Ethernet Digital Signal Data Acquisition Isolation Transmitter

6-channel digital signal input, Ethernet RJ45 & Relay Isolation Output Device SYDDN-RJ45 Series

Features	Applications
●Support 100/10Mbps Ethernet port & MODBUS TCP. ●6-channel digital signal input and isolation, 6-channel relay and RJ45 signal output. ●Wide range power supply: 8-36VDC ●High reliability, easy to program and install. ●User can set address, baud rate by programming. ●Monitor and display data in industrial site. ●Refined external shape, high reliability, easy to program, 35mm DIN Rail-mounted package.	●Monitoring, measuring and remotely controlling the industrial field through local area network. ●Gathering 0-10V/4-20mA PLC/DCS/FCS signal and converting into Ethernet RJ45 output. ●Meters and instruments data acquisition and recording ●Monitoring operating data from electric equipment, transportation and other equipment system. ●Realize network operation in Environment protection equipment, security, medical equipment,etc.

Introduction:

SY DDN-RJ45 Series products can realize switch signal input and output conversion to control the device in remote places. SY DDN-RJ45 Series products can be applied in Ethernet industrial automatic controlling system and signal isolation, long-distance transmission, etc. The isolation transmitter has power isolation, signal input output isolation, Ethernet communication functions. The communication protocol is Modbus TCP, communication commands is integrated into the ADAM module, the parameters can be set by customer. It can be used with the controlling modules from other companies in the same can bus line to make it is easy to program.

SY DDN-RJ45 Series Isolation Transmitter is kind of intelligent monitoring and controlling system designed on the basis of MCU. The IP address, terminal no, status configuration signal are all deposited in the non-volatile EEPROM.

SY DDN-RJ45 series isolation transmitter is designed on the basis of industrial standards, the isolation voltage between signal input and output can withstand 3000VDC. It has high anti-interference ability, high reliability. The operating temperature range: - 25℃～+70℃.

