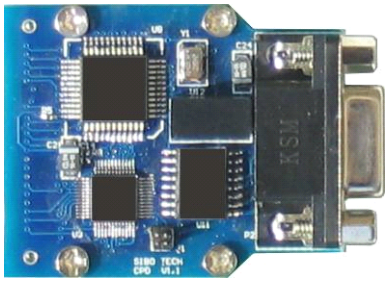
Compact Embedded PROFIBUS DP Module
CPD-511

User Manual



REV 1.6

SiboTech Automation Co., Ltd

Technical Support: +86-21-5102 8348
E-mail: support@sibotech.net

Table of Contents

1 About the Module

1.1 Product Function

Embedded PROFIBUS DP module CPD-511 is one of the compact embedded module series which with the same embedded interface (hardware) and interface protocol (software). Users only need to develop hardware and software once; the device can have a variety of field bus interfaces.

1.2 Feature

- Compact module with PROFIBUS DP standard DB9 connector
- Support PROFIBUS DPV1
- Support data record
- Support transferring parameter data
- Support diagnostic

1.3 Technical Specification

[1] The field bus side of embedded module is PROFIBUS DP V1 slave;

[2] PROFIBUS DP baud rate is self-adaptive, range: 9600～12Mbps;

[3] Input Bytes ≤244 Bytes;

Output Bytes ≤244 Bytes;

[4] The buffer length of data record (acyclic data) is 244 bytes;

[5] The buffer length of diagnostic is 244 bytes;

[6] The max parameter data is 54 bytes;

[7] The max configuration data is 64 bytes;

[8] 1KV isolated PROFIBUS interface;

[9] Power: 3.3VDC（3.0V-3.6V）, Power consumption is about 700mW;

[10] Working temperature: -20℃ to 60℃, Relative humidity: 5% to 95% (No condensation);

[11] Mechanical size: 49mm (L)*40mm (W)*20mm (H);

[12] CPD-511 support UART interface only;

1.4 Related Products

Related products include PM-160, PM-125, and PM-127 and so on.

More information about these products, please visit: http://www.sibotech.net/En/, or dial technical support line: +86-21-5102 8348

1.5 Terms

PROFIBUS DP：PROFIBUS DP protocol

1.6 Versions

◆Rev1.6, April 2013

[1] Modify the example message on page 31;

◆Rev1.5, March 2013

[1] Increase the tolerance range about the module voltage;

[2] Increase the process of product development;

[3] Increase the waiting time after reset instructions;

[4] Increase the description of the error in the initialization of CPD - 511;

[5] Increase the IO cycle DP plug pulls the plug, feedback and treatment of CPD – 511.

◆Rev1.4, September 2012

[1]Change the Power to 700mW ;

[2] Increase the pin tips ;

[3] Modify the connection packet and tips .

◆Rev1.3, May 2011

[1] Increase the illustration of "connection message";

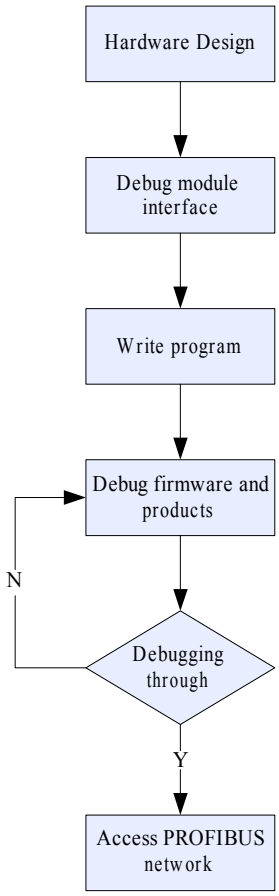
◆Rev1.2, April 2011

[1] Add the illustration of the mechanical size;

◆Rev1.1, March 2011

[1] Add the illustration of the pins of embedded interface.

2 Product Development Process



According to the instruction of CPD -511 "Chapter 3:Hardware Description".Dimensions of the CPD-511 interface pin definitions , PROFIBUS connector definition , four operating modes , complete the user hardware schematics , determine the mode of operation , design PCB diagram according to the actual situation . Can refer to the development board schematics and the development board package library.

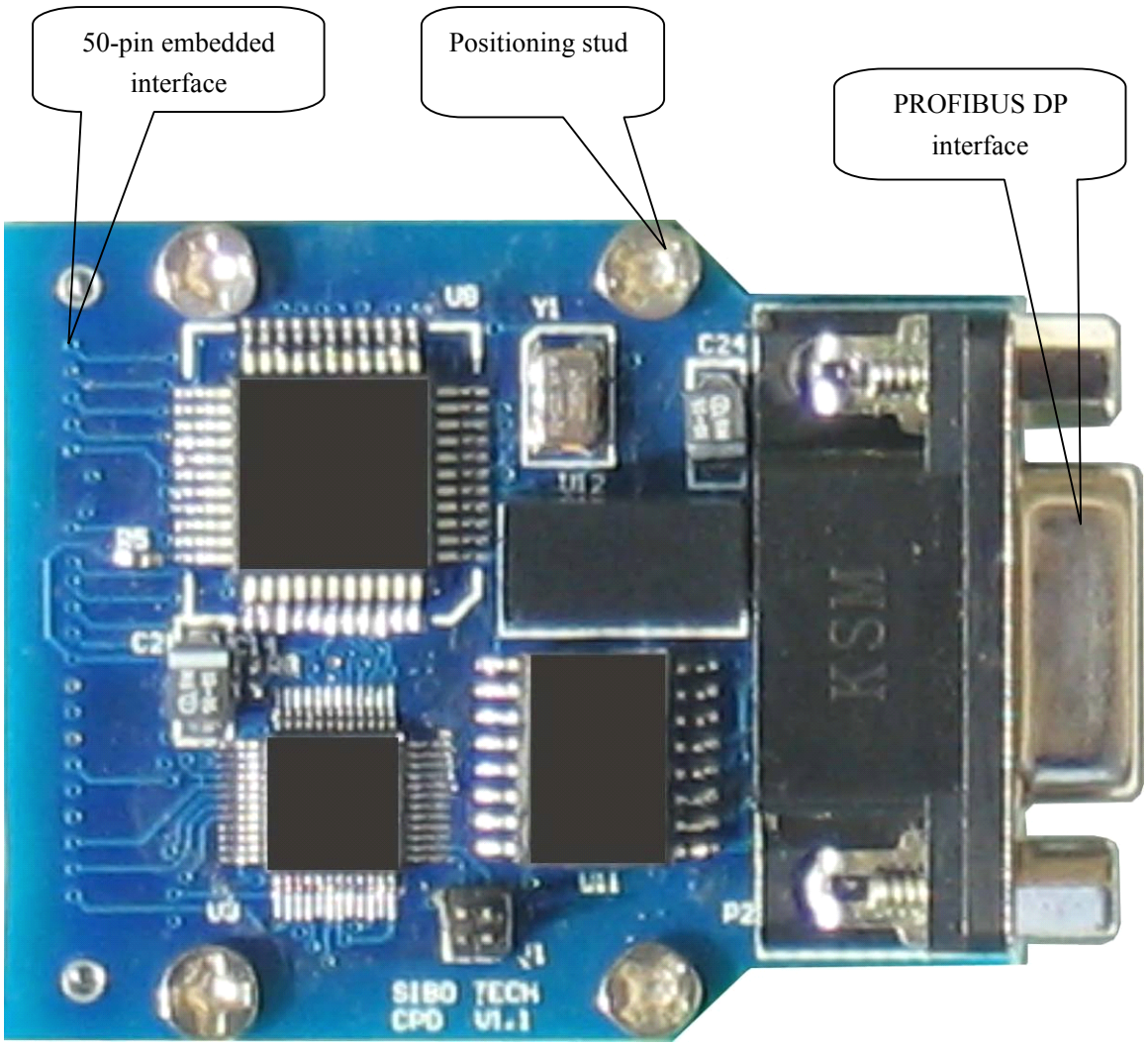
Preliminary debugging ,mainly to check user templates and CPD -511 interface serial data is normal

