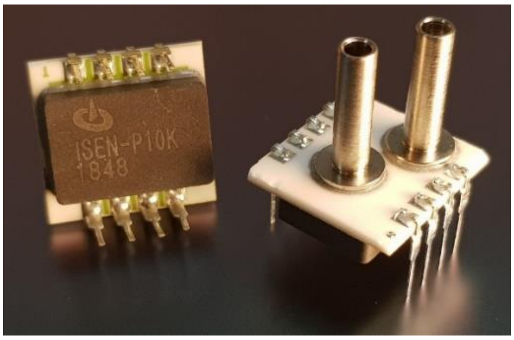
ISEN-P10K

MEMS Piezoresistive Type

Differential Pressure Sensor

1. Features



- Fully Temperature Compensated and Calibrated
- Dual Inline Package (DIP)
- Serial interface control (I²C)

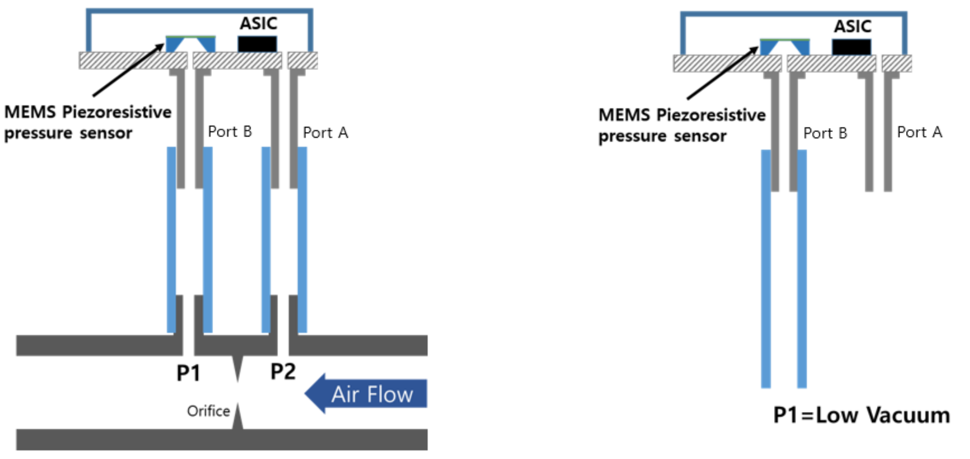
● Summary

The ISEN-P10K is a pressure & temperature sensor. The device includes a piezoresistance type pressure sensor and a band-gap temperature sensor with resistance-to-voltage converter (R-V Converter). This results in superior signal quality, a fast response time and insensitivity to external disturbances at a very competitive price. Each ISEN-P10K is individually calibrated in a precision chamber. The calibration coefficients are programmed into the memory. These coefficients are used internally during measurements to calibrate the signals from the sensors. The serial interface and internal voltage regulation allows easy and fast system integration. Its small size and low power consumption makes it the ultimate choice for even the most demanding applications. The device is supplied in single-in-line type devices. Customer specific packaging options may be available on request.

2. Application

- Industrial Control
- Flow Meter
- Level Detection
- HVAC
- Medical Equipment

ISEN-P10K can be used as a pressure sensor for positive and negative pressure.



(a) (b)

Fig. 1. A differential pressure sensor measuring the airflow and low vacuum(13.33mTorr@ISEN-P10K, Optional ~1.333mTorr)

Note

- Specification of Pressure Sensor Based in "Port A".
- Port A is used for Positive Gauge Pressure Mode.
- Port B is not used for Gauge Pressure Mode.
- Port B is used for Vacuum or Differential Pressure Mode.

