Superb Sensor

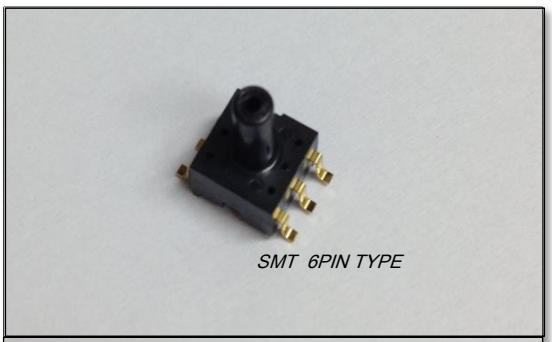
# ASM-3000

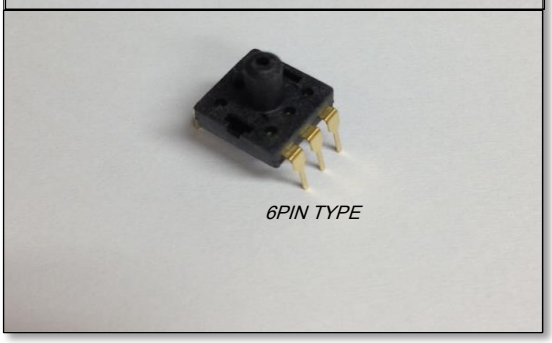
## Pressure sensor

## With I²C Digital Output

- Piezo-resistive silicon micro-machined sensor
- Gauge type pressure sensor
- I²C interface
- Pressure range : -500mbar to +500mbar
- Resolution : 0.5mbar
- Temperature Compensation : 0 ~ 50℃
- Operating voltage 5.0V
- 6PIN type package
- RoHS compliant and Halogen-free

The ASM-3000 is the pressure sensor which measures gauge pressures. It consists of a silicon micro-machined sensing element chip and a signal conditioning ASIC. The pressure sensor element and the ASIC are mounted inside a system-in-package and wire-bonded to appropriate contacts. The ASM-3000 provides the digital output data with the format of I²C interface. It can achieve ESD robustness, fast response time, high accuracy and linearity as well as long-term stability. All measurement data is fully calibrated and temperature compensated. In addition, it allows for easy system integration.

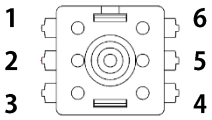

SMT 6PIN TYPE

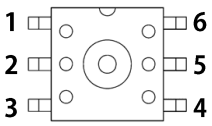

6PIN TYPE

| Specifications | |
|---|---|
| Pressure range | -500mbar to +500mbar |
| Supply Voltage | 5VDC |
| Supply Current | 3.5mA |
| Output Range | 5 to 95% of F.S<br>Can be adjusted to application. |
| Output Signal | I²C |
| Output ripple | 0.2% of F.S |
| Water protection | IP-65 |
| Wetted MAT′ L | PPS |
| Radiated immunity | 10V/m 80~1000MHZ |
| ESD withstand | 2KV |

| Performance & Environmental | |
|---|---|
| Temperature effect | ±0.01% of F.S/℃ |
| Accuracy | ±1.0% |
| Total error band | ±1.0% of F.S(25℃)<br>±1.5% of F.S(0 to 50℃) |
| Linearity | ±0.2% of F.S(0 to 50℃) |
| Cycle life | 500,000 / cycle |
| Operating Temp | -20 to 80℃ |
| Proof pressure<br>Burst pressure | 2 x Full Range<br>3 x Full Range |
| Storage Temp | -30 to100℃ |
| Humidity Limits | 0 to 95% RH |

## ❒ PIN Description

| Description | No. | Name | I/O | Function description | I/O Characteristics |
|---|---|---|---|---|---|
|  DIP TYPE (Top View) | 1 | SCL | I | Serial Data Input | Cin < 10pF |
| | 2 | SDA | I/O | Serial Data Input / Output (Open Drain Output) | Input Mode : Cin < 10pF<br>Output Mode : RON < 60Ω |
| | 3 | GND | - | Connected to GND | |
| | 4 | TEST | - | Connected to VDD | |
| | 5 | VDD | - | Connected to VDD | |
| | 6 | SYSE | I | Connected to GND (Continuous output Mode) | Cin < 10pF |

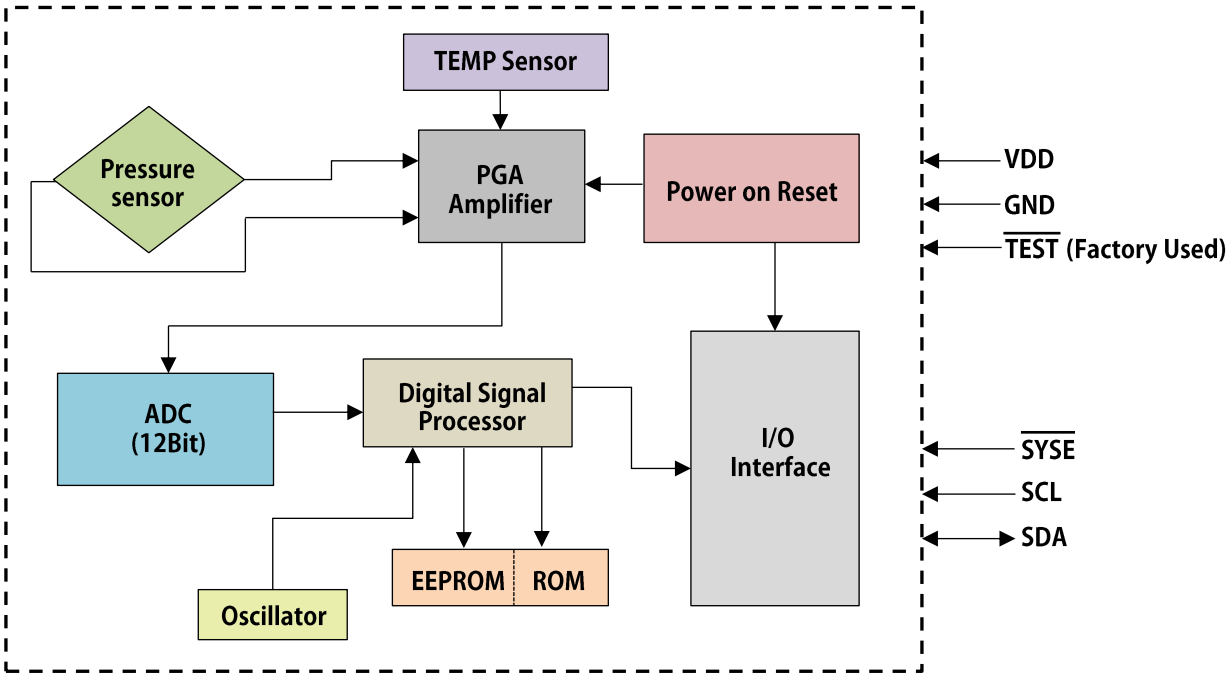
| Description | No. | Name | I/O | Function description | I/O Characteristics |
|---|---|---|---|---|---|
|  SMT TYPE (TOP VIEW) | 1 | SCL | I | Serial Data Input | Cin < 10pF |
| | 2 | SDA | I/O | Serial Data Input / Output (Open Drain Output) | Input Mode : Cin < 10pF<br>Output Mode : RON < 60Ω |
| | 3 | GND | - | Connected to GND | |
| | 4 | TEST | - | Connected to VDD | |
| | 5 | VDD | - | Connected to VDD | |
| | 6 | SYSE | I | Connected to GND (Continuous output Mode) | Cin < 10pF |

