MEMS Pressure Sensor Preliminary Datasheet

1. General Introduction

M0110 is a perfect silicon pressure sensor offering a radiometric analog interface for reading pressure over the specified full scale pressure span. M0110 incorporates a silicon piezoresistive pressure sensor die and an interior Application Specific Integrated Circuit(ASIC) in a SOP package The M0110 is fully calibrated and temperature compensated for specified span, so pressure sensor satisfy the perfect accuracy, which is utilizing a microcontroller or microprocessor with A/D inputs. Pressure sensor is for high volume application at an affordable cost and perfect performance. Customized calibration parameter (pressure range, working voltage, output etc.) are available.

2. Features

- Calibrated Digital Signal(I2C Interface)
- Optional 2.5V～5.5V Power Supply
- Temp. Compensated: 0℃～+60℃(32℉～+140℉)
- MEMS process technology
- Operation range: -1 bar ~ 10 bar
- Gauge type sensor with ASIC

3. Applications

- Blood pressure measurement
- Industrial control
- Pressure gauge

4. Specifications(MEMS Sensor)

Item		Data	Unit	Remark
Available Pressure Range[1]		-100…-0.5～0～0.5…1000	kPa	Customization acceptable
Power Supply Range[2]		2.5~5.5	Vdc	
Max. Excitation Current		3	mA	
ADC Resolution[3]		24	Bit	
SDA/SCL pull up resistor		4.7	Kohm	
ESD HBM		4000	V	
Total Accuracy[4]	10kPa＜Pressure ≤200kPa	±2	%Span	Customization acceptable
	Pressure ≤10kPa or＞200kPa	±2.5		
Long Term Stability[5]		±0.5	%Span	
Over Pressure[6]	Pressure ≤5kPa	5X	Rated Pressure	Differ on specific pressure range
	5kPa＜Pressure ≤200kPa	2.5X	Rated Pressure	
	200kPa＜Pressure	1.5X	Rated Pressure	
Burst Pressure[7]	Pressure ≤5kPa	10X	Rated Pressure	
	5kPa＜Pressure ≤200kPa	3X	Rated Pressure	
	200kPa＜Pressure	2X	Rated Pressure	
Compensation Temp.[8]		0 ～ 60/32 ～ 140	℃/℉	Customization acceptable
Operating Temp.[9]		-20 ～ 100/-4 ～ 212	℃/℉	
Storage Temp.		-30 ～ 125/-22 ～ 257	℃/℉	
Response Time[10]		2.5	mS	

1 Pressure Range(Operating pressure): The available pressure range including various span, not a specific pressure range.

2 Power supply: The default test voltage value: 3.3V, available working power supply voltage range:2.5 ~ 5.5V

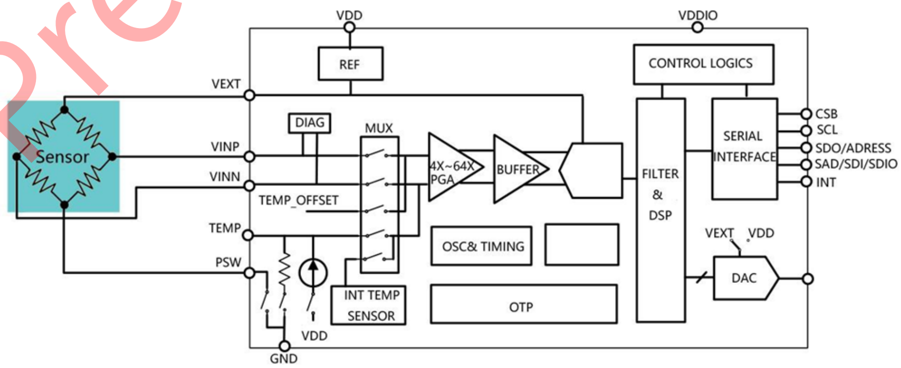
3 ADC Resolution: The ADC resolution is defined as the smallest incremental voltage that can be recognized and thus causes a change in the digital output.

