# SPECIFICATION

Product Name: NDIR CH4 Sensor

Item No.: JW-05

Version: V0.3 (Preliminary)

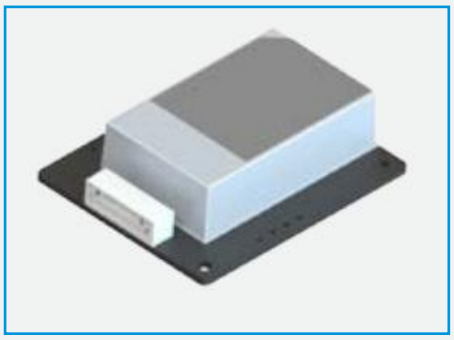
Date: Feb 23rd, 2022

# Revision

| No. | Version | Content | Date |
|---|---|---|---|
| 1 | V0.1 | Preliminary version | 2021.08.26 |
| 2 | V0.2 | Add I2C and PWM communication protocol | 2022.11.26 |
| 3 | V0.3 | ① Gas pressure regulator application added;<br>② External output signal ALARM OC is changed to "LP_EN, low power consumption enable pin: High or floating-work, low-standby";<br>③ The range is changed from 0-50%LEL to 0-100%LEL;<br>④ The accuracy is changed from ±3%LEL to ±5%LEL∈(0%LEL, 50%LEL], ±10%*Reading∈(50%LEL, 100%LEL]<br>⑤ Change the concentration reading from ÷10 to ÷100; | |
| | | | |

# NDIR CH4 Sensor

## JW-05



### Applications

- Home gas alarm
- Portable gas detector
- Natural gas leak detection
- Gas pressure regulator

## Description

JW-05 is a dual-channel methane (CH4) sensor based on non-diffusion infrared technology (NDIR), which can constantly monitor methane gas leakage in the ambient.The sensor has advantages of high accuracy and strong long-term stability, and it's widely used in fields like home gas alarms, portable gas detectors, natural gas leak detection, gas pressure regulator, etc..

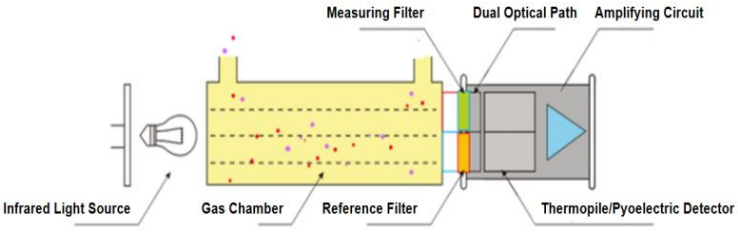
## Features

- NDIR (non-dispersive infrared) technology
- High selectivity to methane, no high concentration poisoning
- Dual-channel detection technology brings less drift and better long-term stability
- High accuracy in full working temperature and measurement range
- Small size, compact structure, easy to install
- 10+ years long lifetime

## Working Principle

JW-05 is mainly composed of one infrared light source, one gas chamber, two optical filters and one detector. The light source radiates infrared light, which passes through the measured gas in the optical path, then the narrow optical filter, finally reaches the infrared detector. To determine the gas concentration by measuring the intensity of the infrared light that enters the infrared detector.

Infrared absorption is a kind of molecular absorption spectrum. CH4 molecules have a specific absorption spectrum in the infrared wavelength region. When the infrared light wavelength matches the CH4 gas absorption spectrum line, infrared energy gets absorbed. The absorption intensity follows the Lambert-Beer's law.