Refer to " Chapter 5 : connection packets ",and development board routines ,Understanding of serial communication message format , send and receive message content , error recovery mechanism and the entire process from the power-on reset initializes to IO communication, write your own firmware on CPD - 511 of the initialization and IO communication.

Establish the debugging systems , including the DP master , debugging between CPD - 511 and user templates . Reference CPD - 511 Manual "Chapter 3 : Hardware Description" and " Chapter 4: Software Description "Understanding LED lights case and CPD - 511 can achieve the function , major debugging previous firmware and discover the possible hardware problems .

3 Hardware Description

3.1 External View

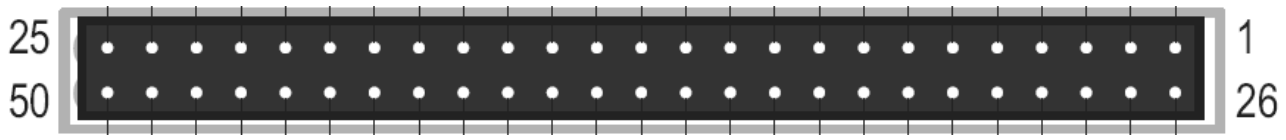


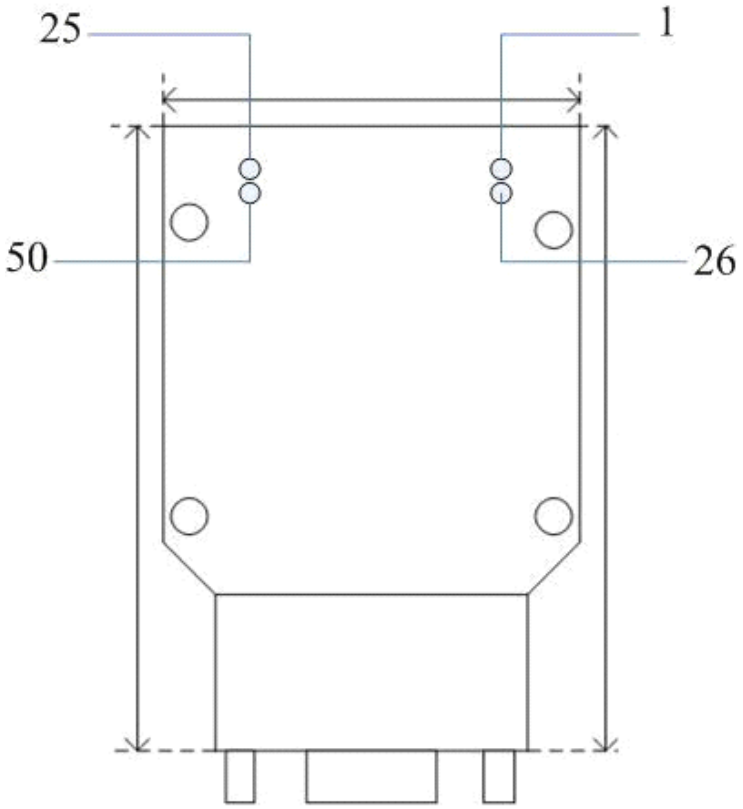
Note: This picture is only for reference and the product appearance should prevail in kind.

3.2 Embedded Interface

Embedded interface uses 50-pin connector socket.

3.2.1 Interface Signals





Note: components face down

Signals are defined as follows:

Pin	Signal	Type	Function
1	VSS	PWR	GND
2	Reserve	O	Reserve
3	TX	O	UART interface
4	LED1B	O	LED output network status

5	LED2B	O	LED output network status
6	SCLK	I	SPI interface
7	SS	O/I	SPI interface
8	RESET-	I	Reset signal, active low
9	IRQ-	O	Interrupt signal of parallel port
10	CE-	I	Control signal of parallel port
11	OM1	I	Operating mode
12	VSS	PWR	GND
13	VDD	PWR	Power: +3.3V DC
14	D0	BI	Data Bus
15	D2	BI	Data Bus
16	D4	BI	Data Bus
17	D6	BI	Data Bus
18	A13	I	Address Bus
19	A11	I	Address Bus
20	A9	I	Address Bus
21	A7	I	Address Bus
22	A5	I	Address Bus
23	A3	I	Address Bus
24	A1	I	Address Bus
25	Reserve	O	Reserve
26	Reserve	O	Reserve
27	Reserve	O	Reserve
28	RX	I	UART interface
29	LED1A	O	LED output network status
30	LED2A	O	LED output network status

31	MOSI	I	SPI interface
32	MISO	O	SPI interface
33	OE-	I	Control signal of parallel port
34	R/W-	I	Control signal of parallel port
35	OM2	I	Operating Mode
36	OM0	I	Operating Mode
37	VSS	PWR	GND
38	VDD	PWR	Power: +3.3V DC
39	D1	BI	Data Bus
40	D3	BI	Data Bus
41	D5	BI	Data Bus
42	D7	BI	Data Bus
43	A12	I	Address Bus
44	A10	I	Address Bus
45	A8	I	Address Bus
46	A6	I	Address Bus
47	A4	I	Address Bus
48	A2	I	Address Bus
49	A0	I	Address Bus
50	VSS	PWR	GND

I: Input; O: Output; BI: Bidirectional, Tri-state; PWR: Power input.