3. Specification

Parameter		Min	Typ	Max	units
Pressure Sensor	Measurement Range	0		10000	Pa
	Digital Output SPAN	3000		13000	DEC
	Zero Point Accuracy[1)]		50		Pa
	Span Accuracy[2)]	-100		+100	Pa
		-100	0	+100	DEC
	Operating Temp.	0		+80	℃
	Linearity	2.5	0.1	2.5	% (BFSL) "Port A"
Temp. Sensor (Bandgap)	Digital Output SPAN (-40~125℃)	102		1791	DEC
	Resolution	-	-	11	bit

Note: Total accuracy is a sum of Zero Point Accuracy[1)] and Span Accuracy[2)]

4. Dimension

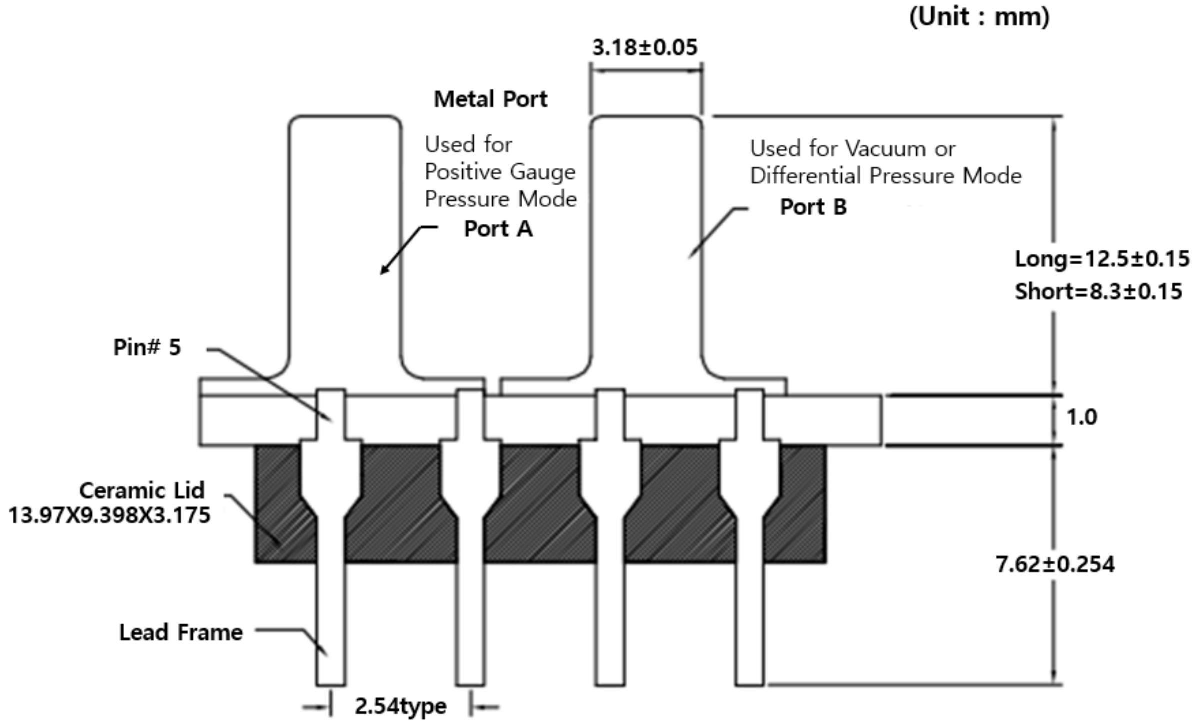
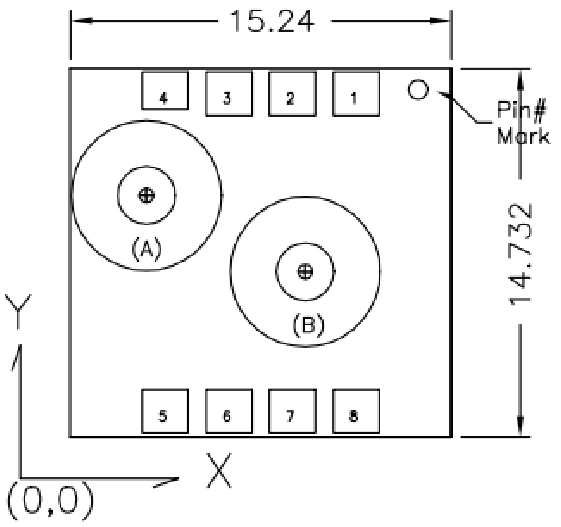