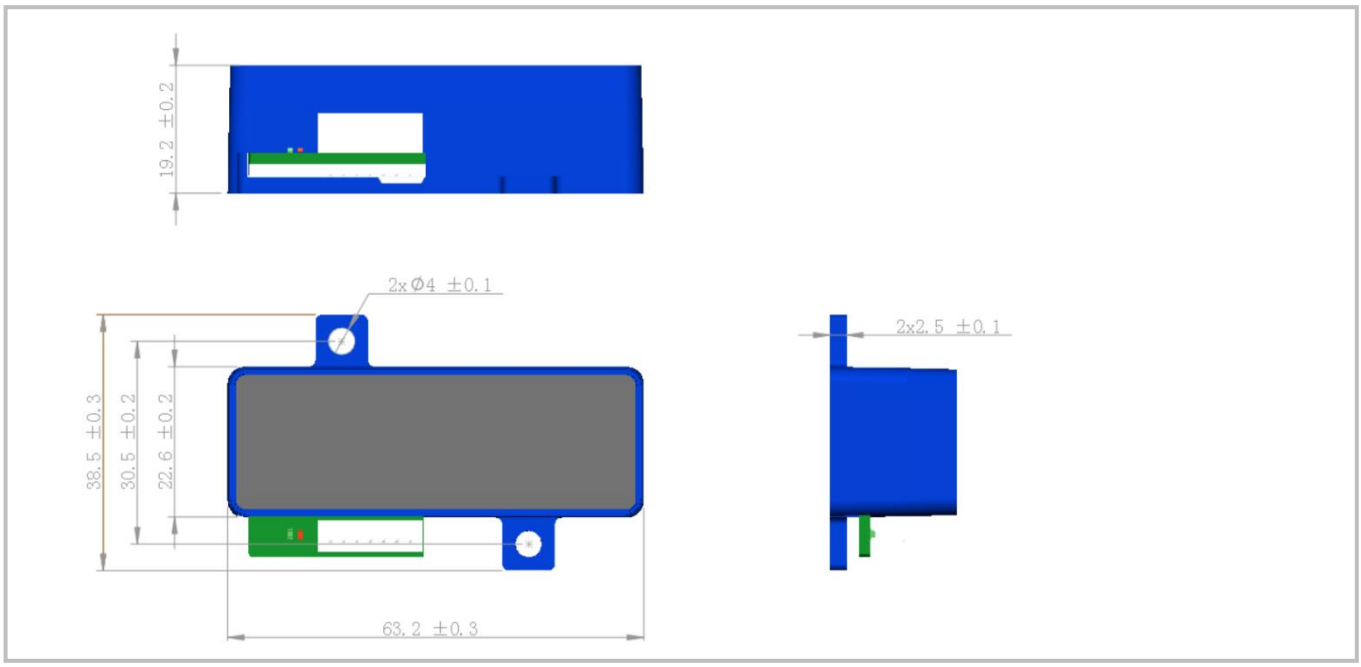
## Specifications

| JW-05 Sensor Specification | |
|---|---|
| Target gas | Methane (CH4) |
| Working principle | NDIR (non-dispersive infrared) technology |
| Measurement range | 0~100%LEL |
| Alarm point | Default 10%LEL. 5%LEL~25%LEL adjustable |
| | |
| Accuracy | ±2%FS (25℃) |
| Repeatability | ≤±1%FS |
| Response time | T90<30s (diffusion) |
| Data refresh time | ≤1s |
| Preheating time | ≤15s |
| Output | 1. Modbus-RTU<br>2. TTL/OC digital output |
| Protection grade | IP54 |
| Working pressure | 80kPa~120kPa① |
| Working condition | -20 ~ 60℃, 0~95%RH (non-condensing) |
| Storage condition | -40 ~ 85℃, 0~95%RH (non-condensing) |
| Power supply | 5V ~ 9V DC, ripple wave<100mV |
| Average working current | ≤130mA |
| Peak current | ≤260mA |
| Lifetime | 10+ years |
| Dimension | L63.2*W38.5*H19.2 mm |

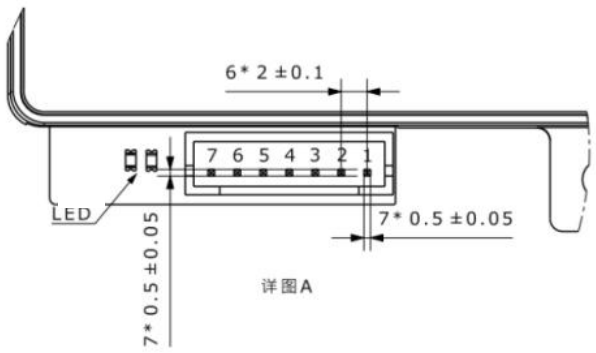
Remarks: ①The response time and detection accuracy test condition are based on atmospheric pressure 101kPa.

## Dimensions and Connector

### 1. Dimensions (Unit: mm)




### 2. Pin Definition




| No. | Pin | Description |
|---|---|---|
| 1 | Power | Power input (+5V) |
| 2 | GND | Power input (ground terminal) |
| 3 | COM | Modbus-RTU |
| 4 | OUT1 | OC digital output |
| 5 | OUT2 | Equipment failure (active low) |
| 6 | OUT3 | Gas leakage alarming (active low) |
| 7 | OUT4 | Reserve |

### 3. Connector Type

| Item | Part Number | Pitch | Brand |
|---|---|---|---|
| Connector | A2001WV-7P | 2 mm | CJT |
| Matching connector | A2001H-7P, A2001-T | 2 mm | CJT |

## 4. LED Signal Status Description

There is one red and one green LED indicator on the external output interface of the sensor.

The status description is as following table:

| LED Green | LED Red | Equipment Condition |
|---|---|---|
| OFF | OFF | No power |
| FLASHING | OFF | Self-checking |
| ON | OFF | Work normally |
| OFF | ON | System error |

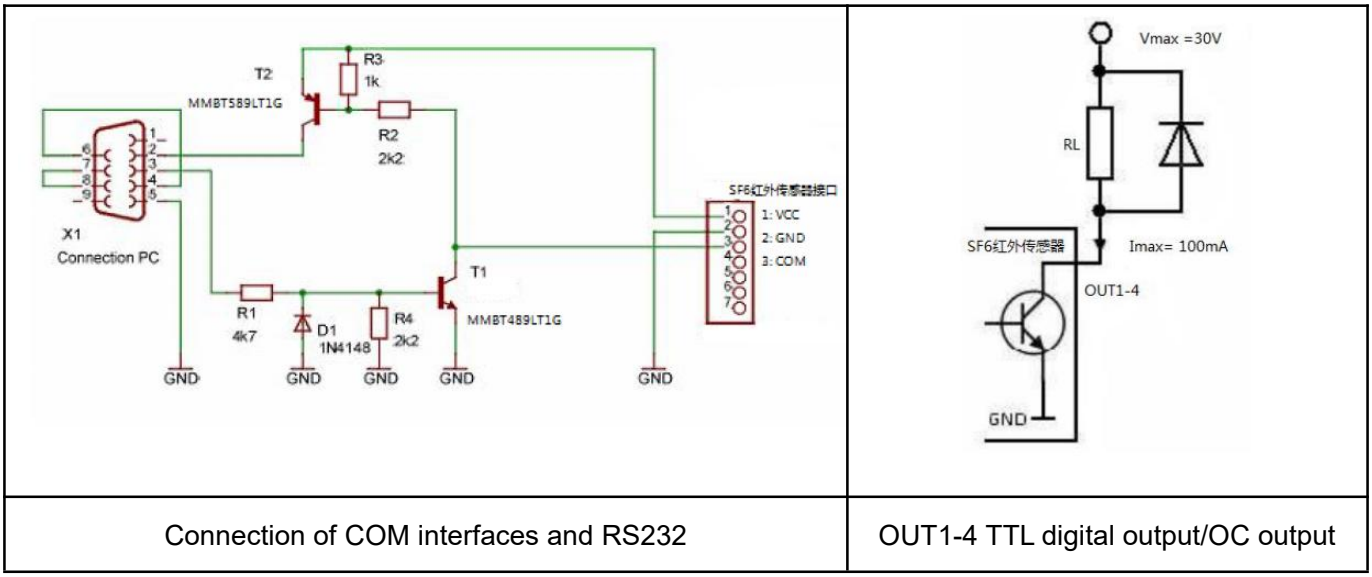
## 5. Hardware Connection Mode

### 5.1 Connection of COM interfaces and RS232

COM and RS232 communication conversion reference circuit, which can also directly work with MCU in single-line communication mode powered by the same voltage.