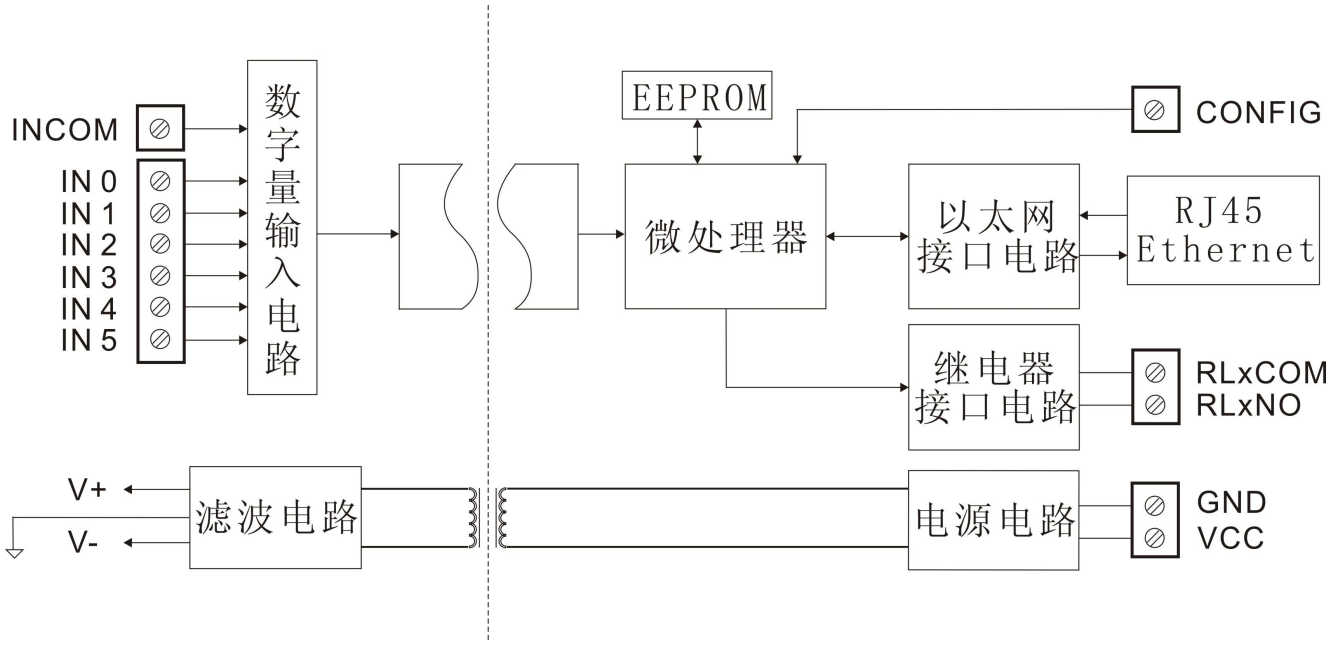


Fig.1 SY DDN-RJ45 Functional Block

SY DDN-RJ45 Function Introduction

SY DDN-RJ45 Relay output and isolated digital signal input module can be used to measure 6-channel switching quantity status and control 6-channel relay output, it is support Ethernet MODBUS TCP protocol.

1. Digital input

6-channel isolated digital input, support switch-contact signal and level signal, support counter function (the mode can be set by customer) and rising/falling latching functions.

2. Relay output

6-channel relay output, support independent switch controlling, power-on state setting, programmable pule width output.

3. Communication

Communication port: 1-channel standard 100/10Mbps Ethernet interface.
Communication protocol: support Modbus TCP, compatible with PLC, computer monitoring system in communication.

4. Anti-interference

Can set checksum based on the actual requirements. In side the module, TVS is used to efficiently control every kinds of surge pulse to protect the internal digital filter and restrain the industrial frequency interference.

SY DDN-RJ45 General Parameters:

(typical @ +25℃，Vs: 24VDC)

Input type: 6-channel isolation digital signal.
Output type: 6-channel relay.

Digital signal input:

Dry contact: GND or collector
Wet contact: logic level 0: +1V (max.)
logic level1: +4V～+30V
Isolation voltage: 3000V
Input channel can be used as 500Hz counter.

Relay output: (A Type)

Joint capacitor: 250VAC@1A
30VDC@2A
Relay well connection time: 7mS
Relay disconnection time: 3mS
Break down voltage: 500VAC
Insulation resistance:≥1000MΩ
Relay can be used as pulse output.

Input terminal protection: over-voltage protection, over-current protection

Communication: Ethernet, support standard Modbus TCP protocol, RJ45 net interface.
Parameters configurable

Operation power supply: +8 — 36 VDC wide range, with anti-reverse connection over-voltage protection circuit.
Power consumption: less than 2W
Operation temperature: - 45 — +80℃
Operation humidity: 10 — 90% (Non-condensation)
Storage temperature: - 45 — +80℃
Storage humidity 10 — 95% (Non-condensation)

Isolation voltage: Between input and output 3KVDC, 1minute, leakage current 1mA (output and power in common ground).
Impulse withstand voltage: 3KVAC, 1.2/50us (peak value)

Dimension: 83 mm x 37 mm x 51mm
Weight: about 105g

Pin Definition:

PIN	Definition	Description
1	**GND**	Power GND
2	**VCC**	Power +
3	**CONGFIG**	Configuration pin
4	**INCOM**	Input common terminal
5	**IN0**	Input channel 0
6	**IN1**	Input channel 1
7	**IN2**	Input channel 2
8	**IN3**	Input channel 3
9	**IN4**	Input channel 4
10	**IN5**	Input channel 5
11	**RL5COM**	Relay 5 COM terminal
12	**RL5NO**	Relay 5 NO terminal
13	**RL4COM**	Relay 4 COM terminal
14	**RL4NO**	Relay 4 NO terminal
15	**RL3COM**	Relay 3 COM terminal
16	**RL3NO**	Relay 3 NO terminal
17	**RL2COM**	Relay 2 COM terminal
18	**RL2NO**	Relay 2 NO terminal
19	**RL1COM**	Relay 1 COM terminal
20	**RL1NO**	Relay 1 NO terminal
21	**RL0COM**	Relay 0 COM terminal
22	**RL0NO**	Relay 0 NO terminal
23	**NC**	-
24	**RJ-45**	Ethernet port