4 Total Accuracy: The max. deviation in output from ideal transfer function at any pressure or temperature over the specified ranges, units are in percent of full scale span (%FSS), which mainly consists of: Offset and Span Shift; Linearity(Non-linearity); Repeatability; Pressure Hysteresis ; TcOffset and TcSpan.
4.1. The accuracy in table is the typical output accuracy. The accuracy is not identical according to different specified pressure range. Contact factory for more information or for higher accuracy requirement(e.g ±1%Span) if need.
4.2 Non-linearity(Linearity): the deviation of measured output from "Best Straight Line" through three points (Offset pressure, FS pressure and ½ FS pressure)at constant temperature.
4.3 Repeatability: the deviation of measured output when the same pressure is applied continuously, with pressure approaching from the same direction within the specified operating pressure range, under the same operating conditions.
4.4 Pressure Hysteresis: the deviation of measured output at any pressure within the specified range, when this pressure is applied continuously, with pressure approaching from opposite directions within the specified operating pressure range, under the same operating conditions.
4.5 TcOffset (TCO:Temp. Coefficient of Offset): the deviation of measured output with minimum rated pressure applied, over the temperature range of 0° to 60°C, relative to 25°C.
4.6 TcSpan (TCS:Temp. Coefficient of Span): the deviation of measured output over the temperature range of 0° to 60°C, relative to 25°C.
5. Long Term Stability: the sensor's output deviation when subjected to 1000 hours pressure test.
6. Over Pressure (Proof pressure): the maximum pressure which may be applied without causing durable shifts of the electrical parameters of the sensing element and remain the specification once pressure is returned to the operating pressure range.
7. Burst Pressure: the maximum pressure which may be applied without causing damage to the sensing die or leaks; The sensor should not be expected to recover function after exposure to any pressure beyond the burst pressure.
8. Compensated Temperature: the temperature range over which the sensor have an output proportional to pressure within the specified performance limits.
9. Operating Temperature (or Ambient Temperature): the temperature range over which the sensor have an output proportional to pressure but may not remain within the specified performance limits.
10. Response Time: it is defined as the time for the incremental change in the output from 10% to 90% of of its final value when subjected to a specified step change in pressure.

4. Specifications(Interior ASIC)

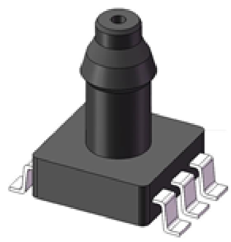
Parameter	Min.	Typ.	Max.	Unit	Notes
LDO Output	1.62	1.8	1.98	V	'regulator_sel'= 0
	3.24	3.6	3.96	V	'regulator_sel'= 1
PSRR		60		dB	
Input Common Mode Rejection Ratio	80	110			
Interior Temp. Sensor Accuracy			±0.5	℃	@25℃
			±1	℃	-40 to 85 ℃
Temp.Output Data Resolution	16			Bit	LSB = (1/256) ℃
I2C Clock frequency			400	khz	I2C interface

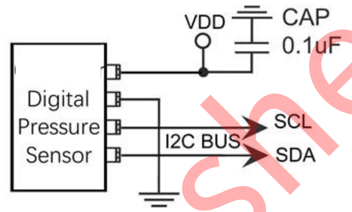
5 · Schematic Diagram & Pin Out



1	2	3	4	5	6
NC	VDD	NC	SDA	SCL	GND

NAME	FUNCTION
NC	Do not connect to external circuitry or ground
VDD	Voltage supply
SDA	Data signal (Send & Receive)
SCL	The clock signal
GND	Ground





Notes:

1. **Implement ESD protection during whole soldering and assembly process.**
2. **Overload voltage(max.6.5Vdc) or current(max.5mA) may burn the ASIC and cause the sensor fail throughly.**
3. **More detalis about soldering and storage etc., refer to overall note.**

6. Customer Ordering Information

Ordering Code
1000kPa = 10D001MPGPN

Date Code Naming Rule
Example : 38A
A=2021 B=2022 C=2023…
38 weeks = 38

7. I2C Interface

I2C bus uses SCL and SDA as signal lines. Both lines are connected to VDD externally via pull-up resistors(Typ value:4.7kΩ) so that they are pulled high when the bus is free. I2C device factory setting slave address: 0X6D.