Notes:

1. The TX and RX pin of CPD-511 must be cross-connected to the user's MCU chip, that is to say: The TX of CPD -511 connected to the RX of MCU; The RX of CPD-511 connected to the TX of MCU. If the user's MCU is 5V, CPD-511 is 3.3V, proposed to increase in the level conversion circuit.

2. Initialization, set the RESET pin of CPD-511 to low, it is recommended to set low time not less than 100ms, start with the RESET pin goes high , wait another 100ms for CPD-511 successfully reset . In other words,

during the RESET pin is set low to the user board begin to sent packets to CPD-511, the interval of time is at least 200ms.

3.2.2 Signals Description

3.2.2.1 Operating mode

OM——Operating Mode

Module reads the state of these signals when starting to determine the module's operating mode.

Operating modes	OM2	OM1	OM0	Descriptions
0	0	0	0	Using configuration of non-volatile memory to start
1	0	0	1	parallel port(not yet to achieve)
2	0	1	0	UART, automatic baud rate
3	0	1	1	UART, 9600bps
4	1	0	0	UART, 115200bps
5	1	0	1	SPI(not yet to achieve)
6	1	1	0	Reserve
7	1	1	1	Reserve

3.2.2.2 LED output network status

Note that these LED output signals can't directly drive the LED lamps. You need use transistor devices to drive LED lamps.

High level signal light LED lamp and low off the LED lamp. Circuit design is shown in the board schematic.

Network status LED output is defined as follows:

Name	Color	Status	Meaning
LED1A	Green	Keep green	Online and in a state of data exchange

		Blinking	Online and not in a state of data exchange
LED1B	Red	Keep green	Offline status
LED2A	Green	Keep green	working properly
LED2B	Red	Keep green	Unrecoverable fault

3.2.2.3 UART interface

UART interface using the following settings:

8 data bits, no parity, 1 stop bit

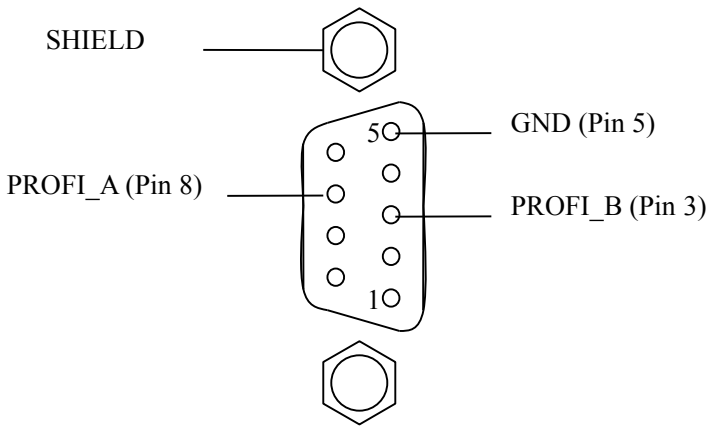
Range of baud rate: 2.4Kbps ~ 1Mbps

The setting mode of baud rate is determined by the OM [0~2] signals.

When baud rate of UART is self-adaptive:

1. User board send 0x55 and send cycle range is 10ms to 500ms, The recommended value is 10ms;
2. When embedded module intercept the baud rate and correctly receive 0x55, it will reply 0xAA(You need to send at least two 0x55);
3. Automatically intercept function of baud rate finish.

3.3 PROFIBUS DP Interface



PROFIBUS DP interface with DB9 female connector, the pin is defined as follows：

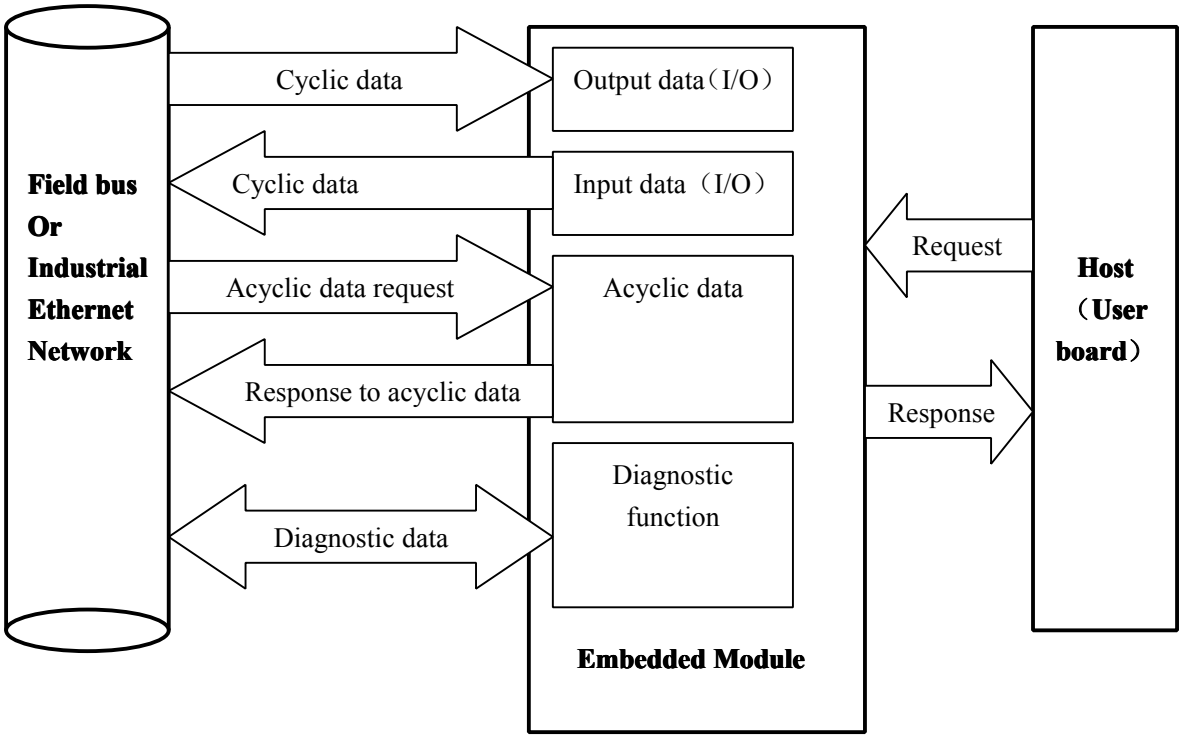
Pin	Signal Description
3	PROFI_B，positive data （must be taken）
4	RTS
5	GND
6	+5V output
8	PROFI_A，negative data （must be taken）
Bolt	SHIELD，Shield of bus cable

PROFI_B（pin 3）,PROFI_A（pin 8）and SHIELD must be taken; RTS（pin 4）can be used to determine the transmission direction of some equipment; +5V（pin 6）and GND （pin 5） can be used in bus terminal and can provide power for fiber transceiver. The maximum output current of pin 5 and pin 6 is 80mA.