## ❒ Electrical Characteristics [VDD = 5.0V]

| Parameter | VDD | Conditions | Min. | Typ. | Max. | Unit | Symbol |
|---|---|---|---|---|---|---|---|
| Operating Voltage | - | - | 4.5 | 5.0 | 5.5 | V | $V_{DD}$ |
| Operating Current | 5.0V | - | - | 1.7 | 2.0 | mA | $I_{DD}$ |
| Standby Current | 5.0V | System standby | - | 0.1 | 1 | μA | $I_{STB1}$ |
| Compensation Temp | - | - | 0 | - | +50 | ℃ | $T_{CMP}$ |
| Resolution | - | - | - | 0.5 | - | mbar | R |
| Human Body Mode | 5.0V | MIL-STD-883E Method 3015.7 | - | 2 | - | KV | ESD |
| Pressure Accuracy | 5.0V | T : 25℃<br>P : -500 to 500mbar | -1.0 | - | +1.0 | % | $P_{ACC}$ |
| | | T : 0 to 50℃<br>P : -500 to 500mbar | -1.5 | - | +1.5 | % | |

## ❒ Main ASIC Part Diagram

**Diagram**

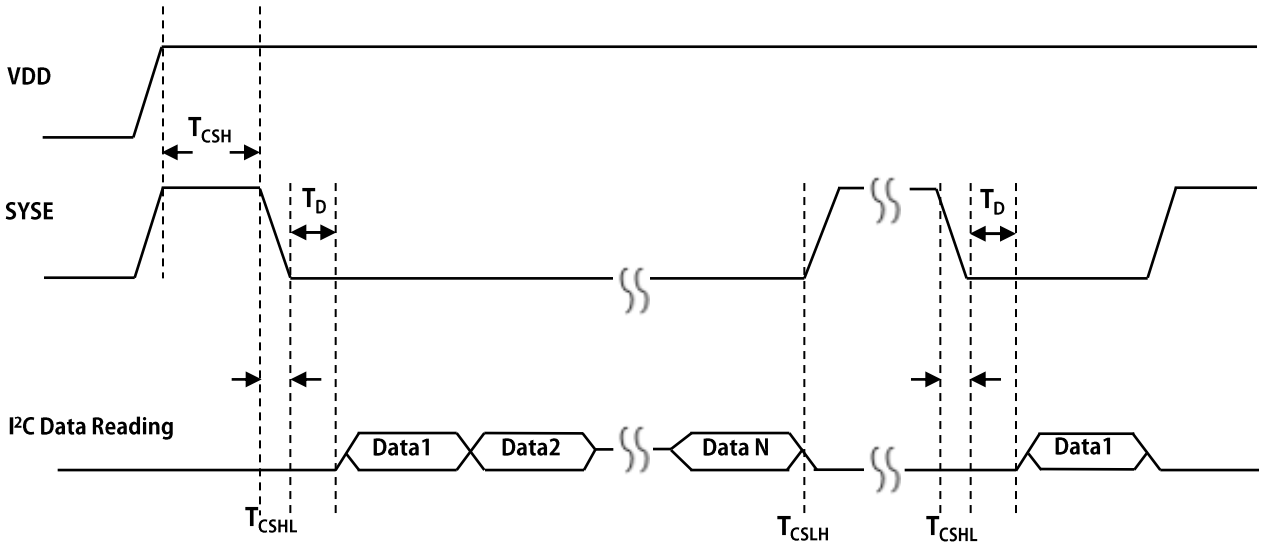



| No | Pin Name | Descriptions | Notice |
|---|---|---|---|
| 1 | Pressure sensor | The semiconductor based pressure sensor makes -50 to -150mV of output signal by changing of water level. | * Setting : 120mV |
| 2 | (PGA)Amplifier | The program amplifier in the ASIC amplify the sensor signal to over specified voltage, or to calibrate the off set. | - |
| 3 | TEMP Sensor | The temperature sensor thermistor in the ASIC compensates the ambient temperature changing from -40 to 120℃ of temperature range. | - |
| 4 | Power on Reset (POR) | The power regulator as 5VDC in the ASIC include "Auto Reset" and noise protection circuit. The POR level depends on the rise time of VDD. | - |
| 5 | ADC(12Bit) | To amplify and convert the analog signal of sensor to 12 bit of digital signal for calculating. | - |
| 6 | Digital Signal Processor | The core part to drive the ASIC store the data into EEPROM due to program of ROM for all of operating function. | - |
| 7 | I/O Interface | The gate function connected between ASIC and outside constitutes I/O logic circuit for communication of data and clock information.<br>* Maximum of inside allowable current of I/O : 20mA<br>* Inside oscillation frequency of I/O : NA | - |
| 8 | Oscillator | The clock oscillates inside of ASIC provides constitution of all digital communication without other additional oscillator outside of ASIC. | * Frequency of oscillator : 200KHz |
| 9 | EEPROM/ROM | The control function of ASIC is saved into ROM, and the algorism of sensor data and temperature compensation information are stored into EEPROM. | * EEPROM size : 120Byte<br>* ROM size : 1KByte |