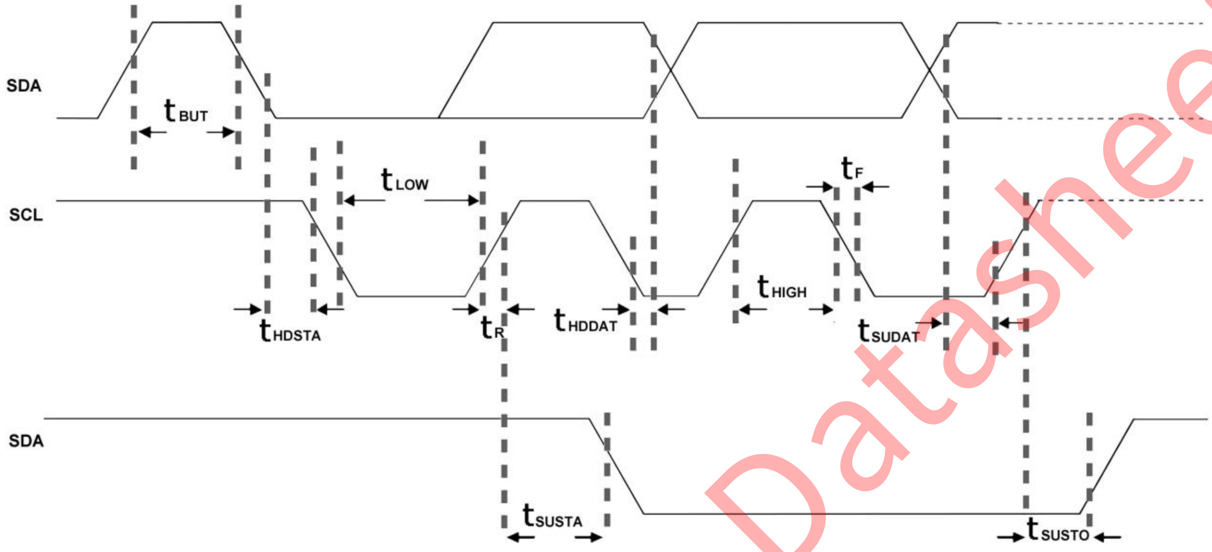
The master device can communicate with the product using commands in the following format:

8. ELECTRICAL SPEC. OF I2C INTERFACE PIN

Symbol	Parameter	Condition	Min	Max	Unit
f_{scl}	Clock frequency			400	KHz
t_{LOW}	SCL low pulse		1.3		us
t_{HIGH}	SCL high pulse		0.6		us
t_{SUDAT}	SDA setup time		0.1		us
t_{HDDAT}	SDA hold time		0.0		us
t_{SUSTA}	Setup Time for a repeated start		0.6		us
t_{HDSTA}	Hold time for a start condition		0.6		us

