υ.		西岸	20	(CT:	大小(1)	10	<u> 実例</u> ID	不包含	祆念	制出产里错	₹			-			
I			test_aa			F -		20	0				1 44		清除				- 11			
I			test_bb					20			10	1	不包含									
I			test_bb					20	0						清除							
l		L																				
l			重启														全部進	回到默认值	1			
													「「「「」」「「」」「「」」「「」」「「」」「「」」「「」」」「「」」「」」「	关到控制	器 <u>从</u> :	空制器传送		比較	1	导入标签组		
	輸出																	••••••	×	控制器状态 在线 ERR/ALM	•	↓ (】) 192.168.250. 运行模式

Step 5: In the Xinje XDPPRO programming software scanner, add the Omron NJ501-1500 slave device and perform relevant configuration operations on the adapter's IP address and compatibility check:

EtherNet/IP Scanner Config		×
Master Config EtherNet/IP Scanner	Routine Connection IOMapping Connection Status Address config IF Address: 192 , 168 , 250 , 1	
Slave Config —StationIdO:NJ501-1500	Compatible check	
	Vendor ID: 1723	
	Froduct Code: 14 🗹 Major Revisions: 1	
	Minor Revisions: 1	

Step 6: Click on the connection to add two types of connections, Input Only (ID Type) and Input Only (tag Type). Connection 1: label name test_bb, data length 10, mapping first address HD10. Connection 2: instance ID (IN_100), input data length 10, mapping first address HD10.

After configuration is completed, click download to download configuration to the PLC. The configured variable types are shown in the following figure:

Master Config	Routine	in romapping con	neetion state		1.200			
therNet/IP Scanner	No Connectio	on Connection Point	DataSize	IN Address	OUT Connection Point	DataSize	OUT Address	Connectio ID
Se Sel March	0 InputOnly	(ID IN_100	10	НОО	<u></u>	1 <u>-</u> 20	-	0
ave Config	1 InputOnly	(Tag test_bb	10	HD10		-	-	1
	Connection Name	InputOnly(Tag Type	e)	~			Add	Delect
	Connection Name Time out(T)	InputOnly(Tag Type RPI*16	≥) ∕ (IN:800ms 0	∨ VT:1600ms)	Configure Instance		Add	Delect
	Connection Name Time out(T) IN(Input from the	InputOnly(Tag Type RPI*16 adapter)	e) / (IN:800ms 0	∨ VT:1600ms)	Configure Instance	adapter)	bbă ~	Delect
	Connection Name Time out(T) IN(Input from the Connection Type	InputOnly(Tag Type RPI*16 adapter) Point to point	8)	∨ UT:1600ms)	Configure Instance OUT(Output to the a Connection Type	adapter) Foint to poi	× Add	Delect
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point	InputOnly(Tag Type RFI*16 adapter) Point to point test_bb	e) - (IN:800ms 0	∨ VT:1600ms)	Configure Instance -OUT(Gutput to the s Connection Type Connection Point	adapter) Foint to poi	Add ~	Delect
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size	InputOnly(Tag Type RFI*16 adapter) Point to point test_bb 10	e) < (IN:800ms 0 (1-724Wor	✓	Configure Instance -OUT(Output to the e Connection Type Connection Point Data Size	adapter) Point to poi	Add	Delect
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size MapMaddres	InputOnly(Tag Type RPI*16 adapter) Point to point test_bb 10 HD10	e) (IN:800ms 0 (1-724Wor	 ✓ ∪T: 1600m ±) ✓ → →	Configure Instance -OUT (Output to the s Connection Type Connection Point Data Size MapMaddres	adapter) Point to poi	xdd	Delect
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size MagMaddres Send trigger	InputOnly(Tag Type RFI*16 adapter) Point to point test_bb 10 HD10 Cycle	e) (IN:800ms 0	✓ UT: 1600m±) ✓ → →	Configure Instance -OUT (Gutput to the s Connection Type Connection Point Data Size MagMaddres EFI (communication cycle)	Point to point to point 100		d)

Step 7: Click on IO mapping or connection status to view the current communication data or connection status.

		+=/	<i>≿</i> /□														
I ∏⊷		- 你公	立组														
	-																
	下法	洛信息		_	_	_	_	_			_		_	_	_		
(*II	- 15														_		
00		ionz sta			2011 34 10	MOX.	_				_	-					
	杤	祾组/最大	た:2/32	标签/最大	: 2 / 2	56							全部注册			~	导出
	46h λ	40.55															
	-1817		标签组合的	17 N	1 1	**** 探	1 ====	(字共)	十小人ない	া জা/ম	lin	- 长期課程太	1 46413		1		
			1002-11-14	r.		27661 1 -	20	•(- -)/ 1	入(1)(1)	100				王伯庆			
		test_da					20			100		小包古					
		test_aa	í				20		0				清除				
	•	test_bb					20			101		不包含					
		test bl)				20		0				清除				
		-									_	1					
		重启														全部返[回到默认值
												#=3475(H-5-8	LIBR I	1142544415	10/#:\¥	11	Lity
												15达到控制	同語	从控制	品15)达	1	6492
监视(工程)1																	- I >
设备公	古称		\$	部		左线 值		修行行	1 34	12	г ж	6/据光刑 1	分配等	1		息示格式	
new Control	ller 0		test aa[1]	1101	10	100		100	, <u>, , , , , , , , , , , , , , , , , , </u>	n+-	INT	AMAXE 1	7J EUJ	9			
new_control	IICI_0	-	test_dd[1]			100		100			INT						
new_control	lier_0	_	test_aa[2]			101		101			INI					Jecimai 🔽	
new_Control	ller_0		test_bb[1]			102		102			INT					Decimal 🔽	
new_Control	ller_0		test_bb[2]			103		103			INT					Decimal 🔻	
new Control	ller 0		输入名称														
		10	Lance and the second second				13								18		
EtherNet/ID S	canne	r 参数而管	-														×
Etherive gir 50	canne	- Se Sou Huge	1			(T										~
主站配置				常规	连接	IO映射	_ 连接状;	 态									
EtherNet/IP	Scann	er		标签		通道		映射地址	数值								
2 01101 110 0, 11	Douran			- Input	Only												
				IN_	100												
				<u>⊞</u> I	N_100[0]	InPut		HDO	100								
从站配置				⊞I	N_100[1]	InPut		HD1	101								
-StationId0	:NJ50	1-1500		H HI	N_100[2.	InPut		HD2	0								
					N_100[3. N_100[4]	InPut		HD3	0								
				H H	N 100[4	InPut		HD4 HD5	0								
				Ē	N 100[6]	InPut		HD6	0								
				Ē	N_100[7]	InPut		HD7	0								
				ΞI	N_100[8]	InPut		HD8	0								
				⊞ I	N_100[9]	InPut		HD9	0								
				Input	Only												
				tes	t_bb	TerDut		ND10	102								
					est_D	InPut		HD10 HD11	102								
				Ēt	est h	InPut		HD12	0								
				⊞t	est_b	InPut		HD13	0								
				⊞ t	est_b	InPut		HD14	0								
				⊞ t	est_b	InPut		HD15	0								
					est_b	InPut		HD16	0								
					est_b	InPut		HD17 HD19	U								
					est_D	InPut		HD19	0								
						mut			×								
				_													
当前从站个数:	1	当前连接	个数: 2/256	理证	论吞吐量:	60 PPS	实际吞吐	量: 162	PPS	导入配置		导出配置 上	传配置	下載配	罟	确定	取消
															_		100000000000000000000000000000000000000

Application 6: Use Xinje PLC XDH-60T-E as the adapter and Keyence PLC KV-5500 as the scanner for implicit communication.

PLC1: XDH-60T-E (IP 192.168.6.6) as the adapter and PLC2: KV-5500 (IP 192.168.6.10) as the scanner, implicit communication between two PLCs can be achieved. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with that of the adapter.

Step 1: Add three connections in the adapter XDH-60T4-E in the direction of Adapter->Scanner(T->O).

Connection 1: instance ID100, label name test_a, mapping first address D0, data length 20.

Connection 2: label name test_c, mapping first address D200, data length 100.

Connection 3: instance ID102, label name test_d, label name test_d, mapping first address D300, data length 80.

Add a connection in the direction of Scanner->Adapter(O->T), instance ID101, label name test_b, mapping first

address D100, data length 20.

aster Config therNet/IP Adapter	Adapter=>Sc	anner (T->0)				Scanner->A	.dapter(0->T)			
	Number	Tag name	Living example	Map first address	Enter data length	Mumber	Tag name	Living example	Map first address	Enter data length
	0	test_a	100	DO	20	0	test_b	101	D100	20
	1	test_c	New York	D200	100					
	2	test_d		D300						
				Add	Delete				Add	Delete
	-Label setti Label	ing	test d	Add	Delete	Label set	ting 1 name	test b	Add	Delete
	Label setti Label	ing . name	test_d	Add	Delete	Label set	ting 1 name	test_b	Add	Delete
	Label setti Label Livin	ing name 1g example ID	test_d	Add U	Delete	Label set Labe Livi	ting 1 name ng example ID	test_b	Add	Delete
	Label setti Label Livin Map f	ing name 1g example ID first address	test_d 102 D300	Add	Delete	Label set Labe Livi Map	ting 1 name ng example ID first address	test_b 101 D100	Add	Delete

Step 2: In the Keyence KV STUDIO programming software, after connecting to the PLC to be communicated, double-click KV-5500 under the unit configuration to enter the unit editor - edit mode. Double click the CUP unit to configure its IP address, ensuring that it is in the same network segment as the adapter.

