BMP581

Barometric Pressure Sensor

BST-BMP581-DS004-11

Document revision	1.11
Document release date	February 2025
Document number	BST-BMP581-DS004-11
Sales Part Number	0 273 017 025
Notes	Data and descriptions in this document are subject to change without notice. Product photos and pictures are for illustration purposes only and may differ from the real product appearance.

1 Basic Description

The BMP581 is an absolute barometric pressure sensor. Its small dimensions, its low power consumption and the highend performance allow the implementation in a wide range of applications.

Key features:

Pressure operating range: 30 .. 125 kPa
Temperature operating range: -40 .. 85°C
Ultra low noise and current consumption:

- Ultra low noise: < 0.1PaRMS natively (without low-pass filter enabled)
- 1.3 µA 1Hz in low power mode

Absolute pressure accuracy: ± 0.5 hPa (max)
Relative pressure accuracy: 0.06 hPa per 10kPa step
Pressure Temperature-induced offset: ± 0.5 Pa/K
BMP581 provides true absolute pressure and temperature, due to on-chip linearization and temperature-compensation
Primary digital interface with 12 MHz slave SPI (4-wire, 3-wire), 12.5MHz I3C and up to 1MHz I²C (Fm+)
Output data rates:

- up to 480 Hz in CONTINUOUS mode
- 0.125 .. 240 Hz in NORMAL mode

Wide power supply range: VDD 1.71 V ... 3.6 V and VDDIO 1.08 V ... 3.6 V, both independent
Programmable low-pass filtering
On-chip FIFO buffer for up to 32 pressure samples
Programmable interrupts, including pressure-changed detection
6 bytes user programmable non-volatile memory
Compact 10-pin metal-lid LGA package with a footprint of only 2.0 × 2.0 mm2 and max 0.8 mm package height.
RoHS compliant, halogen and lead free

Typical applications:

Enhancement of GPS navigation (e.g. time-to-first-fix improvement, dead-reckoning, slope detection)
Indoor navigation (floor detection, elevator detection)
Outdoor navigation
Sports applications like calorie counting, fitness activity identification
Emergency caller location
Weather forecast
Vertical velocity indication (e.g. rise/sink speed)
Altitude control of drones and flying toys

Target devices:

Handsets such as mobile phones, tablet PCs, GPS devices
Navigation systems
Portable health care devices
Home weather stations
Drones and flying toys
Smart watches
Virtual and augmented reality devices

Table of Contents

List of tables

2 Specification

If not stated otherwise,

- All values are valid over the full voltage range
- Minimum/maximum values are ±3 sigma values
- Typical values of currents and state machine timings are determined at 25 °C
- Minimum/maximum values of currents are valid for the temperature range from -40°C…+85°C
- Minimum/maximum values of timings are valid for the temperature range from -40°C…+85°C
- Environmental conditions like temperature, RF, humidity are constant, unless ranges for these are specified

Pressure performance is given in Table 1, temperature performance in Table 2. If not stated otherwise:

- Parameters are valid for the range 300 – 1100 hPa @ -5 – 65 °C
- Parameters are valid before soldering after storage in standard conditions according to Table 6.
- Performance parameters are derived without MSL1 preconditioning.
- "Post-solder " refers to 3x reflow soldering after storage in standard conditions.

Table 1: Pressure Performance[1]

Parameter	Symbol	Comment	Min	Typ	Max	Unit
Pressure measurement range	P		30		125	kPa
Temperature range	T_A	Pressure measured in the entire temperature operational range	-40		85	°C
Relative pressure accuracy	A_{p_rel}	700 – 1100 hPa, 15 – 55 °C, 10 kPa steps		± 6		Pa
Relative pressure accuracy	A_{p_rel}	900 – 1100 hPa hPa, 25 °C, 1 kPa steps		± 0.4		Pa
Absolute Pressure accuracy	A_{p_abs}	300 – 1100 hPa, -5 – 65 °C, including TCO		± 30		Pa

Absolute accuracy pressure[2] (full range)	$A_{p\_abs\_full\_range}$	300 – 1100 hPa, -40 – 85 °C, incl. solder drift, TCO after qualification[3]		± 75		Pa
Offset temperature coefficient	TCO	300 – 1100 hPa, -5 – 65 °C, 10 K steps		± 0.5		Pa/K
Pressure data resolution	A_{p_res}			1/64		Pa
Pressure noise	$V_{p\_low\_power}$	OSR set to „lowest power“		0.78	0.95	PaRMS
Pressure noise	$V_{p\_hi\_res}$	OSR set to “high resolution”		0.21	0.25	PaRMS
Pressure noise	$V_{p\_high\_res}$	OSR set to „high resolution“		0.08		PaRMS
Long term drift	ΔP_{stab_long}	Drift during 1 year[a]		± 10		Pa
Short term drift	ΔP_{stab_short}	Drift during 24 h at constant pressure and temperature		± 1.5		Pa
Solder Drift		5x reflow soldering after storage in standard conditions (25 °C / 100 kPa)		± 30		Pa

a. HTOLderived from 1000h HTOL divided by 5

[2] Accuracy measured based on BST soldering process, in Bosch lab with dedicated pressure chamber and high-accuracy reference equipment

[3] After 1000rhs HTOL and HTS see qualification report

Table 2: Temperature Performance

Parameter	Symbol	Comment	Min	Typ	Max	Unit
Temperature measurement range		Temperature measured in the entire temperature operational range	-40		85	°C
Absolute accuracy temperature	A_{t_abs}	-5 .. 55°C		+/- 0.5		K
Absolute accuracy temperature full range	$A_{t\_abs\_full}$	-40 .. 85°C		+/- 0.5		K
Relative accuracy temperature	$A_{t\_rel\_full}$	-40 .. 85°C, 20 K steps				K
Temperature data resolution	A_{t_res}			1/65536	+ 1.0	°C

Table 3: Mechanical characteristics

Parameter	Symbol	Comment	Min	Typ	Max	Unit
Package footprint dimensions			1.90 x 1.90	2.0 x 2.0	2.10 x 2.10	mm^2
Package height			0.7	0.75	0.8	mm
Number of pins				10		
Package category		Moisture sensitivity level		MSL1		

Table 4: Electrical characteristics

Parameter	Symbol	Comment	Min	Typ	Max	Unit
Power supply voltage	VDD		1.71	1.8 / 3.3	3.6	V
Power supply voltage I/Os	VDDIO		1.08	1.2/ 1.8 / 3.3	3.6	V
Supply ramp time	t_VDDramp & t_VDDIOramp	10% to 90% of target voltage	0.01[a]		10	ms
Operational temperature range	TOP_full		-40		85	°C
ODR Accuracy		-40..85°C	-10		+10	%
Power-up time	t_powup	Time to first communication after both VDD > VDDmin and			2	ms

		VDDIO > VDDIOmin				
Start-up time from SLEEP	t_startup	Time from mode change to start of measurement			3	ms
Start-up time from DEEPSLEEP	$t_{startup\_deep}$	Time from mode change to start of measurement			4	ms
Re-configuration time	t_{reconf}	Time from configuration change in NORMAL or CONTINUOUS mode to start of first measurement			3	ms
Re-configuration time	t_{reconf_deep}	Time from configuration change in NORMAL or CONTINUOUS mode to start of first measurement			4	ms
Time to standby	$t_{standby}$	Time from any mode to STANDBY			2.5	ms
Soft reset duration	t_{soft_res}	Time from trigger of soft reset until device ready			2	ms
I²C interface clock	f_{i2c}	Normal Mode & Fast Mode @ Cbus < 550pF - Fast Mode + @Cbus < 100pF	100		1000	kHz
I3C interface clock	f_{i3C}	@ VDDIO > 1.62 V - Normal Mode & Fast Mode @ Cbus < 550pF - Fast Mode + @Cbus > 100pF	0.1	12.5	12.9	MHz
I3C interface clock low voltage	f_{i3C_lowv}	@ VDDIO < 1.62 V - Normal Mode & Fast Mode @ Cbus < 550pF - Fast Mode + @Cbus > 100pF	0.1		2.5	MHz
SPI interface clock	f_{spi}	@ VDDIO ≥ 1.62 V, Cbus ≤ 80 pF 4-wire/ 3-wire; modes 0 and 3			12	MHz
SPI interface clock low voltage	f_{spi_lowv}	@ VDDIO ≤ 1.62 V, Cbus ≤ 40 pF 4-wire/ 3-wire; modes 0 and 3			7	MHz

Peak current	i_{peak}	maximum DC current			260	µA
Deep Standby current		25 °C and VDDIO=VDD=1.8 V		0.55		µA
Deep Standby current		25 °C and VDDIO=VDD=3.6 V		0.55		µA
Deep Standby current		55 °C and VDDIO=VDD=1.8 V		1.5		µA
Standby current		25 °C and VDDIO=VDD=1.8 V		1.0		µA
Standby current		55 °C and VDDIO=VDD=1.8 V		3.5		µA
Current consumption low power mode		OSR set to „lowest power“ Low Power Mode ODR=1 Hz 25 °C		1.3		µA
Current consumption high resolution		OSR set to „high resolution“ ODR = 30 Hz 25°C		75	80	µA
INT pulse length	t_{int_pulse}	Pulse length in pulsed mode	90	105	121	µs
INT minimum deassert time	$t_{int\_deassert}$	Minimum time between INT pin assert	90	105	121	µs
Maximum output rate (ODR)		in CONTINOUS mode, pressure and temperature measured		489		Hz
Output data rate (ODR) range		in NORMAL mode	0.125		240	Hz
Conversion time pressure	t_{conv_p}	OSR = 1x	-5%	1.0	+5%	ms
		OSR = 2x	-5%	1.7	+5%	ms
		OSR = 4x	-5%	2.9	+5%	ms
		OSR = 8x	-5%	5.4	+5%	ms
		OSR = 16x	-5%	10.4	+5%	ms
		OSR = 32x	-5%	20.4	+5%	ms
		OSR = 64x	-5%	40.4	+5%	ms
		OSR = 128x	-5%	80.4	+5%	ms
Conversion time temperature	t_{conv_t}	OSR = 1x	-5%	1.0	+5%	ms
		OSR = 2x	-5%	1.1	+5%	ms
		OSR = 4x	-5%	1.5	+5%	ms
		OSR = 8x	-5%	2.1	+5%	ms
		OSR = 16x	-5%	3.3	+5%	ms
		OSR = 32x	-5%	5.8	+5%	ms
		OSR = 64x	-5%	10.8	+5%	ms
		OSR = 128x	-5%	20.8	+5%	ms

NVM user write cycles[b]	N_{NVM_WRITE}	Number of write cycles to NVM user range			10,000	writes

a. For supply ramps < 0.01 ms, a 10 Ohm resistor must be connected in series to the power supply (see 5.2.5).
b. power supply must be stabe during the write sequency. Temperature must be in the range of 0 - 65 °C.

Table 5: Interface pin electrical characteristics

Parameter	Symbol	Comment	Min	Typ	Max	Unit
Input low voltage	V_{IL}	@VDDIO=1.2V/ 1.8V/3.3V+/-10%			30	%
Input high voltage	V_{IH}	@VDDIO=1.2V/ 1.8V/3.3V+/-10%	70			%
Input voltage hysteresis	V_{IHYST}	@VDDIO=1.2V/ 1.8V/3.3V+/-10%	10			%
Output low voltage	V_{OL}	@VDDIO=1.2V/ 1.8V/3.3V+/-10%			20	%
Output high voltage	V_{OH}	@VDDIO=1.2V/ 1.8V/3.3V+/-10%	80			%
Pull-up resistance at CSB pin	R_{PU_CSB}	I2C mode, relevant for interface mode selection	74	100	131	kOhm

Table 6: Absolute maximum ratings

Parameter	Symbol	Comment	Min	Max	Unit
Storage temperature[a]		≤ 65 % r.h. Standard conditions: + 25°C and 40% r.h.	-40	+125	°C
Supply voltage VDD			-0.3	4.3	V
Supply voltage VDDIO			-0.3	4.3	V
Max Voltage at I/O Pins			VSSIO-0.3 V	VDDIO + 0.3 V	V

a. Storage should occur at standard conditions. For short time periods, the device may be stored outside of this range, but must stay within above mentioned limits.

Stress above those listed as “absolute maximum ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

3 Quick start guide

This section describes quickly the steps to get the sensor running with an example configuration.

3.1 Sensor API and COINES

An Application Programming Interface (API) called Sensor API is available for BMP581. It is available as C source code. The API provides higher-level functions, for example to switch power modes, or read and write the NVM. It is an abstraction layer, so that the user does not have to issue individual read and write transactions to sensor registers. The API still allows direct low-level register access to the sensor. The Sensor API also provides some basic examples of its usage.
The Sensor API is fully compatible with the COINES library, which provides the low-level functions for the sensor API. It is included in the COINES software package. More information, can be found on https://www.bosch-sensortec.com/.

4 Functional Description and Features

The BMP581 is a barometric pressure sensor that outputs to the host the absolute pressure in Pa. In addition, the absolute temperature in °C can be provided to the host.

4.1 Block diagram

BMP581‘s key components are a pressure sensitive MEMS sensor element and an integrated circuit that drives and reads out the sensor element. Also it provides data and other functions to the host. The block diagram is shown in Figure 1.

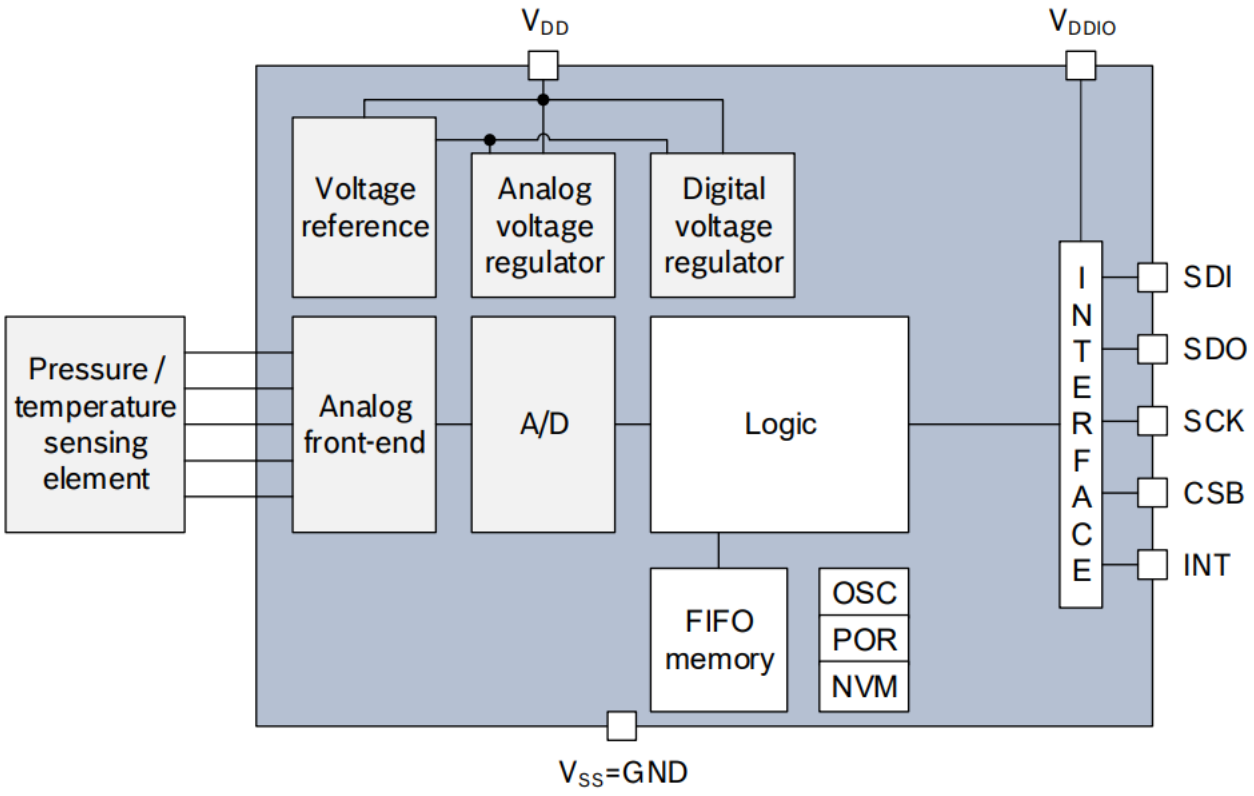


Figure 1: BMP581 block diagram

4.2 Power management

The BMP581 has two separate power supply pins:

- VDD is the main power supply for all internal analog and digital regulator blocks
- VDDIO is a separate power supply pin, used for the supply of the digital interface

VDD and VDDIO pins can be energized in any order. A power-on reset generator is built in which resets the logic circuitry and the register values after the power-on sequence. The slope for ramp up time must be within the limits given by t_- VDDramp and t_VDDIOramp. After powering up, the sensor settles in sleep mode (see also 3.3.9").