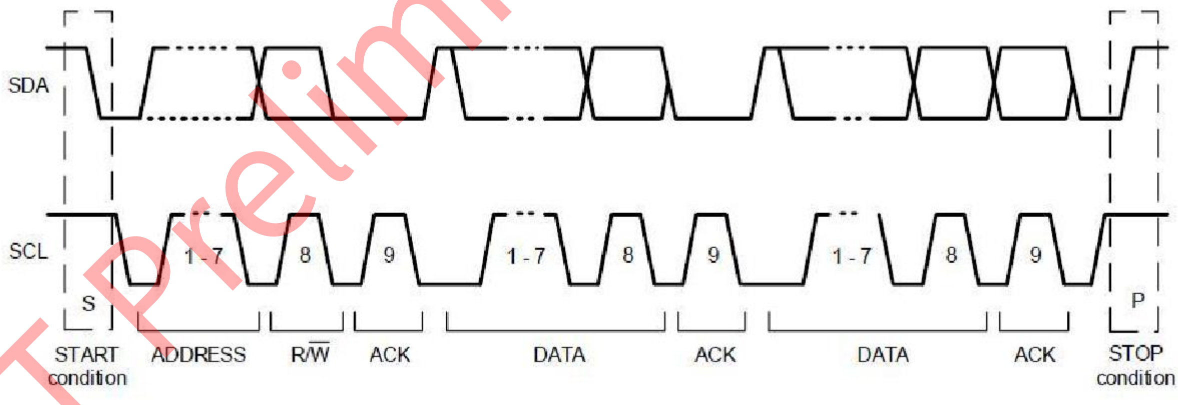
t_{SUSTO}	Setup Time for a stop condition		0.6		us
t_{BUF}	Time before a new transmission		1.3		us

9. I2C TIME DIAGRAM



The I2C interface protocol has special bus signal conditions. Start (S), stop (P) and binary data conditions are shown above. At start condition, SCL is high and SDA has a falling edge. Then the slave address is sent. After the 7 address bits, the direction control bit R/W selects the read or write operation. When a slave device recognizes that it is being addressed, it should acknowledge by pulling SDA low in the ninth SCL (ACK) cycle.
At stop condition, SCL is also high, but SDA has a rising edge. Data must be held stable at SDA when SCL is high. Data can change value at SDA only when SCL is low.

10. I2C PROTOCAL



11. REGISTER DESC.

Add.	Desc.	R/W	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
0x06	DATA_MSB	R	Pressure Data out<23:16>								0x00
0x07	DATA_CSB	R	Pressure Data out<15:8>								0x00
0x08	DATA_LSB	R	Pressure Data out<7:0>								0x00
0x09	TEMP_MSB	R	Temp Data out<15:8>								0x00
0x0A	TEMP_LSB	R	Temp Data out<7:0>								0x00
0x30	CMD	RW	Sleep_time<7:4>				Sco	Measurement_ctrl<2:0>			0x00

0xA5	Sys_config	RW	Aout_config<7:4>			LDO_ config	Unipolar	Data_out_ control	Diag_on	OTP
0xA6	P_config	RW		Input Swap	Gain_P<5:3>		OSR_P<2:0>			OTP

Reg0x06-Reg0x08 :
Pressure ADC data Register

Reg0x09-Reg0x0A :
Temperature ADC data Register

Reg0x30 : Measurement Command Register
Sleep time<7:4>: 0000:0ms; 0001:62.5ms; 0010:125ms ;... 1111: 1s, only active during sleep mode conversion. Sco: 1, Start of conversion, automatically come back to 0 after conversion ends (except sleep mode conversion). Measurement control<2:0>:

010: indicate the combined conversion (namely a temperature conversion immediately followed by a pressure-signal conversion).
011: indicate a sleep mode conversion (periodically perform a combined conversion with an interval time of 'sleep time'),

Reg0xA5(configured at factory)
Aout_config<7:4>: Analog output setting(recommending reserve default value) LDO_config: 0: set with 1.8V;
1: set with 3.6V.
Unipolar: 0: ADC output in bipolar format(signed binary),
1: ADC output in unipolar format. (Unsigned binary, Only take effect when 'raw_data_on' = 1)
Raw_data_on: 0: output calibrated data(as default value),
1: output ADC raw data.
Diag_on: 1, Enable diagnosis function(default).

Reg0xA6(configured at factory)
Input Swap: Swap VINP and VINN inside the ASIC
Gain_<5:3>: set the gain of the sensor signal conversion channel. 000: gain=1, 001: gain=2, 010: gain=4, 011: gain=8, 100: gain=16, 101: gain=32, 110: gain=64, 111: gain=128.
OSR_P<2:0>: set the over sampling ratio of the sensor signal conversion channel. 000:1024X, 001:2048X, 010:4096X, 011:8192X, 100:256X, 101:512X, 110:16384X, 111:32768X.

