



DS4-PID Smart Gas Sensor Module

Modbus-RTU Protocol

UART Interface

This document provides a detailed introduction to the Modbus-RTU Communication Protocol of the DS4-PID Smart Gas Sensor Module, along with specific procedures that users need when using the product.

1. Product Overview and Configuration Parameters

1.1 Product Overview

The product communicates using the UART interface. Please refer to section [1.2 UART Configuration Parameters] for specific configurations.

1.2 UART Configuration Parameters

Baud Rate	Data Bits	Start Bit	Stop Bit	Parity Bit	Read/Write Data Interval
9600	8 Bits	1 Bit	1 Bit	None	>=1s

1.3 CRC-16/Modbus Calculation Method

Calculation Method: $x16+x15+x2+1$

C Language Calculation Function:

```
uint16_t get_modbus_crc(uint8_t *buffer, uint16_t len)
{
    uint16_t wrcr = 0xFFFF;
    uint8_t temp;
    uint16_t i = 0, j = 0;
    for (i = 0; i < len; i++)
    {
        temp = *buffer & 0X00FF;
        buffer++;
        wrcr ^= temp;
        for (j = 0; j < 8; j++)
        {
            if (wrcr & 0X0001)
            {
                wrcr >>= 1;
                wrcr ^= 0XA001;
            }
            else
            {
                wrcr >>= 1;
            }
        }
    }
    return wrcr;
}
```

1.4 Introduction to Data Types

a. This product uses big-endian mode for data transmission.

For example, the byte order in the transmission: byte[0]=0x12, byte[1]=0x34, byte[2]=0x56, byte[3]=0x78.

32-bit data is: 0x12345678

b. The transmission of float data uses 32-bit single-precision floating-point format. (IEEE754 Single precision 32-bit).

For example, the byte order in the transmission is: byte[0]=0x41, byte[1]=0x28, byte[2]=0x00, byte[3]=0x00.

Then, the represented floating-point number is: 10.5f.

If users are using C language for development, they can refer to the following example:

```
// Union used for conversion
typedef union{
    float f_data;
    uint8_t u8_data[4];
}FLOAT_TYPE_U;

float get_float_data(uint8_t* byte)
{
    FLOAT_TYPE_U float_data = {0};
    // In big-endian mode, use the following statement
    float_data.u8_data[0] = byte [0];
    float_data.u8_data[1] = byte [1];
    float_data.u8_data[2] = byte [2];
    float_data.u8_data[3] = byte [3];
    // In little-endian mode, use the following statement
    float_data.u8_data[0] = byte[3];
    float_data.u8_data[1] = byte[2];
    float_data.u8_data[2] = byte[1];
    float_data.u8_data[3] = byte[0];
    return float_data.f_data;
}
```

2. Gas Concentration Unit Table

Value (Hex)	0x02	0x04	0x08	0x10
Concentration Units	ppm	ppb	%vol.	%LEL

3. Sensor Type Table

Sensor type	HCHO	VOC	CO	Cl ₂	H ₂	H ₂ S	HCl	HCN	HF	NH ₃	NO ₂	O ₂	O ₃	SO ₂
Type value	0X17	0X18	0X19	0X1A	0X1B	0X1C	0X1D	0X1E	0X1F	0X20	0X21	0X22	0X23	0X24
Sensor type	HBr	Br ₂	F ₂	PH ₃	AsH ₃	SiH ₄	GeH ₄	B ₂ H ₆	BF ₃	WF ₆	SiF ₄	XeF ₂	TiF ₄	SMELL
Type value	0X25	0X26	0X27	0X28	0X29	0X2A	0X2B	0X2C	0X2D	0X2E	0X2F	0X30	0X31	0X32
Sensor type	IAQ	AQI	NMHC	SO _x	NO _x	NO	C ₄ H ₈	C ₃ H ₈ O ₂	CH ₄ S	C ₈ H ₈	C ₄ H ₁₀	C ₂ H ₆	C ₆ H ₁₄	C ₂ H ₄ O
Type value	0X33	0X34	0X35	0X36	0X37	0X38	0X39	0X3A	0X3B	0X3C	0X3D	0X3E	0X3F	0X40
Sensor type	C ₃ H ₉ N	C ₃ H ₇ N	C ₂ H ₆ O	CS ₂	C ₂ H ₆ S	C ₂ H ₆ S ₂	C ₂ H ₄	CH ₃ OH	C ₆ H ₆	C ₈ H ₁₀	C ₇ H ₈	CH ₃ COOH	ClO ₂	
Type value	0X41	0X42	0X43	0X44	0X45	0X46	0X47	0X48	0X49	0X4A	0X4B	0X4C	0X4D	

