

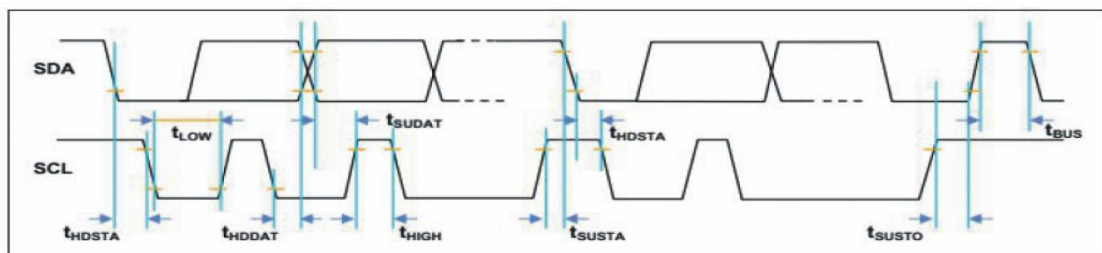
I2C Communication Protocol Specifications

● I2C Address 0x6D

A7	A6	A5	A4	A3	A2	A1	W/R
1	1	0	1	1	0	1	0/1

● I2C Communication Pin Electrical Characteristics

SYMBOL	PARAMETERS	CONDITION	MIN	MAX	UNIT
f_{SCL}	CLOCK FREQUENCY			400	kHz
t_{LOW}	CLOCK LOW FREQUENCY HOLDTIME		1.3		us
t_{HIGH}	CLOCK HIGH FREQUENCY HOLD TIME		0.6		us
t_{SUDAT}	SDA SETUP TIME		0.1		us
t_{HDDAT}	SDA SETUP TIME		0.0		us
t_{SUSTA}	DATA SETUP TIME		0.6		us
t_{HDSTA}	START CONDITION HOLD TIME		0.6		us
t_{SUSTO}	STOP CONDITION SETUP TIME		0.6		us
t_{BUS}	BUS FREE TIME BETWEEN STOP AND START CONDITION		1.3		us



I2C TIMING DIAGRAM

● I2C Data Transfer Sequences

(1): pressure measurement

Measuring data is a 24bit signed integer, stored in 3 registers of address 0x06, 0x07, 0x08.

0x06								0x07								0x08							
23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Datas can be read as per below steps:

float fadc;

```

iic_start();           //I2C start
iic_write(0xDA);       //Device address, write operation
iic_write(0x06);       //Write data start address
iic_start();           //Restart
iic_write(0xDA+1);     //Device address (0xDB), read operation
dat=iic_readbyte(1);   //Read first byte(0x06), ACK
dat <=< 8;              //Shift
dat += iic_readbyte(1); //Read second byte(0x07), ACK
dat <=< 8;              //Shift
dat += iic_readbyte(0); //Read third byte(0x08), NACK
iic_stop();            //Stop
//Calculate pressure value
if(dat & 0x800000)
{
    fadc= dat - 16777216.0;
}
else
{
    fadc = dat;
}
ADC = 3.3* fadc /8388608.0;
P = Range * (ADC-0.5)/2.0;
Note: ADC: intermediate variables;

```

Range: sensor pressure range; for example if sensor measurement range is : 0...500kPa, then

RANGE=500-0=500.

P: measuring pressure value.

(2): Temperature measurement

Measuring data is a 24bit signed integer, stored in 3 registers of address 0x09, 0x0A, 0x0B.

0x09								0x0A								0x0B							
23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Datas can be read as per below steps:

float fadc;

```

iic_start();           //I2C start
iic_write(0xDA);       //Device address, write operation
iic_write(0x09);       //Write data start address
iic_start();           //Restart
iic_write(0xDA+1);     //Device address, read operation
dat=iic_readbyte(1);   //Read first byte(0x09), ACK
dat <=<= 8;            //Shift
dat += iic_readbyte(1); //Read second byte(0x0A), ACK
dat <=<= 8;            //Shift
dat += iic_readbyte(0); //Read third byte(0x0B), NACK
iic_stop();            //Stop
//Calculate temperature value
if(dat & 0x800000)
{
    fadc= dat - 16777216.0;
}
else
{
    fadc = dat;
}
T = 25.0+fadc /65536.0;
T: measuring temperature value, unit: °C.
    
```