## ❒ System Reading Timing

chart

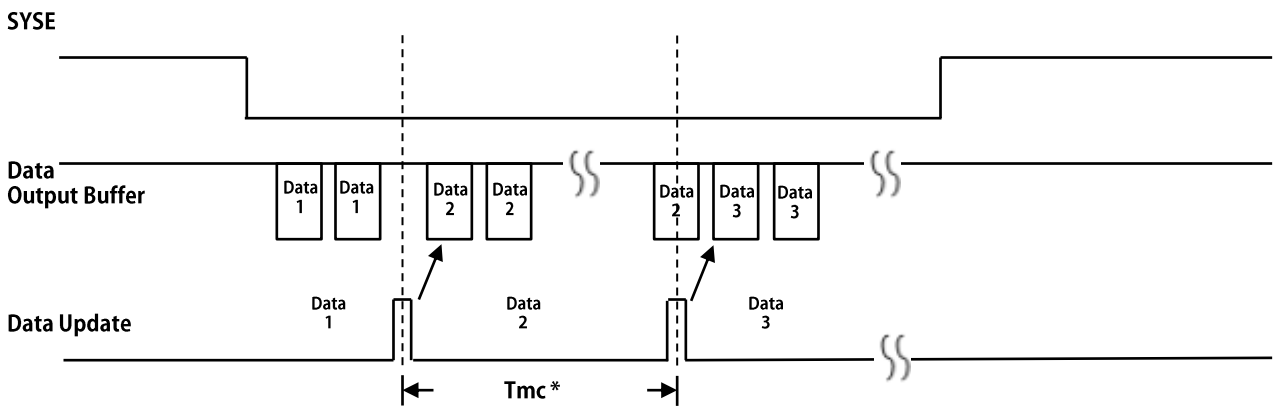
Note : $T_D$ is required for sensor to get thermo stabilized after being activated.
Data retrieved within $T_D$ period might be incorrect.




## ❒ Data Update Timing

chart

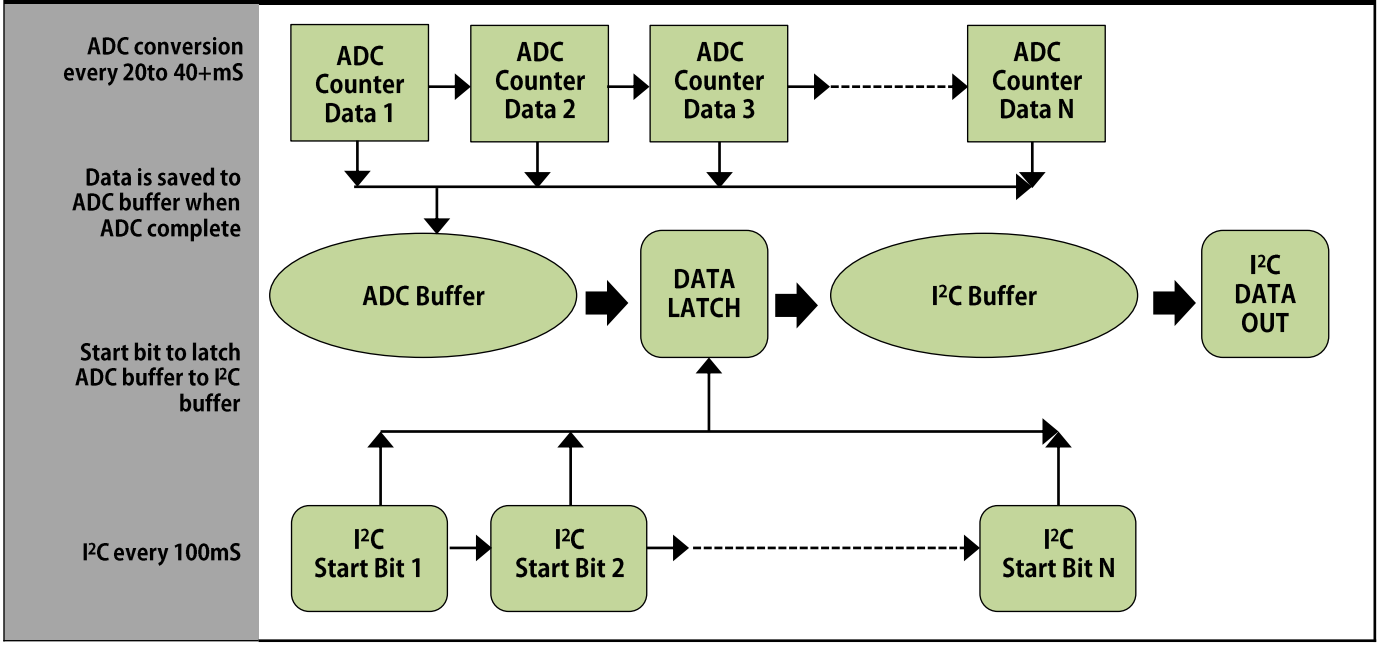
Note : $T_D$ is required for sensor to get thermo stabilized after being activated.
Data retrieved within $T_D$ period might be incorrect.




| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{CSH}$ | CS High Hold Time | 100 | - | - | uS |
| $T_{CSHL}$ | CS High To Low Time | - | - | 1 | mS |
| $T_{CSLH}$ | CS Low To High Time | - | - | 1 | mS |
| $T_D$ | Chip Enable Delay Time | 400 | - | - | mS |
| Tmc * | Data Update Time | 20 | - | 40 | mS |