### 5.2 OUT1-4 TTL digital output/OC output

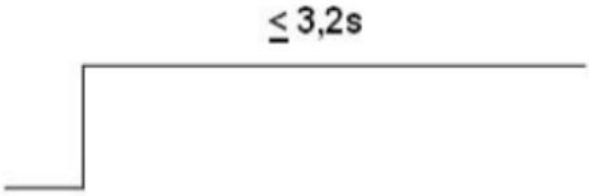
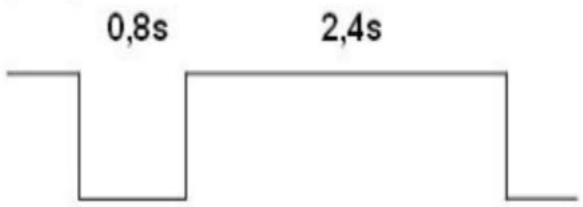
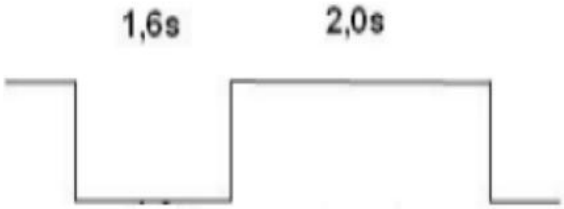
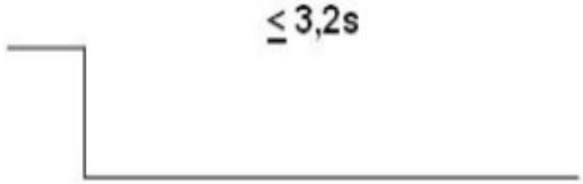
SF6 infrared sensor external circuit reference circuit.




| Connection of COM interfaces and RS232 | OUT1-4 TTL digital output/OC output |
|---|---|

Note: Interfaces OUT 1-4 are open interfaces, and the maximum grounding load is 100mA. The voltage is not allowed to be higher than 30V. In the case of connecting an inductive load, it needs to connect into the freewheeling diode.

## 6. OUT1 TTL Digital Output

| Digital Output Condition | Waveform Sequence Diagram |
|---|---|
| Working normally |  |
| Gas early alarm |  |
| Gas alarm |  |
| System error |  |
| User-defined 1 | |
| User-defined 2 | |

## Precautions

1. During the installation process, please make sure that no mechanical pressure is applied to SF6-015. In addition, it must be installed on a flat surface to prevent twisting or tilting.
2. It is recommended that SF6-015's air inlet is installed perpendicular to the ground or facing downwards. Please don’t install upwards, also prevent dust and other things from blocking the air inlet.
3. In order to maintain good detection accuracy, SF6-015 power supply must be within the required range of technical parameters.
4. Please don’t disassemble SF6-015, or it may cause poor accuracy.
5. It needs to avoid SF6-015 from working in harsh environments containing a lot of dust, and the protection against sandstorms, rain and snow should be provided by the user's equipment.
6. When SF6-015 is installed and used in the system, please make sure the airflow is smooth and there should be no large airflow facing the sensor.

# SF6 Sensor Communication Protocol

## 1. Protocol Overview

This protocol is defined in the communication protocol for obtaining sensor measurement information in the SF6 project. Serial port baud rate: 2400bps, DataBits: 7, StopBits: 1, Parity: Even. In this protocol, the sensor will not actively send data, and will return the corresponding data only after the host sends the command.

## 2. Serial Communication Protocol Format

- The data in this protocol, unless otherwise specified, are all ASCII characters. For example, ":" means hexadecimal [58], "2" means hexadecimal [32]
- [xx] is single-byte data (unsigned, 0-255); (xx) represents double-byte data, with high byte first and low byte after; xx represents data pack containing multiple bytes. Data pack definition refers to the specific protocol description.
- The default Modbus communication address: A0
- Modbus communication frame format

| START | Address | Function code | DATA | CheckSum validation | STOP |
|---|---|---|---|---|---|
| : | A0 | [xx] | xx | [xx] | \r\n |

- Supported function codes: data read code 0x03, data write code 0x06

### 2.1 Read data protocol format

**Send commend**

[START] Start byte (one byte, ":")
[IP] Modbus communication IP address
[CMD] Instruction type, read instruction is 03, write instruction is 06
[RegAddr] Modbus communication register address
[Number] Number of registers
[CheckSum] CheckSum =256－[(IP+CMD+RegAddr+ Number)%256]
[STOP] Instruction frame stop field, content is "Enter", it's "\r\n", corresponding to hexadecimal 0x0D, 0x0A

The sending data format is shown in table 2.1

| START | IP | CMD | RegAddr | Number | CheckSum | STOP |
|---|---|---|---|---|---|---|
| 1 Byte | 2 Bytes | 2 Bytes | 4 Bytes | 4 Bytes | 2 Bytes | 2 Bytes |
| : | 01 | 03 | 0003 | 0001 | B8 | \r\n |

Table 2.1 Reading sensor information sending data format

The above command example is sent in ASCII format: 010300030001B8+"Enter", converted to hexadecimal and sent as: 3A 30 31 30 33 30 30 30 33 30 30 30 31 42 38 0D 0A

**Sensor response:**

[START] Start byte (one byte, ":")
[IP] Modbus communication IP address
[CMD] Instruction type, read instruction is 03, write instruction is 06
[ByteNum] Read bytes number
[Data] Read register result

[CheckSum] CheckSum=256[(IP+CMD+ByteNum+Data)%256]

[STOP] Instruction frame stop field, content is "Enter", it's "\r\n", corresponding to hexadecimal 0x0D, 0x0A

The sending data format is shown in table 2.2

| START | IP | CMD | ByteNum | Data | CheckSum | STOP |
|---|---|---|---|---|---|---|
| 1 Byte | 2 Bytes | 2 Bytes | 2 Bytes | 4 Bytes | 2 Bytes | 2 Bytes |
| : | 01 | 03 | 02 | 0115 | B2 | \r\n |

Table 2.2 Response command structure

The above command example is sent in ASCII format as: 0103020115B2+"Enter", converted to hexadecimal and sent as: 3A 30 31 30 33 30 32 30 31 31 35 42 32 0D 0A

## 2.2 Write data protocol format

**Send command**

[START] Start byte (one byte, ":")

[IP] Modbus communication IP address

[CMD] Instruction type, read instruction is 03, write instruction is 06

[StartAddr] Modbus communication register first address

[Data] The value to be written into the register

[CheckSum] CheckSum =256－[(IP+CMD+RegAddr+ Number)%256]

[STOP] Instruction frame stop field, content is "Enter", it's "\r\n", corresponding to hexadecimal 0x0D, 0x0A

The sending data format is shown in table 2.3

| START | IP | CMD | StartAddr | Data | CheckSum | STOP |
|---|---|---|---|---|---|---|
| 1 Byte | 2 Bytes | 2 Bytes | 4 Bytes | 4 Bytes | 2 Bytes | 2 Bytes |
| : | 01 | 06 | 0047 | 1F1A | 85 | \r\n |

Table 2.3 Sending data format for writing sensor information

The above command example is sent in ASCII format as: 010600471F1A85+"Enter", converted to hexadecimal and sent as: 3A 30 31 30 36 30 30 34 37 31 46 31 41 38 35 0D 0A

**Sensor response**

[START] Start byte (one byte, ":")

[IP] Modbus communication IP address

[CMD] Instruction type, read instruction is 03, write instruction is 06

[StartAddr] Modbus communication register first address

[Data] The value to be written into the register

[CheckSum] CheckSum =256－[(IP+CMD+StartAddr+ Data)%256]

[STOP] Instruction frame stop field, content is "Enter", it's "\r\n", corresponding to hexadecimal 0x0D, 0x0A

