3. Operation

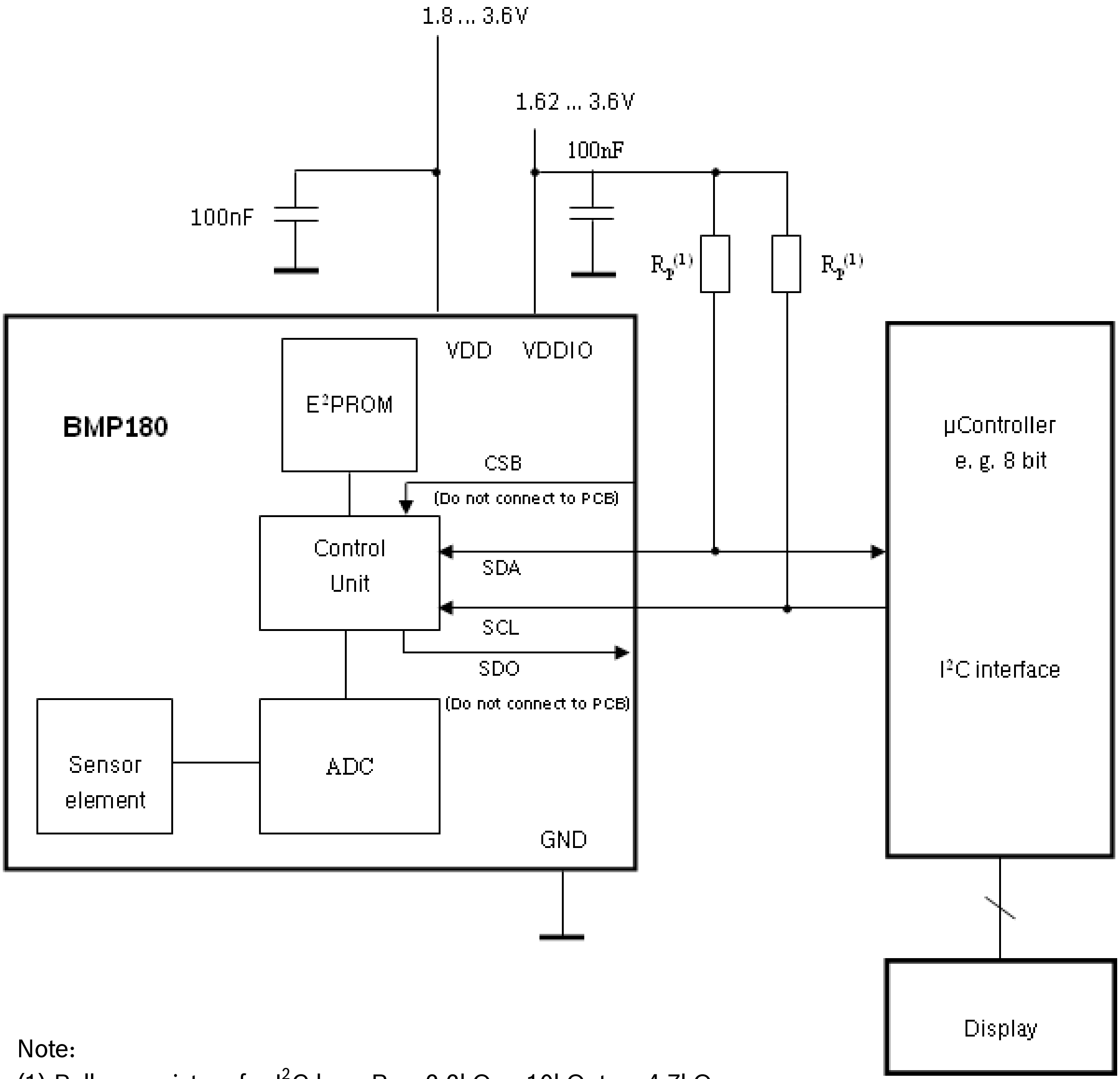
3.1 General description

The BMP180 is designed to be connected directly to a microcontroller of a mobile device via the I^2C bus. The pressure and temperature data has to be compensated by the calibration data of the E^2PROM of the BMP180.

3.2 General function and application schematics

The BMP180 consists of a piezo-resistive sensor, an analog to digital converter and a control unit with E^2PROM and a serial I^2C interface. The BMP180 delivers the uncompensated value of pressure and temperature. The E^2PROM has stored 176 bit of individual calibration data. This is used to compensate offset, temperature dependence and other parameters of the sensor.