坝目 4 X	Main X			
= [0] KV-5500 R000/R500 1	📟 单元编辑器 - 编辑模式		- 🗆 X	
EtherNet/IP R30000 IM10000	文件(E) 编辑(E) 转换(P) 视图(V) 选项(O) 窗口(W) 帮助(H)		9	10
	A DALY DE RAMANA REPERT			
数元件注释				
□□ 10万 □□ CPII 系统设定	······	单元	Ф.	
■ ▲程序: EIP_test	R: SSmm XV-5500 and Unit	选择单元(<u>1</u>) 设定单元(<u>2</u>)		
■ · · · · · · · · · · · · · · · · · · ·	FT: SOmm R000	19: 2	[0] KV-6500	
11600 招快	消耗电流: 320mA	□基本	•	
	±Ξ: 350g -507	- エ・ 首 IM 编号 IM10000		
📄 固定周期模块	R30000	占用 11 約 230		
■ 🗃 宏	-33915	首绯电器编号(按诵道设定) R30000		
	lil.	占用继电器约 640		
1111日11日本	ö	通信速度 100/10Mbps自动(大)		
一一 设定文件寄存器	4	- IP 排射设定方法 固定 IP 排射(火)		
	3	IP 地址 192.168.6.10		
		子阿撞码 255.255.255.0		
		默认网关 0.0.0.0		
		DMS 服务器 0.0.0.0		
		接收超时[s] 10		
		Keep Alive[x] 600		
		—————————————————————————————————————		
		精计与		
	消息		4	
	定理 打 骗亏 飞船 洞思			
	H	<	>	
库 项目		信備課 1 行 1 列 OK 取当	成田	
tela		11, 171 OK 2019	1 12/13	通り十四 102 169 6 10
初、第			Ð	# KAAMI 192.108.0.10

Step 3: In the programming mode of the unit editor, find the Ethernet/IP settings, click the function key on the right side of the Ethernet/IP settings, and enter the Ethernet/IP settings configuration interface.

	单元		ą
XY-5500 End Unit	选择单元(1) 设定单元(2)		
8000	PE 📮 🖂 🖷 🐏 📫 🏪 🖏	[0]	KV-5500
	路由设定	不执行(*)	^
-507	□ EtherWet/IP 设定		
83000	自动分配设定	有效(*)	_
	分配位软元件起始编号	B0000	
	分配字软元件起始编号	90000	
	刷新上限数(字/扫描)	252	
	隐式(I/0)报文通信自动开始	执行(*)	
	隐式(I/O)报文通信错误检测掩码时间	60	
	隐式(I/O)报文通信错误检测掩码时间	5	
	显式报文通信超时 [ms]	10000	
	重试时间(系统扩展)[s]	60	
	组播用 TTL	1	
	组播地址指定方法	自动分配(大)	
	组播地址数	256	
	组播起始地址	239.255.0.0	
	启用 IGHP 查询发送	无效(*)	
	IGMP 查询发送间隔[s]	60	
	EtherNet/IP 设定	〈设定〉	
	□ FIP 客户端设定		
	FTP 客户端设定	〈设定〉	_
	□ FTP 服务器设定		~
	EtherWet/IP 设定 启动 EtherNet/IP 设定。		
			ç

LtherNet/IP 设定						- 🗆	\times
文件(F) 编辑(E) 设定(S) 视图(V) 转	换(C) EDS 文件(D)) 通信(N)	工具(T)	帮助(H)			
📲 🛈 👷 🕾 🖧 🖬 📩 🛤 👌	/ 😪 🙉 🔍 🗹		2				
KV-5500[0] : 192.168.6.10			EtherNe	et/IP设备			ņ
			设备列表	表(1) 设备设定(2)	设f	备查找(3)	
				ŢE ♀= ▓			
				设备名称	Rev.	EDS 文件注释	^
			🔳 🚰 Ke	eyence Corpora			
			EDS	KV-5500	1.1	KV-5500 CPU Unit	
			EOF	KV-7500	1.1	KV-7500 CPU Unit	
			303	KV-8000 Series	1.1	KV-8000 Series C	
				KV-EP02	1.1	EtherNet/IP Comm	•
				KV-N16ER	1.1	16-point relay o	•
				KV-N16ET*	1.1	16-point transis.	•
				KV-NIGEX	1.1	16-point input uni	t
				KV-NJAM	1.1	2+1ch analog 1/0.	•
				KV-NOEK	1.1	S-point relay ou.	· ·
			<				>
輸出							ņ
ra ita M 🎜 🕏 🦓 🔛							
节点节点名称	IP地址		连接名	3称	RPI[I (ms)	N] RPI[OUT] (ms)	超时
	ut						
▶ \消息 < 校验 > 设定列表 /							>
		编辑器	8	OK		取消 应用	

Step 4: Right click on the blank space of "EtherNet/IP Device" or "EDS File (D)" on the function bar to add XDH-60T4-E as the EDS file for the adapter. After adding, you can view the corresponding XINJE EtherNet/IP EDS file in "EtherNet/IP Device".

LtherNet/IP 设定	2				- 0	×
文件(E) 编辑(E) 设定(S) 视图(☑ 转换(C) EDS 文件(D)	通信(N) 工具(T) 帮助	助(<u>H</u>)			
📲 🕼 🛱 🖓 🖓 👘 🐘	🖻 🌮 🗟 🖗 🖗 😿	la lit 🕜				
KV-5500[0] : 192.168.6.1	0	EtherNet/IP	公告			, p
		设备列表(1)	设备设定(2)	设备查	我(3)	
		ìş	备名称	Rev.	EDS 文件注	释
		🕂 🎽 Keyen	ce Corpora			
			ElectricC			
		ELS XINJ	E EtherNet/IP	1.1 XI	NJE EtherNet/	IP S
		<	1			>
输出						Ф
₽ ₽ ₽ ₽ ₽						
节点 节点名称	IP地址	连接名称		RPI[IN] (ms)	RPI[OUT] (ms)	超时
	/	[] < 编辑器	ОК	取	消应	>

Step 5: Double click on the XINJE EtherNet/IP EDS file to use it as an adapter, and configure its adapter IP address in the adapter initial settings dialog box.

📕 EtherNet/IP 设定					- 🗆	×
文件(F) 编辑(E) 设定(S) 视图(V) 转	换(C) EDS 文件(D)) 通信(N) 工具(T) 帮助	(H)			
📲 🕼 🐕 🖏 🖧 🖏 👘 🖧 🖏 👘 🖧	/ 🐔 🚳 🧠 🗹	la li 🕜				
KV-5500[0] : 192.168.6.10		EtherNet/IP设	备			ņ
		设备列表(1)	设备设定(2)	设备查	找(<u>3)</u>	
1: XINJE EtherNet/IP : 192.	168. 6. 1	设备	各名称	Rev.	EDS 文件注	释
		+ Keyence	e Corpora Corporation			
活配器初始设定			lectricC			
		EDS XINJE	EtherNet/IP	1.1 XI	NJE EtherNet/	IP S
节点地址(A)						
IP地址(I) 192 . 168	. 6 . 1					
连接名称 正 10 月 (下) 第	I	/0				
InputUnly(ID Type)	1 14					
	01/ HR	× ×				>
	UK HU	に 対応 対応 対応 対応 に た た た た た た た た た た た た た	rHet/IP[1.1] et/IP Slave S	 tation		
输出						ņ
🖻 💼 🗰 ङ 🗟 🖳 🏪						
节点 节点名称	IP地址	连接名称		RPI[IN] (ms)	RPI[OUT] (ms)	超时

Step 6: Click the "+" corresponding to the added adapter to enter the connection setting configuration interface. Click "Add" in the configuration interface to add the specified connection name according to the application type. Select the corresponding connection name and click on parameter settings. According to the size of the adapter's configured data, configure the data size in the scanner accordingly. After completing the configuration, click OK to complete the parameter configuration. Finally, click Download to download the configuration information to the PLC controller.

KV-5500[0] : 192.168.6.10	REPERCASE TO ATTACK	anomegn i v		EtherNet/IP设备
	连接列表(L) No.	注接名称 応用类型 (T) Type) (T) Type) (T) Type)	参数设定 ×	设备列表(1) 设备设定(2) 设备重找(3)
TARANA STREAM AT THE ACT OF A C.	2 Input0aly(Ta 3) Input0aly(Ta 3) Input0aly(Tb 2) 違振名称(C) 超时(T) 粉新优先权(E) 3 取(从近配器输入)) 違接类型	# Type) [text.c] # input only Type) [IN_102] # input only #(1) ExclusiveOener(ID Type) -般 (部前電化)	**教(中) **(100): EffectiveC21 *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** ***	1 法代表保全 书点地址 17/2址 192.166.6.6 节点名称 17/2 Line/#e/17 产品名称 17/2 Line/#e/17 用の名称 17/2 Line/#e/17 供の名称 17/2 Line/#e/17 供の名称 工程2 Line/#e/17 代表の記載現金 代表の記載現金 代表の記載現金 代表の記載現金 代表の記載現金 代表の名称 日 代表の名称 日本 代表の名称 代表の名称 代表の名称 代表の名称 代表の名称 代表の名称 代表の名称
	连接点 数据大小 发送触发器 时(通信周期) 最小发送间隔 QUT(输出到话配器)	IN_100 ~	10月 Dete Site 型は値 2 万辺 1 - 1448 当崩的分型値 40 byte 発注 毎注 受 (変更力量は値(型)) の版 取済	情影器加张 函 基著性检查 产品化集 装备的名称。
H	连接类型 连接占	点对点 007 101		
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	数据大小 RPI(通信周期)	20 100.0 st (1.0 - 10000.0ss) ▽与IF接持数 6 0次 取済		

Step 7: Double click XINJE Ethernet/IP in monitor mode, operate and monitor the corresponding data in the IO monitoring table, and verify whether communication is normal.

-(E) 編辑(E) 视图(V) 程序(M) ST/曲	本(S) 转换(A) 监	控器/模拟器(N) 调试		1山(W) 帮助(H)	
🕐 🖶 🖻 🛍 🛍 🐘 🖶 🚱 🚺	11 以太网	· D F 4	II 🗈 🖸 🚮 🗄	[표 🕼 📟 🗄 부 환 부 환 후 후 키 키 ㅋ	SF9
🗄 🏣 🌌 🐼 📰 🖷 🏅 🗞 🕏 晃		W A H A W) > O 🚚	- ◎ 💷 监控器 🔹 注释 注释	1 +
4 ×	Main X			1	
单元弱署				•	
				1 (1.5.9)	
E . EtherNet/TP R30000 DW10000	€ 传感器IO监控	:KV-5500[0].XINJE Eth	erNet/IP[1]	3	- 🗆 🗙
[1] XINTE EtherNet/IP			22.22		
	软元件	当前值	显示格式	注释	^
	W00	12858 1	6 位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[0]	
[1] 医结识中	w01	11 10	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[1]	
	W02	12 10	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[2]	
	W03	13 16	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[3]	
	W04	14 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[4]	
ani Main	¥05	12858 1	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[5]	
利潤化模块	W06	15 16	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[6]	
一 后奋模块	w07	16 16	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[7]	
□ 固定周期模块	w08	17 10	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[8]	
■ 宏	W09	18 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[9]	
- 📘 子程序型宏	WOA	123 1	6 位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[10]	
- 📴 自保持型宏	WOB	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[11]	
🖩 软元件初始值	W0C	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[12]	
1 设定文件寄存器	WOD	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[13]	
	WOE	0 16	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[14]	
	WOF	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[15]	
	w010	11 16	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[16]	
	W011	0 16	5位十进制数	KV-5500[0]. XINJE BtherNet/IP[1]IN_100[17]	
	W012	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]IN_100[18]	
	W013	0 16	6位十进制数	<pre>KV-5500[0]. XINJE EtherNet/IP[1]IN_100[19]</pre>	
	W014	666 1	8 位十进制数	<pre>KV-5500[0]. XINJE EtherNet/IP[1]OUT_101[0]</pre>	
	W015	777 1	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]OUT_101[1]	
	W016	888 1	5位十进制数	KV-5500[0].XINJEEtherNet/IP[1]OUT_101[2]	
	W017	0 1	6 位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]0UT_101[3]	
	W018	0 16	5位十进制数	<pre>KV-5500[0]. XINJE EtherNet/IP[1]OUT_101[4]</pre>	
	W019	0 16	5位十进制数	<pre>KV-5500[0]. XINJE EtherNet/IP[1]OUT_101[5]</pre>	
	W01A	0 16	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]0UT_101[6]	
	W01B	0 14	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]OUT_101[7]	
	¥01C	0 16	5位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]0UT_101[8]	
	W01D	0 16	5位十进制数	<pre>KV-5500[0]. XINJE EtherNet/IP[1]OUT_101[9]</pre>	
	W01E	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]00T_101[10]	
	W01F	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]00T_101[11]	
	W020	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]0UT_101[12]	
	W021	0 16	6位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]00T_101[13]	
	#022	0 16	6位十进制数	<pre>KV-5500[0]. XINJE EtherNet/IP[1]OUT_101[14]</pre>	
	W023	0 16	6 位十进制数	KV-5500[0]. XINJE EtherNet/IP[1]0UT_101[15]	~
	la la constante de la constante				

Application 7: Use Xinje PLC XDH-60T-E as the scanner and Keyence PLC KV-5500 as the adapter for implicit communication.

PLC1: XDH-60T-E (IP 192.168.6.6) as the scanner and PLC2: KV-5500 (IP 192.168.6.10) as the adapter, implicit communication between two PLCs can be achieved. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with that of the adapter.

Step 1: In the Keyence KV STUDIO programming software, after connecting to the PLC to be communicated, double-click KV-5500 under the unit configuration to enter the unit editor - edit mode. Double click the CUP unit to configure its IP address, ensuring that it is in the same network segment as the scanner.

	单元编辑器 - 编辑模式			o x		
EtherNet/IP R30000 IM10000	文件(F) 編輯(E) 转换(P) 视图(V) 洗项(O) 窗口(W) 帮助(H)				9	10
】切换单元配置 7元件注释	8 - 7 × 8 6 × 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8	0				
		单元		ф.		
ru 光統反定 [序: EIP_test	党: SSmm KV-55500 and Unit	选择单元(1) 设定单元(2)				
每次扫描执行型模块	高: 90mm 沃: S0mm R000	TE T= T 🕞 🖏 🛋 👫 🖏				
初始化模块	/得耗电流: 320mA015 ■ ● 250	□基本		^		
后备模块	-507	首 패 编号	IM10000			
固定周期模块	R30000	占用 Ⅲ 数	230			
工程度利益	-33910	首继电器编号(按通道设定)	R30000			
「日中皇本」 自保持刑法		占用继电器数	640			
元件初始值	2	通信速度	100/10Mbps自动(大)			
定文件寄存器		19 地址设定方法	固定 IP 地址(水)			
		IP 地址	192.168.6.10			
		子阿掩码	255.255.255.0			
		默认网关	0.0.0			
		DNS 服务器	0.0.0			
		摘收超时[s]	10			
		Keep Alive[s]	600	~		
		1項レーク				
	消息			A		
	处理 行 编号 代码 消息					
	((())))))))	1 <				
		u -				
		10000	The second second			

Step 2: In the programming mode of the unit editor, find the Ethernet/IP settings, click the function key on the right side of the Ethernet/IP settings, and enter the Ethernet/IP settings configuration interface.



Step 3: Double click KV-5500 to enter the label setting interface, click add to add label connection, configure the data size corresponding to the added label, click OK to complete the corresponding information configuration, and finally click download to download the configuration information to the PLC controller.

	92.168.6.10	标体设备					EtherNet/IP设备	
		TRADERAE		~			设备列表(1) 设备设定(2)	设备查找(3)
		林瓷列表(L) No	22 工術 10	7			雅 钟 1 2 W 19	
		1 test 1					□ 扫描器设定	
		2 test_2	10	10			IP地址	192.168.6.10
		4 test_4	10	1			単元注释	124-5500
	1						/ 四百称 供应商名称	Keyence Corporatio
	- 3	追加(A) 删除()	D				版本	1.1
		标签设定					标签设定	(段定)
		标签名(I)	test_1				日 按脑室放用 传感器设定备份设定	〈過空〉
		The delayer (r)					传感器设定批量传输设定	(设定)
	3	校元件分配区域(II) 区域1 100	<u>会社 10 本字</u> 大小字) 10 10 10 10 10 10 10 10 10 10					
							适配器设定	
n 518 4								

Step 4: In the Xinje XDPPRO programming software scanner, add the Omron KV-5500 slave device and perform relevant configuration operations on the adapter's IP address and compatibility check:

EtherNet/IP Scanner Config		×
Master Config EtherNet/IP Scanner	Routine Connection IOMapping Connection Status Address config IP Address: 192 168 6 10	
Slave Config —StationId0:KV-5500	Compatible check Vendor ID: 1723 V Device Type: 12 V Product Code: 14 V Major Revisions: 1 V Minor Revisions: 1	
Slave Number: 1 Connection Number	ber: 2/256 Theory throughput: 0 PPS Actual throughput: 0 PPS Import Export Upload DownLoad 0	k Cancel

Step 5: Add two types of connections: Input Only(ID Type) and Input Only(tag Type).
Connection 1: label name test_1, data length 10, mapping first address D0.
Connection 2: instance ID (IN_100), data length 10, mapping first address D20.
Connection 3: label name test_3, data length 10, mapping first address D40.
Connection 4: instance ID (IN_101), data length 10, mapping first address D60.
After completing the configuration, click download to download the configuration to the PLC. The variable

	<u>ر</u> 1	1	· /1	C 11	•	C
tyneg	configured	are chown	in the	tollow	ino	figure
types	conniguicu		III uic	10110 W	mg	inguic.
21	0				ω	0

1	EtherNet/IP	Scanner	Conf

Master Config StherNet/IP Scanner	No Connectio	Input on Connection	DataSize	IN Address	OVT Connection	DataSize	OUT Addr	ess C	Connection
	Transforlin	Point (Ter tert 1	10	00	Point	12_2	_	0	
Slave Config	1 Inputonly	(TR TN 100	10	120		<u> </u>	_	1	1
-StationId0:KV-5500	2 Tapat0alu	(Tog tost 3	10	D20				2	1 6
	2 Inputonly	(TR TN 102	10	040	9		0	2	
		10 or or							
	Connection Name Time out(T)	InputOnly(ID Type) RPI*16) / (IN:1600ms	✓ OVT:1600ms)	Configure Instance	1	~		
	Connection Name Time out(T) IN(Input from the	InputOnly(ID Type) RPI*16 adapter)) < (IN:1600ms	∨ 00T:1600ms)	Configure Instance -OUT(Output to the s	1 adapter)	~		
	Connection Name Time out(T) IN(Input from the Connection Type	InputOnly(ID Type) RFI*16 adapter) Point to point) / (IN:1600ms	✓ OUT:1600ms)	Configure Instance -OUT(Output to the a Connection Type	1 adapter) Foint to po	~		
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point	InputOnly(ID Type) RFI*16 adapter) Point to point IN_101) (IN:1600ms	✓ OUT:1600ms) ✓	Configure Instance -OVT(Output to the a Connection Type Connection Point	1 adapter) Point to po OUT_254	↓ vint) 	
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size	InputOnly(ID Type) RFI*16 adapter) Point to point IN_101 10) (IN:1600ms (1-724Wor	<pre>> OUT:1600ms) > d)</pre>	Configure Instance -OUT(Output to the a Connection Type Connection Point Data Size	1 adapter) Point to po OVT_254	vint (↓ ↓ (1-1Word)	
	Connection Name Time out(T) -IN(Input from the Connection Type Connection Point Data Size MapMaddres	InputOnly(ID Type) RFI*16 adapter) Foint to point IN_101 10 D60) (IN: 1600ms (1-724Wor	V 0UT: 1600ms) V V V V V V	Configure Instance -OUT(Output to the s Connection Type Connection Point Data Size MapMaddres	1 Adapter) Foint to po OUT_254	~ oint ((1-1Word)	
	Connection Name Time out(T) IN(Input from the Connection Type Connection Point Data Size MapMaddres Send trigger	InputOnly(ID Type) RFI*16 adapter) Foint to point IN_101 10 D60 Cycle) (IN: 1600ms (1-724Wor	UT:1600ms)	Configure Instance -OUT (Output to the a Connection Type Connection Point Data Size MapMaddres RPI (communication cvole)	1 adapter) Point to po OVT_254	>	(1-65535ms)	

Step 6: Click on IO mapping or connection status to operate and monitor corresponding data, and verify whether communication is normal.