The sending data format is shown in table 2.4

| START | IP | CMD | StartAddr | Data | CheckSum | STOP |
|---|---|---|---|---|---|---|
| 1 Byte | 2 Bytes | 2 Bytes | 4 Bytes | 4 Bytes | 2 Bytes | 2 Bytes |
| : | 01 | 06 | 0054 | 05DC | 84 | \r\n |

Table 2.4 Response command structure

The above command example is sent in ASCII format as: 010600005405DC84+"Enter", converted to hexadecimal and sent as: 3A 30 31 30 36 30 30 35 34 30 35 44 38 34 38 0D 0A

## 3. CheckSum

In this project, the CheckSum is the check value of all bytes in the complete data frame except START, STOP and check bytes. For example, read the temperature result, the check formula of CheckSum is CheckSum =256 — [(IP+CMD+StartAddr+ Data)%256]

Send the complete command frame as:

ASCII format: :010300030001B8+"Enter"

Hex format: 3A 30 31 30 33 30 30 30 33 30 30 30 31 42 38 0D 0A

Dec format: 58 48 49 48 51 48 48 48 51 48 48 48 49 66 56 13 10

The data of check calculation is 010300030001, and the corresponding decimal (Dec) format data is 48 49 48 51 48 48 48 51 48 48 48 49

Checksum calculation is CheckSum= 256 – (48+49+48+51+48+48+48+51+48+48+48+49)% 256

= 256 – (584% 256)

= 256 – 72 = 184 = 0xB8

So the ASCII format of checksum result is B8, and the final data frame with START and STOP is: 010300030001B8+"Enter"

## 4. CMD Function Table

| SF6 Sensor Modbus Register Table | | | |
|---|---|---|---|
| **Register address** | **Register name** | **Read/Write** | **Description** |
| 0x00C0 | Modbus_address | Read | Sensor Modbus address |
| 0x0080~0x0083 | DeviceType | Read | Instrument model |
| 0x0084~0x0085 | SoftwareVersion | Read | Software version |
| 0x0086~0x0089 | Internal device number | Read | Instrument serial number (SN) |
| 0x0005 | MOD | Read | Gas concentration setting value |
| 0x000A | Concentration | Read | The gas concentration is stored in numerical form in this register. According to different detection gas, S-Module will use an internal factor when calculating the concentration |
| 0x0003 | T_module (0.1*℃) | Read | Sensor internal temperature, used as a reference value for temperature calibration |
| 0x0045 | Alarm_Level | Read/Write | Provide the main gas alarm value limit, which can be set by the user |
| 0x0044 | Warn_Level | Read/Write | Provides the early alarm limit of the main gas, which can be set by the user |
| 0x0047 | IR_4tagneu | Read/Write | The detection intensity of the stored gas at zero value, the address can be read and written |
| 0x004F | Unit | Read | Value = 0: Unassigned Reserved for special applications<br>Value = 1: ppm x 0.01 or ppb x 10<br>Value = 2: ppm x 0.1 or ppb x 100<br>Value = 3: ppm x 1 or ppb x 1000<br>Value = 4: Vol% x 0.001 or ppm x 10<br>Value = 5: Vol% x 0.01 or ppm x 100<br>Value = 6: Vol% x 0.1 or ppm x 1000<br>Value = 7: %UEG x 0.01 Lower explosion limit<br>Value = 8: %UEG x 0.1 Lower explosion limit |
| 0x0054 | Span | Read/Write | Full scale calibration |

| | | | |
|---|---|---|---|
| 0x0009 | Statusflags | Read | This status signal gives the status information accepted by the module<br>Flag 0 Testflag value "1" represents the test status of the instrument<br>Flag 1 Warmup is set to "1" about 10s after it is turned on<br>Flag 2 Syserr displays "1" when the instrument is in error<br>Flag 3 Alarm The value of the main gas alarm is "1"<br>Flag 4 Warn gas early alarm time value is "1"<br>Flag 5 Startup, the value is "1" when the instrument runs for 90s<br>Flag 6 Korr The value is "1" when the temperature is compensated<br>Flag 7 mw_ok is “1” when calibrating<br>Flags 6 (Korr), 7 (mw_ok) are internal signals, when S-Module is produced<br>set up. Their value is also used for quality inspection, when each S-Module is completed<br>After the temperature compensation and calibration process, their value will be set to "1". |

Table 4.1 Register description

Note:For concentration register (0x000A), if the returned data is FFFF, it means the sensor has just been powered on and is warming up (about 15S); if the returned data is FFFE, It means the sensor is faulty; if the returned data is FFFD, it means the sensor is beyond the range that can be measured.