- UP = pressure data (16 bit)
- UT = temperature data (16 bit)



Note:
(1) Pull-up resistors for I²C bus, R_p = 2.2kΩ ... 10kΩ, typ. 4.7kΩ

Figure 2: Typical application circuit

3.3 Measurement of pressure and temperature

For all calculations presented here an ANSI C code is available from Bosch Sensortec ("BMP180 _API").

The microcontroller sends a start sequence to start a pressure or temperature measurement. After converting time, the result value (UP or UT, respectively) can be read via the I^2C interface. For calculating temperature in °C and pressure in hPa, the calibration data has to be used. These constants can be read out from the BMP180 E^2PROM via the I^2C interface at software initialization.

The sampling rate can be increased up to 128 samples per second (standard mode) for dynamic measurement. In this case, it is sufficient to measure the temperature only once per second and to use this value for all pressure measurements during the same period.

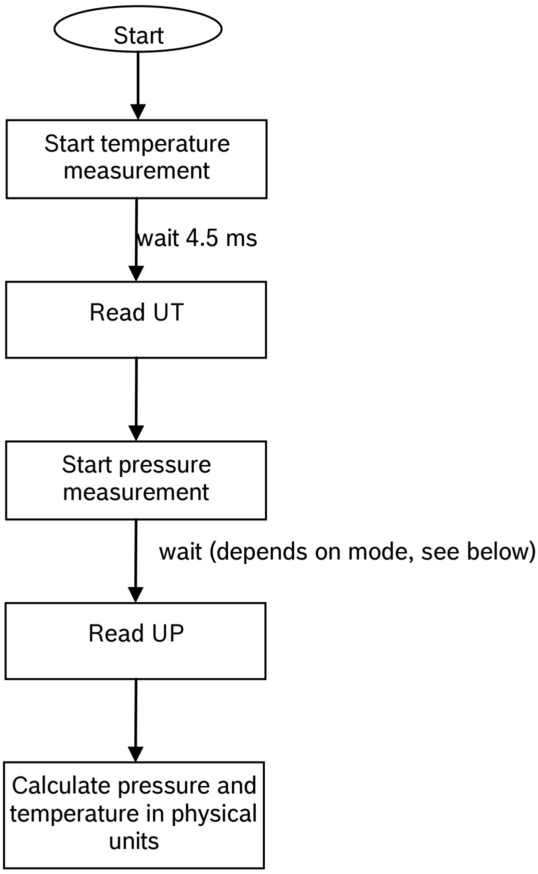


Figure 3: Measurement flow BMP180

3.3.1 Hardware pressure sampling accuracy modes

By using different modes the optimum compromise between power consumption, speed and resolution can be selected, see below table.

Table 3: Overview of BMP180 hardware accuracy modes, selected by driver software via the variable *oversampling_setting*

Mode	**Parameter** *oversampling_setting*	**Internal number of samples**	**Conversion time pressure max. [ms]**	**Avg. current @ 1 sample/s typ. [μA]**	**RMS noise typ. [hPa]**	**RMS noise typ. [m]**
ultra low power	0	1	4.5	3	0.06	0.5
standard	1	2	7.5	5	0.05	0.4
high resolution	2	4	13.5	7	0.04	0.3
ultra high resolution	3	8	25.5	12	0.03	0.25

For further information on noise characteristics see the relevant application note “Noise in pressure sensor applications”.

All modes can be performed at higher speeds, e.g. up to 128 times per second for standard mode, with the current consumption increasing proportionally to the sample rate.

3.3.2 Software pressure sampling accuracy modes

For applications where a low noise level is critical, averaging is recommended if the lower bandwidth is acceptable. Oversampling can be enabled using the software API driver (with OSR = 3).

Table 4: Overview of BMP180 software accuracy mode, selected by driver software via the variable *software_oversampling_setting*

Mode	Parameter *oversampling_setting*	software_ oversampling_setting	Conversion time pressure max. [ms]	Avg. current @ 1 sample/s typ. [µA]	RMS noise typ. [hPa]	RMS noise typ. [m]
Advanced resolution	3	1	76.5	32	0.02	0.17

3.4 Calibration coefficients

The 176 bit E^2PROM is partitioned in 11 words of 16 bit each. These contain 11 calibration coefficients. Every sensor module has individual coefficients. Before the first calculation of temperature and pressure, the master reads out the E^2PROM data.
The data communication can be checked by checking that none of the words has the value 0 or 0xFFFF.

