I2C communication with PHPS 5500, 5600 pressure sensors

1.I2C protocol

1.1.General description

In I2C communication a serial data line (SDA) and a serial clock line (SCL) are required for communication between connected devices to I2C bus. Both connected lines SDA and SCL are bidirectional lines, which are connected to supply voltage with pull-up resistors (see application circuit on Figure 6). As seen from figure 1 there can be more slave devices (up to 127) connected to I2C bus, what is limited with the number of 7 bit slave address. I2C bus is free when both connection lines are HIGH, and can be put LOW by devices connected to the I2C bus.

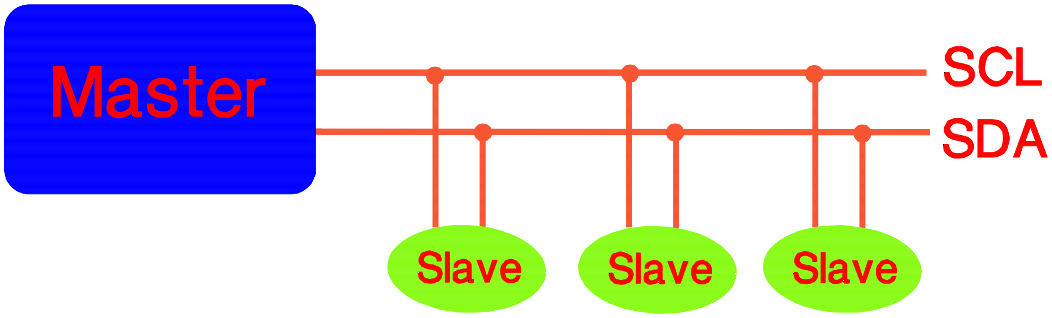


Figure 1: I2C communication example

I2C communication acts as a Master – Slave principle (see figure 2), where there is a master device which generates the clock (SCL) and generates START & STOP command for data transition.

Figure 2: Master – Slave principle

Masters and slaves can act as a transmitter or as a receiver depending of the information which needs to be sent or read. Transmitter is device which sends data to the I2C bus ("master transmitter" normally sends requests to the slave, when "slave transmitter" normally sends information replies to the master). Receiver is device which receives data from I2C bus.

I2C standard protocol is presented in figure 3.