Holding any interface pin (SDI, SDO, SCK or CSB) at a logical high level when VDDIO is switched off can permanently damage the device due caused by excessive current flow through the ESD protection diodes.

4.3 Power modes

The power modes of BMP581 and transitions in between are depicted in Figure 2. After startup or soft-reset, the BMP581 will be in DEEP STANDBY mode. Transitions from one mode to another are only possible by entering SLEEP mode first.

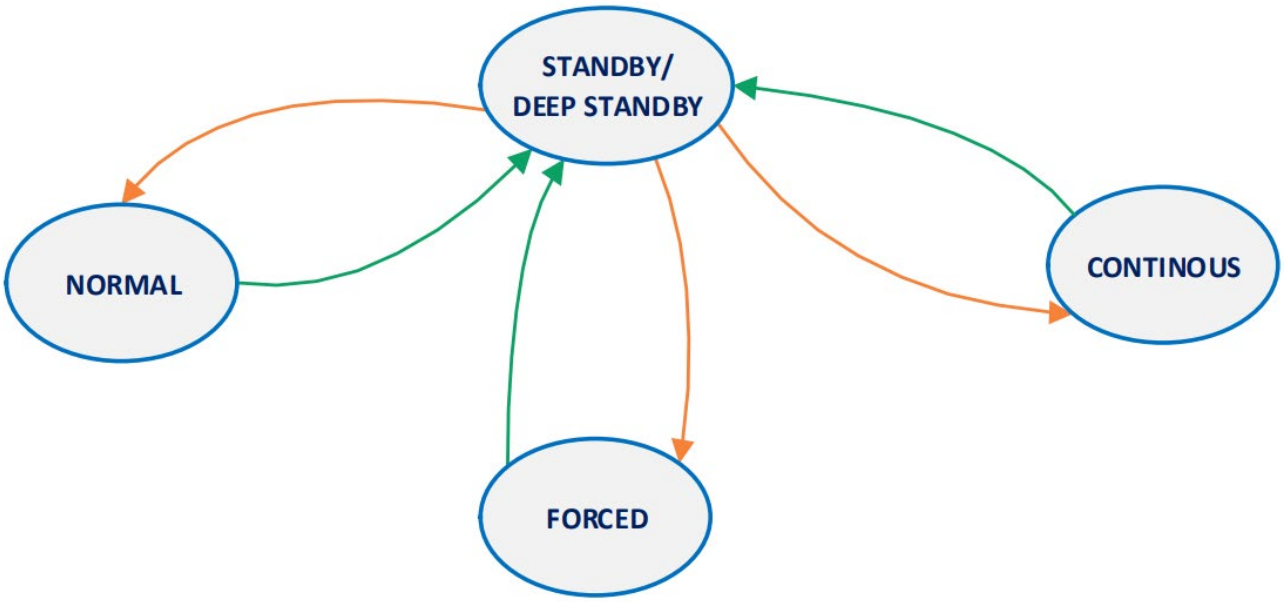


Figure 2: Power modes diagram

4.3.1 STANDBY mode

In STANDBY mode, no measurements are performed and power consumption is at a minimum. All registers are accessible for write and read. Mode transitions to other modes are possible. The pressure and temperature data registers PRESS_DATA_XXX and TEMP_DATA_XXX keep the values of the last measurement executed. The FIFO. if enabled, also maintains it‘s content and can be read.

4.3.2 DEEP STANDBY

In order to further reduce the power consumption further, the BMP581 offers a DEEP_STANDBY mode. In this case, power consumption is even lower than in STANDBY mode.
DEEP_STANDBY will only be entered if also the following conditions apply:

- ODR_CONFIG.deep_dis = 0
- ODR_CONFIG.odr < 5Hz
- FIFO_SEL.fifo_frame_sel = DIS
- DSP_IIR.set_iir_t = BYPASS
- DSP_IIR.set_iir_p = BYPASS

If one of these settings is changed, the BMP581 transitions to STANDBY mode.

4.3.3 FORCED mode

In FORCED mode, a single measurement is performed according to selected measurement and filter options. When the measurement is finished, the sensor returns to sleep mode and the measurement results can be obtained from the data registers. For a next measurement, forced mode needs to be selected again. Forced mode is recommended for applications which require very low sampling rate or host-based synchronization. Forced mode may also be used if an ODR higher than 240 Hz is needed.

4.3.4 NORMAL mode

Normal mode performs pressure measurements with a configurable frequency, which is the output data rate (ODR). The ODR can be set in ODR_CONFIG.odr. Normal mode continuously cycles between an (active) measurement period and an (inactive) standby period. In normal mode, the most recent measurement result can be directly obtained from the data registers. Alternatively, the latest measurement results can be obtained from the FIFO.

Normal mode is recommended if pressure needs to be sampled in regular intervals, but none of the conditions apply where FORCED and CONTINOUS mode may be favorable.

The ODR_CONFIG.odr register field is used to define the output data rate (ODR) if the BMP581. Data rates of 0.125Hz up to 240 can be selected. For the full list of available ODRs, see the register description. Not all combinations of OSR and ODRs are valid, as measurement times may not fit into an ODR cycle. Table 7: maximum nominal ODR setting per OSR settings in NORMAL modeshows the maximum ODR for a given ODR setting.

Table 7: maximum nominal ODR setting per OSR settings in NORMAL mode

max ODR [Hz]		**OSR_T**							
		1	2	4	8	16	32	64	128
OSR_P	1	240.00	240.00	240.00	240.00	200.00	130.00	80.00	40.00
	2	240.00	240.00	240.00	220.00	180.00	120.00	70.00	40.00
	4	220.00	220.00	200.00	180.00	140.00	100.00	70.00	40.00
	8	140.00	140.00	130.00	120.00	100.00	80.00	50.00	35.00
	16	80.00	80.00	80.00	70.00	70.00	50.00	45.00	30.00
	32	45.00	45.00	40.00	40.00	40.00	35.00	30.00	20.00
	64	20.00	20.00	20.00	20.00	20.00	20.00	15.00	15.00
	128	10.00	10.00	10.00	10.00	10.00	10.00	10.00	5.00

Table 8: Maximum nominal ODR setting per OSR settings in NORMAL mode for temperature only measurements

max ODR [Hz]	**OSR_T**							
	1	2	4	8	16	32	64	128
	240.00	240.00	240.00	240.00	200.00	130.00	80.00	40.00

Configuration Check. BMP581 has an automatic configuration checking, which is functional in NORMAL mode and when both temperature and pressure measurements are enabled. If a configuration is not valid, this will be indicated by the OSR_EFF.odr_is_valid register field. If a measurement with an invalid setting is started, the BMP581 will run with the specified ODR, but use a default setting for the ODRs:

- For ODRs ≥ 160Hz, both OSRs will be set to 1
- For all ODRs <160Hz, both OSRs will be set to 2

The effective ODRs are available in the register fields OSR_EFF.osr_t_eff and OSR_EFF.osr_p_eff. This action of alignment and check is done in NORMAL mode only.

4.3.5 Low Power NORMAL mode

If the conditions for deep standby apply, as described in 3.3.1 above, then NORMAL mode will automatically apply DEEP_STANDY phase in between the measurements. This reduces power consumption even further. If one of these settings is changed, the BMP581 transitions back to NORMAL mode.

4.3.6 CONTINOUS mode

Continuous mode performs pressure measurements simular to NORMAL mode. However, the ODR setting is ignored. Sampling is performed with the maximum frequency that is possible with the selected oversampling settings. CONTINOUS mode stays in the (active) measurement period and does not cycle to a standby period. The resulting

ODR is not necessarily a value that is selectable via the ODR register. The resulting ODRs for the recommended OSR settings are shown in Table 9.

4.3.7 Mode transitions

To go in STANDBY status the user must write ODR_CONFIG.pwr_mode = 0b00. The maximum transition time to STANDBY is $t_{standby}$. The effective status of the device is always observable reading back the same register. After a commanded switch to standby, the user either needs to wait for $t_{standby}$ or check the status register for a successful switch, before he can command the device to go to another mode, and before writing to any of the registers named in 3.3.8.

From STANDBY, it can be switched to CONTINUOS, FORCED or NORMAL mode by writing ODR_CONFIG.pwr_mode register. Directly after the transition to an active mode, the first measurement will be performed. It is recommended to set the desired measurement configuration, before switching the mode.

4.3.8 Mode-depending register write restrictions

A number of registers and register fields can only be updated when the device is in STANDBY mode. These are for example the registers for NVM operations (see 3.8), but also configuration registers for the FIFO and IIR configuration. The register descriptions state if this limitation applies to a register field. Write operations to these registers in a mode other than STANDBY are lost. It is generally recommended to write configurations before switching into the measurement mode.

4.3.9 Post-power-up procedure

After power up of the BMP581, it is available after t_powerup. The host should not initiate any communication with the BMP581 before. Depending on the interface configuration, a dummy read should be the first access to the device (see 4.1).

It is recommended that the host checks the following status registers after a power-up:

- read out the CHIP_ID register and check that it is not all 0
- read out the STATUS register and check that status_nvm_rdy==1, status_nvm_err == 0
- read out the INT_STATUS.por register field and check that it is set to 1; that means INT_STATUS==0x10

4.3.10 Soft reset

BMP581 can be reset by writing 0xB6 to the CMD register. The BMP581 will come out of the reset after $t_{soft\_res.}$ Softreset must not be triggered during a NVM user programming sequence.

4.4 Measurements

4.4.1 Pressure and temperature measurement enable

The BMP581 can either measure temperature only, or both temperature and pressure. Pressure-only measurement is not supported, as temperature data is needed for the temperature compensation of the pressure data.[4]

Pressure and temperature will be measured if any of these conditions is true:

- OSR_CONFIG.press_en ==1, or
- FIFO_SEL.fifo_frame_sel == 0b10, or
- FIFO_SEL.fifo_frame_sel == 0b11

If none of these settings is made, the sensor will measure temperature only.

[4] However, the sensor can be configured to output the pressure only data to the FIFO, see Chapter 3.6.1

4.4.2 Pressure and temperature oversampling ratio (OSR)

Oversampling extends the measurement time per measurement by the oversampling factor. Higher oversampling factors offer decreased noise at the cost of higher power consumption.

Oversampling can be set individually for pressure and temperature in register fields OSR_CONFIG.osr_p and OSR_-CONFIG.osr_t. The duration of the sampling phase, is given by t_{conv_p} and t_{conv_t}. Table 7: maximum nominal ODR setting per OSR settings in NORMAL mode shows the maximum ODR for each oversampling setting. Recommended settings are shown in Table 9: Oversampling settings .

Table 9: Oversampling settings

Oversampling setting	*osr_p*	Pressure oversampling	Temperature oversampling	Typical pressure RMS noise at 100kPa	Typical ODR in CONTINUOUS mode
Lowest power	000	×1	×1	0.78 Pa	498 Hz
	001	×2	×1	0.58 Pa	374 Hz
Standard resolution	010	×4	×1	0.41 Pa	255 Hz
	011	×8	×1	0.30 Pa	155 Hz
High resolution	100	×16	×1	0.21 Pa	87 Hz
	101	×32	×2	0.15 Pa	46 Hz
	110	×64	×4	0.11 Pa	24 Hz
Highest resolution	111	×128	×8	0.08 Pa	12 Hz

Note: The noise values refer to the sensor-intrinsic noise. Already at standard resolution, the noise or fluctuations of the air pressure itself may be higher than the noise of the sensor, and thus be dominant. This ambient noise is typically stronger at lower frequencies. Any increase of the ODR does not reduce this type of noise, because this frequency range is of interest for many applications and thus is not attenuated by the sensor. If low frequency noise is a problem in a use case, it is recommended to employ low pass filtering, for example by using the build-in IIR-filter.

4.4.3 Configuration changes in NORMAL and CONTINUOUS mode

If any of these changes is applied during NORMAL and CONTINUOUS mode:

- OSR_CONFIG.press_en
- OSR_CONFIG.osr_t
- OSR_CONFIG.osr_p
- ODR_CONFIG.odr (NORMAL mode only)

Measurements will restart with the new settings after t_{reconf}. If DEEP_SLEEP is enabled in NORMAL mode, it will start after t_{reconf_deep}.

4.4.4 IIR filter

The BMP581 has a dedicated IIR filter built-in, that can be used to reduce noise caused by ambient disturbances. This may for example be the opening of doors or windows, or wind blowing into the sensor. To suppress these disturbances in the output data, the IIR filter can be enabled.

Please note that IIR filtering, like all low pass filtering, also reduces the bandwidth of the signal.

The filter function is the following:

$$data_n = \frac{data_{n-1} \times filtercoefficient + data_{in}}{filtercoefficient + 1}$$

where $data_{n-1}$ is the filtered data from the previous acquisition, and $data_{in}$ is the unfiltered data from the current acquisition.

The step response of different filter settings is displayed in Figure 3 and Figure 4. Table 9 shows the available filter coefficent settings and the according normalized bandwidth (which corresponds to the 3dB cutoff frequency). The resulting bandwidth in Hz can be computed as follows:

$$bandwidth_{Hz} = ODR_{Hz}\ x\ bandwidth_{Hz}$$

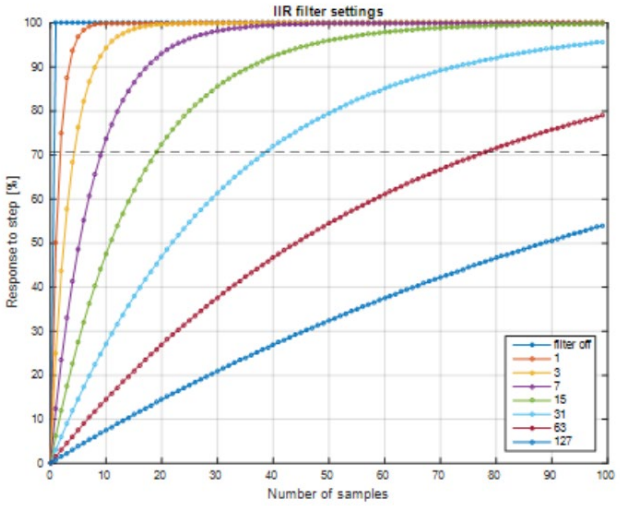


Figure 3: Step response at different IIR filter settings

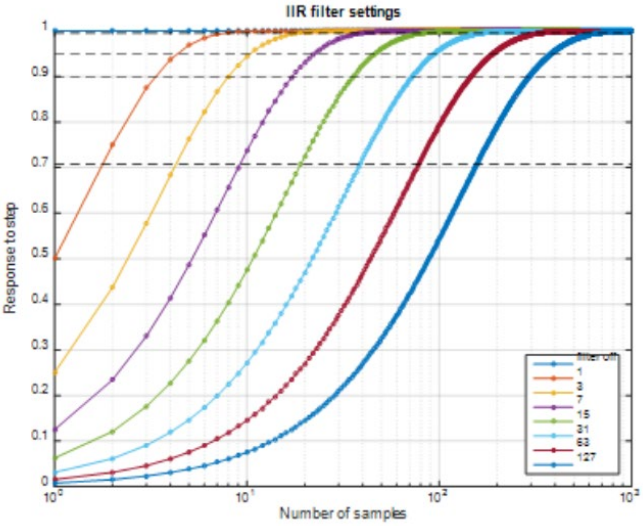


Figure 4: Step response at different IIR filter settings on log scale and different threshold limits in dashed lines (3dB, 90%, 95%, 99.5%, 100%)

Table 10: IIR filter settings and bandwidth

Register value	Filter coefficient	Normalized bandwidth (-3db cutoff frequency)
0x0	0	Bypass
0x1	1	0.1147
0x2	3	0.0459
0x3	7	0.0212
0x4	15	0.01025
0x5	31	0.005041
0x6	63	0.00250
0x7	127	0.00125

The IIR filter can be independently programmed for temperature and pressure with the fields DSP_IIR.set_iir_t and DSP_IIR.set_iir_p. The value “0x0” is used to bypass the filter. The IIR filter is reset when a transition from sleep mode to NORMAL or CONTINUOUS mode occurs. That means that the first measurment value will be the initial content of the filter. A manual reset of the filter, e.g. when using forced mode, can be triggered by asserting the register field DSP_- CONFIG.iir_flush_forced_en.