4 Software Description

4.1 Summary

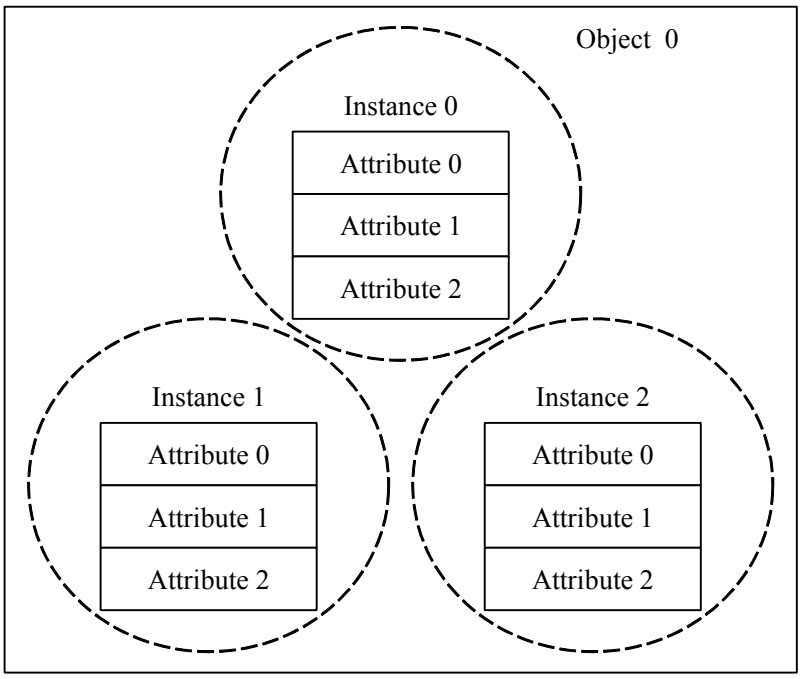
The basic function of general industrial network is the periodic I/O data exchange (cyclic data). Advanced networking functions for the advanced industrial data network also has the aperiodic data (acyclic data), diagnostic, alarm and other functions.



4.2 Object Model

Object model includes objects, instances, attributes.

Relevant information and services are added to the entity called “object”. Each object can save sub-entity called “instance”, which can contain a number of regions called “attribute” field.



4.2.1 Access Method

Objects and their contents can be accessed through using the object messages. Each object message is marked by objects, instances and attributes to appoint the message-related data or settings.

e.g.:

Instance 1 includes an attribute (attribute 2) called “firmware version” in the embedded module. User program can send a “get” request to the object 1(the basic object) instance 1 attribute 2(firmware version) to the module to get the firmware version.

This access method adapt to bidirectional communication. Embedded module and the host application must be able to analyst the object request and rout them to the corresponding program.

4.2.2 Object Types

Divided into two types:

- Embedded module object

These objects are part of the embedded module firmware, usually controlling the behavior of the module and

the actions on the network.

- Host application object

Embedded modules can access these objects located in the host application. This means that the host application must correctly handle the object request.

4.2.3 Standard Objects to Achieve

A standard object application has been designed to meet the need of all the main network system. Generally speaking, it is enough to support the implementation of these objects regardless of the type of network to get enough function.

Network interface of objects show as follows:

Embedded module object

Embedded module firmware contains the following objects:

- Basic object
- Diagnostic object
- Network object
- Network configuration object

Host application object

Host application contains the following objects:

- Application configuration object
- Application data object

4.3 Host Communication Layer

4.3.1 Fundamentals

Host interface communication is based on half-duplex protocol in the bottom of the embedded interface.

Message exchanges in the way of a “ask and reply”. Communication is initiated by the host and embedded modules are responded passively.

4.3.2 Messages

Each frame message consists of three parts:

1) Header

4 bytes

Byte 0——Real-time header (transaction number), which is “1” of the first message, plus one for each update, range is 0 to 255;

Real-time header:

1. Embedded Module->User Board
   a) Output data（I/O） changes to plus one;
   b) Whether the message command or message data changes or not to plus one;
2. User Board ->Embedded Module
   a) Suggest Input data（I/O） changing to plus one;
   b) Suggest Whether the message command or message data changing or not to plus one;
   c) It may increase the burden on the embedded module if don’t handle it.

Byte 1——Message data flag;

Bit 0——The current packet contains a message command;

Bit 1——The current packet contains a message data;

Bit 2——Reserve, 0;

Bit 3——Reserve, 0;

Bit 4——Reserve, 0;

Bit 5——Reserve, 0;

Bit 6——Reserve, 0;

Bit 7——Error flag.

Byte 2——The range of the high-byte of data length is 0 to 320. The maximum number of bytes for

some bus in the bus protocol is less than 320 bytes, and then the value is the maximum in the bus protocol;

Byte 3——Low-byte of data length

2) Data

Contains the corresponding data according to the data header

3) Check

Two bytes, 16-bit CRC

4.4 Message

Messages can be divided into commands and data. Host and embedded module can send a command (the message command frame) and data (message data frame) in the message level.

4.4.1 Message Structure

A message contains 6 bytes header and data.

Header definitions:

Byte 0——Command code;

Byte 1——Object code;

Byte 2——High-byte of instance;

Byte 3——Low-byte of instance;

Byte 4——High-byte of attribute;

Byte 5——Low-byte of attribute;

4.4.2 Command Code

Command Code	Command name
01H	Read attribute value
02H	Write attribute value

4.4.3 Error Code

Error code	Name
01H	Message flag error
02H	Object ’t exist
03H	Instance ’t exist
04H	Attribute ’t exist
05H	The target object ’t not support the command
06H	Data length error
07H	Invalid data

4.4.4 Message Handling

1. Messaging principle
   a) Communications are initiated by the user board, embedded modules respond.
   b) If the message flag is 0, it is I/O data.
   c) If the message command flag is true, it is message command.
   d) If the message data flag is true, it is message data.
   e) The user board can send message command when need to read and write contents of embedded modules.
   f) When embedded module need to read and write user board object contents, it need to set the message command in the response to input data sending by user board.

2. User board read and write the object contents of embedded module
   a) User board send message command
   b) Embedded module reply message data

3. Embedded module read and write the object contents of user board
   a) Embedded module send message command

b) User board reply message data

4. PROFIBUS parameter data

a) Embedded module send them to user board

b) User board response correct when get the parameter. If the response is error, reset the PROFIBUS communication.

5. PROFIBUS configuration data

a) Embedded module send them to user board

b) User board judge and respond, or respond directly without judging.

6. PROFIBUS data record (acyclic data)

a) Embedded module read the instance attribute values of acyclic data of user board

b) According to data length being read by PROFIBUS master to handle, if the reading-length is greater than or equal to the real length, then take the real length; if the reading-length is less than the real length, then take the reading-length.

