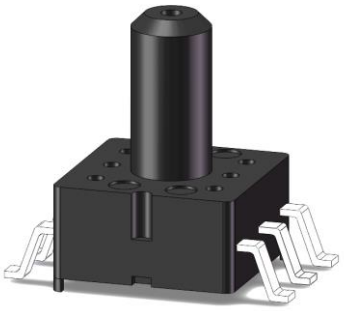
Specification

XST-SV-SOP6-040D gauge pressure sensor



● Product Description

The XST-SV-SOP6-040D is a digital output differential pressure sensor consisting of a MEMS piezoresistance pressure sensor chip and a dedicated signal conditioning chip. The ASIC includes 24-bit Sigma-Delta ADC, OTP memory, and interface circuits. Calibration data stored in the OTP memory can be used for the calibration of the XST-SV-SOP6-040D. Each module is compensated by a linear checksum temperature, so that its voltage output signal is linearly proportional to the input pressure, and it is not affected by the operating temperature.

● Product features

➢Pressure type: gauge pressure type

➢Pressure range: -40~40kPa

➢Standby current: <0.1Aμ

➢High resolution: 2Pa (RMS, high resolution mode)

➢Digital I 2C interface: Air pressure ADC resolution 24 Bit

Temperature ADC resolution of 16 Bit

➢Operating temperature range: -20℃~85℃

➢High reliability, good stability, and low long-term drift

● Product application

➢Water level pressure test

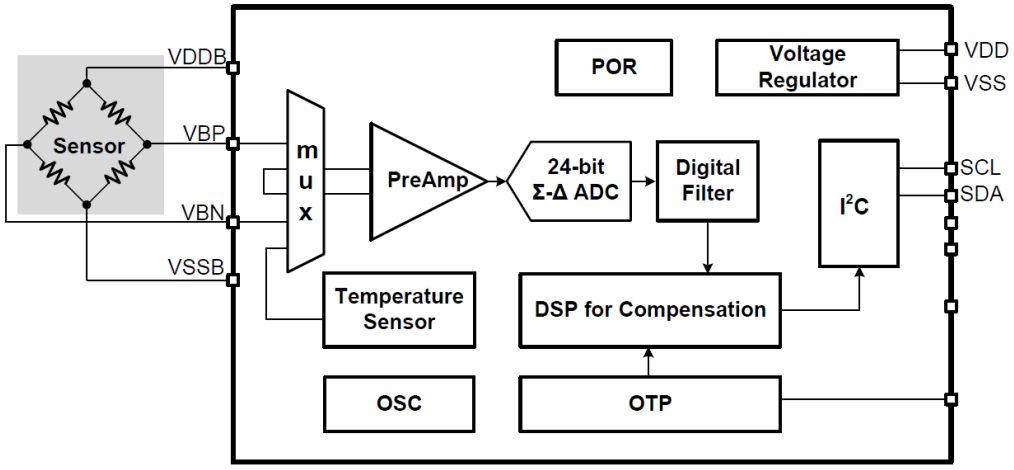
➢Industrial / engineering air pressure control

➢Gauge pressure sensor system

➢medical instruments

catalogue

1. Device internal function block diagram



2. Limit parameters

parameter	condition	minimum	maximum	unit
Storage temperature		-40	105	°C
service voltage	All the pins	-0.3	3.6	V
IO pin voltage	All the pins	-0.3	3.6	V
ESD (HBM)		-2	2	k V
overload			15	k Pa