- shdw_sel_iir_t: select source for temperature data register
- fifo_sel_iir_t: select source for temperature FIFO
- shdw_sel_iir_p: select source for pressure data register
- fifo_sel_iir_p: select source for pressure FIFO
- oor_sel_iir_p: select source for pressure out-of-range interrupt

4.5 Data registers

Data from the most recent measurement is present in data registers. The DSP_CONFIG.swdw_sel_iir_t and DSP_CONFIG.swdw_sel_iir_t select if IIR-filtered data or unfiltered data is presented in the data registers.

Temperature data is contained in the registers TEMP_DATA_MSB, TEMP_DATA_LSB, TEMP_DATA_XLSB. The registers shall be interpreted in the following way:

$$T\,[°C] = \left(\frac{\text{TEMP\_DATA\_MSB,TEMP\_DATA\_LSB,TEMP\_DATA\_XLSB}}{2^{16}}\right)$$

Pressure data is contained in the registers PRESS_DATA_MSB, PRESS_DATA_LSB, PRESS_DATA_XLSB. The registers shall be interpreted in the following way:

$$p\,[Pa] = \left(\frac{\text{PRESS\_DATA\_MSB,PRESS\_DATA\_LSB,PRESS\_DATA\_XLSB}}{2^{6}}\right)$$

In both equations, the divisions can be implemented by a simple and efficient bit-wise right shift operation. To read out data after a conversion, it is strongly recommended to use a burst read and not address every register individually.

4.5.1 Data Shadowing

In normal mode, measurement timing is not necessarily synchronized to readout. This means that new measurement results may become available while the user is reading the results from the previous measurement. In this case, shadowing is performed in order to guarantee data consistency. Shadowing will only work if all data registers are read in a single burst read. Therefore, the user must use burst reads if he does not synchronize data readout with the measurement cycle. Using several independent read commands may result in inconsistent data.

If a new measurement is finished and the data registers are still being read, the new measurement results are transferred into shadow data registers. The content of shadow registers is transferred into data registers as soon as the user ends the burst read, even if not all data registers were read. Reading across several data registers can therefore only be guaranteed to be consistent within one measurement cycle if a single burst read command is used. After the end of the burst read, all user data registers are updated at once with the shadowed data.

4.6 FIFO

The BMP581 contains a first-in first-out (FIFO) data buffer. Pressure and temperature data is stored in the FIFO in frames. Each frame contains the data from one measurement. The maximum number of frames depends on which data is stored in the FIFO:

- 16 frames if both pressure and temperature are stored
- 32 frames if only pressure or temperature is stored

4.6.1 FIFO Configuration

The FIFO frame type is selected by FIFO_SEL.fifo_frame_sel:

- 0b00: FIFO not enabled
- 0b01: Only Temperature data is stored (T-mode)

- 0b10: Only Pressure data is stored (P-mode)
- 0b11: Pressure and temperature data is stored (PT-mode)

The operational mode can be controlled via the FIFO_CONFIG.cfg_fifo_mode register:

- 1'b0: streaming mode
- 1'b1: stop on full mode

The two modes differ in how the FIFO reacts to an overflow. A FIFO overflow occurs if the FIFO is full and a new measurement data is ready to be written to the FIFO. In streaming mode, the FIFO will delete the oldest frame, and write the new frame to the FIFO. As a result, the FIFO contains always the most recent frames.[5]

In stop-on-full mode, frames once written to the FIFO will not be discarded. Instead, new frames will not be written to the FIFO until there is space again.

The FIFO decimation factor (or downsampling) can be adjusted With FIFO_SEL.cfg_fifo_dec_sel. Only every n-th sample will be written to the FIFO, where:

$$n = 2^{FIFO\_SEL.cfg\_fifo\_dec\_sel}$$

The FIFO threshold can be set by FIFO_CONFIG_fifo_threshold. If the fill level of the FIFO reaches the threshold, the FIFO threshold interrupt may be triggered (see Chapter 3.7.2.1). The meaning of the register field is the following:

- 0x00: FIFO threshold disablef
- 0x01..0x1F (or 1..31 decimal): threshold level
- 0x0F: 15 frames. This is the maximum setting in PT-mode. The most significant bit is ignored.
- 0x1F: 31 frames. This is the maximum setting in P- or T-mode.

4.6.2 FIFO status

The fill level of the FIFO in number of frames can be obtained from FIFO_COUNT.fifo_count. FIFO watermark and FIFO full information can be obtain from the interrupt functionality (see Chapter 3.7.2.1).

4.6.3 FIFO data readout

The FIFO can be read out by reading in a burst from register FIFO_DATA.

Reads should be performed in the granularity of the frame size (24 or 48 bit) according to the selected frame type. Frames that have been read incompletely will stay in the FIFO memory and will be retransmitted on the next read. The entire FIFO contents can be read in one single burst.

If the FIFO is empty, disabled or turns empty during a read, it will return the empty frame, which is 0x7f.

Table 10,

Table 11 and Table 13 show the frame formats for the three different frame kinds: PT, T and P. The empty frame is shown in Table 13.

Table 11: FIFO pressure and temperature frame (PT-fram

	7	6	5	4	3	2	1	0
Temperature	temperature XLSB							
	temperature LSB							
	temperature MSB							
Pressure	pressure XLSB							
	press LSB							

2. In order to work properly, streaming mode requires that the clock frequency of host interface is 0.1MHz or above. Otherwise, the FIFO readout bandwidth could be slower than the FIFO write bandwidth, which will cause data loss.

	press MSB

Table 12: FIFO temperature frame (T-frame)

	7	6	5	4	3	2	1	0
Temperature	temperature XLSB							
	temperature LSB							
	temperature MSB							

Table 13: FIFO pressure frame (P-frame)

	7	6	5	4	3	2	1	0
Pressure	pressure XLSB							
	press LSB							
	press MSB							

Table 14: FIFO empty frame

	7	6	5	4	3	2	1	0
Empty	0x7F							

4.6.4 FIFO configuration changes

The FIFO is flushed on any of the following conditions:

- a change in the sensor configuration:
  - OSR_CONFIG.osr_t
  - OSR_CONFIG.osr_p
  - ODR_CONFIG.odr
  - ODR_CONFIG.pwr_mode
- a change in the frame configuration:
  - FIFO_SEL.fifo_frame_sel
  - FIFO_SEL.cfg_fifo_dec_sel

The flush will empty the FIFO, reset the FIFO_COUNT register, and clear the interrupt conditions.
The completion of the FIFO flush is finished within t_{reconf}, or t_{reconf_deep} if the device is in deep sleep.The FIFO_COUNT should not be read before the flush has been finished, as the result may be inconsistent.
If the register FIFO_CONFIG_fifo_threshold is written, the resulting interrupt status bits will be immediately updated according to the new threshold.
The register FIFO_SEL must only be changed in STANDBY mode.
A change of FIFO_SEL.cfg_fifo_dec_sel during NORMAL or CONTINUOUS mode will only be applied after a transition to STANDBY. It will also not flush the FIFO.
A change of FIFO_SEL.fifo_frame_sel during NORMAL or CONTINOUS mode may be ignored as well, depending on if the press_en bit is also changed.

4.7 Interrupts

The BMP581 provides an interrupt pin (INT), which allows to signal certain events to the host processor. Different events can be mapped to the interrupt pin, which all are processed with a logical OR.
BMP581 also supports I3C‘s in-band interrupt (IBI). This is allows the use of interrupt functionality without the need of a dedicated INT signal line. For documentation of the I3C IBI functionality, see Chapter 4.7.2 "I3C In-band Interrupts".

The available interrupts are listed below, and will be detailed in following subsections:

- FIFO watermark interrupt
- FIFO full interrupt
- Data ready interrupt
- Pressure out-of-range interrupt
- Power-on reset (POR) interrupt

4.7.1 Interrupt enabling

The individual interrupts sources can be enabled in the INT_SOURCE register. An exception is the POR interrupt, which is always enabled. With enabled interrupt sources:

- their individual status is available from the INT_STATUS register,
- I3C in-band interrupts can be used (see Chapter 4.7.2 "I3C In-band Interrupts"), and
- the interrupt pin can be used, see Chapter 3.7.3.

4.7.2 Interrupt sources

4.7.2.1 FIFO interrupts

The FIFO provides two sources of interrupts:

- FIFO full: The fill level is at the maximum number of frames. This means 16 PT frames or 32 P or T frames, depending on the configuration of the FIFO.
- FIFO threshold reached: The fill level is at or above the FIFO threshold level (see Chapter 3.6.1).

Both interrupts will be asserted at the end of a measurement (when data is ready), when the respective condition is fulfilled. They will stay asserted as long as the corresponding condition is active. A read of the INT_STATUS register will not change the FIFO interrupts. FIFO interrupts can only occur if the FIFO is enabled.

If a burst read from the FIFO causes the fill level to drop below the fill level that causes an interrupt, the interrupt will be deasserted at the end of the burst read.

The FIFO interrupts can be enabled by setting INT_SOURCE.fifo_full_en and INT_SOURCE.fifo_ths_en.

4.7.2.2 Data ready interrupt

The data ready interrupt and status register INT_STATUS.drdy_data_reg is asserted when new pressure and/or temperature is available in the data registers (see Chapter 3.5 "Data registers"). Also, the new measurement data is available in the FIFO after the data ready interrupt.

The interrupt can be enabled by setting INT_SOURCE.drdy_data_reg_en.

4.7.2.3 Out-of-range interrupt

The out-of-range (OOR) interrupt is triggered when the pressure value is outside a defined range for a defined number of samples.

The benefit of this interrupt is that the host system does not need to read the sensor data continously to detect of there is a significant change of the measured pressure. Instead the host can configure the interrupt, und read sensor data only if the interrupt triggered.

For the OOR interrupt, the BMP581 checks if the pressure value is within a window around a reference pressure. The reference pressure can be defined in [Pa] with a width of 17 bit, which covers the complete measurement range of the sensor. The reference values can be written by access the register fields OOR_CONFIG.oor_thr_p_16, OOR_THR_P_MSB.oor_thr_p_15_8 and OOR_THR_P_LSB.oor_thr_p_7_0.

The range is also given in [Pa] and can be defined via the register OOR_RANGE.oor_range_p. As the register has a width of 8bit, the range can span up to +/- 255 Pa around the reference value.

The OOR is out of range if observed pressure P_Pa in is:

$$P\_Pa > reference + window$$

or

$$P\_Pa \ < referenc - window$$

If one of these conditions is satisfied for the number of samples defined by OOR_CONFIG.cnt_lim, the interrupt will be triggered.

If subsequent measurements are still out of range, the interrupt will be re-triggered after each of those measurements.

Example. Assumed the user wants to get an interrupt if the pressure is outside the range of 97100 Pa - 97200 Pa. In this case, the reference should be set to the middle value 97150 Pa, which is 0x17B7E. The window value is half of the range, which is 50 Pa, or 0x32. This means that the registers need to be set to the following values:

- OOR_CONFIG.oor_thr_p_16 = 0x1
- OOR_THR_P_MSB.oor_thr_p_15_8 = 0x7B
- OOR_THR_P_LSB.oor_thr_p_7_0 = 0x7E
- OOR_RANGE.oor_range_p = 0x32

4.7.2.4 Power-on reset interrupt

The power-on rest (POR) interrupt is triggered each time the BMP581 comes out of a power-up reset. This can happen if the supply to the device is ramped up, or if the supply was so instable that the BMP581 performed a brown-out with subsequent power-up reset. The POR interrupt signals that the BMP581 is ready to use.

POR interrupts are not supported with I3C IBI, as the device is not in a state where I3C is initialized after power-up reset. Also, the interrupt pin will not flag an POR interrupt, as the interrupt pin is disabled after power-up.
The status of the interrupt can be read from INT_STATUS.por. A read of the INT_STATUS will clear the status.

4.7.3 Interrupt pin

The BMP581 provides an interrupt pin (INT), which allows to signal certain events to the host processor.

4.7.3.1 Interrupt pin configuration

The behavior of the interrupt pin can be configured in INT_CONFIG with these fields:

- int_mode: The interrupt mode can be „pulsed“ or „latched“. Latching determines when an interrupt is released (see Chapter 3.7.3.2 for details)
  int_pol: The interrupt polarity can be configured to be either „active high“ or „active low“
  int_od: The interrupt pin can be configured to be „open-drain“ or push-pull“
- int_en: The interrupt pin can be enabled. With enabled interrupt pin, all interrupt sources configured in INT_-SOURCE will be ORed on the interrupt pin.

4.7.3.2 Interrupt Timings

Interrupt timings depend strongly on the int_mode setting:

Pulsed mode. In the pulsed mode the INT pin creates a pulse on the interrupt pin, each time an interrupt condition changes from FALSE to TRUE, and the interrupt source is enabled in INT_SOURCE. Figure 5 shows the timing of pulsed mode.
The pulse length is t_{int_pulse}. Between two pulses, there is a minimum gap of $t_{int\_deassert}$ in which the pin will stay de-asserted.

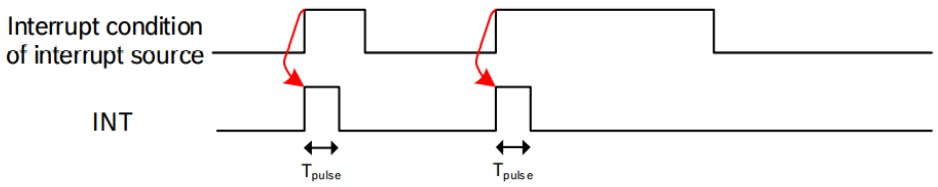


Figure 5: INT pin timing in pulsed mode

Latched Mode. In latched mode, the INT pin is asserted as long as an interrupt condition is TRUE, and the interrupt source is enabled in INT_SOURCE. Between two adjacent assertions if the INT pin, there is a minimum gap of $t_{int\_deassert}$. Figure 6 shows the timing of latched mode.
The deassertion of the INT pin in latched mode depends is handled in the following way:

- FIFO interrupts will be de-asserted when the interrupt condition does not apply any more. There is no dependency on the setting of INT_STATUS.
- The data ready interrupt will be de-assered after reading the INT_STATUS.
- The pressure out-of-range interrupt will be de-assered after reading the INT_STATUS. If the data ready interrupt is asserted, a new measurement data becomes available, the INT pin will stay asserted. There is no de-assertion phase.

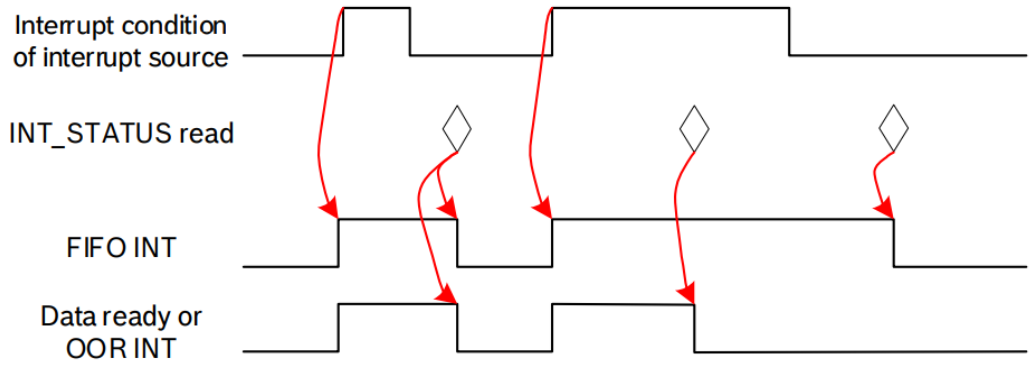


Figure 6: INT pin timing in latched mode

Exceptions. In the following cases, the minimal pulse length and minimal gap between pulses may be violated:

- If the FIFO gets disabled or is flushed (see Chapter 3.6.4 "FIFO configuration changes" for a description of the conditions that cause a FIFO flush), an asserted FIFO interrupt will be de-asserted immediately and the corresponding bits in INT_STATUS will be cleared. This may cause a violation of t_{int_pulse} or $t_{int\_deassert}$.
- If the conditons apply that would cause a FIFO flush, the behavior of an asserted out-of-range interrupt is the same as for the FIFO interrupt desribed above: the asserted interrupt will be de-asserted immediately and the corresponding bits in INT_STATUS will be cleared. This may cause a violation of t_{int_pulse} or $t_{int\_deassert}$.
- If the host reconfigures the FIFO threshold while INT is asserted, INT will get de-asserted immediately and the INT_STATUS.fifo_ths will get cleared. This may cause a violation of t_{int_pulse}. If the new FIFO threshold condition still holds true, INT will reassert after $t_{int\_deassert}$.
- If data ready and FIFO interrupts are used together, $t_{int\_deassert}$ may be violated.
- $t_{int\_deassert}$ can be violated if FIFO interrupts are enabled at the same time with the data ready or the out-of-range interrupt. There is no violations when data ready and out-of-range interrupts are enabled at the same time.
- **Latched/pulsed mode switch**. Any change between latched/pulsed mode has to be applied while interrupt is disabled. The following operations must be executed:
- Turn off all INT sources (INT_SOURCE -> 0x00)
- Read the INT_STATUS register to clear the status
- Set the desired mode in INT_CONFIG.int_mode

INT_STATUS. Independently of the int_mode setting, the interrupt status bit in INT_STATUS will not be cleared automatically. The FIFO status will be cleared only when the interrupt condtion does not apply any more and the INT_STATUS register has been read. The data ready and the out-of-range status will be cleared when the INT_STATUS register has been read.

