

```

U8 bufptr[10];
unsigned char AS1318_SendByte(unsigned char byte)
{
    unsigned char error;
    IIC_Send_Byte(byte);
    /* now for an ack */
    /* Master generates clock pulse for ACK */
    SDA_IN();//SDA set input
    IIC_SDA = 1; /* release SDA ... listen for ACK */
    delay_us(2);
    IIC_SCL=1; /* ACK should be stable ... data not allowed to change when SCL is high */
    delay_us(2);
    //SDA_IN();//SDA set input
    /* SDA at 0 ? */
    error = (READ_SDA & 0x01); /* ack didn't happen if bit 0 = 1 */
    delay_us(2);
    IIC_SCL=0;
    delay_us(2);
    return(error);
}

u8 Data_Fetch = 4;
unsigned char Read_AS1318(unsigned char DF_Command)
{
    u8 I2c_Address=0x28;//0x28,0x36,0x46
    IIC_Start();
    if(AS1318_SendByte((I2c_Address<<1)+1))
    {
        return (1);
    }
    delay_ms(1);
    DF_Command = DF_Command-1;
    AIoTensing
    bufptr [3] = IIC_Read_Byte (1);
    bufptr [2] = IIC_Read_Byte (1);
    bufptr [1] = IIC_Read_Byte (1);
    bufptr [0] = IIC_Read_Byte (0);
    IIC_Stop();
    return (0);
}

void Read_AS1318_Pressure(void)
{
    u8 i;
    float Pressure, Temperature,data1,data2,Range;
    unsigned int Dpressure, Dtemperature;

```

```

float P1=1638.3; /* P1= 10% * 16383 - A type*/
float P2=14744.7; /* P2= 90% * 16383 - A type*/
float Pmax=1600;//1.6Mpa 1600kPa
float Pmin=0;
Range = (Pmax-Pmin) / (P2-P1);
data1 = 0;
data2 = 0;
IIC_Init();
Read_AS1318 (Data_Fetch);/*Read_DF4 command - data fetch 4 bytes */
if ((bufptr [3] & 0xc0) ==0x00) /*test status of the 2 MSBs of the bridge high byte of data*/
{
Dpressure= ((unsigned int) (bufptr [3] & 0x3f) <<8) + (bufptr [2]);
Dtemperature= (((unsigned int) bufptr [1]) <<3) + bufptr [0];
}
delay_ms(1);
}
Pressure= (((float) Dpressure)-P1) * Range + Pmin;
Temperature= ((float) Dtemperature) * 200 / 2047 -50;
}

```