Sensor type	H ₂ O ₂	N ₂ H ₄	C ₂ H ₆ N ₂	C ₂ HCl ₃	CHCl ₃	C ₂ H ₃ Cl ₃	H ₂ Se	LEL	CO ₂	PID_VOCS	C ₂ H ₂
Type value	0X4E	0X4F	0X50	0X51	0X52	0X53	0X54	0X55	0X56	0X57	0X58

4. Modify the Modbus-RTU Address

Suggested modification range: 01 (0x01) <= address <= 127 (0x7F), with the default set to 01 (0x01).

Send :

Instruction Header										addr
0x53	0x45	0x54	0x41	0x44	0x44	0x52	0x45	0x53	0x53	Addr

Receive:

Instruction Header										addr
0x53	0x45	0x54	0x41	0x44	0x44	0x52	0x45	0x53	0x53	Addr

5. Maintaining the Register Table, along with Read/Write Examples

This protocol only utilizes holding registers, with the operation code for reading multiple holding registers being 0x03, for writing a single holding register being 0x06, and for writing multiple holding registers being 0x10.

5.1 Measurement Data Registers

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF000	Byte[0]	Gas Concentration Value (byte[0]<<24) (byte[1]<<16) (byte[2]<<8) byte[3] Then convert according to the 32-bit float data type.	float	R
0xF001	Byte[1]			
0xF002	Byte[2]	Internal Chip Temperature (Celsius) (byte[4]<<24) (byte[5]<<16) (byte[6]<<8) byte[7] Then convert according to the 32-bit float data type.	float	R
0xF003	Byte[3]			
0xF004	Byte[4]	Reserved Fixed 0		R
0xF005	Byte[5]			
0xF006	Byte[6]	Sensor Status 0: Normal Operation 1: Sensor Replacement Suggested 2: Sensor Failure	uint16	R/W
	Byte[7]			

Example of reading:

Send: 0x01 0x03 0xF0 0x00 0x00 0x07 0x37 0x08

Receive: 0x01 0x03 0x0E 0x00 0x00 0x00 0x00 0x41 0x86 0x8E 0xD9 0x00 0x00 0x00 0x00 0x37 0x43

Send explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0xF0 0x00-----Starting register address
 0x00 0x07-----Read register length - 7
 0x37 0x08-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0x0E-----Read bytes length - 14
 0x00 0x00 0x00 0x00-----Concentration value - 0x00000000 - 0.0f (IEEE754 single precision 32-bit)
 0x41 0x86 0x8E 0xD9-----Temperature value - 0x41868ED9 - 16.8197 (IEEE754 single precision 32-bit)
 0x00 0x00 0x00 0x00-----Reserved - 0x00000000
 0x00 0x00-----Sensor status - Normal operation
 0x37 0x43-----Modbus-RTU CRC

5.2 Sensor Parameter Register

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF100	Byte[0]	Sensor Type, please refer to [3. Sensor Type Table]	float	R
	Byte[1]			
0xF101	Byte[2]	Concentration Unit, please refer to [2.Gas Concentration Unit Table]		
	Byte[3]			
0xF102	Byte[4]	Sensor Range (byte[0]<<24) (byte[1]<<16) (byte[2]<<8) byte[3] Then convert according to the 32-bit float data type.	float	R
	Byte[5]			
0xF103	Byte[6]			
	Byte[7]			

Example of reading:

Send: 0x01 0x03 0xF1 0x00 0x00 0x04 0x76 0xF5

Receive: 0x01 0x03 0x08 0x00 0x57 0x00 0x02 0x42 0xC8 0x00 0x00 0x5F 0x54

Send explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0xF1 0x00-----Starting register address
 0x00 0x04-----Read register length
 0x76 0xF5-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0x08-----Read bytes length - 8
 0x00 0x57-----Sensor type, please refer to [3. Sensor Type Table] - PID_VOCS
 0x00 0x02-----Concentration unit, please refer to [2.Gas Concentration Unit Table] - ppm
 0x42 0xC8 0x00 0x00-----Sensor range - 0x42C80000 - 100.0f
 0x5F 0x54-----Modbus-RTU CRC