## ❒ ADC Buffer and I²C Buffer Data Transfer Logic and sequence illustration

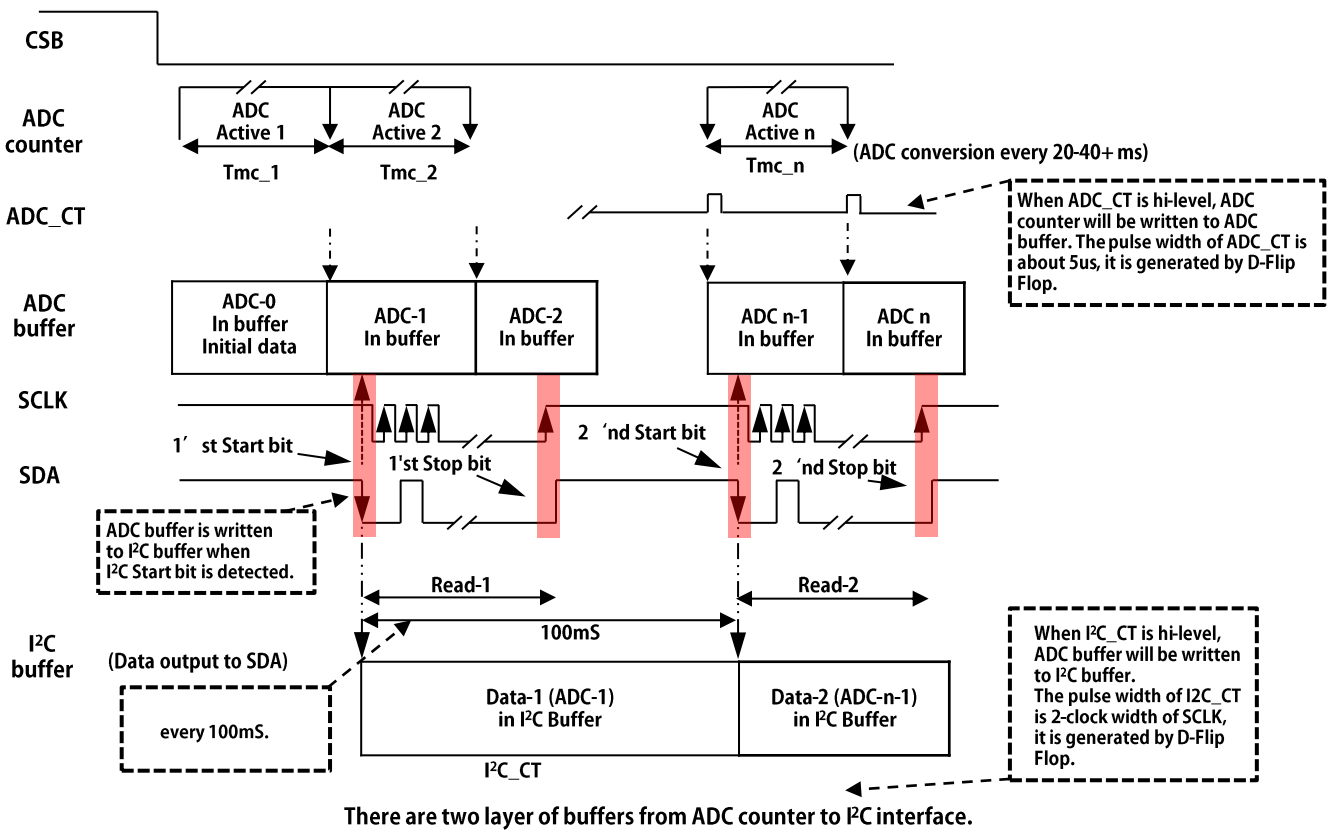
**Diagram**




## ❒ The data load Timing of Pressure ASIC

**Diagram**

Note：Tmc_1, Tmc_2, ,Tmc_n-1,Tmc_n are ADC conversion time.(ADC conversion every 20 to 40+ms)Tmc is not fixed value, it is variable with measure pressure, frequency of ASIC etc.

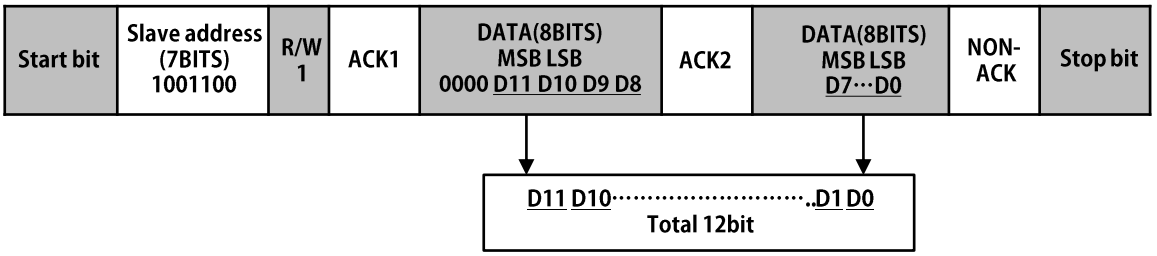



There are two layer of buffers from ADC counter to I²C interface.

## ❒ ASIC I²C Interface

**Chart**

The data transfer format of I²C (ADC with 12 bits resolution) Master read(Master—Receiver;Slave—Transmitter)




Each sensor is referenced on the bus by a 7 bit slave address. The slave address is 1001100. The eighth bit of control address is read or written which assigned by processor.

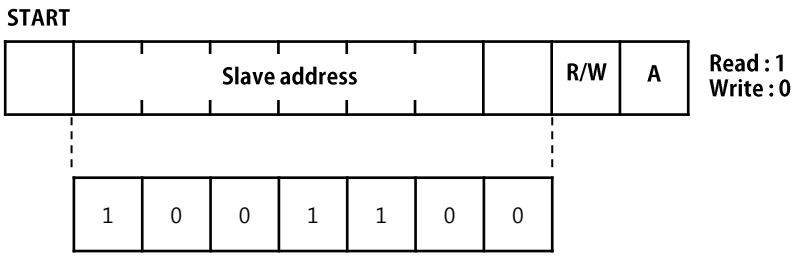


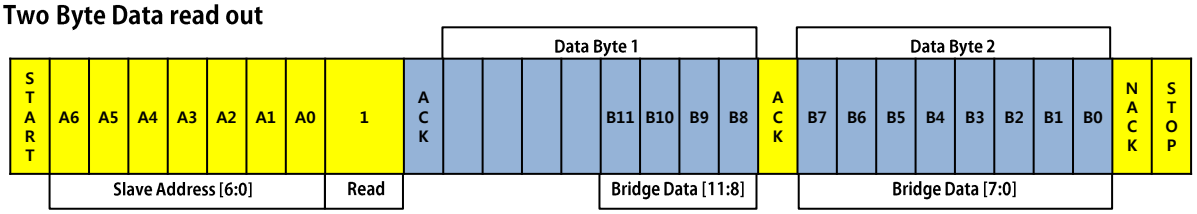

Figure 1. Control address byte

## ❒ I²C Pressure Measurement Packets

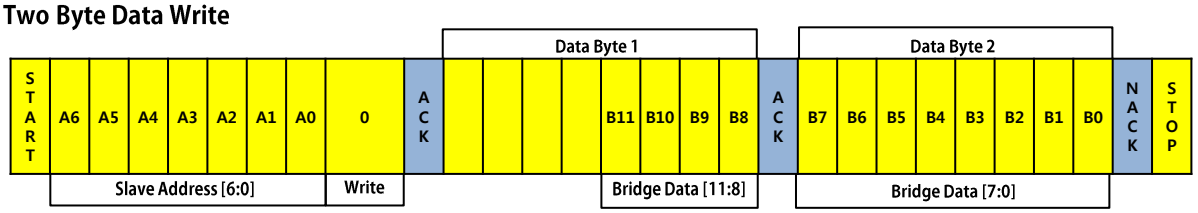
**Chart**

- ❖ ACK1 : That should respond a ACK1 (Low) signal to the microcontroller.
- ❖ ACK2 : The microcontroller should respond a ACK2 (Low) signal to the sensor.
- ❖ NON ACK : The microcontroller should respond a NON-ACK (High) signal the sensor.
- ❖ I²C Reading Code Example : If needed, you can refer to example in this document.

