



BMP388 Barometric Pressure Sensor User Manual

OVERVIEW

This is a 24-bit high precision barometric pressure sensor, which allows accurate altitude tracing, as well as barometric pressure / temperature measuring. It supports both I2C and SPI interfaces, also is compatible with 3.3V/5V voltage levels.

Due to its ultra-small form factor, low power consumption, low noise and high precision, the BMP388 Barometric Pressure Sensor is suited for applications such as drones, accurate altimeter, environment monitoring, and IoT projects, and so on.

FEATURES

- Supports I2C (default) / SPI interface
- Onboard voltage level translator, compatible with 3.3V/5V operating voltage
- Comes with development resources and manual (examples for Raspberry Pi/Arduino/STM32)

SPECIFICATION

- Operating voltage: 3.3V/5V
- Communication interface: I2C/SPI
- Barometric pressure operation range: 300~1250hPa
- Barometric pressure absolute accuracy: $\pm 0.40\text{hPa}$ (@900~1100hPa, 25~40°C)
- Barometric pressure relative accuracy: $\pm 0.08\text{hPa}$ (@900~1100hPa, 25~40°C)
- Temperature coefficient offset: $\pm 0.75\text{Pa/K}$ (@700~1100hPa, -20~65°C)

- Temperature absolute accuracy: $\pm 0.5^{\circ}\text{C}$ (0~65°C)
- Possible resolution: 0.016Pa (high precision mode)
- Possible sampling rate: 200Hz
- Operating voltage: -40~85°C
- Dimension: 32mm × 20mm
- Mounting hole size: 2.0mm

PINOUTS

PIN	I2C
VCC	3.3V/5V power supply
GND	Ground
SDA	I2C data
SCL	I2C clock
INT	Interrupt output, can be connected to I/O

WORKIGN WITH RASPBERRY PI

INSTALL LIBRARIES.

To run the demo codes successfully, you need to first install necessary libraries. Open

Terminal of Raspberry Pi and connect it to network.

Install wiringpi libraries:

```
cd  
  
git clone git://git.drogon.net/wiringPi
```

```
cd ~/wiringpi/  
  
./build
```

Python libraries:

```
cd  
  
sudo apt-get install python-smbus
```

SETTING AND RUNNING CODES**Hardware connection:**

Sensor	Pi
VCC	3.3V
GND	GND
SDA	SDA
SCL	SCL
INT	Not Connected

Software setting:

Enable I2C interface of Raspberry Pi as below:

```
sudo raspi-config
```

Choose Interfacing Options->I2C->Yes

Python Codes:

Enter the folder of python demo code, execute command to run the code:

```
sudo python bmp388Demo.py
```

The data are printed as below:

```
pi@retroPie:~/Raspberry/python/bmp388Demo $ sudo python bmp388Demo.py
BMP388 Test Program ...

Pressure sensor is BMP388!

_load_calibration

Temperature = 27.7 Pressure = 101090.32 Altitude =19.56
Temperature = 27.7 Pressure = 101092.55 Altitude =19.37
Temperature = 27.8 Pressure = 101093.74 Altitude =19.27
Temperature = 27.8 Pressure = 101087.92 Altitude =19.76
Temperature = 27.8 Pressure = 101097.86 Altitude =18.93
Temperature = 27.8 Pressure = 101093.91 Altitude =19.26
Temperature = 27.8 Pressure = 101098.89 Altitude =18.84
Temperature = 27.8 Pressure = 101099.91 Altitude =18.76
Temperature = 27.8 Pressure = 101089.97 Altitude =19.59
Temperature = 27.8 Pressure = 101097.86 Altitude =18.93
Temperature = 27.8 Pressure = 101092.03 Altitude =19.41
```

If data are display abnormally, please check the hardware connection, software setting and if the device address are correct.

wiringpi Codes:

Enter folder of wiringpi demo codes, compile and run the codes as below:

```
make
```

```
sudo ./bmp388Demo
```

After running, the data will be printed as below:

```
pi@retroPie:~/Raspberry/wiringpi/bmp388Demo $ sudo ./bmp388Demo
Pressure sensor is BMP388

/-----/

Pressure: 12636.94 Altitude: 2.38 Temperature: 3.4
Pressure: 25273.40 Altitude: 4.81 Temperature: 6.8
Pressure: 37910.34 Altitude: 7.20 Temperature: 10.2
Pressure: 50547.29 Altitude: 9.59 Temperature: 13.7
Pressure: 63185.11 Altitude: 11.91 Temperature: 17.1
Pressure: 75822.32 Altitude: 14.28 Temperature: 20.5
Pressure: 88458.77 Altitude: 16.71 Temperature: 23.9
Pressure: 101096.21 Altitude: 19.05 Temperature: 27.4
Pressure: 101097.45 Altitude: 18.95 Temperature: 27.4
Pressure: 101097.94 Altitude: 18.91 Temperature: 27.4
Pressure: 101099.42 Altitude: 18.79 Temperature: 27.4
Pressure: 101099.81 Altitude: 18.76 Temperature: 27.4
Pressure: 101098.69 Altitude: 18.85 Temperature: 27.4
Pressure: 101098.20 Altitude: 18.89 Temperature: 27.4
```