Model Selection: SY DDN-RJ45 Series

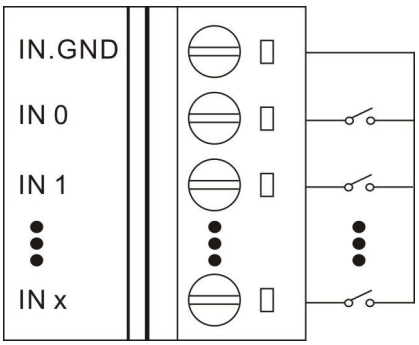
SY DD N□ — O□ — RJ45

- N□: Input/output channel
- O□: Output — A: Relay
- RJ45: Interface — RJ45: Ethernet

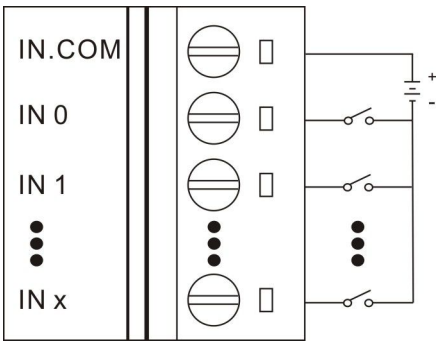
Fig. 2 SY DDN-RJ45 External View

SY DDN-RJ45 Module wiring diagram (Left side is connection methods for user reference)

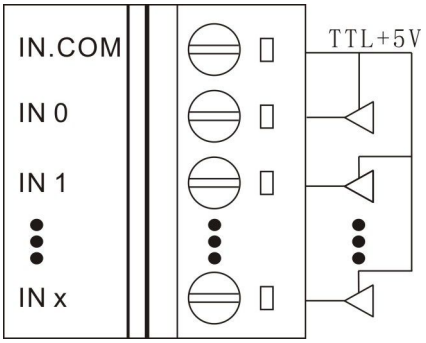
Dry contact signal input:



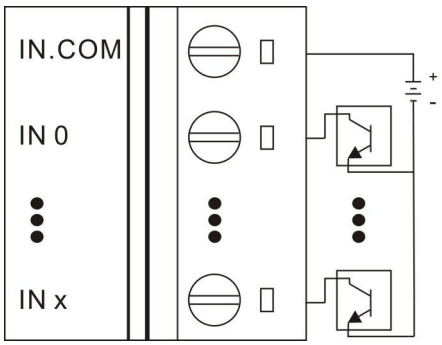
Wet contact signal input:

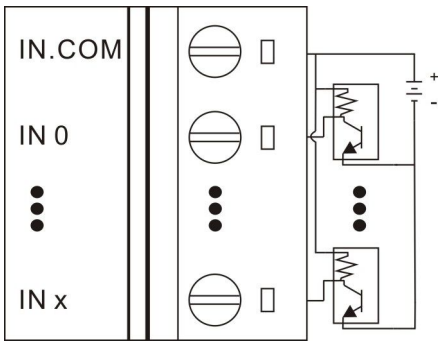


TTL/CMOS signal input:



Open collector signal input:

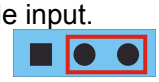




Hardware Jumper Setting:

1. The internal embedded jumper Pin1 and Pin2 are in short-circuited, the input mode is dry contact input.

2.The internal embedded jumper Pin2 and Pin3 are in short-circuited, the input mode is wet contact common anode input.



Initialization of SY DDN-RJ45 Module

For all the SYDDN-RJ45 module, every module must be allocated an unique IP address code and terminal no. The SYDDN-RJ45 module initial default setting before factory below:

IP Address: 192.168.0.225

Terminal NO: 2225

All the new module IP address are the same, the addresses will be contradictory with each other without configuration, so when users setting up the system, the re-configuration on IP address and terminal no for each module must be done. Users can modify SYAD08-RJ45 module's parameters through setting commands after well connecting module power cable and communication cable. Before the adjusting parameters, user must let the module be in the default state firstly; otherwise it can not be modified.

How to Enter into Configuration State

SYDDN-RJ45 module has a special terminal/pin CONFIG. Set the CONFIG terminal/pin and GND pin into short-circuit connection mode, then power on, thus the module is in configuration state. The details below:

MAC Address: 51.51.51.51.51.51
IP Address: 192.168.0.80
Terminal No: 80
Subnet Mask: 255.255.255.0
Default gateway: 192.168.0.1

User can modify the parameters of SYDDN-RJ45 through the setting commands. If the detailed parameters setting state of a module are required, user can set the module into configuration state by add line jumper, then do the configuration again.