### Two Byte Data read out




### Two Byte Data Write




From master to slave  From slave to master

## ❒ Bus protocol

**chart**

❖ Data transfer may be initiated only when the bus is not busy.
❖ During data transfer, the data line must keep stable whenever the clock is HIGH level. Changes in the data line while the clock line is HIGH will be interpreted as a start or stop condition.
❖ Following bus conditions has been defined as Figure 2.

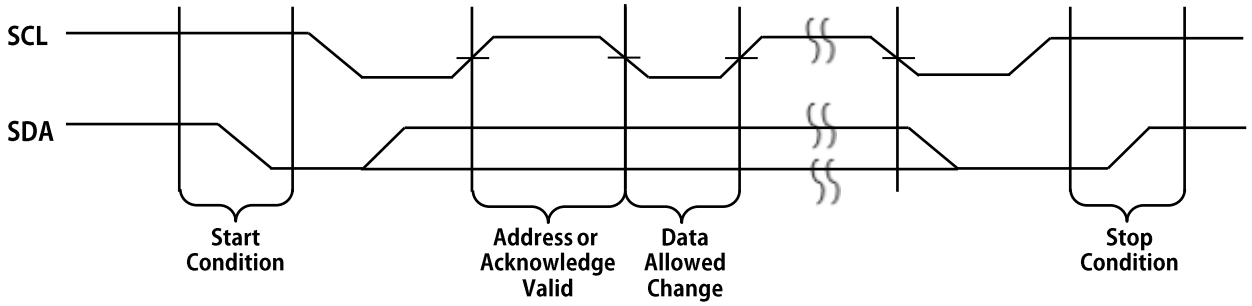
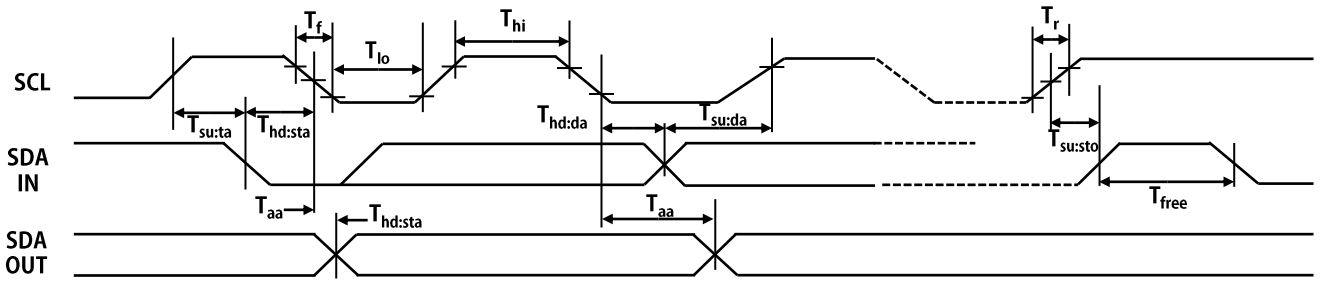



Figure 2. Data transfer sequence on serial bus

## ❒ The data bus timing

**chart**




When the bus is free, both lines are pulled up to +ADD. Data on the I²C-bus can be transferred at a rate up to 100 kbit/s in the standard-mode, or up to 400 kbit/s in the fast-mode.