站配置	常规 连接	IO映射 道	接状态				
herNet/IP Scanner	标签	進通	映射地址	到1且			
	Input Only.						
	test_1						
	test_1[0] InPut	DO	1			
站配置	test_1[1] InPut	D1	2			
StationId0:KV-5500	test_1[2] InPut	D2	3			
	🕀 test_1[3] InPut	D3	100			
	test_1[4] InPut	D4	14			
	test_1[5] InPut	D5	10071			
	test_1[6] InPut	D6	15			
	test_1[7] InPut	D7	16			
	test_1[8] InPut	D8	17			
	test_1[9] InPut	D9	18			
	test_3						
	test_3[0] InPut	D40	7			
	🗄 test_3[1] InPut	D41	8			
	🗄 test_3[2] InPut	D42	9			
	test_3[3] InPut	D43	300			
	test_3[4] InPut	D44	0			
	test_3[5] InPut	D45	0			
	test_3[6] InPut	D46	0			
	🕂 test_3[7] InPut	D47	0			
	test_3[3] InPut	D48	0			
	test_3[9] InPut	D49	0			
	Input Only.						
	IN_100						
	H IN 100[0] InPut	D20	4			
	⊞ IN_100[] InPut	D21	5			
	IN_100[2] InPut	D22	6			
	H IN 100[3] InPut	D23	200			
	H IN 100[4] InPut	D24	0			

Application 8: Implicit communication between Xinje PLC XDH-60T4-E as adapter and Xinje PLC XSDH-60A32-E as scanner.

PLC1: XDH-60T-E (IP 192.168.6.6) as the adapter and PLC2: XSDH-60A32-E (IP 192.168.6.200) as the scanner to achieve implicit communication between two PLCs. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with the data size of the adapter.

Step 1: Add three connections in adapter XDH-60T4-E in the direction of Adapter->Scanner(T->O).

Connection 1: instance ID100, label name test1, mapping first address D0, data length D10.

Connection 2: label name test11, mapping first address D200, data length 5.

Connection 3: label name test22, mapping first address D250, data length 20.

Add one connection in the direction of Scanner->Adapter(O->T). Instance ID101, label name test2, mapping first address D20, data length 10.

Master Config EtherNet/IP Adapter	Adapter->Sc	canner (T->0)				Scanner->A	dapter(0->T)			
	Number	Tag name	Living example	Map first address	Enter data length	Mumber	Tag name	Living example	Map first address	Enter data length
	0	test1	100	DO	10	0	test2	101	D20	10
		test11		D200	5					
	2	test22	-	D250	20					
					-	5				
	Label sett: Label	ing	test22	Add	Delete	Label set	ting 1 name	test2	Add	Delete
	Label sett: Label Livin	ing L name ng example ID	test22	Add	Delete	Label set Labe Livi	ting 1 name ng example ID	test2	bb A	Delete
	Label sett: Label Livin Map f	ing name 1g example ID first address	test22	Add	Delete	Label set Labe Livi Map	ting 1 name ng example ID first address	test2 101 D20	Add U	Delete

Step 2: In the XS Studio programming software, click on the tool to import the EDS file as an adapter.



Step 3: Click on Network Configuration, add an EthernetIP master station in the network configuration, create an Ethernet-IP_Scanner, double-click Ethernrt to select the network card to use.









Step 5: Double click on XINJE EtherNetIP to enter the corresponding configuration interface, click on General to configure the IP address of the adapter.

Devices 👻 👎 🗙	Network configuration	Ethernet THE EtherNet_IP_Scanner	XINJE_EtherNetIP X
■	General	Address Settings	
😑 🎥 Network configuration	Connections	IP address 192 . 168 . 6 . 6	EtherNet/IP
Ell PLC Logic Application	Assemblies		
Library Manager	User-Defined Parameters	Electronic Keying	
	Log	Compatibility check	
EthIPScannerIOTask	EtherNet/IP I/O Mapping	Vendor ID 1723 Check match	
😑 😻 ENIPScannerServiceTask 🕀 EtherNet_IP_Scanner.ServiceCyde	EtherNet/IP IEC Objects	Device type 12 Check match	
⊟ 😂 MainTask ⊕⊞ PLC_PRG	Status	Major revision 1 Check match	
If Ethernet (Ethernet) EtherNet IP Scanner (EtherNet/IP Scanner)	Information	Minor revision 1 Check match	
XINJE_EtherNetIP (XINJE EtherNetIP)			
Local High Speed IO			

Step 6: Click on the connection to add a label connection that matches the adapter data size. The first connection is to establish an exclusive owner transmission type, with a point-to-point connection type and a data size of 20 bytes. The second connection is to create a label name test11, with a transmission type of input only and a connection type of point-to-point and a data size of 10 bytes. The third connection is to create a label name test22, with a transmission type of input only and a connection type of point-to-point and a data size of 40 bytes.

	General Connections Assemblies User-Defined Parameters Log	Connection Name	RPI (ms) 100 10 10	0>T Size (Bytes) 20 0	T>O Size (Bytes) 20 10	Proxy Config Size (Bytes)	Target Conf		
	Connections Assemblies User-Defined Parameters Log	Connection Name 1. ExclusiveOwner(ID Type) 2. Generic connection 3. Generic connection Edit Connection	RPI (ms) 100 10 10	0>T Size (Bytes) 20 0	T>O Size (Bytes) 20 10	Proxy Config Size (Bytes)	Target Conf		
	Connections Assemblies User-Defined Parameters Log	1. ExclusiveOwner(ID Type) 2. Generic connection 3. Generic connection Edit Connection	100 10 10	20 0	20				
PUC Logic Poptication Dirary Manager PIC, PAG (PRG) EVERS (PRG) State Configuration State Configuration State Configuration State Configuration	Assemblies User-Defined Parameters Log	2. Generic connection 3. Generic connection Edit Connection	10 10	0	10				
Constant Application Divery Manager Divery Manager Divery Manager Diverse (PRG) Second Se	User-Defined Parameters	Edit Connection	10	- 14.3					
fff Ubrary Manager ■ PLC_PRG (PRG) ■ 55 Task Configuration ■ 55 ENUPScanner10Task	User-Defined Parameters	Edit Connection		20	40				
Kit Task Configuration Sector ENIPScannerIOTask	Log								
ENIPScannerIOTask		Connection Path Settings					ОК		
면 EtherNet_IP_Scanner.IOCycle	EtherNet/IP I/O Mapping	O Automatically generated path							
Section 2 S	EtherNet/IP IEC Objects	Class ID: 16#4	Instance	e ID: 16# 0 Att	ribute ID: 16#3				
= S MainTask □ ⊕ PLC_PRG	Status	Class ID: 16#4	oly (0>T) Instance	e ID: 16# 0 Att	ribute ID: 16# 3				
If Ethernet (Ethernet) If EtherNet_IP_Scanner (EtherNet/IP Scanner)	Information	Producing assemb	y (T->0)						
XINJE_EtherNetIP (XINJE EtherNetIP)		Class ID: 16#4	Instance	e ID: 16# 0 Att	ribute ID: 16#3				
SoftMotion General Axis Pool		O User-defined path							
Local High Speed 10		Path defined by symbol	lic name						
		General Parameters							
		Symbolic name test 11							
		Trigger type Cydic		∼ R	PI (ms) 10	÷			
		Transport type Input	only	~ Ti	meout multiplier 4	~			
		Scanner to Target (Output)		Tar	get to Scanner (Input)				
		0>T size (bytes)	0	T-	>0 size (bytes) 10				
		Proxy config size (bytes)	0						
		Target config size (bytes)	0						
		Connection type Mul	icast	~ 0	onnection type Poin	t to Point 🗸 🗸			
		Connection Priority Low		~ 0	onnection priority Low	~			
		Fixed/Variable Fixe	d	~ Fi	xed/Variable Fixe	d v			
¢	Configure device information output	Transfer format 32-	oit run/idle	~ Ti	ansfer format 32-t	it run/idle 🗸 🗸			
Ĩ	Device Information List	Inhibit time (ms) 0	÷	In	hibit time (ms)	×			
	Machine alat Davies same	Heartbeat multiplier	0						

Note:

(1) When creating a new connection using the "instance ID" and "exclusive owner" connection type, the configuration information is roughly as follows:

Generic connection (f	reely configurable)	0				OK
Predefined connectio	on (EDS file)					Cance
 Automatically gen Automatically gen Configuration Class ID: 6 Consuming a Class ID: 6 Producing a Class ID: 6 User-defined path 	ps nerated path n assembly #4 Instance ssembly (0>T) #4 Instance sembly (T>O) #4 Instance	ID: 10 #0 ID: 10 #0 ID: 10 #0	Attribute ID: 15#3 Attribute ID: 15#3 Attribute ID: 15#3 S			
 Path defined by s eneral Parameters 	symbolic name					
Connection Path	20 04 24 00 2C 00 2C 0	0	RPI (ms)	10		
Connection Path [Trigger type [Transport type]	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner	0 ~ 6	RPI (ms) Timeout multiplier	10		
Connection Path	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner put)	0	RPI (ms) Timeout multiplier Target to Scanner (Inp	10 🔹 4 🗸		
Connection Path [Trigger type Transport type anner to Target (Outp 0>T size (bytes) Proxy config size (by Target config size (by	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner put) 0 tes) 0 ytes) 0		RPI (ms) Timeout multiplier Target to Scanner (Inp T>0 size (bytes)	10 🚖 4 ~		
Connection Path [Trigger type [Transport type] anner to Target (Outp 0>T size (bytes) Proxy config size (by Target config size (by Connection type	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner put) 0 tes) 0 ytes) 0 Multicast		RPI (ms) Timeout multiplier Target to Scanner (Inp T>0 size (bytes)	10 🔹 4 × nut) 0	7	
Connection Path [Trigger type [Transport type] anner to Target (Outp 0>T size (bytes) Proxy config size (by Target config size (by Connection type Connection Priority	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner put) (0 tes) 0 ytes) 0 Multicast Low		RPI (ms) Timeout multiplier Target to Scanner (Inp T>0 size (bytes) Connection type Connection priority	10 ÷ 4 × nut) 0 Multicast Low		
Connection Path [Trigger type [Transport type] anner to Target (Outp 0>T size (bytes) Proxy config size (by Target config size (by Connection type Connection Priority Fixed/Variable	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner put) 0 tes) 0 ytes) 0 Multicast Low Fixed		RPI (ms) Timeout multiplier Target to Scanner (Inp T>0 size (bytes) Connection type Connection priority Fixed/Variable	10 🔹 4 🗸 uut) 0 Multicast Low Fixed		
Connection Path [Trigger type [Transport type] anner to Target (Outp 0>T size (bytes) Proxy config size (by Target config size (by Connection type Connection Priority Fixed/Variable Transfer format	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner put) 0 tes) 0 ytes) 0 Multicast Low Fixed 32-bit run <i>f</i> idle		RPI (ms) Timeout multiplier Target to Scanner (Inp T>O size (bytes) Connection type Connection priority Fixed/Variable Transfer format	10 🔹 4 🗸 nut) 0 Multicast Low Fixed 32-bit run/idle		
Connection Path [Trigger type [Transport type] Transport type] Transport type] Connection Target (Outp Connection g size (by Connection type] Connection type] Connection Priority Fixed/Variable] Transfer format] Inhibit time (ms)	20 04 24 00 2C 00 2C 0 Cyclic Exclusive owner put) 0 tes) 0 ytes) 0 Multicast Low Fixed 32-bit run/idle 0		RPI (ms) Timeout multiplier Target to Scanner (Inp T>0 size (bytes) Connection type Connection priority Fixed/Variable Transfer format Inhibit time (ms)	10 ÷ 4 × ut) 0 Multicast Low Fixed 32-bit run/idle 0 ¢		

1	Select automatic path generation to enable instance ID configuration
2	Check the corresponding boxes for configuration assembly, consuming assembly, and producing
	assembly
3	Class ID is default value 4
4	Instance ID: The instance ID for configuration assembly is set to 1 by default. When creating a
	"exclusive owner" connection, the instance ID for consuming assembly (O ->T) should be
	consistent with the instance ID configured by the adapter. If "input only" connection is created, data
	in the direction of configuration (O ->T) will not be configured. The instance ID for consuming
	assembly (O ->T) must be filled in as FE, and the instance ID for producing assembly (T ->O)
	should be consistent with the instance ID configured in adapter.
5	Attribute ID is default value 3
6	Transport type select as actual using condition
7	The size of data to be transmitted for corresponding configuration
8	Configure the corresponding connection types as needed

(2) When creating a new connection using the "tag" and "exclusive owner" connection type, the configuration information is roughly as follows:

The connection path needs to be generated based on the tag name configured by the adapter, and the connection path in the T ->O direction needs to be placed before the connection path in the O ->T direction;

Quick generation of connection path: Click on the path defined by symbolic name, fill in the required label name, and then click on user-defined path to obtain a connection path code.

Connection Path Setting	S					OK
O Automatically gen	nerated path					UN
Configuration	assembly					Cancel
Class ID: 16;	#4 II	nstance ID; 16# 0	Attribute ID: 16# 3			
Consuming as	ssembly (0->	T)				
Class ID: 16	#4 II	nstance ID: 16# 0	Attribute ID: 16# 3			
Producing ass	sembly (T>0)				
Class ID: 16;	#4 I	nstance ID: 16# 0	Attribute ID: 16# 3			
() User-defined path	n T	>0	0>T			
O Path defined by s	ymbolic name	4	-			
			/			
Trigger type	9106746573	~	RPI (ms)	10		
Trigger type Transport type Gcanner to Target (Outp	91 06 74 65 73 Cyclic Exclusive owne put)		RPI (ms) Timeout multiplier Target to Scanner (Inj	10 🚖 4 ~		
Trigger type Transport type Geanner to Target (Outp O>T size (bytes) Proxy config size (byt Target config size (byt	Cyclic Exclusive owne out) [10 tes) 0 ytes) 0		RPI (ms) Timeout multiplier Target to Scanner (In T>0 size (bytes)	10 🚖 4 ~ put)		
Trigger type Transport type canner to Target (Outp 0>T size (bytes) Proxy config size (byt Target config size (byt Connection type	Cyclic Exclusive owne out) (10 (tes) 0 (vtes) 0 Point to Point		RPI (ms) Timeout multiplier Target to Scanner (In T>O size (bytes) Connection type	10 🚖 4 🗸 put) 10 Point to Point		
Trigger type Transport type canner to Target (Outp 0>T size (bytes) Proxy config size (byt Target config size (byt Connection Type Connection Priority	Cyclic Exclusive owne put) [10 [tes) 0 [vites) 0 [Point to Point Low		RPI (ms) Timeout multiplier Target to Scanner (Inj T>O size (bytes) Connection type Connection priority	10 🔹 4 V put) 10 Point to Point Low	×	
Trigger type Transport type teanner to Target (Outp 0>T size (bytes) Proxy config size (byt Target config size (byt Connection type Connection Priority Fixed/Variable	Cyclic Exclusive owne out) (tes) 0 Point to Point Low Fixed		RPI (ms) Timeout multiplier Target to Scanner (In T>O size (bytes) Connection type Connection priority Fixed/Variable	10 🚖 4 🗸 Pout) 10 Point to Point Low Fixed		
Trigger type Transport type Connection Target (Outp O>T size (bytes) Proxy config size (byt Target config size (byt Connection type Connection Priority Fixed/Variable Transfer format	SI UE /4 65 /3 Cyclic Exclusive owner out) 10 tes) 0 vtes) 0 Point to Point Low Fixed 32-bit run/idle		RPI (ms) Timeout multiplier Target to Scanner (Inj T>O size (bytes) Connection type Connection priority Fixed/Variable Transfer format	10 🔹 4 🗸 put) 10 Point to Point Low Fixed 32-bit run/idle	> > >	
Trigger type Transport type Connection Target (Outp O>T size (bytes) Proxy config size (byt Target config size (byt Connection type Connection type Connection Priority Fixed/Variable Transfer format Inhibit time (ms)	Cyclic Exclusive owne out) [10 tes) 0 [vites) 0 [vites) 0 [Point to Point Low Fixed [32-bit run/idle]		RPI (ms) Timeout multiplier Target to Scanner (In T>O size (bytes) Connection type Connection priority Fixed/Variable Transfer format Inhibit time (ms)	10 4 > 4 > 10 10 Point to Point Low Fixed 32-bit run/idle 0	× × ×	

Step 7: Click on the assemblies to configure the data types in the specified connection input/output components as needed.

Seneral	Connections									
Connections	Connection Name	0	>T Size (Bytes)	T>O Size (Bytes)	Proxy Config Size (Bytes)	Target Co	nfig Size (Bytes)			
	1. ExclusiveOwner()	ID Type) 20)	20						
Assemblies	2. Generic connection	on O		10						
	3 Ganaric connactio	n 0		10						
User-Defined Parameters	0	10.7					- Input Assembly "Gather	(Ω<Τ)"		
22	Output Assembly 'Gathe	er" (0>1)					JL Add N Delete	(1 ×0)	Maria Davia	
LUY	r Add X Delete	I Move Up	Move Down			12	-P Aug A Delete	a wove op	wiese bown	
EtherNet/IP I/O Mapping	Name	Data Type	Bit Length	Help String			Name	Data Type	Bit Length	Help String
	ProduceDataSize	UINT	16	Data Size			ProduceDataSize	UINT	16	Data Size
EtherNet/IP IEC Objects	Gather_Param1	BYTE	8				Gather_Param1	BYTE	8	
	Gather_Param2	BYTE	8				Gather_Param2	BYTE	8	
Status	Gather_Param3	BYTE	8				Gather_Param3	BYTE	8	
	Gather_Param4	BYTE	8				Gather_Param4	BYTE	8	
Information	Gather_Param5	BYTE	8				Gather_Param5	BYTE	8	
	Gather_Param6	BYTE	8				Gather_Param6	BYTE	8	
	Gather_Param7	BYTE	8				Gather_Param7	BYTE	8	
	Gather_Param8	BYTE	8				Gather_Param8	BYTE	8	
	Gather_Param9	BYTE	8				Gather_Param9	BYTE	8	
	Gather_Param10	BYTE	8				Gather_Param 10	BYTE	8	
	Gather_Param11	BYTE	8				Gather_Param11	BYTE	8	
	Gather_Param12	BYTE	8				Gather_Param12	BYTE	8	
	Gather_Param13	BYTE	8				Gather_Param13	BYTE	8	
	Gather_Param14	BYTE	8				Gather_Param14	BYTE	8	
	Gather_Param15	BYTE	8				Gather_Param15	BYTE	8	
	Gather_Param16	BYTE	8				Gather_Param16	BYTE	8	
	Gather_Param17	BYTE	8				Gather_Param17	BYTE	8	
	Gather_Param 18	BYTE	8				Gather_Param 18	BYTE	8	

Step 8: Check the current communication status of the corresponding left tree, click on IO mapping to monitor whether data transmission is normal.

