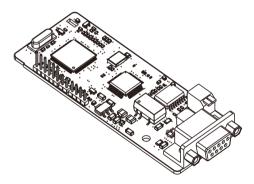


Operation Manual Communication Card



SHENZHEN INVT ELECTRIC CO., LTD.



The expansion card can be installed and operated only by people who have taken part in professional training on electrical operation and safety knowledge, obtained the certification, and been familiar with all steps and requirements for installing, performing commissioning on, operating, and maintaining the device, and are capable of preventing all kinds of emergencies.

Before installing, removing, or operating the communication card, read the safety precautions described in this manual and the variable-frequency drive (VFD) operation manual carefully to ensure safe operation.

For any physical injuries or damage to the device caused due to your neglect of the safety precautions described in this manual and the VFD operation manual, our company shall not be held liable.

- You need to open the housing of the VFD when installing or removing the communication card. Therefore, you must disconnect all power supplies of the VFD and ensure that the voltage inside the VFD is safe. For details, see the description in the VFD operation manual. Severe physical injuries or even death may be caused if you do not follow the instructions.
- Store the communication card in a place that is dustproof and damp-proof without electric shocks or mechanical pressure.
- The communication card is electrostatic sensitive. Take measurements to prevent electrostatic discharge when performing operations involving it.
- Tighten the screws up when installing the communication card. Ensure that it is firmly fixed and properly grounded.

Terminology and abbreviations

CAN	Controller Area Network				
	Communication object, a transmitted unit on a CAN network.				
COB	Communication objects (COBs) carry data and can be transmitted through				
	the whole network. A COB is part of a CAN message frame.				
	Electronic datasheet, an ASCII file for node configuration, required when a				
EDS	CANopen network is configured. An EDS file contains general information				
	about nodes and their dictionary objects (parameters).				
	Network management, one of the CAN application-layer service elements in				
NMT	the CAN reference model. It is used for the initialization, configuration, and				
	fault handling of a CAN network.				
Object	Stores information about all COBs identified by a device				
dictionary	Stores information about all COBs identified by a device.				
PDO	Process data object, a type of COBs, used to transmit process data, such as				
TBO	control command, set values, state values, and actual values.				
PDOn Tx	PDO command transmitted by a slave to the master, where n refers to 1, 2,				
1 DOI 1X	3, 4.				
PDOn Rx	PDO command transmitted by the master and received by a slave, where n				
TDOIN	refers to 1, 2, 3, 4.				
SDO	Service data object, a type of COB, used to transmit non-time key data, such				
300	as parameter values.				
RO	Indicates read-only access.				
RW	Indicates the read and write access.				
SYNC	Indicates synchronous transmission.				
Node-ID	Node ID, that is, address of a communication card.				
	Indicates that a number with this prefix is a hexadecimal value, for example,				
0x	0x10 indicates the decimal value 16.				

Contents

Chapter 1 Product confirmation	1
Chapter 2 PROFIBUS communication card	2
2.1 Overview	2
2.2 Features	2
2.3 Electrical connection	3
2.4 Bus network connection	3
2.5 System configuration	6
2.6 PROFIBUS-DP communication	
2.7 Example of PROFIBUS-DP communication networking	
Chapter 3 CANopen communication card	29
3.1 Overview	29
3.2 Features	29
3.3 Electrical wiring	31
3.4 Terminal wiring	31
3.5 Communication	32
3.6 Process data object (PDO)	
3.7 Monitoring process data through SDO commands	53
3.8 Baud rate and communication address setting	
3.9 Example of communication between CANopen and IVC3	59
3.10 Example of communication between CANopen and AX70	
Chapter 4 BACnet MSTP communication card	74
4.1 Overview	74
4.2 Features	74
4.3 Electrical wiring	75
4.4 Terminal wiring	
4.5 Connecting the BACnet MSTP card to the PLC	77
Chapter 5 CAN master/slave expansion card	103
5.1 CAN master/slave expansion card operation	
5.2 Related function parameters	
Appendix A CANopen object dictionary	
Appendix B Related function codes	116

Chapter 1 Product confirmation

Check the following after receiving a communication expansion card product:

- Whether the communication card is damaged.
- Whether the received communication card is the one you purchase according to the bar code label on the PCB.
- Whether all the following items are contained in the product package:
- One communication card, one tie wrap, one tie, one M3 screw, and one manual.
- If the communication card is damaged, a wrong model is delivered, or some items are missing, contact the supplier in a timely manner.
- Obtain the EDS file of the communication card from INVT. The file is named communication card model.eds.
- Confirm the environmental requirements for application.

Item	Requirement
Operation temperature	-10-+50°C
Storage temperature	-20-+60°C
Relative humidity	5%–95%
Other weather	No condensation, ice, rain, snow, or hail;
conditions	solar radiation < 700 W/m ²
Air pressure	70–106 kPa
Vibration and impact	5.8m/s ² (0.6g) at the sine vibration of 9 Hz to 200 Hz

Table 1-1 Environmental requirements

Chapter 2 PROFIBUS communication card

2.1 Overview

PROFIBUS communication cards are optional accessories for VFDs. They can be used to connect VFDs to PROFIBUS networks. On a PROFIBUS network, VFDs are slave devices. The following functions can be performed by using a PROFIBUS communication card:

- Transmit control commands (such as start, stop, and fault reset) to a VFD.
- Transmit speed or torque reference signals to a VFD.
- Obtain state values and actual values from a VFD.
- Modify parameter values of a VFD.

2.2 Features

- PROFIBUS is an international open fieldbus standard that can implement data exchange between various automation components. It is widely applicable to automation in various industries, such as the manufacturing, process, building, transportation, and power industries. It provides effective solutions for implementing integrated automation and intelligentization of field devices.
- 2. PROFIBUS consists of three mutually compatible components, namely PROFIBUS-Decentralised Peripherals (DP), PROFIBUS-Process Automation (PA), and PROFIBUS-Fieldbus Message Specification (FMS). It adopts the master-slave mode and is generally used for periodic data exchange between VFD devices. PRNV PROFIBUS-DP adapter modules support only the PROFIBUS-DP protocol.
- 3. The transmission media of a PROFIBUS field bus are twisted pairs (complying with the RS-485 standard), paired cables, or optical cables. The maximum length of a fieldbus cable must be within the range of 100 m to 1200 m, and the specific length depends on the selected transmission rate (see the chapter of "Technical Data" in the VFD manual). A maximum of 31 nodes can be connected to one PROFIBUS network segment when no repeater is used. If repeaters are used, a maximum of 127 nodes (including the repeaters and master stations) can be connected.
- 4. In PROFIBUS communication, tokens are transmitted between master stations or by master stations to slave stations. Single-master or multi-master systems are supported. The node to respond to the command of a master is selected by the master station, generally a programmable logic controller (PLC). For cyclic master-slave user data transmission and non-cyclic master-master data transmission, a master can also transmit commands to multiple nodes in broadcast mode. When the broadcast mode is adopted, the nodes do not need to transmit feedback signals to the master. On PROFIBUS networks, nodes cannot communicate with each other.

 The PROFIBUS protocol is described in details in the EN50170 standard. For more information about PROFIBUS, refer to the EN50170 standard.

2.3 Electrical connection

1. Node selection

The node address of a device is unique on a PROFIBUS bus. The node address is set through the function parameter P15.01, and the value ranges from 0 to 127.

2. Fieldbus terminator

Each fieldbus segment is configured with two bus terminators, one on each end, to prevent operation errors. Bus terminators can protect the fieldbus signal against electrical reflections. The dual in-line package (DIP) switch on the printed circuit board (PCB) of a communication card is used to connect to the fieldbus terminator. If the communication card is the last or first module on the network, the bus terminator must be set to ON. When a PROFIBUS D-sub connector with a built-in terminator is used, you must disconnect the communication card from the terminator.

2.4 Bus network connection

1. Bus communication interfaces

The most common PROFIBUS transmission mode is the shielded twisted-pair copper cable transmission, in which shielded twisted-pair copper cables (complying with the RS-485 standard) are used.

The basic characteristics of this transmission technology are described as follows:

- · Network topology: Linear bus with one active fieldbus terminal resistor on each end
- Media: Shielded or unshielded twisted-pair cables, depending on the EMC environmental conditions
- Number of stations: 32 on each network segment (without repeater); a maximum of 127 (with repeaters)
- Plug connection: 9-pin D-type plug. The following figure shows the pins of the connector.



Figure 2-1 Plug of the connector

Connect	or pin	Description
1	-	Unused
2	-	Unused
3	B-Line	Data+ (twisted-pair wire 1)
4	RTS	Transmitting requests
5	GND_BUS	Isolation ground
6	+5V BUS	Isolated 5 V DC power supply
7	-	Unused
8	A-Line	Data- (twisted-pair wire 2)
9	-	Unused
Housing	SHLD	PROFIBUS cable shielding wire

Table 2-1 Connector pins

The +5V and GND_BUS pins are used for bus terminators. Optical transceivers (RS-485) and some other devices may need to obtain external power supplies through these pins.

For some devices, the transmission direction is determined by using the RTS pin. In regular application, only the A-Line, B-Line, and SHLD pins are used.

It is recommended that you use the standard DB9 connectors manufactured by Siemens. If the communication baud rate is required to be higher than 187.5 kbps, strictly follow the wiring standards stipulated by Siemens.

2. Repeaters

A maximum of 32 stations (including the master station) can be connected to each fieldbus segment. If the number of stations to be connected to a fieldbus segment exceeds 32, you need to use repeaters to connect the fieldbus segments. Generally, the number of repeaters connected in series cannot exceed 3.

Note: No station address is provided for repeaters, but they are calculated as stations.

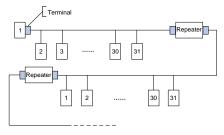


Figure 2-2 Repeaters

3. Transmission rates and maximum transmission distances

The maximum length of a cable depends on the transmission rate. Table 2-2 describes the transmission rates and corresponding transmission distances.

Table 2-2 Transmission rates and corresponding transmission distances

Transmission rate (kbps)	A-type wire (m)	B-type wire (m)
9.6	1200	1200
19.2	1200	1200
93.75	1200	1200
187.5	1000	600
500	400	200
1500	200	

Table 2-3 Transmission wire parameters

Parameter	A-type wire	B-type wire
Impedance (Ω)	135–165	100–130
Capacitance of a unit length (pF/m)	< 30	< 60
Circuit resistance (Ω/km)	110	
Wire core diameter (mm)	0.64	> 0.53
Sectional area of wire core (mm ²)	> 0.34	> 0.22

Besides the shielded twisted-pair copper cables, you can also use optical fibers for transmission in a PROFIBUS system. When a PROFIBUS system is applied in an environment with strong electromagnetic interference, you can use optical fiber conductors to increase the high-speed transmission distance. Two types of optical fiber conductors can be used. One is low-cost plastic fiber conductors that can be used when the transmission distance is shorter than 50 m; and the other is glass fiber conductors that can be used when the transmission distance is shorter than 1 km.

4. PROFIBUS bus connection diagram

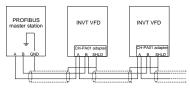


Figure 2-3 PROFIBUS bus connection

Figure 2-3 shows the terminal wiring. The cables are standard PROFIBUS cables, each

consisting of a twisted pair and shielding layer. The shielding layers of PROFIBUS cables are directly grounded on all nodes. You can select a proper grounding mode based on the actual situation on site.

Note:

- When connecting the stations, ensure that the data cables are not twisted together. For systems to be used in environments with strong electromagnetic radiation, you need to use cables with shielding layers. The shielding layers can improve electromagnetic compatibility (EMC).
- 2. If shielding braid or shielding foil is used, connect the two ends of it to the protective ground and cover an area as large as possible to ensure high conductivity. In addition, data cables need to be separated from high-voltage cables.

3. When the data transmission rate is higher than 500 kbit/s, do not use short stub. Use the plugs available in the market. Data input and output cables can be directly connected to those plugs, and the plug of the communication card can be connected or disconnected at any time without interrupting data communication of other stations.

2.5 System configuration

1. System configuration

After the communication card is properly installed, you need to configure the master station and VFD to enable the communication between the master station and communication card.

One device description file named GSD file is required for each PROFIBUS slave station on the PROFIBUS bus. The GSD file is used to describe the characteristics of the PROFIBUS-DP device. The software we provide for users includes information about the GSD file of the VFD. You can obtain the type definition files (GSD files) of various masters from us.

Parameter No.	Parameter name	Setting options		Default setting
0	Module type	Read-only		PROFIBUS-DP
1	Node address	0–99		2
	Baud rate setting		0: 9.6	
2		kbit/s	1: 19.2	C
2			2: 45.45	6
			3: 93.75	

Table 2-4 Communication card	l configuration parameters
------------------------------	----------------------------

Parameter No.	Parameter name	Setting options		Default setting
			4: 187.5	
			5: 500	
			6: 1.5	
			7: 3	
		Mbit/s	8: 6	
			9: 9	
			10: 12	
3	PZD3	0–6	5535	0
4	PZD4	0-65535		0
		0–65535		0
10	PZD12	0–65535		0

2. Module type

This parameter displays the model of the communication card detected by the VFD. You cannot modify the value of this parameter. If the parameter is not defined, communication between the communication card and VFD cannot be established.

3. Node address

On the PROFIBUS network, each device corresponds to one unique node address. The node address is set through P15.01.

4. GSD file

One device description file named GSD file is required for each PROFIBUS slave station on the PROFIBUS bus. The GSD file is used to describe the characteristics of the PROFIBUS-DP device. The GSD file includes all parameters defined for the device, including the supported bard rate, supported information length, input/output data amount, and definitions of diagnosis data.

You can obtain the type definition files (GSD files) of various masters from INVT's official website and copy the GSD files to the corresponding subdirectories on the configuration tool software. For details about the operation and how to configure the PROFIBUS system, see the instructions for the related system configuration software.

2.6 PROFIBUS-DP communication

1. PROFIBUS-DP

PROFIBUS-DP is a distributed input/output (I/O) system. It enables a master to use a large number of peripheral modules and on-site devices. Data transmission is periodic: The master reads information input from a slave and then transmits a feedback signal to

the slave.

2. SAP

The PROFIBUS-DP system uses the services at the data link layer (Layer 2) through service access points (SAPs). Functions of each SAP are clearly defined. For more information about SAPs, see the related PROFIBUS master user manuals, that is, PROFIdrive—PROFIBUS models or EN50170 standards (PROFIBUS protocol) for variable-speed drives.

3. PROFIBUS-DP information frame data structure

The PROFIBUS-DP system allows fast data exchange between the master and VFD devices. For VFD devices, data is always read and written in the master/slave mode. VFDs always function as slave stations, and one address is clearly defined for each slave station. PROFIBUS transmits 16-bit packets periodically. Figure 2-4 shows the structure of the packet.

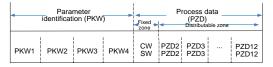


Figure 2-4 PROFIBUS-DP information frame data structure

Parameter zone:

PKW1—Parameter identification

- PKW2—Array index number
- PKW3—Parameter value 1
- PKW4—Parameter value 2

Process data:

CW-Control word (transmitted from the master to a slave. For description, see Table 2-5)

SW—State word (transmitted from a slave to the master. For description, see Table 2-7.)

PZD—Process data (defined by users)

(When the process data is output by the master to a slave, it is a reference value; and when the process data is input by a slave to the master, it is an actual value.)

PZD zone (process data zone): The PZD zone in a communication packet is designed for controlling and monitoring a VFD. The master and slave stations always process the received PZD with the highest priority. The processing of PZD takes priority over that of PKW, and the master and slave stations always transmit the latest valid data on the interfaces.

CWs and SWs

Using CWs is the basic method of the fieldbus system to control VFDs. A CW is transmitted by the fieldbus master station to a VFD device. In this case, the EC-TX-103 communication card functions as a gateway. The VFD device responds to the bit code information of the CW and feeds state information back to the master through an SW.

Reference value: A VFD device may receive control information in multiple channels, including analog and digital input terminals, VFD control panel, and communication modules (such as RS485 and EC-TX-103 communication cards). To enable the control over VFD devices through PROFIBUS, you need to set the communication module as the controller of the VFD device.

Actual value: An actual value is a 16-bit word that includes information about VFD device operation. The monitoring function is defined through VFD parameters. The conversion scale of an integer transmitted as an actual value from the VFD device to the master depends on the set function. For more description, see the related VFD operation manual.

Note: A VFD device always checks the bytes of a CW and reference value.

Task packet (master station -> VFD)

CW: The first word in a PZD task packet is a VFD CW. Table 2-5 describes Goodrive350 series VFD CWs.

Bit	Name	Value	State to be entered/description
	-	1	Forward running
		2	Reverse running
		3	Forward jogging
	Communication-based	4	Reverse jogging
0–7	D-7 Communication-based control command	5	Decelerating to stop
		6	Coasting to stop
		7	Fault reset
		8	Jogging stopped
		9	Emergency stop
8	Enabling writing	1	Enabling writing (mainly through PKW1 to PKW4)
0.40		00	Motor 1
9–10	-10 Motor group setting		Motor 2
11	Control mode switching	1	Enabling the switching between torque control and speed control

Table 2-5 Goodrive350	series	VFD CWs
-----------------------	--------	---------

Bit	Name	Value	State to be entered/description
		0	No switching
10	Resetting power		Enabling the function for resetting power consumption to zero
12	12 consumption to zero	0	Disabling the function for resetting power consumption to zero
13		1	Enabling pre-excitation
13	Pre-excitation	0	Disabling pre-excitation
	DC hashing	1	Enabling DC braking
14	14 DC braking	0	Disabling DC braking
15	Listentia estimation formation	1	Enabling heartbeat
15	15 Heartbeat reference		Disabling heartbeat

Reference value (REF): The second to twelfth words in a PZD task packet are the main settings. The main frequency settings are provided by the main setting signal source. Table 2-6 describes the settings of Goodrive350 series VFD.

Function code	Word	Value range	Default value
P15.02	Received PZD2	0–31 0: Invalid	0
P15.03	Received PZD3	1: Set frequency (0–Fmax, unit: 0.01 Hz) 2: PID reference (0–1000, in which 1000 corresponds to	0
P15.04	Received PZD4	100.0%) 3: PID feedback (0–1000, in which 1000 corresponds to 100.0%)	0
P15.05	Received PZD5	4: Torque setting (-3000-+3000, in which 1000 corresponds to 100.0% of the rated current of the	0
P15.06	Received PZD6	motor) 5: Setting of the upper limit of forward running frequency (0-Fmax, unit: 0.01 Hz)	0
P15.07	Received PZD7	6: Setting of the upper limit of reverse running frequency (0-Fmax, unit: 0.01 Hz)	0
P15.08	Received PZD8	 Upper limit of the electromotive torque (0–3000, in which 1000 corresponds to 100.0% of the rated current of the motor) 	0
P15.09	Received PZD9	8: Upper limit of the brake torque (0–3000, in which 1000 corresponds to 100.0% of the rated current of the	0
P15.10	Received	motor)	0

Function code	Word	Value range	Default value
	PZD10	,	
P15.11	Received PZD11	(corresponding to S8, S7, S6, S5, HDIB, HDIA, S4, S3, S2, and S1 in sequence) 10: Virtual output terminal command, 0x00–0x0F	0
P15.12	Received PZD12		

Response packet (VFD -> master station)

SW: The first word in a PZD response packet is a VFD SW. Table 2-7 describes the VFD SWs.

Bit	Name	Value State to be entered/description	
		1	In forward running
	Running state	2	In reverse running
0–7		3	Stopped
0-7		4	Faulty
		5	POFF
		6	In pre-excitation
8	Bus voltage established Motor group feedback	1	Ready to run
0		0	Not ready to run
9–10		0	Motor 1
9-10		1	Motor 2
11	Motor type feedback	1	Synchronous motor

Table 2-7 Goodrive350 series VFD SWs

Bit	it Name		State to be entered/description
		0	Asynchronous motor
40	Overload pre-alarm	1	Overload pre-alarm generated
12	feedback	0	No overload pre-alarm generated
13		0	Keypad-based control
	Run/Stop mode	1	Terminal-based control
		2	Communication-based control
14		3	Reserved
15	Heartbeat feedback	1	Heartbeat feedback
		0	No heartbeat feedback

Actual value (ACT): The second to twelfth words in a PZD task packet are the main actual values. The main actual frequency values are provided by the main actual value signal source.

Function	Word	Value range	Default
code	Word	value range	value
P15.13	Transmitted PZD2	0–31	0
P15.14	Transmitted PZD3	0: Invalid	0
P15.15	Transmitted PZD4	1: Running frequency (×100, Hz)	0
P15.16	Transmitted PZD5	2: Set frequency (×100, Hz)	0
P15.17	Transmitted PZD6	3: Bus voltage (×10, V)	0
P15.18	Transmitted PZD7	4: Output voltage (×1, V)	0
P15.19	Transmitted PZD8	5: Output current (×10, A)	0
P15.20	Transmitted PZD9	6: Actual output torque (×10, %)	0
P15.21	Transmitted PZD10	7: Actual output power (×10, %)	0
P15.22	Transmitted PZD11	8: Rotating speed of the running (x1, RPM) 9: Linear speed of the running (x1, m/s)	0
P15.23	Transmitted PZD12	10: Ramp frequency reference 11: Fault code 12: Al1 value (×100, V) 13: Al2 value (×100, V) 14: Al3 value (×100, V) 15: HDIA frequency (×100, kHz) 16: Terminal input state 17: Terminal output state 18: PID reference (×100, %) 19: PID feedback (×100, %) 20: Rated torque of the motor 21: MSB of position reference (signed number) 22: LSB of position reference (unsigned	0

Table 2-8 Actual state values of Goodrive350 series VFD

Function code	Word	Value range	Default value
		number) 23: MSB of position feedback (signed number) 24: LSB of position feedback (unsigned number) 25: State word 2 26: HDIB frequency value (×100, kHz) 27: High-order bit of PG card pulse feedback 28: Low-order bit of PG card pulse	value
		feedback 29: High-order bit of PG card pulse reference 30: Low-order bit of PG card pulse reference 31: Function parameter mapping (PZD2–PZD12 correspond to P14.60–P14.70)	

PKW zone (parameter identification flag PKW1—numerical zone): The PKW zone describes the processing mode of the parameter identification interface. A PKW interface is not a physical interface but a mechanism that defines the transmission mode (such reading and writing a parameter value) of a parameter between two communication ends.

Structure of the PKW zone						
Parameter identification (PKW)			Proces	ss data		
PKW1	PKW2	PKW3	PKW4		PZD2 PZD2	
Request No. Response No.	Parameter address	Parameter value error No.	Parameter value		·	

Figure 2-5 Parameter	identification zone
----------------------	---------------------

In the periodic PROFIBUS-DP communication, the PKW zone consists of four 16-bit words. Table 2-9 describes each word in the PKW zone.

First word PKW 1 (16 bits)					
Bits 15-00	0–7				
	Second word PKW2 (16 bits)				
Bits 15-00	0–247				
	Third word PKW3 (16 bits)				
Bits 15–00 Value (most significant word) of a parameter or error code of the returned value		00			
Fourth word PKW4 (16 bits)					
Bits 15–00 Value (least significant word) of a parameter		0–65535			

Table 2-9 Each word in the PKW zone

Note: If the master station requests the value of a parameter, the values in PKW3 and PKW4 of the packet that the master station transmits to the VFD are no longer valid.

Task request and response: When transmitting data to a slave, the master uses a request number, and the slave uses a response number to accept or reject the request.

Table 2-10 describes the request and response functions.

Requ	est No. (from the master to a slave)	Response signal	
Request No.	Function		Rejection
0	No task	0	-
1	Requesting the value of a parameter	1, 2	3
2	Modifying a parameter value (one word) [modifying the value only on RAM]	1	3 or 4
3	Modifying a parameter value (two words) [modifying the value only on RAM]	2	3 or 4
4	Modifying a parameter value (one word) [modifying the value on both RAM and EEPROM]	1	3 or 4
5	Modifying a parameter value (two words) [modifying the value only on both RAM and EEPROM]	2	3 or 4

The requests #2, #3, and #5 are not supported currently.

	Response No. (from a slave to the master)
Response No.	Function
0	No response
1	Transmitting the value of a parameter (one word)
2	Transmitting the value of a parameter (two words)
3	The task cannot be executed and one of the following error number is returned: 1: Invalid command 2: Invalid data address 3: Invalid data value 4: Operation failure 5: Password error 6: Data frame error 7: Parameter read only 8: Parameter cannot be modified during VFD running 9: Password protection 10: Function code mapping failed
4	Reserved

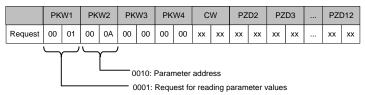
Table 2-11	Response	identification	flag	PKW1
------------	----------	----------------	------	------

PKW examples

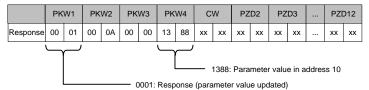
Example 1: Reading the value of a parameter

You can set PKW1 to 1 and PKW2 to 0A to read a frequency set through keypad (the address of the frequency set through keypad is 10), and the value is returned in PKW4. The following data is in hexadecimal format.

Request (master station -> VFD)



Response (VFD -> master station)



Example 2: Modifying the value of a parameter (on both RAM and EEPROM)

You can set PKW1 to 4 and PKW2 to 10 to modify a frequency set through keypad (the address of the frequency set through keypad is 10), and the value to be modified (50.00) is in PKW4.

Request (master station -> VFD)

	PK	W1	PK	W2	РК	W3	PK	W4	C	W	ΡZ	D2	PZ	D3	 PZI	D12
Request	00	04	00	0A	00	00	13	88	хх	xx	xx	хх	xx	xx	 хх	xx
1388: Parameter value in address 10 0004: Parameter value to be modified																

Response (VFD-> master station)

	PK	W1	PK	W2	PK	W3	PK	W4	C	W	PZ	D2	PZ	D3	 PZI	D12
Response	00	01	00	0A	00	00	13	88	xx	xx	xx	xx	xx	xx	 хх	xx
0001: Response (parameter value updated)																

PZD examples: The transmission of the PZD zone is implemented through VFD function code settings. For the function codes, see the related INVT VFD operation manual.

Example 1: Reading the process data of a VFD

In this example, PZD3 is set to "8: Rotating speed of the running" through the VFD parameter P15.14. This operation sets the parameter forcibly. The setting remains until the parameter is set to another option.

Response (VFD -> master station)

	PK	W1	PK	W2	PK	W3	PK	W4	С	W	ΡZ	D2	ΡZ	D3	 PZ	D12
Response	хх	00	0A	 хх	хх											

Example 2: Writing process data to a VFD device

In this example, PZD3 is set to "2: PID reference" through the VFD parameter P15.03. The parameter specified in each request frame is updated with the information contained in PZD3 until another parameter is specified.