Modbus TCP Communication Protocol

Introduction

Modbus TCP Protocol is another version of Modbus protocol, which is developed in 1999 to allow internet users to access to Ethernet equipment. Modbus TCP Protocol is open, easy to learn and understand, so it become one the world leading industrial Ethernet Protocols at present.

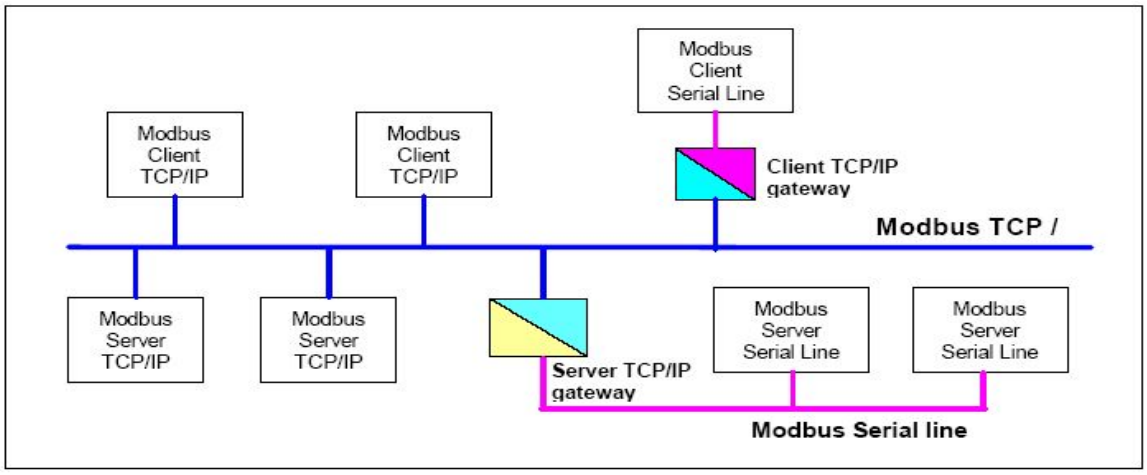


Fig.8 Industrial Ethernet Equipments Network

The protocol defines a message structure which can be recognized by the controller regardless of the network used in communication. It describes the process for controllers' request to access to other equipments,such as, response to the request of other equipments,how to detect errors and record them.It formulates the message domain format and the universal format of contents.

In Modbus network communication, the protocol decides that every controller must know its equipments address, recognize the message sent based on address and decide taking what kinds of actions.If response required,the controller will generate feedback information and send through Modbus Protocol. In other network,the messages with Modbus Protocol are converted into frame or package structures used in that network. The conversion also diversifies the methods to solve the problems on bit address,routing path and error detection.

1. Modbus TCP Mode

MBAP Header	Function Code	Data Quantity	Data 1	……	Data n

Protocol Head Functional Domain Data Domain

Frame Alignment: In Modbus TCP mode, the address of a module is determined by the IP address, so there is no address domain contents. And TCP network is the reliable data transmission network, so there is no calibration data. But because the in-conformity of the arriving order between the data from IP network and the data expected, a data serial number is added. And the Modbus TCP carries the Modbus protocol, so an address domain is added in the leading data.

MBAP Header and Protocol Head include the domains below:

Domain	Length	Description	Client Server	Server
Transaction Elements Identifier	2 bytes	MODBUS request/ response code	Client server start-up	Server re-copy from the request it received
Protocol Identifier	2 bytes	0=MODBUS Protocol	Client server start-up	Server re-copy from the request it received
Length	2 bytes	The quantity of the bytes below	Client server start-up(request)	Server (Response)start-up
Unit Identifier	1 byte	Serial link identifier	Client server start-up	Server re-copy from the request it received

Header format has 7 bytes.
Transaction Process Identifier: used for affairs process and match operation. In response, Modbus server copies transaction process identifier of request.
Protocol Identifier: used in multiple channels in the system. Identify the Modbus protocol.
Length: length domain is the byte count of next domain, which includes unit identifiers and the data domain.
Unit Identifier: the route of the system uses the domain. Achieve the connection between MODBUS/MODBUS and sub-webs of serial links through the gateway between Ethernet TCP-IP network and MODBUS serial links. MODBUS client machine sets the domain in request, the server must use the same value to return to this domain in response.