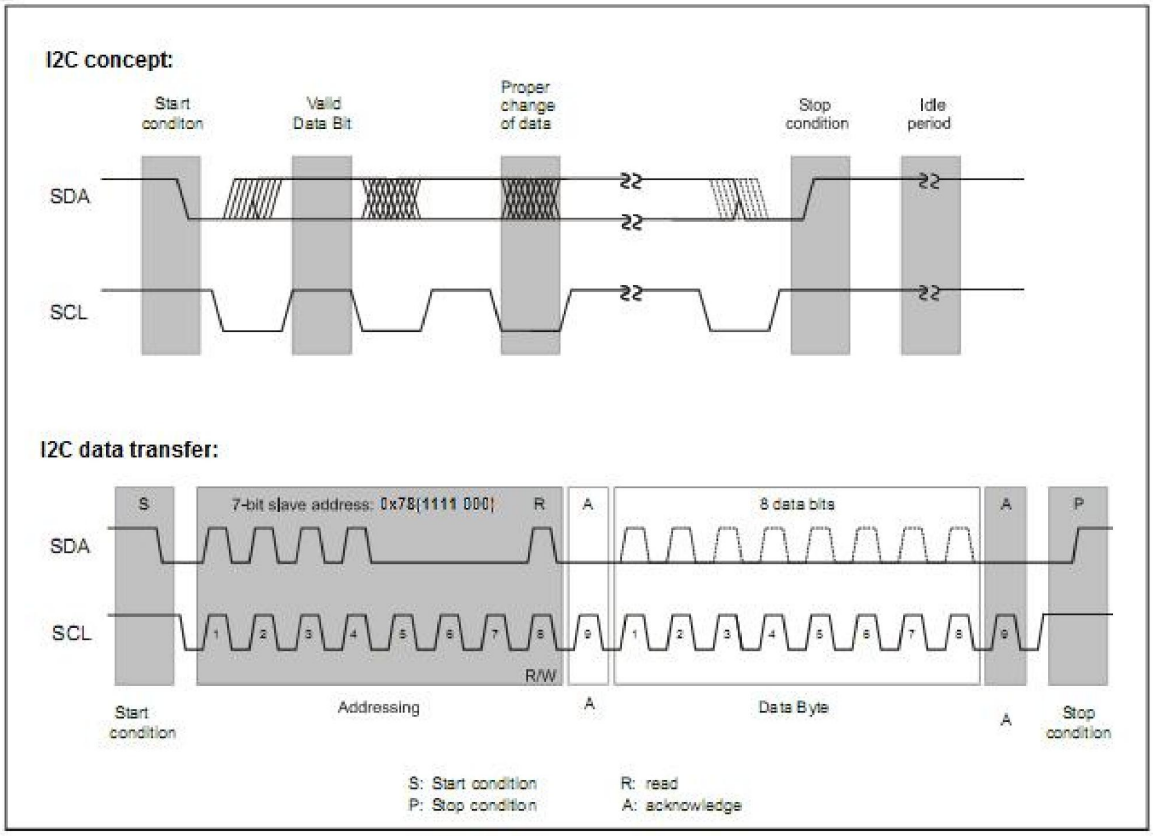


Figure 3: I2C standard communication protocol

1.2. Communication phases

- **Buss free – Idle state:** When bus is free both lines SCL and SDA are pulled up – HIGH.
- **START condition (S):** Each data transfer starts with the start condition, which is always sent by the master. This start condition acts as a signal to all I2C connected devices giving information that there will be something transmitted. Start condition is defined as transition of HIGH to LOW on SDA line when SCL line is HIGH (see figure 3).

- **STOP condition (P):** Each data transfer stops with stop condition, which is also generated by a master, when a data transfer has finished. Stop condition is defined as transition of LOW to HIGH on SDA line when SCL line is HIGH (see figure 3).
- **Valid data :** Data is always transmitted in bytes (8 bits) starting with the MSB (most significant bit). One data bit is transferred with each clock pulse. Transmitted data are valid (after generating start condition) only during HIGH period of clock and data can changes can be done during LOW period of clock (see figure 3).
- **Acknowledge (A):** Each sent byte needs to be followed with the acknowledge bit generated from the receiver that correct data has been received. Acknowledge means also that device can continue with further data transfer. For that purpose must master generate extra clock pulse. Transmitter releases clock HIGH during acknowledge clock pulse, if not then further byte will not be sent.
- **Slave address:** After start condition master sends addressing byte - slave address to define with which slave device he wants to communicate. This addressing byte includes 7 bit slave address (up to 128 devices) + 1 R/W bit (data direction bit). If R/W bit is set to “0” (W) then master wishes to transmit data to selected slave. If R/W bit is set to “1” (R) then master request data from the slave. The addressed slave answers with an acknowledge, all other slaves connected with the I2C-bus ignore this communication.

  PHPS pressure sensors have default programmed slave address to 0x78 (1111 000b). For connected more slave devices to I2C bus each connected device should have its own slave address (up to 128 devices).

1.3. I2C communication overview

In figure 3 is presented complete data transfer. After generating start condition master also sends slave address with data direction bit (R/W), which gives read or write transfer. Addressed slave replies to this always with acknowledge (A) first. Now can be transferred unlimited numbers of data (bytes) which needs to be always confirmed with acknowledge bit. This transfer can be stopped by the master with generating the stop condition. If master wishes to communicate also with other slave address, it can generate also a second start condition without stopping the first one.