Request (master station -> VFD)

	PK	W1	PK	W2	PK	W3	PK	W4	C	W	ΡZ	D2	ΡZ	D3	 PZ	D12
Response	хх	хх	xx	хх	хх	xx	хх	хх	хх	xx	хх	хх	00	00	 хх	хх

Subsequently, the information contained in PZD3 is used as tractive force reference in each request frame until another parameter is specified.

2.7 Example of PROFIBUS-DP communication networking

1. Preparation before networking

Hardware: One PC, three PPROFIBUS communication cards, three GD350 VFDs, and one Siemens PLC S7-300.

Software: Win10 system and Siemens TIAPORTAL V13.

2. GD350 VFD parameter configuration

Set P00.01 (Channel of running commands) to 2 (Communication).

Set P00.02 (Communication channel of running commands) to 1 (PROFIBUS communication).

Set P00.06 (Frequency A command setting mode) to 9 (PROFIBUS communication).

Module address (P15.01) of three VFDs are set to 3, 4, and 5 respectively.

Set P15.02 (Received PZD2) to 1 (Set frequency).

Set P15.13 (Transmitted PZD2) to 1 (Running frequency).

Set P15.14 (Transmitted PZD3) to 3 (Bus voltage).

3. PLC configuration

(1) Create a project.

Click **Create new project**, fill in **Project name**, and select the path where the project is stored, as shown in the following figure.

	Create new project	
Open existing project	Project name: Path:	
Create new project	Author:	Administrator
Migrate project	Comment:	
Close project		
Welcome Tour		
First steps		
Installed software		
💮 Help		
S User interface language		

(2) Add GSD files.

In the following project view, choose **Options** on the toolbar, and choose **Manage general station description files (GSD)** from the drop-down list. Enter the directory where the INVT GSD file is located in the source path, select the GSD file, and click the **Install** button to start the installation.

M Siemens - D:/PLC/PROFIBUS/PRO	FIBUS		
Project Edit View Insert Onlin	e Options Tools Window Help Y Settings Support packages	e 🖋 Goattine 🔥 🖪 🕞 🛪 🗖 🛄	
Devices	Manage general station description files (GSD) Start Automation License Manager Show reference text I Global libraries	-	
Ad new device Ad new device Devices & networks Particle 3 networks Common data Common data Common data Common data Common data			
Age online access Top Card Reader/USB memory			

Manage genera	l station descriptio	1 files			×
Source path:	D:\PLC\PROFIBUS\Addi	tionalFiles\GSE)		
Content of im	ported path				
File		Version	Language	Status	Info
invtv2dp.gsd	i		Default	Not yet installed	
۲		1	11		>
				Delete Install	Cancel

After the installation was complete successfully, a prompt pops up, indicating that the GSD file has been installed successfully.

Manag	je general station d	escription file	S					
Insta	Installation result							
! M	lessage							
0	Installation was com	pleted successfu	illy.					
	Save log	Inst	all additional	files		Close		
	save log	Inst	auditional	lites	l	Close		

(3) Add the project device and PROFIBUS network.

In the Hardware catalog on the right sidebar, choose Controllers > SIMATIC S7-300 > CPU > CPU 315-2 PN/DP > 6ES7 315-2EH14-0A0B, and double click the 6ES7 315-2EH14-0A0B icon or drag it to the project.

PROFIBUS	_#=×	Hardware catalog
🖉 Topology view 🔒	Network view Device view	Options
💦 Network 🔛 Connections HM connection 💌 🖽 🐂	Network overview	
	A V Device	✓ Catalog
	S7300/ET200M station_1	<search> ing</search>
PLC1 OU3152 MIDF	i i	Pine **
		CPU 314C-2 DP
		CPU 314C-2 PN/DP
K III > 100%	• • • • • • • • • • • • • • • • • • •	 CPU 314C-2 PtP
		CPU 315-2 DP
S7300/ET200M station_1 [S7-300 station] C Properties	nfo 🚺 😼 Diagnostics 👘 🗐 🖃 🥆	- CPU 315-2 PN/DP
General IO tags System constants Texts		6ES7 315-2EH13-0AB0
General	^	CPU 317-2 DP

In the Hardware catalog pane, choose Other field devices > PROFIBUS DP > General > INVT ELECTRIC CO.,LTD > INVT > INVT-6SE70, and double click the INVT-6SE70 icon or drag it to the view of Devices & networks. The communication card is displayed as Not assigned. The example shows three slave bus networking, thus you need to add two slaves.

PROFIBUS	💶 🗗 🗮 🗙 Hardware catalog 🛛 🖷 🛾
🛃 Topology view 📑 N	Network view Device view Options
💦 Network 📋 Connections HM connection 💌 📰 🖽 🔍 ±	Network ()
AC.3152 PHOP	Network V Catalog V Drice V State Veta State • Social Veta Vota • Social Veta State • Social Veta Social • Social Veta Social • Social • Social • Social • Soc
	Control devices Control devices

As shown in the following figure, click the **Not assigned** option of **INVT-6SE70** and select **PLC_1.MPI/DP interface_1**, and CPU and INVT-6SE70 in the network view are connected to the same PROFIBUS network. Click the **Not assigned** option of the remaining two PROFIBUS slaves and select **PLC_1.MPI/DP interface_1**, then PROFIBUS master and three slaves are connected to the same PROFIBUS network.

Communication card

PROFIBUS			_ # =×
	Topology view	A Network view	Device view
Network Connections HM connection	• 🗉 🔍 ±		rk overview
C_1 PU 315-2 PN/DP Int assigned scient master: FC_1.MPDP metrice_1	Slave 2 INVT-65270		vice \$7300ET200M station_1 > PLC_1 GSD device_1 GSD device_2 GSD device_2 GSD device_3 Slave_3
< III 100%	▼ <u></u> 8-		>
PROFIBUS > Devices & networks			
Network	🦉 🔠 🍳 ±		Master system: PLC_1.
PLC_1 CPU 315-2 PN/DP PLC_1 PLC_1 PLC_1 PLC_1	Slave_2 INVT-65E70 PLC_1 _1.DP-Mastersystem (1)	DP-NORM	Slave_3 NVT-65E70 PLC_1

(4) Perform PROFIBUS master and slave setting.

PROFIBUS master setting

Click **PROFINET interface_1** network interface position in the PLC icon to enter the PROFINET interface_1 property editing interface of the PLC, as shown in the following figure. Click the **Ethernet addresses** option in the **General** list to set the IP address of the PLC and the IP address of the PC to be in the same subnet.

Click **MPI/DP** interface_1 in the PLC icon to enter the MPI/DP interface_1 property editing interface of PLC. Click the **PROFIBUS** address option in the **General** list to set PROFIBUS address of PLC to 2.

Communication card

	Contraction	4 1 2 −4 5 6 7 8 9 19 11 1000	
Provide Interface		< II 3 100%	·
Control addresses Image Ima			
Contractions Contreactions Contractions Contractions Contractions			
• Operating mode • Mainted space • Mainted space • Particle • Mainted space • Particle • Mainted space • Particle • Particle • • • • • • • • • • • • • • • • • • •		Ethernet addresses	
Departies addresses Medicare subset Medicare subset		Operating mode Advanced options Subnet: Not networked	
Image: Interface Image: Im		IP protocol	
Normality Subservantik 120:233 220 Image: Contract Subservantik Subservantik 220:233 220 Image: Contract Subservantik Subservantik 220:233 220 Image: Contract Subservantik Subservantik 220:233 220 Image: Contract Subservantik Subservantik Subservantik 220:233 220 Image: Contract Subservantik Subservantik Subservantik Subservantik Subservantik Subservantik Image: Contract Subservantik	✓ Details view		
MONUS Monta existe Monta ex	Name	Raddress: 192168.0 Subnet mask: 255255	. 0
MONUS Monta existe Monta ex	N00		
Microire Microire General 10 tags General 10 tags General PROFIBUS address General Index Subret Operating mode Index Subrets Degroup contraction Subrets Degroup contraction Subrets Degroup contraction Subrets Degroup contraction Subrets Parameters Interface networks gate Interface networks gate Parameters	Add new device Add new device Add new device Comparison Compa		
		MPUOD Interface 1 Module General II Datage System constants Texts General II Datage System constants Texts General Interface networked with Interface networked with Submet: Phoneus_1 Disposition addresses Interface networked with	8 100% *
	✓ Details view	Address: 2	

PROFIBUS slave setting

Double click the network interface position in the **INVT-6SE70** slave icon to enter the PROFIBUS interface editing interface. Click the **PROFIBUS address** option in the **General** list, set the slave address to 3, and set addresses of the remaining two slaves to 4 and 5 respectively.

Project tree	14	PROFIBUS > PLC_1 [CPU 315	15-2 PN/DP] Distributed I/O DP-Mastersystem (1): PROFIBUS_1 Slave_1	
Devices				
<u> </u>		de Slave_1	- 🖬 🗹 🚄 🗉 Q.1	
PROFIBUS		1		
Add new device				
Devices & networks				
[1] PLC_1 [CPU 315-2 PN/DP] [2] Common data		· · · · ·		
Documentation settings				
Languages & resources				
Gonline access		_		
Ig Card ReadenUSB memory			(P 2008)	
		Slave 1 [Module]	(3) 100%	•
			System constants Texts	
		 General Catalog information 	PROFIBUS address	
		PROFIBUS address General DP parameters	Interface networked with	
		Watchdog	Subnet: PROFIBUS_1	
		SYNCIFREEZE		
		Diagnostics addresses	•	
			Parameters	
 Details view 		1	Address: 3	
			Highest address: 126	
Name			Transmission speed: 1.5 Mbps	

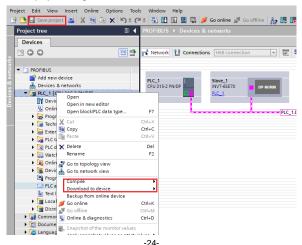
PROFIBUS slave module setting

Double click the INVT-6SE70 slave icon in the Devices & networks view to enter the INVT device view interface. Double click the IN/OUT:32Byte(16word) module or drag it to the blank space in Device view. After IN/OUT:32Byte(16word) module is added to the project, you need to set I address and Q address of IN/OUT:32Byte(16word) to 0...31. The remaining two slaves repeat this operation, and the addresses are increased by degrees.

🖉 Topology view	📩 Ne	twork vi	iew 🛐	Device vie	_	Hardware catalog Options
Device overview Module Slave_1 IN/OUT: 32 Byte (16 word)_1	Rack O	Slot 0	I address 2043* 031	Q address	Type INV	✓ Catalog
	0	2				Fiter MVF6520 Universal module PVW module PVW module PF0 : 4 FWV, 6220 PF0 2: 4 FWV, 6220 PF0 2: 4 FWV, 6220 PF0 2: 4 FWV, 6220 PF0 3: 4 FWV, 6220 PF0 4: 0 FWV, 6220 PF0 5: 4 FWV, 10220 MVDUT 32 Byte (16 wee)

4. Save, compile, and download the project.

After PLC configuration is completed, you need to download the project configuration information to PLC S7-300, as shown in the following figure. Click **Save project** to save the entire project, and right click **PLC_1[CPU 315-2 PN/DP]** and choose **Compile > Hardware and software (change only)** to compile the entire project. Click the **Download to device** icon to download the project configuration to the PLC controller.



Select **PN/IE** from the drop-down list of **Type of the PG/PC interface**, and Click the **Start search** button in the lower right corner to start scanning and detecting PLC devices in the subnet, as shown in the following figure. After searching is completed, the PLCs that are connected to the PC will be displayed in the list of **Compatible devices in target subnet**. Click the **Download** button to download the configuration information and PLC program to the selected PLC.

	Device	Device type	Slot	Type	Address	Subnet
	PLC_1	CPU 315-2 PN/DP	2 X2	PN/IE	192,168.0,1	
		CPU 315-2 PN/DP	2 X1	PROFIBUS	2	PROFIBUS_1
		Type of the PG/PC int	erface:	PN/IE		•
		PG/PC int			rnet Connection (4) I2	
		Connection to interface/s		Direct at slot '2		- 0
		1st ga	teway:			- 💎
		vices in target subnet:			Show all compat	
	Device	Device type	Туре		ddress	Target device
B.			PN/IE	1	92.168.0.1	Target device PLC_1
	Device	Device type		1		
Flash LED	Device	Device type	PN/IE	1	92.168.0.1	
Flash LED	Device PLC_1 -	Device type	PN/IE	1	92.168.0.1	PLC_1 -
Flash LED	Device PLC_1 	Device type	PN/IE	1	92.168.0.1	
Flash LED ine status informa Retrieving device	Device PLC_1 - tion: information	Device type CPU 315-2 PNDP —	PN/IE	1	92.168.0.1	PLC_1 -
Flash LED ine status informa Retrieving device	Device PLC_1 	Device type CPU 315-2 PNDP —	PN/IE	1	92.168.0.1	PLC_1 -

5. View VFD parameters.

Double click **Add new watch table** to create three watch tables for monitoring three VFD parameters respectively, as shown in the following figure.

Project tree	□ ◀	PRO							
Devices									
B 0 0	🔤 🖻		€0	1 10 91 1	62	a aan aan ⊳ 1			
			i	Name		Address	Display format	Monitor value	
PROFIBUS		1				<add new=""></add>			
🚔 Add new device									
📥 Devices & networks									
PLC_1 [CPU 315-2 PN/DP]									
Device configuration									
😡 Online & diagnostics									
Program blocks									
Technology objects									
External source files									
PLC tags									
PLC data types									
 Watch and force tables 									
Add new watch table									
Force table									
👸 Watch table_1									
Watch table_2									
Watch table_3									
Online backups									
🕨 🧱 Device proxy data									
Program info									

Create target watch variables—PZD, PKW, control word and status word variables of the VFD in the newly created **Watch table_1**, and click **Watch all** and **Modify selected values at one time immediately**, as shown in the following figure. Operations in **Watch table_2** and **Watch table_3** are similar to those in **Watch table_1**.

	Name	Address	Display format	Monitor value	Modify value	4	Comment
1		%QW0	Hex	16#0001	16#0001	Í 🖂 🧃	
2		%QW2	Hex	16#000A	16#000A		PKW2
3		%QW4	Hex	16#0000			PKWB
4		%QW6	Hex	16#0000			PKW4
5		%QW8	Hex	16#0101	16#0101	Image: A state of the state	CW
5		%QW10	DEC	5000	5000	Image:	PZD2(INPUT)
7		%QW12	Hex	16#0000			PZD3(INPUT)
в		%QW14	Hex	▼ 16#0000			PZD4(INPUT)
9		%QW16	Hex	16#0000			PZD5(INPUT)
10		%QW18	Hex	16#0000			PZD6(INPUT)
11		%QW20	Hex	16#0000			PZD7(INPUT)
12		%QW22	Hex	16#0000			PZD8(INPUT)
13		%QW24	Hex	16#0000			PZD9(INPUT)
14		%QW26	Hex	16#0000			PZD10(INPUT
15		%QW28	Hex	16#0000			PZD11(INPUT
16		%QW30	Hex	16#0000			PZD12(INPUT)

The above figure shows the examples of setting PKW read parameters and PZD.

- Set PKW1 to 0001 and PKW2 to 000A, indicating the request to read the value of P00.10 (set frequency through keypad).
- Set CW to 0101, indicating that PKW reading and writing function is enabled, and the VFD is controlled to run forward.
- Received PZD2 is set to 5000, indicating that the running frequency of the VFD is set to 50.00Hz.

17	%IW0	Hex	16#0001	PKW1
18	%IW2	Hex	16#000A	PKW2
19	%IW4	Hex	16#0000	PKW3
20	%IW6	DEC	5000	PKW4
21	%IW8	Hex	16#4101	SW SW
22	%IW10	DEC	5000	PZD2(OUTPUT)
23	%IW12	DEC	5656	PZD3(OUTPUT)
24	96IW14	Hex	16#0000	PZD4(OUTPUT)
25	%IW16	Hex	16#0000	PZD5(OUTPUT)
26	%IW18	Hex	16#0000	PZD6(OUTPUT)
27	%IW20	Hex	16#0000	PZD7(OUTPUT)
28	%IW22	Hex	16#0000	PZD8(OUTPUT)
29	%IW24	Hex	16#0000	PZD9(OUTPUT)
30	%IW26	Hex	16#0000	PZD10(OUTPUT)
31	%IW28	Hex	16#0000	PZD11(OUTPUT)
32	%IW30	Hex	16#0000	PZD12(OUTPUT)

The above figure shows the response results after setting PKW read parameters and PZD.

- PKW1: 0001, PKW2: 000A, PKW4: 5000, indicating that the value read from P00.10 is 5000, and 5000 indicates that the frequency set by keypad is 50.00Hz.
- SW: 4101, indicating that the running mode is set to communication control, the bus voltage is established, the VFD is ready to run, and the VFD is in forward running.
- Transmitted PZD2: 5000, indicating that the running frequency of the VFD is 50.00Hz.
- Transmitted PZD3: 5656, indicating that the bus voltage of the VFD is 565.6V.

1	%OW0	Hex	16#0004	16#0004		PKW1
	%QW2	Hex	16#000A	16#000A		PKW2
2				16#000A	M 🔨	
3	%QW4	Hex	16#0000			PKW3
4	%QW6	DEC	4000	4000	🗹 🛕	PKW4
5	%QW8	Hex	16#0105	16#0105	M 🛓	CW
6	%QW10	DEC	5000	5000	M 🛓	PZD2(INPUT)
7	%QW12	Hex	16#0000			PZD3(INPUT)
8	%QW14	Hex	16#0000			PZD4(INPUT)
9	%QW16	Hex	16#0000			PZD5(INPUT)
10	%QW18	Hex	16#0000			PZD6(INPUT)
11	%QW20	Hex	16#0000			PZD7(INPUT)
12	%QW22	Hex	16#0000			PZD8(INPUT)
13	%QW24	Hex	16#0000			PZD9(INPUT)
14	%QW26	Hex	16#0000			PZD10(INPUT)
15	%QW28	Hex	16#0000			PZD11(INPUT)
16	%QW30	Hex	16#0000			PZD12(INPUT)

The above figure shows the examples of setting PKW write parameters and PZD.

- Set PKW1 to 0004, PKW2 to 000A and PKW4 to 4000, indicating that the value of P00.10 (set frequency through keypad) is changed to 40.00Hz.
- Set CW to 0105, indicating that PKW reading and writing function is enabled, and the VFD is controlled to decelerate to stop.
- Received PZD2 is set to 5000, indicating that the running frequency of the VFD is set to 50.00Hz.

17	%IWO	Hex	16#0001	PKW1
18	%IW2	Hex	16#000A	PKW2
19	%IW4	Hex	16#0000	PKWB
20	%IW6	DEC	4000	PKW4
21	%IW8	Hex	16#4103	SW
22	%IW10	DEC	0	PZD2(OUTPUT)
23	%IW12	DEC	5683	PZD3(OUTPUT)
24	%IW14	Hex	16#0000	PZD4(OUTPUT)
25	%IW16	Hex	16#0000	PZD5(OUTPUT)
26	%IW18	Hex	16#0000	PZD6(OUTPUT)
27	%IW20	Hex	16#0000	PZD7(OUTPUT)
28	%IW22	Hex	16#0000	PZD8(OUTPUT)
29	%IW24	Hex	16#0000	PZD9(OUTPUT)
30	%IW26	Hex	16#0000	PZD10(OUTPUT)
31	%IW28	Hex	16#0000	PZD11(OUTPUT)
32	%IW30	Hex	16#0000	PZD12(OUTPUT)

The above figure shows the response results after setting PKW write parameters and PZD.

- PKW1: 0001, PKW2: 000A, PKW4: 4000, indicating that the value read from P00.10 is 4000, and 4000 indicates that the frequency set by keypad is 40.00Hz.
- SW: 4103, indicating that the running mode is set to communication control, the bus voltage is established, the VFD is ready to run, and the VFD is in stopping.
- Transmitted PZD2: 0, indicating that the running frequency of the VFD is 0.00Hz.
- Transmitted PZD3: 5683, indicating that the bus voltage of the VFD is 568.3V.

Chapter 3 CANopen communication card

3.1 Overview

- Thanks for choosing INVT CANopen communication cards. This manual describes the function specifications, installation, basic operation and settings, and information about the network protocol. To ensure that you install and operate the product properly, read this manual and the communication protocol section in the VFD operation manual carefully before you use the product.
- This manual only describes how to operate the CANopen communication card and the related commands but does not provide details about the CANopen protocol. For more information about the CANopen protocol, read the related specialized articles or books.
- This communication card is defined as a CANopen slave station communication card and is used on a VFD that supports CANopen communication.
- 4. The CANopen communication of this communication card supports access to VFDs through process data objects (PDOs) and service data objects (SDOs). PDOs and SDOs are used to read the object dictionary defined by the manufacturer.

Table 3-1 Protocol selection relationship for switch SW2

Switch SW2						
1	1 2 Protocol					
OFF	OFF	CANopen				
ON	OFF	CAN master/slave				

Note: For EC-TX505C, before power-on, set the switch according to the protocol selection relationship to correspond to the actually used protocol.

3.2 Features

- 1. Supported functions
- Supports the CAN2.0A protocol.
- Supports CANopen DS301.
- 2. Supported CANopen services
- PDO: Supports four pairs of PDO services (PDO1 TX to PDO4 TX, and PDO1 RX to PDO4 RX), where the PDO1 pair is used to read and write parameters of a VFD, and the PDO2 to PDO4 pairs are used to control and obtain the actual parameter values of the VFD in real time.
- SDO: SDO information adopts the "client/server" mode and is used to configure slave nodes and provide access to the object dictionary of each node.

- Supports the emergency service.
- Supports node protection (NMT Node Guarding).
- Supports heartbeat packets (Heartbeat Producer).
- Supports network management (NMT).
 - Supports NMT module control.
 - Supports NMT broadcast addresses.
 - Supports NMT error control.
 - Supports boot-up.
- Supports SYNC (1–240).
- Supports asynchronous transmission of 254 and 255.
- Supports disabled time.
- Supports event timers.
- Supports manufacturer-defined object dictionary. You can use SDOs to control and obtain the actual parameter values of a VFD in real time.
- 3. Non-supported CANopen services
- Saves object dictionary parameters at power outage
- Time stamp service
- 4. Supported CANopen addresses and baud rates

Table 3-1 Supported addresses and baud rates

Item	Supported specification
Address	1–127 (decimal)
	1000 kbps
	800 kbps
	500 kbps
Baud rate	250 kbps
Baud rate	125 kbps
	100 kbps
	50 kbps
F	20 kbps

Note: To enable the CANopen functions (except the CANopen communication timeout fault time and baud rate), you need only to select the related PROFIBUS channels. If modification is made on the VFD operation manual, the operation is subject to the CANopen channel, without prior notice in this manual.

3.3 Electrical wiring

Use shielding wires as the bus cable, if possible. It is recommended that you connect the shielding wire to the PE terminal of the communication card. When there are only two devices for CAN master-slave communication, both devices shall be connected to the terminal resistor. When there are more than two devices, the starting device and terminal device shall be connected to the terminal resistor. The terminal resistor of the communication card can be connected through its terminal resistor switch. Figure 3-1 shows the electrical wiring.

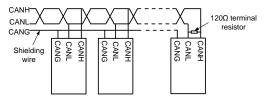


Figure 3-1 Electrical wiring diagram

3.4 Terminal wiring

3.4.1 Terminal layout

Table 3-2 CANopen card	terminal layout
------------------------	-----------------

PO	GND	PE	CANH	CANL

3.4.2 Terminal functions

Signal	Port	Terminal function
PGND	/	Isolation ground
PE	/	CAN bus shield
CANH	/	CAN bus high-level signal
CANL	/	CAN bus low-level signal
	OFF-bounced	No terminal resistor is connected between CAN_H and
Terminal	up	CAN_L.
resistor	ON-pressed	A 120Ω terminal resistor is connected between CAN_H
	ON-pressed	and CAN_L.
Switch	00	CANopen
applicable to	10	CAN master/slave
cards	01	Reserved

Signal	Port	Terminal function
	11	Reserved

Table 3-4 Indicator functions

Indicator	Definition	Function	
LED1	Status indicator	On: The expansion card is establishing a connection with the control board. Blinking periodically: The expansion card is properly connected to the control board (the period is 1s, on for 0.5s, and off for the other 0.5s). Off: The expansion card is disconnected from the control board.	
LED2	Run indicator	On: The communication card is running. Off: A fault occurs. Check whether the reset pin of the communication card and the power supply are properly	
LED3	Alarm indicator	On: The CAN controller bus is off or the VFD has a fault. Off: The communication card is in the working state.	
LED4	Power indicator	It is on since the control board feeds power to the communication card.	

3.5 Communication

3.5.1 Packet format

CAN2.0A packets are used to transmit data between the master station and bus nodes through data frames.

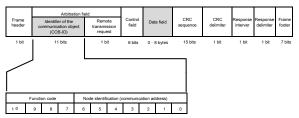


Figure 3-2 Packet structure

Communication object	Function code (binary)	COB-ID (hexadecimal)
NMT	0	0x00
SYNC	1	0x80
EMERGENCY	1	0x81–0xFF
PDO1 Tx	11	0x181–0x1FF
PDO1 Rx	100	0x201–0x27F
PDO2 Tx	101	0x281-0x2FF
PDO2 Rx	110	0x301-0x37F
PDO3 Tx	111	0x381-0x3FF
PDO3 Rx	1000	0x401–0x47F
PDO4 Tx	1001	0x481–0x4FF
PDO4 Rx	1010	0x501–0x57F
SDO Tx	1011	0x581–0x5FF
SDO Rx	1100	0x601–0x67F
Node protection	1110	0x701–0x77F

COB-IDs vary according to communication address, but for one command, the COB-IDs are within a certain range.

Note: The commands described in this manual are all data frames if it is not specified that they are remote frames.

3.5.2 CANopen state transition

The start sequence defined in the CANopen communication protocol is supported. Figure 3-3 shows the NMT state transition diagram.

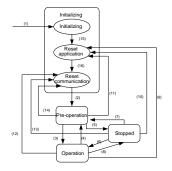


Figure 3-3 NMT state diagram

State transition	Required triggering event
(1)	Automatic initialization after power-on
(2)	Automatic change after initialization
(3), (6)	Command of the NMT master station for starting a remote node
(4), (7)	Command of the NMT master station for entering the pre-operation state
(5), (8)	Command of the NMT master station for entering the stopped state
(9), (10), (11)	Command of the NMT master station for resetting a remote node
(12), (13), (14)	Command of the NMT master station for resetting a remote node communication parameter

Table 3-2 NMT state transition

Different services are supported in different states, as described in Table 3-3.