Fig. 2. Side View



Pin	Description
1	VDD
2	Sda
3	Vss (GND)
4	Scl
5	NC
6	NC
7	NC
8	NC

Fig. 3. Top View

5. Operating Condition

1) Absolute Maximum Ratings

Parameter	Symbol	Min	Typ	Max	Units
Analog Supply Voltage	V_{DD}	-0.3		6.0	V
Voltages at Analog I/O In Pin	V_{INA}	-0.3		V_{DD}+0.3	V
Voltages at Analog I/O Out Pin	V_{OUTA}	-0.3		V_{DD}+0.3	V
Storage Temperature Range	T_{STOR}	-55		150	˚C

2) Operating Conditions

<table>
<tr><th colspan="2">Parameter</th><th>Symbol</th><th>Min</th><th>Typ</th><th>Max</th><th>Units</th></tr>
<tr><td colspan="2">Supply Voltage to GND</td><td>V_{SUPPLY}</td><td>2.3</td><td>5</td><td>5.5</td><td>V</td></tr>
<tr><td colspan="2">I^2C Pull-up Resistor</td><td>R_{PU}</td><td>1</td><td>2.2</td><td>10</td><td>kΩ</td></tr>
<tr><td colspan="2">SDA Load Capacitance</td><td>C_{SDA}</td><td></td><td></td><td>0.2</td><td>nF</td></tr>
<tr><td>Current</td><td>Digital output</td><td>I_{sleep}</td><td></td><td>1</td><td>3</td><td>μA</td></tr>
</table>

6. Circuit Interface

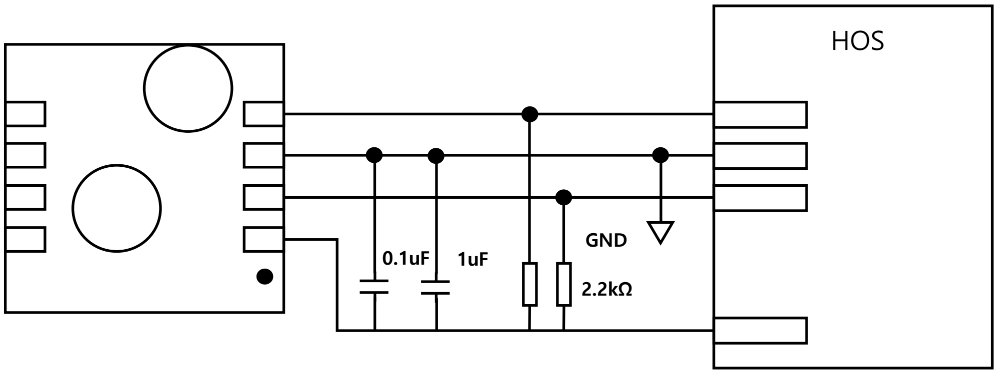


Fig. 4. ISEN-P10K I2C Interface

1) Power

The ISEN-P10K requires a voltage supply between 2.3 and 5.5V. After power on, the device needs 10ms to reach measurement state. No commands should be sent. If you send command at that time, you cannot receive correct data from sensors.

2) Serial Clock(SCL)

The SCLK is used to synchronize the communication between a microcontroller and the ISEN-P10K. Since the interface consists of fully static logic there is no minimum SCL frequency.

3) Serial Data(SDA)

The SDA pin is used to transfer data in and out of the device. Data changes after the falling edge and is valid on the rising edge of the serial clock SCL. During transmission the SDAT line must remain stable while SCL is high. An external pull-up resistor is required to pull the signal high.

7. I²C Interface

For integration with the micro-controller, the ISEN-P10K has a I²C-compatible interface which supports both 100 kHz and 400 kHz bit rate. The I²C slave address is programmed by default on 28_H and can be adjusted in the entire address range of (00_H to $7F_H$).

