GF 7000

Gas Flow Sensor

Analog and Digital(I2C) Output

Datasheet

Version: V1.3

Issued Date:2025.10.10

info@Sencoch.com | 2024-2025 All Rights Reserved | Gas Flow Series

Table of contents

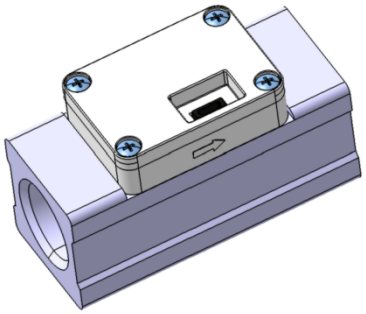
Document Revision History

Revision	Description	Date
V 1.0	Initial version	2024.12.10
V 1.1	Add N PT1/2 and G 1/2 threads	2025.05.18
V 1.2	Add Rc3/4, NPT3 /4, and G3 /4 threads.	2025.05.27
V 1.3	Increase the range and types of gases: O2, Ar, etc.	2025.10.10

The company reserves the right to change the specifications included therein without prior notice. The copyright of the product specifications and the right to final interpretation of the product belong to Sencoch.

1. Product Description

1.1 Product Features



- High flow range to 2000SLM
- High precision, High sensitivity
- Reliable Performance
- Compatible with Analog and Digital Output
- State-of-the-art MEMS technology
- Customizable Thread and Flow Range

1.2 Application Areas

- High-pressure pump flow control
- Gas storage tank flow monitoring
- Semiconductor equipment
- Flow monitoring of coating equipment

2. Function Description

The GF 7000 series measures gas flow rate in a flow channel based on thermodynamic principles. It can measure dry air, nitrogen and other non-volatile substances with high precision. Corrosive gas. It contains a MEMS thermal sensor chip and a high-performance CMOS microprocessor. Combined with a unique calibration scheme, It can output real-time and accurate flow signals.