2. DIGITAL DATA TRANSFER ON I2C BUS

2.1. PRESSURE & TEMPERATURE DATA TRANSFER ON I2C BUS

Digital data transfer is presented in figure 4.

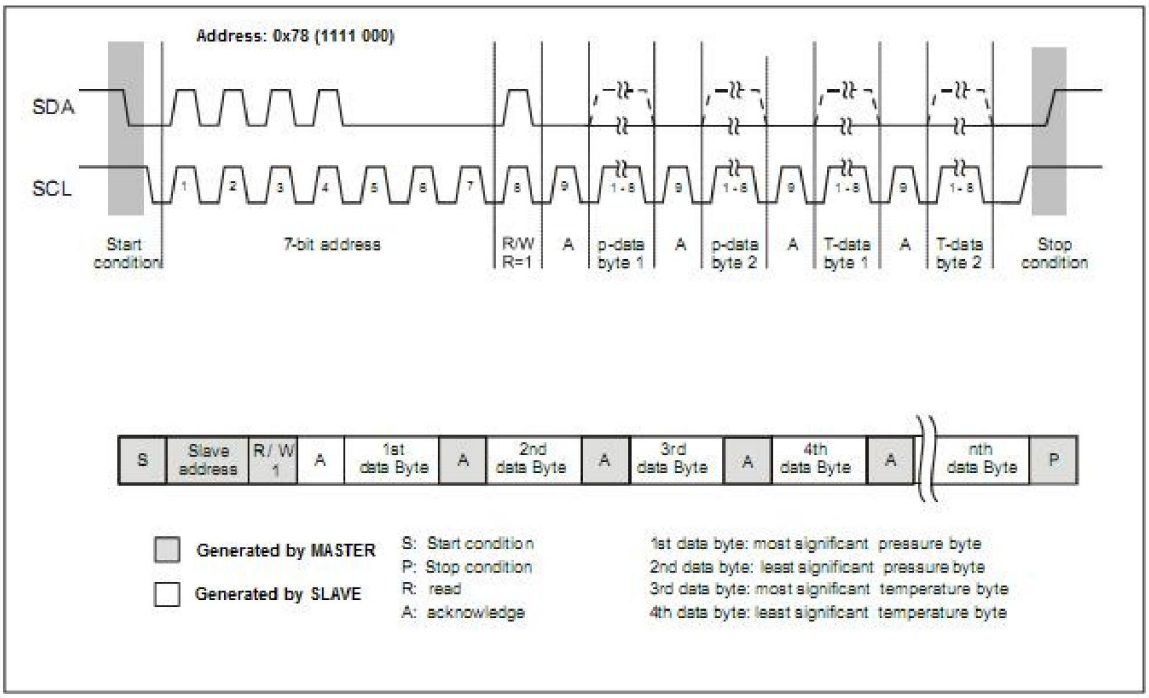


Figure 4: Digital pressure & temperature data transfer from PHPS sensors

Pressure and temperature output signals form PHPS pressure sensors come as 15 bit values to the output register. Master, which would like to read this data, starts communication with the start condition. After that master sends 7 bit slave address (factory default is 0x78) and data direction bit R/W (for read data R/W="1"). Slave confirms this address with acknowledge (A) bit first and afterwards sends desired data with bytes (8 bits): first byte is most significant byte for pressure value, second byte is least significant by for pressure value, third byte is most significant byte for temperature value, fourth byte if least significant byte for temperature value. Master must confirm each received byte with acknowledge bit (see figure 4). Master can stop the data transfer by sending the stop condition or it can generate additional acknowledge bit after 4 receiving bytes of data (pressure and temperature) for continues data receiving from slave (PHPS sensor).

2.1. Calculation pressure formula

Master receives pressure data as a 15 bit values which can be converted to actual pressure data with pressure units in mbar using simple below formula.