4.5 Embedded Module Objects

4.5.1 Basic Object (Code 00H)

4.5.1.1 Object attributes (Instance 0)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	"SiboTech"
	Number of instances	read	UINT16	0002H
	Maximum instance number	read	UINT16	0002H

4.5.1.2 Instance attributes (Instance 1)

Property code	Name	Read/Write	Data type	Description
0	Module type	read	CHAR[20]	Model
	Serial number	read	UINT32	Serial number
	Production date	read	UINT32	Production date
	Hardware major version	read	UINT16	Hardware major version
	Hardware sub-version	read	UINT16	Hardware sub-version
	Firmware major version	read	UINT16	Firmware major version
	Firmware sub-version	read	UINT16	Firmware sub-version

4.5.1.3 Instance attributes (Instance 2)

Property code	Name	Read/Write	Data type	Description
0	Start	read/write	UINT8	0: Offline 1: Online
1	Communication status	read	UINT8	0: Non-data exchange 1: Data exchange
2	Watchdog timeout	read/write	UINT16	0: Off (default); Other value: Timeout (ms).
3	Indicator status	read	UINT8	0: Off; 1: Start bit0: LED1A status;

				bit1: LED1B status; bit2: LED2A status; bit3: LED2B status; bit4~7: Reserve

4.5.2 Diagnostic Object (Code 01H)

4.5.2.1 Object attributes (Instance 0)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	"Diagnostic"
	Number of instances	read	UINT16	0001H
	Maximum instance number	read	UINT16	0001H

4.5.2.2 Instance attributes (Instance 1)

Property code	Name	Read/Write	Data type	Description
0	Diagnostic type	read/write	UINT8	Diagnostic type
	Slot number	read/write	UINT8	Slot number
	Specified parameters	read/write	UINT8	Specified parameters
	Diagnostic data	read/write	UINT8 Queue	Diagnostic data

4.5.3 Network Object (Code 02H)

4.5.3.1 Object attributes (Instance 0)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	"Network"
	Number of instances	read	UINT16	0001H
	Maximum instance number	read	UINT16	0001H

4.5.3.2 Instance attributes (Instance 1)

Property code	Name	Read/Write	Data type	Description
0	Network type	read	CHAR[20]	"PROFIBUS DPV1"
	Data format	read	UINT8	Network data format 1: High-byte priority
	Acyclic data services	read	UINT8	1: Support acyclic data services
	The length of writing process data	read	UINT16	byte
	The length of	read	UINT16	byte

	reading process data			

4.5.4 Network Configuration Object (Code 03H)

4.5.4.1 Object attributes (Instance 0)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	"Network Config"
	Number of instances	read	UINT16	Depend on the network
	Maximum instance number	read	UINT16	Depend on the network

4.5.4.2 Instance attributes (Instance 1)

Code	Name	Read/Write	Data type	Description
0	Parameter name	read	CHAR[20]	Parameter name, such as node address, IP configuration , etc.
	Data type	read	UINT8	Data type
	Number of elements	read	UINT8	Number of elements specifying the data type

	Access	read	UINT8	bit0: read access; bit1: write access; bit2: sharing access.
1	value	Decided by access	Decided by data type	

4.6 Host Application Objects

4.6.1 Configuration Object (Code F0H)

4.6.1.1 Object attributes (Instance 0)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	"Application Config"
	Number of instances	read	UINT16	0003H
	Maximum instance number	read	UINT16	0003H

4.6.1.2 Application data object attributes (Instance 1)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	"Application Data"
	Number of	read	UINT16	Depend on the

	instances			application
	Maximum instance number	read	UINT16	Depend on the application

4.6.1.3 Instance attributes (Instance 2)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	“Config Data”
1	Configuration data	write	UINT8 queue	

4.6.1.4 Instance attributes (Instance 3)

Property code	Name	Read/Write	Data type	Value
0	Name	read	CHAR[20]	“Parameter Data”
1	Configuration data	write	UINT8 queue	

4.6.2 Application Data Object (Code F1H)

Application data object is acyclic data, which is defined by the user. With PROFIBUS DPV1 as an example for explaining the relationship among acyclic data of fieldbus, instances and attributes:

Application data objects——PROFIBUS DPV1

Instance number——Slot number

Attribute number——Index number

The range of the instance number: 0 ~ 254;

The range of the attribute number: 0 ~ 254.

5. Connection Messages

5.1 Message Format

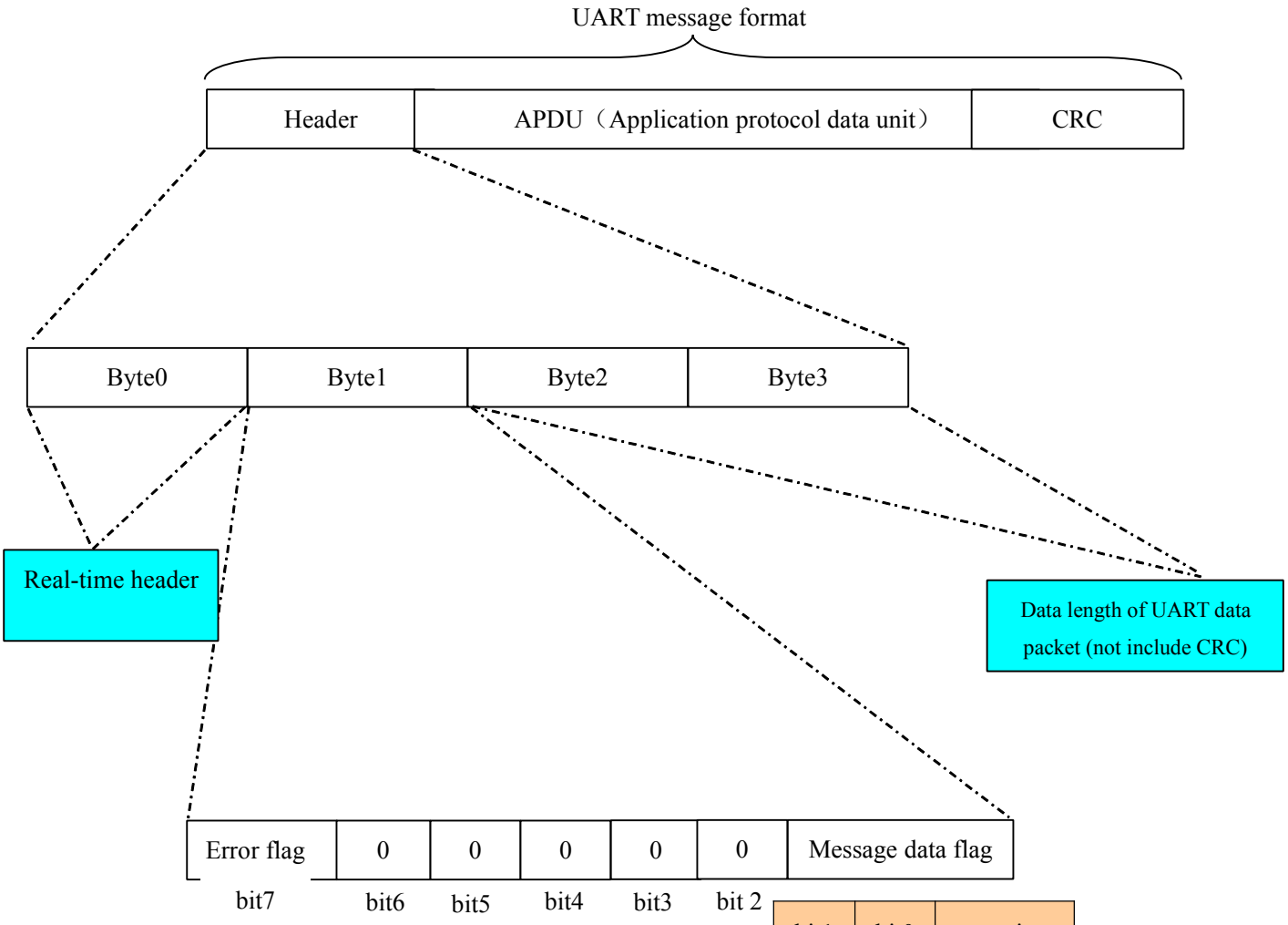
0	1	2 3	4	5	6 7
8 9	10……n	n+1 n+2			
Real-time header	Note 1	Data length	01: read 02: write	Object code	Instance code
Attribute code	Data	CRC check			