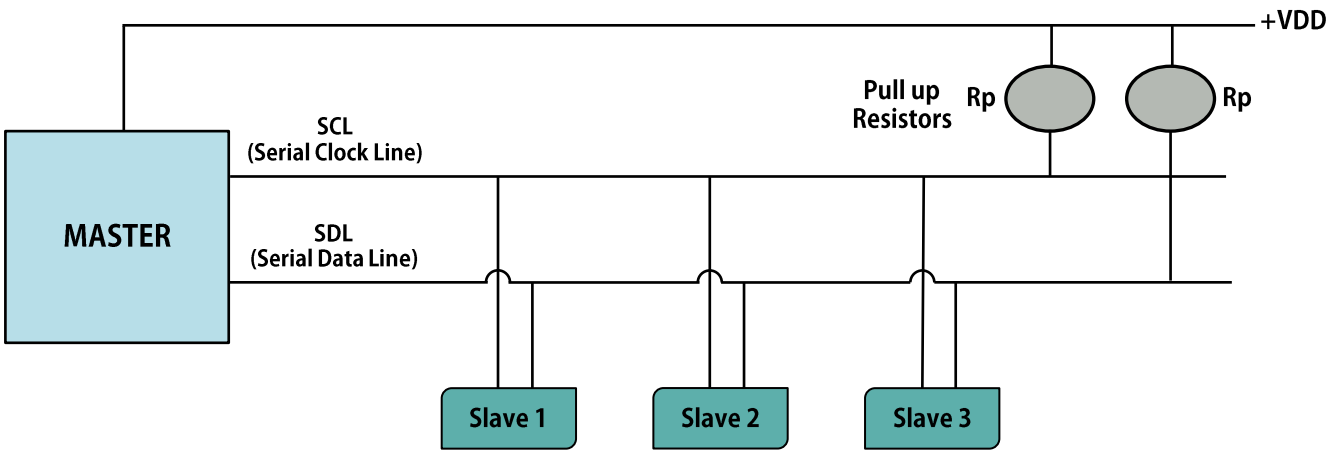



Figure 3. I²C Bus Configuration

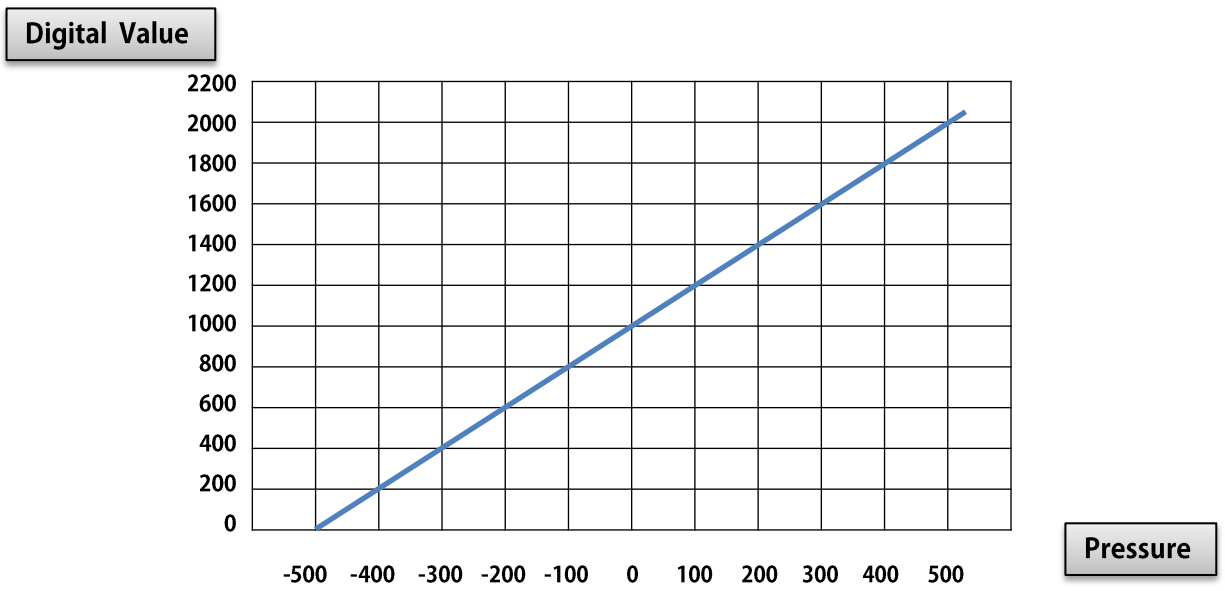
## ❒ A.C. Characteristic

| Symbol | Parameter | Standard mode | | Fast mode | | | Unit |
|---|---|---|---|---|---|---|---|
| | | MIN | MAX | MIN | Typical | MAX | |
| $F_{scl}$ | Clock frequency | - | 100k | - | 400k | 2M | Hz |
| $T_{hi}$ | Clock high time | 4,000 | - | 600 | - | - | nS |
| $T_{lo}$ | Clock low time | 4,700 | - | 1,300 | - | - | nS |
| $T_r$ | SCL and SDA rise time | - | 1,000 | - | 300 | - | nS |
| $T_f$ | SCL and SDA fall time | - | 300 | - | 300 | - | nS |
| $T_{hd:sta}$ | START condition hold time | 4,000 | - | 600 | - | - | nS |
| $T_{su:sta}$ | START condition setup time | 4,700 | - | 600 | - | - | nS |
| $T_{hd:da}$ | DATA input hold time | 0 | - | - | 0 | - | nS |
| $T_{su:da}$ | DATA input setup time | 250 | - | 100 | - | - | nS |
| $T_{su:sto}$ | STOP condition setup time | 4,000 | - | 600 | - | - | nS |
| $T_{aa}$ | Output valid from clock | - | 3,500 | - | 900 | - | nS |
| $T_{free}$ | Bus free time | 4,700 | - | 1,300 | - | - | nS |

## ❒ Pressure versus Digital output value (Typical)

chart

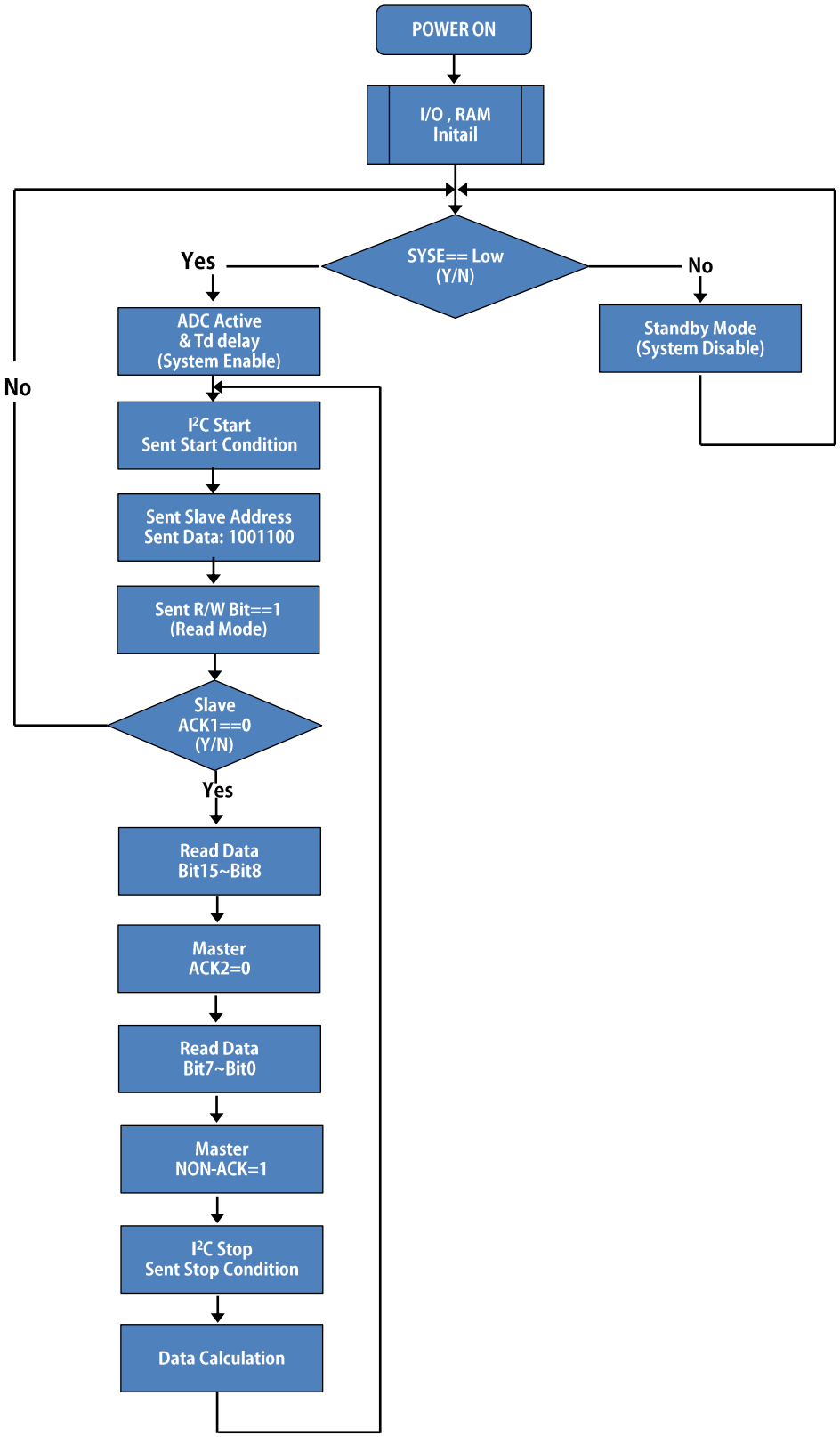
The relationship between digital output value and pressure is given as show below