Table 5: Calibration coefficients

	BMP180 reg adr	
Parameter	**MSB**	**LSB**
AC1	0xAB	0xAA
AC2	0xAD	0xAC
AC3	0xAF	0xAE
AC4	0xB1	0xB0
AC5	0xB3	0xB2
AC6	0xB5	0xB4
B1	0xB7	0xB6
B2	0xB9	0xB8
MB	0xBB	0xBA
MC	0xBD	0xBC
MD	0xBF	0xBE

4. Global Memory Map

The memory map below shows all externally accessible data registers which are needed to operate BMP180. The left columns show the memory addresses. The columns in the middle depict the content of each register bit. The colors of the bits indicate whether they are read-only, write-only or read- and writable. The memory is volatile so that the writable content has to be re-written after each power-on.
Not all register addresses are shown. These registers are reserved for further Bosch factory testing and trimming.

Register Name	Register Adress	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Reset state
out_msb	F7h	adc_out_msb<7:0>								80h
out_lsb	F6h	adc_out_lsb<7:0>								00h
ctrl_meas	F4h	oss<1:0>		sco	measurement control					00h
soft reset	E0h	reset								00h
id	D0h	id<7:0>								55h
calib21 downto calib0	BFh *down* to AAh	calib21<7:0> *down* to calib0<7:0>								n/a

Registers:	Control registers	Calibration registers	Data registers	Fixed
Type:	read / write	read only	read only	read only

Figure 6: Memory map

Measurement control (register F4h <4:0>): Controls measurements. Refer to Table 8 for usage details.

Sco (register F4h <5>): Start of conversion. The value of this bit stays “1” during conversion and is reset to “0” after conversion is complete (data registers are filled).

Oss (register F4h <7:6>): controls the oversampling ratio of the pressure measurement (00b: single, 01b: 2 times, 10b: 4 times, 11b: 8 times).

Soft reset (register E0h): Write only register. If set to 0xB6, will perform the same sequence as power on reset.

Chip-id (register D0h): This value is fixed to 0x55 and can be used to check whether communication is functioning.

After conversion, data registers can be read out in any sequence (i.e. MSB first or LSB first). Using a burst read is not mandatory.

5. I^2C Interface

- I^2C is a digital two wire interface
- Clock frequencies up to 3.4Mbit/sec. (I^2C standard, fast and high-speed mode supported)
- SCL and SDA needs a pull-up resistor, typ. 4.7kOhm to V_{DDIO}
  (one resistor each for all the I^2C bus)

The I^2C bus is used to control the sensor, to read calibration data from the E^2PROM and to read the measurement data when A/D conversion is finished. SDA (serial data) and SCL (serial clock) have open-drain outputs.

For detailed I^2C-bus specification please refer to:
http://www.nxp.com/acrobat_download/literature/9398/39340011.pdf

5.1 I^2C specification

Table 6: Electrical parameters for the I^2C interface

Parameter	Symbol	Min.	Typ	Max.	Units
Clock input frequency	f_{SCL}			3.4	MHz
Input-low level	V_{IL}	0		0.2 * V_{DDIO}	V
Input-high level	V_{IH}	0.8 * V_{DDIO}		V_{DDIO}	V
Voltage output low level @ V_{DDIO} = 1.62V, I_{OL} = 3mA	V_{OL}			0.3	V
SDA and SCL pull-up resistor	$R_{pull\text{-}up}$	2.2		10	kOhm
SDA sink current @ V_{DDIO} = 1.62V, V_{OL} = 0.3V	I_{SDA_sink}		9		mA
Start-up time after power-up, before first communication	t_{Start}	10			Ms

5.2 Device and register address

The BMP180 module address is shown below. The LSB of the device address distinguishes between read (1) and write (0) operation, corresponding to address 0xEF (read) and 0xEE (write).

Table 7: BMP180 addresses

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

5.3 I²C protocol

The I²C interface protocol has special bus signal conditions. Start (S), stop (P) and binary data conditions are shown below. At start condition, SCL is high and SDA has a falling edge. Then the slave address is sent. After the 7 address bits, the direction control bit R/W selects the read or write operation. When a slave device recognizes that it is being addressed, it should acknowledge by pulling SDA low in the ninth SCL (ACK) cycle.
At stop condition, SCL is also high, but SDA has a rising edge. Data must be held stable at SDA when SCL is high. Data can change value at SDA only when SCL is low.
Even though V_{DDIO} can be powered on before V_{DD}, there is a chance of excessive power consumption (a few mA) if this sequence is used, and the state of the output pins is undefined so that the bus can be locked. Therefore, V_{DD} *must* be powered before V_{DDIO} unless the limitations above are understood and not critical.