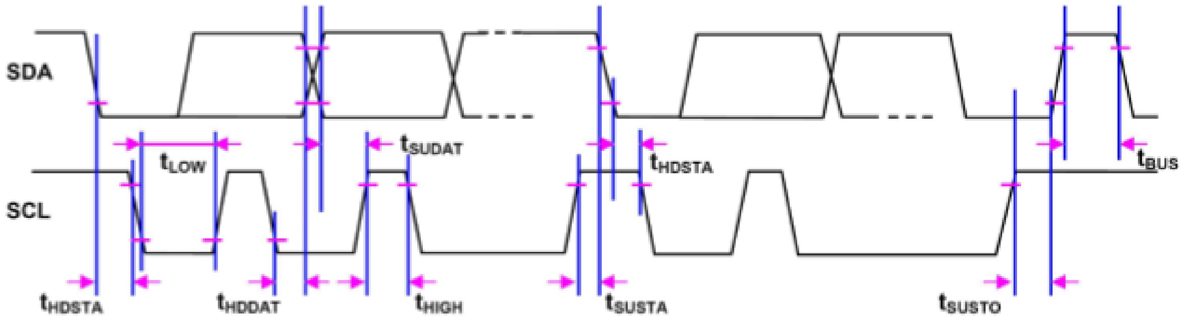


Fig. 5. I²C Timing Diagram

1) I²C Parameters

Parameter	Symbol	Min	Max	Units
SCL clock frequency	f_{SCL}	100	400	kHz
Start condition hold time relative to SCL edge	t_{HDSTA}	0.1		μs
Minimum SCL clock low width	t_{LOW}	0.6		μs
Minimum SCL clock high width	t_{HIGH}	0.6		μs
Start condition setup time relative to SCL edge	t_{SUSTA}	0.1		μs
Data hold time on SDA relative to SCL edge	t_{HDDAT}	0		μs
Data setup time on SDA relative to SCL edge	t_{SUDAT}	0.1		μs
Stop condition setup time on SCL	t_{SUSTO}	0.1		μs
Bus free time between stop condition and start condition	t_{BUS}	1		μs

8. I²C commands

As detailed in below table, there are two types of commands for user operating the ISEN-P10K. The Measurement Request (MR) is wake up command sent by the master for a new measurement cycle.

The Data Fetch (DF) command is used to fetch data in I2C communication. With the start of communication the entire output packet will be loaded in a serial output register. The register will be updated after the communication is finished. The output is always scaled to 14-bits programmed resolution.

I²C command types

Type	Descriptions
Measurement Request(MR)	Start measuring cycle
Data Fetch(DF)	Used to fetch data in any digital mode

Pressure modules do not carry out internal arithmetic operation to minimize on the current consumption. A measurement process is carried out only after the command measuring request (MR) is received.

1) Measurement Requests (MR)

By a measurement request command, the ISEN-P10K is woke up and it executes a measuring cycle. The measuring cycle begins with the temperature measurement, followed by pressure measurement, digital signal processing and finally writing the measured values into the output register.

The MR command consists of the address of the ISEN-P10K, with which the R/W bit is transferred as 0(= write). The real signal of MR is 0x50 adding write bit(0) to address(0x28). After the pressure
module is answered with ACK (= measurement started), the master finalized the transfer with NACK (=stop condition).

Fig. 6. Measurement Request Command

2) Data Fetch(DF)

The DF command is began by the micro-controller (master) as sending address(7bit : 0x28) and read bit (1bit : 1). The real signal is 0x51. The ISEN-P10K sends back an acknowledgement (ACK) to

indicate success. The others data are two status bit and measurement data with pressure and temperature to sending a NACK (= stop condition) by the master. The first two bytes of measurement data contain the two status bits as MSB, and then followed by the pressure value with 14 bits.

If the temperature data is also needed, then these can be read after the pressure value. The most significant 8 bits of the temperature value will be transferred as third byte. Then the least significant 3 bits of the temperature value can be read as the fourth byte. The last five bits are not used and should be masked away. But if the temperature data is not needed, the master can send a NACK signal. If it doesn't appear "state bit 00", does not communicate.