FIFO threshold interrupt during FIFO read. Interrupt generation is not blocked during an ongoing FIFO read. If the fill level drops below the threshold during a FIFO read, and reaches the threshold again (due to a new sample being wrtten to the FIFO), the interrupt will be asserted. If such behavior is not wanted, it can be avoided by any of the following strategies:

- Use non-latched interrupts, and ignore the interrupt during read.
- Read-out the FIFO fast enough that the fill level is 2 frames below the watermark level before the next sample is taken (which occurs ~1/ODR seconds after the interrupt assertion).

4.8 NVM Programmability

The BMP581 contains a non volatile memory (NVM) that contains trimming and configuration parameters that are used internaly by the sensor. In addition, there is a user range.

User write to the memory is restricted to the NVM User Range. User read can also be performed to the other NVM addresses.

4.8.1 NVM User Range

The host can write and read the memory of the user range. The range is located at addresses 0x20-0x22. Each address holds 2 bytes. This memory area may be used for an end-of-line trim at OEM or ODM sites. The maximum number of writes during the lifetime of BMP581 is specified by N_{NVM_WRITE}. During the write procedure, the power supply to the BMP581 must be stable, and no soft-reset must be issued. Otherwise, permanent damage to the device may occur.

In order to read or write the entire user range, the read/write procedure has to be executed repeatedly for the three addresses 0x20-0x22. Reads always follow the procedure given in chapter 3.8.1.2.

4.8.1.1 NVM Read procedure

- Switch to STANDBY mode by writing ODR_CONFIG.pwr_mode and ensuring that DEEP STANDBY is disabled[6]
- Wait until STATUS.nvm_rdy is equal to 1
- Write the NVM_ADDR register, with nvm_row_address containing the address to read, and nvm_prog_en set to 0
  Write the USR_READ sequence (0x5D, 0xA5) into to CMD register. All write transactions to NVM_ADDR and USR_READ must be individual transactions, and must not be combined in burst writes.
- Wait until STATUS.nvm_rdy is equal to 1. This takes approximately 200 µs
- Read the data from the NVM_DATA_MSB and NVM_DATA_LSB registers
- Check for errors in STATUS.nvm_err, STATUS.nvm_cmd_err. Read data will not be valid if one of the error flags is set

4.8.1.2 NVM Write procedure

- Switch to STANDBY mode by writing ODR_CONFIG.pwr_mode and ensuring that DEEP STANDBY is disabled[7]
- Wait until STATUS.nvm_rdy is equal to 1
- Write the NVM register, with nvm_row_address containing the address to write, and nvm_prog_en set to 1
- Write the data to be programmed to NVM_DATA_MSB and NVM_DATA_LSB
  Write the USR_PROG sequence (0x5D, 0xA0) into the CMD register. All write transactions to NVM_ADDR, NVM_-DATA_MSB and NVM_DATA_LSB and USR_READ must be individual transactions, and must not be combined in burst writes.
- Wait untill STATUS.nvm_rdy is equal to 1.This takes approximately 10 ms
- Check for errors in STATUS.nvm_err, STATUS.nvm_cmd_err. The write was not successfully performed if one of the error flags is set
- Reset NVM_ADDR.nvm_prog_en to 0

[6] DEEP STANDBY is disabled when at least one of the conditions described in Chapter 4.3.2 is not fulfilled
[7] DEEP STANDBY is disabled when at least one of the conditions described in Chapter 4.3.2 is not fulfilled

Writes to other NVM addresses than the user range will be ignored.

4.8.1.3 UID

The unique device identifier is also stored in the NVM. It can be computed as follows:
UID = ((read_nvm_reg(0x26) & 0xFF) << 40) | (read_nvm_reg(0x25) << 24) | (
read_nvm_reg(0x24) << 8) | ((read_nvm_reg(0x23) & 0xFF00) >> 8)
where read_nvm_reg refers to the NVM read procedure to the given address.

4.8.1.4 NVM CRC check

Integrity of the NVM can be checked using the CRC. The CRC is calculated based content of all memory except for the aforementioned user range at final test, and except for the error correction bits at 0x1E. That means, the CRC is calculated on the addresses 0x00 - 0x1D, 0x1F and 0x23 -0x26.The result is written to the CRC NVM address 0x27. The user can check the integrity of the NVM by repeating the calculation and comparing against the value in address 0x27. If there is a mismatch, the NVM has been altered or corrupted since final test.
The CRC-16 is calculated with CCITT-16, selected polynomial is: x16 + x12 + x5 + 1. The initial content of the register used to compute the remainder of the division is preset to 0xFFFF. Refer also to: ITU - T Recommendation X.25 (10/96).

Following C-code sample shows the computation of the CRC-16 CCITT checksum:

```
int datalength = 0;
int datawidth = 0;
int SIZE = 62;
uint16_t Poly = 0x1021;
uint16_t Initval = 0xFFFF;
uint16_t crc ;
uint16_t crc_temp;
crc = Initval;
for( datalength =0; datalength< SIZE; datalength++){
crc_temp = (OTP_DATA[datalength] << 8) ^ crc;
for (datawidth = 0; datawidth < 8; datawidth++)
{
if ((crc_temp & 0x8000) != 0)
{
crc_temp = (crc_temp << 1) ^ Poly;
}
else
{
crc_temp = crc_temp << 1;
}
}
crc = crc_temp;
}
```

4.9 Final test result

The final test result (good / bad part) can be obtained from the NVM, see Table 15.

Table 15: Final test result

NVM Reg	Designition NVM-Reg	Bit-Number NVM	Parameter	Value
0x06	trim_rev_id	<15..13>	Bad part	0
0x06	trim_rev_id	<15..13>	Trim_ID	≥1

5 Digital Interface

The device provides one serial interface to the host. It acts as a slave to the host The serial interface is configurable to the interface protocols SPI, I3C and I2C.

5.1 Protocol Selection

The protocol is automatically selected based on the behavior of the signal on the chip select pin CSB after power-up. After soft reset or power-up, the primary interface of the device is in I²C/I3C mode. If the CSB is connected to VDDIO during power-up and not changed, the primary interface works in I²C or I3C mode.
For using I²C and I3C, it is recommended to hard-wire the CSB line to VDDIO. Since power-on-reset is only executed when both VDD and VDDIO are stable, there is no risk of an incorrect protocol detection due to the power-up sequence.

Once the CSB input pin is falling low, the I²C/I3C mode is disabled. The HIF switches over to SPI mode if there are at least 16 full serial clock (SCK) edges during the CSB low phase, and CSB has risen again. Hence, it is recommended to perform a single read via SPI of a registers (e.g. to CHIP_ID) before the actual SPI communication with the device. Note: the content of the retrieved data will be invalid.

The switch from I²C to I3C follows the MIPI I3C specification. Upon power up, the chip stays in I²C mode and once the dedicated Broadcast I3C Address (7'h7E) is seen on the bus, the chip will disable its I2C feature and the interface stays in the I3C mode until a soft reset or the next power-up occurs.

The possibles switches among the modes on the digital interface are summarized in Table 15.

Table 16: Possible switches between interface modes

Protocol switch	to I²C	to I3C	to SPI
from I²C	--	Device ID 7E sent	dummy SPI read
from I3C	power-down or soft-reset	--	dummy SPI read
from SPI	power-down or soft-reset	power-down or soft-reset	--

5.2 Interface timing

The general interface parameters are given in the table below.

Table 17: General interface parameters

Parameter	Comment	Symbol	Unit	Min	Typ	Max
Input Low Voltage	@VDDIO=1.2V/1.8V/3.3V+/-10%	V_IL	%			30
Input High Voltage	@VDDIO=1.2V/1.8V/3.3V+/-10%	V_IH	%	70		
Input Voltage Hysteresis	@VDDIO=1.2V/1.8V/3.3V+/-10%	V_IHYST	%	10		
Output Low Voltage	@VDDIO=1.2V/1.8V/3.3V+/-10%	V_OL	%			20
Output High Voltage	@VDDIO=1.2V/1.8V/3.3V+/-10%	V_OH	%	80		
Pad Input Leakage Current (no pull-R)	Input = 'Low'	I_IL	µA			1
Pad Input Leakage Current (no pull-R)	Input = 'High'	I_IH	µA			1
Pull-up resistance at CSB pin	I2C mode, relevant for interface mode selection	R_PU_CSB	kΩ	74	100	131

5.2.1 Interface timing

The timing diagram for SPI is given in Figure 7 and is valid for all SPI configurations. The corresponding values are given in

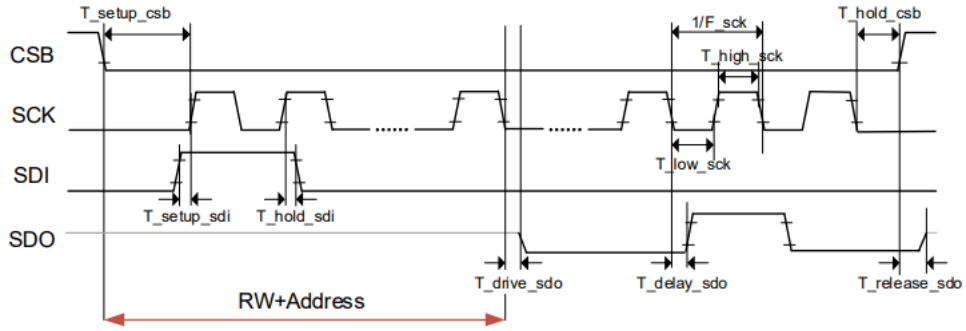


Figure 7: SPI timing diagram (SPI4, Mode 0

Table 18: SPI timings

Parameter	Comment	Symbol	Unit	Target value		
				min	typ	max
CSB lag time		T_hold_csb	ns	40		
SDX setup time		T_setup_sdx	ns	19		
SDX hold time		T_hold_sdx	ns	7		
SCL to SDO turnaround time	90%/10% Master rise/fall time = [2, 10]ns @ Cbus = 40pF, drive_strength = 7	T_delay_scl2sdo_1p2	ns			52.5
	90%/10% Master rise/fall time = [2, 10]ns @ Cbus = 80pF, drive_strength = 7	T_delay_scl2sdo_1p8	ns			27.8
	90%/10% Master rise/fall time = [2, 10]ns @ Cbus = 80pF, drive_strength = 6	T_delay_scl2sdo_3p3	ns			20.2
SDO rise/fall time (90%/ 10%)	@ VDDIO = 1.2V		ns			tbd
	@ VDDIO = 1.8V/3.3V		ns			10
SCX frequency	@ VDDIO = 1.8 V/3.3 V	F_sck,nv	MHz	1		12
SCX Pulse High Time	5% duty cycle variation	T_high_scx	ns	37.5		
SCX Pulse Low Time		T_low_scx	ns	37.5		
Idle time after write access		T_wr_idle, spi	ns	80		
Idle time after read access		T_rd_idle,spi	ns	80		

5.2.2 I2C timing specifications

BMP581 follows the I²C specification for standard mode, fast mode and fast mode plus. For timing specifications, please consult the „I2C-bus specification and user manual“, UM10204, Rev.6, NXP Semiconductors. This is valid for the entire VDDIO voltage range.

5.2.3 I3C timing specifications

BMP581 follows the I3C specification for SDR mode. For timing specifications, please consult the „ Specification for I3C Version 1.1 “ and the corresponding errata document „ Errata 01 for MIPI I3C Specification “, both from the Mipi Alliance. For VDDIO < 1.62V, there are deviations from the specification, which are summarized in Table 18.

Table 19: I3C timing limitations for VDDIO < 1.62 V

Parameter	Symbol	Condition	Min	Typ	Max	Units
SCL clock frequency	fSCL	VDDIO < 1.62V	0.1		2.5	MHz
Fall time of SDA signal	tFDA				35	ns
SDA Signal Data Setup in Push-Pull Mode	t_{SU_PP}		5			Ns
Clock in to Data Out for Slave	t_{SCO}				215	
Fall Time of SDA Signal	t_{fDA_OD}				35	ns

5.3 Pad drive stength

The drive strenghts to drive a pad to logical high (IOH, Vout=20%*VDDIO) or to logical low(IOL, Vout=80%*VDDIO) are shown in Table 17 and Table 18.

Table 20: Drive strength in IOH

VDDIO	DRIVE_STRENGTH	PARAMETER	Current over T, process corners and ±10% VDDIO			Current at 25°C, typical process, and ±10%VDDIO	
			Typ	Min	Max	Min(V) @RT;TT	Max(V) @RT; TT
1.2	0	Ioh	305u	90.98u	575.90u	195.4u	426.9u
	1	Ioh	903.80u	270.20u	1.71m	579.7u	1.265m
	2	Ioh	1.50m	448.70u	2.83m	962.1u	2.098m
	3	Ioh	2.10m	627.90u	3.96m	1.346m	2.936m
	4	Ioh	2.70m	808.30u	5.10m	1.733m	3.78m
	5	Ioh	3.30m	987.50u	6.22m	2.118m	4.618m
	6	Ioh	3.90m	1.17m	7.35m	2.5m	5.451m
	7	Ioh	4.50m	1.34m	8.48m	2.884m	6.289m
1.8	0	Ioh	1m	600.90u	1.64m	772.1u	1.242m
	1	Ioh	2.96m	1.78m	4.86m	2.286m	3.676m
	2	Ioh	4.91m	2.95m	8.06m	3.792m	6.097m
	3	Ioh	6.88m	4.13m	11.28m	5.306m	8.532m
	4	Ioh	8.85m	5.32m	14.52m	6.832m	10.99m
	5	Ioh	10.82m	6.50m	17.74m	8.346m	13.42m
	6	Ioh	12.77m	7.67m	20.94m	9.852m	15.84m
	7	Ioh	14.73m	8.85m	24.16m	11.37m	18.28m
3.3	0	Ioh	3.24m	2.22m	4.72m	2.714m	3.769m
	1	Ioh	9.58m	6.57m	13.96m	8.033m	11.16m
	2	Ioh	15.89m	10.90m	23.16m	13.32m	18.51m
	3	Ioh	22.24m	15.25m	32.40m	18.64m	25.9m
	4	Ioh	28.64m	19.64m	41.73m	24.01m	33.35m
	5	Ioh	34.98m	23.99m	50.97m	29.33m	40.74m
	6	Ioh	41.29m	28.32m	60.17m	34.62m	48.08m
	7	Ioh	47.64m	32.67m	69.41m	39.94m	55.47m

Table 21: Drive strengths for IOL

VDDIO	DRIVE_STRENGTH	PARAMETER	Current over T, process corners and ±10% VDDIO			Current at 25°C, typical process, and ±10%VDDIO	
			Typ	Min (PVT)	Max (PVT)	MIN (V) @RT;TT	MAX (V) @ RT;TT
1.2	0	Iol	356.20u	99.57u	838u	200.9u	537.1u
	1	Iol	1.11m	291.90u	2.48m	639.4u	1.64m
	2	Iol	1.87m	488.80u	4.13m	1.085m	2.755m
	3	Iol	2.62m	681.10u	5.77m	1.524m	3.858m
	4	Iol	3.32m	846.60u	7.32m	1.921m	4.893m
	5	Iol	4.07m	1.04m	8.96m	2.359m	5.997m
	6	Iol	4.84m	1.24m	10.61m	2.805m	7.111m
	7	Iol	5.59m	1.43m	12.25m	3.244m	8.215m
	8	Iol	6.42m	1.66m	14.02m	3.738m	9.41m
	9	Iol	7.17m	1.85m	15.66m	4.176m	10.51m
	10	Iol	7.93m	2.05m	17.32m	4.622m	11.63m
	11	Iol	8.68m	2.24m	18.96m	5.06m	12.73m
	12	Iol	9.38m	2.40m	20.50m	5.458m	13.77m
	13	Iol	10.13m	2.60m	22.14m	5.896m	14.87m
	14	Iol	10.89m	2.79m	23.80m	6.342m	15.98m
	15	Iol	11.65m	2.99m	25.44m	6.78m	17.09m
1.8	0	Iol	1.42m	722.10u	2.73m	1.064m	1.787m
	1	Iol	4.08m	2.14m	7.41m	3.125m	5.073m
	2	Iol	6.77m	3.58m	12.11m	5.204m	8.383m
	3	Iol	9.44m	5m	16.79m	7.265m	11.67m
	4	Iol	11.99m	6.35m	21.30m	9.228m	14.82m
	5	Iol	14.66m	7.77m	25.98m	11.29m	18.11m
	6	Iol	17.35m	9.20m	30.68m	13.37m	21.42m
	7	Iol	20.01m	10.62m	35.36m	15.43m	24.7m
	8	Iol	22.84m	12.14m	40.26m	17.62m	28.17m
	9	Iol	25.51m	13.56m	44.94m	19.68m	31.46m
	10	Iol	28.19m	15m	49.65m	21.76m	34.77m
	11	Iol	30.86m	16.42m	54.33m	23.82m	38.05m
	12	Iol	33.41m	17.77m	58.83m	25.79m	41.2m
	13	Iol	36.08m	19.19m	63.51m	27.85m	44.49m
	14	Iol	38.77m	20.62m	68.22m	29.93m	47.8m
	15	Iol	41.43m	22.04m	72.89m	31.99m	51.09m