## 5. Communication Protocol Example:

### 5.1 Set device address (register 0x00C0)

**Send**

| Start | Adr.1 | Write | Start register 0x00C0 | Addr: 160 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 06 | 00 C0 | 00 A0 | 95 |

If the command has been sent correctly, then the same command will return. Register 0x00C0 can check whether the value has been written correctly.

### 5.2 Search device address (register 0x00C0)

**Send**

| Start | Adr.160 | Read | Start register 0x00C0 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | A0 | 03 | 00 C0 | 00 01 | 98 |

**Receive**

| Start | Adr.160 | Read | No. of bytes | Addr: 0xA0 | CheckSum |
|---|---|---|---|---|---|
| : | A0 | 03 | 02 | 00 A0 | F9 |

Device address = 0x00*256 + 0xA0 = 0xA0, the example here reads the sensor's IP as 0xA0.

### 5.3 Search device type (register 0x0080-0x0083)

**Send**

| Start | Adr.1 | Read | Start register 0x0080 | Register no.: 4 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 80 | 00 04 | B0 |

**Receive**

| Start | Adr.1 | Read | No. Of bytes | S | M | - | S | F | 6 | - | 1 | CheckSum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| : | 01 | 03 | 08 | 53 | 4D | 2D | 53 | 46 | 36 | 2D | 31 | 36 |

The above example reads the device result as SM-SF6-1.

### 5.4 Search SN number (register 0x0086-0x0089)

**Send**

| Start | Adr.1 | Read | Start register 0x0086 | Register no.: 4 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 86 | 00 04 | AA |

**Receive**

| Start | Adr.1 | Read | No. Of bytes | SN0 | SN1 | SN2 | SN3 | SN4 | SN5 | SN6 | SN7 | CheckSum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| : | 01 | 03 | 08 | CS | 03 | A1 | 21 | 11 | 23 | 00 | 99 | 3A |

SN0~SN7 are converted to SN number CS03A12111230099

### 5.5 Search status flag (register 0x0009)

**Send**

| Start | Adr.1 | Read | Start register 0x0009 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 09 | 00 01 | B2 |

**Receive**

| Start | Adr.1 | Read | No. of bytes | Data: 192 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 00 C0 | A7 |

Status flag = 0x00 * 256 + 0xC0 = 0xC0, specific status information search table 4.1 Register table description

### 5.6 Search software version number (register 0x0084-0x0085)

**Send**

| Start | Adr.1 | Read | Start register 0x0084 | Register no.: 2 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 84 | 00 02 | AE |

**Receive**

| Start | Adr.1 | Read | No. of bytes | V | 1 | 0 | 0 | CheckSum |
|---|---|---|---|---|---|---|---|---|
| : | 01 | 03 | 04 | 56 | 31 | 30 | 30 | 43 |

The software version number read in the above example is V1.0.0

### 5.7 MODE (register 0x0005)

**Send**

| Start | Adr.1 | Read | Start register 0x0005 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 05 | 00 01 | B6 |

**Receive**

| Start | Adr.1 | Read | No. of bytes | Data: 1800 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 07 08 | 0B |

MODE = 0x07 * 256 + 0x08 = 1800, the example here reads the sensor range as 1800ppm.

### 5.8 Set pre-alarm concentration (register 0x0044)

**Send**

| Start | Adr.1 | Write | Start register 0x0044 | Data: 500 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 06 | 00 44 | 01 F4 | 96 |

If the command has been sent correctly, then the same command will return. Register 0x0044 can check if the value has been written correctly

### 5.9 Search pre-alarm concentration (register 0x0044)

**Send**

| Start | Adr.1 | Read | Start register 0x0044 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 44 | 00 01 | B3 |

**Receive**

| Start | Adr.1 | Read | No. of bytes | Data: 500 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 01 F4 | FF |

Main alarm concentration = 0x01*256 + 0xF4 = 500, the example here reads sensor's main alarm concentration as 500ppm.