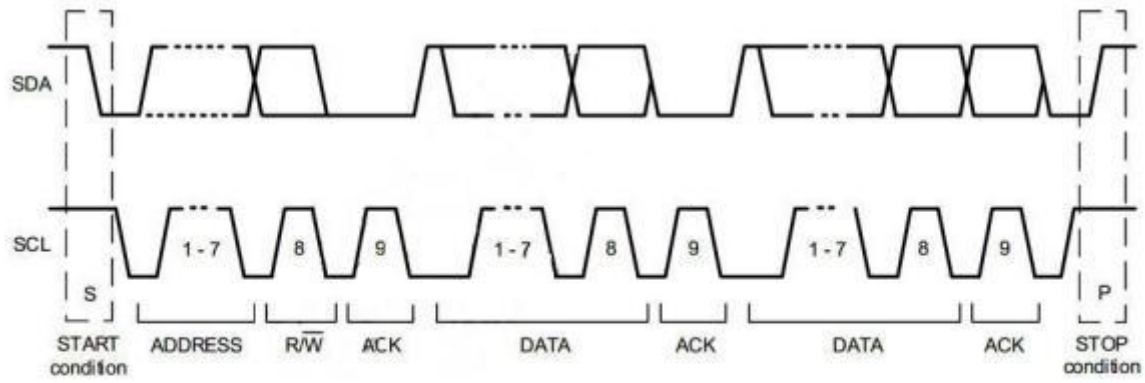
3. Specifications

（Vs=3.3V DC, T_A=25℃）

parameter	symbol	condition			min	TYP	MAX	UNIT
working temperature	TA				-20		85	°C
Work range	P				-40		40	kPa
service voltage	VDD				1.8	3.3	3.6	V
Current consumption (1 Hz)	Idd	work pattern	OSR_P	OSR_T				µA
		Low power	1024X	2048X		10		
		Lower-low-power mode	2048X			14		
		standard	4096X			19		
		High precision	8192X			28		

		Ultra high precision mode	16384X			46		
Standby current	Id dsbm	@25℃				0.1		
Pressure / temperature measurement time	T c	work pattern	OSR _P	OSR _T				m s
		Low power consumption mode	1024X	2048X		13		
		Lower-low-power mode	2048X			19		
		standard mode	4096X			31		
		High precision mode	8192X			56		
		Ultra high precision mode	16384X			105		
ADC, resolution ratio		pressure				24		B it
		temperature				16		B it
Pressure accuracy	P _A	0-65°C			- 1		1	%F .S
Temperature accuracy	T _A	0~65°C			-1.5		1.5	°C
Serial data clock	fI2C	I2C					3.4	M Hz

4. I 2C communication protocol



I 2C communication protocol

➢START Condition

The SDA switches from idle high to low, when the SCL remains high; repeat start condition repeatedly during transmission, indicating that communication will restart without an intermediate stop bit.

➢ Address Bits

During the first byte transfer, the first 7-bits provides the specified address of the device, subject by default to 0x78, where the device will answer this communication.

➢ Read /Write Direction Bit

During the first byte transmission, the last bit indicates the communication direction, 0 indicates the master device write operation, 1, indicates the master device read operation; if the master device requests the read from the device, the main device will control the SDA line output data in the later byte.

➢ Data Byte

All other bytes, except for the address and read and write bits, transmit the data bytes considered for communication on the SDA.

➢ Acknowledge or Not Acknowledge Bit

The response bit is used to tell the sender that the bytes have been received, the device needs to answer each byte, including the address byte; at this moment, the bus device sending the data stops driving the SDA, the line and the SDA line is raised; does not answer a byte, the receiving device does not need to do anything; answer a byte, the receiving device needs to pull the SDA down.

A receiving slave device does not need to answer, if the receiving device is not addressed or the device cannot process the received bytes; if the main device is in receiving and wants to end the communication, the device needs to generate a stop bit if the response is not encountered.

➢ Stop Condition

The SDA switches from low to high state, and the SCL remains high, when the I²C communication ends.

5. Working mode and operation sequence

The product is powered up only after receiving the corresponding I^2After the C command, the pressure and temperature measurement and calibration process will be initiated, and the measurement will automatically enter the deep dormant state to save power consumption. In deep dormancy, divide I in the internal circuit^{2}All other circuits outside the C interface are closed, and the current consumed is about 0 μ A..1

Sensor IIC Address: The sensor 7bits I2C default device address is 0x78

working order:

1) Need to wait after power, refer to the timing of power
2) The IIC write is made
3) Waiting
4) The IIC reads
5) Value conversion