VDDIO	DRIVE_STRENGTH	PARAMETER	Current over T, process corners and ±10% VDDIO			Current at 25°C, typical process, and ±10%VDDIO	
			Typ	Min (PVT)	Max (PVT)	MIN (V) @RT;TT	MAX (V) @ RT;TT
3.3	0	Iol	4.67m	2.84m	7.55m	3.953m	5.359m
	1	Iol	12.49m	7.82m	19.44m	10.67m	14.27m
	2	Iol	20.36m	12.82m	31.38m	17.43m	23.22m
	3	Iol	28.19m	17.80m	43.27m	24.14m	32.13m
	4	Iol	35.76m	22.60m	54.82m	30.63m	40.75m
	5	Iol	43.59m	27.57m	66.71m	37.35m	49.66m
	6	Iol	51.46m	32.58m	78.65m	44.11m	58.62m
	7	Iol	59.29m	37.56m	90.54m	50.82m	67.53m
	8	Iol	67.46m	42.79m	102.90m	57.95m	76.82m
	9	Iol	75.29m	47.76m	114.80m	64.56m	85.72m
	10	Iol	83.16m	52.77m	126.70m	71.32m	94.68m
	11	Iol	90.98m	57.75m	138.60m	78.04m	103.6m
	12	Iol	98.56m	62.54m	150.10m	84.53m	112.2m
	13	Iol	106.40m	67.52m	162m	91.24m	121.1m
	14	Iol	114.30m	72.53m	174m	98m	130.1m
	15	Iol	122.10m	77.51m	185.90m	104.7m	139m

5.4 Read burst address increment

For read bursts in all protocols, the BMP581 performs an automatic address increment with each read byte. That means, if the user reads for example 10 bytes starting address 0x01, the BMP581 will return the data for register 0x01..0x0A. An exception to this rule is the FIFO_DATA register. If a read starts at FIFO_DATA, the address will not be incremented, but the read will continue on the register to support FIFO read-out. The same applies if the FIFO_DATA register is addressed during a read burst that started with an address below the FIFO_DATA register. For more inforation on FIFO read-out, see Chapter 3.6.3 "FIFO data readout".

5.5 SPI Protocol

The SPI interface is compatible with SPI mode '00' (CPOL = CPHA = '0') and mode '11' (CPOL = CPHA = '1'). The automatic selection between mode '00' and '11' is determined by the value of SCK after the CSB falling edge.

The SPI interface has two modes: 4-wire and 3-wire. The protocol is the same for both. The 3-wire mode is selected by setting DRIVE_CONFIG.spi3_en = 1. The pad SDI is used as a data input/output pad in 3-wire mode.

Table 19 shows the usage of pins for the SPI protocol. MOSI refers to **M**aster-**O**ut, **S**lave-**I**n data direction. MISO refers to the **S**lave-**I**n, **M**aster-**O**ut data direction.

Table 22: SPI interface pin usage

Name	Description	Function in 4-wire mode	Function in 3-wire mode
CSB	chip select, active low	chip select, active low	chip select, active low
SCK	serial clock	serial clock	serial clock
SDX	serial data in/output	MOSI data	MOSI and MISO data
SDO	serial data output	MISO data	--

Refer to Chapter 5 "Pin out and connection diagrams" for connection instructions.

Data on SDX is latched by the device at SCK rising edge and SDO is changed at SCK falling edge. Communication starts when CSB goes to low and stops when CSB goes to high; during these transitions on CSB, SCK must be stable. The SPI protocol is shown in the following subsections.

5.5.1 SPI3 Wire Mode

SDX must be left floating in SPI3 mode. The reason is that the device starts in SPI4 mode after power-up, and drives SDX until the switch to SPI3 is commanded.

SDI must always be tied to either low or high voltage and not left floating, even when the CSB is high, and no communication with the device takes place. A floating SDI may create excessive power consumption (and on the longterm potentially also damage to the device).

5.5.2 SPI Write Operation

SPI write operation supports single-byte as well as multi-byte (burst) writing. Figure 8 shows the SPI single-byte write protocol. The host sends the write command, write address, write data, and then terminates the transaction.

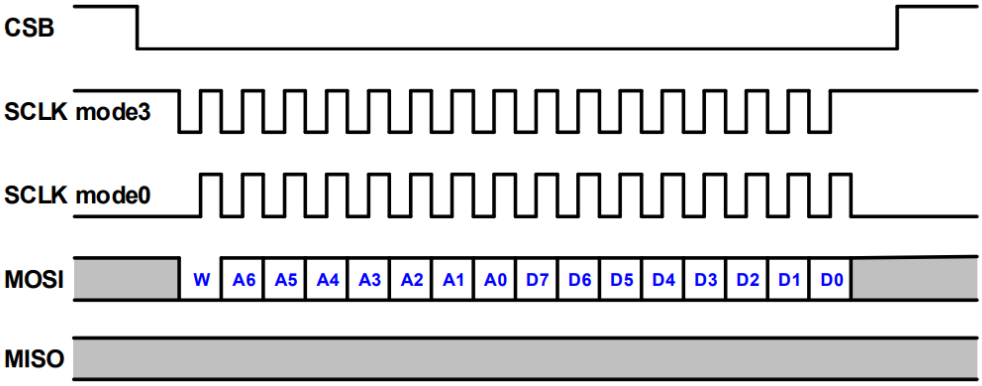


Figure 8: SPI single-byte write operation

shows the SPI multi-byte (burst) write protocol. The host sends the write command, multiple pairs of write address/data and finally terminates the transaction. Note that for each write byte the address has to be sent over separately.

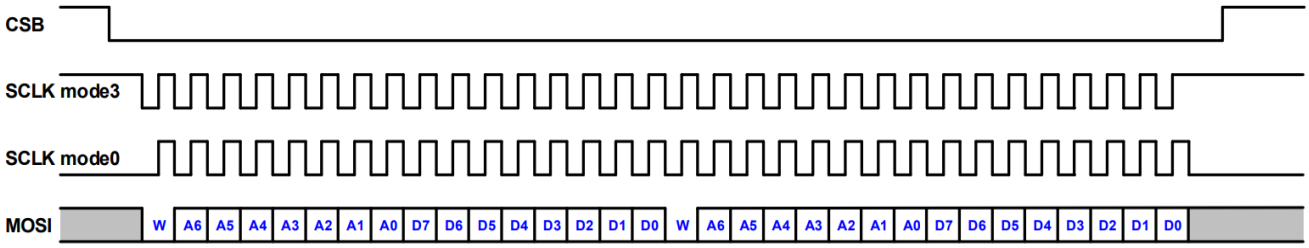


Figure 9: SPI burst write operation

5.5.3 SPI read operation

SPI read operation supports single-byte as well as multi-byte (burst) reading. Figure 10 shows the SPI single-byte read protocol. The host sends the read command, read address, gets the read data, and then terminates the transaction.

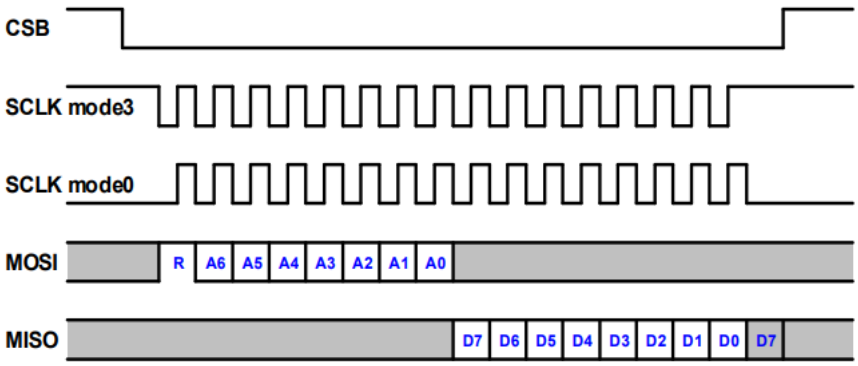


Figure 10: SPI single-byte read operation

Figure 11shows the SPI multi-byte (burst) read protocol. The host sends the read command, read address, gets multiple byte read data, and then terminates the transaction.

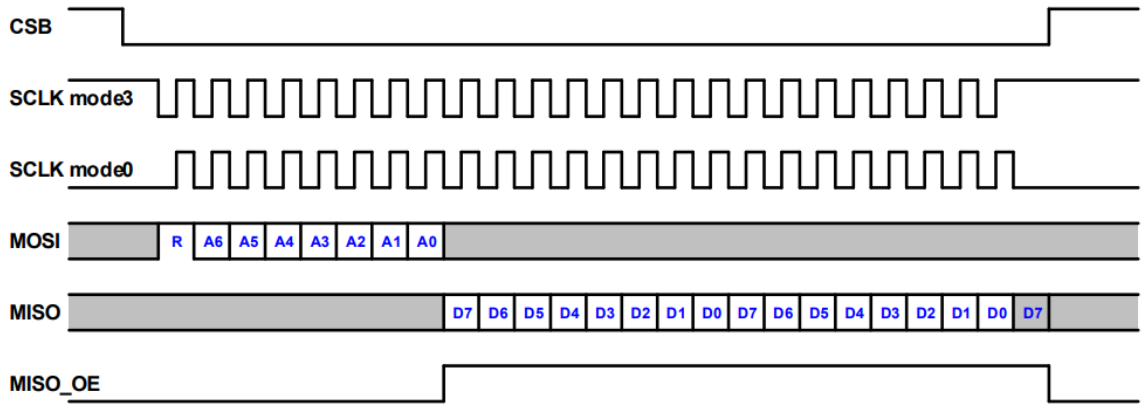


Figure 11: SPI multi-byte read operation

5.5.4 SPI hybrid bursts

SPI also supports a combined write-read operation called hybrid burst. Figure 12 shows this protocol. The host may decide to combine a read (single or burst) transaction together with a write (single or burst) transaction together.

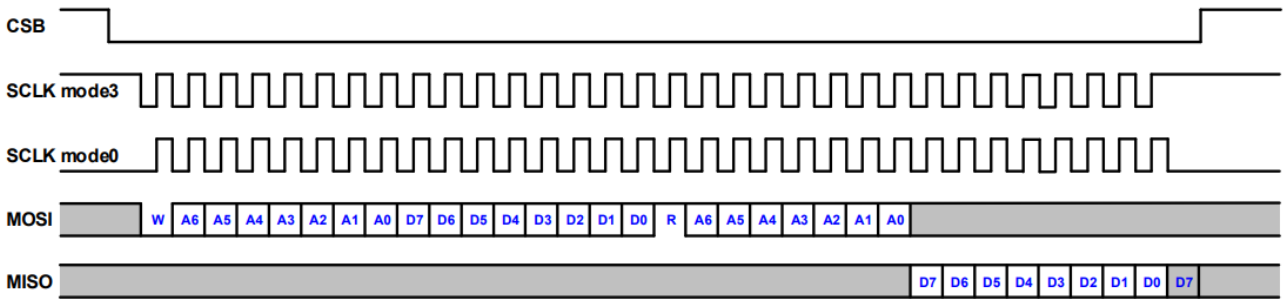


Figure 12: SPI hybrid write-read burst

A CSB idle time of 1us must be ensured for reads following writes for the following registers:

- DRIVE_CONFIG
- INT_CONFIG
- NVM_DATA_LSB
- NVM_DATA_MSB

As a consequence, hybrid accesses on these registers are not allowed.

5.6 I²C protocol

BMP581 supports the following I²C modes:

- normal mode (100 kHz)
- fast mode (100 - 400 kHz)
- fast mode plus (400 kHz - 1 MHz)

The I²C slave address of BMP581 is 7'h46 for SDO = 1'b0 and 7'h47 for SDO = 1'b1. SDO must not be floating when I²C is used, otherwise the I²C device address is undefined.

CSB has an integrated pull-up resistor, which can be enabled in I²C and I3C mode by setting DRIVE_CONFIG.i2c_csb_pup_en.

5.6.1 I²C write operation

I²C write operation supports single-byte as well as multi-byte (burst) writing. Figure 13 depicts the I²C write transfer for single-byte write operation. The transfer begins with a start condition generated by the host, followed by 7 bit slave address and a write bit (R/W = 1'b0). The slave sends an acknowledge bit (ACK = 1'b0) and releases the bus. Subsequently the host is expected to send the register address. Only the first 7 bit (right aligned) are the valid address bit and the MSB is ignored. The slave shall again acknowledge the transmission and wait for the 8 bit data, which shall be written to the specified register address. After slave acknowledges the data byte, the host generates a stop signal and terminates the writing protocol.

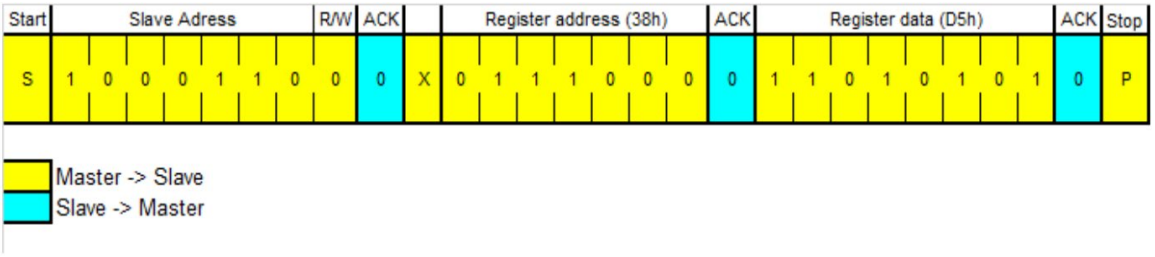


Figure 13: I²C single-byte write

BMP581 also supports multi-byte write operation in I2C mode. The multi-byte write telegram is depicted in Figure 14. The telegram begins with a start condition generated by the host, followed by 7 bit slave address and a write bit (R/W = 1'b0). The slave sends an acknowledge bit (ACK = 1'b0) and releases the bus. Subsequently the host sends the one byte register address. Only 7 bit (right aligned) are the valid address bits and the MSB shall be ignored. The slave shall again acknowledge the transmission and wait for several 8 bit wide data words. The first data word is written to the specified register address. The register address pointer is automatically incremented for each data word. Each received data word is written to the register referenced by the current register address pointer. The slave acknowledges each data byte. When no more data words need to be written, the host generates a stop signal and terminates the writing protocol.