12. Read Operation

As the following instruction sequences for reading data:
(Only for combined conversion (namely a temperature conversion immediately followed by a pressure-signal conversion))

1. Send instructions 0x0A to 0x30 register for one temperature acquisition, one pressure data acquisition.
2. Read the 0x30 register address. If Sco bit is 0, signify the acquisition end, the data can be read. (Or, Hold 20mS and then skip this step to read data directly)
3. Read 0x06, 0x07, 0x08 register address data to form a 24-bit AD value (pressure data AD value); Read 0x09, 0x0A register address data to form a 16-bit AD value (temperature data AD value)

13. Read Pressure

The total pressure output value which include 0x06, 0x07 and 0x08 registers are 24 bits. The highest position is the symbol bit, and the value is "1", it represents "negative pressure"; The symbol digit value is "0", it represents "positive pressure".
Pressure_ADC value: = (Pressure 3rd Byte [23:16] x 65536+Pressure 2nd Byte [15:8] x 256 + Pressure1st Byte [7:0])

Note：
1 Pressure 3rd Byte [23:16] is the hexadecimal value read out by REG0x06 and need convert into decimal value;
2 Pressure 2nd Byte [15:8] is the hexadecimal value read out by REG0x07 and need convert into decimal value;

3 Pressure 1st Byte [7:0] is the hexadecimal value read out by REG0x08 and need convert into decimal value.

Pressure range(kpa)	K(value)
500<P≤1000	8
260<P≤500	16
130<P≤260	32
65<P≤130	64
32<P≤65	128
16<P≤32	256
8<P≤16	512
4<P≤8	1024
2≤P≤4	2048
1≤P<2	4096

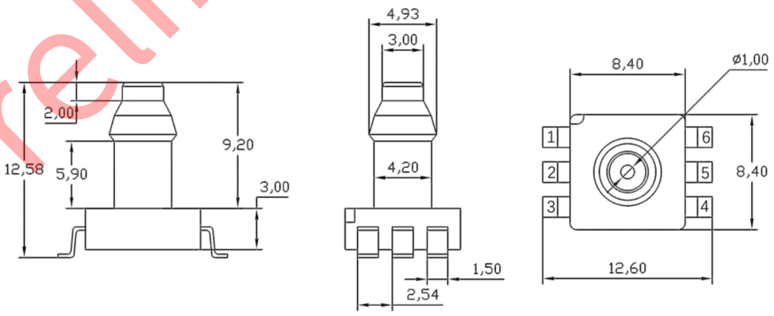
For Pressure conversion formula are as follows:
The highest bit is "0", which means positive pressure = Pressure_ ADC /k;
The highest bit is "1", which means negative pressure Pressure=(pressure_ADC-2^24)/k;
Note:
1. the unit is Pa (default). If need to display other units, fill the corresponding coefficient in the conversion formula for conversion;
2 the K value is selected according to the positive pressure value only, like -100 ~ 100kPa,the K value is 64.

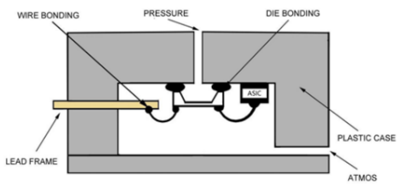
14. Read Temperature

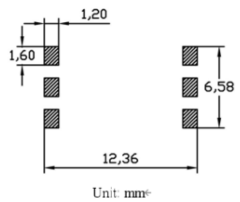
The bits of temperature output values in the 0x09 and 0x0A registers are 16 bits, the highest is the symbol bit.
The symbol digit value is "1" when it represents "negative", and the symbol digit value is "0" when it represents "positive". Supposing if the decimal values of REG0x09 and REG0x0A readout are X, Y,
For Temperature ADC value and conversion formula are as:
Temperature AD value: N=X*256+Y
If n<2^15, Temperature is positive value, actual temperature T =N/256; (℃).
If n>2^15, Temperature is negative value, actual temperature value = (N-2^16) /256; (℃)