Table 3-3 Services supported in various NMT states

Service	Pre-operation state	Operation state	Stopped state
PDOs	No	Yes	No
SDOs	Yes	Yes	No
SYNC packets	Yes	Yes	No
Emergency packets	Yes	Yes	No
Network management	Yes	Yes	No
Error control	Yes	Yes	Yes

3.5.3 Management service command (NMT)

This function is used by the master station to control the NMT states of slave station nodes.

Command

Master station -> slave station

COB-ID	Byte0	Byte1
0x000	Command specifier (CS)	Node-ID (Node ID)

Description

In this command, the COB-ID is 0×00 . If Node-ID is set to 0, the command is broadcast to all CANopen slave stations, and each slave station must execute the NMT command. Table 3-4 describes the function of each CS.

Table 3-4 Function of each CS

NMT CS	NMT service (control action)	
0x01	Starts a slave station device.	
0x02	Stops a slave station device.	
0x80	Enables a slave station to enter the pre-operation state.	
0x81	Resets a slave station.	
0x82	Resets communication of a node.	

Example

For example, the command to enable EC-TX505, whose node ID is 3, to enter the pre-operation state is described as follow.

COB-ID	Byte0	Byte1
0x000	0x80	0x03

For another example, the command to start all EC-TX505 nodes on the CANopen network is described as follows.

COB-ID	Byte0	Byte1
0x000	0x01	0x00

3.5.4 Node protection (NMT Node Guarding)

By using the node protection service, the NMT master node can detect the current state of each node.

Command

Request: Master station (remote frame) -> slave station

COB-ID	No data
0x700 + Node-ID	

Response: Slave station -> master station

COB-ID	Byte0 (state value)
0x700 + Node-ID	Bit 7: Triggering bit; Bits 0 to 6: State

Description

The most significant bit (MSB) bit 7 of Byte0 (state value) in the response command is the triggering bit, that is, the value of bit 7 is alternated between 0 and 1 each time when the slave station transmits a response frame to distinguish frames. Bits 0 to 6 indicate the state of the slave station. Table 3-5 describes the state values and their corresponding state.

Table 3-5 State	values and	their	corresponding states
-----------------	------------	-------	----------------------

State value (Byte0: Bits 0-6)	State
0x00	Initializing
0x04	Stopped
0x05	Operation
0x7F	Pre-operational

Example

For example, the command for the master station to detect the state of slave station 3.

Master station (remote frame) -> slave station

COB-ID	No data
0x703	/

After receiving the node protection command transmitted by the master station, the slave station transmits the following command response to the master station.

COB-ID	Byte0 (state value)
0x703	0x85

In the command, bit 7 of Byte0 is 1, and the state value is 0×05 , indicating that slave station 3 is in the operation state. If receiving another node protection command, the slave station transmits a command frame in which the state value is 0×05 to the master station, and the value of bit 7 is alternated to 0.

3.5.5 Heartbeat packet (Heartbeat Producer)

In some cases, the master station requires that a slave station automatically transmits a frame of heartbeat packets at an interval, so that it can learn the state of the slave station in real time. The interval parameter (data length: 16 bits; unit: ms) is defined in the object dictionary 0x1017. If the interval is set to 0, the slave station does not transmit heartbeat packets. For this CANopen communication card, the interval is set to 0 by default.

Command

Slave station -> master station

COB-ID	Byte0
0x700 + Node-ID	State value

Description

The heartbeat packets are in the same format with the node protection response frames. The difference between them is that no triggering bit alternation is performed for heartbeat packets (the triggering bit is always 0). Table 3-5 describes the state values.

Example

For example, if slave station 3 is in the operation state and the interval parameter in 0x1017 is set to 100, slave station 3 transmits a frame of heartbeat packets every 100 ms.

COB-ID	Byte0
0x703	0x05

SDOs can be used to disable heartbeat packets, transmitting 2B 17 10 00 00 00 00 00 (setting the interval to 0).

Note: On the communication card, node protection and heartbeat packets cannot be used simultaneously.

3.5.6 Start packet (NMT Boot-up)

After being initialized (booted up), the communication card transmits a start packet.

Command

Slave station -> master station

COB-ID	Byte0
0x700 +Node-ID	0x00

Example

For example, after being initialized, the communication card whose node ID is 3 transmits the following start packet.

COB-ID	Byte0
0x703	0x00

3.5.7 Synchronous packet object (SYNC)

Generally, SYNC signals are transmitted by the CANopen master station cyclically. A SYNC signal does not contain any data and is used mainly to request PDO Tx of a slave station node of the synchronous transmission type. 0x1005 in the object dictionary defines COB-IDs of the objects that receive synchronous packets, and they are set to 0x80 in the CANopen pre-defined connection set. For PDO Tx, the transmission types of 1 to 240 indicate synchronous transmission.

Command

Master station -> slave station

COB-ID	No data
0x80	/

3.5.8 Emergency packet object (EMCY)

This packet is transmitted when an internal error occurs on the communication card or VFD, or an error is deleted.

Command

Slave station -> master station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x80 +	Emergency error code		Error		V	'FD error c	ode	
Node-ID	LSB	MSB	register	bit7-0	bit15-8	bit23-16	bit31-24	bit39-32

Description

An emergency error code is two bytes. Byte0 is the least significant byte (LSB), and Byte1 is the most significant byte (MSB). A VFD error code is five bytes. Byte3 is the LSB, and Byte7 is the MSB.

An emergency error code indicates the type of the current error, as described in Table 3-6. The error register stores the type of the current error. You can determine the error type indicated by the current emergency packet according to the value stored in the register. Table 3-7 describes the indication of the bits of the error register. For information about the VFD error codes, see the VFD operation manual. The function code P07.27 in Appendix B describes the error codes of Goodrive350 VFD.

Emergency error code (hex)	Code function description			
00xx	Error reset or no error			
10xx	Generic error			
20xx	Current			
21xx	Current error on the, device input side			
22xx	Current error inside the device			
23xx	Current error on the device output side			
30xx	Voltage error			
31xx	Mains voltage			
32xx	Voltage inside the device			
33xx	Output voltage			
40xx	Temperature			
41xx	Ambient temperature			
42xx	Device temperature			
50xx	Device hardware			
60xx	Device software			
61xx	Internal software			
62xx	User software			
63xx	Data set			

Table 3-6 Emergency error codes

Emergency error code (hex)	Code function description
70xx	Additional modules
80xx	Monitoring
81xx	Communication error
8110	CAN overrun
8120	Error passive
8130	Life guard Error or heartbeat error
8140	Recovered from Bus-Off
82xx	Protocol error
8210	PDO not processed due to length error
8220	Length exceeded
90xx	External error
F0xx	Additional functions
FFxx	Device specific

Table 3-7 Error register bits

Error register bit	Error type
0	Generic error or no error
1	Current error
2	Voltage error
3	Temperature error
4	Communication error
5	Device description error
6	Reserved (=0)
7	Manufacturer-defined error

Example

For example, if the "inverter unit phase U protection (OUT1)" fault occurs on the Goodrive350 VFD whose node ID is 3, and the fault type is 1 (that is, the VFD error code is 1), the communication card transmits the following emergency packet.

COB-ID	Emerg error	gency code	Error register	VFD error code				
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x83	0x00	0x30	0x04	0x01	0x00	0x00	0x00	0x00

As you can see in the command, the emergency error code is 0x3000, indicating a voltage error. The error register is 0x04, that is, the second bit is "1", indicating a voltage error. The device error code is 0x0000000001. See the Goodrive350 VFD operation manual, and you can find that the error code 1 indicates the "inverter unit phase U protection (OUT1)" fault.

After the fault is reset, the communication card transmits the following emergency packet to notify the master station that the slave station is no longer faulty.

COB-ID		gency code	Error register	VFD error code				
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x83	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00

3.5.9 Service data object (SDO)

SDOs are mainly used to transmit non-time key data. By using SDOs, the master station can read data from and write data to the object dictionary of a device.

Command

Request: master station -> slave station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x600+NodeID	Request	Objec	t index	Cubindau	Response data			
	code	LSB	MSB	Subindex	bit7-0	bit15-8	bit23-16	bit31-24

Response: Slave station -> master station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0E00 · NadalD	Respons	Objec	t index	Cultile days		Respor	se data	
0x580+NodeID	e code	LSB	MSB	Subindex	bit7-0	bit15-8	bit23-16	bit31-24

Description

An object index is two bytes. Byte1 is the LSB, and byte2 is the MSB. For information about the indexes and subindexes, see the object dictionary in the appendix. Request codes include request codes for reading and those for writing.

Request codes for writing vary according to the character length of items in the object dictionary, and the request code for reading are 0x40. See Table 3-8.

Response codes indicating successful reading vary according to the character length of items in the object dictionary, and the response code indicating successful writing are 0×60 . The response codes indicating reading failure and writing failure are both 0x80. See Table 3-9.

Request	Request	Command	Requested data					
code type	code	description	Byte4	Byte5	Byte6	Byte7		
	0x23	Writes 4-byte data	bit7-0	bit15-8	bit23-16	bit31-24		
Write	0x2B	Writes 2-byte data	bit7-0	bit15-8	-	-		
0x2	0x2F	Writes 1-byte data	bit7-0	-	-	-		
Read	0x40	Reads data	-	-	-	-		

Table 3-8 SDO request codes and requested data

Response	Response	Command	Response data					
code type	code	description	Byte4	Byte5	Byte6	Byte7		
	0x43	Reads 4-byte data	bit7-0	bit15-8	bit23-16	bit31-24		
Read 0x4B 0x4F	Reads 2-byte data	bit7-0	bit15-8	-	-			
	0x4F	Reads 1-byte data	bit7-0	-	-	-		
Write	0x60	Writing succeeds	-	-	-	-		
Deed/write	0x80	Reading/writing		Interruption	error code			
Read/write	0880	fails	bit7-0	bit15-8	bit23-16	bit31-24		

Table 3-9 SDO response codes and response data

Note: The symbol "-" in Table 3-8 and Table 3-9 indicates that the byte is reserved and provides no function.

Table 3-10 describes the interruption error codes.

Table 3-10 Interruption error codes

Interruption code	Code function description
0503 0000	Triggering bit not alternated
0504 0000	SDO protocol times out
0504 0001	Invalid or unknown client/server
0504 0002	Invalid block size
0504 0003	Invalid sequence number
0504 0004	CRC error
0504 0005	Memory overflow
0601 0000	No access to the object
0601 0001	Attempts to read a write-only object
0601 0002	Attempts to write information to a read-only object
0602 0000	Object cannot be found in the object dictionary
0604 0041	Object cannot be mapped to PDO
0604 0042	Number and length of the object to be mapped exceeds the PDO length
0604 0043	Common parameter incompatibility
0604 0047	Common internal incompatibility of the device
0606 0000	Object access failure caused by hardware error
0607 0010	Data type not matched; service parameter length not matched
0609 0011	Subindex cannot be found in the object dictionary
0609 0030	Parameter value range exceeded
0609 0031	Written parameter value too large

Interruption code	Code function description
0609 0032	Written parameter value too small
0609 0036	Max. value less than Min. value
0800 0000	Common error
0800 0020	Data failed to be transmitted or stored in the application
0800 0021	Data failed to be transmitted or stored in the application due to
0800 002 1	device control
0800 0022	Data failed to be transmitted or stored in the application due to the
0800 0022	current state of the device
0800 0023	Error occurs dynamically on the object dictionary or object
0000 0023	dictionary cannot be found

Example

For example, slave station 3 reads data from and writes data to the object whose index is 0x1801 and subindex is 03. (The object whose index is 0x1801 and subindex is 03 indicates the disabled time of PDO2 Tx. For more information, see Chapter 4.)

Write operation example: To modify the disabled time of PDO2 Tx to 1000 ms, the master station transmits the following write operation command.

COB-ID	Request code	Object	index	Subindex	Requested data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x603	0x2B	0x01	0x18	0x03	0xe8	0x03	0x00	0x00

After receiving the command transmitted by the master station, the slave station transmits the following command response if the modification is successful.

COB-ID	Response code	Object	index	Subindex	Response data				
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	
0x583	0x60	0x01	0x18	0x03	0x00	0x00	0x00	0x00	

Read operation example: To read the disabled time of PDO2 Tx, the master station transmits the following read operation command.

COB-ID	Request code	Object index		Subindex	Requested data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x603	0x40	0x01	0x18	0x03	0x00	0x00	0x00	0x00

After receiving the command transmitted by the master station, the slave station transmits the following command response if the current disabled time of PDO2 Tx is 1000 ms.

COB-ID	Respons e code	Object index		Subindex	Response data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x583	0x43	0x01	0x18	0x03	0xe8	0x03	0x00	0x00

Read/write error example: The master station transmits the following read operation command to read an object (whose index is 0x6000 and subindex is 0x00) that cannot be found.

COB-ID	Request code	Object	index	Subindex	Requested data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x603	0x40	0x00	0x60	0x00	0x00	0x00	0x00	0x00

The object cannot be found, and therefore the slave station transmits the following read/write error command response.

COB-ID	Response code	Object index		Subindex	Response data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x583	0x80	0x00	0x60	0x00	0x00	0x00	0x02	0x06

The error code in the response is 0x06020000, indicating that "Object cannot be found in the object dictionary".

3.6 Process data object (PDO)

The communication card provides four PDO Tx commands (whose indexes are 0x1800 to 0x1803) and four PDO Rx commands (whose indexes are 0x1400 to 0x1403). PDO Rx is a PDO command transmitted by the master station to a slave station, that is, it is a master station command. PDO Tx is a PDO command transmitted by a slave station to the master station.

The CW, SW, setting, and return value of each PDO of the communication card are all defined with a "manufacturer-defined object dictionary". In this way, the process data of a VFD can be monitored not only through PDOs but also through SDOs. For more information, see the next chapter. Each PDO command is labeled with "manufacturer-defined object dictionary" in the format of 0xXXXX.HH, where XXXX indicates an index, HH indicates a subindex, and both of them are hexadecimal.

3.6.1 Triggering mode of PDO Tx

Each PDO Tx is defined with a transmission type, disabled time, and event timer. The corresponding subindex of the transmission type is 0x02, that of the disabled time is 0x03, and that of the event timer is 0x05. Therefore, the object dictionary index corresponding to PDO2 Tx is 0x1801, and the subindex is 0x02. The same principle applies to other PDO Tx commands. For more information, see Appendix A.

Synchronous triggering: When the transmission type is set to 1 to 240, PDO Tx is synchronous transmission. For example, if you set the transmission type of PDO2 Tx to n ($1 \le n \le 240$), a slave station transmits one PDO2 Tx command every time after it receives n synchronous packet objects. The same principle applies to other PDO Tx commands.

Asynchronous triggering (254): When the value of the event timer is not zero, a slave station transmits PDO Tx commands periodically. For example, if the event timer of PDO2 Tx is set to 200, the slave station transmits a PDO2 Tx command at the interval of 200 ms. When the value of the event timer is zero, the slave station transmits a PDO Tx command once the corresponding PDO Tx data changes, and the transmission interval is subject to the disabled time. A PDO Tx packet can be transmitted only once in the disabled time, which effectively reduces the load of the bus. When the disabled time is set to a period shorter than 50 ms, 50 ms is used as the disabled time.

Asynchronous triggering (255): When the value of the event timer is not zero, a slave station transmits PDO Tx commands periodically. For example, if the event timer of PDO2 Tx is set to 200, the slave station transmits a PDO2 Tx command at the interval of 200 ms. When the value of the event timer is zero, the slave station transmits a PDO Tx command once a corresponding PDO Rx command is received. For example, after receiving a PDO2 Rx command, the slave station transmits a PDO2 Tx command.

Triggering mode	Transmission type (decimal)		PDO1 TX	PDO2 TX	PDO3 TX	PDO4 TX
Synchronous	1–240	/	Non-supported	Supported	Supported	Supported
Asynchronous	254	Event timer	Non-supported	Supported	Supported	Supported
		Disabled time	Non-supported	Supported	Supported	Supported
		Event timer=0	Supported	Supported	Supported	Supported
	255	Event timer=0	Non-supported	Supported	Supported	Supported

Table 3-11 Triggering modes supported by the communication card

Table 3-12 Default PDO Tx settings of the communication card

	PDO1 TX	PDO2 TX	PDO3 TX	PDO4 TX
Transmission type	255	254	254	254
Event timer (ms)	0	0	0	0
Disabled time (ms)	500	500	500	500

For how to set the triggering type of PDO Tx, see the description of SDO commands.

3.6.2 PDO1

PDO1 is used to read and write parameters of the VFD. The function of PDO1 is similar to that of an SDO. SDOs are used to read and write objects of an object dictionary, and PDO1 is used to read and write parameters of the VFD.

Note: PDO1 Tx support only the transmission type of asynchronous transmission 255. Do not set it to other transmission types, and do not try to set the event timer to periodically transmits PDO1 Tx to the master station.

3.6.2.1 PDO1 Rx

Command

Request: Master station -> slave station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5
0x200+NODE-ID	Request code		Parameter address		Requested data	
	0x2100.00		0x210	0.01	0x2100.02	

Description

A request code is two bytes. Byte0 is the LSB, and Byte1 is the MSB. The manufacturer defines the index 0x2100 and subindex 0x00 for the request codes. Table 3-13 describes the functions of the request codes.

Table 3-13 Request codes

Request code	Function
0	No task
1	Reading the value of a parameter
2	Modifying a parameter value [modifying the value only on RAM]
4	Modifying a parameter value [modifying the value only on both RAM and EEPROM] (reserved)

A parameter address is two bytes. Byte2 is the LSB, and Byte3 is the MSB. It indicates the address of the parameter to be read or modified.

Goodrive350 series VFD function code address representation rules: The MSB is the hexadecimal form of the number before the dot mark, and LSB is that of the number behind the dot mark. Take P10.01 as an example, the number before the dot mark is 10, that is, the MSB of the parameter address is 0x0A; and the number behind the dot mark is 01, that is, the LSB is 0x01. Therefore, the function code address is 0x0A01.

Function code	Name	Detailed parameter description	Default value	Modify
P10.00	Simple PLC mode	0: Stops after running once 1: Keeps running in the final value after running once 2: Cyclic running	0	0
P10.01	Simple PLC memory selection	0: Not saving data at power outage 1: Saving data at power outage	0	0

Table 3-14 Goodrive350 series VFD parameter addresses

VFD parameter address representation rules: You can see the function code in the function parameter list in the VFD operation manual. The hexadecimal form of the value corresponding to the function code is the parameter address. For example, the value corresponding to the function code P13.14 is 1314, and therefore the parameter address of the function code is 0x522 (that is, 1314 in the decimal form).

A piece of requested data is two bytes. Byte4 is the LSB, and Byte5 is the MSB. It indicates the data to be modified. When the command is transmitted for reading data, the requested data is not used.

Note: The data domain of PDO1 Rx must be six bytes. Otherwise, the communication card reports an emergency packet.

3.6.2.2 PDO1 Tx

Command

Response: Slave station -> master station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x180+NODEID	Response code		Error code		Response data		0x00	0x00
	0x200	00.00	0x20	00.01	0x2000.02		-	-

Description

Byte6 and Byte7 are reserved and both are 0x00.

A response code is two bytes. Byte0 is the LSB, and Byte1 is the MSB. Table 3-15 describes the functions of the response codes.

Response code	Function
0	No response
1	Reading or writing succeeds
2	A reading or writing error occurs. Table 3-16 describes the error
3	codes.

Table 3-15 Response codes

A piece of response data is four bytes. Byte4 is the LSB, and Byte7 is the MSB. When a write command is responded, the response data is the data to be modified; and when a read command is responded, the response data is the data to be read.

An error code is two bytes. Byte2 is the LSB, and Byte3 is the MSB. Error codes are valid only when the response code is 3. An error code indicates the reason why it fails to respond to PDO1 Rx. Table 3-16 describes the definitions of the error codes.

Code	Name	Definition
00H	No error	/
01H	Invalid command	 The operation corresponding to the request code is not allowed to be executed. The possible causes are as follows: The function code is applicable only on new devices and is not implemented on this device. The slave station is in the faulty state when processing this request.
02H	Invalid data address	For a slave device, the data address in the request of the master station is not allowed. In particular, the combination of the register address and the number of the to-be-transmitted bytes is invalid.
03H	Invalid data value	The received data domain contains a value that is not allowed. The value indicates the error of the remaining structure in the combined request. Note: It does not mean that the data item submitted for storage in the register includes a value unexpected by the program.
04H	Operation failure	The parameter is set to an invalid value in the write operation. For example, a function input terminal cannot be set repeatedly.
05H	Password error	The password entered in the password verification address is different from that set by the user.
06H	Data frame error	The length of the data frame transmitted by the upper computer is incorrect, or in the RTU format, the value of the CRC check bit is inconsistent with the CRC value calculated by the lower computer.
07H	Parameter read-only	The parameter to be modified in the write operation of the master station is a read-only parameter.

Table 3-16 Error codes

Code	Name	Definition				
08H	Parameter cannot be modified in	The parameter to be modified in the write operation of the master station cannot be modified during the running of the				
0011	running	VFD.				
09H	Password protection	A user password is set, and the master station does not provide the password to unlock the system when performing a read or write operation. The error of system locked is reported.				

Example of PDO1

The VFD is a Goodrive350 series VFD, and the slave station address is 3. Assume that you want to set the function code P15.13 of the VFD to 1.

Command analysis: The parameter address of P15.13 is 0×0 F0D. According to the protocol, the request code of PDO1 Rx is 0×02 , the parameter address is 0×0 F0D, and the requested data is 0×01 , and therefore PDO1 Rx transmitted by the master station is as follows.

COB-ID	Reques	st code	Paramete	r address	Requested data		
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	
0x203	0x02	0x00	0x0D	0x0F	0x01	0x00	

If the VFD parameter is successfully modified, the following PDO1 Tx command is returned.

COB-ID	-ID code		Error code		Response data		-	
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x183	0x01	0x00	0x00	0x00	0x01	0x00	0x00	0x00

3.6.3 PDO2 Rx

PDO2 Rx is used to modify CWs and real-time process data (setting 1, setting 2, and setting 3) of a VFD. A CW is used to control the start and stop of a VFD, and settings are used to control the real-time running values of the VFD, such as set frequency.

Command

Master station -> slave station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x300+NODEID	CW		Setting 1		Setting 2		Setting 3	
	0x21	01.00	0x2100.03		0x2100.04		0x2100.05	

Description

A CW is two bytes. Byte0 is the LSB, and Byte1 is the MSB. Table 3-17 describes Goodrive350 series VFD CWs.

Bit	Name	Value	Description
		1	Forward running
		2	Reverse running
		3	Forward jogging
	Communication-based	4	Reverse jogging
0–7	control command	5	Stop
	control command	6	Coast to stop
		7	Fault reset
		8	Stop jogging
		9	Emergency stop
8	Enable write	1	Enable writing (mainly through PKW1 to
0		1	PKW4)
9–10	Motor group setting	00	Motor 1
5-10	Motor group setting	01	Motor 2
11	Control mode switching	1	Enable torque/speed control switching
11	Control mode switching	0	Disable switching
12	Reset power consumption	1	Enable
12	to zero	0	Disable
13	Pre-excitation	1	Enable
13	Pre-excitation	0	Disable
14	DC braking	1	Enable
14	DC blaking	0	Disable
15	Heartbeat reference	1	Enable
15	meanbear reference	0	Disable

Table 3-17 Goodrive350 series VFD CWs

The function of each setting can be set through the corresponding function code of the VFD. The setting method is the same as that for "received PZD" in PROFIBUS communication. For details, see the VFD operation manual. Setting 1, setting 2, and setting 3 correspond to received PZD2, received PZD3, and received PZD4, respectively. To set the function of setting 1 to "Set frequency", you need only to set "Received PZD2" to "1: Set frequency". The same principle applies to other settings. When multiple settings are enabled, the failure to set one setting (for example, the set value exceeds the setting range) does not affect the setting of other settings.

Example

Assume that the VFD is a Goodrive350 series VFD, the slave station address is 3, you control the running of the VFD through CANopen communication, and you want to set the running frequency to 50 Hz through CANopen communication.

Command analysis: You need to set the VFD start mode and frequency reference mode to CANopen communication (P00.01=2, P00.02=1, P00.06=9) first. In this example, use Setting 2 to set the running frequency (P15.03=1, that is, set Received PZD3 to "1: Set frequency").

When a CW is 0×01 , it indicates that the VFD is to be run. To set the frequency to 50 Hz, you need to set Setting 2 to 5000, that is, 0×1388 .

The PDO2 Rx command transmitted by the master station is as follows.

COB-ID	CW		Setting 1		Setting 2		Setting 3	
COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x303	0x01	0x00	0x00	0x00	0x88	0x13	0x00	0x00

3.6.4 PDO2 Tx

PDO2 Tx is a command transmitted by a VFD to the master station. It contains a SW and real-time process data (Returned value 1, returned value 2, and returned value 3). A SW is used to notify of the state of the VFD, and returned values are used to transmit the real-time running values of VFD, such as running frequency.

The default transmission type of PDO2 Tx is 254, and therefore PDO2 Tx is transmitted once data corresponding to a SW or returned value changes.

Command

Slave station -> master station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x280+NODEID	SW		Returned value 1		Returned value 2		Returned value 3	
	0x200	01.00	0x200	00.03	0x200	00.04	0x200	00.05

Description

A SW is two bytes. Byte0 is the LSB, and Byte1 is the MSB. Table 3-18 describes the definitions of the Goodrive350 series VFD SWs. For VFD of other series, see the corresponding VFD operation manual.

Bit	Name	Value	Description
	-7 Running state	1	In forward running
		2 In reverse running	
0–7		3	Stopped
		4	Faulty
		5	POFF
8	Bus voltage established	1	Ready to run

Bit	Name	Value	Description
		0	Not ready to run
9–10	Mater meur faarle als	0	Motor 1
9-10	Motor group feedback	1	Motor 2
44	Matan tura faadhaali	1	Synchronous motor
11	Motor type feedback	0	Asynchronous motor
40	Overload pre-alarm	1	Overload pre-alarm generated
12	feedback	0	No overload pre-alarm generated
		0	Keypad-based control
10 11	Devision and the	1	Terminal-based control
13–14	Run/stop mode	2	Communication-based control
		3	Reserved
45	l le entir e et fe e elle e els	1	Heartbeat feedback
15	Heartbeat feedback	0	No heartbeat feedback

The function of each returned value can be set through the corresponding function code of the VFD. The setting method is the same as that for "Transmitted PZD" in PROFIBUS communication. For details, see the VFD operation manual. Returned value 1, returned value 2, and returned value 3 correspond to transmitted PZD2, transmitted PZD3, and transmitted PZD4, respectively. To set the function of returned value 1 to "Running frequency", you need only to set "Transmitted PZD2" to "1: Running frequency". The same principle applies to other returned values. Multiple returned values can be enabled simultaneously.