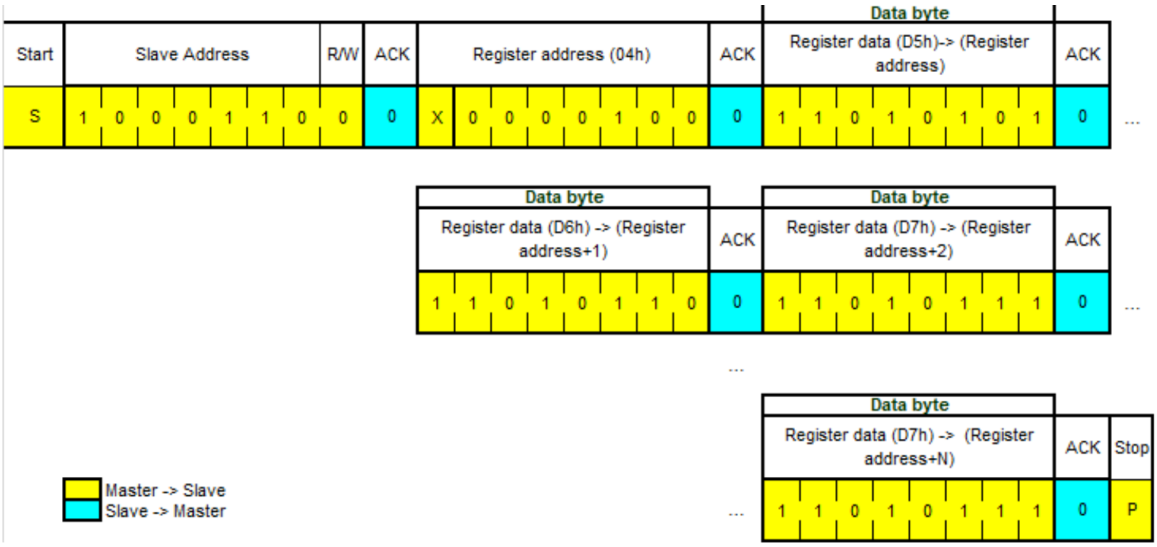


Figure 14: I²C multi-byte write

5.6.2 I²C read operation

I²C read operation supports single-byte as well as multi-byte (burst) reading. A read command consists of a 1 byte I²C write phase followed by an I²C read phase. The two I²C transmissions must be separated by a repeated start condition (Sr) as shown in Figure 15 or a stop followed by start condition (P followed by S) as shown in Figure 16. The I²C write phase addresses the slave and sends the register address to be read. After the slave acknowledges the transmission, the host is expected to generate a start condition and then to send the slave address together with a read bit (R/W = 1'b1). Then the host releases the bus and waits for the data bytes to be read out from slave. After each data byte the host has to generate an acknowledge bit (ACK = 1'b0) to enable further data transfer. A NACK (ACK = 1'b1) from the host stops the data transferring from slave. Slave releases the bus so that the host can generate a STOP condition and terminate the transmission. During a mutli-byte read transfer, the register address is automatically incremented such that more than one byte can be sequentially read out. Once a new data read transmission starts, the start address is set to the register address specified in the latest I²C write command (see Figure 17). By default the start address is set at 8h'00.

In this way repetitive multi-byte reads from the same starting address are possible.

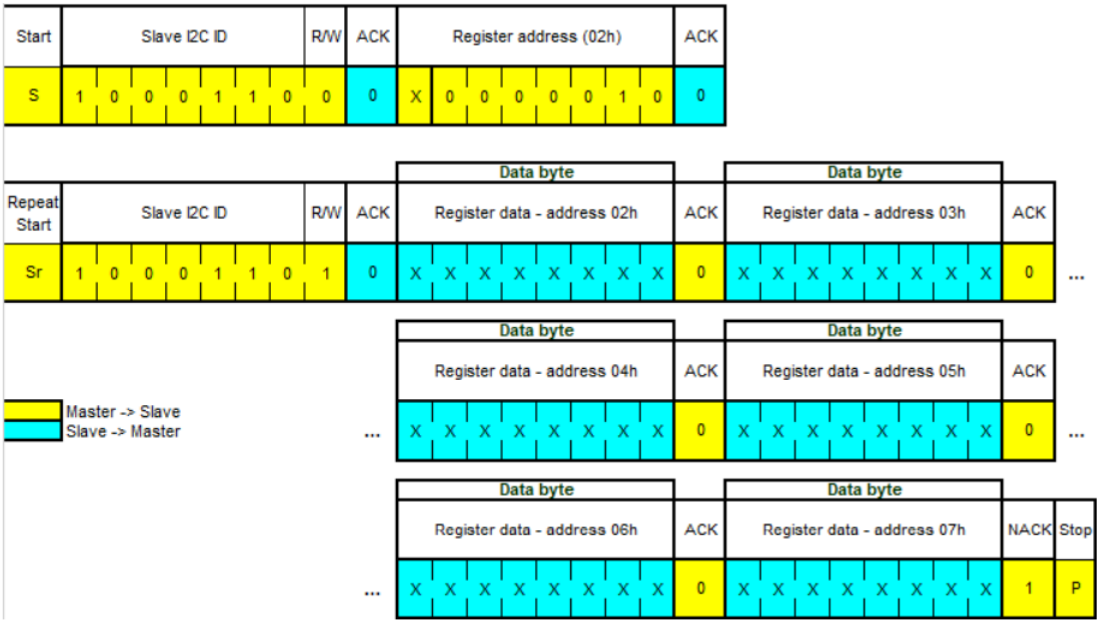


Figure 15: I²C multi-byte read protocol with repeated start

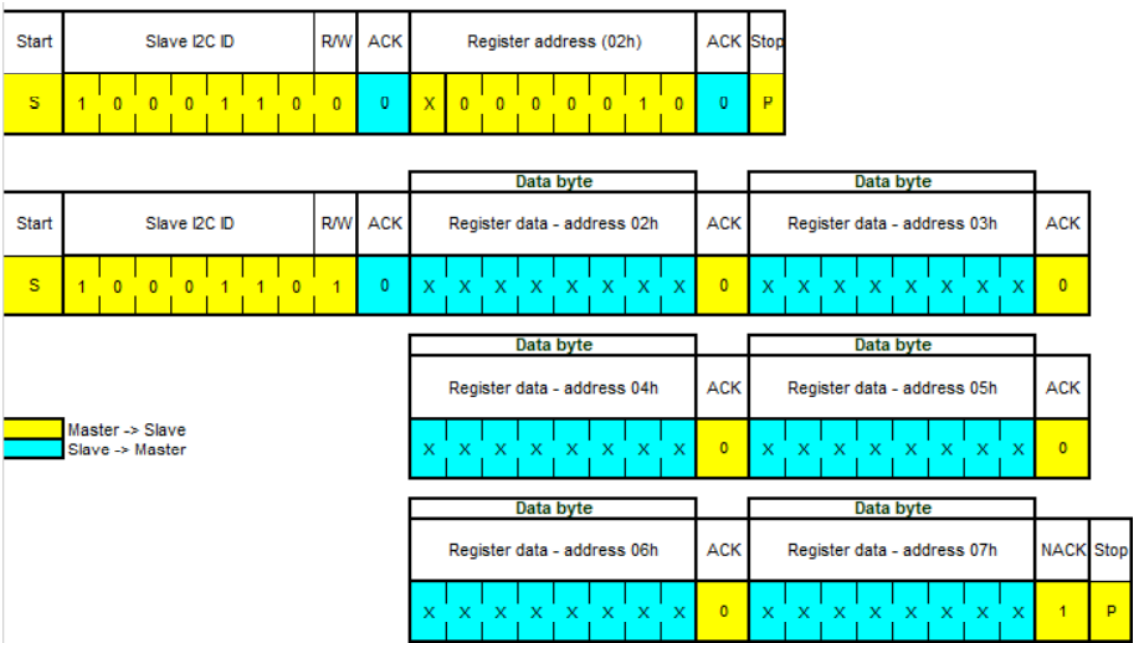


Figure 16: multi-byte read protocol with stop-start

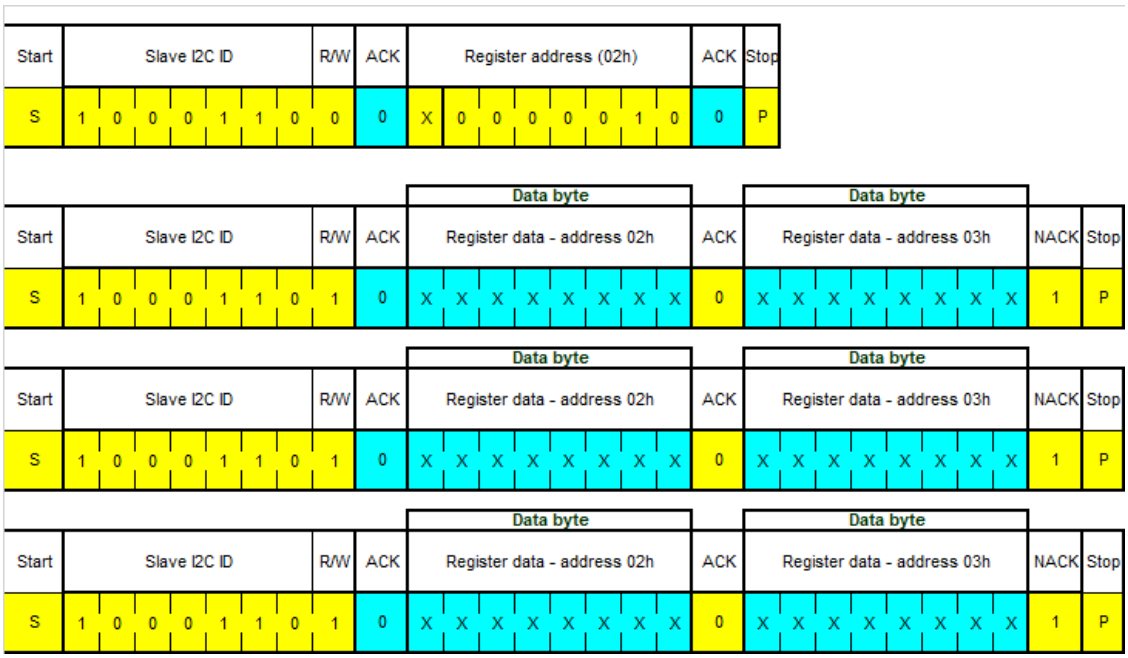


Figure 17: I²C multi-byte read from same start address with stop-start

5.7 I3C Protocol

BMP581 supports the I³C protocol 1.0. Following I³C features are supported:

I²C compatibility including:
Support of I2C-like SDR messages to the BMP581
bus traffic of I2C messages to legacy I2C slaves
I3C single data rate (SDR) mode with up to 12.5 MHz data rate
In-Band Interrupt (IBI)
Timing control asynchronous 0 mode (restricted to maximum 11 MHz I3C frequency and a minimum of 200 kHz)
Timing control synchronous mode

The I3C bus uses the pin SCL for serial clock and SDX as SDA for serial data input and output for the signal lines.

5.7.1 I3C Identifiers

The I3C protocol uses several identifiers and codes to handle communication between several masters and slaves. For communication with this device, there are defined the I3C provisional ID, the Device Characteristics Register (DCR), the Bus Characteristics Register (BCR) and the Mandatory Byte (MDB) for IBIs. The I3C provisional ID has the value defined in Table 20.

Table 23: I3C provisional identifier

Byte	Byte 5								Byte 4								Byte 3								Byte 2								Byte 1								Byte 0							
Bit of byte	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Bit of word	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
De-scrip-tion	MIPI member ID															IDT	Bosch group ID								Device ID								Instance ID				Reserved													
Bit value	0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	SDO	0	0	0	0	0	0	0	0	0	0	0	0
Hex value	0				7				7				0				1				0				4				1				0 or 1				0				0				0			

The value of the Device Characteristics Register (DCR) is fixed to 0x62 to indicate a pressure sensor, as shown in Table 21. See also https://www.mipi.org/MIPI_I3C_device_characteristics_register.

Table 24: I3C device characteristics register (DCR)

DCR<7>	DCR<6>	DCR<5>	DCR<4>	DCR<3>	DCR<2>	DCR<1>	DCR<0>
Device ID							
0	1	1	0	0	0	1	0

The value of the Bus Characteristics Register (BCR) is fixed to 0x06, as shown in Table 22.

Table 25: I3C bus characteristics registers (BCR)

BCR <7:6>	BCR <5>	BCR <4>	BCR <3>	BCR <2>	BCR <1>	BCR <0>
I3C Slave	Reserved	Not a bridge	Device will always respond to I3C Bus commands	Mandatory payload after IBI	IBI capable	Max data Speed: no limitation
2'b00	1'b0	1'b0	1'b0	1'b1	1'b1	1'b0

5.7.2 I3C In-band Interrupts

The device supports the in-band interrupt (IBI) feature of I3C, as described in the I3C specification. In case there is an IBI event, the device will emit its address into the arbitrated address header following a START (but not following a repeated START).
If no START is forthcoming within the Bus Available Condition, then the chip will actively pull down the SDA line to issue a START. The IBI feature can be enabled by the Common Command Code (CCC)' ENEC with the ENINT bit set to 0b1.The IBI feature can be disabled by the Common Command Code (CCC)' DISEC with the DISINT bit set to 0b1.

Upon power up, the feature is disabled by default. The IBI mandatory byte is defined in Table 26. This is also the IBI payload that will be returned upon an GETSTATUS CCC command.
More information on the meaning of the interrupts can be found in Chapter 3.7 "Interrupts".

Table 26: Content of IBI mandatory byte and IBI payload byte

7	6	5	4	3	2	1	0
0	0	0	O	OOR	FIFO thres	FIFO full	data ready

5.7.3 Common Command Codes (CCC)

Supported I3C command control codes (CCCs) are listed in Table 27.

Table 27: List of I3C CCCs

CCC Code	CCC Type	CCC name	Description	BMP581 Supported
0x00	Broadcast	ENEC	Enable events	Y
0x01	Broadcast	DISEC	Disable events	Y
0x02	Broadcast	ENTAS0	Enter active state 0	N
0x03	Broadcast	ENTAS1	Enter active state 1	N
0x04	Broadcast	ENTAS2	Enter active state 2	N
0x05	Broadcast	ENTAS3	Enter active state 3	N
0x06	Broadcast	RSTDAA	Reset dynamic address	Y
0x07	Broadcast	ENTDAA	Enter dynamic address assignment	Y
0x08	Broadcast	DEFSLVS	Define list of slaves	N
0x09	Broadcast	SETMWL	Set max write length	N
0x0A	Broadcast	SETMRL	Set max read length	N
0x0B	Broadcast	ENTTM	Enter test mode	N
0x20	Broadcast	ENTHDR0	Enter HDR mode 0	N
0x21	Broadcast	ENTHDR1	Enter HDR mode 1	N
0x22	Broadcast	ENTHDR2	Enter HDR mode 2	N
0x23	Broadcast	ENTHDR3	Enter HDR mode 3	N
0x24	Broadcast	ENTHDR4	Enter HDR mode 4	N
0x25	Broadcast	ENTHDR5	Enter HDR mode 5	N
0x26	Broadcast	ENTHDR6	Enter HDR mode 6	N
0x27	Broadcast	ENTHDR7	Enter HDR mode 7	N
0x28	Broadcast	SETXTIME	Exchange timing information	Y
0x80	Direct	ENEC	Enable events	Y
0x81	Direct	DISEC	Disable events	Y
0x82	Direct	ENTAS0	Enter active state 0	N
0x83	Direct	ENTAS1	Enter active state 1	N
0x84	Direct	ENTAS2	Enter active state 2	N
0x85	Direct	ENTAS3	Enter active state 3	N
0x86	Direct	RSTDAA	Reset dynamic address	Y
0x87	Direct	SETDASA	Set dynamic address from static address[a]	Y
0x88	Direct	SETNEWDA	Set new dynamic address	Y
0x89	Direct	SETMWL	Set max write length	N
0x8A	Direct	SETMRL	Set max read length	N
0x8B	Direct	GETMWL	Get max write length	N
0x8C	Direct	GETMRL	Get max read length	N
0x8D	Direct	GETPID	Get provisional ID	Y
0x8E	Direct	GETBCR	Get bus characteristics register	Y
0x8F	Direct	GETDCR	Get device characteristics register	Y
0x90	Direct	GETSTATUS	Get device status	Y
0x91	Direct	GETACCMST	Get accept mastership	N
0x93	Direct	SETBRGTGT	Get bridge status	N
0x94	Direct	GETMXDS	Get max data speed	N
0x95	Direct	GETHDRCAP	Get HDR capability	Y
0x98	Direct	SETXTIME	Exchange timing information	Y
0x99	Direct	GETXTIME	Get timing information	Y

a. CCC SETDASA can be used once to assign the dynamic address. If the address shall be changed without resetting or power-cycling the device, CCC SETNEWDA must be used.

5.7.4 I3C SDR Operations

The BMP581's I3C follows the standard I3C specification and defines the private protocol part to meet the data transfer requirements.
The address for all read and write transactions can be set according to the timing diagram in Figure 18. The 7-bit address is allocated in byte section right after the dynamic address transmission. There is a 1-bit dummy content and the address transfers from MSB to LSB.
The read data transfer itself is shown in timing diagrams in Figure 19 for single byte reads, and in Figure 20 for read bursts. Data is provided in 8-bit granularity. In both read and write operations, the data transfer bit order is from MSB to LSB.
A read data transfer may be followed directly by another read data transfer, without setting a new address. In this case the automatic address increment feature of BMP581 increments the addresses for all subsequent data reads until a new address is set.