Pressure(mbar) = (Output value -1000) / 2

## ❒ Reading Flow (I²C Interface)

**Chart**




| Bit No. | Description | Note |
|---|---|---|
| Bit 15 | ADC Data Bit | Normal = 0 |
| Bit 14 | ADC Data Bit | Normal = 0 |
| Bit 13 | ADC Data Bit | Normal = 0 |
| Bit 12 | ADC Data Bit | Normal = 0 |
| Bit 11 | ADC Data Bit | |
| Bit 10 | ADC Data Bit | |
| Bit 9 | ADC Data Bit | |
| Bit 8 | ADC Data Bit | |
| Bit 7 | ADC Data Bit | |
| Bit 6 | ADC Data Bit | |
| Bit 5 | ADC Data Bit | |
| Bit 4 | ADC Data Bit | |
| Bit 3 | ADC Data Bit | |
| Bit 2 | ADC Data Bit | |
| Bit 1 | ADC Data Bit | |
| Bit 0 | ADC Data Bit | |

**Notice:**

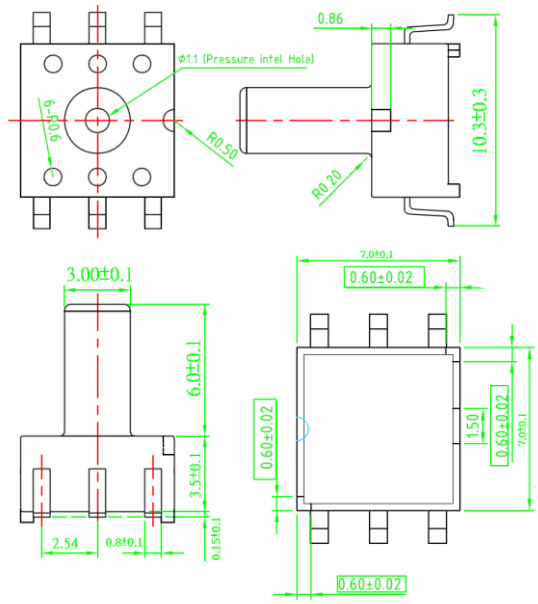
❖ ACK1: The main sensor part should respond a ACK1 (Low) signal to the microcontroller.

❖ ACK2: The microcontroller should respond a ACK2 (Low) signal to the main sensor part.

❖ NON ACK: The microcontroller should respond a NON-ACK (High) signal to the main sensor part .

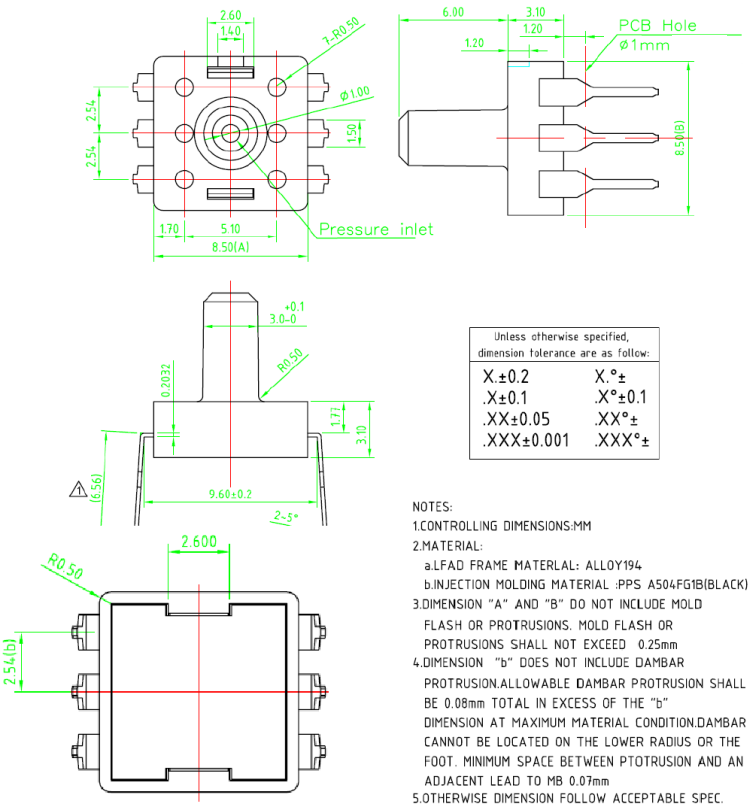
## Outline Drawing Dimension (mm)

### ❖ SMT TYPE




## Outline Drawing Dimension (mm)

### ❖ DIP TYPE




| Unless otherwise specified, dimension tolerance are as follow: | |
|---|---|
| X.±0.2 | X.°± |
| .X±0.1 | .X°±0.1 |
| .XX±0.05 | .XX°± |
| .XXX±0.001 | .XXX°± |

NOTES:
1.CONTROLLING DIMENSIONS:MM
2.MATERIAL:
a.LFAD FRAME MATERLAL: ALLOY194
b.INJECTION MOLDING MATERIAL :PPS A504FG1B(BLACK)
3.DIMENSION "A" AND "B" DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED 0.25mm
4.DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION.ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08mm TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION.DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PTOTRUSION AND AN ADJACENT LEAD TO MB 0.07mm
5.OTHERWISE DIMENSION FOLLOW ACCEPTABLE SPEC.