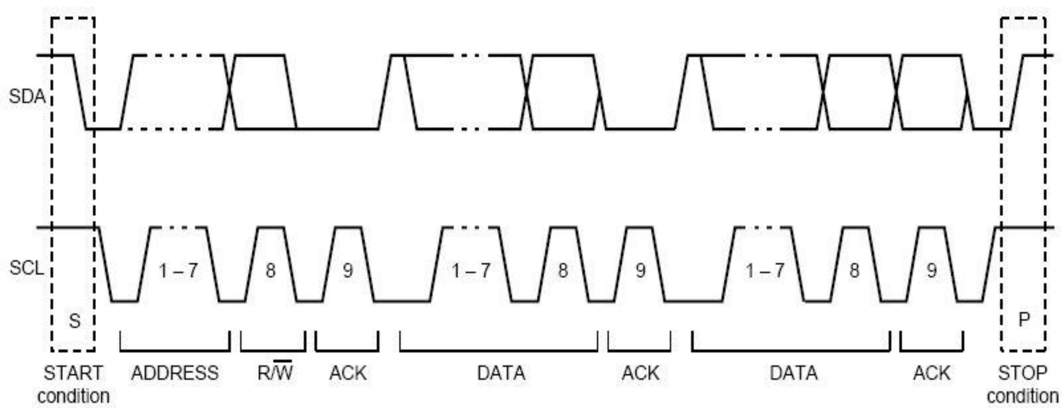


Figure 7: I²C protocol

5.4 Start temperature and pressure measurement

The timing diagrams to start the measurement of the temperature value UT and pressure value UP are shown below. After start condition the master sends the device address write, the register address and the control register data. The BMP180 sends an acknowledgement (ACKS) every 8 data bits when data is received. The master sends a stop condition after the last ACKS.

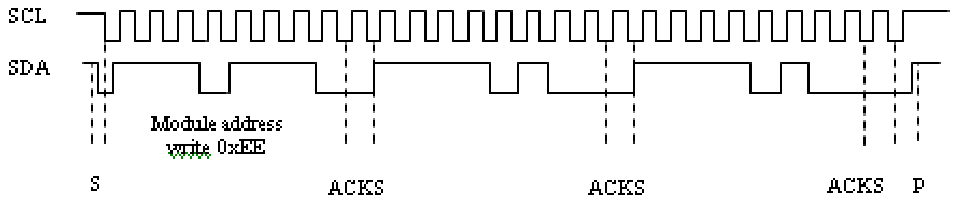


Figure 8: Timing diagram for starting pressure measurement

Abbreviations:

S	Start
P	Stop
ACKS	Acknowledge by Slave
ACKM	Acknowledge by Master
NACKM	Not Acknowledge by Master

Table 8: Control registers values for different internal oversampling_setting (oss)

Measurement	**Control register value (register address 0xF4)**
Temperature	0x2E
Pressure (oss = 0)	0x34
Pressure (oss = 1)	0x74
Pressure (oss = 2)	0xB4
Pressure (oss = 3)	0xF4

6. Package

6.1 Pin configuration

Picture shows the device in top view. Device pins are shown here transparently only for orientation purposes.

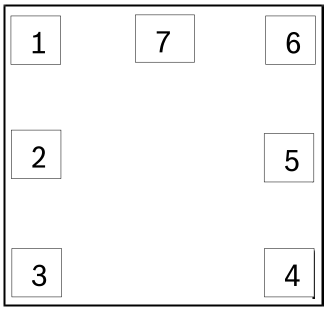


Figure 10: Layout pin configuration BMP180

Table 9: Pin configuration BMP180

in No	Name	Function
1	CSB*	Chip select
2	VDD	Power supply
3	VDDIO	Digital power supply
4	SDO*	SPI output
5	SCL	I2C serial bus clock input
6	SDA	I2C serial bus data (or SPI input)
7	GND	Ground

* A pin compatible product variant with SPI interface is possible upon customer's request. For I^2C (standard case) CSB and SDO are not used, they have to be left open.
All pins have to be soldered to the PCB for symmetrical stress input even though they are not connected internally.