5.1 Uppower time sequence

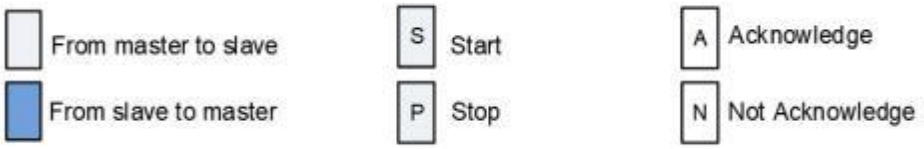
The electric timing				
start time	T STA 1	The VDD rises to the time of interface communication	1	m s
	T STA 2	VDD rises to the start of measurement	2.5	m s
Wake up time	T WUP 1	Dormant time until the interface starts	0.5	m s
		communication		
	T WUP 2	Time from the dormant state to the start of the measurement	2	m s

.25 The I I C write command

Write the command 0xAC or 0xBX (X represents a number from 1 to 6) to get data calibrated using an internal algorithm, either, and recommending the default mode.

5.2.1. Send the Get_Cal command (0xA C), in the default mode

Send the 0xAC command to obtain the calibration value as:



S	0XF 0	A	0X AC	A	P

Table: Send the Get_Ca l write command

The 0xF 0 in the write command indicates that the default 7bits I2C device address is 0x78, and the last 1bit of 0 indicates the master device write operation. The over-sampling rate of the Get _Cal command is determined by the configuration in the OTP, and is fixed and constant.

5.2.2 Or send the Get _Cal _S command (0xBX), (X represents a number in 1 to 6)

The Get _Cal _S command (0xB1 ~ 0xB6) is almost the same as the Get_Cal command (0xAC), except that the oversampling rate of the ADC is measured differently. The oversampling rate of the Get _Cal _S command is determined by the content of the command itself. After the factory has calibrated the sensor and burned the OTP data, the Get _Cal _S command allows the user to complete the measurements with different overrates to achieve high, moderate, or low precision measurements.

Name explanation: oversampling, similar to hardware filtering, the larger the number, the better the accuracy, the more time consumed, the higher the power consumption.

S	0XF 0	A	0X BX	A	P

Table: Send the Get_Cal_S write command

Co mmand 0xBX (HEX)	F u nction	Deta il
The first [3] Bit of the X	ADC, OSR_T	0:2048x Oversampling rate
X's # 00 [2:0] Bit	Oversampling rate of the ADC OSR _P when measuring a pressure bridge	001:16384x Oversampling rate 010:8192x Oversampling rate 011:4096x Oversampling rate 100:2048x Oversampling rate 101:1024x Oversampling rate 110:512x, oversampling rate

Note: Sending 0xB 3 (standard mode) has the same effect as sending (0xAC)

5.3 Waiting

The finished write command needs to wait time before sending the read command because it takes time to complete the entire measurement internally. The length of waiting time depends on

the pressure oversampling rate and the temperature oversampling rate setting (the temperature oversampling rate is fixed to 2048x). Different over-sampling rates correspond to different waiting times.

The waiting time does not need to be calculated. You can judge whether the acquisition has been completed by constantly reading the IIC status words.

The following time is the time of temperature measurement + pressure measurement.

ADC change								
ADC slew rate	FS ,raw	The OSR ranges from 1024 X to 16384 X			20		1350	Hz
Pressure / temperature MT	T c	work pattern	OSR _P	OSR _T				m s
		Low power consumption mode	1024X	2048X		13		
		Lower-low-power mode	2048X			19		
		standard mode	4096X			31		
		High precision mode	8192X			56		
		Ultra high precision mode	16384X			105		

Table: The ADC conversion time

Bitmain	meaning	description
Bit 7	continue to have	Fixed to 0
Bit 6	Power indication (Power indication)	1 Equipment on power (VDDB on); 0 Equipment off the electricity
Bit 5	Busy and busy instructions (Busy indication)	1 The device is busy, indicating that the data required by the latest I2C command is not valid. If the device is busy, the new command is not processed. 0 Indicates that the data required by the latest I2C command is ready to be read
Bit 4	continue to have	Fixed to 0
Bit 3	operative mode (Mode Statu s)	0 NOR mode (the default mode) 1 C MD mode
Bit 2	Memory data integrity indicatio n (Memory integrity/error fla g)	0 Indicates that the OTP memory data integrity test (CRC) has been passed; 1 The integrity test has failed. The test of data integrity is calculated only only once during the power process (POR), so the new CRC value written can only be used after the next POR.(Default mode)
Bit 1	continue to have	Fixed to 0
Bit 0	continue to have	Fixed to 0

Table: Status byte bitbit description

.45 The I I C Read command

To ensure that the time interval between writing instructions and reading instructions is greater than the measured time to read out the calibration data, the read number format is shown in Figure 4. The 0xF1 in the read command indicates that the default 7bits I2C device address is 0x78, and the last 1bi t of 1 indicates the main device read operation. The calibration data read is 6 bytes, then the 1-byte status word, 3, the byte air pressure correlation value BridgeDat [23:16], BridgeDat [23:16], BridgeDat [23:16]

2 Byte temperature correlation value: TempDat [15:8], TempDat [7:0].

Read the pressure and temperature to read out together. If the temperature values are not required, the temperature conversion can not be calculated.

S	0XF 1	A	Statu s	A	Bri dgeDat [23:16]	A	Bri dgeDat [15:8]	A	Bri dgeDa t [7:0]	A	Te mpDat [15:8]	A	Te mpDat [7:0]	N	P

Figure: I2C reads 3 byte pressure correlation or 2 byte temperature correlation

5.5 Conversion

Pressure value conversion formula: input and output relationship:

$$\text{Pressure} = \cdot\frac{P_{MAX} - P_{MIN}}{D_{MAX} - D_{MIN}}(\text{D test} - \text{D MIN}) + \text{P MIN}$$

Pressure: actual pressure value; Dtest: digital output value of sensor; P MIN: sensor zero pressure value; P MAX: pressure value of sensor full range; D MIN: digital output value at sensor zero; D MAX: digital output value of sensor full range;

Example: After reading the calibration data, a simple conversion of the unsigned number in a percentage.

To facilitate understanding, we assume that the calibration data read is: 0x04 0x9B 0xB0 0xC5 0x56 0x AA

0 The x 04 is the status word: Bit5 If 1 indicates that the last I2C is busy, you to wait while. If a Bit5 of 0 indicates that the device is not busy, the data can be read. Referring to the previous table for a detailed description of each bit of the status word.

0x 9B 0xB0 0xC5 Three bytes are the pressure-related bridge calibration values

0x56 0xAA Two bytes are the temperature calibration values

Pressure conversion: convert the unsigned number 0x9B 0xB0 0xC5 to the decimal number of 10203333,

This calculation assumes that the calibration range used is-40 ~ 40kPa, and the corresponding AD output is 2516582.4~14260633.6 (15%AD ~85%AD)

According to the calibration formula of air pressure input and output relationship:

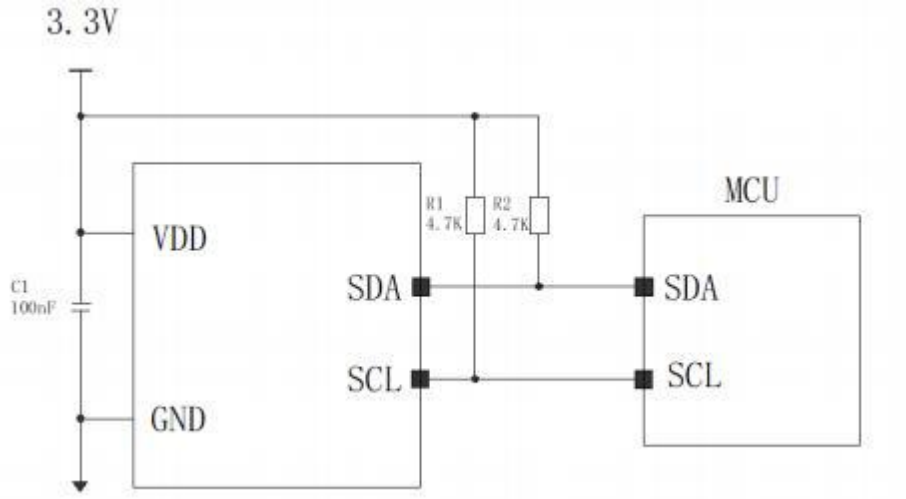
Actual pressure value = (40 + 40) / (14260633.6-2516582.4) * (10203333-2516582.4) -40 = 12.362 kPa

Temperature conversion: To convert the unsigned number 0x 56 0xAA to a decimal number of 22186, because the read calibration data is expressed in percentage form, which is numerically equal to the ratio of the converted decimal number to the maximum of 16 bits (65535), the conversion percentage can be calculated as follows

22186/65536*100% = 33.85%

The calibration range of the temperature is specified as-40℃ ~150℃, so the calibration value = (150- (-40)) * 33.85% -40 = 24.32℃

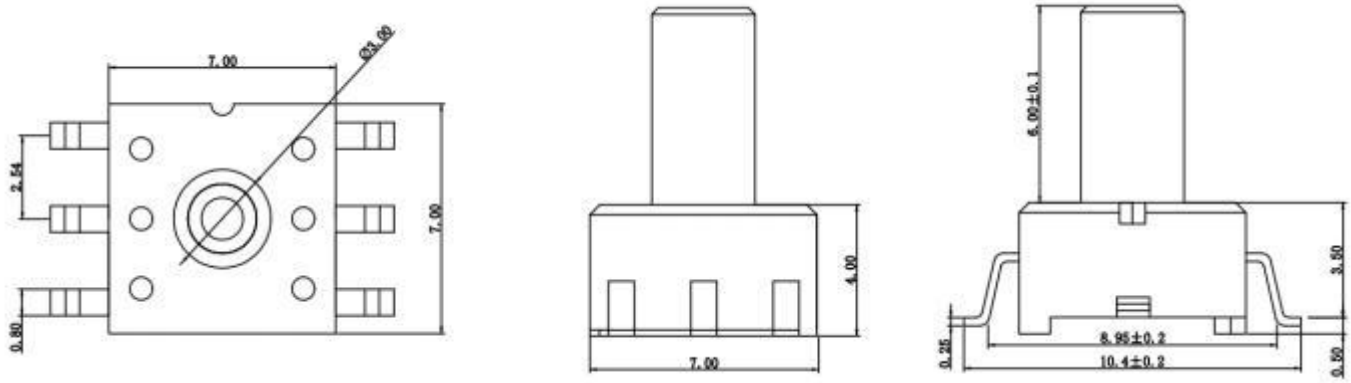
6. Typical applications



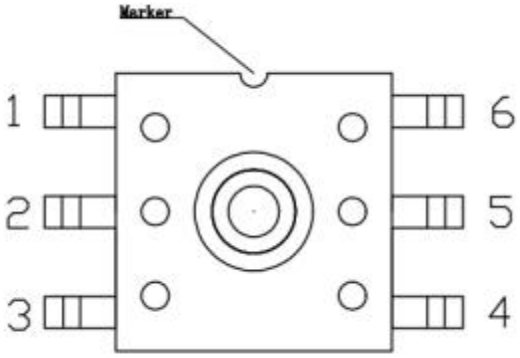
7. Mechanical characteristics

7.1 Overall dimensions

unit:mm



7.2 Pin definition



Pin configuration diagram

pin	definition	explain
1	V DD	Power positive pole
2	S CL	I^2C Communication time sequence
3	S DA	I^2C Communication data
4	NC	Empty not to receive
5	GN D	Power anode
6	NC	Empty not to receive

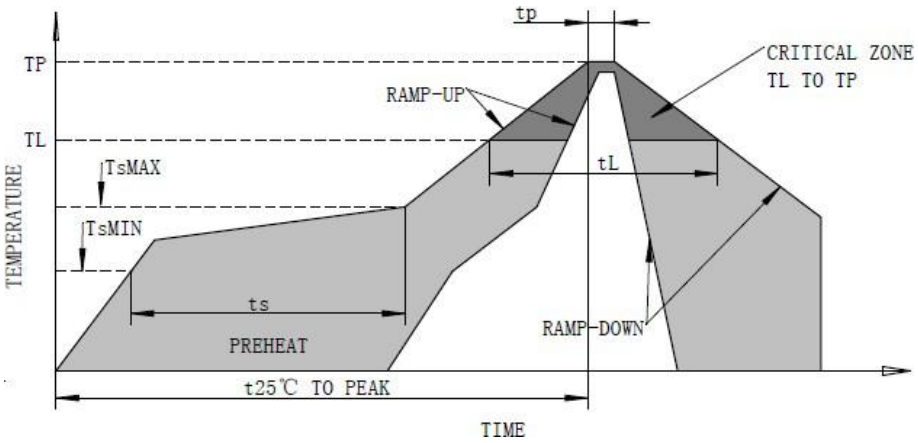
Note: NC feet should be suspended, otherwise it may cause product function failure.

8. Precautions

8.1. Welding

8.1.1. Reflow welding (SMD type)

Recommended reflux welding parameters:



a) Average heating rate: MAX 3℃ / S

b) Preheat: minimum temperature (TsMIN) 150℃, maximum temperature (TsMAX) 200℃, Ts 60~80S

c) Time holding: temperature (TL) 217℃, time (tL) 60~150S

d) Peak temperature (TP): 260℃, 25℃ to peak temperature time: MAX 8Mi n

e) Time at actual peak temperature (TP) (tP): 20~40S

f) Cooling rate: MAX 4 ℃/S

8.1.2. Wave soldering (DIP type)

A) Preheat for 30~120 seconds, the temperature is controlled at 90~130℃, too low temperature will lead to poor welding;
b) The welding temperature is set at 240~260℃, and the welding time is not more than 5 seconds. For too long, tip bending and tin connection will occur elephant;

8.1.3. Manual welding

a) During welding, 25W externally heated or 20W internally heated electric soldering iron shall be selected, and the set temperature shall be 260~320 ℃. Single point welding shall not exceed 5 seconds. cover The welding metal surface shall be kept clean. Oxide, dust, oil stain, etc. will prevent the solder from wetting the surface of the metal to be welded, which can be removed by mechanical or chemical methods before welding;
b) When welding is completed, pay attention to the time, direction and speed of moving the welding head to avoid the risk of poor welding due to product displacement;

8.1.4. Welding and repair

A) Repair the pressure sensor product and its surrounding devices are strictly prohibited to use the hot air gun, SMD type products should choose the use of point tin paste over the reflux way;

In b) Manual maintenance with soldering iron, avoid long contact with sensor Pad to avoid product failure or loss of welding disc (including PCB disc);

c) Pay attention to the protection of the pressure sensor during the welding process of the connector, do not directly touch the device by hand, and prevent the solder from falling into the device.

8.2. Cleaning

After welding as far as possible with a special printing board cleaning agent cleaning

A) It is strictly prohibited to clean or wipe the pressure sensor in a liquid, even during the cleaning of the surrounding devices or the liquid wiping
Can have liquid into the barometer inside, to cause abnormal output;

b) Pay attention to the protection of the pressure sensor during the welding of the connector, do not directly touch the device by hand, and prevent the solder from falling into the device to avoid product failure;

8.3. Design recommendations

A) The pressure sensor is designed in the area away from the stress concentration;;

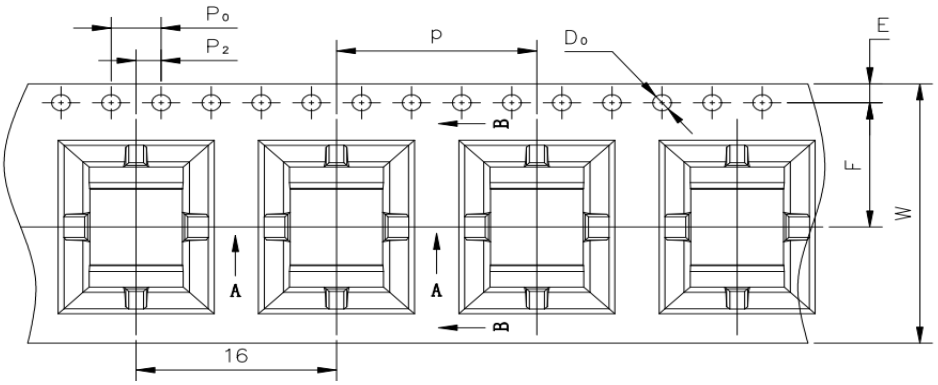
b) The pressure sensor is designed away from the heat source devices (including PCB double sides);

