

52-Series Communication protocol

Read Example (Full Measurement)

Byte#				
Send by Master		0x51		
	S	101000	1	A
		Address(0x28)	R	

Byte#	0	1	2	3	4	5	6		
Receive from Slave	A	[15:8] A	[7:0] A		A	[15:8] A	[7:0] A		A P
	Status	Pressure Data		xx		Temperature Data		xx	

Pressure conversion:

	Raw Readout (Byte 2&3)
At 0 Pressure	6553
At FS Pressure	58982

Formula: $(\text{RawData} - 6553) / (58982 - 6553) * \text{FS Pressure}$

Example with 52DL-007P at 7 Psi: $(58982 - 6553) / (52429) * 7 = 7\text{Psi}$

Temperature conversion:

$\text{TempData} / 65535 * 125 - 40 = ^\circ\text{C}$

Status Byte:

Standard status byte: 0x68. In case of other status byte please contact support.

Bit#	7	6	5	4	3	2	1	0
Meaning	0	Power	Busy	Mode		Memory Error	Connection Error	Math Saturation

Example code:

```
#include <Wire.h> // i2c library
#include <stdio.h>

int Adress = 0x28; //52-Series adress found on Datasheet or I2C Scanner

void setup() {
  Serial.begin(9600);
  Wire.begin(); //start communication for i2c
}

void loop() {
  Wire.requestFrom(Adress, 6); // requesting to read all 6 bytes from the Sensor adress
  //put the output in variable c
  byte c1 = Wire.read(); //read 6 bytes and define variable for them
  byte c2 = Wire.read();
  byte c3 = Wire.read();
  byte c4 = Wire.read();
  byte c5 = Wire.read();
  byte c6 = Wire.read();

  unsigned short c = (c2 << 8) + c3; //Byte 2 & 3 for Pressure output
  unsigned short d = (c5 << 8) + c6; //Byte 5 & 6 for Temperature output
  float p, t; //p = Pressure output, t = Temperature output

  p = (float)(c - 6553) / 52429 * 482.63301051; // 7psi = 482.63301051mBar: insert Range of your
  Version there

  t = (float)d / 65535 * 125 - 40; //Formula for Temp. calculation: 16-bit temperature output / 65535
  * 125 - 40 = temperature(degC)

  Serial.print("Pressure(mBar):");
  Serial.print(p);
  Serial.print('\n');
  Serial.print("Temperature(°C):");
  Serial.print(t);
  Serial.print('\n');
  Serial.print("Status: 0x");
  Serial.print(c1, HEX); //c1 is Status Byte
  Serial.print('\n');
  Serial.print('\n');
  delay(500);
}
```

Headquarter Switzerland:
Pewatron AG
Thurgauerstrasse 66
CH-8050 Zurich
Phone +41 44 877 35 00
info@pewatron.com

Office Germany:
Pewatron Deutschland GmbH
Edisonstraße 16
D-85716 Unterschleißheim
Phone +49 89 374 288 87 00
info.de@pewatron.com



PEWATRON
SENSORS · POWER SOLUTIONS

We are here for you. Addresses and Contacts.

Sales Germany & Austria

Geometrical sensors Other products

Kurt Stritzelberger
Phone +49 89 374 288 87 22
kurt.stritzelberger@pewatron.com

Pressure sensors Other products

Gerhard Vetter
Phone +49 89 374 288 87 26
gerhard.vetter@pewatron.com

Gas sensors and modules

Peter Felder
Phone +41 44 877 35 05
peter.felder@pewatron.com

Sales Switzerland & Liechtenstein

Postcode 3000 – 9999

Basil Frei
Phone +41 44 877 35 18
basil.frei@pewatron.com

Postcode 1000 – 2999

Christian Mohrenstecher
Phone +41 76 444 57 93
christian.mohrenstecher@pewatron.com

Sales International Key Accounts

Peter Felder
Phone +41 44 877 35 05
peter.felder@pewatron.com

Sales Other Countries / Product Management

Pressure Sensors Load Cells

Philipp Kistler
Phone +41 44 877 35 03
philipp.kistler@pewatron.com

Gas sensors Gas sensor modules

Dr. Thomas Clausen
Phone +41 44 877 35 13
thomas.clausen@pewatron.com

Flow / Level / Medical products

Dr. Adriano Pittarelli
Phone +49 89 374 288 87 67
adriano.pittarelli@pewatron.com

Power supplies

Sebastiano Leggio
Phone +41 44 877 35 06
sebastiano.leggio@pewatron.com

Linear position sensors Angle sensors

Eric Letsch
Phone +41 44 877 35 14
eric.letsch@pewatron.com

Accelerometers Sensor elements

Christoph Kleye
Phone +49 89 374 288 87 61
christoph.kleye@pewatron.com

Drive technology

CH Postcode 5000 – 9999 / DE

Roman Homa
Phone +41 76 444 00 86
roman.homa@pewatron.com

Drive technology

CH Postcode 1000 – 4999 / AT / IT / FR

Christian Mohrenstecher
Phone +41 76 444 57 93
christian.mohrenstecher@pewatron.com

Harald Thomas

Phone +49 89 374 288 87 23
harald.thomas@pewatron.com