2.1 Electrical Characteristics

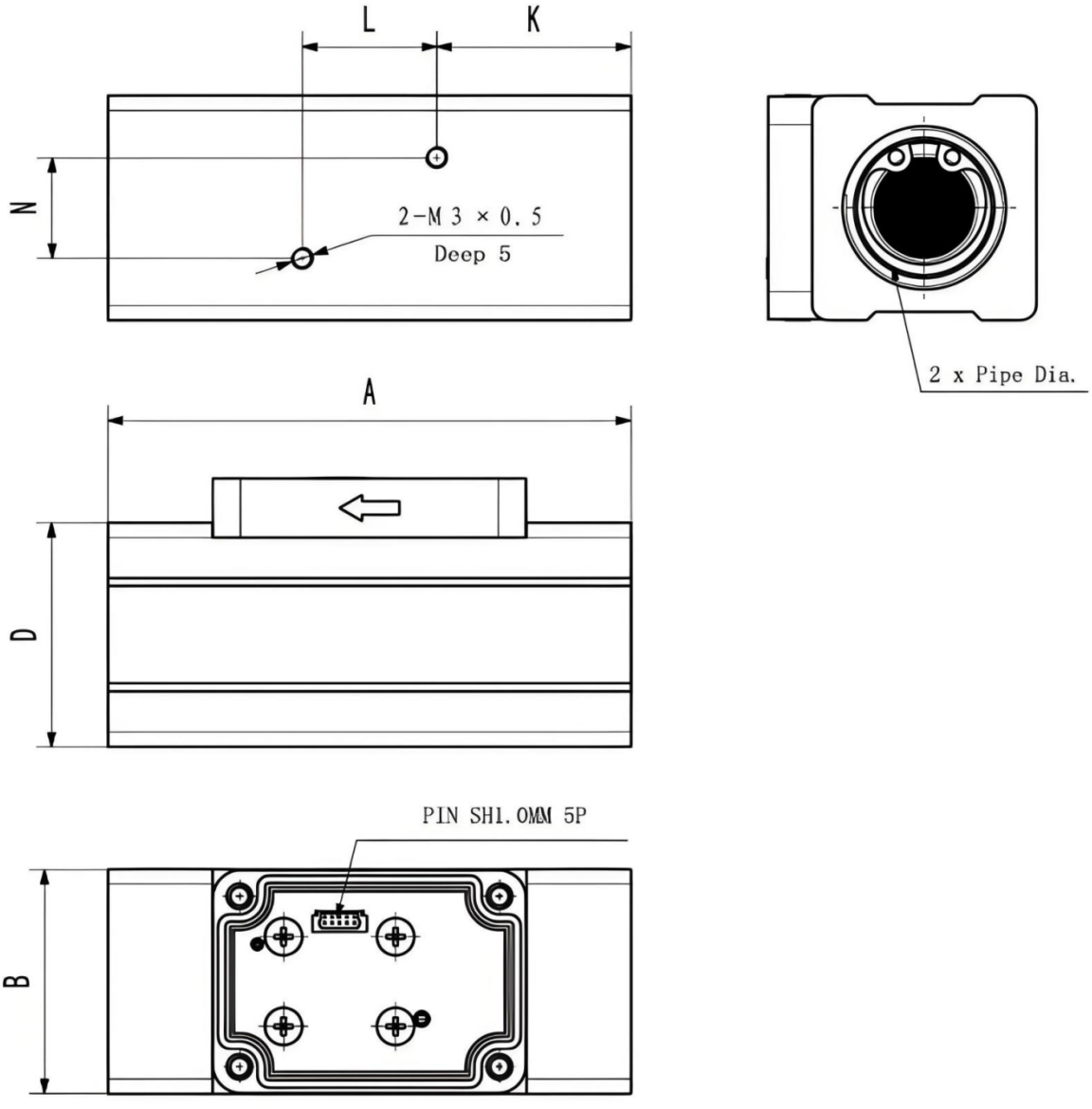
Power supply: (8~24) VDC The default value: 12 ± 0.1VDC

Reference Temperature: 25°C Relative Humidity: 40% ~ 60 % RH

Parameter	Min.	Typ.	Max.	Unit
Accuracy	-	-	± 2.5	%FS
Repeatability		±1%		%FS
Resolution	-	0.1	-	%FS
Response time	-	5 0	-	mSec
Analog output voltage	1.0		5.0	V
SCL clock frequency		-	100	KHz
low-level input voltage	-	-	0.5	V
High-level input voltage	4	-	-	V
Low-level output voltage	-	-	0.5	V
High-level output voltage	4	-	-	V
Operating voltage	8	12	24	V
Operating current	5	15	20	mA
Work pressure	-	-	800	kPa
Pressure tolerance	-	-	1200	kPa
Compensation temperature	0	-	50	°C
Operating temperature	-10	-	60	°C
Storage temperature	-20	-	70	°C
Electrical interface	1.0 mm 5 Pin vertical connector			
Materials that flow gas contact	Silicon,SUS304 , aluminum alloy FR4, PPS , EPDM , epoxy resin			

Note: The product has an internal 4.7k pull-up resistor to the +5V microcontroller power supply.

2.2 External Structure (Tolerance/ Unit: ± 0.3mm)



Pipe Diameter	A	B	D	K	L	N
Rc1/2, NPT1/2	70	30	30	26	18	13.6
G1/2	76	30	30	26	18	13.6
Rc3/4, NPT3/4, G3/4	90	35	35	31	28	16.8

Figure 1 Sensor dimensions(without cover)

2.3 Electrical Connections

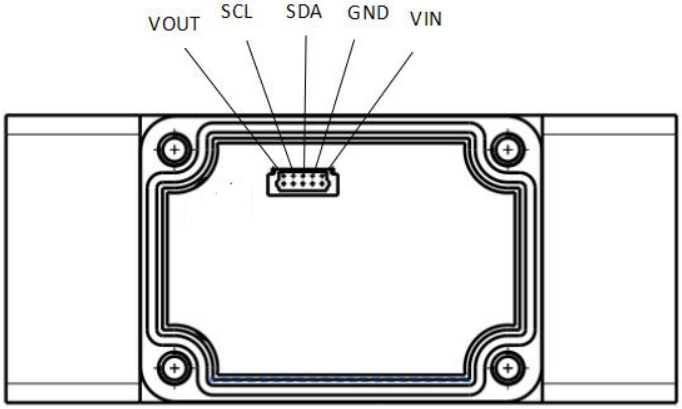


Figure 2. 1.0 mm 5 Pin Vertical Mounting Connector

Pin Definitions

Serial Number	Description	Remark
1	VOUT	Product output voltage signal , 1V~5V
2	SCL	I2C clock signal
3	SDA	I2C data port
4	GND	Power Negative
5	VIN	Power input port, 8~24V

3. Order Guide

G F 7501 M - XY01 - R04 - A

GF	Product Category Flow sensor
7501	Range 501: 500SLM;
M	Output M: compatible with both digital and analog outputs
XY01	Scheme Code
R	Types of Threads R: Rc N: NPT F: G
0 4	Piping Diameter 04 : 1/2 06 : 3/4
A	Measured Gas A: Air; N: Nitrogen; O: Oxygen; Ar: Argon; C: Carbon dioxide
WX	Company interior code

4. Routine Measuring Ranges

Model	Flow Range (SLM)
GF 7101	1-100
GF 7301	3-300
GF 7501	4-500
GF 7102	8-1000
GF 7202	16-2000

Note: S L M Standard liters per minute. Standard conditions. 0°C, 101.325 kPa .

5. Flow Rate Calculation Formula

Analog output:

GF7501

Flow=[(Vout - 1.0 V) / 4 V] ×Full-scale flow rate

For example, GF7501 When the output voltage is read as 3.0 V,

The instantaneous flow rate is [(3.0V - 1.0 V) / 4V × 50 0 S L M] = 250 SLM

Digital output:

GF7501

flow =[I2C output reading]/proportional coefficient K

(K=10 @FS ≥ 500SLM, K=10 0@FS ≤ 300SLM)

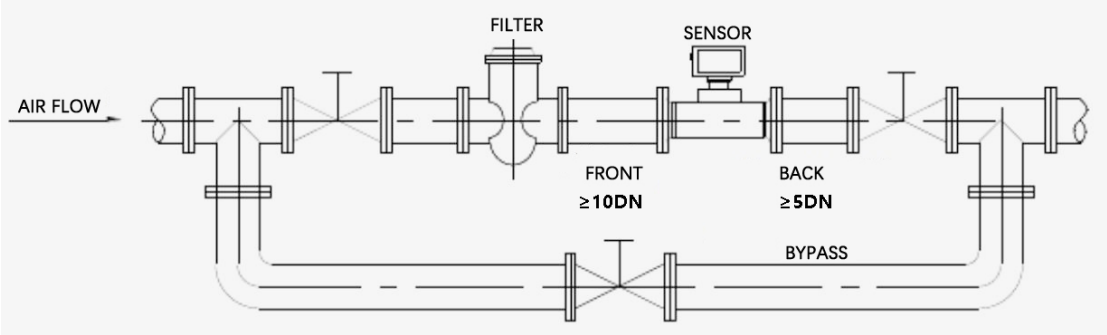
For example, GF 7501 When the output is 3055 The instantaneous flow rate is

[3055]/10 = 3 05.5 S LM

6. Precautions for Use

6.1 Recommended installation instructions

GF 7000 Series Installation The recommended pipe length is 10 parts at the front and 5 parts at the back. That is, the length of the inlet pipe is ten times the diameter of the product. The length of output pipe is five times the diameter; the products are installed concentrically; Air flow direction need be consistent with product indication direction; product pin connections must correspond to product pin definitions. After confirming that everything is correct, power on and it will work. See the diagram below:



6.2 Precautions for use

1) The product can only be used normally in the environment defined in this specification;

2) Pay attention to the gas flow direction indicator during installation. Connection and leak detection should be carried out in accordance with the relevant procedures;

3) During product use, it is prohibited to simultaneously install pipes, clean pipes, or perform other improper operations that introduce large amounts of impurities, as this may damage the product.