2. SYDDN-RJ45 Supports Commands below

At present, all the Ethernet Data Acquisition Modules from SUNYUAN support MODBUS TCP Protocol. The function codes (hexadecimal) supported mainly include:

01 Read the relay state/status
02 Read switching value input
03 Read holding register
05 Set a single relay
06 Set single holding register
0F Set multiple relays
10 Set multiple holding register
14 Read files record
15 Write file record

3. Command and Address Mapping Table

(1) Read the relay state/status

Function Code: 01
Instruction: Read output relay state.
Data Explanation:.

Address	Description	Note
00000	Channel1 Switch quantity current output state	=1High level: =0Low level
00001	Channel 2 Switch quantity current output state	=1High level: =0Low level
00002	Channel 3 Switch quantity current output state	=1High level: =0Low level
00003	Channel 4 Switch quantity current output state	=1High level: =0Low level
00004	Channel 5 Switch quantity current output state	=1High level: =0Low level
00005	Channel 6 Switch quantity current output state	=1High level: =0Low level
00006-00015	<Hold>	
00016	Channel 1 Switch quantity output power on state	=1High level: =0Low level
00017	Channel 2 Switch quantity output power on state	=1High level: =0Low level

00018	Channel 3 Switch quantity output power on state	=1High level: =0Low level
00019	Channel 4 Switch quantity output power on state	=1High level: =0Low level
00020	Channel 5 Switch quantity output power on state	=1High level: =0Low level
00021	Channel 6 Switch quantity output power on state	=1High level: =0Low level
00022-00031	<Hold>	
00032	Channel 1 Switch quantity output security state	=1High level: =0Low level
00033	Channel 2 Switch quantity output security state	=1High level: =0Low level
00034	Channel 3 Switch quantity output security state	=1High level: =0Low level
00035	Channel 4 Switch quantity output security state	=1High level: =0Low level
00036	Channel 5 Switch quantity output security state	=1High level: =0Low level
00037	Channel 6 Switch quantity output security state	=1High level: =0Low level
00038-00039	<Hold>	

Modbus Request

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x01
Leading Address	2byte	0x0000 to 0xFFFF
Read Quantity	2byte	0x0000 to 0x0006

Modbus Response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x01
Byte Count	1byte	n
Coil State	2byte	

Error Response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x01+0x80
Error Code	1byte	0x01 ～ 0x07

Examples:

ⅰ.Request: Read Switch quantity current output state

00 00 00 00 00 06 00 01 00 00 00 06

Response:

00 00 00 00 00 04 00 01 01 3F

(2) Read switch quantity input

Function code: 02

Explanation: Read input switch quantity state

Data explanation:

Address	Description	Note
10000	Channel 1 Switch quantity input state	=0 power off:=1 power on
10001	Channel 2 Switch quantity input state	=0 power off:=1 power on

10002	Channel 3 Switch quantity input state	=0 power off:=1 power on
10003	Channel 4 Switch quantity input state	=0 power off:=1 power on
10004	Channel 5 Switch quantity input state	=0 power off:=1 power on
10005	Channel 6 Switch quantity input state	=0 power off:=1 power on
10006-10015	<Hold>	
10016	Channel 1 rising latching state	=0 no latch:=1 have latch
10017	Channel 2 rising latching state	=0 no latch:=1 have latch
10018	Channel 3 rising latching state	=0 no latch:=1 have latch
10019	Channel 4 rising latching state	=0 no latch:=1 have latch
10020	Channel 5 rising latching state	=0 no latch:=1 have latch
10021	Channel 6 rising latching state	=0 no latch:=1 have latch
10022-10031	<Hold>	
10032	Channel 1 falling latching state	=0 no latch:=1 have latch
10033	Channel 2 falling latching state	=0 no latch:=1 have latch
10034	Channel 3 falling latching state	=0 no latch:=1 have latch
10035	Channel 4 falling latching state	=0 no latch:=1 have latch
10036	Channel 5 falling latching state	=0 no latch:=1 have latch
10037	Channel 6 falling latching state	=0 no latch:=1 have latch
10038-10039	<Hold>	