10-15 100 (回) Device 「左摘弁引(XSDH-60A32)	佣	查找		过滤 显示所有	r.		•	寺 给10	重道添加	IFB, → 转到实例
副 PLC 逻辑 - O Application [运行]	讅	变量 + · 📴 ExclusiveOwner(ID Type	時期	通道	地址	美型	当前值	预备值	单元	描述 ExclusiveOwner
● 保管理器 ● 译 化 PRG (PRG) ● 译 代表配置 ● ③ 代表配置 ● ③ 代表配置 ● ③ 行为配置 ● ③ 行为配置 ● ③ 行为配置 ● ③ 行为配置 ● ③ 行为配置 ● ③ 行为化于 少 Samer.ServiceCyde ■ ④ Entherlet JP_Samer.ServiceCyde ■ ④ Maritak	け	□ □ 普通 连接		Input_Param0	%1820	BYTE	43			Data Size
	志	8-19 8-19 8-10		Input_Param1 Input_Param2	%IB21 %IB22 %IB23	BYTE BYTE	2 154 2			
	herNet/IPI/O映射	18-X9		Input_Param4	%IB24	BYTE	9			
	herNet/IPIEC只 计			Input_Param5 Input_Param6	%1825	BYTE	120			
Ethernet (Ethernet)	200 100	8.79		Input_Param8	%IB28 %IB29	BYTE	0			
G S XINJE_EtherNetIP (XINJE EtherNetIP)		□ □ 普通连接		Input Param0	%JW15	INT	1			
3 本地の		* *9		Input_Param1	%IW16	INT	2			
		10 Mg		Input_Param3	%IW18	INT	4			
		8-19		Input_Param5	%IW20	INT	6			
		6.79		Input_Param6 Input_Param7	%IW21 %IW22	INT	8			
		⊕_ 19 ⊕_ 1 9		Input_Param8 Input_Param9	%IW23 %IW24	INT	9 10			
		* *9 * *9		Input_Param10 Input_Param11	%IW25 %IW26	INT INT	11 12			
		Evclusive@wner		臣位帝	8.1	直接新空田	. Dent	時以一直力	台谷海口	2(((冬中))

Application 9: Using Xinje PLC XDH-60T4-E as a scanner and Xinje PLC XSDH-60A32-E as an adapter for implicit communication.

PLC1: XDH-60T-E (IP 192.168.6.6) as the scanner and PLC2: XSDH-60A32-E (IP 192.168.6.200) as the adapter to achieve implicit communication between two PLCs. During the connection creation process, it is important to ensure that the data size of the connection point used is consistent with the data size of the adapter.

Step 1: Click on Network Configuration. Add an EthernetIP slave to the network configuration and create an Ethernet_IP_Adapter. Double click on Ethernet to select the network card to use.



Step 3: Right click on Ethernet IP Adapter to add the required Ethernet/IP module.

	Add Device				>	
evices ▼ Untitled2 ■ Device (XSDH-60A32) ■ Network configuration	名称 EtherNet_IP_Module 动作 ④ Append device 〇 Insert device 〇) <u>P</u> lug device O	<u>U</u> pdate device			
PO Frame PLC Logic	String for a full text search	Vendor	<all vendors=""></all>		~	
Application Library Manager Library Manager Library Manager Library Manager Library Manager Start Configuration Start Configura	Name - M Fieldbuses - EtherNet/IP - EtherNet/IP Module	Vend <mark>o</mark> r		Version	ion Description	
 ⇒ ENIPAdapterIOIask → EtherNet_IP_Adapter.IOCyde ⇒ ENIPAdapterServiceTask → EtherNet_IP_Adapter.ServiceCyc → → EtherNet_IP_Adapter.ServiceCyc → → PLC_PRG → Ethernet (Ethernet) 	EtherNet/IP Module	35 - Smart Soft	vare Solutions GmbH	3.5.14.0	A device that wor	
SoftMotion General Axis Pool Local High Speed IO						

Step 4: Double click on the corresponding Ethernet/IP module to be added, and select Word Output Module and Word Input Module on the usual interface.



Step 5: Double click Ethernet IP Adapter to export the configured information in the form of an EDS file.



Step 6: On the Xinje XDPPro programming software scanner, load the exported EDS file, add the corresponding slave device after loading, and perform relevant configuration operations on the adapter's IP address and compatibility check:

EtherNet/IP Scanner Config			×
Master Config EtherNet/IP Scanner	Routine Connection IOMapping Connection Status Address config IP Address: 192 168 6 200		
Slave Config —StationIdO:EtherNet/IP Adapter	Compatible check Vendor ID: 1723		
	Device Type: 12 🗹 Product Code: 14 🗹 Major Revisions: 1 🗸		
	Mînor Revisions: 1		
Slave Number: 1 Connection Number: (0/256 Theory throughput: 0 PPS Actual throughput 0 PPS Import Export Upload DownLoad	0k	Cancel

Step 7: Click on the connection to view the connection type, and the imported connection type can also be modified according to actual needs.

aster Config therNet/IP Scanner	No Conne	ction	Input Connection Point	DataSize	IN Address	OUT Connection Point	DataSize	OVT Address	Connectio ID
26 V.C. MARYI	0 Exclus	iveOwne	IN_101		100	OUT_100		HD10	
ave Config									
	Connection Na	me Excl	usiveOwner (Tag	Type)	~			Add	Delect
	Time out(T)	RPI*	16 🗸	(IN:1600ms	OVT:1600ms)	Configure Instance		~	
	IN(Input from	the adapter	·)			OVT (Output to the a	dapter)		
	Connection Ty	pe Poin	t to point		~	Connection Type	Point to po	int	~
	Connection Po	int IN_1	01		T	Connection Point	OVT_100		
	Data Size	1		(1-724Wor	d)	Data Size	1	(1-724₩	ord)
	MapMaddres	HDO				MapMaddres	HD10		
	Sand trian	r Cycl	e		\sim	RPI(communication cycle)	100	(1-6553)	ōms)
	Sella Li 1886								

Step 8: Click on IO mapping or connection status to operate and monitor corresponding data, and verify whether communication is normal.

EtherNet/IP Scanner Config						×
	Routine Connection	IOMapping	Connection Status			
Master Config EtherNet/IP Scanner	Tag ExclusiveOwner(IN_101	Channel	MapAdress	Value		
Slave Config		InPut	HDO	0		
-StationIdO:EtherNet/IP Adapter		OutPut	HD10	0		
Slave Number: 1 Connection Number:	1/256 Theory throughput: 0 F	PPS Actual thro	ughput 0 PPS	Import E	Export Upload Do	wnLoad Ok Cancel

5-4-2. Explicit communication

Explicit message is a point to point communication method in which the client sends a request to the server and waits for the server to respond; Label communication is a communication method based on label address that reads or writes data by accessing the label address in the device. Explicit messages consist of two parts: the client and the server.

5-4-2-1. Explicit server

The name (case insensitive), label type, data type, length, and mapping address of explicit messages need to be defined in the global variable table of the server in advance. After definition, click download to download the configuration to the PLC and wait for the client to establish a connection with it.

Add Delet	e Move-Up Move-Do	own Imp	ort Export	Searc	h		
Name	Туре	Кеер	Initial va	Con	Network status	Map address	Comment
-tag_1	INT		1000		Public		
-tag_2	INT		100		Public	3	
-tag_3	INT		122		Public		
-tag_4	INT		2		Public		

- (1) Names are not case sensitive, meaning tag_1 is equivalent to TAG_1.
- (2) The mapping address is power off holding registers, please select this to maintain the value.
- (3) The network status please set to public.

5-4-2-2. Explicit client

1. Add device

EtherNet/IP Explicit Setting			×
EtherNet/IP Explicit Setting Master configuration EtherNet/IP (display commun Add Device Slave configuration	Target device configu IP Address: Port: Time out(ms): Reissued Number: Enable control: Connection flag:	iguration – – × 192,168,6,7 44818 500 1 1 M0 2 M100	×
Slave Numbers: 0 Connection Numbers: 0/3000		OK Cancel	Cancel

1	Right click on	EthetNet/IP (Display Communication) in the main station configuration to add devices.
	Configure the	target devices for adding slave stations accordingly.
	IP address	As the IP address of the server PLC; Default 192.168.6.1, starting from 1, the next one defaults
		to the previous address +1
	Port	Default 44818, fixed and cannot be modified.
	Time out	The default setting is 500ms, with a range of 1-65535.
	Reissue	When the triggering method is conditional triggering, if the communication timeout occurs, it
2	number	will be resent with a default number of times of 1, and the allowed input range is 1-15.
2	Enable	By default, it is not enabled. Enabled to set local coil control.
	control	When not enabled: PLC automatically establishes a connection to the target IP after running;
		When enabled: Only bit registers are supported, and a connection to the target IP is only
		established when the coil is normally ON. Close the connection when the conditions are not
		met.
	Connection	Store the result of the successful connection of this device in the corresponding connection flag
	flag	register.
	Import	Import the configured information into the current configuration interface in the form of an
		XML file.
	Export	Export the configured information in the form of an XML file.
	Upload	Upload the configuration information downloaded to the PLC to the current configuration
		interface, and the uploaded configuration information will overwrite the existing configuration
3		information on the current interface.
	Download	Download the configuration information of the current configuration interface to the PLC. The
		downloaded configuration information will overwrite the original configuration information in
		the PLC and take effect in real time with the new configuration information.
	OK	Click OK to save the configuration information for the current page.
	Cancel	Click to discard the configuration information for the current page.