5.3 Device Serial Number Register

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF200	Byte[0]	Device Serial Number Example: "1234567890123456"	Ascii Code	R
	Byte[1]			
0xF201	Byte[2]			
	Byte[3]			
0xF202	Byte[4]			
	Byte[5]			
0xF203	Byte[6]			
	Byte[7]			
0xF204	Byte[8]			
	Byte[9]			
0xF205	Byte[10]			
	Byte[11]			
0xF206	Byte[12]			
	Byte[13]			
0xF207	Byte[14]			
	Byte[15]			

Example of reading:

Send: 0x01 0x03 0xF2 0x00 0x00 0x08 0x76 0xB4

Receive: 0x01 0x03 0x10 0x31 0x32 0x33 0x34 0x35 0x36 0x37 0x38 0x39 0x30 0x31 0x32 0x33 0x34 0x35 0x36 0xC7 0xFB

Send explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0xF2 0x00-----Starting register address
 0x00 0x08-----Read register length
 0x76 0xB4-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0x10-----Read bytes length - 16
 0x31 0x32 0x33 0x34-----“1234”
 0x35 0x36 0x37 0x38-----“5678”
 0x39 0x30 0x31 0x32-----“9012”
 0x33 0x34 0x35 0x36-----“3456”
 0xC7 0xFB-----Modbus-RTU CRC

5.4 Sensor Serial Number Register

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF208	Byte[0]	Sensor Serial Number Example: “abcdefghijklmnop”	Ascii Code	R
	Byte[1]			
0xF209	Byte[2]			
	Byte[3]			
0xF20A	Byte[4]			
	Byte[5]			
0xF20B	Byte[6]			
	Byte[7]			
0xF20C	Byte[8]			
	Byte[9]			
0xF20D	Byte[10]			
	Byte[11]			
0xF20E	Byte[12]			
	Byte[13]			
0xF20F	Byte[14]			
	Byte[15]			

Example of reading:

Send: 0x01 0x03 0xF2 0x08 0x00 0x08 0xF7 0x76

Receive: 0x01 0x03 0x10 0x61 0x62 0x63 0x64 0x65 0x66 0x67 0x68 0x69 0x6A 0x6B 0x6C 0x6D 0x6E 0x6F 0x70 0xD7 0xDF

Send explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0xF2 0x08-----Starting register address
 0x00 0x08-----Read register length
 0xF7 0x76-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0x10-----Read bytes length - 16
 0x61 0x62 0x63 0x64-----“abcd”
 0x65 0x66 0x67 0x68-----“efgh”
 0x69 0x6A 0x6B 0x6C-----“ijkl”
 0x6D 0x6E 0x6F 0x70-----“mnop”
 0xD7 0xDF-----Modbus-RTU CRC

5.5 Software Version Number Register

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF210	Byte[0]	Software Version Example: “DS4PID20231012_V0002”	Ascii Code	R
	Byte[1]			
0xF211	Byte[2]			
	Byte[3]			
0xF212	Byte[4]			
	Byte[5]			
0xF213	Byte[6]			
	Byte[7]			
0xF214	Byte[8]			
	Byte[9]			
0xF215	Byte[10]			
	Byte[11]			
0xF216	Byte[12]			
	Byte[13]			
0xF217	Byte[14]			
	Byte[15]			
0xF218	Byte[16]			
	Byte[17]			
0xF219	Byte[18]			
	Byte[19]			

Example of reading:

Send: 0x01 0x03 0xF2 0x10 0x00 0x0A 0xF6 0xB0

Receive: 0x01 0x03 0x14 0x44 0x53 0x34 0x50 0x49 0x44 0x32 0x30 0x32 0x33 0x31 0x30 0x31 0x32 0x5F 0x56 0x30 0x30 0x30 0x32 0x39 0x46

Send explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0xF2 0x10-----Starting register address
 0x00 0x0A-----Read register length
 0xF6 0xB0-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0x10-----Read bytes length - 20
 0x44 0x53 0x34 0x50-----“DS4P”
 0x49 0x44 0x32 0x30-----“ID20”
 0x32 0x33 0x31 0x30-----“2310”
 0x31 0x32 0x5F 0x56-----“12_V”
 0x30 0x30 0x30 0x32-----“0002”
 0x39 0x46-----Modbus-RTU CRC