Definitions:

P= pressure (mbar)
Pmin= min pressure (mbar)
Pmax= max pressure (mbar)
D = digital pressure (counts)
Dmax = max digital pressure (counts)
Dmin = min digital pressure (counts)
S= sensitivity (count/mbar)

$$S = \frac{Dmax - Dmin}{Pmax - Pmin} \qquad P = \frac{D - Dmin}{S} + Pmin$$

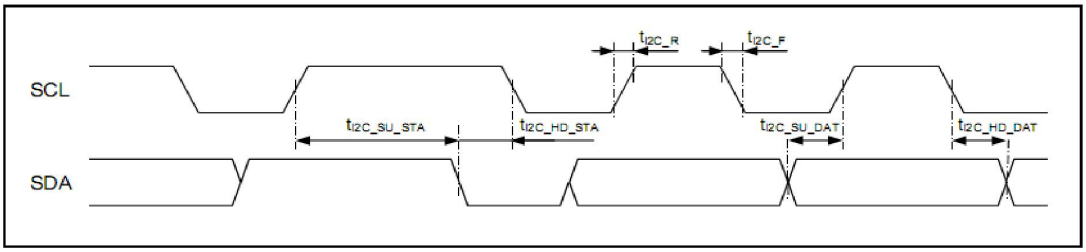
Example: For pressure sensor with pressure range 0 to 350 mbar with analog output 0.5 to 4.5 V (equivalent digital output 3277 to 29491 counts) we measure digital value of 7850 counts. Let's calculate this value in pressure with pressure units in mbar:

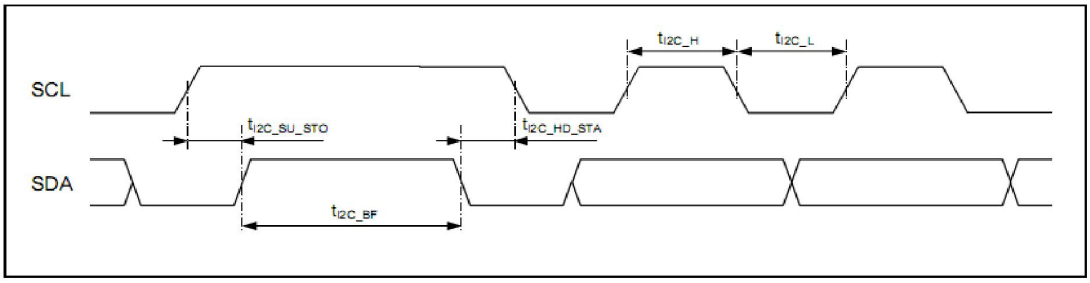
$$S = \frac{29491 - 3277}{350mbar - 0mbar} = 74{,}90 \text{ counts/mbar}$$

$$P = \frac{7850 - 3277}{74{,}90} + 0mbar = 61{,}05 \text{ mbar}$$

Temperature values from digital temperature values are calculated in the same manner.

2.2. I2C Timings parameters





Nr.	Parameter	Symbol	min	typ	max	Unit	Conditions
1	SCL Clock frequency1	f_{SCL}			400	kHz	$f_{CLK} \geq 2MHz$
2	Bus free time betw. start and stop condition	t_{I2C_BF}	1.3			µs	
3	Hold time start condition	$t_{I2C\_HD\_STA}$	0.6			µs	
4	Setup time repeated start condition	$t_{I2C\_SU\_STA}$	0.6			µs	
5	Low period SCL/SDA	t_{I2C_L}	1.3			µs	
6	High period SCL/SDA	t_{I2C_H}	0.6			µs	
7	Data hold time	$t_{I2C\_HD\_DAT}$	0			µs	
8	Data setup time	$t_{I2C\_SU\_DAT}$	0.1			µs	
9	Rise time SCL/SDA	t_{I2C_R}			0.3	µs	
10	Fall time SCL/SDA	t_{I2C_F}			0.3	µs	
11	Setup time stop condition	$t_{I2C\_SU\_STO}$	0.6			µs	
12	Noise interception SDA	t_{I2C_NI}			50	ns	Spikes are suppressed