2. Add connection

EtherNet/IP Explicit Setting	lomal 1014				12.			×
Master configuration EtherNet/IP (display communication)	Add Del	Clear Up Down]					
3	No.	Name(Tag)	Data type	Quantity	Trigger mode	Trigger condition	Function code	Mapping address
	0	tag_1	INT	1	Condition trigger	Ml	Read the label	DO
StationId0:192.168.6.7:44818								
Slave Numbers: 1 Connection Numbers: 2/3	00 4				Import	t Ibioad	Download	1K Cancel

1	Normal	Add the specified node to the slave station in this interface to establish a connection.
	IO mapping	View or monitor detailed address information for adding connection mappings.
	Add	Clicking on add will create a new connection.
	Delete	Select the corresponding established connection, click delete to delete the selected
2		connection.
2	Clear	Delete all configuration information on this interface.
	Up	For the selected established connection, click Move Up to move it up by one unit.
	Down	For the selected established connection, click move down to move down one unit.
	No.	Click on add to create a connection. This number will automatically increase by 1
	Name	The name of the connection should be consistent with the label of the explicit server,
		ensuring the correct label name and data type. Note: The number of data corresponding to
		the name must not exceed the data defined by the server, otherwise communication will fail
	Data type	For specific supported data types, please refer to 5-3-3. client and server support variable
		types
	Quantity	The number of label variables corresponding to the read or write operation of this
		connection
3	Trigger mode	Cyclic triggering: Triggering in cycles according to the set triggering conditions;
		Conditional triggering: When the set triggering condition changes state, it triggers the
		explicit client
	Function code	Read tag (0x4c): Read tag service, where the client reads the specified tag data from the
		server;
		Write tag (0x4d): Write tag service, where the client writes the specified tag data to the
		server;
		Mapping address: Maps read label data or cached label data to be written into PLC
		registers.
4	Slave number	Count the number of slave stations connected under the current master station.
	Connection	Count the number of connections established between the master station and all connected
	number	slave stations. The specific specifications for the number of slave stations or supported

	connections	supported	by	Ethernet/IP	communication	can	be	found	in	section	5-3.
	Ethernet/IP c	ommunicat	ion	specifications	S.						

5-4-2-3. Application

Application 1: Two Xinje XDH-60T-E for explicit label communication.

PLC1: XDH-60T-E (IP 192.168.6.6) as the explicit server and PLC2: XDH-60T-E (IP 192.168.6.7) as the explicit client to achieve explicit label communication between two PLCs.



Step 1: Create corresponding variables in the global variable table on the server, and select the network state corresponding to the label as public state. The specific operation configuration is as follows:

File Edit Search View Online Conf	igure Option W	indow Help	3 ▶ - €	1	6	dia 🖸 🖸	🔯 🕰 • 🗯	A.
+++ 몸 -※ 몯 -+ -++ -+++	-++- † + -	≁	0-	— ×	· *	20 ×0 I	• 🖬 • 🖸	- 5 ++ (
Project 4	× PLC1 - Ladde	Global Variable Tab	ole					
PLC1	Add Delete	Move-Up Move-Down	n Impo	ort Export	Search			
Code	Name	Type	Keep	Initial va	Constant	Network status	Map address	Comment
Ladder	⊕-tag_1	INT[2]		-		Public	[D0,D1]	
Address Management	tag_2	INT				Public	D2	
User Data Type Table	-tag_3	INT		- 22		Not public	D3	6
	-tag_4	INT		-		Public	D4	
- SYS_ETHERNET								

Step 2: Add a server device to the client and configure the corresponding IP address and related parameters for the specified communication server:



Step 3: Add a connection to the client. Users can choose the corresponding triggering method and triggering conditions based on the actual application situation, and perform tag reading and writing operations on the server to the mapping address specified by the client.

EtherNet/IP Explicit Setting								
web up to	Normal IOM	lapping			trigger tag_	1 at the rising e	dge of M10	
Master configuration	Add Del	Clear Up Down			1			
EtherNet/IF (display communication)	No.	Name(Tag)	Data type	Quantity	Trigger mode	Trigger condition	Function code	Mapping address
	0	tag_1	INT	2	Condition trigger	M10	Read the label	DO
Slave configuration	1	tag_2	INT	1	Loop trigger(ms)	1000	Write the labe	D10
-StationId0:192.168.6.6:44818					1			
					trigger tag	2 every 1 seco	nd	
Slave Numbers: 1 Connection Numbers:	2/3000				Import Expo	ort UpLoad	DownLoad	K Cancel

Step 4: After adding the configuration, click "Download" to download the configuration information to the PLC. After downloading, monitor the corresponding mapping address and check its communication status.

++ H		++⊢ †	ţ	+		-()-	-(R⊁	-(S)	D		×-	1	* 20 %0	I - I -	С -
1	PLC1 - 梯研	個									•	x	PLC1-自由监控2		ą
	I I							1					监控窗口 • 添加	修改删除 当	自部删除
1 脉山熱粉											1		名称	监控值	类型
山 扩展模块	0	Client											- 🔷 SD1 720	0	INT
BD模块											L		- 🔷 SD1721	0	INT
ED模块	1.34		NT ON	ablad	4								- 🧼 SD1 722	0	INT
4GBOX	1.1	hen M	N, en	MIC	1	NT.	mite	aliont					— 🧼 SD1 723	0	INT
WBOX	2. W	te com	o and	0	JU (ЛN,	write	chem					— 🧇 SD1 724	0	INT
- (0) 赤統18面	2 10	to serv		Z NN -	and		tor T	0 11	to				- 🔷 MD	ON	BIT
. CS Ethernetip	J. W	+ DO T		11, 1	cau	SCI	verL	10, 171	10				- 🔷 DO	100	INT
ElpScanner	Clich	a 170, 1	1										- 🔷 D2	0	INT
EipAdapter													- 🧇 M100	ON	BIT
ElpExplicit													- M 10	ON	BIT
M ModbusTcp	<											>	1		
- EthercatMaster	DICI WIENI			_	_	_			_	_	п	v			
■ 1 Alamin (1) Al	吃坊 探索・	D11		x	v	м	S	SM 1	FT	I.C.	HM	~			
🚡 轴调试	1000		102	~						0					
· 補組配置		+0	*1	+2	+3	*4	+5	+6	*/	*8	+9				
CAM	DO	100	20	0	0	0	0	0	0	0	0	-			
	▶ D10	300	0	0	0	0	0	0	0	0	0				
	D20	0	0	0	0	0	0	0	0	0	0				
派 扩展模块信息	D30	0	0	0	0	0	0	0	0	0	0	1			
₽ BD模块信息 ✓									.0.	0.		~			
	10.开制 2.开	制 16进	南 九	行号	ASC								1		

Image: Server Image: Server Image: Server	• 6 型 * [、] > -
- 採売回・全局支録表 ・採売回・全局支録表 2 ※ 全局支録表 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 4 3 4 4 4 4 4 4 4 5 4 4 4 5 4 4 5 5 4 4 4 5 4 5 4 5 4 4 4 5	a ^ر *
- 梯形圏 全局支録表 一株形圏 学科 得計 初値 常曜 网络状态 除射地址 注释 NUTI21 公开 (D0,D1) NUT 公开 02 NUT 公开 03 NUT 公开 04 Server	-
上修 下修 時入 時出 推奏 黄型 保持 初値 常里 网络状态 映射地址 注降 NTT21 - 公开 [D0,01] NT - 公开 02 NT - 公开 03 NT - 公开 04 Server - - - NT - - - NT - - - - NT - - - - NT - - - - - NT - - - - - NT - - - - - Server - - - - - NT - - - - - - Server - - - - - -	
NT[2] - 公开 [D0,D1] NT - 公开 02 NT - 公开 03 NT - 公开 04 Server	
NT - 公开 02 NT - 公开 03 NT - 公开 04 NT - 公开 04	
NT - 公开 D3 NT - 公开 D4 Server	
NT 口 - 口 公开 D4 Server	
server	
server	
Net to	
	a
TER: DZ • X Y M S SM T ET C HM HS HT HC HSC D SD	ID QD
+0 +1 +2 +3 +4 +5 +6 +7 +	+8 +9
100 20 300 0 0 0 0 0 0	0 0
0 0 0 0 0 0 0 0	0 0
	0 0
	0 0
2进制 16进制 无符号 ASCI	0
11 PLC1-数据出种	

Application 2: Use Xinje XDH-60T-E as the client and Omron NJ501-1500 as the server for explicit tag communication.

Step 1: Define the variables that need to be communicated in the Omron Sysmac Studio programming software, and select the public state of the established label network as public.

	全局变	🛃 🗙 🖬 内置El	herNet/IP端口设置								•	工具箱 🗸 🖓
	组筛选器	▼ (没有组)	T									<检索> ▼ 2 ×
	-	名称	数据类型	▲ 初始値	i 分配到	保持	常量	1	网络公开	注释		
	test_	xx	ARRAY[09] OF INT			2		公开	Ψ.			
	test_	ZZ	ARRAY[09] OF INT			X		公开				
:												
H	靓(工程)	-									- 1 ×	控制器状态 🚽 🗸

Step 2: Add a server device to the client and configure the corresponding IP address and related parameters for the specified communication server:

EtherNet/IP Explicit Setting								×
Master configuration EtherNet/IP (display communication)								
Slave configuration	Target device configu IP Address: Port: Time out(ms): Reissued Number: ☑ Enable control: ☑ Connection flag:	rration – 192 168 250 1 44818 500 1 M0 M100 OK	Cancel					
Slave Numbers: 0 Connection Numbers: 0/30	00		Import	Export	UpLoad	DownLoad	UK	Uancel

Step 3: Add a connection in the client, with the first connection established as a read label method and the second connection established as a write label method.

EtherNet/IP Explicit Setting								
Master configuration	Normal ION	lapping Clear Up Down						
EtherNet/IF (display communication)	No.	Name(Tag) tag_zz	Data type INT	Quantity 10	Trigger mode Loop trigger(ms)	Trigger condition	Function code Read the label	Mapping address D5000
Slave configuration — StationId0:192.168.250.1:44618	j.	tag_xx	INT	10	Loop trigger(ms)	1000	Write the labe	D6000
Slave Numbers: 1 Connection Numbers: 3	 2/3000			[Import Expo	ort UpLoad	DownLoad 0	K Cancel

Step 4: After setting ON M0, when the M100 enable connection flag is set successfully, it indicates that the connection has been established. Click on the IO mapping to check if the communication between read and write data is normal.