15. Pin Layout and Definition

Unit : mm



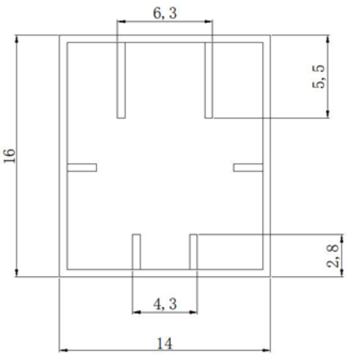


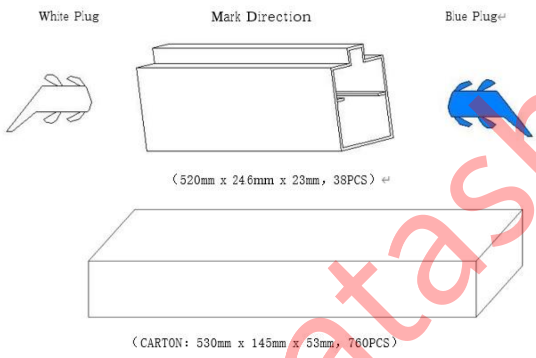


16. Packing Instructions

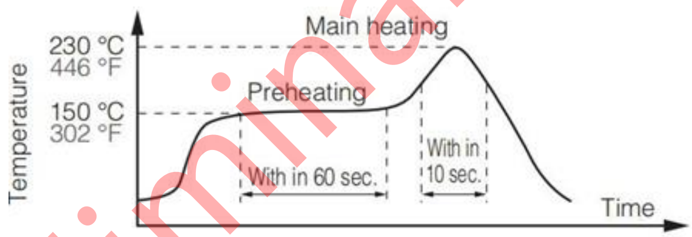
- 58 pcs per plastic tube packing
- Vibration-proof packing