Modbus Request

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x02
Leading Address	2byte	0x0000 to 0xFFFF
Reading quantity	2byte	0x0000 to 0x0006

Modbus Response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x02
Byte Count	1byte	n
Input State	2byte	

Error Response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x02+0x80
Error Code	1byte	0x01 ～ 0x07

Examples:

ⅰ.Request: Read switch quantity input state.

00 00 00 00 00 06 00 02 00 00 00 06

Response:

00 00 00 00 00 04 00 02 01 3F

(3) Read holding register

Function code: 03

Explanation: Read holding register value.

Data explanation: read the 16-bit integer or no-sign integers.

Address	Description	Note
400000	Channel1 digital pulse output low level width	hexadecimal format（16 bit）
400001	Channel1 digital pulse output high level width	hexadecimal format（16 bit）
400002	Channel2 digital pulse output low level width	hexadecimal format（16 bit）
400003	Channel2 digital pulse output high level width	hexadecimal format（16 bit）
400004	Channel3 digital pulse output low level width	hexadecimal format（16 bit）
400005	Channel3 digital pulse output high level width	hexadecimal format（16 bit）
400006	Channel4 digital pulse output low level width	hexadecimal format（16 bit）
400007	Channel4 digital pulse output high level width	hexadecimal format（16 bit）
400008	Channel5 digital pulse output low level width	hexadecimal format（16 bit）
400009	Channel5 digital pulse output high level width	hexadecimal format（16 bit）
400010	Channel6 digital pulse output low level width	hexadecimal format（16 bit）
400011	Channel6 digital pulse output high level width	hexadecimal format（16 bit）
40012-40031	<Hold>	
40032	Channel1 switch quantity pulse count low bit	hexadecimal format（16 bit）
40033	Channel1 switch quantity pulse count high bit	hexadecimal format（16 bit）
40034	Channel2 switch quantity pulse count low bit	hexadecimal format（16 bit）
40035	Channel2 switch quantity pulse count high bit	hexadecimal format（16 bit）
40036	Channel3 switch quantity pulse count low bit	hexadecimal format（16 bit）
40037	Channel3 switch quantity pulse count high bit	hexadecimal format（16 bit）
40038	Channel4 switch quantity pulse count low bit	hexadecimal format（16 bit）
40039	Channel4 switch quantity pulse count high bit	hexadecimal format（16 bit）
40040	Channel5 switch quantity pulse count low bit	hexadecimal format（16 bit）
40041	Channel5 switch quantity pulse count high bit	hexadecimal format（16 bit）
40042	Channel6 switch quantity pulse count low bit	hexadecimal format（16 bit）
40043	Channel6 switch quantity pulse count high bit	hexadecimal format（16 bit）
40044-40063	<Hold>	
40064	Pule output enable register	Set bit: open :Clear bit: prohibit
40065	Input latching enable register	Set bit: open Clear bit: prohibit
40066	Input counter enable register	Set bit: open Clear bit: prohibit
40067	Input counter mode register	Set bit: rise Clear bit: fall
40068-40095	<Hold>	

Modbus request

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function code	1byte	0x03
Leading address	2byte	0x0000 to 0x00FF
Read quantity	2byte	1 to 125（0x7D0）

Modbus response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function code	1byte	0x03
Byte count	1byte	2n
Input state	2byte	

Error response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function code	1byte	0x03+0x80
Error code	1byte	0x01 ～ 0x07

Examples:

ⅰ. Request: Read digital 1-6 channel pulse output high-low level width

00 00 00 00 00 06 00 03 00 00 00 0C

Response:

00 00 00 00 00 1B 00 03 18 00 64 00 64 00 64 00 64 00 64 00 64 00 64 00 64 00 64 00 64 00 64 00 64 00 64

ⅱ. Request: Read switching value 1-6 channel pulse count value.