5.6 Manufacturer Identification Code Register

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF21A	Byte[0]	Manufacturer Identification Code	User Define	R/W
	Byte[1]			
0xF21B	Byte[2]			
	Byte[3]			
0xF21C	Byte[4]			
	Byte[5]			
0xF21D	Byte[6]			
	Byte[7]			
0xF21E	Byte[8]			
	Byte[9]			
0xF21F	Byte[10]			
	Byte[11]			
0xF220	Byte[12]			
	Byte[13]			
0xF221	Byte[14]			
	Byte[15]			

Example of writing:

Send: 0x01 0x10 0xF2 0x1A 0x00 0x08 0x10 0x45 0x43 0x2D 0x53 0x45 0x4E 0x53 0x45 0x2D 0x41 0x51 0x53 0x2D 0x44 0x45 0x56 0xEC 0xDC

Receive: 0x01 0x10 0xF2 0x1A 0x00 0x08 0xD2 0xB0

Send explanation:

0x01-----Modbus address
 0x10-----Writing to multiple holding registers
 0xF2 0x1A-----Starting register address
 0x00 0x08-----Number of registers to write - 8
 0x10-----Number of bytes to write - 16
 0x45 0x43 0x2D 0x53-----Write data
 0x45 0x4E 0x53 0x45-----Write data
 0x2D 0x41 0x51 0x53-----Write data
 0x2D 0x44 0x45 0x56-----Write data
 0xEC 0xDC-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x10-----Writing to multiple holding registers
 0xF2 0x1A-----Starting register address
 0x00 0x08-----Number of registers to write - 8
 0xD2 0xB0-----Modbus-RTU CRC

Example of reading:

Send: 0x01 0x03 0xF2 0x1A 0x00 0x08 0x57 0x73

Receive: 0x01 0x03 0x10 0x45 0x43 0x2D 0x53 0x45 0x4E 0x53 0x45 0x2D 0x41 0x51 0x53 0x2D 0x44 0x45 0x56 0x8C 0x6D

Send explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0xF2 0x1A-----Starting register address
 0x00 0x08-----Read register length - 8
 0x57 0x73-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x03-----Read hold register
 0x10-----Read bytes length - 16
 0x45 0x43 0x2D 0x53-----Read data
 0x45 0x4E 0x53 0x45-----Read data
 0x2D 0x41 0x51 0x53-----Read data
 0x2D 0x44 0x45 0x56-----Read data
 0x8C 0x6D-----Modbus-RTU CRC

5.7 Calibration Register

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF300	Byte[0]	User-calibrated concentration value, the value must be greater than or equal to 0 (≥ 0). It is recommended to use a two-point calibration method: 1. Introduce pure air first, write 0 to this register. 2. Introduce the target gas, write the concentration of the target point to this register.	float	W
	Byte[1]			
0xF301	Byte[2]			
	Byte[3]			
0xF302	Byte[4]	Restore Factory Calibration	uint16_t	W
	Byte[5]	1: Restore Factory Calibration Other values: Invalid		

Zero point calibration example:

Send: 0x01 0x10 0xF3 0x00 0x00 0x02 0x04 0x00 0x00 0x00 0x00 0xE3 0x5B

Receive: 0x01 0x10 0xF3 0x00 0x00 0x02 0x72 0x8C

Send explanation:

0x01-----Modbus address

0x10-----Writing to multiple holding registers

0xF3 0x00-----Starting register address

0x00 0x02-----Number of registers to write - 2

0x04-----Number of bytes to write - 4

0x00 0x00 0x00 0x00-----Concentration value written - 0x00000000 - 0.0f (IEEE754 single precision 32-bit)

0xE3 0x5B-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address

0x10-----Writing to multiple holding registers

0xF3 0x00-----Starting register address

0x00 0x02-----Number of registers to write - 2

0x72 0x8C-----Modbus-RTU CRC

Target concentration calibration example:

Send: 0x01 0x10 0xF3 0x00 0x00 0x02 0x04 0x41 0x20 0x00 0x00 0xF6 0xAD

Receive: 0x01 0x10 0xF3 0x00 0x00 0x02 0x72 0x8C

Send explanation:

0x01-----Modbus address

0x10-----Writing to multiple holding registers

0xF3 0x00-----Starting register address

0x00 0x02-----Number of registers to write - 2

0x04-----Number of bytes to write - 4

0x41 0x20 0x00 0x00-----Concentration value written - 0x41200000 - 10.0f (IEEE754 single precision 32-bit)

0xE3 0x5B-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x10-----Writing to multiple holding registers
 0xF3 0x00-----Starting register address
 0x00 0x02-----Number of registers to write - 2
 0x72 0x8C-----Modbus-RTU CRC

Restore factory calibration example:

Send: 0x01 0x10 0xF3 0x02 0x00 0x01 0x02 0x00 0x01 0xA5 0x7D

Receive: 0x01 0x10 0xF3 0x02 0x00 0x01 0x93 0x4D

Send explanation:

0x01-----Modbus address
 0x10-----Writing to multiple holding registers
 0xF3 0x02-----Starting register address
 0x00 0x01-----Number of registers to write - 1
 0x02-----Number of bytes to write - 2
 0x00 0x01-----Restore factory calibration
 0x93 0x4D-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
 0x10-----Writing to multiple holding registers
 0xF3 0x02-----Starting register address
 0x00 0x01-----Number of registers to write - 1
 0x93 0x4D-----Modbus-RTU CRC

5.8 No Inhibition Data Register

Can be selectively provided to users who need it.

REG (Hex)	Data	Description	Data Type (Big-Endian)	R/W
0xF000	Byte[0]	Gas Concentration Value (byte[0]<<24) (byte[1]<<16) (byte[2]<<8) byte[3] Then convert according to the 32-bit float data type.	float	R
	Byte[1]			
0xF001	Byte[2]			
	Byte[3]			
0xF002	Byte[4]	Internal Chip Temperature (Celsius) (byte[4]<<24) (byte[5]<<16) (byte[6]<<8) byte[7] Then convert according to the 32-bit float data type.	float	R
	Byte[5]			
0xF003	Byte[6]			
	Byte[7]			
0xF004	Byte[8]	Reserved Fixed 0		R
	Byte[9]			
0xF005	Byte[11]			
	Byte[12]			
0xF006	Byte[13]	Sensor Status 0: Normal Operation 1: Sensor Replacement Suggested 2: Sensor Failure	uint16	R/W
	Byte[14]			

Example of reading:

Send: 0x01 0x03 0xF4 0x00 0x00 0x07 0x36 0x38

Receive: 0x01 0x03 0x0E 0xBF 0xB0 0x5D 0xB0 0x41 0x97 0x56 0x87 0x00 0x00 0x00 0x00 0x00 0x3F 0xB3

Send explanation:

0x01-----Modbus address
0x03-----Read hold register
0xF4 0x00-----Starting register address
0x00 0x07-----Read register length - 7
0x36 0x38-----Modbus-RTU CRC

Receive explanation:

0x01-----Modbus address
0x03-----Read hold register
0x0E-----Read bytes length - 14
0xBF 0xB0 0x5D 0xB0-----Concentration value - 0x00000000 – -1.377859f (IEEE754 single precision 32-bit)
0x41 0x97 0x56 0x87-----Temperature value - 0x41975687 - 18.917 (IEEE754 single precision 32-bit)
0x00 0x00 0x00 0x00-----Reserved - 0x00000000
0x00 0x00-----Sensor status - Working normally
0x3F 0xB3-----Modbus-RTU CRC



**Business Centre
Europe and the Rest of the World**

EC Sense GmbH
Wangener Weg 3
82069 Hohenschäftlarn, Germany
Tel: +49(0)8178-99992-10 Fax: +49(0)8178-9999-211
Email: office@ecsense.com
www.ecsense.com, www.ecnose.de

**Business Centre
Asia**

Ningbo AQSystems Technology Co., Ltd.
F4-17 Building, Zhong Wu Technology Park No.228,
Jin Gu Bei Road, Yinzhou District NingBo,
Zhejiang Province, P.R. China Post Code: 315100
Tel: +86(0)574 88097236, 88096372
Email: info@aqsystems.cn
www.ecsense.cn, www.ecnose.de