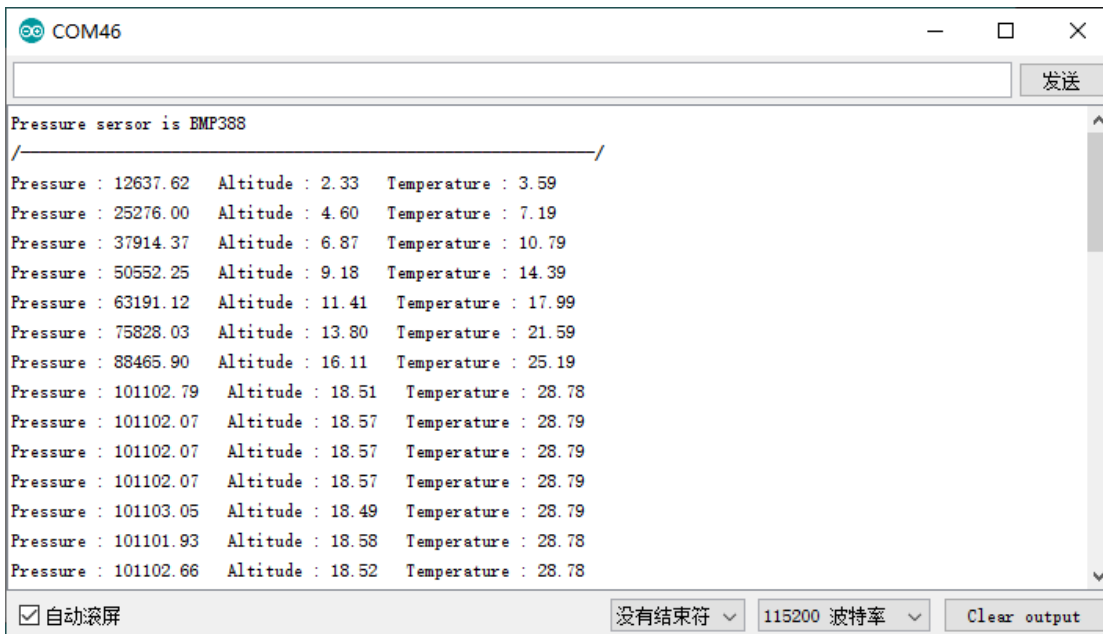
WORKIGN WITH ARDUINO

Hardware connection:

Sensor	Arduino
VCC	5V
GND	GND
SDA	SDA
SCL	SCL
INT	Not Connected

Check the hardware connect and devices address. Compile and download the code to your Arduino board.

Open Serial monitor, set the baud rate to 115200:



The screenshot shows the Arduino Serial Monitor window titled 'COM46'. The window contains the following text:

```

Pressure sensor is BMP388
/-----/
Pressure : 12637.62  Altitude : 2.33  Temperature : 3.59
Pressure : 25276.00  Altitude : 4.60  Temperature : 7.19
Pressure : 37914.37  Altitude : 6.87  Temperature : 10.79
Pressure : 50552.25  Altitude : 9.18  Temperature : 14.39
Pressure : 63191.12  Altitude : 11.41  Temperature : 17.99
Pressure : 75828.03  Altitude : 13.80  Temperature : 21.59
Pressure : 88465.90  Altitude : 16.11  Temperature : 25.19
Pressure : 101102.79  Altitude : 18.51  Temperature : 28.78
Pressure : 101102.07  Altitude : 18.57  Temperature : 28.79
Pressure : 101102.07  Altitude : 18.57  Temperature : 28.79
Pressure : 101102.07  Altitude : 18.57  Temperature : 28.79
Pressure : 101103.05  Altitude : 18.49  Temperature : 28.79
Pressure : 101101.93  Altitude : 18.58  Temperature : 28.78
Pressure : 101102.66  Altitude : 18.52  Temperature : 28.78
  
```

At the bottom of the window, there are several controls: a checked checkbox for '自动滚屏' (Auto scroll), a dropdown menu for '没有结束符' (No terminator), a dropdown menu for '115200 波特率' (115200 baud rate), and a 'Clear output' button.

If data are display abnormally, please check the hardware connection, software setting and if the device address are correct.

WORKIGN WITH STM32

Open the STM32 project, which located in STM32->open103V->bmp388_HAL->MDK-ARM->project

The demo codes is based on STM32F103VET6.

Hardware connection:

Sensor	STM32
VCC	5V
GND	GND
SDA	PB7
SCL	PB6
INT	No Connected

Programming:

Compile the codes and program it to STM32 board. USART2 is used to print sensor data. You need to connect USART2 interface of STM32 board to PC by TTL to USB module, open serial assistance software, set the serial port to 115200, 8N1:

