GF3000

Gas Flow Sensor

Linear output

Datasheet

Version: V1.5

Issued Date:2024.10.08

Table of contents

Document Revision History

Revision	Description	Date
V 1.0	Initial version	2020.11.03
V 1.1	Revise parameter, protocol	2020.12.26
V 1.2	Revise curve, parameter	2021.01.08
V 1.3	Revise parameter	2023.01.09
V 1.4	Add custom version	2024.06.13
V 1.5	Add temperature definition	2024.10.08

1. Product Description

1.1 Product Features

- ●State-of-the-art MEMS technology
- ●High sensitivity, Wide measurement range
- ●High precision, High resolution
- ●Reliable performance
- ●Fast response optional
- ●Bidirectional air flow optional
- ●Analog or digital output optional

1.2 Application Areas

- ● Portable ventilator, small home oxygen concentrator
- ● Continuous Positive Airway Pressure (CPAP) device
- ● Anesthesia for childbirth
- ● Critical care equipment
- ● Air purifier
- ● Environmental climate monitoring
- ● Fuel cell control
- ● HVAC (Heating, Ventilation, and Air Conditioning)

2. Function Description

The GF3000 series measures the small gas flow rate from 0 to 3000SCCM based on thermodynamic principles. It can measure dry air, nitrogen and other non-corrosive gas with high precision. 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
Optional Flow Range		0		3000	
Accuracy①	≤1000SCCM	-	-	± 1.5	%FS
	> 1000SCCM&≤2000SCCM	-	-	± 2.5	%FS
	> 2000SCCM&≤3000SCCM	-	-	± 3.5	%FS
Offset Drift		-	0.2	-	%FS
Resolution		-	0.1	-	%FS
Analog output voltage②		0.5		4.5	V
SCL clock frequency③			-	100	KHz
low-level input voltage③		-	-	0.5	V
High-level input voltage③		4	-	-	V
Low-level output voltage③		-	-	0.5	V
High-level output voltage④		4	-	-	V
Operating voltage	Standard Version	8	12	24	V
	Low Power Supply	4.75	5	5.25	V
Operating current		5	15	30	mA
Response time⑤	Standard Version	-	20	-	mSec
	High Response	-	5	-	mSec
Work pressure		-	-	200	kPa
Compensation temperature		0	-	50	°C
Operating temperature		-10	-	60	°C
Storage temperature		-20	-	70	°C
Electrical interface		2.54 mm 5 Pin vertical connector			
Materials that flow gas contact		Silicon， FR4, PPS , Fibreglass, epoxy resin			

Note:

① Different flow range have different accuracy

② This specification is for analog output only. Contact Sencoch for custom voltage.

③ This specification is for digital output only.

④ The SDA/SCL of the product is internally pulled up to the power supply, so no external pull-up resistor is required.

The range is 0 – (100–3000 SCCM) and can be customized.

⑤ Different response times correspond to different output stability, please select according to actual requirement

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

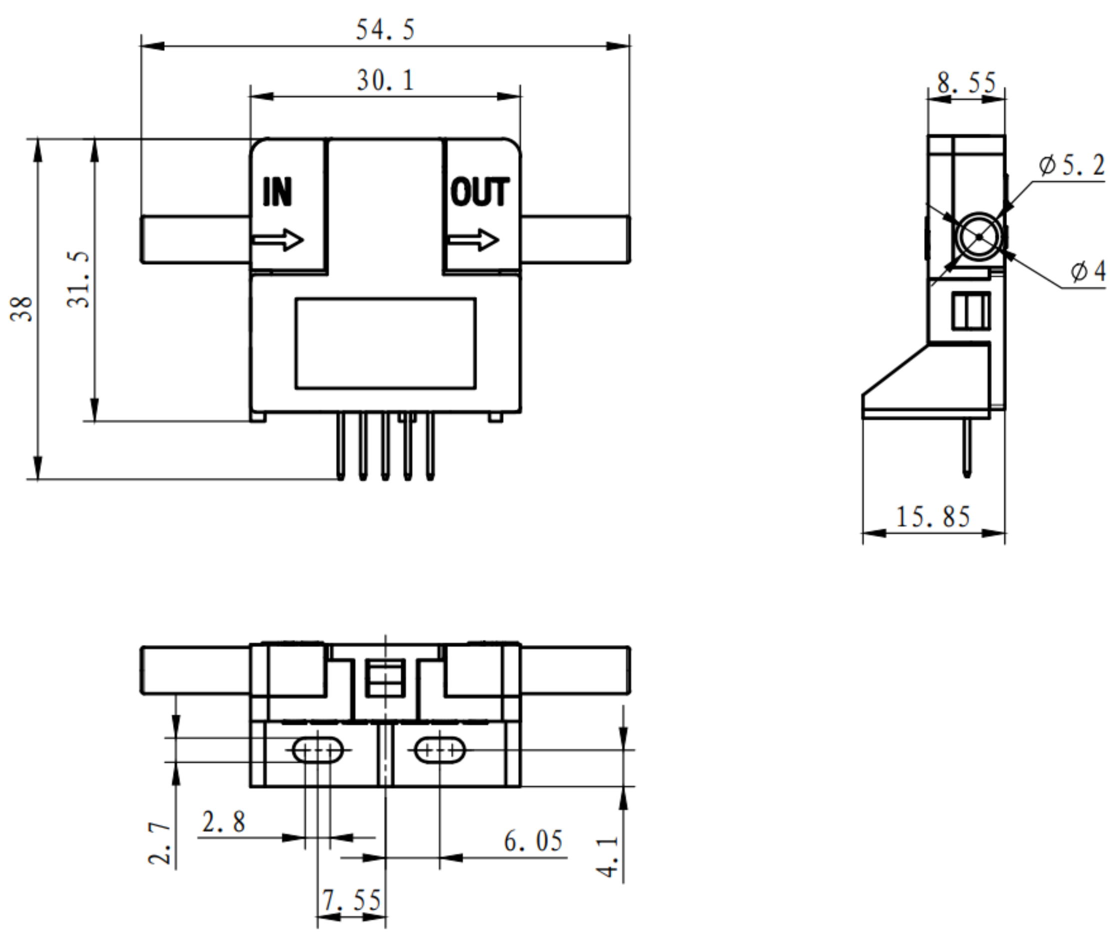


Fig.1 Sensor dimensions

2.3 Electrical Connections

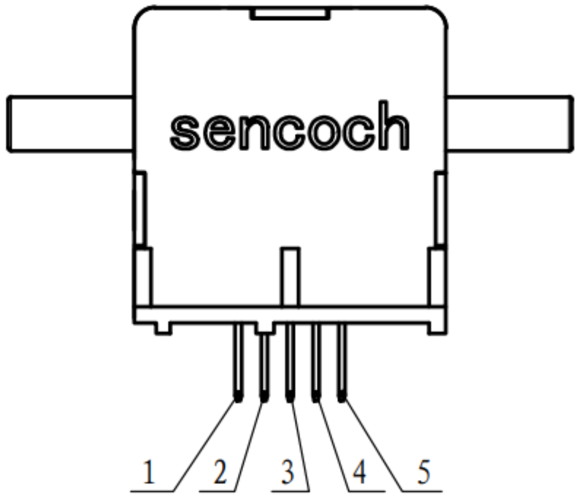


Fig.2 2.54mm 5 Pin Vertical Mounting Connector

Pin Definitions for Analog output

Serial Number	Description	Remark
1	VOUT	Product output voltage signal
2	VDD	Power supply input positive
3	GND	Power supply input negative
4	NC	No connect
5	NC	No connect

Pin Definitions for I2C Interface

Serial Number	Description	Remark
1	NC	No connect
2	VDD	Power supply input positive
3	GND	Power supply input negative
4	SCL	I2C clock signal
5	SDA	I2C data port

※※※ This product has no reverse polarity protection, please pay attention to the power polarity during assembly.

3. Order Guide

GF 3 202 F -L- B - A0545 A WX

GF	Product Category Flow sensor
3XXX	Product Series 3000 Series
202	Range 202: 2000SCCM;
F(Optional)	High response version; Default Standard version
L(Optional)	5V Power supply voltage; Default 8-24V
B	B: Bidirectional version; Default 8-24V
A0545	Output A0545: 0.5~4.5V or A1050: 1.0 ~5.0V; D: I2C interface
A	Gas Type A: Air; N: Nitrogen; O: Oxygen; C: Carbon dioxide
WX	Company interior code

4. Routine Measuring Ranges

Model	Flow range (SCCM)
GF3012	0-100
GF3022	0-200
GF3032	0-300
GF3042	0-400
GF3052	0-500
GF3062	0-600
GF3102	0-1000
GF3152	0-1500
GF3202	0-2000
GF3302	0-3000

Note: SCCM Standard milliliter per minute. Standard conditions. 0°C, 101.325 kPa.

5. Flow Rate Calculation Formula

Analog output:

GF3102-A0545(Unidirectional)

Flow rate = [(Vout - 0.5 V)/4 V] x full scale flow rate

For example: XGZF3102-A-A, when reading the output voltage of 2.5V,

The Instantaneous flow rate is [(2.5V - 0.5V)/4V x 1000 SCCM] = 500 SCCM

GF3102-B-A0545(Bidirectional)

Flow rate = [(Vout - 0.5 V)/2 V] x full scale flow rate

For example: XGZF3102-B-A0545-A, when reading the output voltage of 1.0V,

The Instantaneous flow rate is [(1.0V - 2.5V)/2V x 1000 SCCM] = -750 SCCM

Digital Output

GF3102-D(Unidirectional)

Flow rate = [I2C output reading]/proportionality factor

Example: XGZF3102-D-A, when reading output 5055

Instantaneous flow rate is [5055]/10 = 505.5 SCCM

K Full Scale Range	K
100SCCM≤ flow rate < 500SCCM	100
500SCCM≤ flow rate≤3000SCCM	10

GF3022-B-D(Bidirectional)

Flow rate = [I2C output reading]/proportionality factor K

Example: XGZF3022-B-D-A, when reading output 15306

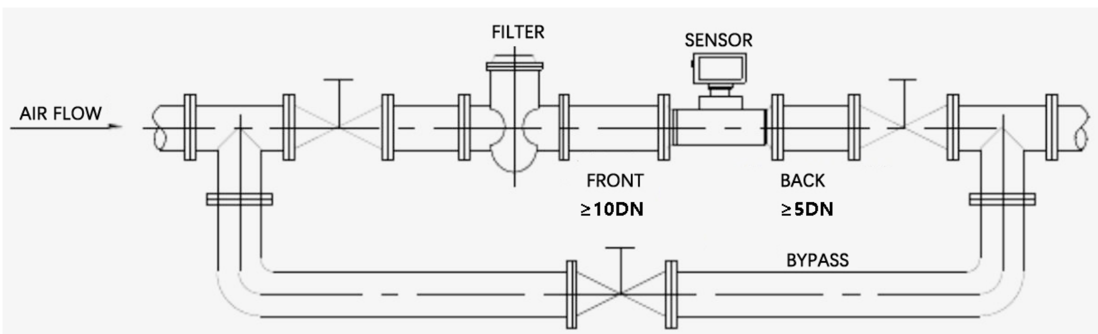
Instantaneous flow rate is [15306]/100 = 153.06 SCCM

K Full Scale Range	K
100SCCM≤ flow rate < 300SCCM	100
300SCCM≤ flow rate≤3000SCCM	10

6. Precautions for Use

6.1 Recommended installation instructions

GF 3000 Series Installation 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 may cause 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 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

7 BIT SLAVE ADDRESS FOLLOWED BY 2 BYTES OF FLOW DATA （READ MODE)																												
READ FLOW DATA																												
S	6	5	4	3	2	1	0	R	A	15	14	13	12	11	10	9	8	A	7	6	5	4	3	2	1	0	N	S
	I2C SLAVE ADD									FLOW HIGH BYTE(8 BIT)									FLOW LOW BYTE(8 BIT)									
	1	0	1	0	0	0	0			1ST									2ND									

- 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 (milliliter/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 I 2C_ 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();

}
```