秘	通道	类型	映射地址	数值		
test_zz	读标签(4C)	INT[10]	[D500, D509]			
test_zz[0]	读标签(4C)	INT	D500	456		
test_zz[1]	读标签(4C)	INT	D501	0		
test_zz[2]	读标签(4C)	INT	D502	0		
-test_zz[3]	读标签(4C)	INT	D503	0		
test_zz[4]	读标签(4C)	INT	D504	0		
-test_zz[5]	读标签(4C)	INT	D505	0		
test_zz[6]	读标签(4C)	INT	D506	0		
test_zz[7]	读标签(4C)	INT	D507	0		
test zz[8]	读标签(4C)	INT	D508	0		
test zz[9]	读标签(4C)	INT	D509	0		
test xx	写标签(4D)	INT[10]	[D600, D609]			
-test xx[0]	写标签(4D)	INT	D600	123		
test_xx[1]	写标签(4D)	INT	D601	0		
test_xx[2]	写标签(4D)	INT	D602	0		
-test_xx[3]	写标签(4D)	INT	D603	0		
-test_xx[4]	写标签(4D)	INT	D604	0		
-test_xx[5]	写标签(4D)	INT	D605	0		
-test_xx[6]	写标签(4D)	INT	D606	0		
-test_xx[7]	写标签(4D)	INT	D607	0		
-test_xx[8]	写标签(4D)	INT	D608	0		
-test_xx[9]	写标签(4D)	INT	D609	0		

全局变量	✓ 全局委員 × 協 内置EtherNet/P端口设置								
组筛选器 🍸	(没有组) 🔻								
	名称	数据类型 🔺	初始值 分香	(3) 保持	常量	网络公开	注释		
test_xx	ARRAY[09] C	OF INT				(田) マ			
test_zz	ARRAY[09] C	OF INT				2开 🔻			
监视(工程)1									🗸 🖡 🗸
设备名	称	名称	线值 修改	1 注	释	数据类型	分配到	显示格式	
new_Controll	ler_0 test_xx[0]	123				INT		Decimal 🔻	
new_Controll	ler_0 test_zz[0]	456	456			INT		Decimal 🔻	
new_Controll	ler_0								

Appendix

Code	Explanation	Reason and solution
0xFF	No extension	-
0x100	FWD repeated opening	-
0x103	Class triggering invalid	-
0x106	Conflict of ownership	Error reason: The connection point in the O ->T direction of the slave station configuration has already been used. Problem point: The connection point in the O>T direction of the slave station configuration has already been used. Solution: Replace the connection points in the O ->T direction of the main station.
0x107	Connection not found	Error reason: Connection not found. Problem point: It is highly likely that EDS does not match or is missing configuration items. Solution: Determine if the EDS of the slave station is correct.
0x108	Invalid connection type	-
0x109	Invalid connection size	Reason for error: T ->O or O ->T data size setting error, or configuration data length setting error. Problem point: T ->O or O ->T data size setting error, or configuration data length setting error. Solution: Modify the length of the master station data and modify the length of the slave station data
0x110	Device not configured	-
0x111	RPI not support	Error reason: The RPI slave (adapter) setting is not supported. Problem point: It is highly likely that EDS does not match. Solution: Choose the correct EDS.
0x112	RPI value is unacceptable	Error reason: ListenOnly (ID type) RPI configuration error. Problem point: ListenOnly (ID type) RPI requires the same configuration as RPI that depends on InputOnly and ExclusiveOwner. Solution: Learn the correct usage of ListenOnly.
0x113	Connection limit reached	-
0x114	Supplier product code mismatch	Reason for error: Supplier ID or product code check error in compatibility check. Problem point: The EDS selected for the configuration of the slave station equipment does not match that of the master station. Solution: Choose the correct EDS.
0x115	Product type mismatch	Error reason: Device type check error in compatibility check. Problem point: The EDS selected for the configuration of the

Code	Explanation	Reason and solution
		slave station equipment does not match that of the master station.
		Solution: Choose the correct EDS.
		Reason for error: The main revision check error in compatibility
		check.
0x116	Revision mismatch	Problem point: The EDS selected for the configuration of the
		slave station equipment does not match that of the master station.
		Solution: Choose the correct EDS.
		Reason for error: The connection point selection for T ->O or O
		->T is incorrect.
0v117	T 1'1 / '	Problem point: The connection points selected by the master
UXII/		station (scanner) for T ->O or O ->T do not match the
		configuration of the slave station (adapter).
		Solution: Choose the correct connection point.
0x118	Invalid configuration	_
UXIIO	format	
		Error reason: ListenOnly (ID type) connection type configuration
		error.
		Problem point: The ListenOnly (ID type) connection type cannot
0x119	No control connection	be configured as point-to-point, or when configured as multicast,
		it requires an InputOnly and ExclusiveOwner connection point to
		also be configured as multicast.
		Solution: Learn the correct usage of ListenOnly.
0x11A	Reaching target	_
	connection limit	
0x11B	RPI is less than the limit	-
0x11C	Transfer class not	
	supported	
0x11D	Production trigger not	
	supported	
0x11E	Direction not supported	
0x11F	O-T fixed variable	
	invalid	Problem point: These issues basically belong to EDS mismatch
0x120	T-O fixed variable	Solution: Choose the correct EDS
0X120	invalid	
0x121	O-T priority invalid	
0x122	T-O priority invalid	
0x123	O-T connection type	
	invalid	
0x124	T-O connection type	
	invalid	
0x125	O-T redundant owner	
0X125	invalid	
0x126	T-O redundant owner	-

-

Code	Explanation	Reason and solution
	invalid	
0x127	O-T size invalid	Reason for error:(1) The data size setting for O ->T connection is incorrect;(2) Connection point option configuration error.Problem points:(1) The data size setting for O ->T connection is incorrect;(2) The connection point selected by the master station does notmatch the connection point configured by the slave station.Solution:(1) Modify the O ->T data length of the main station forconnection or modify the data length of the consumer connectionpoint configured by the slave station;(2) Modify the connection points of the master station forconnection or modify the connection points of the slave station forslave station.
0x128	T-O size invalid	Error reason: The data size setting for T ->O connection is incorrect. Problem point: The data size setting for T ->O connection is incorrect. Solution: Modify the T ->O data length of the main station's connection or modify the data length of the slave station's configuration producer connection point.
0x129	Invalid configuration path	-
0x12A	Invalid consumption path	Error reason: Connection point option configuration error. Problem point: The connection point selected by the master station does not match the connection point configured by the slave station. Solution: Modify the connection points of the master station for connection or modify the connection points configured by the slave station.
0x12B	Production path is invalid	Error reason: Connection point option configuration error. Problem point: The connection point selected by the master station does not match the connection point configured by the slave station. Solution: Modify the connection points of the master station for connection or modify the connection points configured by the slave station.
0x12C	No configuration	-
0x12D	No consumption symbol	Error reason: Connection label name configuration error. Problem point: The connection label name selected by the master site does not match the connection label name configured by the

-

Code	Explanation	Reason and solution
		slave station.
		Solution: Modify the connection label name of the master station
		for connection or modify the connection label name configured
		by the slave station.
		Error reason: Connection label name option configuration error.
		Problem point: The connection label name selected by the master
	No production symbol	site does not match the connection label name configured by the
0x12E		slave station.
		Solution: Modify the connection label name of the master station
		for connection or modify the connection label name configured
		by the slave station.
0-12E	Invalid application path	
UX12F	combination	-
	Inconsistent	
0x130	consumption data	-
	format	
0121	Inconsistent production	
0x131	data format	-
0122	Not support empty	
0X132	FORWARD OPEN	-
0122	Wrong connection	
0X155	timeout multiplier	-
0v13/	T-O connector size	
07124	mismatch	-
0v135	T-O fixed variable	
0X133	mismatch	-
0v136	T-O connection priority	
0X150	mismatch	
0v137	Transport category	
0.11.57	mismatch	-
		Error reason: ListenOnly (ID type) trigger condition
		configuration error.
0v138	T-O production trigger	Problem point: The trigger conditions for ListenOnly (ID type)
04150	mismatch	need to be configured the same as those for InputOnly and
		ExclusiveOwner.
		Solution: Learn the correct usage of ListenOnly.
0x139	T-O production	_
	inhibition mismatch	
		Error reason: Communication timeout.
0x203	Connection timeout	Problem point: There is no data packet within the timeout
		multiple time in the O ->T or T ->O direction.
		Solution: Usually, set PRI higher.
0x204	Unconnected send	-

-
Code	Explanation	Reason and solution
	timeout	
0x205	parameter error	-
0x206	Message too large	-
0x207	Unconnected packet	
	without reply	-
0x208	Service demand	
	connection	-
0x301	No available buffer	
	memory	
0x302	Bandwidth unavailable	-
0x303	Label filter not	_
	available	
0x304	Real time data not	_
	configured	
0x311	Port unavailable	-
0x312	Link address not	-
	available	
0x315		Reason for error: The default connection point for O ->T is
	Invalid segment type	incorrect.
	value	Problem point: EDS mismatch.
	D. d	Solution: Replace with the correct EDS.
0x316	Path connection	-
	mismatch	
0x317	Invalid network	-
0.210		
0x318	The second necession	-
0x319	The second resource is	-
0x214	Connaction astablished	
UXJIA	Established direct	-
0x31B	connection	-
0x31C	Others	
0x31D	Redundant connection	
	mismatch	-
0x31E	No more consumer	
	resources available	-
0x31F	No target path resources	-
0x320	Supplier specific	-
011020	Unconfigured outside	
0x813	subnet mask	-

-



WUXI XINJE ELECTRIC CO., LTD.

816 Jianshe West Road, Binhu District, Wuxi City, Jiangsu Province 214072, China Tel: 0510-85134136 Fax: (510) 85111290 Email: <u>Fiona.xinje@vip.163.com</u> sales@xinje.com