4) If the gaseous medium contains water vapor Impurities maycause a decrease or damage to the sensor 's sensitivity.

5) Pay attention to the positive and negative terminals of the power supply. If the positive and negative terminals of the power supply are reversed This will cause the internal circuitry of the sensor to burn out. This affects the normal use of the product.

■ Please confirm under actual usage conditions.

Since this specification is for a single product, please verify the performance and quality under actual use to improve reliability.

6.3 Safety Precautions

This product is used in general electronic devices (communication equipment) Measuring equipment (Work machinery, etc.) It is made from semiconductor components. Products using these semiconductor components It may malfunction or fail due to external interference and surges. , Therefore, please use it in practice. Verify performance and quality under test conditions. Just in case. Please incorporate safety features into the device (fuse). Circuit breakers and other protective electrical circuits Road layout (Multi-functional devices, etc.) Even if a misoperation occurs, it will not endanger life, body and property , etc. To prevent injury and the occurrence of the accident Please be sure to comply with the following:

(1) The drive current and voltage should be used below the rated values ;

(2) Please wire according to the electrical definitions. Especially after reversing the power supply connection. It may be due to fever, smoke accidents caused by circuit damage such as fire, therefore please take note;

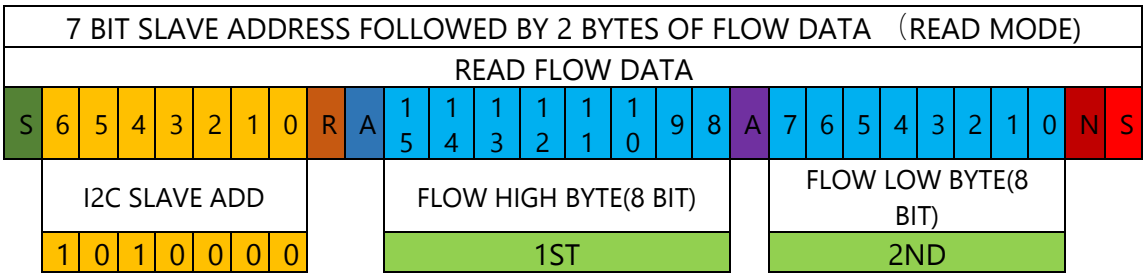
(3) Please be careful when securing the product and connecting the pressure inlet.

7. I2C Communication Protocol

The I2C protocol is a standardized protocol for exchanging information between integrated circuits or functional units; The I2C bus uses a single data line (SDA), add a clock line (SCL) to complete data transmission and expand peripheral devices. There are three data transmission methods on the I2C bus. Transmission speed: Standard mode, Fast mode, and High-speed mode. Standard mode is 100 Kbps, and Fast mode is 400 Kbps. (This sensor is only) Supports standard mode transmission speeds. Addressing of each node is done via soft addressing, saving chip select lines; the standard addressing byte SLAM is 7 bits. It can address 127 units.

The default address for I2C is 0x32.

I2C Read Flow Data Communication Format



- S: START Condition
- 6: SLAVE Address Bit
- R: READ Bit(Read=1)
- A: ACK from SLAVE
- 15: Date Bit
- A: ACK from Master
- N: NACK from Master
- S: STOP Condition

The flow data is divided into 2 bytes, the first byte is the high byte of the flow data, the second byte is the low byte of the flow data, the flow data is K (see airflow calculation formula) times the actual flow rate. The default unit of flow data is mL/M (millilitres/minute), for example, when K=10, read the hexadecimal data as 03E8, then convert to decimal current actual flow rate is 100.00mL/M.