Unit : mm





16. Reflow Profile Chart



Copy Rights and Disclaimer

READ CODE(C51 Example) :

```
#include <reg52.h> #include <math.h> #define DELAY_TIME 600
#define TRUE 1
#define FALSE 0
#define uchar unsigned char #define uint unsigned int

//----define IIC SCL, SDA port---- sbit SCL = P1 ^ 7;
sbit SDA = P1 ^ 6;


//----delay time_us---- void DELAY(uint t)
{
while (t != 0) t--;
}

//----IIC START CONDITION----
void I2C_Start(void)
{
SDA = 1;        //SDA output high DELAY(DELAY_TIME);
SCL = 1;
DELAY(DELAY_TIME); //SCL output high SDA = 0;
DELAY(DELAY_TIME); SCL = 0; DELAY(DELAY_TIME);
}

//----IIC STOP CONDITION----
void I2C_Stop(void)
{
SDA = 0;        //SDA OUTPUT LOW DELAY(DELAY_TIME);
SCL = 1; DELAY(DELAY_TIME); SDA = 1;

DELAY(DELAY_TIME);
SCL = 0;        //SCL OUTPUT LOW DELAY(DELAY_TIME);
}

//----IIC SEND DATA "0"----
void SEND_0(void)
{
SDA = 0; DELAY(DELAY_TIME); SCL = 1; DELAY(DELAY_TIME); SCL = 0;
DELAY(DELAY_TIME);
}

//----IIC SEND DATA "1"----
void SEND_1(void)
{
SDA = 1; DELAY(DELAY_TIME); SCL = 1; DELAY(DELAY_TIME); SCL = 0;
DELAY(DELAY_TIME);
}

//----Check SLAVE's Acknowledge---- bit Check_Acknowledge(void)
{
SDA = 1; DELAY(DELAY_TIME); SCL = 1;
DELAY(DELAY_TIME / 2); F0 = SDA; DELAY(DELAY_TIME / 2);
SCL = 0; DELAY(DELAY_TIME); if (F0 == 1)
return FALSE; return TRUE;
}


//----Write One Byte of Data---- void WriteI2CByte(uchar b) reentrant
{
char i;
for (i = 0; i < 8; i++)
if ((b << i) & 0x80) SEND_1();

else


}
```

```
SEND_0();

//----Read One Byte of Data---- uchar ReadI2CByte(void) reentrant
{
char b = 0, i;
for (i = 0; i < 8; i++)
{
SDA = 1; DELAY(DELAY_TIME); SCL = 1; DELAY(DELAY_TIME); F0 = SDA;
DELAY(DELAY_TIME); SCL = 0;
if (F0 == 1)
{



}
else


}

b = b << 1;
b = b | 0x01;


b = b << 1;

return b;
}

//----write One Byte of Data,Data from MASTER to the SLAVER----
void Write_One_Byte(uchar addr, uchar thedata) //Write "thedata" to the SLAVER's
address of "addr"
{
bit acktemp = 1; I2C_Start(); //IIC START
WriteI2CByte(0x6D << 1 + 0);        //IIC WRITE operation, SLAVER address bit: 0x6D

acktemp = Check_Acknowledge(); //check the SLAVER WriteI2CByte(addr);  //address
acktemp = Check_Acknowledge(); WriteI2CByte(thedata);       //thedata acktemp =
Check_Acknowledge(); I2C_Stop(); //IIC STOP
}

//----Read One Byte of Data,Data from SLAVER to the MASTER---- uchar
Read_One_Byte(uchar addr)
{
bit acktemp = 1; uchar mydata;

I2C_Start();
WriteI2CByte(0x6D << 1 + 0);        //IIC WRITE operation, SLAVER address bit: 0x6D
acktemp = Check_Acknowledge();
WriteI2CByte(addr);
acktemp = Check_Acknowledge(); I2C_Start();
WriteI2CByte(0x6D << 1 + 1);        //IIC READ operation, SLAVER address bit: 0x6D
acktemp = Check_Acknowledge();
mydata = ReadI2CByte(); I2C_Stop();
return mydata;
}

//----Delay_ms---- void Delay_xms(uint x)
{
uint i, j;
for (i = 0; i < x; i++)
for (j = 0; j < 112; j++)
;
}

//----The main function---- void main(void)
{
```

```
uchar pressure_H, pressure_M, pressure_L, temperature_H, temperature_L;
//temporary variables of pressure and temperature long int pressure_adc,
temperature_adc;

//The value of pressure and temperature converted by the sensor's ADC long float
pressure, temperature;
//The calibrated value of pressure and temperature Delay_xms(1000);          //delay
1000ms

while (1)
{
Write_One_Byte(0x30, 0x0A);
//indicate a combined conversion (once temperature conversion immediately followed
by once sensor signal conversion)
//more measurement method, check Register 0x30
while ((Read_One_Byte(0x30) & 0x08) > 0); //Judge whether Data collection is over, if
above action has no response, delete that command and execute this command as:
Delay_xms(20);

pressure_H = Read_One_Byte(0x06); pressure_M = Read_One_Byte(0x07); pressure_L =
Read_One_Byte(0x08);
// Read ADC output Data of Pressure
pressure_adc = pressure_H * 65536 + pressure_M * 256 + pressure_L;
//Compute the value of pressure converted by ADC


if (pressure_adc > 8388608)
pressure = (pressure_adc - 16777216) / K; //unit is Pa, select appropriate K value
according to pressure range.
else
pressure = pressure_adc / K; //unit is Pa, select appropriate K value according to
pressure range.
//The conversion formula of calibrated pressure, its unit is Pa


temperature_H = Read_One_Byte(0x09); temperature_L = Read_One_Byte(0x0A);
//Read ADC output data of temperature
temperature_adc = temperature_H * 256 + temperature_L;
//Compute the value of temperature converted by ADC
temperature = (temperature_adc – 65536) / 256;  //unit is ℃
else
temperature = temperature_adc / 256;      //unit is ℃
//The conversion formula of calibrated temperature, its unit is Centigrade

Delay_xms(100); //delay 100ms
}
}
```