## ❒ I²C Reading Code Example (One Shot Mode)

**Code Example**

```
Void clock()
{
SCL=1;
_delay(100);
SCL=0;
_delay(100);
}
void start()
{
SDAC=0; //SET SDAC TO Output
SDA=1; //SDA Output High
SCL=1; //SCL Output High
SDA=0; //SDA Output Low
_delay(100);
SCL=0;
_delay(100);
}
void stop()
{
SDAC=0; //SET SDAC TO Output
SDA=0; //SDA Output Low
SCL=0;
_delay(100);
SCL=1;
_delay(100);
SDA=1;
_delay(100);
}

void NACK()
{
SDAC=0; //SET SDAC TO Output
SDA=1;
_delay(100);
clock();
_delay(100);
}
void MACK()
{
SDAC=0; // SDA: SET OUTPUT
SDA=0; // SDA OUTPUT LOW
_delay(100);
clock();
}
void SACK()
{
uchar Time_out=0; // clear time_out counter
SDAC=1; //SET SDAC TO Input
_delay(100);
SCL=1;
_delay(50); // Delay Time 50 us//
if (SDA) {Time_out+=1; _delay(50);} // SDA CHECK 1
if (SDA) {Time_out+=1; _delay(50);} // SDA CHECK 2
if (SDA) {Time_out+=1;} // SDA CHECK 3
/*If No_ACK_FLAG=1 and Time_out 3 times: means that the system disable.*/
if (Time_out==3) { No_ACK_FLAG=1 ;}
if (SDA==0) { No_ACK_FLAG=0 ;} /* If SDA=0, No_ACK_FLAG=0 (System Enable)*/
clock();
}
```

## ❒ I²C Reading Code Example (One Shot Mode)

**Code Example**

```
void I2C_READ()
{
//===============START CONDITION=====================
start(); // I2C start condition
//=======Write Address bit: Set 1001100 ================================
SDA=1;clock();
SDA=0;clock();
SDA=0;clock();
SDA=1;clock();
SDA=1;clock();
SDA=0;clock();
SDA=0;clock();
//=======Write R/W Bit: set 1=====================================
SDA=1;clock();
//=======SLAVE ACK LOW ============================
SACK();
//=======Read Data (Bit 8 to Bit 15) ==============================
counts=0;
if (SDA==1) counts+=32768; clock(); // Reading Bit 15
if (SDA==1) counts+=16384; clock(); // Reading Bit 14
if (SDA==1) counts+=8192; clock(); // Reading Bit 13
if (SDA==1) counts+=4096; clock(); // Reading Bit 12
if (SDA==1) counts+=2048; clock(); // Reading Bit 11
if (SDA==1) counts+=1024; clock(); // Reading Bit 10
if (SDA==1) counts+=512; clock(); // Reading Bit 9
if (SDA==1) counts+=256; clock(); // Reading Bit 8
//=======MASTER: SENT ACK2 (HIGH) ==================
MACK();

//=======Read Data (LOW BYTE) ==================
SDAC=1; // SDA: SET INPUT
_delay(100);
if (SDA==1) counts+=128; clock(); // Reading Bit 7
if (SDA==1) counts+=64; clock(); // Reading Bit 6
if (SDA==1) counts+=32; clock(); // Reading Bit 5
if (SDA==1) counts+=16; clock(); // Reading Bit 4
if (SDA==1) counts+=8; clock(); // Reading Bit 3
if (SDA==1) counts+=4; clock(); // Reading Bit 2
if (SDA==1) counts+=2; clock(); // Reading Bit 1
if (SDA==1) counts+=1; clock(); // Reading Bit 0
//=========NON ACK==============================
NACK();
//=========STOP CONDITION=====================
stop();
//=========check No_ACK_FLAG AND sensor_fail_flag==========//
//===== if Bit-15:0 = 0xffff , No_ACK_FLAG=1: System Disable.
if (counts == 0xffff)
{
No_ACK_FLAG=1;
dsp_NO_ACK();
lcm_dsp_output(counts);
}
//====== Pout(mBar) or POUT(COUNTS) ================//
dsp_txt_mBar(); // display mBar unit
if(counts >= 1000)
{ pout_mbar=(counts-1000)*10/2; lcm_dsp_output(pout_mbar) ;}
else { pout_mbar=(1000-counts)*10/2; NP_flag=0x01; lcm_dsp_output(pout_mbar) ;}
//======== Delay Time ======//
_delay(200000); //delay 200ms
_delay(50000); //delay 50ms
NP_flag=0x00;
}
```

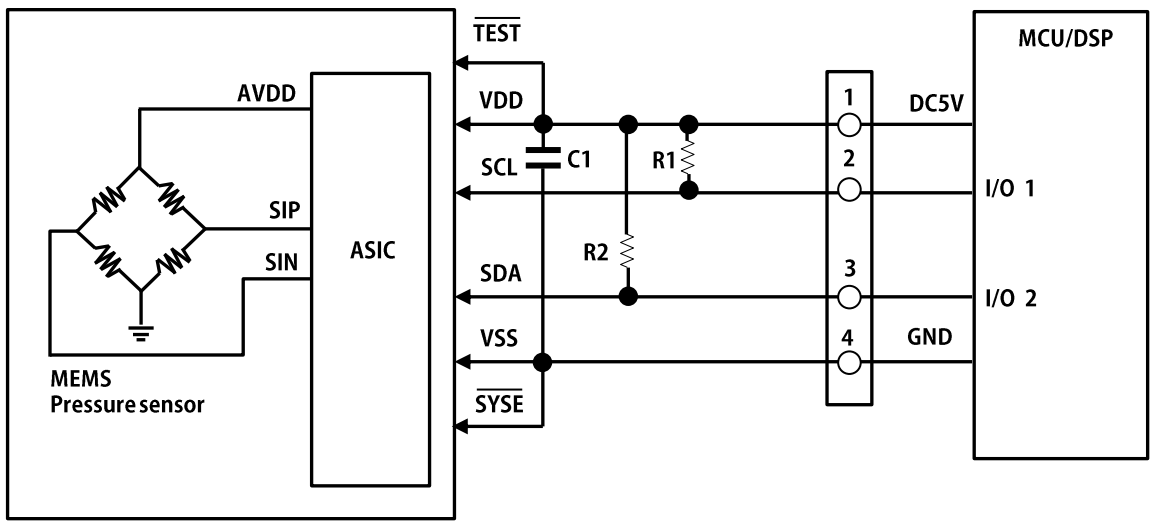
## ❒ I²C Reading Code Example (One Shot Mode)

**Code Example**

```
//* MAIN PRORGAM *//
void main()
{
status_init(); // I/O Initial
//========LCM Dispplay (Unit Option) ============///
lcm_init(); // LCM Initial
dsp_txt_mbar();
No_ACK_FLAG=0x00; // If No_ACK_FLAG=1: system disable .
sensor_fail_flag=0x00; // If sensor_fail_flag=1: means that the sensor fail.
//======== system enable and TD delay ====== //
// SYSE connected to GND: System Enable
_delay(200000); //delay 200ms
_delay(200000); //delay 200ms
//======== Data Reading Loop (Continuous Mode, I2C Interface) ====== //
while(1)
{ I2C_READ(); // Reading Pressure Value //// }
}
```

## ❒ Application Circuit

**Circuit**




**Notice:**
❖ R1 , R2 : Pull-Up Resister (1~10kΩ / If needed)
❖ C1 : 4.7uF