Note 1: Byte 1, 00 stands for I/O data; 01 stands for CMD, the message is a command; 02 stands for DATA, the message is a data;

Note 2: The number of 0 to n stands for the 0 byte to the nth byte;

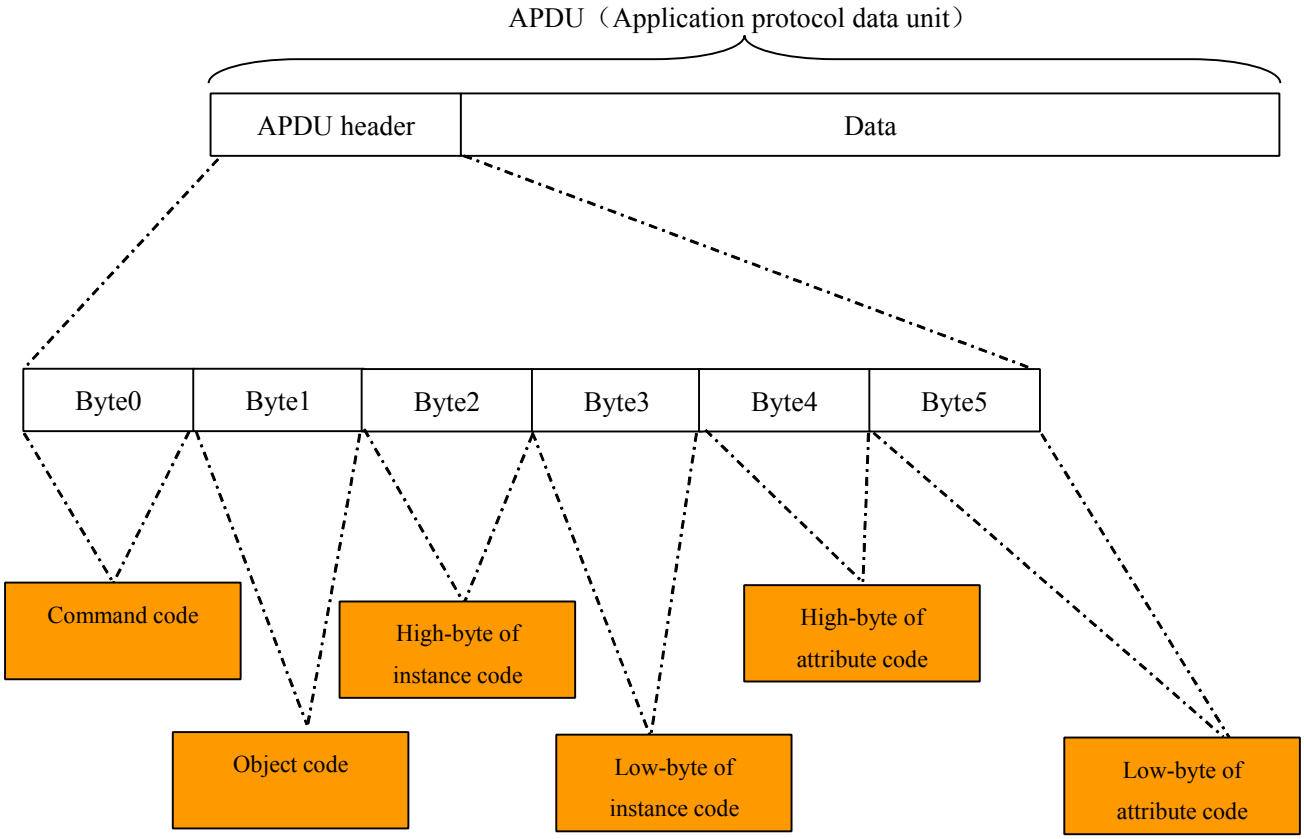
Note 3: Real-time header will automatically add one.

5.2 Message Details



bit7=0，Indicates no error
bit7=1，Indicates error

bit1	bit0	meaning
0	0	IO data
0	1	Command request
1	0	Command response
1	1	Reserve



Note: If the message is I/O data, the message remove the APDU header and the other is the same.

5.3 Example of the Actual Message

The following message applies the basic object in 3.5.1 and instance properties in 3.5.1.2:

- User board send the message to CPD-511:

00 01 00 06 01 00

00 01 00 00 Two bytes CRC

Real-time header Stands for CMD Length "01" stands for reading Object code

Instance code Attribute code CRC

- CPD-511 respond the message to User board:

00 02 00 0X 01 01 00 01

00 00 XX Two bytes CRC

Real-time header Stands for DATA Length "01" stands for reading Object code Instance code

Attribute code Note 1 CRC check

Note 1: XX include all the value in the attribute code 0: the module type, serial number, production date and the hardware firmware version.