00 00 00 00 00 06 00 03 00 00 00 80

Response:

00 00 00 00 00 1B 00 03 18 30 30 00 36 30 30 30 30 00 32 46 46 54

55 58 10 00 25 08 B1 C0 A8 00 E1

⑷ Set a single relay

Function code: 05

Modbus request

Domain Name	Byte Count	Vaule
MBAP Header	7byte	
Function code	1byte	0x05
Set address	2byte	0x0000 to 0x0005
Set contents	2byte	0x0000 Release relay >=0x0001 Close/stick relay

Modbus response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x05
Set Address	2byte	0x0000 to 0x0005
Set Contents	2byte	0x0000 Release relay >=0x0001 Close/Stick relay

Error response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x05+0x80
Error Code	1byte	0x1 ～ 0x7

Examples:

ⅰ.Request: Set single relay (3rd-channel close or stick relay.)

00 00 00 00 00 06 00 05 00 02 FF 00

Response：
00 00 00 00 00 06 00 05 00 02 FF 00

⑸ Set single holding register.

Function Code: 06

Modbus request

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x06
Set Address	2byte	0x0000 to 0xFFFF
Set Contents	2byte	0x0000 to 0xFFFF

Modbus response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x06
Set Address	2byte	0x0040 to 0xFFFF
Set Contents	2byte	0x0000 to 0xFFFF

Error response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x06+0x80
Error Code	1byte	0x1 ～ 0x7

Examples

ⅰ.Request：
00 00 00 00 00 06 00 06 00 2B AA AA
Response：
00 00 00 00 00 06 00 06 00 2B AA AA

⑹ Set multiple relays

Function Code: 0F

Modbus request

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x0F
Set starting address	2byte	0x0000 to 0xFFFF
Set length	2byte	0x0000 to 0xFFFF
Byte count	1byte	n
Set Contents	2byte	

Modbus response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x0F
Set starting address	2byte	0x0000 to 0xFFFF
Set length	2byte	0x0000 to 0xFFFF

Error response

Domain Name	Byte Count	Value
MBAP Header	**7byte**	
Function Code	1byte	0x0F+0x80
Error Code	1byte	0x1 ～ 0x7

Examples:

ⅰ.Request:
00 00 00 00 00 09 00 0F 00 00 00 06 01 3F
Response:
00 00 00 00 00 06 00 0F 00 00 00 06

⑺ Set multiple holding registers

Function Code: 10

Modbus request

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x10
Set starting address	2byte	0x0000 to 0xFFFF
Set length	2byte	0x0000 to 0x0100
Byte count	1byte	2n
Set Contents	2nbyte	

Modbus response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x10
Set starting address	2byte	0x0000 to 0xFFFF
Set length	2byte	0x0000 to 0x0100

Error response

Domain Name	Byte Count	Value
MBAP Header	7byte	
Function Code	1byte	0x10+0x80
Error Code	1byte	0x1 ～ 0x7

Examples:

ⅰ.Request: Digital 1-6 channel pulse output high-low level width

00 00 00 00 00 1F 00 10 00 00 00 0C 18 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8

response：

00 00 00 00 00 06 00 10 00 00 00 0C

⑻ Read files records.

Function Code：14/06

Reading files records, in Modbus, it recognizes the file as a set of array consist of 16-bit byte string, the addressing mode is based on the the address. In files reading process, it has specified starting address to be read and the reading length, by changing the reading address and length, it can reach all the part of the file. The file has no specified name, only has serial no. This system only support read a single file.

EEPROM allocation

//file 0 0x0000 Digital output power-on value.
0x0001 Digital output safety value

//file 4 0x0006 Module operation mode setting parameters.

file 4 Description

Record No	Description	Note
0x0000	Serial port ID high-bit	（None）
0x0001	Baud rate code	（None）
0x0002	Measuring range high-bit	（None）
0x0003	ASCII data format LRC checking	（None）
0x0004	Communication protocol type	（None）
0x0005	Channel enable function	（None）
0x0006	NET MAC address high-bit	hexadecimal format
0x0007	NET MAC address middle-bit	hexadecimal format