Example: Standard program for reading flow sensor via digital output

```
//Read the flow sensor module routine

// SDA, SCL correspond to the respective I/O pins of the microcontroller.

//host sends  0 xA 1  Give to slave machine From the machine back  2  Byte data to host

#include " IIC_Master.h "
#define SDA PA 0
#define SCL PA1
Unsigned int IIC_RX_Buf[2];
bit ErrorBit;
void I2C_Init(void)
{
   SDA _ INPUT =0; //Program initialization, the SDA pin
   is set to output
   SCL_INPUT. =0; // Program initialization sets the SCL
   pin as an output.   SDA =1;
   SCL =1;
}
Void I 2C_Start (void)
{
   SDA =1;
   Delay_Us(20);
   SCL =1;
   Delay_Us(20);
   SDA =0;
   Delay_Us(20);
   SCL =0;
   Delay_Us(20);
}


//-------------------------------------------------------------------------
void I2C_Stop(void) {
   SCL =0;
   Delay_Us(20);
   SDA =0;
   Delay_Us(20);
```

```
    SCL=1;
    Delay_Us(20);
    SDA=1;
    Delay_Us(20);
}
//------------------------------------------------------------------------------
-
void I2C_ACK(void) {
    SDA=0;
    Delay_Us(20);
    SCL=1;
    Delay_Us(20);
    SCL=0;
    Delay_Us(20);
}
//------------------------------------------------------------------------------
-
void I2C_NoAck(void) {
    SDA=1;
    Delay_Us(20);
    SCL=1;
    Delay_Us(20);
    SCL=0;
    Delay_Us(20);
}
//------------------------------------------------------------------------------
-
Unsigned int I2C_ ReadByte (void)
{
    Unsigned int ucValue=0;
    Unsigned int ucIndex;


    SDA=1;
    Delay _ Us (20);
    SDA _ INPUT=1; // Set the SDA pin as input
```

```
  Delay_Us(20);
  for ( ucIndex = 0; ucIndex < 8; ucIndex++ )
  {
     ucValue <<= 1;
     SCL=0;
     Delay_Us(20);
     SCL=1;
     Delay _ Us (20);
     if ( IIC_DAT )==1) // IIC _ DAT That is, SDA is set as input, Read SDA pin level
      { ucValue = ucValue |0x01;}
     else
     { ucValue = ucValue & 0 xfe ;}
     Delay _ Us (20);
     SCL=0;
     Delay _ Us (20); }
  SDA _ INPUT=0; // will SDA is set to output
  Delay _ Us (20);
  return ucValue; }
//------------------------------------------------------------------------------
void I2C_WriteByte( unsigned int ucData )
{
  u8 i;
  for( i = 0; i < 8; i++ )
  {
     SCL=0;
     Delay_Us(20);
     if((ucData & 0x80) == 0x80)
     {
        SDA=1;
        Delay_Us(20);
     }
     else
     {
        SDA=0;
        Delay_Us(20);
```

```
        }

        SCL=1;
        Delay_Us(20);

        SCL=0;
        Delay_Us(20);
        ucData <<= 1;

    }
    SCL=1;
    Delay_Us(20);
    ErrorBit = IIC_DAT;
    Delay_Us(20);
    SCL=0;
    Delay_Us(20);
}

void iic_master_proc(void) {
    Unsigned int count=2,i;
    I2C_Init() ;
    I2C_Start();
    I2C_ WriteByte (0 xa 1); //Write address 0 xA 1
    for(i = 0;i < count;i++)
    {
        IIC_RX_Buf[i] = I2C_ReadByte();
        if(i < (count -1)) I2C_ACK();
        else               I2C_NoAck();
    }
    I2C_Stop();

}
```