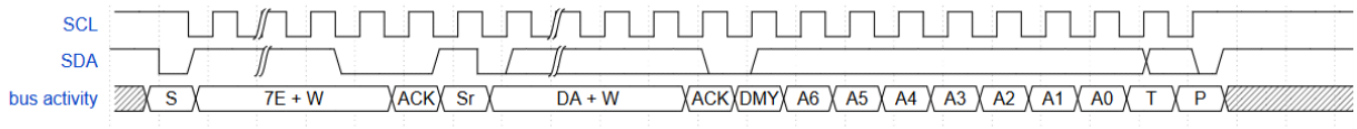


Figure 18: I3C address setting timing diagram

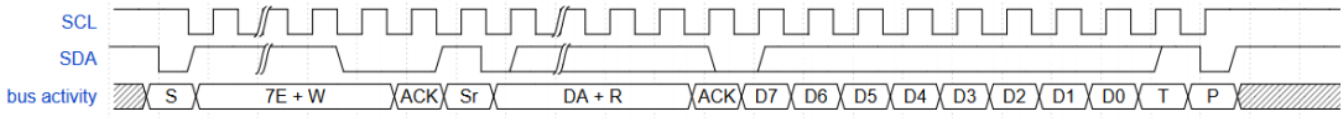


Figure 19: I3C SDR read timing diagram (single byte)

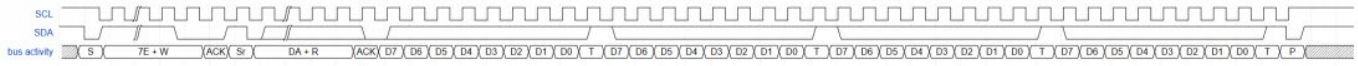


Figure 20: I3C SDR read timing diagram (multiple bytes)

The write transaction (see Figure 21) always contains the target address before the data content is transferred. In case multiple byte datum content are sent out by host, the internal address pointer is incremented automatically and the content will be written into the preceding address byte by byte.

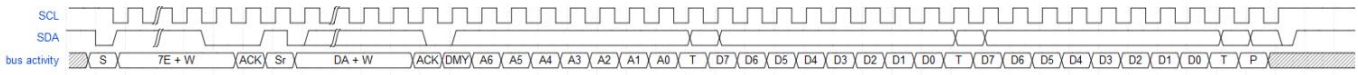


Figure 21: I3C write timing diagram

In accordance to the MIPI I3C specification, the host may skip the 7E header and start with the dynamic address section. This applies to all I3C transactions.

5.7.5 S0/S1 error recovery

BMP581 supports S0/S1 error recovery method b) according to „Table 49 SDR Slave Error Types" of the MIPI I3C specification v1.1. Method a) is not supported. After an S0/S1 error, the BMP581 may stop transmitting IBIs until the next I3C start or stop condition on the bus. Therefore, it is recommended to implement a Start-Stop sequence (for example by a dummy read to a device) after such an error.

6 Pin out and connection diagrams

6.1 Pin Out

Figure 22 shows the pin-out of the device from top and bottom view, respectively. Table 28 shows the related pin descriptions.

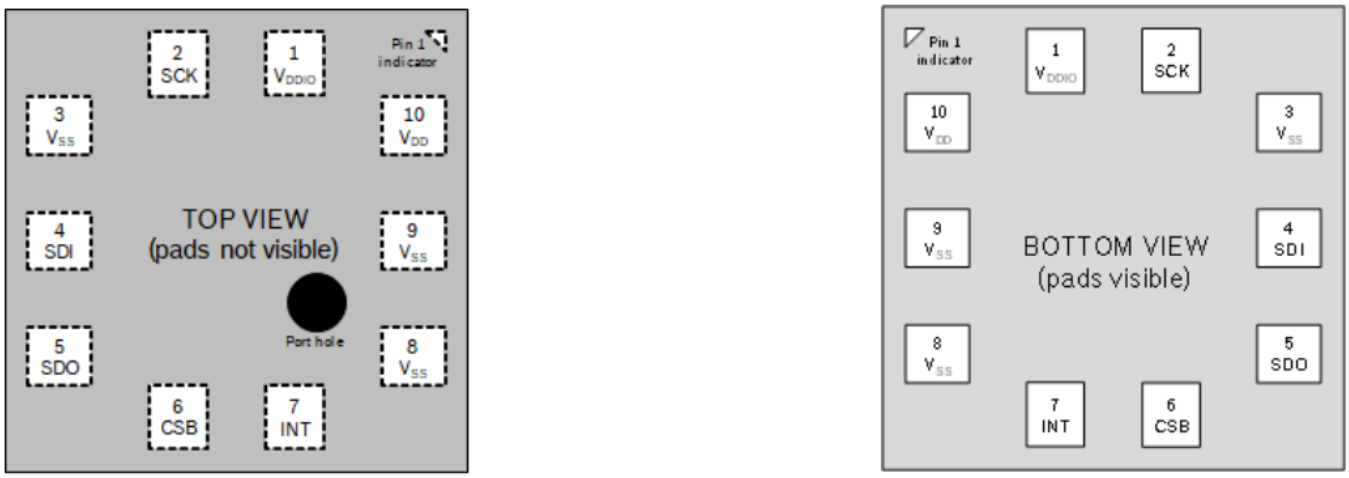


Figure 23: Pin out top and bottom view

Table 28: Pin description

Pin	Name	I/O Type	Description	Connect to	
				SPI 4W	**SPI 3W**
1	VDDIO	Supply	Digital interface supply	V_{DDIO}	
2	SCK	In	Serial clock input	SCK	SCK
3	VSS	Supply	Ground	GND	
4	SDI	In/Out	Serial data input	SDI	SDI/SDO
5	SDO	In/Out	Serial data output	SDO	DNC
6	CSB	In	Chip select	CSB	CSB
7	INT	Out	INT output	host INT input, GND[a] or DNC	
8	VSS	Supply	Ground	GND	
9	VSS	Supply	Ground	GND	
10	VDD	Supply	Analog supply	V_{DD}	

a. GND connection is allowed, as long as the IRQ pin is not activated

6.2 Connection Diagrams

The sensor (including the ASIC) should be used in one of the following four configurations on application level. In all connection scenarios:

- all VSS pins must be connected to GND.

  if the INT pin is not used, it is recommended to be connect it to GND, rather than leaving it floating. In the case of GND connection, the interrupt pin must be disabled by keeping INT_CONFIG.int_en disabled. . If the INT pin must be unconnected in the application, it is recommended to use the following settings for INT_CONFIG:
- INT_CONFIG.int_en = 1
- INT_CONFIG.int_od = 0
- INT_CONFIG.pad_int_drv =

6.2.1 SPI 3-wire

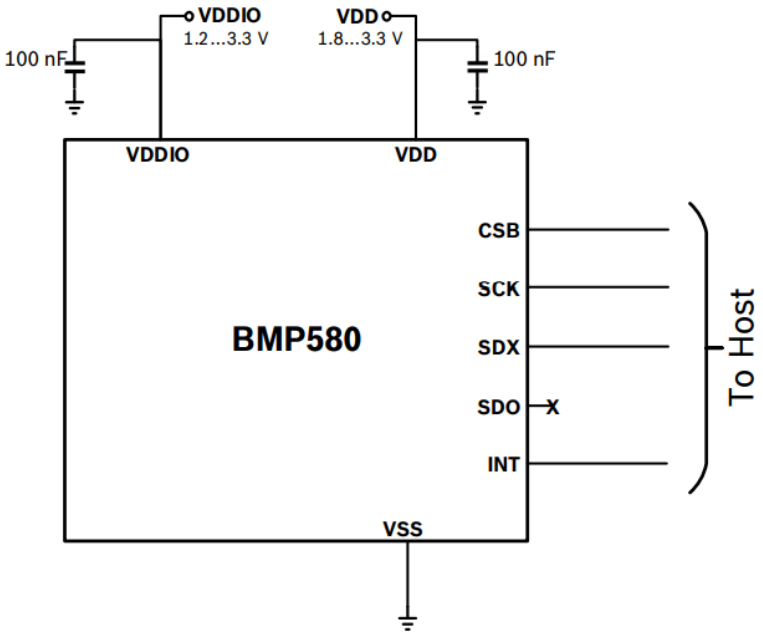


Figure 24: SPI 3-wire connection diagram

The SDO pin must be left floating. The reason is, that the device starts in SPI4 mode after power-up, and drives SDO until the switch to SPI3 is commanded.
The SDI pin must be driven to either low or high voltage when no communication takes place. Otherwise, a floating SDI may create excessive power consumption (and on the longterm potentially also damage to the device), as the input pad is not disabled.

6.2.2 SPI 4-wire

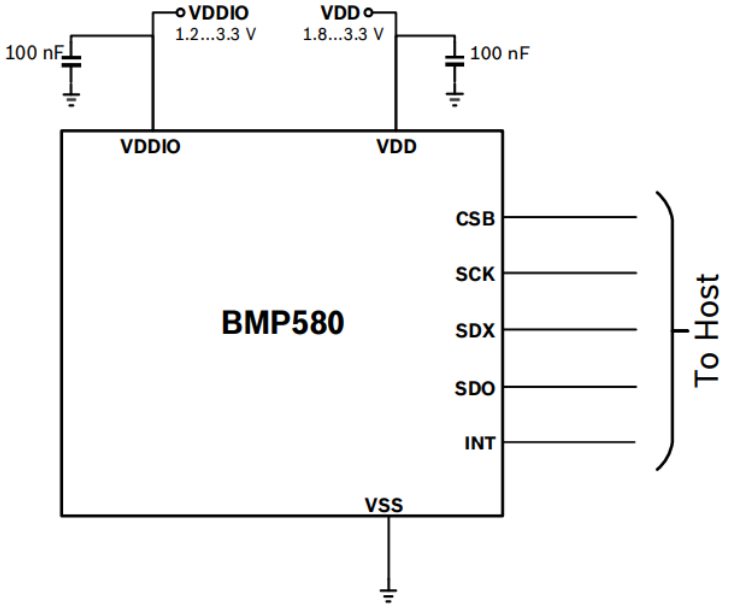


Figure 25: SPI 4-wire connection diagram

6.2.3 I²C

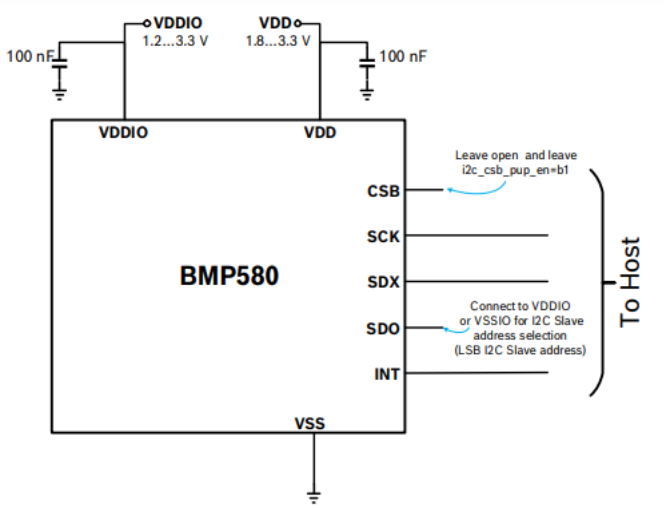


Figure 26: I²C connection diagram

In I²C mode, the CSB pin must be connected to VDDIO.

6.2.4 I3C

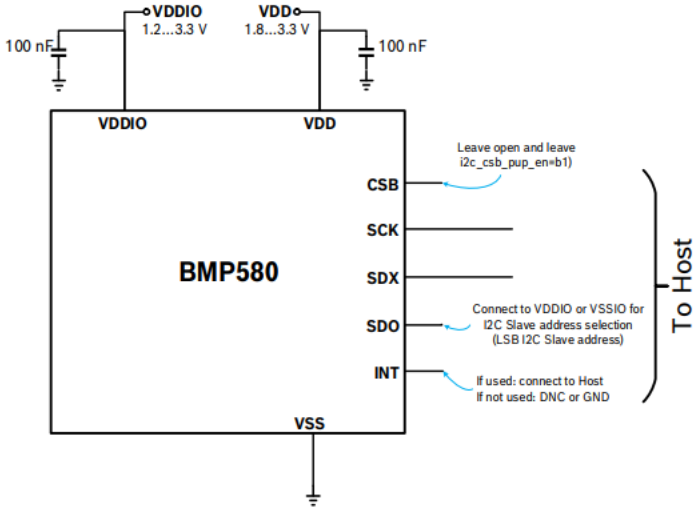


Figure 27: I3C connection diagram

In I3C mode, the CSB pin must be connected to VDDIO.

6.2.5 SPI/I²C/I3C Configuration with VDD, VDDIO ramp-up time <10 µs

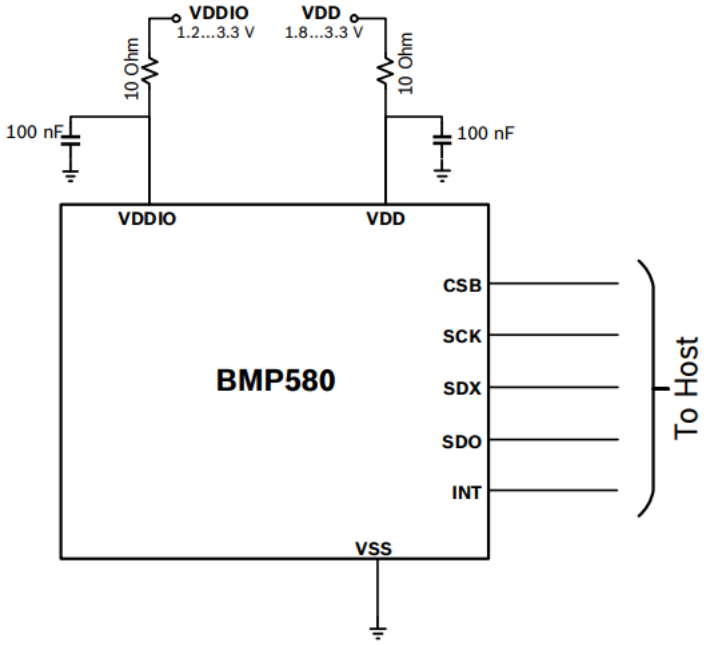


Figure 28: SPI/I²C/I3C Configuration with fast VDD, VDDIO ramp-up times

If VDD or VDDIO ramp-up times are not controlled and are faster than 10us, like in a direct connection to battery, the BMP581 inrush current should be externally limited to avoid damages from repeated power cycles using a 10 Ohm resistance, as depicted in Figure 28.