0x0008	NET MAC address low-bit	hexadecimal format
0x0009	NET Terminal No	hexadecimal format
0x000A	NET IP address high-bit	hexadecimal format
0x000B	NET IP address low-bit	hexadecimal format
0x000C	Default gateway high-bit	hexadecimal format
0x000D	Default gateway GND bit	hexadecimal format
0x000E	Sub-net mask high-bit	hexadecimal format
0x000F	Sub-net mask GND bit	hexadecimal format

Modbus request

Domain Name	**Byte Count**	**Value**
MBAP Header	7byte	
Function Code	1byte	0x14
Byte count	1byte	0x07 to 0xFF
Sub-function Code	1byte	0x06
File No	2byte	0x0000 to 0x0007
Record No	2byte	0x0000 to 0x000F
Read length	2byte	n

Modbus response

Domain Name	**Byte Count**	**Value**
MBAP Header	7byte	
Function Code	1byte	0x14
Byte count	1byte	0x07 to 0xFF
Sub-function byte count	1byte	0x07 to 0xFF
Sub-function Code	1byte	0x06
Data	2nbyte	

Error response

Domain Name	**Byte Count**	**Value**
MBAP Header	7byte	
Function Code	1byte	0x14+0x80
Error Code	1byte	0x1 ～ 0x7

Examples:

ⅰ. Request: Read information of file 4.

00 00 00 00 00 0A 00 14 07 06 00 04 00 00 00 10

Response：

00 00 00 00 00 25 00 14 22 21 06 30 30 00 36 30 30 30 30 00 32 46 46 54 55 58 10 00 25 08 B1 C0 A8 00 E1 C0 A8 00 01 FF FF FF 00

(9) Write files records.

Function Code: 15/06

Modbus request

Domain Name	**Byte Count**	**Value**
MBAP Header	7byte	
Function Code	1byte	0x15
Byte count	1byte	0x07 to 0xFF
Sub-function Code	1byte	0x06
File No	2byte	0x0000 to 0x0007
Record No	2byte	0x0000 to 0x000F
Writing length	2byte	n
Data	2nbyte	

Modbus response

Domain Name	**Byte Count**	**Value**
MBAP Header	7byte	
Function Code	1byte	0x15
Byte count	1byte	0x07 to 0xFF
Sub-function Code	1byte	0x06
File No	2byte	0x0000 to 0x0007
Record No	2byte	0x0000 to 0x000F
Writing length	2byte	n
Data	2nbyte	

Error response

Domain Name	**Byte Count**	**Value**
MBAP Header	7byte	
Function Code	1byte	0x15+0x80
Error Code	1byte	0x1 ～ 0x7

Examples:

ⅰ. Request: Write the information of file 0 (modify digital output power-on value)

00 00 00 00 00 0c 00 15 09 06 00 00 00 00 00 01 00 0F

Response：

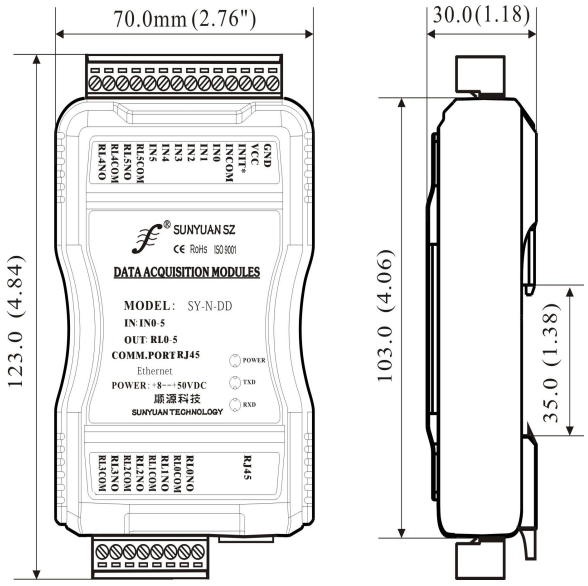
00 00 00 00 00 0C 00 15 09 06 00 00 00 00 00 01 00 0F

Error 提示码表

Response hint code	Description
01	Function Code Error (Not support change function or invalid function code)
02	Byte Count and data length are not match
03	Data format is not correct
04	Register address error
05	Data byte count error
06	The operating/working mode and the function are not match.
07	Sub-function no error

Dimension:

Unit:mm



*Note: The specification is subject to change without notice.