Example

Assume that the VFD is a Goodrive350 series VFD, the slave station address is 3, the VFD is running, and the running frequency is 50.00 Hz. Returned value 1 is set to "Running frequency", returned value 2 is set to "Output voltage", and returned value 3 is set to no function.

Command analysis: You need to set returned value 1 to the running frequency of the VFD (P15.13=1), returned value 2 to the output voltage of the VFD (P15.14=4), and returned value 3 to invalid (P15.15=0) first.

The VFD is running and the bus voltage has been established, and therefore the SW is 0x0101. The running frequency is 50.00 Hz, and therefore returned value 1 is 5000, that is, 0x1388. If the output voltage is 380 V, returned value 2 is 0x017C.

The PDO2 Tx command transmitted by the VFD is as follows.

COB-ID	sw		Returned value 1		Returned value 2		Returned value 3	
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x283	0x01	0x01	0x88	0x13	0x7C	0x01	0x00	0x00

3.6.5 PDO3 Rx and PDO4 Rx

PDO3 Rx and PDO4 Rx are used to modify the real-time process data of a VFD, such as set frequency.

PDO3 Rx command

Master station -> slave station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
	Setting 4		Setting 5		Setting 6		Setting 7	
0x400+NODEID	0x210	0.06	0x210	0.07	0x21	80.00	0x21	00.09

PDO4 Rx command

Master station -> slave station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x500+NODEID	Setting 8		Setting 9		Setting 10		Setting 11	
0x500+NODEID	0x210	00.0a	0x21	00.0b	0x2100.0c		0x2100.0d	

Description

The application methods for PDO3 Rx and PDO4 Rx are the same as that for PDO2 Rx. For the relationship between the settings and PZD in PROFIBUS communication, see Table 3-19.

3.6.6 PDO3 Tx and PDO4 Tx

PDO3 Tx and PDO4 Tx are used by the VFD to transmit real-time process data to the master station, such as running frequency.

The default transmission type of PDO3 Tx and PDO4 Tx is 254, and therefore PDO3 Tx or PDO4 Tx is transmitted once data corresponding to a returned value in the same command changes.

• PDO3 Tx command

Slave station -> master station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
	Returned	d value 4	Returned	d value 5	Returned	d value 6	Returned	d value 7
0x380+NODEID	0x200	0.06	0x200	0.07	0x200	80.00	0x200	00.09

PDO4 Tx command

Slave station -> master station

COB-ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x480+NODEID	Returned value 8		Returned value 9		Returned value 10		Returned value 11	
UX460+NODEID	0x2000	.0a	0x2000.0b		0x2000.0c		0x2000.0d	

Description

The application methods for PDO3 Tx and PDO4 Tx are the same as that for PDO2 Tx. For the relationship between the returned values and PZD in PROFIBUS communication, see

Table 3-20.

3.7 Monitoring process data through SDO commands

The communication can use SDOs as well as PDOs to monitor the process data of a VFD. You can select a monitoring mode as required. You can monitor the VFD by using SDOs to read the manufacturer-defined object dictionary.

For the definition and application of the CWs, SWs, settings, and returned values in the manufacturer-defined object dictionary, see the PDO description section. For application of SDOs, see the SDO description section. Do not try to use SDOs to read and write VFD parameters.

Table 3-19 and

Table 3-20 describe the manufacturer-defined object dictionary.

Table 3-19 Objects with the control function in the manufacturer-defined object dictionary

Index (hexadecimal)	Subindex (hexadecimal)	Function	Access permission	Data length	Corresponding to
	0	Request code (do not use it)	RW	2 bytes	/
	1	Parameter address (do not use it)	RW	2 bytes	/
	2	Requested data (do not use it)	RW	2 bytes	/
2100	3	Setting 1	RW	2 bytes	Received PZD2
	4	Setting 2	RW	2 bytes	Received PZD3
	5	Setting 3	RW	2 bytes	Received PZD4
	6	Setting 4	RW	2 bytes	Received PZD5
	7	Setting 5	RW	2 bytes	Received PZD6
	8	Setting 6	RW	2 bytes	Received PZD7
	9	Setting 7	RW	2 bytes	Received PZD8
	А	Setting 8	RW	2 bytes	Received PZD9

Index (hexadecimal)	Subindex (hexadecimal)	Function	Access permission	Data length	Corresponding to
	В	Setting 9	RW	2 bytes	Received PZD10
	С		RW	2 bytes	Received PZD11
	D	Setting 11	RW	2 bytes	Received PZD12
	E	Reserved	RW	2 bytes	/
	F	Reserved	RW	2 bytes	/
2101	0	CW	RW	2 bytes	/

Table 3-20 Objects with the monitoring function in the manufacturer-defined object dictionary

Index (hexadecimal)	Subindex (hexadecimal)	Function	Access permission	Data length	Corresponding to
	0	Response code (do not use it)	RO	2 bytes	/
	1	Error code (do not use it)	RO	2 bytes	/
	2	Response data (do not use it)	RO	2 bytes	/
	3	Returned value 1	RO	2 bytes	Transmitted PZD2
	4	Returned value 2	RO	2 bytes	Transmitted PZD3
	5	Returned value 3	RO	2 bytes	Transmitted PZD4
2000	6	Returned value 4	RO	2 bytes	Transmitted PZD5
	7	Returned value 5	RO	2 bytes	Transmitted PZD6
	8	Returned value 6	RO	2 bytes	Transmitted PZD7
	9	Returned value 7	RO	2 bytes	Transmitted PZD8
	A	Returned value 8	RO	2 bytes	Transmitted PZD9
	В	Returned value 9	RO	2 bytes	Transmitted PZD10

Index (hexadecimal)	Subindex (hexadecimal)	Function	Access permission	Data length	Corresponding to
	С	Returned value 10	RO	2 bytes	Transmitted PZD11
	D	Returned value 11	RO	2 bytes	Transmitted PZD12
	E	Reserved	RO	2 bytes	/
	F	Reserved	RO	2 bytes	/
2001	0	SW	RO	2 bytes	/

Examples

Example 1: To instruct the VFD whose address is 3 to run forwardly, the master station transmits the following SDO command.

COB-ID	Request code	Object	index	Subindex	ubindex Requested data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x603	0x2B	0x01	0x21	0x00	0x01	0x00	0x00	0x00

Example 2: Assume that the address of the VFD slave station is 3, and the function of setting 1 is defined as "Set frequency". To set the frequency to 50.00 Hz (that is, setting 1=0x1388), the master station transmits the following SDO command.

COB-ID	Request code	Object	index	Subindex				
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x603	0x2B	0x00	0x21	0x03	0x88	0x13	0x00	0x00

Example 3: To read the running state of the VFD whose address is 3, the master station transmits the following SDO command.

COB-ID	Request code	Object	index	Subindex		Requested data		
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x603	0x40	0x01	0x20	0x00	0x00	0x00	0x00	0x00

If the VFD is running forward, the following SDO command is returned to the master station.

COB-ID	Request code	Object	index	Subindex		Requested data		
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x583	0x4B	0x01	0x20	0x00	0x01	0x01	0x00	0x00

Example 4: Assume that the address of the VFD slave station is 3, and the function of setting 1 is defined as "Set frequency". To set the frequency to 50.00 Hz (that is, setting 1=0x1388), the master station transmits the following SDO command.

COB-ID	Request code	Object	index	Subindex	Requested data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x603	0x40	0x00	0x20	0x03	0x00	0x00	0x00	0x00

If the running frequency of the VFD is 50.00 Hz, the following SDO command is returned to the master station.

COB-ID	Request code	Object	index	Subindex	ubindex Requested data			
	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x583	0x4B	0x00	0x20	0x03	0x88	0x13	0x00	0x00

3.8 Baud rate and communication address setting

3.8.1 Baud rate setting

After setting the CANopen baud rate and communication address, you need to restart the VFD to enable the settings to take effect.

The CANopen baud rate is set through the corresponding VFD function parameter. For description of function code addresses, see the VFD operation manual. Table 3-21 describes the values of the function parameter and their corresponding baud rates.

Table 3-21 Baud rate setting

Function parameter value	Baud rate (bit/s)
0	1000 k
1	800 k
2	500 k
3	250 k
4	125 k
5	100 k
6	50 k
7	20 k

3.8.2 Communication address setting

The CANopen communication address is set through the function parameter P15.01.

3.8.3 Function codes related to transmitted and received PZD

Table 3-22 Received PZD

Function code	Word	Value range	Default value
P15.02	Received PZD2	0–31 0: Invalid	0
P15.03	Received PZD3	1: Set frequency (0–Fmax, unit: 0.01 Hz) 2: PID reference (0–1000, in which 1000 corresponds to	0
P15.04	Received PZD4	100.0%) 3: PID feedback (0–1000, in which 1000 corresponds to	0
P15.05	Received PZD5	100.0%) 4: Torque setting (-3000-+3000, in which 1000	0
P15.06	Received PZD6	corresponds to 100.0% of the rated current of the motor) 5: Setting of the upper limit of forward running frequency	0
P15.07	Received PZD7	(0–Fmax, unit: 0.01 Hz) 6: Setting of the upper limit of reverse running frequency	0
P15.08	Received PZD8	(0–Fmax, unit: 0.01 Hz) 7: Upper limit of the electromotive torque (0–3000, in	0
P15.09	Received PZD9	which 1000 corresponds to 100.0% of the rated current of the motor)	0
P15.10	Received PZD10	8: Upper limit of the brake torque (0–2000, in which 1000 corresponds to 100.0% of the rated current of the motor) 9: Virtual input terminal command, 0x000–0x3FF	0
P15.11	Received PZD11	(corresponding to S8, S7, S6, S5, HDIB, HDIA, S4, S3, S2, and S1 in sequence)	0
P15.12	Received PZD12	 10: Virtual output terminal command, 0x00–0x0F (corresponding to RO2, RO1, HDO, and Y1 in sequence) 11: Voltage setting (for V/F separation) (0–1000, in which 1000 corresponds to 100.0% of the rated voltage of the motor) 12: AO output setting 1 (-1000–+1000, in which 1000 corresponds to 100.0%) 13: AO output setting 2 (-1000–+1000, in which 1000 corresponds to 100.0%) 14: MSB of position reference (signed number) 15: LSB of position feedback (unsigned number) 17: LSB of position feedback (unsigned number) 18: Position feedback setting flag (position feedback can 	0

Function code	Word	Value range	Default value
		be set only after this flag is set to 1 and then to 0) 19: Function parameter mapping (PZD2–PZD12 correspond to P14.49–P14.59) 20–31: Reserved	

Table 3-23 Transmitted PZD

Function code	Word	Value range	Default value
P15.13	Transmitted PZD2	0–31	0
P15.14	Transmitted PZD3	0: Invalid	0
P15.15	Transmitted PZD4	1: Running frequency (x100, Hz)	0
P15.16	Transmitted PZD5	2: Set frequency (×100, Hz)	0
P15.17	Transmitted PZD6	3: Bus voltage (×10, V)	0
P15.18	Transmitted PZD7	4: Output voltage (×1, V)	0
P15.19	Transmitted PZD8	5: Output current (×10, A)	0
P15.20	Transmitted PZD9	6: Actual output torque (×10, %)	0
P15.21	Transmitted PZD10	7: Actual output power (×10, %)	0
P15.22	Transmitted PZD11	8: Rotating speed of the running (x1, RPM)	0
P15.23	Transmitted PZD12	9: Linear speed of the running (x1, m/s) 10: Ramp frequency reference 11: Fault code 12: Al1 value (x100, V) 13: Al2 value (x100, V) 14: Al3 value (x100, V) 15: HDIA frequency (x100, kHz) 16: Terminal input state 17: Terminal output state 18: PID reference (x100, %) 19: PID feedback (x100, %) 20: Rated torque of the motor 21: MSB of position reference (signed number) 22: LSB of position reference (unsigned number) 23: MSB of position feedback (signed number) 24: LSB of position feedback (unsigned	0

Function code	Word	Value range	Default value
		number)	
		25: State word	
		26: HDIB frequency value (×100, kHz)	
		27: High-order bit of PG card pulse	
		feedback	
		28: Low-order bit of PG card pulse	
		feedback	
		29: High-order bit of PG card pulse	
		reference	
		30: Low-order bit of PG card pulse	
		reference	
		31: Function parameter mapping	
		(PZD2–PZD12 correspond to	
		P14.60–P14.70)	

3.9 Example of communication between CANopen and IVC3

Step 1 Create a project. Open INVT small PLC programming software Auto Station, choose File > New project and then fill in a program name, location, PLC type, and other required information. The interface is shown as follows.

New project	22
Program name	CANopentest
Location	nistrator \Documents \CANopentest \
PLC type	IVC3
Default editor	Ladder chart 👻
Project description	
	OK Cancel

Step 2 Complete the CANopen configuration. Choose **Project manager** > **System block** > **CANopen configuration** to enter PLC master station setting. The interface is shown as follows.

Communication card

tem setting		
Saving Range Output Table	Protocol type: Master V Master Config	
Set Time Input Filter Input Point Advanced Settings Serial Port	Station Number By software By DIP switch Station number 10 (1°53)	
Priority Level Of Intern Communication Modul MDI Config Ethernet Configuration CANopen Config	Baud Rate By software O By DIP switch	
	Baud rate: 250 v Kops Heart beat interval: 1000 ms	

PLC station number and baud rate can be set through the software and dial-up. In this example, PLC station number and baud rate are set to 10 and 250K respectively through the software (by default).

Step 3 Import the EDS file for the slave node. Choose **Master configuration** > **Import** to import the EDS file **EC-TX505.eds** of GD350 series high performance vector VFD, select slave station number, set transmission speed and interval time of synchronous messages, and other information. The interface is shown as follows.

Catalogue		Network			
Import Delete		Del Up D	Nown		
EC-TX505 eds		# Slaves	Supervision		
EC-TX505		1 EC-TX505	NONE		
		2			
		3			
		5			
	_	6			
	>	7			
		8			
		9			
		10			
		12			
		13			
		14			
		15			
		16			
		18			
roperti Value		Parameter			
		Trans Speed	250k	▼ bit/s	
		SYNC COB-II	J. UX00		

In this example, the slave number is 1, baud rate is 250K, and synchronization cycle period is 20ms.

Step 4 Configure the slave PDO data. Each slave station has four received PDOs and four transmitted PDOs. Since each PDO has multiple transmission modes, you can configure the response data and transmission modes according to the actual communication situation. Take transmitted PDO 1, transmitted PDO 2, received PDO 1 and received PDO 2 for example. The interface is shown as follows.

twork Map	ping s	yabol								
Slaves		Available Objects	PD	0			Type:	Send	•	1
# Slaver		(#)-2100						Concession of		
1 EC-TX	505	· 2101	2	Nane	Index COB	ID Tran	s Ty In	habit	Event	Tin
2		6040		Receive F	1600	201	255	500		0
3		-6041		Receive F		301	1	0		0
4		- 6042		Receive F			254	0		0
5		- 6043		Receive F	1603	501	254	0		0
5 6 7		- 6044								
		- 6071								
8		6077								
9										
10										
11										
11 12			Ma	pped Objects						
11 12 13	10			pped Objects						
11 12 13 14			D	el Defa						
11 12 13 14 15				el Defa		Index	Size			•
11 12 13 14 15 16			D	el Defa Nane RPDO	ut	2100			16	
11 12 13 14 15 16 17	10		D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18			D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16 16 16	
11 12 13 14 15 16 17 18 19			D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18 19 20	H		D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18 19 20 21	iii		D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18 19 20 21 22			D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18 19 20 21 22 23	H		D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18 19 20 21 22 23 24			D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18 19 20 21 22 23			D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	
11 12 13 14 15 16 17 18 19 20 21 22 23 24			D	el Defa Nane RPDO CANopen f	ut	2100 2100sub1			16	

Double click one PDO to set the transmission mode, such as **sync** (mode 1–240) and **asyn** (mode 254 and 255). The interface is shown as follows.

PDO		1.00	<u> </u>
Trans Type			
Sync(acyclic)(0)			
Sync(cyclic)(1-24)	0) 1	× SYNC cycle period	
Async(manufactu)	rer specific) (2	54)	
Async(configuration)	ion files) (255)		
Properties			
Inhibit time(0-65535	i):	500 🚔 x 100 ms	
		0 🚔	
Event timer(0-65535	i):	o ms	
ОК		Cancel	
	•	Cancel	

Refer to the description of CANopen communication of GD350 VFD, transmitted PDO

1 only supports 255 transmission mode, and does not support event timer mode. Therefore, received PDO 1 mode is configured as asynchronous 255 mode, and event timer is configured as 0ms, inhibition time is configured as 50ms, that is, transmitted PDO 1 message is sent at most once within 50ms.

Transmitted PDO 2 supports all transmission modes. Generally, it is configured as 254 mode with an appropriate inhibition time. Transmitted PDO 2 message is sent upon data change, but it can be only sent once in each inhibition time so as to use bus resources reasonably. In the following interface, transmission mode is configured as 254, and inhibition time is configured as 50ms.

PDO	5 0	1.00		1	23
	Trans Type Sync(acyclic)(0) Sync(cyclic)(1-240) Async(manufacturer s Async(configuration fil		▲ ▼ SYNC cy (4)	cle period	
	Properties Inhibit time(0-65535):		500	x 100 ms	
	Event timer(0-65535):		0	ms	
	ОК		Cancel		

Received PDOs support all transmission modes. Generally, received PDO 1 is configured as 254 mode with an appropriate inhibition time while received PDO 2–4 are configured as synchronous mode. Different synchronous modes are configured according to real-time requirements of communication data. In the following interface, transmission mode of received PDO 1 is configured as 254, and inhibition time is configured as 50ms.

PDO	-	1.00	1.10	100	8
	Trans Type Sync(acyclic)(0) Sync(cyclic)(1-240) Async(manufactures Async(configuration		▲ ▼ ¥)	ycle period	
	Properties Inhibit time(0-65535):		500	x 100 ms	
	Event timer(0-65535):		0	ms	
	ОК		Cancel		

In the following interface, transmission mode of received PDO 2 is configured as 1, that is, the master transmits received PDO 2 once per sync cycle period. The sync cycle period is configured as 20ms, that is, the master transmits received PDO 2 once every 20ms.

Trans Type Sync(acyclic)(0)	
Sync(cyclic)(1-240)	1 x SYNC cycle period
Async(manufacturer speci	fic) (254)
Async(configuration files)	(255)
Properties	
Inhibit time(0-65535):	0 🔷 x 100 ms
Event timer(0-65535):	0 🖍 ms
,	

Step 5 Perform symbol mapping. Map the configuration data to the internal storage area of the PLC. Choose **Symbol > Reset > Generate global variables**. The interface is shown as follows.

etwo	rk Mappin	ng Symbol					
			First D Element of BFM Areas:	5000	Reset	Gen Global Var	h
2	Type	Slaves	Objects	Sizes	Access		1
1	Send	EC-TX505	TPDO Respond	16	D5000		
1	Send	EC-TX505	CANopen read fuction	16	D5001		
1	Send	EC-TX505	error resp	16	D5002		
1	Send	EC-TX505	TPDO Respond	16	D5003		
1	Send	EC-TX505	return_value1	16	D5004		
1	Send	EC-TX505	return value2	16	D5005		
1	Send	EC-TX505	return_value3	16	D5006		
1	Send	EC-TX505	return value4	16	D5007		
1	Send	EC-TX505	return_value5	16	D5008		
1	Send	EC-TX505	return_value6	16	D5009		
1	Send	EC-TX505	return_value7	16	D5010		
1	Send	EC-TX505	return_value8	16	D5011		
1	Send	EC-TX505	return_value9	16	D5012		
1	Send	EC-TX505	return_value10	16	D5013		
1	Send	EC-TX505	return_value11	16	D5014		
1	Receive	EC-TX505	RPDO	16	D5512		
	Receive	EC-TX505	CANopen fuction	16	D5513		
1	Receive	EC-TX505	CANopen write fuction	16	D5514		
	Receive	EC-TX505	RPDO	16	D5515		
	Receive	EC-TX505	set_value1	16	D5516		
	Receive	EC-TX505	set_value2	16	D5517		
1	Receive	EC-TX505	set_value3	16	D5518		
	Receive	EC-TX505	set_value4	16	D5519		
	Receive	EC-TX505	set_value5	16	D5520		
	Receive	EC-TX505	set_value6	16	D5521		

Step 6 Set VFD function parameters. The parameters are set as follows.

Function code	Setting	Description
P00.01	2	Start/stop through
F 00.01	Z	communication
P00.02	1	CANopen communication
F 00.02	I	mode
P00.06	9	Set frequency through
F00.08	9	CANopen communication
P15.01	1	Communication node
F 15.01	I	number
P15.02	1	Set frequency
P15.13	1	Running frequency
P15.14	3	Bus voltage
P15.15	4	Output voltage
P15.16	5	Output current
D15.07	3	Communication baud rate
P15.27	3	250kbps

The configuration is completed. The data variables corresponding to transmitted PDO 1 are D5000–D5002, data variables corresponding to transmitted PDO 2 are D5003–D5006, data variables corresponding to received PDO 1 are D5512–D5514, and data variables corresponding to received PDO 2 are D5515–D5518.

Of which, D5000 is a request code for reading and writing, D5001 is parameter address,

D5002 is request data. Set D5000=1, indicating a request for reading the value of a parameter. Set D5001=11, namely, P00.11 (Acceleration time) address, indicating that received PDO 1 is to read the acceleration time of slave 1.

Return data D5512=1, indicating that the parameter is read successfully. D5514=400, indicating P00.11 is set to 40.0.

Set D5515=1, indicating that the VFD starts in the forward direction. Set D5516=264, indicating that communication frequency of the VFD is set to 2.64Hz.

Transmitted PDO 2 returns the running state and data regularly, in which the state word is D5003=16#4101 (heartbeat feedback, ready to run, VFD is in forward running), D5004=264 (running frequency 2.64Hz), D5005=5793 (bus voltage 579.3V), D5006=18 (output Voltage 18V), and D5007=0 (output current 0.0A). The interface is shown as follows.

	Element Name	data type	display format	current value	new value
1	D5512	WORD	Decimal		1
2	D5513	WORD	Decimal		11
3	D5514	WORD	Decimal		5
4	D5515	WORD	Decimal		1
5	D5516	WORD	Decimal		264
6	D5000	WORD	Decimal		
7	D5001	WORD	Decimal		
8	D5002	WORD	Decimal		
9	D5003	WORD	Hexadecimal		
10	D5004	WORD	Decimal		
11	D5005	WORD	Decimal		
12	D5006	WORD	Decimal		
13	D5007	WORD	Decimal		

3.10 Example of communication between CANopen and AX70

1. Set parameters of the VFD.

Function code	Setting value	Description
P00.01	2	Start/stop through communication
P00.02	1	CANopen communication mode
P00.06	9	Set frequency through CANopen communication
P15.01	1	Communication node number
P15.02	1	Set frequency
P15.13	1	Running frequency
P15.14	3	Bus voltage
P15.15	4	Output voltage
P15.16	5	Output current
P15.27	3	Communication baud rate 250kbps

2. Open CODESYS V3.5 SP15 Patch 1, click **New project**, select **Templates**, and fill in **Name** and **Location**.

Categories	1	Templates	
	xaries ojects	Empty project HMI project	Standard project w
a project c lame ocation	ontaining one device, one app canopen_test [C:IUsers\Honey\Desktop\ca	lication, and an empty implementation for PLC.	
	C: Users Honey Desktop (ca	inopen_test	~ .

3. Select the device and programming language.

Standard	l Project		×
	objects withi - One progra - A program I - A cyclic tasl	it to create a new standard project. This wizard will cr n this project: mmable device as specified below PC_PRG in the language specified below which calls PLC_PRG to the newest version of the Standard library current	
	<u>D</u> evice	INVT AX7X (Shenzhen INVT Electric Co., Ltd.)	~
	PLC_PRG in	Structured Text (ST)	~
		0	Cancel

4. Click **Tools** in the menu bar and select **System Repository** as shown in the following figure. Click **Install** to import the EDS file.

	System Repositor	'Y			~	Edit Locations
	(C:\ProgramData	<pre>\CODESYS\D</pre>	evices)			
stalled d	evice descriptions					
tring for	a fulltext search		Vendor	<all vendors=""></all>	~	Install
Name		Vendor	Version	Description		
е- 🗊 п	ieldbuses					Export
	MI devices					
е 🔟 Р						
-	LCs oftMotion drives					
-						
-						
-						
-						

5. Right click Device (INVT AX7X) in the Devices pane, and choose Add Device... > CANbus > Add Device.

String for a fulltext search		Vendor	<all vendors=""></all>		
Name Fieldbuses	Vendo	or		Version	Description
CANbus	35 - Sm	art Softwar	e Solutions GmbH	3.5.15.0	Needed for all fie
■ MetX CANbus	35 - Sm	art Softwar	e Solutions GmbH	3.5.15.0	CANbus on a net
<					>
Name: CANbus			only) 🗌 Display (outdated versi	ions
Group by category Displa Name: CANbus Vendor: 35 - Smart Softwa Categories: CANbus Version: 3.5. 15.0 Order Number:			only) 🗌 Display (^	ions

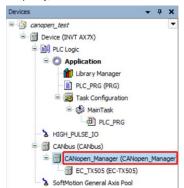
6. Right click CANbus in the Devices pane, and choose Add Device... > CANopen_Manager > Add Device.

tring for a fulltext search	Vendor	<all vendors=""></all>		
Name	Ver	dor	Version	D 1
CANopen Manager	35 -	Smart Software Solutions GmbH	3.5.15.0	C
CANopen Manager	SIL2 35 -	Smart Software Solutions GmbH	3.5.15.0	c
CANopen_Manager_	SoftMotion 3S -	Smart Software Solutions GmbH	3.5.15.0	a
<		Smart Software Solutions GmbH	3.5.15.0	L0 '
Knowski Stranger (Konger), Manager Vendor: 35 - Smart Software Sol Categories: CANopen/Manager Version: 3.5.15.0 Order Number: Description: CANopen Manager			Ż	

7. Right click CANopen_Manager in the Devices pane, and choose Add Device... > EC-TX505 > Add Device.

China for a l	fulltext search	Vendor	<al yendors=""></al>	
Name	Untext sear on	Velidul	<all vendors=""></all>	Vendor
Name	- III CMMP-AS-C5-11A-P3			Festo AG & Co. KG
	- CMMP-AS-C5-11A-P3			Festo AG & Co. KG Festo AG & Co. KG
	- CMMP-AS-CS-3A-M0			Festo AG & Co. KG Festo AG & Co. KG
	DIS-2 SoftMotion	SoftMotion		Pesto AG & Co. KG Metronix GribH
	EC-TX505			sherahen IN/T electronic co.
	ECOSTEP200 SoftMo	No.		Jenaer Antriebstechnik Gmbi
	ECOVARIO 114/214/414 SoftMotion			Jenaer Antriebstechnik Gribb
	- ECONALD 114214/414_30104001			Festo AG & Co. KG
	IBCAN CMZ CANopen node. SoftMotion			CMZ Sistemi Elettronici
	IBDCAN CH2 CANopen node_softMotion			CMZ Sistemi Elettronici
		IBDCAN CM2 CANopen node_softMotion Encoder IBMDCAN Bonfiglioli Vectron MDS GmbH_SoftMotion		
	11 IBMDCAN Bonfigliol V			Bonfiglioli Vectron MDS GmbH Bonfiglioli Vectron MDS GmbH
Group by	category Display all ve	rsions (for experts o	only) 🗌 Display ou	tdated versions
	ne: EC-TX505 dor: shenzhen INVT electronic	un bel		
	egories: Remote Device	L CD. (HD		
	sion: Revision = 16#00000000	, FileVersion=1.1, Fil	e=EC-TX505.eds	
	er Number: 0			~
Des	cription: Imported from EC-T	X505.eds		

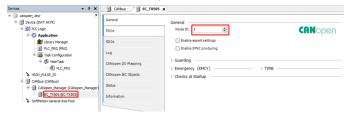
After devices are added completely, the interface is shown as follows.