5.4 Connecting CPD-511 to PLC

- Operating mode 3 and mode 4（UART, 9600 bps /115200bps）

In operating mode 3 and mode 4, user board should send the following Initialization messages from the serial port to the CPD-511 after hardware connection (including PROFIBUS DP side and serial side):

1、 Write the address

2、 Send Online command

3、 Send IO cycle data (To be sent cyclically until a response is received)

4、 Send confirmation acknowledgment message

Among them, The user board needs loop transmit the packets until the PLC reply a configuration packets to

CPD - 511 (in the case of connected PLC) , the user board then sends the configuration acknowledgment message can be connected to the PLC . (Otherwise not connecting to the PLC, are described in detail in Section 5.5).

- Operating mode 2 (UART, auto-baud rate)

In operating mode 2 (auto-baud rate mode), first user board loop send 0x55 (it is recommended to send the time interval between two 0x55 for 10ms) , until the CPD - 511 Reply 0xAA , or two 0x55 continuous transmission thus received 0xAA .CPD-511 has automatically detected the baud rate when receiving 0xAA. The serial port then sends the following initialization message (the same with operating mode 3 and mode 4).

1、 Write address

2、 Send Online command

3、 Send IO cycle data (To be sent cyclically until a response is received)

4、 Send confirmation acknowledgment message

- Operating mode 0 (Use configuration in non-volatile memory to start)

User board only sends two following initialization message in operating mode 0:

1、 Send IO cycle data (To be sent cyclically until a response is received)

2、 Send confirmation acknowledgment message

5.5 The Details of Connection Message

0、 User board continuously send twice 0x55 (need send in auto-baud rate mode)

（User board->CPD-511）Request: 0x55

（CPD-511 -> User board）Response: 0xAA　　//LED1B keep red, offline

1、 Write the slave address

（User board->CPD-511）Request: 01 01 00 07 02 03 00 01 00 01 07 70 66（Slave address is 7,the last two bytes are CRC bytes）

（CPD-511 -> User board）Response: 01 02 00 06 02 03 00 01 00 01 8B 40

2、 Online command

（User board ->CPD-511）Request: 02 01 00 07 02 00 00 02 00 00 01 FE C7

(CPD-511 -> User board）Response: 02 02 00 06 02 00 00 02 00 00 0E 8F // LED1B off，LED1A blinking，no connection

3、 IO cycle data

（User board ->CPD-511）Request: 03 00 00 05 01 02 03 04 05 10 EE / / This message needs cyclic sending until it receives the response with the configuration data. It's divided into two cases from this down Article packet,

Case 1: CPD-511 is connected to the PLC (correct)

(CPD-511 -> User board） Response: 03 01 00 08 02 F0 00 02 00 01 57 67 25 36 //Connect to the PROFIBUS DP master（PLC）, This is the response with the configuration data , wherein the "57 67 " is the configuration data, the configuration data is determined by the contents of the configuration of the actual PLC .

Case 2: CPD - 511 is not connected to the PLC (error)

(CPD-511 -> User board) Response: 03 00 00 00 00 60 / / The length of IO data is 0 bytes , the user can do judgment , the program runs here , if you receive messages of 2, 3, 4 bytes are all with 0 , which means that the DP is not connected or connection error . If the user continues to send IO packets will continue to receive the packet. If you do not connect PLC, the initialization can not continue to this step.

4、 It is assumed that the current is case 1 , next the serial will send configuration acknowledgment packet (configure the acknowledgment packet from the PLC IO cycle data in response to the " 01 " (CMD) of the second byte changed to " 02 " (DATA) ,and configuration data , for example " 57 67 " is deleted , it becomes the following packets)

（User board ->CPD-511）Request: 03 02 00 06 02 F0 00 02 00 01 DE 9F

(CPD-511 -> User board）Response: 03 00 00 00 00 60 // LED1A Keeping light shows IO data cycle with PLC has been established.

5、 IO Cycle Data

（User board ->CPD-511）Request: 03 00 00 05 01 02 03 04 05 10 EE / / Loop to send this packets, and"01 02 03 04 05" may be replaced with the actual data, the length of data can not exceed the number of input and output bytes which the PLC configuration, the data content can fill where user board want to pass to the actual data of the

PLC.

(CPD-511 -> User board）Response: 03 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 6D ED // LED1A Keeping light shows in the state of data exchange, the length of bytes is decided by the number of input and output bytes of the PLC configuration . This packet data is the PLC passed to the user board .At this point, the user board - CPD - 511 - PLC has entered the state of the data of the IO cycle.

The state of the IO cycle, when the DP is disconnected, the user board will receive the following packets:

(CPD-511 -> User board）Response: 04 00 00 00 01 14 // The length of IO packets is 0

If then the DP plug is connected, the user receives the following packet:

(CPD-511 -> User board）Response: 04 01 00 08 02 F0 00 02 00 01 57 67 D0 FD // This configuration packets is PLC sent by CPD – 511

Note:

It will receive in the case of the user board continuously sends IO packet.

CPD - 511 is a passive response, will not take the initiative to send packets.

Now should jump to the Article 4 of the packets of Section 5.5, the user board re-establish the IO cycle by sending packet 4 and packet 5.

6 GSD File

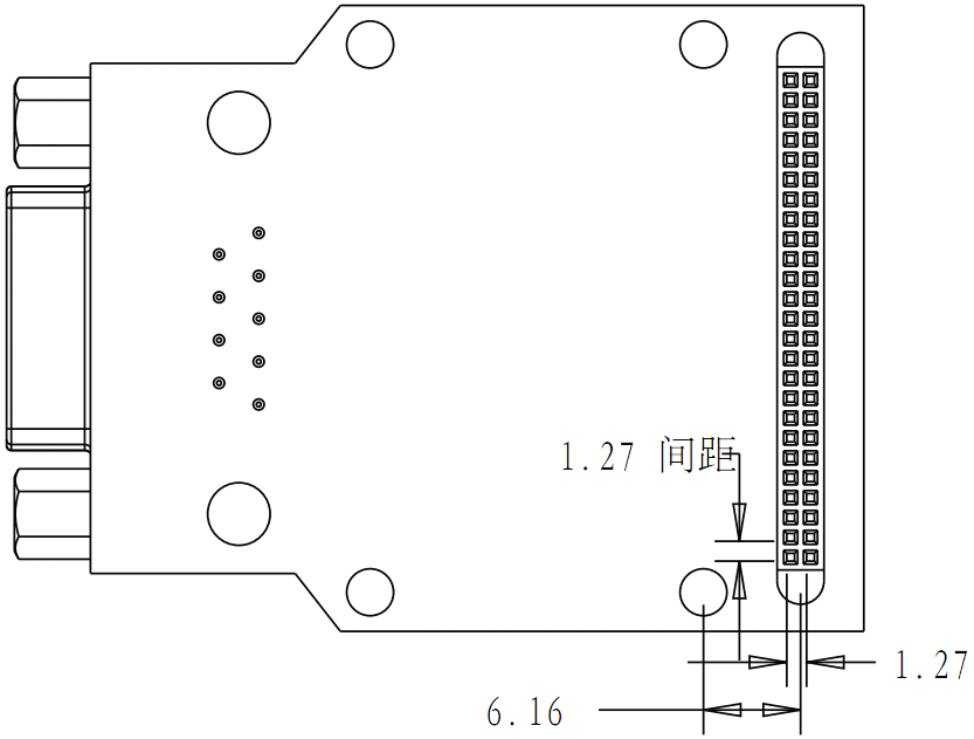
This product provides GSD file for user to use.