### 5.10 Set main alarm concentration (register 0x0045)

**Send**

| Start | Adr.1 | Write | Start register 0x0045 | Data: 1000 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 06 | 00 45 | 03 E8 | 90 |

If the command has been sent correctly, then the same command will return. Register 0x0045 can check if the value has been written correctly

### 5.11 Search main alarm concentration (register 0x0044)

**Send**

| Start | Adr.1 | Read | Start register 0x0045 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 45 | 00 01 | B2 |

**Receive**

| Start | Adr.1 | Read | No. of bytes | Data: 1000 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 03 E8 | FA |

Main alarm concentration = 0x03*256 + 0xE8 = 1000, the example here reads sensor's main alarm concentration as 1000ppm.

### 5.12 Search concentration factor (register 0x004F)

**Send**

| Start | Adr.1 | Read | Start register 0x004F | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 4F | 00 01 | A1 |

**Receive**

| Start | Adr.1 | Read | No. of bytes | Data: 3 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 00 03 | 17 |

Status flag = 0x00 * 256 + 0x03 = 3, specific status information query table 4.1register table description.

### 5.13 Search gas concentration (register 0x000A)

**Send**

| Start | Adr.1 | Read | Start register 0x000A | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 0A | 00 01 | AA |

**Receive**

| Start | Adr.1 | Read | No. of bytes | Concentration: 456 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 01 C8 | 9E |

Concentration = 0x01*256 + 0xC8 = 456, in ppm. The example here reads 456ppm..

### 5.14 Zero calibration (register 0x0047)

**Send the command in the stable environment of 0ppm SF6 gas**

**Send**

| Start | Adr.1 | Write | Start register 0x0047 | Data: 0 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 06 | 00 47 | 00 00 | AE |

If the command has been sent correctly, then the same command will return. Register 0x0047 can check if the value has been written correctly

### 5.15 Search zero calibration concentration (register 0x0047)

**Send**

| Start | Adr.160 | Read | Start register 0x0047 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 47 | 00 01 | B0 |

**Receive**

| Start | Adr.160 | Read | No. of bytes | Data: 0 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 00 00 | 1A |

Zero calibration concentration is 0.

### 5.16 Full scale calibration (register 0x0054)

**Send the command in the stable environment of 1500ppm SF6 gas**

**Send**

| Start | Adr.1 | Write | Start register 0x0054 | Data: 1500 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 06 | 00 54 | 05 DC | 84 |

If the command has been sent correctly, then the same command will return. Register 0x0054 can check if the value has been written correctly

### 5.17 Search full-scale calibration concentration (register 0x0054)

**Send**

| Start | Adr.160 | Read | Start register 0x0054 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 54 | 00 01 | B2 |

**Receive**

| Start | Adr.160 | Read | No. of bytes | Concentration: 1500 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 05 DC | 0B |

Full scale calibration concentration is 1500ppm.

### 5.18 Search sensor's internal temperature (register 0x0003)

**Send**

| Start | Adr.160 | Read | Start register 0x0003 | Register no.: 1 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 00 03 | 00 01 | B8 |

**Receive**

| Start | Adr.160 | Read | No. of bytes | Concentration: 182 | CheckSum |
|---|---|---|---|---|---|
| : | 01 | 03 | 02 | 00 B6 | 02 |

The temperature searched is: 0xB6 * 0.1 = 18.2 ℃

## After-Sales Services and Consultancy

Cubic Sensor and Instrument Co.,Ltd.
Tel: +86 (0)27 81628827 Fax: +86 (0)27 81628821
Add: Fenghuang No.3 Road, Fenghuang Industrial Park, Eastlake Hi-tech Development Zone,
Wuhan 430205, China
Website: www.gassensor.com.cn
Email: info@gassensor.com.cn