8. Double click CANbus in the Devices pane to set the baud rate of the network.

Devices 👻 🕈	CANbus X A EC_TXS	05		
canapen_test G Device (INVT AX7X)	General	General		
PLC Logic Application	Log	Network	0	CAN
Library Manager Dec_PRG (PRG)	CANbus IEC Objects	Baudrate (kbit/s)	250 ~	
😑 🧱 Task Configuration	Status			
B 🥵 MainTask	Information			
- & HIGH_PULSE_IO				
CANbus (CANbus)		-		
CANopen_Manager (CANopen_Manager (CANopen_Manager) EC_TXS05 (EC-TXS05)	er)			
SoftMotion General Axis Pool				

9. Double click EC_TX505 in the Devices pane to set the node ID of the slave station.



10. Double click **PDOs** in the above figure, and double click PDO mapping to set the PDO as shown in the following figure.

		Receive PDOs (Master => Slave	:)		Transmit PDOs (Slave => Master)	
		+ Add PDO + Add Mapping	🖋 Edit 🗙 Delete 🕆 Move	Up =	🕂 Add PDO 🕂 Add Mapping 🦼	PEdit 🗙 Delete 🕆 M
PDOs		Name	Object	Bit length	Name	Object
DOs		✓ 16#1400: Receive PDO:	p 16#201 (\$NODEID+16	48	✓ 16#1800: Transmit PDO1	16#181 (\$NODEID+
		read or write	16#2100:16#00	16	CANopenresp	16#2000:16#00
g		CANopen fuction	16#2100:16#01	16	CANopen read fuction	16#2000:16#01
Nooro I	/0 Mapping	CANopen write fuction	16#2100:16#02	16	error resp	16#2000:16#02
			2 p 16#301 (\$NODEID+16	64	✓ 16#1801: Transmit PDO2	16#281 (\$NODEID
Nopen I	EC Objects	Operation Command	16#2101:16#00	16	Operation Statusword	16#2001:16#00
tus		set_value1	16#2100:16#03	16	return_value1	16#2000:16#03
cus		set_value2	16#2100:16#04	16	return_value2	16#2000:16#04
ormation	n	set_value3	16#2100:16#05	16	return_value3	16#2000:16#05
			8 p 16#401 (\$NODEID+16	64	✓ 16#1802: Transmit PDO3	16#381 (\$NODEID
		set_value4	16#2100:16#06	16	return_value4	16#2000:16#06
		set_value5	16#2100:16#07	16	return_value5	16#2000:16#07
		set_value6	16#2100:16#08	16	return_value6	16#2000:16#08
		set_value7	16#2100:16#09	16	return_value7	16#2000:16#09
		and the second second second second second	p 16#501 (\$NODEID+16	64	✓ 16#1803: Transmit PDO4	16#481 (\$NODEID
		set_value8	16#2100:16#0A	16	return_value8	16#2000:16#0A
		set_value9	16#2100:16#0B	16	return_value9	16#2000:16#08
		set_value10 set_value11	16#2100:16#0C 16#2100:16#0D	16	return_value10 return_value11	16#2000:16#0C 16#2000:16#0D
	PDO Prope	rties				×
	PDO Prope	rties	\$NODEID+16#1	80		X
	COB ID		= 16#181 (385)			
				80		
	COB ID	e (x 100µs)	= 16#181 (385) 300	•	file specific (Type 25:	RTR
	COB ID Inhibit time	e (x 100µs) ontype	= 16#181 (385) 300	•		RTR

11. Double click CANopen I/O Mapping and select Enabled 2 (always in bus cycle task).

	Variable	Mapping	Channel	Address	Type	Unit	Description	
200s	· · · ·	mapping				onit	Description	
	8-59		read or write	%QW22	UDNT			
SDOs			CANopen fuction	%QW23	UINT			
.09	8.10		CANopen write fuction	%QW24	UINT			
.09	*- * +		Operation Command	%QW25	UDNT			
CANopen I/O Mapping	÷.*•		set_value1	%QW26	UDNT			
	÷-*•		set_value2	%QW27	UINT			
CANopen IEC Objects	* *		set_value3	%QW28	UDNT			
	* *		set_value4	%QW29	UDNT			
Status	÷.*•		set_value5	%QW30	UINT			
information			set_value6	%QW31	UDNT			
an ormation	* *		set value7	%OW32	UDIT			
	a. 14		set_value8	%QW33	UDIT			
	* *		set_value9	%QW34	UDNT			
			set_value10	%QW35	UDNT			
			set value11	%OW36	UDNT			
	B-10		CANopen resp	%IW2	UDNT			
	8.9		CANopen read fuction	%IW3	UDNT			
	8-10		error resp	%EW4	UINT			
	8.49		Operation Statusword	%IW5	LUNT			

12. Double click Device (INVT AX7X) in the Devices pane, choose Scan Network..., and choose the PLC.

canapen_test						
Topyce (INVT AX70	Communication Settings	Scan Network Gateway	Device •			
Device (DVT AX70 PLC Logic						
B Application	Applications					
Library Manager	Backup and Restore			·		
PLC_PRG (PRG)				100 000 000 000	**	
B 😹 Task Configuration	Files			Gateway	•	
🖹 🥵 MainTask	Log					
- @ PLC_PRG - ℃ HIGH_PULSE_IO	PLC Settings		Gateway-1	Ý	DESKTOP-2F6L015	
HIGH_POLSE_IO CANbus (CANbus)	PCC Settings		IP-Address: localhost			
CANopen_Manager (CANopen_Manager)	PLC Shell		Ports			
EC_TX505 (EC-TX505)	Users and Groups		1217			
SoftMotion General Axis Pool	Osers and droups					
	Symbol Rights					
				Device Address: 0301.800A Block driver:		
				UDP		
				Number of		
				channels:		
				4		
				4		
				4 Serial number:		
				4		
				4 Serial number: 807E113D0860		
				4 Serial number: B07E113D0B60 Target ID:		
				4 Serial number: 807E113D0860		
				4 Serial number: B07E113D0B60 Target ID:		

13. Click the Compile icon in the toolbar.

	Window						
গ গ	4 🔒	8 .	Application [Device: PLC Logic]	- 05 0 ğ	Þ	а,	Ľ

Make sure that there is no error.

Messages - Total 0 error(s), 0 warning(s), 0 message(s)						-	
Build	-	O error(s)	0 warning(s)	🚯 0 me:	sage(s)	X	ś.
Description	L				P	oject	Object
Build started: Application: Device.Application							
Typify code							
Compile complete 0 errors, 0 warnings							

14. Click the Login icon.

	Window					
ণা শ	1	‱ • ⊡`	Application [Device: PLC Logic]	- <mark>05</mark> 09	Þ	Ľ

Click Yes.

CODESYS		×
Warnin there is replace	ng: An application 'Application' is currently in RUN mode on the s no matching compile information, this existing application ne ed.	PLC. As eds to be
Click 'Y	es' to download the latest code or 'No' to abort.	
	Yes No	Details

15. Click the Run icon.

Tools	Window	Help			
위 케	M 🛱	🏜• 🖻	Application [Device: PLC Logic]	- 0; 0 ;	▶ = ⅔

The normal operation of the device is shown as follows.



16. Open **CANopen I/O Mapping** in the EC_TX505 page, modify the parameters of the VFD, and view the status of the VFD.

General	Find		Filter Show all			• 🗣 Add FB fo	r IO Channel *	Go to	Instance
PDOs	Variable	Mapping	Channel	Address	Type	Current Value	Prepared Value	Unit	Description
1005	8.50		Operation Command	%QW25	UINT	0			
SDOs	8.40		set_value1	%QW26	UINT	0			
	8.50		set_value2	%QW27	UDIT	0			
Log	8.70		set_value3	%QW28	UDVT	0			
CANopen I/O Mapping	* **		set_value4	%QW29	UDVT	0			
contraction of constants	× *o		set_value5	%QW30	UINT	0			
CANopen IEC Objects	8 10		set_value6	%QW31	UINT	0			
	* **		set_value7	%QW32	UINT	0			
Status	8.50		set_value8	%QW33	UDVT	0			
Information	* *		set_value9	%QW34	UDNT	0			
	8.70		set_value10	%QW35	UINT	0			
	8-10		set_value11	%QW36	UINT	0			
	* *		CANopen resp	%IW2	UINT	0			
	a **		CANopen read fuction	%ZW3	UINT	0			
	8.30		error resp	%ZW4	UINT	0			
	· · · ·		Operation Statusword	%ZW5	UINT	16643			
	· · · · ·		return_value1	%IW6	UINT	0			
	·····		return_value2	%IW7	UINT	0			
	18 - 19		return_value3	%IW8	UINT	0			
	* *		return_value4	%EW9	UDVT	0			

Chapter 4 BACnet MSTP communication card

4.1 Overview

As a BACnet slave, it can realize operations such as the reading and writing of VFD process data and function codes, reading of VFD status words, and writing of VFD control words. The application layer supports setting 32 analog value objects and supports host controllers such as Yet Another BACnet Explorer (YABE) and PLC controller.

4.2 Features

In market demand, BACnet MS/TP is a data communication protocol used primarily in the building automation and HVAC (heating, ventilation and air conditioning) industries. The protocol supports devices such as fans, pumps and ventilation units to establish communication with PLCs. This facilitates a high level of automation in buildings.

The physical transmission medium for the bus is twisted pair (RS485 compliant), two-wire cable or fiber optic cable. The baud rate ranges from 9.6 kbit/s to 115.2 kbit/s. The maximum length of the bus cable is in the range of 100 m to 1200 m, depending on the selected transmission rate. A maximum of 31 nodes can be connected to the same network segment when repeaters are not used, while the number of nodes connected to the network (including repeaters and master nodes) can be increased to 127 when repeaters are used.

Cat	egory	Specifications
	Feature	 Supports the BACnet protocol and BACnet MSTP devices. Provides one BACnet MSTP port and supports half-duplex operations of 115.2kbps. Supports the line, star, and daisy chain connection network topologies, with the number of slave nodes up to 31. Supports timeout detection.
Main functions	Service	Single-property reading service Multi-property reading service Single-property writing service Multi-property writing service I-Am service I-Have service Device communication control service Device re-initialization service
	Function	 Supports the setting of 32 analog objects. Supports the reading and writing of VFD process data and function codes, reading of VFD status words, and writing of VFD control words.

Table 4-1 BACnet MSTP card technical specifications

Cat	egory	Specifications				
		 Supports host controllers such as YABE and PLC. 				
Input power		Power voltage of 24V, current of 200mA, and maximum power of $4.8 \ensuremath{W}$				
	Installation method	GD350 option, inserted to the slot				
Others	Running environment temperature	-10 – +50 °C				
	Storage temperature	-20 – +60 °C				
	Relative humidity	5%–95% (No condensation)				
	Distance	Up to 100 m (cable of shielded twisted pair or with magnetic ring) from the controller				

4.3 Electrical wiring

1. Node selection

A node address is the unique address of a device on the bus, and the address number of a node is set by the function code P15.01.

2. Bus terminator

Each segment has a bus terminator at the head and at the tail to ensure error-free operation. The switches on the PCB are used to switch on the bus terminators, which prevent signal reflection at the bus cable end. If the communication card is the last or first module in the network, the bus terminator must be set to ON. When a D-sub connector with a built-in terminator is used, the communication card terminator must be disconnected.

Item	Description
Number of ports	1
Isolation	1.5kV electrical isolation
Standard	TIA/EIA-485
Communication speed and duplex	0.96k-115.2k baud (protocol depended), half-duplex
Connector type	Swappable, 3-bits, screw-type wiring board
Cable type	Cable of shielded twisted pair or with magnetic ring
Cable length	100 m
Topology	Line, star, and daisy chain connection
Unit load	1/8

Table 4-2 RS485 hardware specifications

4.4 Terminal wiring

4.4.1 DS-P-1004 control terminal layout

Table 4-3 BACnet MSTP card terminal layout

485+	485-	PGND	PE	

4.4.2 DS-P-1004 control terminal functions

Table 4-4 BACnet MSTP card terminal functions

Signal	Port	Terminal function description
485+		Positive differential signal
485-	RS-485	Negative differential signal
PGND	/	Signal ground
PE	/	Earth

Table 4-5 Indicator functions

Indicator	Definition		Function	
LED1	Status indicator	control board. Blinking periodically the control board (th 0.5s).	r: The expansion ca ne period is 1s, on f	a connection with the rd is properly connected to or 0.5s, and off for the other ed from the control board.
		Off	Not powered or in fault state	Not powered or in fault state
LED2	Run indicator (Green)	Blinking periodically (on-off at a 0.5s interval) On	Online. Waiting to receive BACnet data frames BACnet communication state	Waiting to receive data Received BACnet data frames
		Off	No fault	No fault
LED3	Fault indicator (Red)	Blinking periodically (on-off at a 0.5s interval for twice and 2s off) Blinking	Faulty	Duplicate address. The VFD keypad reports E-bAC.
LED3	indicator	Blinking periodically (on-off at a 0.5s interval for twice and 2s		Duplicate address. VFD keypad re

Indicator	Definition		Function	
		periodically (on-off at a 0.5s interval for three times and 2s off)		received in specified time. (Timeout detection must be enabled, that is, the timeout time cannot be 0). The VFD keypad reports E-bAC.
LED4	Power indicator	It is on once the co card.	ntrol board feeds	power to the communication
		Off	No transmission	The device does not transmit data on the RS485 network.
LED5	485TX indicator (Green)	Blinking	Transmitting	The device is transmitting data on the RS485 network.
		On	Transmitting	The device is transmitting data fast on the RS485 network.
		Off	No receiving	The device does not receive data on the RS485 network.
LED6	485RX indicator (Green)	Blinking	Receiving	The device is receiving data on the RS485 network.
		On	Receiving	The device is receiving data fast on the RS485 network.

4.5 Connecting the BACnet MSTP card to the PLC

4.5.1 Bus communication networking

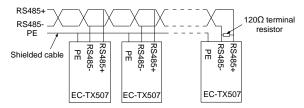
4.5.1.1 Communication packet structure

The communication data frame structure (PKW+PZD) of BACnet MSTP is similar to that of PROFINET. See the following figure.

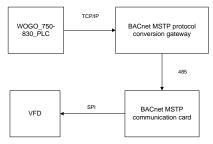
•	Para	meter ion (PKW)	Fixed	P ◀	rocess d (PZD) Free all zor	ocation	
PKW1	PKW2	PKW3	PKW4			PZD3 PZD3		PZD12 PZD12

4.5.1.2 Networking topology

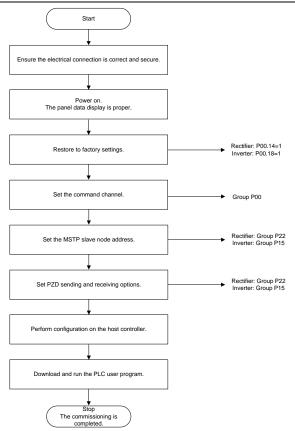
The BACnet MSTP communication card adopts standard 4-pin terminal interface, which can be used in line network topology, star network topology and daisy chain topology, and its electrical wiring diagram is shown in the following. BACnet MSTP cards form a communication network with the PLC by inserting a BACnet MSTP communication card into each unit.



Due to the constraints, the connection method currently used is to add the BACnet MSTP protocol conversion gateway between the PLC and the communication card. See the following figure.



The BACnet MSTP networking commissioning flowchart is as follows.



4.5.2 Configuring the BACnet MSTP protocol conversion gateway on the Sunfull PLC

 Open X2BACnet, select the New Driver option in the menu bar of the host controller configuration interface, as shown below.

-	ACnet - Demo.x2b							×
File (Ed	dit View Tools BA	Cnet Server	Web Serve					
	New Channel New Device New Group New Tag	Ctrl+D	m ID	Register Type	Register	DataType	Value	
	Cut Copy Copy Tag Name Paste Delete	Ctrl+X Ctrl+C Ctrl+V Del						
	Batch Modify Properties							
		*						

In the displayed window, select **BACnetMSTP1**. (For every COM port that is enabled to collect MSTP, an MSTP drive needs to be created. If two buses are used and there are COM1 and COM2, you can create BACnetMSTP1 and BACnetMSTP2 to correspond to COM1 and COM2. Create BACnetMSTP1, BACnetMSTP2, BACnetMSTP3 and BACnetMSTP4 in total to correspond to 4 COM ports, as shown in the following figure.

ſ	K2BACnet - Demo.x2b					- 0	23
	File Edit View Tools BACnet Sen	ver Web Server G	ateway <u>H</u> elp				
	□ ☞ 🖬 🗊 🧖 🗄 💣 🗇 🖆	š 🖻 🖻 🗙 🐞	▶ = 🛠 🛱 👯				
		Item ID	Register Type	Register	DataType	Value	Qu
	Driver Properties				×		
					_		
	Driver List						
	BACnetWSTP1		-	OK			
1	BACnet#STF2 BACnet#STF3		<u>^</u>	Cancel			
	BACnet#SIFS BACnet#SIF4 BACnetBouter				- 1		
	BHHB 212Seria	1			- 1		
	BIGBANIAS_AIB BOSCH_IST400X	I		_			
	BOSCH_IS7400X BOSCH_WISY_UD Buhler_WEAFC1	I_TCP PServer					
	CDT Slave						
	CHANG_AI_CI_P ChengDu Weath	CS6 er					
	cibsolar CA3 Clipsel CBus						
	Condair PH CBEATECH CA A	27					
	Curtain_Wotor DAIKIN DSCH-B	485					+
	I Monitor Mode:Remote DAIKIN HIS			www.opcmast	er.com 13564	4889340 sup	por /

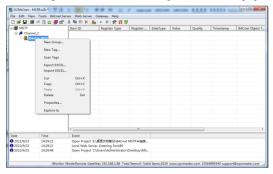
 After drives are added, choose to create channels. As serial communication, the default BACnet MSTP channel serial port is COM1, and the other default communication parameters are as follows: 38400bps baud rate, 8 data bits, 1 stop bit, no parity bit, 50ms timeout time (the timeout time does not need to be modified in most cases), and master node MAC address of 127, which is not recommended to modify. See the following figure.

Serial Port Communication Configuration
Channel Name
Communication Parameters
Port: [0011] • Baud Rate: 38400 •
Data Bits: 8 V Stop Bits: 1 Parity: None V Flow Control: None
Response Timeout: 50 ms Source ID: 127
OK Cancel

After the channel is created, choose **New Device**, set the device properties in the pop-up dialog box, including the device ID. You can query which devices are online on the bus through BACnetScan automatic scanning. The request frame interval (**Delay Between Polls**) is 50 milliseconds. See following figure.

Device Properties	×
Name: Device_1	
Device ID: 1	
Delay Between Polls: 50	ms
Delay After Write: 50	ms
-Bulk Transfer	
Analaog Adjacent Span: 0	
Analaog Max Span: 16	_
Binary Adjacent 0	
Binary Max Span: 2	
OK Cancel	

3. Choose Scan Tags to scan the device for all tags. See the following figure.

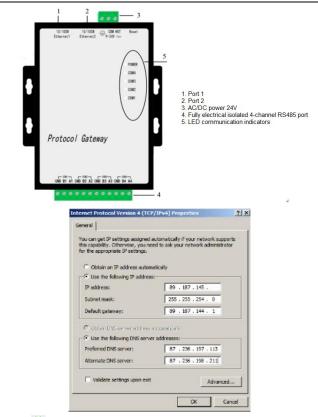


X2BACnet connects to the BACnet MSTP card through RS485. Add all scanned records, and save the project. See the following figure.

dSTP Bern 10		Register Type	Register	Datelype	Value	Quality.	Tesestang	BACnet Object T	\$4Cret Instance	Update C.,	Description
COM2 COM2		AVGraing Value!	0	Nowt		Gecartain.		AN	0	0	CONTROL WORK
Dutes 1	LRECIVE	AV(Araing Value)	28	float		Lincertain.		£14	28		PRINT_RECIVE
Concert Concert	1 SEND	AVIAraiog Valuei	24	float		Decertain		AV	24		PRIVIL SENIO
	2.RECEVE	AV(Zealog Value)	29	Roat		Uncertain		85/	29	0	PRINZ, RECEVE
(Corner)		AV(Araing Value)	25	Ploat		Uncertain		8.17	25	0	PKIN2_SEND
	R. RECIVE	AV(Araing Value)	30	Float		Uncertain.		811	35	0	PRIND, RECIVE
COLOR OF CHILD		AV(Analog Value)	28	Host		Uncertain		#1/	26	.0	PR083_52ND
	8,4\$CIVE	AV(Xealog Value)	25	Roat		Uncertaire.		80	31	.0	PRIMA, RECEVE
(CPCN)		AVGraing Value1	27	Now		Gecertain		AN .	27	.0	PRIMA_SEND
	ID, RECIVE	AV(Arelog Value)	21	Float		Uncertain.		#N	21		P2D107_RECIVE
2 9200		AV(Aralog Value)	9	Roat		Uncertain		XX		0	F2DD0_SEND
	LI.RECIVE	AV(Zealog Velce)	22	Reat		Gecertain		310	22		FEDILL RECTVE
2 /200		AV(Arating Value)	10	Rowt		Uncertain		#24	20	0	F2D11_SEND
	12,RECIVE	AV(Analog Value)	23	Roat		Uncertain		#37	23		FEDG2, RECIVE
S 9200		AV(Aralog Value)	22	Float		Uncertain		#N	22	10	P2012_SEND
10 PZ02	RECIVE	AV(Acalog Value)	13	Rost		Uncertain		AV	13	. 6	FED2, AECIVE
de azoc	SEND	AV(Araing Value)	1	float		Uncertain		811	1	0	FED2_SEND
C PZD1		AV(Analog Value)	24	Rost		Decertain		#10	. 54		F2D3,RECIVE
1 9200		AV(Analog Yake)	2	float		Uncertain		81	2		PEDI3_SEND
10 0204	ABCIVE	AV(Analog Value)	15	float		Uncertain		AV	15		FED4 ALCIVE
SP204		AV(Analog Value)	3	float		Uncertain		AV	4	.0	P2D4,SEND
B 7206		AV(Analog Value)	10	Float		Uncertain		AU	26	.0	FEDS_SECIVE
6 9205	SENO	AV(Leains Value)	4	Float		Uncertain		EN .	4		PEDS SEND
1.00											

 The PC port connects to port 2 of the protocol gateway 192.168.1.88. COM2 connects to the RS485 port of the BACnet MSTP card. Set the PC port properties. See the following figure.

Open the X2BACnet host controller and import the saved project.



Click ^{‡‡} to upload the project to the gateway. Enter the user name and password in the pop-up dialog box. The user name is fixed to **admin**, and the password is fixed to **admin123456**. Then you can set the system parameters related to the gateway. See the following figure.

jie <u>E</u> dit <u>V</u> iew <u>⊺</u> ools BACnetSe						
D 🗳 🖬 🗊 🕫 🗒 💣 🚳 😭	👃 🐘 🖹 🗙 🦓	▶ = 🛠 🙀 !	1			
ModbusRTU	Item ID	Register Type	Register	DataType	Value	
- St Chann Login		- X -	n i			

	Cazo	a	J			

You can choose the monitoring mode under **Monitor Mode** in the toolbar, or double click **Monitor Mode** in the bottom status bar to switch the mode. To use gateway monitoring, you need to switch the monitoring mode to gateway mode. Configure the project in X2BACnet and upload it to the hardware gateway to realize the protocol conversion function through the hardware gateway. Click **>** to start monitoring on the X2BACnet software to monitor the communication status of the hardware gateway simultaneously. See the following figure.