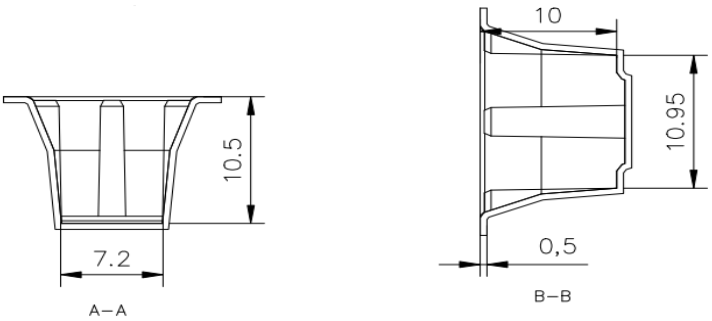
c) The pressure sensor cannot be enclosed by non-breathable material, and can not the inlet or vent can be blocked;

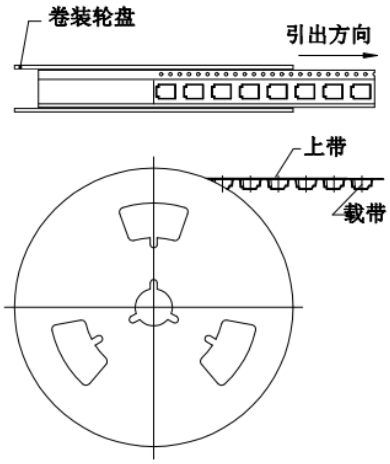
9. Packaging specifications

Load information: unit:mm

Number per volume: 400pcs







330*100*24

12. Attachment: IIC Reference Routine

```
#include "PressureSensor.h"
#include "PressureSensor_IIC.h"

//IIC clock line
sbit SCL = P1 ^ 1;

//IIC data line
sbit SDA = P1 ^ 0;

//Set the input and output mode of IIC data pin #define Set_SDA_INPUT()
    P1MDOUT &= 0xFE;
P1 |= 0X01
#define Set_SDA_OUTPUT() P1MDOUT |= 0x01;

////Delay function needs to be defined
void DelayUs(unsigned char i)
{
}

//Start signal
void Start(void)
{

    SDA = 1;
    DelayUs(2);
    SCL = 1;
    DelayUs(2);
    SDA = 0;
    DelayUs(2);
    SCL = 0;
}

//Stop signal
void Stop(void)
{
    Set_SDA_OUTPUT();
    SDA = 0;
    DelayUs(2);
    SCL = 1;
    DelayUs(2);
    SDA = 1;
    DelayUs(2);
}

//Read ACK signal
unsigned char Check_ACK(void)
{
    unsigned char ack;
    Set_SDA_INPUT();
    SCL = 1;
    DelayUs(2);
    ack = SDA;
```

```
    SCL = 0;
    Set_SDA_OUTPUT();
    return ack;
}

//Send ACK signal
void Send_ACK(void)
{
    Set_SDA_OUTPUT();
    SDA = 0;
    DelayUs(2);
    SCL = 1;
    DelayUs(2);
    SCL = 0;
    DelayUs(2);
}

//Send one byte
void SendByte(unsigned char byte)
{
    unsigned char i = 0;
    Set_SDA_OUTPUT();
    do
    {
        if (byte & 0x80)
        {
            SDA = 1;
        }
        else
        {
            SDA = 0;
        }
        DelayUs(2);
        SCL = 1;
        DelayUs(2);
        byte <<= 1;
        i++;
        SCL = 0;
    } while (i < 8);
    SCL = 0;
}

//Receive one byte
unsigned char ReceiveByte(void)
{
    unsigned char i = 0, tmp = 0;
    Set_SDA_INPUT();
    do
    {
        tmp <<= 1;
        SCL = 1;
        DelayUs(2);
        if (SDA)
        {
```

```
            tmp |= 1;
        }
        SCL = 0;
        DelayUs(2);
        i++;
    } while (i < 8);
    return tmp;
}

//Write a byte of data through IIC
uint8 BSP_IIC_Write(uint8 address, uint8 *buf, uint8 count)
{
    unsigned char timeout, ack;
    address &= 0xFE;
    Start();
    DelayUs(2);
    SendByte(address);
    Set_SDA_INPUT();
    DelayUs(2);
    timeout = 0;
    do
    {
        ack = Check_ACK();
        timeout++;
        if (timeout == 10)
        {
            Stop();
            return 1;
        }
    } while (ack);

    while (count)
    {
        SendByte(*buf);
        Set_SDA_INPUT();
        DelayUs(2);
        timeout = 0;
        do
        {
            ack = Check_ACK();
            timeout++;
            if (timeout == 10)
            {
                return 2;
            }
        } while (0);
        buf++;
        count--;
    }
    Stop();
    return 0;
}
```

```
//Read a byte of data through IIC
uint8 BSP_IIC_Read(uint8 address, uint8 *buf, uint8 count)
{
    unsigned char timeout, ack;
    address |= 0x01;
    Start();
    SendByte(address);
    Set_SDA_INPUT();
    DelayUs(2);
    timeout = 0;

    do
    {
        ack = Check_ACK();
        timeout++;
        if (timeout == 4)
        {
            Stop();
            return 1;
        }
    } while (ack);
    DelayUs(2);

    while (count)
    {
        *buf = ReceiveByte();
        if (count != 1)
            Send_ACK();
        buf++;
        count--;
    }
    Stop();
    return 0;
}

// Define the upper and lower limits of the calibration pressure
#define PMIN -40000    // Zero Point Pressure Value for example -40kPa
#define PMAX 40000 // Full range pressure Value, for example 40kPa
#define DMIN 2516582.4 //AD value corresponding to pressure zero, for example 15%AD
#define DMAX 14260633.6 //AD Value Corresponding to Full Pressure Range, for example 85%AD

//The 7-bit IIC address of the PressureSensor is 0x78
uint8 Device_Address = 0x78 << 1;

//Delay function needs to be defined
void DelayMs(uint8 count)
{
}

//Read the status of IIC and judge whether IIC is busy
uint8 PressureSensor_IsBusy(void)
{
    uint8 status;
    BSP_IIC_Read(Device_Address, &status, 1);
```

```
    status = (status >> 5) & 0x01;
    return status;
}

void PressureSensor_get_cal(void)
{
    uint8 buffer[6] = {0};
    uint32 Dtest = 0;
    uint16 temp_raw = 0;
    double pressure = 0.0, temp = 0.0;

    //Send 0xAC command and read the returned six-byte data
    buffer[0] = 0xAC;
    BSP_IIC_Write(Device_Address, buffer, 1);
    DelayMs(5);
    while (1)
    {
        if (PressureSensor_IsBusy())
        {
            DelayMs(1);
        }
        else
            break;
    }
    BSP_IIC_Read(Device_Address, buffer, 6);

    //The returned pressure and temperature values are converted into actual values according to the calibration
range
    Dtest = ((uint32)buffer[1] << 16) | ((uint16)buffer[2] << 8) | buffer[3];
    temp_raw = ((uint16)buffer[4] << 8) | (buffer[5] << 0);

    //pressure unit: Pa
    pressure = (PMAX-PMIN)/(DMAX-DMIN)*(Dtest-DMIN)+PMIN;

    temp = (double)temp_raw / 65536;

    //temp unit:    0.01℃
    temp = temp * 19000 - 4000;
}
```

地址：东莞市万江区拔蛟窝苏屋工业区
Add: Suwu Industrial Zone,Bajiaowo Village,Wangjiang District,Dongguan, Guangdong
TEL：86+769+22176561 Fax：86+769+22778617
Https://xstsensor.en.alibaba.com E-mail:manager@xstsensor.com