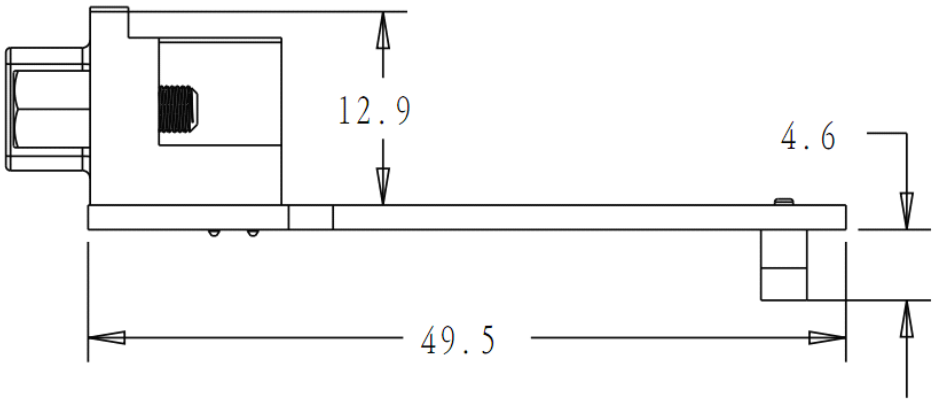
GSD file can be modified through using dedicated software or using notepad to open and edit.

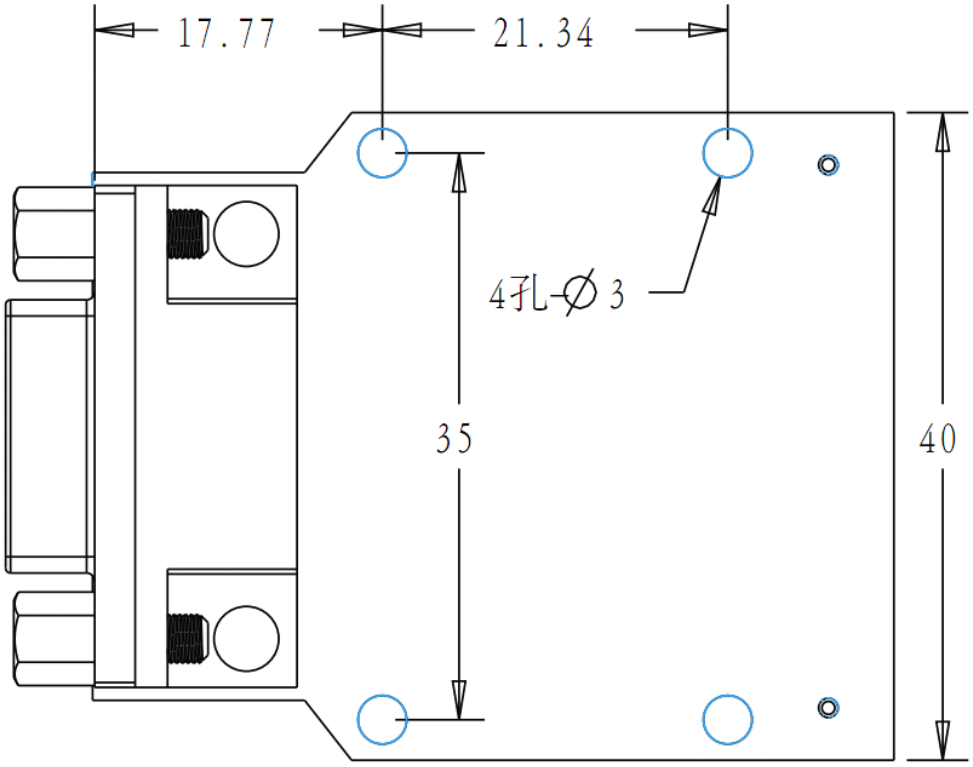
7 Installation

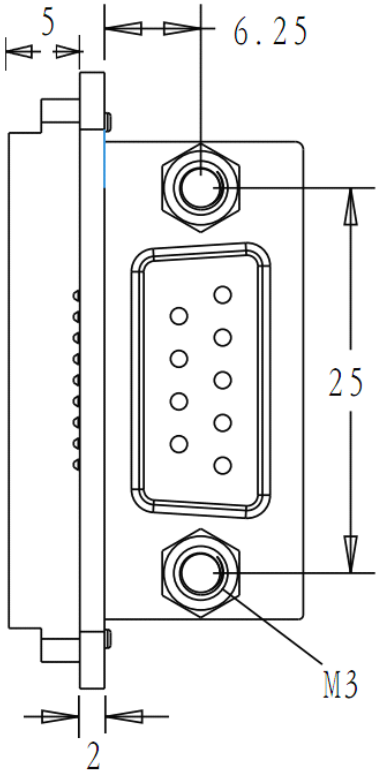
Mechanical size: 49mm（L）*40mm（W）*20mm（H）

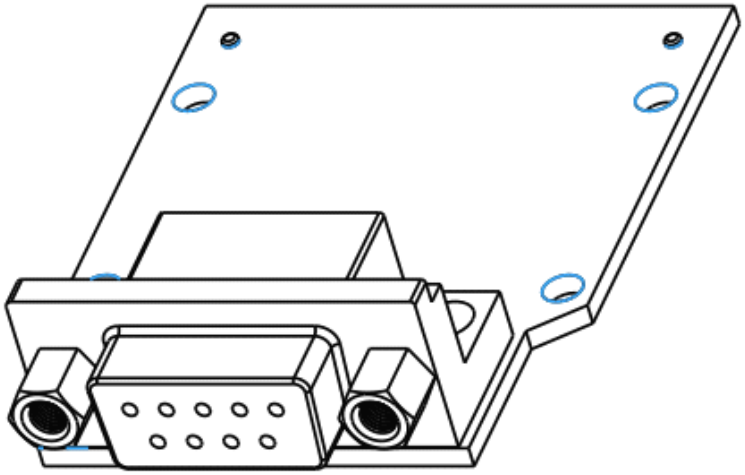
Installation: Use four positioning studs installation.


1.27 间距
1.27
6.16


12.9
4.6
49.5


17.77
21.34
4孔-∅3
35
40


5
6.25
25
M3
2



比例2.000