B # COM2		Bree ID	Register Type	Revister	OstaTune .	Value	Quilty	Terretares	RACret Object 7	RdCost Instance .	Undate C.	Description
		CONTROL WO.		0	Finat	0.000000			AV	A CONTRACTOR OF	1	CONTROL WORD
		PENT RSCNT	AVGraing Valuel	28	Float	0.000000	Good	2022-06-151-	AV AV	28		PKW1 RECIVE
		PROVI SEND	AVGAralog Valuel	24	Deat	0.000000	Good	2022-06-151	AV	24		PKW1_SEND
		PRINT RECENT	AVIAraioo Valuei	29	float	0.0000000	Good	2022-06-111-	AV	28	2	PKW2 BECTVE
		PRINZ SEND	AVIAraiog Valuel	25	finat	0.000000	Good	2022-06-151	202	25	-	PKW2 SEND
		PRIME RECIVE	AV(Aralog Value)	10	Reat	0.500000	Gand	2022-06-157-	414	10	2	PRW3.RECIVE
		C PRIVE SEND	AV(Aralog Value)	16	Reat	0.500000	Good	2022-06-157	AV	26	1	PKW3 SEND
		PRIME RECEVE	AV(Araing Value)	11	Float	0.0000000	Good	2022-06-117-	AV.	11	-	PKW4 RECIVE
		PRIME SEND	AVIAraiog Valuei	27	Elnat	0.0000000	Good	2022-06-157	411	22		PKW4 SEND
		C PZDID RECTVE	AVIAnalog Valuel	11	finat	0.0000000	Gand	2022-05-111-	AV	23		PZD10_RDCIVE
		P2DID SEND	AVIAnalog Valuel		Roat	0.500000	Good	2022-06-157-	AV	4		PZD10 SENO
		F2D11 RECIVE	AVIAnelog Valuel	22	Reat	0.000000	Good	2022-06-157-	49	22		PZD11 RECIVE
		C P2D11 SEND	AV(Araing Value)	10	Float	0.0000000	Good	2022-06-157	AV	10	1	PZD11 SEND
		P2D12 RECIVE	AV(Aralog Value)	23	float	0.000000	Good	2022-08-157.	AV	23		PZD12 BECIVE
		PZD12 SEND	AVIAraioo Valuei	11	float	0.0000000	Good	2022-06-117_	201	12	1	PZD12 SEND
		PEDE RECIVE	AVIAralog Valuel	10	Finat	0.0000000	Gand	2022-06-137_	417	13	1	PZD2.RECTVE
		P2D2 SEND	AVGAralog Valuel	1	Reat	0.500000	Gand	2022-06-157-	AV	1		FZD2 SEND
		C PZDB RECIVE	AVIAnalog Value)	14	Boat	0.0000000	Good	2022-06-157	49	14	1	PZD3 RECIVE
		P2DR SEND	AV[Asaino Value]	2	Roat	0.0000000	Good	2022-06-157	44	2	1	PZD3 SEND
		COLUMN RECTOR	AVGAralog Valuel	15	Deat	0.000000	Good	2022-06-157	AV	15	1	PZD4_RECTVE
		PZD4 SEND	AVIAraiog Valuel		ficat	0.0000000	Good	2022-00-117-	AV	1	1	FZD4 SEND
		P2D5 RECIVE	AVIAralog Valuel	15	float	0.0000000	Good	2022-06-137_	40	18	1	PZD5 RECIVE.
		C P2DS SEND	AV/Analog Valuel	4	Rowt	0.000000	Good	2022-06-151.	AV	4	1	PZD5 SEND
late	Time	Event			1							
2022/6/15	1050-15	MSTR COM2 Desig	# 1/DO 55 FF 00 02	48.00.00.01								
2022/4/15	105015		# 1,(T0) 55 FF 00 03									
2022/6/15	10.50.15	MSTR.COM2/Devic	# 1.(TO) 55 FF 00 03	68 00 00 03								
2022/6/15	105015		* 1/TO 55 FF DO 02									
2022/0/15	10.50.15	MSTR.COM2.Desk	1.CTX0 55 FF 00 02	68 00 00 03								
2022/6/15	10.90.15		14_1.(TH) 55 FF DD D2									
2022/6/15	10:50:15	MSTP.COM2.Devic	e 1.(TO) 55 FF 00 02	68 00 00 03								
2022/6/15	105015		* 1.(TX) 55 FF 00 03									
2022/6/15	105015	METRICOM3 Deale	# 1,(TO) 55 FF 05 02	48 66 56 28								

The configuration is completed.

4.5.3 Configuring WOGO 750-830 PLC

Connect the PLC to your PC with a network cable. Set your PC IP address (such as 192.168.1.250) on your network. Connect the PLC to the PC USB port with a burn-in cable for setting the PLC IP address (such as 192.168.1.1)

4.5.3.1 Configuring the PLC (750-830) IP address through the WAGO Ethernet Settings

 Open the WAGO Ethernet Settings software, click on Settings in the upper right corner, and then choose Communication, as shown in the following figure.

K	WAGO Et Version 5.1	hernet Settings .2				®
S. Exit	Identify	Frite Restart	Default Format	Extract	Se <u>t</u> ting	*
750-8	30, WAGO BACne	t/IP-FBC		-	Q Cor	mmunication
dentif	ication TCP/IP	Network Identification	Date and Time MODBUS	SNTP Stat		guage
	Item Number	750-830			Sho	w <u>M</u> essages
	Description	WAGO BACnet/IP-FBC				
	SW Version	04. 03. 12 (04)				
	HW Version	06				
	FWL Version	FBK V01.00.01 IDX-02				
	Serial Number	SN20140304T130116-052274	15#PFC 0030DE0861BF			
	MAC address	0030DE0861BF				
	IP address	192.168.1.1 (Static	Configuration)			
Ready			40	COM7: 19200, E, 8	, 1	

 Select Serial Ports from the Connection drop-down list box, select the corresponding port number from the Port drop-down list box, and select the corresponding serial port number (which can be viewed in by choosing Device Manager > Ports on your PC), as shown in the following figure.

	GO Ethe						
Exit Ide	Please sele	ication Co ct the kind of associated pa	connection a	nd 🚺	A		Seiting
750-830, TAG dentification	OK (Cance	al Def	k ault	Test	() Help	atus
Item Nu	Connect		(COM, USB, 8	kuetooth)			<u> </u>
Descrip			COM, USB, B	luetooth,)		0	
SW Vers	Settings	Port:					
HW Vers		COM7: WAG	30 USB Servic	e Cable	•	2	
FWL Ver		Baud rate:	Parity:	Data bits:	Stop bits:		
Serial 1		19200 ~	Even ~	8 +	1 *		
MAC add							
IP addr		Timeout [s]:		10 0	rout (s):		
	-						

3. Click Identify, and wait for a while. If the PC and PLC are connected properly, the area in the red box as shown in the following figure will appear, showing information such as the PLC hardware and software versions, MAC address, and IP address. If the content in the red box area indicates that the connection has failed, check whether the connection between the PC and PLC is normal, as shown in the following figure.

K	Version 5.1	hernet Settings .2		
S. Exit	1 Identify	Frite Restart	Default Format	🔓 🍣 Extract Setting
	30, WAGO BACm			
2		Network Identification	Date and Time MODEU	S SNTP Status
۳	Item Number	750-830		
	Description	WAGO BACnet/IP-FBC		
	SW Version	04.03.12(04)		
	HW Version	06		
	FWL Version	FBK V01.00.01 IDX-02		
	Serial Number	SN20140304T130116-05227	45#PFC 00300E0861BF	
	MAC address	00300E0861BF		
	IP address	192. 168. 1. 1 (Statio	Configuration)	

4. Click TCP/IP and set the PLC IP address in red box ②, and ensure the PC IP address and the PLC IP address are in the same network segment. Then click Write to write the set IP address to the PLC. See the following figure.

🎉 WAGO Ethernet Settings *	
WAGO Ethernet Settings Version 5.1.2	
Exit Identify	efault Format Extract Settings
750-830, WAGO BACnet/IP-FBC	
Identification TCP/IP Network Identification Da	te and Time MODBUS SNTP Status
132 . 100 . 1	mitcally from
Profess	<u>uateway:</u> 0 . 0 . 0 . 0 ed DNS-Server: 0 . 0 . 0 . 0
	ve DNS-Server: 0 . 0 . 0 . 0
Ready	^{су} г. сон7: 19200, Е. В. 1

- 5. After the writing is successful, open CoDeSys V2.3.
 - A. Use CoDeSys V2.3 to configure the PLC.

CoDeSys V2.3 has the same functions with WAGO-IO-PRO V2.3.

a) Open the CoDeSys software, click File-New to create a new document. The following window is displayed. Select the PLC model WAGO_750-830 from the drop-down list box, and click OK. See the following figure.

CoDeSys - (Untitled)*		
File Edit Project Insert Extras Online	Window Help	
266 000000		
Tanget Setting Contpartor	Non Non VADD_TO_ES Non	Canad

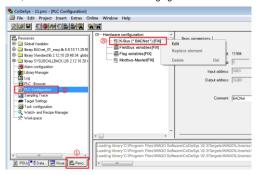
 b) Click the General tab, choose Load boot project automatically, and click OK. See the following figure.

et Platform Memory Layout General	Network functionality Visualization	1
/O-Configuration		
🔽 Configurable	Download as file	
	No address check	
Support preemptive multitasking	Download symbol file	VAR_IN_OUT as reference
Single task in multi-tasking	Symbol config from INI file	Initialize inputs
Byte addressing mode	PLC Browser	Load boot project automatically
	✓ Trace	C SoftMotion
Online Change	Cycle independent forcing	🗖 Save

c) Create a POU program. Since the PLC must contain a PLC_PRG program, and keep the default settings and click **OK**. See the following figure.

New POU		×
Name of the new POU:	PLC_PRG	OK
Type of POU	Language of the POU	Cancel
Program	СL	
C Function Block	C LD	
C Function	C FBD	
Return Type:	O SFC	
BOOL	In ST	
	C CFC	

d) Configure the PLC hardware. Select PLC Configuration from the Resource drop-down list box. At the right, choose Hardware configuration, right-click K-Bus, and choose Edit. See the following figure.



e) Click the Input/Output tab in the Configuration dialog box. Click the Q icon in the toolbar to scan the devices on the PLC hardware bus. When the scan is completed, the devices in red box ③ will appear. Click on any device. The corresponding device address, data type and function description will appear in the right window. Select PLC, PLC from PI Assignment in the PI allocation tab, indicating the PLC controlled IO devices. If other types are selected, the IO devices are communication controlled. See the following figure.

	😅 🖬	। 🤉 🖻 🔍 🕯	4 4 3 1	X 🕇 🖡	750-9999	/500-002 - 2	DO Gen	eric	
P It		Description		Comment	Name	Address	Туре	Comment	
	0-9999/500-002 0-9999/400-002	2 D0 Generic 2 DI Generic				%QX0. 0 %QX0. 1	BOOL	Ch_1 Digita Ch_2 Digita	l outpu l outpu
. 10		b bi senerre							1 carps
* [,			m	08	Cana
Configu	ration		_	,	×		m	08	Can
Configu		tion 1		,					
Configur	ation stput PI allocat						er 14.	1 1 0	
Configur nput / O Pos.	ation atput PI allocat Item Number		Descriptio	'n		PI Assignment	Comme	1 1 0	
Configur nput / On Pos. 1	ation atput PI allocat Item Number 750-9999/500-00	02	2 D0 Gener	m		PI Assignment	Come	11 Ef Q	
Configur nput / O Pos.	ation atput PI allocat Item Number	02		m		PI Assignment	Come	11 Ef Q	
Configur nput / On Pos. 1	ation atput PI allocat Item Number 750-9999/500-00	02	2 D0 Gener	m	E	PI Assignment LC, PLC	Come	11 Ef Q	۹, -
Configur nput / On Pos. 1	ation atput PI allocat Item Number 750-9999/500-00	02	2 D0 Gener	m	E	PI Assignment AC, PLC ACNet, BACnet lodbus RTU, Mo lodbus TCP/UD	Come	11 Ef Q	۹, -
Configur nput / On Pos. 1	ation atput PI allocat Item Number 750-9999/500-00	02	2 D0 Gener	m	E	PI Assignment AC, PLC ACNet, BACnet lodbus RTU, Mo lodbus TCP/UD	Come	11 Ef Q	۹, -
Configur nput / On Pos. 1	ation atput PI allocat Item Number 750-9999/500-00	02	2 D0 Gener	m	E	PI Assignment AC, PLC ACNet, BACnet lodbus RTU, Mo lodbus TCP/UD	Come	11 Ef Q	۹, -
Configur nput / On Pos. 1	ation atput PI allocat Item Number 750-9999/500-00	02	2 D0 Gener	m	E	PI Assignment AC, PLC ACNet, BACnet lodbus RTU, Mo lodbus TCP/UD	Come	11 Ef Q	۹, -
Configur nput / On Pos. 1	ation atput PI allocat Item Number 750-9999/500-00	02	2 D0 Gener	m	E	PI Assignment AC, PLC ACNet, BACnet lodbus RTU, Mo lodbus TCP/UD	Come	11 Ef Q	۹, -

If no device is scanned, check the following:

- Whether the PLC is properly powered.
- Whether the PLC and PC are connected properly and are in the same network segment.
- f) Configure tasks.

Create a task.

😓 CoDeSys - Temp.pro* - [Task configuratio	n]			
Eile Edit Project Insert Extras On	line Window Help			
DC - 00 - 0	2			
Renorme R	D J J Tax configuration	Insert Element Append Task Cut Copy Paste Delete Set Debug Task Enable / disable task	Ctrl+X Ctrl+C Ctrl+C Ctrl+V Del	•
└℀ Workspace	•		,	

Set the circular interval to 30ms and append a program. See the following figure.

	System events NewTask Insert Task Append Program Call		NewTask 1
	Cut Copy Paste Delete Set Debug Task	Ctrl+X Ctrl+C Ctrl+V Del	seing d by gvent d by egternal event
•	Enable / disable task	Watchdog	(e.g. t#200ms) [T#30ms] [T#30ms]

Select and add the program PLC_PRG. See the following figure.

B → Back configuration System events B → Back NewTask	Program Call	1
Input assistant		
	User defined Programs	

g) Import the BACnet library file.

Copy all files from **00_Libraries_BA** to the **Building** folder in the CoDeSys installation directory, for example, *C:\Program Files\WAGO Software\CoDeSys V2.3\Targets\WAGO\Libraries\Building*.

After the copy is complete, choose Library Manager in the Resource tab in the CoDeSys main interface. Right-click on the blank area marked as ③, and choose Additional Library, choose Building > BACnet_01_easy.lib, and click Open. See

the following figure.

💊 CoDeSys - Temp.pro*	
🚺 Eile Edit Project Insert Extras On	line <u>W</u> indow <u>H</u> elp
Resource Bibbl Vaideles Bibbl Vaideles Veidele, Contraction VAR, CONPG) Veidele, Contraction VAR, CONPG) Weidele, Contraction VAR, CONPG)	BARGRED 2 28 15 20 45 32 015 LBCALLBACK B2 15 10 20 45 32
Construction Construction Construction Surveyor S	
POUs Clate Date	PC03 #58(588:113.111%) 4

B. Create PLC programs.

This section describes how to write a simple application program, using the ST structured language for programming. The program achieves the following functions: If the VFD frequency is set to 5Hz, when the I0.0 terminal input of the digital input module 750–400 is high, the VFD runs forward; when the terminal input is low, the VFD decelerates to stop. (The VFD frequency is given by BACnet communication, the control mode is communication, and VFD received PZD2 is the set frequency.)

Variable declaration is in red box ①, and the main program is in red box ②. The format of data variable declaration is in the format of *Variable name*: *Data type of variable*.

🔩 CoDeSys - Temp.pro* - [PLC_PRG (PRG-S	57)]
🔩 File Edit Project Insert Extras Q	nline <u>W</u> indow <u>H</u> elp
1000011224	
Ca Pous - A Puc Pric (Pric	000 PROGRAMPLC_PRO 0000 PRO 0000 PRO

Description of the content in red box ①:

CONTROL_WORD is VFD control word; PZD_SEND2 is the given PZD of VFD, DI_1 is the channel 0 of digital input module, and %IX0.0 is the address of channel 0 of digital output module.

BACNET_ANALOG_VALUE is a data type of BACnet communication, which is defined in the library file BACnet_01_easy.lib.

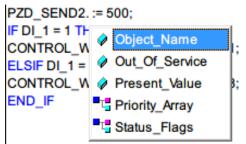
```
PROGRAM PLC_PRG
VAR
CONTROL_WORD :BACNET_ANALOG_VALUE;
PZD_SEND2 :BACNET_ANALOG_VALUE;
DI_1 AT%IX0.0 :BOOL;
```

END VAR

Description of the content in red box 2:

PZD_SEND2: Several properties of this object can follow at the right of the decimal point.

Present_Value indicates the current value.



The syntax ":=" is the assignment symbol.

```
PZD_SEND2.Present_Value := 500;
IF DI_1 = 1 THEN
CONTROL_WORD.Present_Value := 1;
ELSIF DI_1 = 0 THEN
CONTROL_WORD.Present_Value := 8;
```

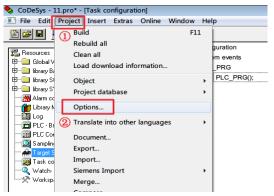
END IF

After the program is completed, press F11 to compile the project and proceed to the next step if there are no errors.

Note: After the program is completed, the PLC does not control the VFD directly, and you need to map the defined variables to the VFD variable table through WAGO BACnet configurator. Next, you need to generate the SYM_XML variable table and import it into the WAGO BACnet configurator.

C. Configure the symbol file.

In the main interface, choose **Project** > **Options** from the menu bar. In the pop-up **Options** dialog box, select **Dump XML Symbol table**, and click **Configura symbol file**. In the pop-up **Set object attributes** dialog box, select all, as shown in red box ④, and select the two items shown in red box ⑤. See the following figures.



Communication card

System events	Set object attributes	×
B- @ PLC_PRG Program	11,00	OK
Options	POUs @-@ Configuration	Cancel
Category:	EL_PRG (FRG)	
Load & Save User Information	Resources Global Variables	4
Desktop		
Directories	Herary Standard Ib 2 12 10 20:48:34: global variable Herary SYSLIBCALLBACK LIB 2 12 10 20:48:32: global variable	
Log Build		
Passwords 1 Source download		
Symbol configuration Database connection Macros		
PIG.03		
	Export variables of object Export data entries	
	Export structure components	
	Export array entries	
	☐ Write access	

After the setup, when the POU program is completed and the PLC is in login state, the file **xxx.sys_xml** (xxx is the project name) will be generated automatically and saved in the created project folder.

The **SYS_XML** file contains all variables in the project and can be imported directly into the **WAGO BACnet configurator** to configure the properties of each variable or object. This file is essential for the subsequent use of the PLC.

Note:

- If the system prompts that there is no relevant file locally during the generation process, you need to check whether the project built in CoDySys has been saved.
- If there are no variables declared in the POU program in the generated variable table, you need to check whether you have followed the steps in e) strictly.
- The XML file is generated only during the first PLC login. If the POU program
 has not changed, the XML file will not be regenerated subsequently, even if you
 log in again.
- D. Download the program to the PLC to generate the SYM_XML file.
- a) Choose Online > Communication Parameters. See the following figure.

	S Login	Alt+F8
	Logout	Ctrl+F8
POUL		
PLC_PRG (PRG) 000		
000	Children and Chi	FS
000	Stop	Shift+F8
	 Reset 	
000		
000		
000		FS
000		h
000		
000	Step over	F10
000	Step in	FE
001		Ctrl+F5
001		Ctrl+FJ
001		F7
001		Shift+F7
001		Ctrl+Shift+F7
001	4	Cur+Shirt+F/
	Show Call Stack	
	Display Flow Control	
	Simulation Mode	
	Communication Parameters (2)	
111	Sourcerode download	

b) In the Communication Parameters window, click the NEW button. Select Tcp/lp (3S Tcp/lp driver) in the new pop-up window, and then click OK. See the following figure.

Communication Paramete	Communication Parameters: New Channel Neme Ne				
Channels	[]		OK		
- OPC Client standar	annels				
Communication F	arameters: New Channel	×	New		
Name Tocalhost	via Tcp/lp_ 3	ОК	Remove		
Device		Cancel	Gateway		
Ethernet_TCP_IP OPC Client OPC Client 2 Tans Tcp/Ip	WAGO Ethernet TCP/IP driver WAGO OPC client driver WAGO OPC client driver 3S Tcp/Ip driver (2)		Update		

c) Select the newly created channel in the Communication Parameters window, and then fill in the PLC IP address in the Value column next to Address, and click OK. See the following figure.

Communication Parameters	×
Channels ③ ⊡ Tocohotť via Tcp/lp Uterret TCP/IP UPC Client standar. Usadess 1192163IP address or hostname Pot 2455 ② Molorola byteorder No	OK Cancel New Remove Sateway Update

d) Choose Online > Login on the menu bar. The software will compile the project and generate the SYM_XML file, which is located in the root directory of the project. After that, choose Create boot project to import the program file into the PLC, so that the PLC can run independently even if the PLC is disconnected from the PC. See the following figure.

1000 B 4000 B B 40	Login (1)	Alt+F8
and and the second seco	Logout	Ctrl+F8
POUs POUs POUs POUs POUs	Download	
BUC HIG (HIG)	Bun	15
	Stop	Shift+F8
	Reset	
	Reset (cold)	
	Reset (original)	-
	Toggle Breakpoint	F9
	Breakpoint Dialog	
	Step over	F10
	Step in	FB
	Single Cycle	Ctrl+F5
	Write Values	Ctrl+F7
	Force Values	17
	Release Force	Shift+F7
	Write/Force-Dialog	Ctr1+Shift+F7
	Show Call Stack	
	Display Flow Control	
	Simulation Mode	-
	Communication Parameters	
	Sourcecode download	
	Send marked text to RemoteControl Master (e	.g. as parameter)
	Create boot project (2)	
	Write file to PLC	
	Read file from PLC	=

- E. Configure the PLC through the WAGO BACnet configurator.
- a) Set the IP addresses of the PLC, PC, VFD and other slave stations, which are required to be in the same network segment, and connect each node through the network cable.
- b) Open the WAGO BACnet configurator and it will automatically scan the devices on the bus. Alternatively, scan manually by right-clicking on Device Pools-Scan

and choosing Scan. See the following figure.

WAGO BACnet Configurator - 1	New Project						
File Edit View Pool Dev		s Help	evice Auto	o Discov e r 💡	Monito	r: Devices + Prop	erties 🝷
El Database (0) El Import (0)	,* X Sca No	n Device: Device_0					
Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon Image: Second Carbon	Devi Mon Serv Netv Rem Colle Expa	an all Devices ce Auto Discovery itor All Devices	Nano	VendorId 222 486	#0bjs 18 31	* In Dat No No	BAChet MAC 0:192.168.1.1-bac 0:192.168.1.3-bac

In the preceding figure, **Controller [2721366]** is the VFD node in the bus and **Device_0 [549311]** is the PLC node. The numeric part in [] is the decimal number converted from this node MAC address.

c) Import the SYM_XML file.

Right click on **Device_0** (PLC device), and choose **Configure**. See the following figure.

Device Pools	<u>ه</u>	×	Scan Device: Device_0
Database (0) Import (0) Scan (2) Ontroller [27213 Alarms Alarms P → Device R → Values	66]		Name: Device_0 Instance Nr: 549311 Online Description: Device Link: Configurations \ Device 0
■ () Bevice 0 [549311] () Device () Files () Inputs () B10: BINARY () B10: BINARY () B00: BINARY () B00: BINARY () B00: BINARY () Schedulers () Values () AV0: PLC_PF () AV1: PLC_PF	_INPUT_1 _OUTPUT_ _OUTPUT_ G. CONTRO	0 🗸	Configure 2 nt Add to Database Rescan selected Device(s) Remove Device Monitor Services Import Export Device Manager
			Device 0

Import the previously generated **SYM_XML** file. After successful import, the objects (CONTROL_WORD, PZD_SEND2) declared in the previous project will appear in area ③. See the following figure.

Device: Device_0 BBMD Communication IEC Variables				
Name: Device_0				
Instance Nr: 549311 Online Address0:192.168.1.1-bac0				
Description			06;	jects: 18
Device Link:Scan \ Device 0 [599311]				
Configuration Info				
Version:		Creation		
Author:			UTC Date: 7/6/2016 3:	39:52 PM
Comments:		. Creation		50
		Device Ty		
		* Format Ye		
Override FilDatabase		Intert.	Upload from	Renove
STN XML:		Import.	Upload from	Remove
Import SymXML File				
* Object Name Pr	T	8		^
	remp	~		
B FILE 10 - 组织 * 新建文件夹	UI • 0	1 0		
FILE 3	63:D18	80		
B FILE_4 - A GABOR				
8 FILE_5 - ■ #Ⅲ	1016/7/7 9:44	SYM_XM		
8 FILE 6				
FILE 8 - 3. 最近的月期的位置				1
B FILE 9 3 -				
PLC_JPR. CONTROL_FC4D 0 PLC_FPR. FZD_SEND2 0 Im #				
SCREEKLE 0			4	U
-3 #010		_		•
Check		St	ore and Download	Store

Note:

- After the import is completed, check the I/O indicator of the PLC. If the import is correct, the indicator is steady on in green. If it is blinking in red periodically, you need to follow the instructions to perform troubleshooting.
- You can check the PLC status through the Status window of WAGO Ethernet Settings.
- For details about PLC panel indicators, see WAGO 750-830 description document.
- d) Map PLC variables to VFD variables.

Since the WAGO 750-830 PLC can only be used as a slave, not as a master, when used in a BACnet network. In actual use, it is necessary to map the relevant variables of the VFD or other devices to the PLC so that the VFD can be controlled by only operating the mapped address in the PLC through the PLC program.

The following example associates the **CONTROL_WORD** control word in the PLC program with the **CONTROL_WORD** in the VFD to control VFD startup, stop, and reset. The associated operations of other variables are similar.

 Select all devices that are scanned through the Scan menu. Right-click Add to Database. Choose Device Pools > Database to view the added devices. See the following figure.



 Access the Configuration interface of Device_0 (PLC device). Chose Values > AV0: PLC_PRG.CONRTOL_WORD. In the right area, choose Present Value > Client Mappings. See the following figure.

evice Configuration	Device: Device_0 Object	t: PLC_PRG. CONTROL_W	ORD			
Device_0 [549311] Device Files	Naze: PLC_P	RG.CONTROL_	NORD			
Dinputs Outputs Schedulers	Type: Analog Va Instance Nr:)	lue				
Values		: PLC_PRG. CONTROL_WORD				
AVI: PLC_PRG. PZD_SEN02	Nane	A Value	Type		Opt	Acc
IEC Variables Node-Configuration	OV Increment		Real			
Node-Configuration	Event State	STATE_NORMAL (0)	Enun: Eve	ntState		8
	🗄 💽 Object Identifier		Object Id	antifier		8
	Object Name	PLC_PRG. CONTROL ····	CharString			8
	€ Object Type	ANALOG_VALUE (2)	Enum: Obj	lect Type		8
	Out Of Service		Bool			4
	Present Value	2)	Real	1		4
	Priority Array	Rescan Object		ority Value		8
	Relinguish Defa	Add Property	•			-
	Status Flags	Remove Propert	y +	: Status Flags		6
	• Units	Reset Value		neering Units		6
		Add Element				
		Add Elements				
		Remove Element	(s)			
	1	Copy Value				
		B Paste Value				
	1	Paste Value as n	ew			
		Client Mappings	3			
	Check	Internal Mapping	3 5			
	(mene)	Auto-Size Colum		-		

-99-

c) In the Client Mapping Editor window, select the VFD variable to be mapped with the PLC variable AV0: PLC_PRG.CONRTOL_WORD. Select Controller – AV0: CONTROL_WORD-Present Value, and click Add Write. See the following figure.