Figure 5: Timing I2C protocol

3. APPLICATION SHEME

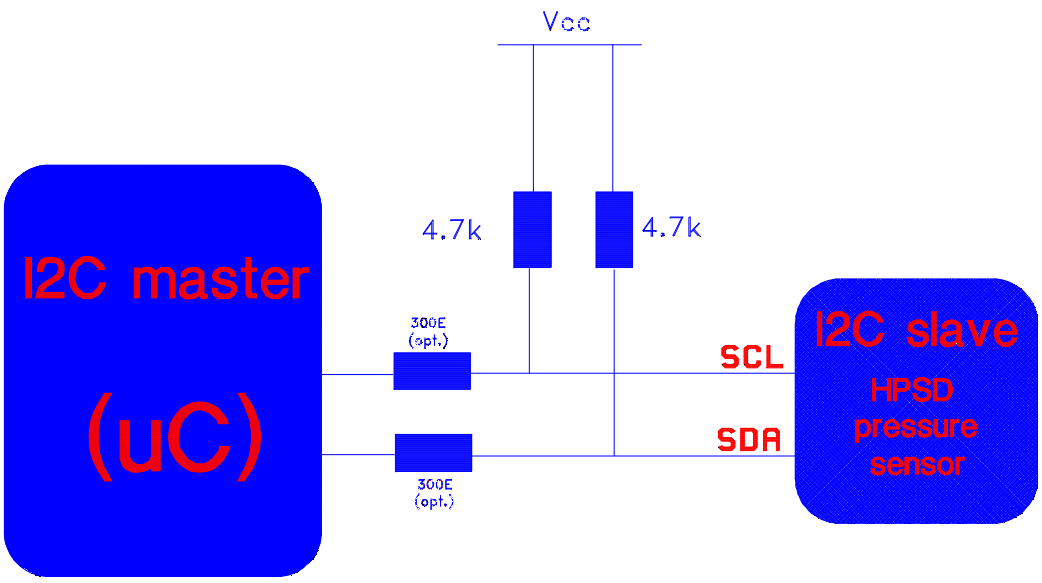


Figure 6: Application circuit

SCL and SDA lines needs to be connected to power supply via pull-up resistors as shown in figure 6. We recommend to use 4.7k ohms resistors as a pull-up resistors and 300 ohm resistors as serial resistors.

4. EXAMPLE PROGRAM CODE

Simple example code for pressure readings from PHPS pressure sensor is presented below:

```
byte msb, lsb; // 2 x 8bit values
int16 pressure; // 1 x 16bit value
// Set I2C unit to I2C master mode, clock speed 100 kHz and 7 bit addressing
configureI2C (I2C_MASTER | CLK_SPEED_100KHZ | ADDRESSING_7BIT);
// Set the target default slave address (0x78 = 120dec)
I2C_set_target(0x78);
// Send start condition (slave)
I2C_send_start_read();
// Read first data byte (msb) & answer with ACK (continue communication)
I2C_read (&msb, SEND_ACK);
// Read second data byte (lsb) and answer with NACK (end communication)
I2C_read (&lsb, SEND_NACK);
// Send stop condition
I2C_send_stop();
// Put both values together
pressure = ((int16)msb << 8) | lsb;
```

Angst+Pfister
Sensors and Power

We are here for you. Addresses and Contacts.

Headquarter Switzerland:

Angst+Pfister Sensors and Power AG
Thurgauerstrasse 66
CH-8050 Zurich
Phone +41 44 877 35 00
sensorsandpower@angst-pfister.com

Office Germany:

Angst+Pfister Sensors and Power Deutschland GmbH
Edisonstraße 16
D-85716 Unterschleißheim
Phone +49 89 374 288 87 00
sensorsandpower.de@angst-pfister.com

Scan here and get an overview of personal contacts!

sensorsandpower.angst-pfister.com