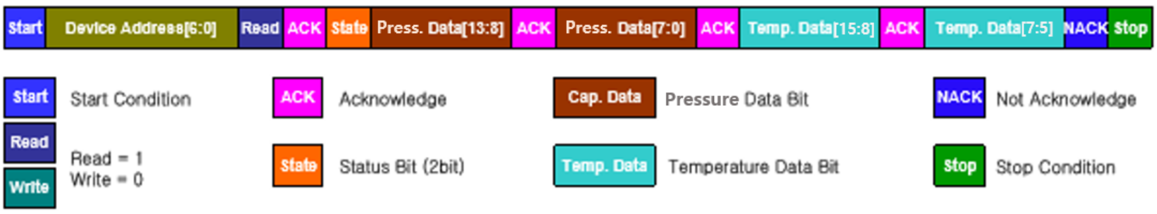


Fig. 7. Measurement Packet Reads

Status Bit Table

Status Bits	Output	Definition
00B	Clipped normal output	Data that has not been fetched since the last measurement cycle.
01B	Not applicable	Data that has already been fetched since the last measurement cycle.
10B	Not used	Not used
11B	Not used	Not used

3) Calculating Pressure and Temperature Output

The entire output of the ISEN_P10K is 4 bytes. The Pressure (in Pa) and the temperature (in degrees Celsius) are calculated with Equation 1 Equation 2 , respectively.

The Pressure_High and The Pressure_Low bytes can be read in Pa

Pressure[Pa] = {(Press. Data[13:8] <<8 + Press. Data[7:0])-3000})/(2^{14}-1) *10000 (Equation 1)

The Temp_High and the Temp_Low bytes can be read as temperature output in ℃.

Temp output[C] = 200*((Temp. Data[15:8] <<3 + Temp. Data[7:5]>>5)/(2^{11}-1)) − 50 (Equation 2)

4) ISEN_P10K I2C driver Code (CoreTex M3 Based)

```
/* P10K.c */

#include "i2c.h"
#include "..₩..₩P10K.h"

#define DEV_ADDRESS ((uint8_t)0x28)

uint8_t data_fetch[4];
float Press,Temp;
uint16_t Press_decimal,Temp_decimal;

// Wake up
void P10K_Measurement_Request(void)
{
        HAL_StatusTypeDef   status;

        status = HAL_I2C_IsDeviceReady(&hi2c1,DEV_ADDRESS<<1,10,1000);
        if(status!=HAL_OK)
        {
            printf("No find i2c address₩n₩r");
        }
}

// Read Pressure, Temperature
void P10K_Data_Fetch(void)
{
        HAL_I2C_Master_Receive(&hi2c1,DEV_ADDRESS<<1|0x01,&data_fetch[0],4,1000);
}

// Calculating Press, Temperature
void P10K_Convert(void)
{
  Press = (float)((Press_decimal-3000)/(16383.0-3000))*10000 ;
  Temp= 200.0 * (float)(Temp_decimal)/2047.0 - 50.0 ;
}

void P10K_Raw(void)
{
  Press_decimal=(((data_fetch[0] & 0x3F) << 8) | data_fetch[1]);
  Temp_decimal=((data_fetch[2] << 3) | (data_fetch[3] >> 5));
}
```

```
/*   P10K-C Press Sensor Header file */

#ifndef __P10K_H
#define __P10K_H

#ifdef __cplusplus
extern "C" {
#endif

/* include file */

#include "i2c.h"
#include "main.h"
#include<stdio.h>


 extern uint8_t data_fetch[4];
 extern float Press,Temp;
 extern uint16_t Press_decimal,Temp_decimal;

 // Wake up
 void P10K_Measurement_Request(void);
 // Read Pressure, temperature
 void P10K_Data_Fetch(void);

 // DATA conversion
 void P10K_Raw(void);
 void P10K_Convert(void);


#ifdef __cplusplus
}
#endif

#endif
```

9. Document Revision History

Revision	Date	Description
1	Jul. 19, 2019	1^{st} Release
2	Oct. 19, 2020	2^{nd} Release: Temperature Calculation Function Revision Application Area Schematic Include
3	Oct. 27, 2020	3^{rd} Release: Added Temperature Sensor Specification