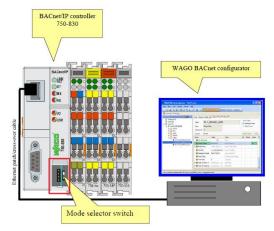
ient Mapping Editor		6
Present Value, PLC_PRG.CONTROL	_WORD (0), Device_0 [549311]	
Database 🖄 🔍 🗸	Type Remote Property	
Database (2)	Writing Present Value, CONTROL WORD (0), Controller [
□		
E Device	(4)	
E C Values		
AVO: CONTROL WORD		
Present Value		
AV1: PZD SEND 2		
AV2: PZD_SEND 3		
AV3: PZD SEND 4		
AV4: PZD SEND 5		
AV5: PZD SEND 6		
AV6: PZD SEND 7		
AV7: PZD SEND 8		
AV8: PZD SEND 9		
AV9: PZD SEND 10		
AV10: PZD_SEND 11		
AV12: STATUS_WORD		
AV13: PZD_RECEIVE 2		
AV14: PZD_RECEIVE 3		
AV15: PZD_RECEIVE 4		
AV16: PZD_RECEIVE 5	Parameters COV Write Priori Available 16 Vuse Subscription If available v	
	*rite rriori Available 10 • Use Subscription If available -	
AV18: PZD_RECEIVE 7	Poll Cycle: 3 COW Expiry: 120	
AV19: PZD_RECEIVE 8	Real Increment Threshol 0 Request Confirmed Notifications	
	Real Increment Inreshold	
	Cyclic sending: 30 Read Property Multiple	
AV23: PZD_RECEIVE 12	Add Read Map Add Write Delete Map Close	

d) You can view the added variable in the Client Mappings column in the property window of the PLC variable AV0: PLC_PRG.CONRTOL_WORD. Map other variables with the same method. After mapping all variables, click Store and Download to download the property to the PLC. See the following figure.

Type:) Instance Nr:								Supported Services COW Server COW Property Server Server Server		
Source: S		_FRG, CONTROL_WORD	Type	Ont	ker	Med	Internal Mappings	Client Mepp	ines	Persistence value
OW Incr			Real		4					0
Event Sta	ate	STATE_NORMAL (0)	Enum: EventState		0					STATE_NIRMAL (0)
= 💿 Object I	dentifier		Object Identifier		8					(NALOG_VALUE, 0)
Object N	ame	PLC_PRG. CONTROL	CharString 💌		۵		Object_Name			P.C_PRS. CONTROL_WORD
📵 Object T	ype	ANALOG_VALUE (2)	Enum: Object Type		۵					ANALOG_VALUE (2)
😠 Out Of S	ervice		Bool		4		Out_Of_Service			
Present	Value		Real		6		Present_Value	Present Val	ue, CONTR	4
🗉 👩 Priority	Array		Array: Priority Value		8		Priority_Array	Descard Malar	CONTROL N	WRD (0), Controller [272
😣 Relingui	sh Default		Real		6			Present value	, CONTROLLE	with [0], controller [277
🖲 Status F	lags		Bit String: Status Flags		6		Status_Flags			
 Units 			Enum: Engineering Units		۵					ND_UNITS (95)

F. Run the program.

Disconnect the PLC from the PC. Slide the PLC run switch to the top. The mode selection switch is in the red rectangular in the following figure. When the switch is



slid to the top, it is in the running state. See the following figure.

4.5.4 Controlling the VFD through WAGO BACnet configurator

If you only need to read and write VFD parameters and achieve basic control VFD functions, you can control the VFD directly through the WAGO BACnet configurator, without writing a PLC program.

- Set the PZD related function codes of the VFD, and set the VFD control mode to communicate.
- In the scanned devices, select the device (VFD) that needs to be controlled, and take Controller [2721366] as an example. Select the PZD functions or control words to be operated, taking PZD_SEND2 as an example. Enter the required value in the Present Value field. Then click Commit.

WAGO BACnet Configurator - Temp.wbc*					
Device Auto Discover 🔂 Mo	eiter Devicer + Despertier				
	nuor, bences + properties -				
File Edit View Pool Device Extras Help					
Browsing & Monitoring 🔦 Configure: Device_0 [549311]		•			
evice Pools 🖄 🗷 × Scan Device: 6	Controller Object: FZD_SEND 2				
Bi Database (2) Bi Import (0) Name: D				Show Values	
El Scan (2)	ZD_SEND 2			Munitored	Values
Controller [2721366] Type: Ana	log Value			@ Edited Val	ses
Alaras Instance Nr:					
Device Source: Gene	-		- 5		-
Values Source: Gen			3	Commit	Clear
AV1: P22 SEND 2 Name	_ Value		Туре	0pt	-
AV2: PZD_SEND 3	STATE_NORMAL (0)	0	Erun: EventState		
● AN3: PZD_SEND 4 ● AN4: PZD_SEND 5	tifier (ANALOG_VALUE, 1)	G	Object Identifier		
AV4: P2D_SEND 5	PZD_SEND 2	0	CharString	•	
ANG: PZD_SEND 7 (B) Object Type	ANALOG_VALUE (2)	0	Enum: Object Type		
AV7: PZD_SEND 8 AV8: PZD_SEND 9 Out Of Serv	rice 📃	0	Bool		
AV9: PZD SEND 9			Real		
AV10: P2D_SEND 11 Status Flag		0	Bit String: Status	Flare	
AV11: P2D_SEND 12	NO UNITS (95)		Erum: Engineering U		
AV12: STATUS_WORD Wolts AV13: PZD_RECEIVE 2	NO_COLLO (NO)		toon. togrowering t		
AV14: PZD RECEIVE 3					
AV15: PZD_RECEIVE 4					
AV16: PZD_RECEIVE 5					
AN17: PZD_RECEIVE 6					
AV18: PZD_RECEIVE 7					
AV19: PZD_RECEIVE 8					
AV20: PZD_RECEIVE 9					
AV21: PZD RECEIVE 10 AV22: PZD RECEIVE 11 4					
AV22: PZD_RECEIVE 11					

Set other parameters in the same way.

Chapter 5 CAN master/slave expansion card

5.1 CAN master/slave expansion card operation

5.1.1 Electrical connection

Use shielding wires as the bus cable, if possible. It is recommended that you connect the shielding wire to the PE terminal of the communication card. When there are only two devices for CAN master-slave communication, both devices shall be connected to the terminal resistor. When there are more than two devices, the starting device and terminal device shall be connected to the terminal resistor. The terminal resistor of the communication card can be connected through its terminal resistor switch. Figure 5-1 shows the electrical wiring.

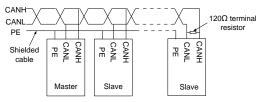


Figure 5-1 Electrical wiring diagram

5.1.2 Electrical wiring

Connection operation	Description
1. Determine the master count and slave count, and	Pay attention to the cable
prepare the shielded twisted pairs.	specifications.
 Connect the cable according to Figure 5-1, and install the terminal resistor at the head and tail properly. 	Pay attention to the terminal
3. Check the card hardware version according to the	Pay attention to the board silk model
order finished-product card model.	number.
4. Power on in a unified way. Check card type related function codes P19.00/P19.01/P19.02, card version related function codes P19.03/P19.04/P19.05, and control board software version function code P07.13 for consistency.	used.
5. Use the following function code setting examples for reference.	Pay attention to the actual configuration.
	U U
6. Power on in a unified way, and check whether the	
running is normal.	properly, the networking is

Connection operation	Description
	successful.
7. Set torque and frequency related function codes	Pay attention to the actual
according to the actual application.	application.

5.1.3 Function code setting

Master setting example						
Function code viewing and setting	Value	Description				
P19.00/P19.01/19.02	15	CAN master/slave communication card type is successfully identified according to the card slot.				
P19.03/P19.04/19.05	*_**	CAN master/slave communication card is successfully identified according to the card type and version number.				
P28.00	1	The local device is a master.				
P28.06	N	Slave count				
P15.28	/	The communication address is invalid when it is a master.				
P15.29	2	The communication baud rate is 125kbps.				
P15.30	5.0	The CAN communication timeout time is 5s.				

Example of slave 1						
Function code viewing and setting	Value	Description				
P19.00/P19.01/19.02	15	CAN master/slave communication card type is successfully identified according to the card slot.				
P19.03/P19.04/19.05	* **	CAN master/slave communication card is successfully identified according to the card type and version number.				
P28.00	2	The local device is a slave.				
P28.06	/	The count is invalid when it is a slave.				
P15.28	1	The slave CAN communication				

Example of slave 1					
Function code viewing and setting	Value	Description			
		address starts from 1.			
P15.29	2	The communication baud rate is 125kbps.			
P15.30	5.0	The CAN communication timeout time is 5s.			

	Example of slave N					
Function code viewing and setting	Value	Description				
P19.00/P19.01/19.02	15	CAN master/slave communication card type is successfully identified according to the card slot.				
P19.03/P19.04/19.05	* **	CAN master/slave communication card is successfully identified according to the card type and version number.				
P28.00	2	The local device is a slave.				
P28.06	/	The count is invalid when it is a slave.				
P15.28	N	The slave CAN communication address ends at N.				
P15.29	2	The communication baud rate is 125kbps.				
P15.30	5.0	The CAN communication timeout time is 5s.				

5.1.4 Fault handling

Fault	Description	
Neither the master nor the slave has a fault.	The master/slave networking is successful.	
	The cause is that the slave timeout time is	
The salve reports fault code 58: CAN	reached. Check the slave electrical wiring and	
communication fault (ESCAN)	function code settings. Check whether the	
	fault is removed after reset.	
The meeter reports foult and CO. CAN alous	The cause is that a slave encounters a fault.	
The master reports fault code 69: CAN slave	Check the slave electrical wiring and function	
fault in master/slave synchronization (S-Err).	code settings. Check whether the fault is	

Fault	Description		
	removed after reset.		
	The cause is that the master timeout time is		
The salve reports fault code 58: CAN	reached. Check the master electrical wiring		
communication fault (ESCAN).	and function code settings. Check whether		
	the fault is removed after reset.		

5.2 Related function parameters

Function code	Name	Parameter description	Default	Modify
P15.28	CAN communication address	0–127	1	O
P15.29	CAN communication baud rate	0–5 0: 50kbps 1: 100kbps 2: 125kbps 3: 250kbps 4: 500kbps 5: 1Mbps	2	O
P15.30	CAN communication timeout period	0.0–60.0(s) Note: It is invalid when the value is 0.0.	5.0	0
P19.00	Type of expansion card at slot 1	0–65535 0: No card	0	•
P19.01	Type of expansion card at slot 2	1: PLC card 2: I/O card	0	•
P19.02	Type of expansion card at slot 3	3: Incremental PG card 4: Incremental PG card with UVW 5: Ethernet 6: DP 7: Bluetooth card 8: Resolver PG card 9: CANopen Communication card 10: WIFI card 11: PROFINET communication card 12: Sine-cosine PG card without CD signals 13: Sine-cosine PG card with CD signals	0	•

Function code	Name	Parameter description	Default	Modify	
		14: Absolute encoder PG card			
		(reserved)			
		15: CAN master/slave			
		communication card			
		16: Modbus TCP communication			
		card			
		17: EtherCAT communication card			
		(reserved)			
		18: BACnet communication card			
		(reserved)			
		19: DeviceNET communication card			
		(reserved)			
		20: PT100/PT1000 temperature			
		detection card			
		21: EtherNet IP card			
		22: MECHATROLINK card			
		(reserved)			
		23–65535: Reserved			
		0–2			
P28.00	Master/slave mode	0: Master/slave control is invalid.		0	O
		1: The local device is a master.			
	Maatar/alaus	2: The local device is a slave.			
D00.04	Master/slave	0–1 0: CAN	0		
P28.01		0: CAN 1: Reserved	0	O	
	selection				
		0x000–0x112			
		Ones place: Master/slave running mode selection			
		0: Master/slave mode 0			
		1: Master/slave mode 1			
	Master/slave control	2: Master/slave mode 2			
P28.02	mode	Tens place: Slave start command	0x001	0	
	moue	source			
		0: Master			
		1: Determined by P00.01			
		Hundreds place: Whether to enable			
		master/slave to send/receive data			

Function code	Name	Parameter description	Default	Modify
		0: Enable		
		1: Disable		
P28.03	Slave speed gain	0.0–500.0%	100.0%	0
P28.04	Slave torque gain	0.0–500.0%	100.0%	0
P28.05	Speed/torque mode switching frequency point in master/slave mode 2	0.00–10.00Hz	5.00Hz	0
P28.06	Slave count	1–15	1	O
P28.07	Hidden	0–0	0	•
P28.08	Hidden	0–0	0	•
P28.09	CAN slave torque offset	-100.0–100.0%	0.0%	0

Appendix A CANopen object dictionary

Index (hexadecimal)	Subindex	Description	Access permission	Data type	Default value
1000	0	Device type	RO	Unsigned32	0x0000 0000
1001	0	Error register	RO	Unsigned8	/
1003	0	Number of subindexes	RW	/	/
	1	Error code	RO	Unsigned32	/
1005	0	COB-ID SYNC	RW	Unsigned32	/
1006	0	Communication cycle period	RW	Unsigned32	/
1007	0	Length of synchronous window	RW	Unsigned32	/
1008	0	Manufacturer- defined device name	CONST	String	INVT CANopen
1009	0	Manufacturer- defined hardware version	CONST	String	V1.00
100A	0	Manufacturer- defined software version	CONST	String	V1.00
100C	0	Protection time	RW	Unsigned16	0
100D	0	Life cycle factor	RW	Unsigned16	0
		Consu	mer heartbe	at time	
1016	0	Number of subindexes	RO	Unsigned8	/
	1	Consumer heartbeat time	RW	Unsigned32	/
1017	0	Producer heartbeat time	RW	Unsigned16	0
		ld	entifier objed	cts	
1018	0	Number of subindexes	RO	Unsigned8	4
	1	Supplier ID	RO	Unsigned32	0x0000 0000
	2	Product code	RO	Unsigned32	0x0000 00000

Index (hexadecimal)	Subindex	Description	Access permission	Data type	Default value
	3	Revision No.	RO	Unsigned32	0x0000 0000
	4	Sequence No.	RO	Unsigned32	0x0000 0000
			Servo SDO		
	0	Number of subindexes	RO	Unsigned8	/
1200	1	COB-ID Client -> server (Rx)	RO	Unsigned32	600H+Node ID
	2	COB-ID Server -> client (Tx)	RO	Unsigned32	580H+Node ID
			SDO		
	0	Number of subindexes	RO	Unsigned8	/
1280	1	COB-ID Client -> server (Rx)	RO	Unsigned32	/
	2	COB-ID Server -> client (Tx)	RO	Unsigned32	/
	3	Node ID of server SDO	RO	Unsigned8	/
		PDO1 Rx co	mmunication	n parameters	
	0	Supported Max. number of subindexes	RO	Unsigned8	/
1400	1	COB-ID used by PDO	RW	Unsigned32	/
	2	Transmission type	RW	Unsigned8	/
	3	/	/	Unsigned16	/
	4	/	/	Unsigned8	/
	5	Event timer	RW	Unsigned16	/
		PDO2 Rx co	mmunication	n parameters	
	0	Supported Max. number of subindexes	RO	Unsigned8	/
1401	1	COB-ID used by PDO	RW	Unsigned32	/
	2	Transmission type	RW	Unsigned8	/
	3	/	/	Unsigned16	/
	4	/	/	Unsigned8	/

Index (hexadecimal)	Subindex	Description	Access permission	Data type	Default value
	5	Event timer	RW	Unsigned16	/
		PDO3 Rx co	mmunication	n parameters	
	0	Supported Max. number of subindexes	RO	Unsigned8	/
1402	1	COB-ID used by PDO	RW	Unsigned32	/
	2	Transmission type	RW	Unsigned8	/
	3	/	/	Unsigned16	/
	4	/	/	Unsigned8	/
	5	Event timer	RW	Unsigned16	/
		PDO4 Rx co	mmunication	n parameters	
	0	Supported Max. number of subindexes	RO	Unsigned8	/
1403	1	COB-ID used by PDO	RW	Unsigned32	/
	2	Transmission type	RW	Unsigned8	/
	3	/	/	Unsigned16	/
	4	/	/	Unsigned8	/
	5	Event timer	RW	Unsigned16	/
		PDO1 Rx	mapping pa	rameters	
	0	Number of application program objects mapped in PDO	RW	Unsigned8	3
1600	1	First mapped object	RW	Unsigned32	0x21000010
	2	Second mapped object	RW	Unsigned32	0x21000110
	3	Third mapped object	RW	Unsigned32	0x21000210
		PDO2 Rx	mapping pa	rameters	
1601	0	Number of application program objects mapped in PDO	RW	Unsigned8	4

Index (hexadecimal)	Subindex	Description	Access permission	Data type	Default value
, , ,	1	First mapped object	RW	Unsigned32	0x21010010
	2	Second mapped object	RW	Unsigned32	0x21000310
	3	Third mapped object	RW	Unsigned32	0x21000410
	4	Fourth mapped object	RW	Unsigned32	0x21000510
		PDO3 Rx	mapping pa	rameters	
	0	Number of application program objects mapped in PDO	RW	Unsigned8	4
1602	1	First mapped object	RW	Unsigned32	0x21000610
	2	Second mapped object	RW	Unsigned32	0x21000710
	3	Third mapped object	RW	Unsigned32	0x21000810
	4	Fourth mapped object	RW	Unsigned32	0x21000910
		PDO4 Rx	mapping pa	rameters	
	0	Number of application program objects mapped in PDO	RW	Unsigned8	4
1603	1	First mapped object	RW	Unsigned32	0x21000a10
	2	Second mapped object	RW	Unsigned32	0x21000b10
	3	Third mapped object	RW	Unsigned32	0x21000c10
	4	Fourth mapped object	RW	Unsigned32	0x21000d10
		PDO1 Tx co	mmunication	n parameters	
1800	0	Supported Max. number of	RO	Unsigned8	/

Index (hexadecimal)	Subindex	Description	Access permission	Data type	Default value	
		subindexes				
	1	COB-ID used by PDO	RW	Unsigned32	/	
	2	Transmission type	RW	Unsigned8	255	
	3	Disabled time	RW	Unsigned16	500	
	4	Reserved	RW	Unsigned8	/	
	5	Event timer	RW	Unsigned16	0	
		PDO2 Tx co	mmunicatior	n parameters		
	0	Supported Max. number of subindexes	RO	Unsigned8	/	
1801	1	COB-ID used by PDO	RW	Unsigned32	/	
	2	Transmission type	RW	Unsigned8	254	
	3	Disabled time	RW	Unsigned16	500	
	4	Reserved	RW	Unsigned8	/	
	5	Event timer	RW	Unsigned16	0	
	PDO3 Tx communication parameters					
	0	Supported Max. number of subindexes	RO	Unsigned8	/	
1802	1	COB-ID used by PDO	RW	Unsigned32	/	
	2	Transmission type	RW	Unsigned8	254	
	3	Disabled time	RW	Unsigned16	500	
	4	Reserved	RW	Unsigned8	/	
	5	Event timer	RW	Unsigned16	0	
		PDO4 Tx co	mmunicatior	n parameters		
	0	Supported Max. number of subindexes	RO	Unsigned8	/	
1803	1	COB-ID used by PDO	RW	Unsigned32	/	
	2	Transmission type	RW	Unsigned8	254	
	3	Disabled time	RW	Unsigned16	500	
	4	Reserved	RW	Unsigned8	/	
	5	Event timer	RW	Unsigned16	0	

Index (hexadecimal)	Subindex	Description	Access permission	Data type	Default value	
		PDO1 Tx mapping parameters				
	0	Number of application program objects mapped in PDO	RW	Unsigned8	3	
1A00	1	First mapped object	RW	Unsigned32	0x20000010	
	2	Second mapped object	RW	Unsigned32	0x20000110	
	3	Third mapped object	RW	Unsigned32	0x20000210	
		PDO2 Tx	mapping pa	rameters		
	0	Number of application program objects mapped in PDO	RW	Unsigned8	4	
1A01	1	First mapped object	RW	Unsigned32	0x20010010	
	2	Second mapped object	RW	Unsigned32	0x20000310	
	3	Third mapped object	RW	Unsigned32	0x20000410	
	4	Fourth mapped object	RW	Unsigned32	0x20000510	
		PDO3 Tx	mapping pa	rameters	-	
	0	Number of application program objects mapped in PDO	RW	Unsigned8	4	
1A02	1	First mapped object	RW	Unsigned32	0x20000610	
	2	Second mapped object	RW	Unsigned32	0x20000710	
	3	Third mapped object	RW	Unsigned32	0x20000810	
	4	Fourth mapped object	RW	Unsigned32	0x20000910	

Index (hexadecimal)	Subindex	Description	Access permission	Data type	Default value	
	PDO4 Tx mapping parameters					
1A03	0	Number of application program objects mapped in PDO	RW	Unsigned8	4	
	1	First mapped object	RW	Unsigned32	0x20000a10	
	2	Second mapped object	RW	Unsigned32	0x20000b10	
	3	Third mapped object	RW	Unsigned32	0x20000c10	
	4	Fourth mapped object	RW	Unsigned32	0x20000d10	

Appendix B Related function codes

Function code	Name	Parameter description	Setting range	Default value
	Channel of	0: Keypad		
P00.01	running	1: Terminal	0–2	0
	commands	2: Communication		
P00.02	Communication mode of running commands	0: Modbus/Modbus TCP communication 1: PROFIBUS/CANopen/DeviceNet/BACnet MSTP communication 2: Ethernet communication 3: EtherCAT/PROFINET/EtherNet IP communication 4: Programmable card 5: Wireless communication card 6: USB communication (reserved) Note: The values 1–5 correspond to extended functions that are available	0–6	0
P00.06	A frequency command selection	only with respective cards. 0: Set via keypad 1: Set via Al1 2: Set via Al2	0–15	0
P00.07	B frequency command selection	 a: Set via Al3 4: Set via high speed pulse HDIA 5: Set via simple PLC program 6: Set via multi-step speed running 7: Set via PID control 8: Set via Modbus/Modbus TCP communication 9: Set via PROFIBUS/CANopen/DeviceNet/BACn et MSTP communication 10: Set via Ethernet communication 11: Set via high speed pulse HDIB 12: Set via pulse train AB 13: Set via EtherCAT/PROFINET communication 	0–15	15

Function code	Name	Parameter description	Setting range	Default value
		14: Set via programmable card		
		15: Reserved		
P00.11	ACC time 1	0.0–3600.0s	0.0– 3600.0s	Model depended
P03.11	Torque setting mode selection	0–1: Keypad (P03.12) 2: Al1 3: Al2 4: Al3 5: Pulse frequency HDIA 6: Multi-step torque 7: Modbus/Modbus TCP communication 8: PROFIBUS/CANopen/DeviceNet/BACn et MSTP communication 9: Ethernet communication 10: Pulse frequency HDIB 11: EtherCAT/PROFINET/EtherNet IP 12: Programmable card Note: For these settings, 100% corresponds to the motor rated current.	0–12	0
P03.14	Setting source of FWD rotation frequency upper limit in torque control	Conception of the indefinition of the event	0–12	0

Function code	Name	Parameter description	Setting range	Default value
		12: Reserved		
		Note: For these settings, 100%		
		corresponds to the max. frequency. 0: Keypad (P03.17)	-	
		1: Al1		
		2: AI2		
		3: AI3		
		4: Pulse frequency HDIA		
		5: Multi-step setting		
		6: Modbus/Modbus TCP		
	Setting source of	communication		
	REV rotation	7:		
P03.15	frequency upper	PROFIBUS/CANopen/DeviceNet/BACn	0-12	0
1 00.10	limit in torque	et MSTP communication	0-12	0
	control	8: Ethernet communication		
		9: Pulse frequency HDIB		
		10: EtherCAT/PROFINET/EtherNet IP		
		communication		
		11: Programmable card		
		12: Reserved		
		Note: For these settings, 100%		
		corresponds to the max. frequency.		
		0: Keypad (P03.20)		
		1: Al1		
		2: AI2		
		3: AI3		
		4: Pulse frequency HDIA		
		5: Modbus/Modbus TCP		
	Setting source of	communication		
P03.18	electromotive	6:	0–11	0
	torque upper limit	PROFIBUS/CANopen/DeviceNet/BACn		
		et MSTP communication		
		7: Ethernet communication		
		8: Pulse frequency HDIB		
		9: EtherCAT/PROFINET/EtherNet IP		
		communication		
		10: Programmable card		

Function code	Name	Parameter description	Setting range	Default value
		11: Reserved		
		Note: For these settings, 100%		
		corresponds to the motor rated current.		
		0: Keypad (P03.21)		
		1: Al1		
		2: AI2		
		3: AI3		
		4: Pulse frequency HDIA		
		5: Modbus/Modbus TCP		
		communication		
		6:		
D00.40	Setting source of	PROFIBUS/CANopen/DeviceNet/BACn	0.44	
P03.19	braking torque	et MSTP communication	0–11	0
	upper limit	7: Ethernet communication		
		8: Pulse frequency HDIB		
		9: EtherCAT/PROFINET/EtherNet IP		
		communication		
		10: Programmable card		
		11: Reserved		
		Note: For these settings, 100%		
		corresponds to the motor rated current.		
		0: Keypad; output voltage is determined		
		by P04.28		
		1: Al1		
		2: AI2		
		3: AI3		
		4: HDIA		
		5: Multi-step (the set value is		
	Voltage setting	determined by P10 group)		
P04.27	channel	6: PID	0–13	0
		7: Modbus/Modbus TCP		
		communication		
		8:		
		PROFIBUS/CANopen/DeviceNet/BACn		
		et MSTP communication		
		9: Ethernet communication		
		10: HDIB		

Function code	Name	Parameter description	Setting range	Default value
		11: EtherCAT/PROFINET/EtherNet IP	•	
		12: Programmable card		
		13: Reserved		
P06.01	Y1 output	0: Invalid	0–63	0
P06.02	HDO output	1: In running	0–63	0
D 00.00	Relay output	2: In forward running	0.00	
P06.03	RO1	3: In reverse running	0–63	1
		4: In jogging		
		5: VFD fault		
		6: Frequency level detection FDT1		
		7: Frequency level detection FDT2		
		8: Frequency reached		
		9: Running in zero speed		
		10: Reach upper limit frequency		
		11: Reach lower limit frequency		
		12: Ready to run		
		13: In pre-exciting		
		14: Overload pre-alarm		
		15: Underload pre-alarm		
		16: Simple PLC stage completed		
		17: Simple PLC cycle completed		
P06.04	Relay output	18: Reach set counting value	0–63	5
	RO2	19: Reach designated counting value		
		20: External fault is valid		
		21: Reserved		
		22: Reach running time		
		23: Virtual terminal output of Modbus		
		communication		
		24: Virtual terminal output of		
		POROFIBUS/CANopen/DeviceNet/BA		
		Cnet MSTP communication		
		25: Virtual terminal output of Ethernet		
		communication		
		26: DC bus voltage established		
		27: z pulse output		
		28: During pulse superposition		

Function code	Name	Parameter description	Setting range	Default value
		29: STO act		
		30: Positioning completed		
		31: Spindle zeroing completed		
		32: Spindle scale-division completed		
		33: In speed limit		
		34: EtherCAT/PROFINET/EtherNet IP		
		35: Reserved		
		36: Speed/position control switchover		
		completed		
		37: Any frequency reached		
		38–40: Reserved		
		41: Y1 from the programmable card		
		42: Y2 from the programmable card		
		43: HDO from the programmable card		
		44: RO1 from the programmable card		
		45: RO2 from the programmable card		
		46: RO3 from the programmable card		
		47: RO4 from the programmable card		
		48: EC PT100 detected OH pre-alarm		
		49: EC PT1000 detected OH pre-alarm		
		50: AI/AO detected OH pre-alarm		
		51: Stopped or running at zero speed		
		52: Disconnection detected in tension		
		control		
		53: Roll diameter setting reached		
		54: Max. roll diameter reached		
		55: Min. roll diameter reached		
		56: Fire control mode enabled		
		57–63: Reserved		
Dog 4 1	Analog output	0: Running frequency (0–Max. output	0.00	
P06.14	AO1	frequency)	0–63	0
		1: Set frequency (0–Max. output		
		frequency)		
P06.16	HDO high-speed	2: Ramp reference frequency (0-Max.	0–63	0
	pulse output	output frequency)		
		3: Rotational speed (100% corresponds		

Function code	Name	Parameter description	Setting range	Default value
		to the speed at max. output frequency.)		
		4: Output current (100% corresponds to		
		twice the VFD rated current.)		
		5: Output current (100% corresponds to		
		twice the motor rated current.)		
		6: Output voltage (100% corresponds to		
		1.5 times the VFD rated voltage.)		
		7: Output power (100% corresponds to		
		twice the motor rated power.)		
		8: Set torque (100% corresponds to		
		twice the motor rated current.)		
		9: Output torque (Absolute value; 100%		
		corresponds to twice the motor rated		
		torque.)		
		10: Al1 input (0–10V/0–20mA)		
		11: Al2 input (0–10V)		
		12: AI3 input (0–10V/0–20mA)		
		13: HDIA input (0.00–50.00kHz)		
		14: Value 1 set through Modbus		
		(0–1000)		
		15: Value 2 set through Modbus		
		(0–1000)		
		16: Value 1 set through		
		PROFIBUS/CANopen/DeviceNet/BACn		
		et MSTP (0–1000)		
		17: Value 2 set through		
		PROFIBUS/CANopen/DeviceNet/BACn		
		et MSTP (0–1000)		
		18: Value 1 set through Ethernet 1		
		(0–1000)		
		19: Value 2 set through Ethernet 2		
		(0–1000)		
		20: HDIB input (0.00–50.00kHz)		
		21: Value 1 set through		
		EtherCAT/PROFINET/EtherNet IP		
		(0–1000)		

Function code	Name	Parameter description	Setting range	Default value
couc		22: Torque current (bipolar; 100%	Tange	Value
		corresponds to triple the motor rated		
		current.)		
		23: Exciting current (bipolar; 100%		
		corresponds to triple the motor rated		
		current.)		
		24: Set frequency (bipolar; 0–Max.		
		output frequency)		
		25: Ramp reference frequency (bipolar;		
		0-Max. output frequency)		
		26: Rotational speed (bipolar; 0-Speed		
		corresponding to max. output		
		frequency)		
		27: Value 2 set through		
		EtherCAT/PROFINET/EtherNet IP		
		communication (0-1000)		
		28: AO1 from the programmable card		
		(0–1000)		
		29: AO2 from the programmable card		
		(0–1000)		
		30: Rotational speed (100%		
		corresponds to twice the motor rated		
		synchronous speed)		
		31: Output torque (Actual value, 100%		
		corresponds to twice the motor rated		
		torque)		
		32: AI/AO temperature detection output		
		33–63: Reserved		
		Note:		
		When the output comes from the		
		programmable card (28–29), if the card		
		is a Codesys programmable card,		
		P27.00 must be set to 1.		
		When AO1 is of the current output type,		
		100% corresponds to 20mA; when AO1		
		is of the voltage output type, 100%		
		corresponds to 10V; 100% of HDO		

Function code	Name	Parameter description	Setting range	Default value
		corresponds to the output of P06.30.		
P07.27	Type of current fault	0: No fault 1: Inverter unit U phase protection	0–79	0
P07.28	Type of last fault	(OUt1)	0–79	0
P07.29	Type of 2nd-last fault	2: Inverter unit V phase protection (OUt2)	0–79	0
P07.30	Type of 3rd-last fault	3: Inverter unit W phase protection (OUt3)	0–79	0
P07.31	Type of 4th-last fault	4: Overcurrent during acceleration (OC1)	0–79	0
P07.32	Type of 5th-last fault	5: Overcurrent during deceleration (OC2) 6: Overcurrent during constant speed (OC3) 7: Overvoltage during acceleration (OV1) 8: Overvoltage during deceleration (OV2) 9: Overvoltage during constant speed (OV3) 10: Bus undervoltage fault (UV) 11: Motor overload (OL1) 12: VFD overload (OL2) 13: Phase loss on input side (SPI) 14: Phase loss on output side (SPO) 15: Rectifier module overheat (OH1) 16: Inverter module overheat (OH2) 17: External fault (EF) 18: Modbus/Modbus TCP communication fault (CE) 19: Current detection fault (ItE) 20: Motor autotuning fault (ItE) 21: EEPROM operation fault (EP) 22: PID feedback offline fault (PIDE) 23: Brake unit fault (bCE) 24: Running time reached (END) 25: Electronic overload (OL3) 26: Keypad communication error (PCE)	0–79	0

Function code	Name	Parameter description	Setting range	Default value
		27: Parameter upload error (UPE)		
		28: Parameter download error (DNE)		
		29: PROFIBUS communication fault		
		(E-DP)		
		30: Ethernet communication fault (E-NET)		
		31: CANopen communication fault		
		(E-CAN)		
		32: To-ground short-circuit fault 1		
		(ETH1)		
		33: To-ground short-circuit fault 2		
		(ETH2)		
		34: Speed deviation fault (dEu)		
		35: Mal-adjustment fault (STo)		
		36: Underload fault (LL)		
		37: Encoder offline fault (ENC1o)		
		38: Encoder reversal fault (ENC1d)		
		39: Encoder Z pulse offline fault		
		(ENC1Z)		
		40: Safe torque off (STO)		
		41: Channel H1 safety circuit exception		
		(STL1)		
		42: Channel H2 safety circuit exception		
		(STL2)		
		43: Channel H1 and H2 exception		
		(STL3)		
		44: Safety code FLASH CRC fault (CrCE)		
		45: Programmable card customized		
		fault 1 (P-E1)		
		46: Programmable card customized		
		fault 2 (P-E2)		
		47: Programmable card customized		
		fault 3 (P-E3)		
		48: Programmable card customized		
		fault 4 (P-E4)		

Function code	Name	Parameter description	Setting	Default value
code		40. Deserve weble could suptomize d	range	value
		49: Programmable card customized fault 5 (P-E5)		
		50: Programmable card customized		
		fault 6 (P-E6)		
		51: Programmable card customized		
		fault 7 (P-E7)		
		52: Programmable card customized		
		fault 8 (P-E8)		
		53: Programmable card customized fault 9 (P-E9)		
		54: Programmable card customized fault 10 (P-E10)		
		55: Repetitive expansion card type fault		
		(E-Err)		
		56: Encoder UVW loss fault (ENCUV)		
		57: PROFIBUS communication fault		
		(E-PN)		
		58: CANopen communication fault (SECAN)		
		59: Motor over-temperature fault (OT)		
		60: Card slot 1 card identification failure		
		(F1-Er)		
		61: Card slot 2 card identification failure		
		(F2-Er)		
		62: Card slot 3 card identification failure		
		(F3-Er)		
		63: Card slot 1 card communication		
		timeout fault (C1-Er)		
		64: Card slot 2 card communication		
		timeout fault (C2-Er)		
		65: Card slot 3 card communication		
		timeout fault (C3-Er)		
		66: EtherCAT communication fault		
		(E-CAT)		
		67: BACnet communication fault		
		(E-BAC)		
		68: DeviceNet communication fault		
		(E-DEV)		

Function code	Name	Parameter description	Setting range	Default value
code		69: Master-slave synchronous CAN slave fault (S-Err) 70: EC PT100 detected overheating (OtE1) 71: EC PT1000 detected overheating (OtE2) 72: EtherNet/IP communication timeout (E-EIP) 73: No upgrade bootload (E-PAO) 74: Al1 disconnected (E-Al2) 75: Al2 disconnected (E-Al2) 76: Al3 disconnected (E-Al3) 77: Al/AO detected OH (OH3)	range	value
		78: Brake feedback fault (E-brF)79: Stalling in V/F control (E-StK)80: Losing steps in V/F control (E-LSt)		
P08.31	Motor 1 and motor 2 switching channel	LED ones place: Switching channel 0: Terminal 1: Modbus/Modbus TCP communication 2: PROFIBUS/CANopen/DeviceNet/BACn et MSTP communication 3: Ethernet communication 4: EtherCAT/PROFINET/EtherNet IP communication LED tens place: Switching in running 0: Disabled 1: Enabled	0x00–0x1 4	0x00
P09.00	PID reference source	0: Set by P09.01 1: Al1 2: Al2 3: Al3 4: High-speed pulse HDIA 5: Multi-step 6: Modbus communication 7:	0–12	0

Function code	Name	Parameter description	Setting range	Default value
		PROFIBUS/CANopen/DeviceNet/BACn		
		et MSTP communication		
		8: Ethernet communication		
		9: High-speed pulse HDIB		
		10: EtherCAT/PROFINET		
		communication		
		11: Programmable card		
		12: Reserved		
		0: Al1		
		1: AI2		
		2: AI3		
		3: High-speed pulse HDIA		
		4: Modbus communication		
		5:		
	PID feedback	PROFIBUS/CANopen/DeviceNet/BACn		
P09.02	source	et MSTP communication	0–10	0
		6: Ethernet communication		
		7: High-speed pulse HDIB		
		8: EtherCAT/PROFINET		
		communication		
		9: Programmable expansion card		
		10: Reserved		
P15.01	Module address	0–127	0–127	2
P15.02	Received PZD2	0–31	0–31	0
P15.03	Received PZD3	0: Invalid	0–31	0
P15.04	Received PZD4	1: Set frequency (0–Fmax, unit: 0.01	0–31	0
P15.05	Received PZD5	Hz)	0–31	0
P15.06	Received PZD6	2: PID reference (-1000–1000, in which	0–31	0
P15.07	Received PZD7	1000 corresponds to 100.0%)	0–31	0
P15.08	Received PZD8	3: PID feedback (-1000–1000, in which	0–31	0
P15.09	Received PZD9	1000 corresponds to 100.0%)	0–31	0
P15.10	Received PZD10	4: Torque setting (-3000–+3000, in	0–31	0
P15.11	Received PZD11	which 1000 corresponds to 100.0% of	0–31	0
P15.12	Received PZD12	the rated current of the motor) 5: Setting of the upper limit of forward running frequency (0–Fmax, unit: 0.01 Hz)	0–31	0

Function code	Name	Parameter description	Setting range	Default value
		6: Setting of the upper limit of reverse		
		running frequency (0-Fmax, unit: 0.01		
		Hz)		
		7: Upper limit of the electromotive		
		torque (0-3000, in which 1000		
		corresponds to 100.0% of the rated		
		current of the motor)		
		8: Upper limit of the brake torque		
		(0-3000, in which 1000 corresponds to		
		100.0% of the rated current of the		
		motor)		
		9: Virtual input terminal command,		
		0x000–0x3FF		
		(corresponding to S8, S7, S6, S5,		
		HDIB, HDIA, S4, S3, S2, and S1 in		
		sequence)		
		10: Virtual output terminal command,		
		0x00–0x0F		
		(corresponding to RO2, RO1, HDO, and		
		Y1 in sequence)		
		11: Voltage setting (for V/F separation)		
		(0-1000, in which 1000 corresponds to		
		100.0% of the rated voltage of the motor)		
		12: AO1 output setting 1 (-1000-+1000,		
		in which 1000 corresponds to 100.0%)		
		13: AO2 output setting 2 (-1000-+1000,		
		in which 1000 corresponds to 100.0%)		
		14: MSB of position reference (signed number)		
		15: LSB of position reference (unsigned		
		number)		
		16: MSB of position feedback (signed		
		number)		
		17: LSB of position feedback (unsigned		
		number)		
		18: Position feedback setting flag		
		(position feedback can be set only after		

Function code	Name	Parameter description	Setting range	Default value
		this flag is set to 1 and then to 0) 19: Function code mapping (PZD2–PZD12 correspond to P14.49–P14.59 respectively.) 20–31: Reserved		
P15.13	Transmitted PZD2	0-47 0: Invalid	0–47	0
P15.14	Transmitted PZD3	1: Running frequency (×100, Hz) 2: Set frequency (×100, Hz)	0–47	0
P15.15	Transmitted PZD4	3: Bus voltage (×10, V) 4: Output voltage (×1, V)	0–47	0
P15.16	Transmitted PZD5	5: Output current (×10, A) 6: Actual output torque (×10, %)	0–47	0
P15.17	Transmitted PZD6	7: Actual output power (x10, %) 8: Rotating speed of the running (x1,	0–47	0
P15.18	Transmitted PZD7	RPM) 9: Linear speed of the running (x1, m/s)	0–47	0
P15.19	Transmitted PZD8	10: Ramp frequency reference 11: Fault code	0–47	0
P15.20	Transmitted PZD9	12: Al1 value (×100, V) 13: Al2 value (×100, V)	0–47	0
P15.21	Transmitted PZD10	14: Al3 value (×100, V) 15: HDIA frequency (×1000, kHz)	0–47	0
P15.22	Transmitted PZD11	16: Terminal input state 17: Terminal output state	0–47	0
P15.23	Transmitted PZD12	 18: PID reference (x10, %) 19: PID feedback (x10, %) 20: Rated torque of the motor 21: MSB of position reference (signed number) 22: LSB of position feedback (signed number) 23: MSB of position feedback (unsigned number) 24: LSB of position feedback (unsigned number) 25: Status word 26: HDIB frequency value (x1000, kHz) 	0–31	0

Function code	Name	Parameter description	Setting range	Default value
		27: MSB of PG card pulse feedback		
		count		
		28: LSB of PG card pulse feedback		
		count		
		29: MSB of PG card pulse reference		
		count		
		30: LSB of PG card pulse reference		
		count		
		31: Function code mapping		
		(PZD2–PZD12 correspond to		
		P14.60–P14.70 respectively.)		
		32: Status word 3		
		33–47: Reserved		
D45.05	DP			5.0.
P15.25	communication	0.0 (invalid)–60.0s	0.0–60.0	5.0s
	timeout time			
P15.26	CANopen communication	0.0 (invalid)–60.0s	0.0-60.0	5.0s
P 15.20	timeout time	0.0 (Invalid)-60.0s	0.0-60.0	5.05
	timeout time	0: 1Mbps		
		1: 800K bps		
		2: 500K bps		
	CANopen	3: 250K bps		
P15.27	communication	4: 125K bps	0–7	3
	baud rate	5: 100K bps		
		6: 50K bps		
		7: 20K bps		
	CAN			
P15.28	communication	0–127	0-127	1
F 15.20	address	0-127	0-127	1
	address			
		0: 50K bps		
		1: 100K bps		
	CAN baud rate	2: 125K bps		
P15.29	setting	3: 250K bps	0–5	2
	5	4: 500K bps		
		5: 1M bps		
			1	

Function code	Name	Parameter description	Setting range	Default value
P15.30	CAN communication timeout time	0.0 (invalid)–60.0s	0.0–60.0	5.0s
P15.31	DeviceNet communication timeout time	0.0 (invalid)–60.0s	0.0–60.0	5.0s
P15.41	BACnet communication mode selection	0: P15.46 is valid. 1: P15.42 is valid.	0–1	0
P15.42	BACnet_MSTP baud rate	0–5 0: 9600bps 1: 19200bps 2: 38400bps 3: 57600bps 4: 76800bps 5: 115200bps	0–5	0
P15.43	Communication control word expression format	0: Decimal format 1: Binary format	0–1	0
P15.44		0: Display currently identified card (only one) 1: DP card 2: CANopen card 3: PROFINET card 4: Ethernet IP card 5: Modbus TCP card 6: EtherCAT card	0–6	0
P15.45- P15.69	Reserved			
P16.01	Reserved			
P16.02	Ethernet monitoring card IP address 1	0–255	0–255	192
P16.03	Ethernet monitoring card IP address 2	0–255	0–255	168

Function code	Name	Parameter description	Setting range	Default value
P16.04	Ethernet monitoring card IP address 3	0–255	0–255	0
P16.05	Ethernet monitoring card IP address 4	0–255	0–255	1
P16.06	Ethernet monitoring card subnet mask 1	0–255	0–255	255
P16.07	Ethernet monitoring card subnet mask 2	0–255	0–255	255
P16.08	Ethernet monitoring card subnet mask 3	0–255	0–255	255
P16.09	Ethernet monitoring card subnet mask 4	0–255	0–255	0
P16.10	Ethernet monitoring card subnet gateway 1	0–255	0–255	192
P16.11	Ethernet monitoring card subnet gateway 2	0–255	0–255	168
P16.12	G Ethernet monitoring card subnet gateway 3	0–255	0–255	1
P16.13	Ethernet monitoring card subnet gateway 4	0–255	0–255	1
P16.14	Ethernet card monitoring variable address 1	0x0000-0xFFF	0000– FFFF	0x0000

Function code	Name	Parameter description	Setting range	Default value
P16.15	Ethernet card monitoring variable address 2	0x0000–0xFFFF	0000– FFFF	0x0000
P16.16	Ethernet card monitoring variable address 3	0x0000-0xFFFF	0000– FFFF	0x0000
P16.17	Ethernet card monitoring variable address 4	0x0000-0xFFFF	0000– FFFF	0x0000
P16.18– P16.23	Reserved			
P16.24	Expansion card identification time of slot 1	0.0–600.0s When this parameter is set to 0.0, disconnection fault detection is not performed.	0.0–600. 0	0.0s
P16.25	Expansion card identification time of slot 2	0.0–600.0s When this parameter is set to 0.0, disconnection fault detection is not performed.	0.0– 600.0	0.0s
P16.26	Expansion card identification time of slot 3	0.0–600.0s When this parameter is set to 0.0, disconnection fault detection is not performed.	0.0– 600.0	0.0s
P16.27	Expansion card communication timeout time of slot 1	0.0–600.0s When this parameter is set to 0.0, disconnection fault detection is not performed.	0.0– 600.0	0.0s
P16.28	Expansion card communication timeout time of slot 2	0.0–600.0s When this parameter is set to 0.0, disconnection fault detection is not performed.	0.0– 600.0	0.0s
P16.29	Expansion card communication timeout time of slot 3	0.0–600.0s When this parameter is set to 0.0, disconnection fault detection is not performed.	0.0– 600.0	0.0s

Function code	Name	Parameter description	Setting range	Default value
P16.30	Reserved		•	
P16.31	PROFINET communication	0.0 (invalid)–60.0s	0.0-60.0	5.0s
1 10.51	timeout time	0.0 (111/2110) -00.03	0.0-00.0	0.03
P16.32	Received PZD2	0: Invalid	0–31	0
P16.33	Received PZD3	1: Set frequency (0-Fmax, unit: 0.01	0-31	0
P16.34	Received PZD4	Hz)	0-31	0
P16.35	Received PZD5	2: PID reference (-1000–1000, in which	0-31	0
P16.36	Received PZD6	1000 corresponds to 100.0%)	0-31	0
P16.37	Received PZD7	3: PID feedback (-1000–1000, in which	0-31	0
P16.38	Received PZD8	1000 corresponds to 100.0%)	0-31	0
P16.39	Received PZD9	4: Torque setting (-3000-+3000, in	0-31	0
P16.40	Received PZD10	which 1000 corresponds to 100.0% of	0–31	0
P16.41	Received PZD11	the rated current of the motor)	0–31	0
P16.42	Received PZD12	running frequency (0–Fmax, unit: 0.01 Hz) 6: Setting of the upper limit of reverse running frequency (0–Fmax, unit: 0.01 Hz) 7: Upper limit of the electromotive torque (0–3000, in which 1000 corresponds to 100.0% of the rated current of the motor) 8: Upper limit of the brake torque (0–3000, in which 1000 corresponds to 100.0% of the rated current of the motor) 9: Virtual input terminal command, 0x000–0x3FF (bit9–bit0 correspond to S8/S7/S6/S5/HDIB/HDIA/S4/S3/S2/S1 in sequence) 10: Virtual output terminal command, 0x00–0x0F (bit3–bit0 correspond to RO2/RO1/HDO/Y1 in sequence) 11: Voltage setting (for V/F separation) (0–1000, in which 1000 corresponds to 100.0% of the rated voltage of the	0–31	0

Function code	Name	Parameter description	Setting range	Default value
		motor) 12: AO1 output setting 1 (-1000-+1000, in which 1000 corresponds to 100.0%) 13: AO2 output setting 2 (-1000-+1000, in which 1000 corresponds to 100.0%) 14: MSB of position reference (signed number) 15: LSB of position reference (unsigned number) 16: MSB of position feedback (signed number) 17: LSB of position feedback (unsigned number) 18: Position feedback setting flag (position feedback can be set only after this flag is set to 1 and then to 0) 19: Function code mapping (PZD2-PZD12 correspond to P14.49-P14.59 respectively.) 20-31: Reserved		
P16.43	Transmitted PZD2	0: Invalid 1: Running frequency (×100, Hz)	0–47	0
P16.44	Transmitted PZD3	2: Set frequency (×100, Hz) 3: Bus voltage (×10, V)	0–47	0
P16.45	Transmitted PZD4	4: Output voltage (×1, V) 5: Output current (×10, A)	0–47	0
P16.46	Transmitted PZD5	6: Actual output torque (×10, %) 7: Actual output power (×10, %)	0–47	0
P16.47	Transmitted PZD6	8: Rotating speed of the running (×1, RPM)	0–47	0
P16.48	Transmitted PZD7	9: Linear speed of the running (×1, m/s) 10: Ramp frequency reference	0–47	0
P16.49	Transmitted PZD8	11: Fault code 12: Al1 value (×100, V)	0–47	0
P16.50	Transmitted PZD9	13: Al2 value (×100, V) 14: Al3 value (×100, V)	0–47	0
P16.51	Transmitted PZD10	15: HDIA frequency (×1000, kHz) 16: Terminal input state	0–47	0

Function code	Name	Parameter description	Setting range	Default value
P16.52	Transmitted PZD11	17: Terminal output state 18: PID reference (x10, %)	0–47	0
P16.53	Transmitted PZD12	 PID feedback (x10, %) Rated torque of the motor Rated torque of the motor MSB of position reference (signed number) LSB of position reference (unsigned number) MSB of position feedback (signed number) LSB of position feedback (unsigned number) Status word HDIB frequency value (x1000, kHz) Status word BS of PG card pulse feedback count LSB of PG card pulse reference count LSB of PG card pulse reference count LSB of PG card pulse reference count ISB of PG card pulse reference count ISB of PG card pulse reference count Struction code mapping (PZD2–PZD12 correspond to P14.60–P14.70 respectively.) Status word 3 A7: Reserved 	0-47	0
P16.54	EtherNet IP communication timeout time	0.0–60.0s	0.0–60.0s	5.0s
P16.55	EtherNet IP communication rate setting	0: Self-adaption 1: 100M full duplex 2: 100M half duplex 3: 10M full duplex 4: 10M half duplex	0–4	0
P16.56	Bluetooth pairing code	0–65535	0–65535	0

Function code	Name	Parameter description	Setting range	Default value
P16.57	Bluetooth host type	0: No host connection 1: Mobile APP 2: Bluetooth box 3–8: Reserved	0–8	0
P16.58	Industrial Ethernet communication card IP address 1	0–255	0–255	192
P16.59	Industrial Ethernet communication card IP address 2	0–255	0–255	168
P16.60	Industrial Ethernet communication card IP address 3	0–255	0–255	0
P16.61	Industrial Ethernet communication card IP address 4	0–255	0–255	20
P16.62	Industrial Ethernet communication card subnet mask 1	0–255	0–255	255
P16.63	Industrial Ethernet communication card subnet mask 2	0–255	0–255	255

Function code	Name	Parameter description	Setting range	Default value
P16.64	Industrial Ethernet communication card subnet mask 3	0–255	0–255	255
P16.65	Industrial Ethernet communication card subnet mask 4	0–255	0–255	0
P16.66	Industrial Ethernet communication card gateway 1	0–255	0–255	192
P16.67	Industrial Ethernet communication card gateway 2	0–255	0–255	168
P16.68	Industrial Ethernet communication card gateway 3	0–255	0–255	0
P16.69	Industrial Ethernet communication card gateway 4	0–255	0–255	1
P19.00	Type of card at slot 1	0: No card 1: Programmable card	0–50	0
P19.01	Type of card at slot 2	2: I/O card 3: Incremental PG card 4: Incremental PG card with UVW	0–50	0
P19.02	Type of card at slot 3	Incremental PG card with UVW Eithernet communication card Eithernet communication card Eithernet communication card Eithernet communication card Si Resolver PG card VIFI card I1: PROFINET communication card I2: Sine-cosine PG card without CD	0–50	0

Function code	Name	Parameter description	Setting range	Default value
		signals 13: Sine-cosine PG card with CD signals 14: Absolute encoder PG card (reserved) 15: CAN master/slave communication card 16: Modbus TCP communication card 17: EtherCAT communication card (reserved) 18: BACnet communication card (reserved) 19: DeviceNet communication card (reserved) 20: PT100/PT1000 temperature detection card 21: EtherNet IP communication card 21: EtherNet IP communication card 22: MECHATROLINK communication card (reserved) 23: Bluetooth card 2 24–31: Reserved 32: SSI-PG card 33–50: Reserved		



E-mail: overseas@invt.com.cn Website: www.invt.com

The products are owned by Shenzhen INVT Electric Co., Ltd. Two companies are commissioned to manufacture: (For product code, refer to the 2nd/3rd place of S/N on the name plate.) Shenzhen INVT Electric Co., Ltd. (origin code: 01) INVT Power Electronics (Suzhou) Co., Ltd. (origin code: 06) Address: INVT Guangming Technology Building, Songbai Road, Address: No. 1 Kunlun Mountain Road, Science & Technology Matian, Guangming District, Shenzhen, China Town, Gaoxin District, Suzhou, Jiangsu, China Industrial Automation: HMI PLC VFD Servo System Elevator Intelligent Control System Rail Transit Traction System Energy & Power: UPS DCIM Solar Inverter SVG New Energy Vehicle Powertrain System New Energy Vehicle Charging System New Energy Vehicle Motor

Copyright© INVT.

Manual information may be subject to change without prior notice.