7 Register Map

Legend			Read-only		Read/Write		Write-only		Reserved	
Addr	**Name**	**Reset value**	**bit7**	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
...	-	-	reserved							
0x7E	CMD	0x00	cmd							
...	-	-	reserved							
0x38	OSR_EFF	0x00	odr_is_...	reserved_6	osr_p_eff			osr_t_eff		
0x37	ODR_CONFIG	0x70	deep_ dis	odr					pwr_mode	
0x36	OSR_CONFIG	0x00	reserved_7	press_en	osr_p			osr_t		
0x35	OOR_CONFIG	0x00	cnt_lim		reserved_5_1				oor_- thr...	
0x34	OOR_RANGE	0x00	oor_range_p							
0x33	OOR_THR_P_MSB	0x00	oor_thr_p_15_8							
0x32	OOR_THR_P_LSB	0x00	oor_thr_p_7_0							
0x31	DSP_IIR	0x00	reserved_7_6		set_iir_p			set_iir_t		
0x30	DSP_CONFIG	0x03	oor_- sel...	fifo_- se..	shdw_- se...	fifo_- se...	shdw_- se...	iir_flu.. .	reserved_0_1	
...	-	-	reserved							
0x2D	NVM_DATA_MSB	0x00	nvm_data_msb							
0x2C	NVM_DATA_LSB	0x00	nvm_data_lsb							
0x2B	NVM_ADDR	0x00	reserved_7	nvm_pro ...	nvm_row_address					
...	-	-	reserved							
0x29	FIFO_DATA	0x7F	fifo_data							
0x28	STATUS	0x02	reserved_7	reserved_6_5		reserved_4	status_...	status_...	status_...	reserved_0
0x27	INT_STATUS	0x00	reserved_7_5			por	oor_p	fifo_ths	fifo_- full	drdy_- da...
0x26	RESERVED_REG4	0x00	reserved_reg4							
0x25	RESERVED_REG3	0x00	reserved_reg3							
0x24	RESERVED_REG2	0x00	reserved_reg2							
0x23	RESERVED_REG1	0x00	reserved_reg1							
0x22	PRESS_DATA_ MSB	0x7F	press_23_16							
0x21	PRESS_DATA_LSB	0x7F	press_15_8							
0x20	PRESS_DATA_XLSB	0x7F	press_7_0							
0x1F	TEMP_DATA_MSB	0x7F	temp_23_16							
0x1E	TEMP_DATA_LSB	0x7F	temp_15_8							
0x1D	TEMP_DATA_XLSB	0x7F	temp_7_0							
0x1C	RESERVED_REG_0	0x00	reserved_reg0							
...	-	-	reserved							
0x18	FIFO_SEL	0x00	reserved_7_5			fifo_dec_sel			fifo_frame_sel	
0x17	FIFO_COUNT	0x00	reserved_7_6		fifo_count					
0x16	FIFO_CONFIG	0x00	reserved_7_6		fifo_- mode	fifo_threshold				
0x15	INT_SOURCE	0x00	reserved_7_4				oor_p _en	fifo_th...	fifo_- fu...	drdy_- da...
0x14	INT_CONFIG	0x35	pad_int_drv				int_en	int_od	int_pol	int_- mode
0x13	DRIVE_CONFIG	0x30	pad_if_drv				reserved_3	reserved_2	spi3_e n	i2c_cs b...
...	-	-	reserved							
0x11	CHIP_STATUS	0x00	reserved_7_4				i3c_er r_3	i3c_er r_0	hif_mode	
...	-	-	reserved							
0x02	REV_ID	0x32	asic_rev_id							
0x01	CHIP_ID	0x50	chip_id							
...	-	-	reserved							

7.1 Register (0x01) ASIC identification ID

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	0	1	0	0	0	0
Content	chip_id							

chip_id:(bit offset: 0) ASIC ID

7.2 Register (0x02) ASIC revision ID

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	1	1	0	0	1	0
Content	asic_rev_id							

asic_rev_id:(bit offset: 0) ASIC revision

7.3 Register (0x11) ASIC status register

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7_4				i3c_err_3	i3c_err_0	hif_mode	

hif_mode:(bit offset: 0) HIF mode (NVM-backed)

Value	Description
0b00 (0x0)	I2C Mode Only [SPI disabled]
0b01 (0x1)	SPI Mode1 and Mode2
0b10 (0x2)	SPI Mode0 and Mode3
0b11 (0x3)	SPI and I2C Available (Autoconfig) Interface selection is automatically configured. Default is I2C mode. During Power-on CSB pin should be tied to VDDIO to pull it high at power-on. If CSB goes low during, I2C interface will be disabled until the next power-on-reset.

i3c_err_0:(bit offset: 2) SDR parity error occurred

i3c_err_3:(bit offset: 3) S0/S1 error occurred. When S0/S1 error occurs, the slave will recover automatically after 60us as if an HDR-exit pattern is executed on the bus. Flag will persist for notification purpose. This flag is clear-onread type. It is cleared automatically once read.

reserved_7_4:(bit offset: 4) reserved

7.4 Register (0x13) Configure host interface related settings (NVM-backed)

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	1	1	0	0	0	0
Content	pad_if_drv				reserved_3	reserved_2	spi3_en	i2c_csb...

i2c_csb_pup_en:(bit offset: 0) CSB pullup selection (valid in I2C mode only)

Values	Description
0b0 (0x0)	disabled
0b1 (0x1)	enabled

spi3_en:(bit offset: 1) SPI 3-wire mode enabling

Values	Description
0b0 (0x0)	SPI 4-wire mode
0b1 (0x1)	SPI 3-wire mode

- reserved_2:(bit offset: 2) reserved
- reserved_3:(bit offset: 3) reserved
- pad_if_drv:(bit offset: 4) Pad drive strength for serial IO pins SDX, SDO (MSB should be set in I2C mode only)
  Note: these register fields should be read-back only after waiting at least 1¬µs after they have been written.

7.5 Register (0x14) Interrupt configuration register

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	1	1	0	1	0	1
Content	pad_int_drv				int_en	int_od	int_pol	int_mode

- int_mode:(bit offset: 0) INT mode:

Values	Description
0b0 (0x0)	pulsed
0b1 (0x1)	latched

- int_pol:(bit offset: 1) INT polarity:

Values	Description
0b0 (0x0)	active low
0b1 (0x1)	active high

▶ int_od:(bit offset: 2) INT pin:

Values	Description
0b0 (0x0)	push-pull
0b1 (0x1)	Open_drain

▶ int_en:(bit offset: 3) Interrupt enabling:

Values	Description
0b0 (0x0)	disabled
0b1 (0x1)	enabled

▶ pad_int_drv:(bit offset: 4) Pad drive strength for INT (MSB should be set in INT open drain config only.) Note: these register fields should be read-back only after waiting at least 1¬µs after they have been written

7.6 Register (0x15) INT source selection

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7_4				oor_p_en	fifo_th...	fifo_fu...	drdy_da...

drdy_data_reg_en:(bit offset: 0) Data Ready
fifo_full_en:(bit offset: 1) FIFO Full (FIFO_FULL)
fifo_ths_en:(bit offset: 2) FIFO Threshold/Watermark (FIFO_THS)
oor_p_en:(bit offset: 3) Pressure data out-of-range (OOR_P)
reserved_7_4:(bit offset: 4) reserved
0x00: Disable INT. Except the POR and Software_reset completion

7.7 Register (0x16) FIFO configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7_6		fifo_mode	fifo_threshold				

▶ fifo_threshold:(bit offset: 0) FIFO threshold

Values	Description
0x0	Disable the FIFO threshold
0x1F	Set the FIFO threshold to 31 frames

▶ fifo_mode:(bit offset: 5) FIFO Mode CTRL

Value	Description
0b0 (0x0)	Stream-to-FIFO Mode
0b1 (0x1)	STOP-on-FULL Mode

▶ reserved_7_6:(bit offset: 6) reserved

7.8 Register (0x17) Number of frames in FIFO

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7_6		fifo_count					

- fifo_count:(bit offset: 0) Number of frames in FIFO
- reserved_7_6:(bit offset: 6) reserved

7.9 Register (0x18) FIFO selection configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7_5			fifo_dec_sel			fifo_frame_sel	

- fifo_frame_sel:(bit offset: 0) FIFO frame data source selection

Value	Description
0b00 (0x0)	FIFO not enabled
0b01 (0x1)	Temperature data
0b10 (0x2)	Pressure data
0b11 (0x3)	Pressure and temperature data

- fifo_dec_sel:(bit offset: 2) FIFO decimation selection
- reserved_7_5:(bit offset: 5) reserved

7.10 Register (0x1C) Reserved

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_reg0							

▶ reserved_reg0:(bit offset: 0) reserved (read returns always 0x00)

7.11 Register (0x1D) Temperature XLSB

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	1	1	1	1	1	1
Content	temp_7_0							

▶ emp_7_0:(bit offset: 0) Temperature XLSB Temp_Data arithmetic representation: (signed, 24, 16) [degC]

7.12 Register (0x1E) Temperature LSB

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	1	1	1	1	1	1
Content	temp_15_8							

▶ temp_15_8:(bit offset: 0) Temperature LSB Temp_Data arithmetic representation: (signed, 24, 16) [degC]

7.13 Register (0x1F) Temperature MSB

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	1	1	1	1	1	1
Content	temp_23_16							

▶ temp_23_16:(bit offset: 0) Temperature MSB Temp_Data arithmetic representation: (signed, 24, 16) [degC]

7.14 Register (0x20) Pressure XLSB

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	1	1	1	1	1	1
Content	press_7_0							

▶ press_7_0:(bit offset: 0) Pressure XLSB Press_Data arithmetic representation: (signed, 24, 6) [Pa]]

7.15 Register (0x21) Pressure LSB

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	1	1	1	1	1	1
Content	press_15_8							

press_15_8:(bit offset: 0) Pressure LSB Press_Data arithmetic representation: (signed, 24, 6) [Pa]

7.16 Register (0x22) Pressure MSB

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	1	1	1	1	1	1
Content	press_23_16							

press_23_16:(bit offset: 0) Pressure MSB Press_Data arithmetic representation: (signed, 24, 6) [Pa]

7.17 Register (0x23) Reserved

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_reg1							

reserved_reg1:(bit offset: 0) reserved (read returns always 0x00)

7.18 Register (0x24) Reserved

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_reg2							

reserved_reg2:(bit offset: 0) reserved (read returns always 0x00)

7.19 Register (0x25) Reserved

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_reg3							

reserved_reg3:(bit offset: 0) reserved (read returns always 0x00)

7.20 Register (0x26) Reserved

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_reg4							

reserved_reg4:(bit offset: 0) reserved (read returns always 0x00)

7.21 Register (0x27) Interrupt status register (clear-on-read).

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7_5			por	oor_p	fifo_ths	fifo_full	drdy_da...

drdy_data_reg:(bit offset: 0) Data Ready

fifo_full:(bit offset: 1) FIFO Full

fifo_ths:(bit offset: 2) FIFO Threshold/Watermark

oor_p:(bit offset: 3) Pressure data out-of-range

por:(bit offset: 4) POR or software reset complete

reserved_7_5:(bit offset: 5) reserved

7.22 Register (0x28) Status register

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	1	0
Content	reserved_7	reserved_6_5		reserved_4	status_...	status_...	status_...	reserved_0

- status_core_rdy:(bit offset: 0) If asserted, the digital core domain is accessible
- status_nvm_rdy:(bit offset: 1) If asserted, device is ready for NVM operations
- status_nvm_err:(bit offset: 2) If asserted, indicates an NVM error, due to at least one of the following reasons: - PMU power transition fail on NVM power request - NVM timeout errors in P/E - NVM Charge Pump voltage fail in PROGRAM/ERASE - During last boot/load command, ECC has detected 2+ errors This bit is cleared/updated upon a new NVM command, if Boot command is executed.
- status_nvm_cmd_err:(bit offset: 3) If asserted, indicates a boot command error, due to at least one of the following reasons: - nvm_usr_read / nvm_usr_prog command submitted when CFG.STATE.nvm_rdy = 0 (FCU is not in STANDBY mode). Boot command not executed -> Boot.STATUS.* and CFG.status.nvm_err not updated. - nvm_usr_prog command submitted when CFG.MISC.nvm_prog_en = 0. Boot command executed -> Boot.STATUS.* and CFG.status.nvm_err updated. - nvm_usr_prog command submitted with CFG.NVM_ADDR.nvm_row_addr outside valid range (6'h08, 6'h09, 6'h1E-6'h22). Boot command executed -> Boot.STATUS.* and CFG.status.nvm_err updated. This bit is cleared upon a new NVM command
- status_boot_err_corrected:(bit offset: 4) If asserted, indicates that an error has been corrected by ECC during last boot-loading Note: This bit is cleared upon a new NVM user command. Additionally, in Deep sleep (Normal and Forced) the NVM is downloaded at beginning of each measurement. It is valid after DRY until the next measurement starts.
- reserved_6_5:(bit offset: 5) reserved
- st_crack_pass:(bit offset: 7) If asserted, crack check has been successfully executed without detecting a crack. Note: this bit is cleared with deasserting ASIC_SELFTEST_CTRL.st_crack_check

7.23 Register (0x29) FIFO output port

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	1	1	1	1	1	1	1
Content	fifo_data							

fifo_data:(bit offset: 0) FIFO read data

7.24 Register (0x2B) NVM address

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7	nvm_pro...	nvm_row_address					

- nvm_row_address:(bit offset: 0) NVM (row) address (Note: this field cannot be written during an ongoing P/T conversion.)
- nvm_prog_en:(bit offset: 6) If set, enables NVM programming (Note: this field cannot be written during an ongoing P/ T conversion.)
- reserved_7:(bit offset: 7) reserved

7.25 Register (0x2C) NVM data (LSB)

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	nvm_data_lsb							

- nvm_data_lsb:(bit offset: 0) NVM Data LSB, bits 7:0 Note: This field cannot be written during an ongoing conversion and should be read-back only after an idle time of at least 1¬µs.

7.26 Register (0x2D) NVM data (MSB)

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	nvm_data_msb							

- nvm_data_msb:(bit offset: 0) NVM Data MSB, bits 15:8 Note: This field cannot be written during an ongoing conversion and should be read-back only after an idle time of at least 1¬µs.

.

7.27 Register (0x30) DSP configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	1	1
Content	oor_sel...	fifo_se...	shdw_-se...	fifo_se...	shdw_-se...	iir_flu...	reserved_0_1	

- comp_pt_en:(bit offset: 0) Pressure sensor compensation

Value	Description
0b0 (0x0)	Press.: no compensation, Temp.: no compensation
0b1 (0x1)	Press.: no compensation, Temp.: compensation
0b10 (0x2)	Press.: compensation, Temp: compensation
0b11 (0x3)	Press.: compensation, Temp: compensation

- iir_flush_forced_en:(bit offset: 2) If set, an IIR filter flush is executed in FORCED mode (Note: This field cannot be written during an ongoing P/T conversion.)
- shdw_sel_iir_t:(bit offset: 3) Temperature Data Registers IIR selection temperature data (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b0 (0x0)	value selected before IIR filter
0b1 (0x1)	value selected after IIR filter

- fifo_sel_iir_t:(bit offset: 4) FIFO IIR selection temperature data (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b0 (0x0)	value selected before IIR filter
0b1 (0x1)	value selected after IIR filter

- shdw_sel_iir_p:(bit offset: 5) Shadow Registers IIR selection pressure data (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b0 (0x0)	value selected before IIR filter
0b1 (0x1)	value selected after IIR filter

- fifo_sel_iir_p:(bit offset: 6) FIFO IIR selection pressure data (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b0 (0x0)	value selected before IIR filter
0b1 (0x1)	value selected after IIR filter

- oor_sel_iir_p:(bit offset: 7) OOR IIR selection (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b0 (0x0)	value selected before IIR filter
0b1 (0x1)	value selected after IIR filter

7.28 Register (0x31) DSP IIR configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7_6		set_iir_p			set_iir_t		

- reserved:write 0x0.
- set_iir_p:(bit offset: 0) Pressure IIR LPF band filter selection. The filter coefficient. (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b000 (0x0)	Bypass
0b001 (0x1)	Filter Coefficient: 1
0b010 (0x2)	Filter Coefficient: 3
0b011 (0x3)	Filter Coefficient: 7
0b100 (0x4)	Filter Coefficient: 15
0b101 (0x5)	Filter Coefficient: 31
0b110 (0x6)	Filter Coefficient: 63
0b111 (0x7)	Filter Coefficient: 127

- set_iir_t:(bit offset: 0) Pressure IIR LPF band filter selection. The filter coefficient (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b000 (0x0)	Bypass
0b001 (0x1)	Filter Coefficient: 1
0b010 (0x2)	Filter Coefficient: 3
0b011 (0x3)	Filter Coefficient: 7
0b100 (0x4)	Filter Coefficient: 15
0b101 (0x5)	Filter Coefficient: 31
0b110 (0x6)	Filter Coefficient: 63
0b111 (0x7)	Filter Coefficient: 127

- reserved_7_6:(bit offset: 6) reserved

7.29 Register (0x32) Out-of-range (OOR) threshold for pressure (LSB)

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	oor_thr_p_7_0							

▶ oor_thr_p_7_0:(bit offset: 0) OOR pressure threshold, bits 7:0

7.30 Register (0x33) Out-of-range (OOR) threshold for pressure (MSB)

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	oor_thr_p_15_8							

▶ oor_thr_p_15_8:(bit offset: 0) OOR pressure threshold, bits 15:8

7.31 Register (0x34) Out-of-range (OOR) range configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	oor_range_p							

▶ oor_range_p:(bit offset: 0) OOR pressure range

7.32 Register (0x35) Out-of-range (OOR) configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R	R	R	R	R	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	cnt_lim		reserved_5_1					oor_thr...

▶ oor_thr_p_16:(bit offset: 0) OOR pressure threshold, bit 16

▶ reserved_5_1:(bit offset: 1) reserved

▶ cnt_lim:(bit offset: 6) OOR count limit (Note: This field cannot be written during an ongoing P/T conversion.)

Value	Description
0b00 (0x0)	Counter limit of 1
0b01 (0x1)	Counter limit of 3
0b10 (0x2)	Counter limit of 7
0b11 (0x3)	Counter limit of 15

7.33 Register (0x36) Over-sampling rate (OSR) configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	reserved_7	press_en	osr_p			osr_t		

- osr_t:(bit offset: 0) OSR_T selection

Value	Description
0b000 (0x0)	oversampling rate = 1x
0b001 (0x1)	oversampling rate = 2x
0b010 (0x2)	oversampling rate = 4x
0b011 (0x3)	oversampling rate = 8x
0b100 (0x4)	oversampling rate = 16x
0b101 (0x5)	oversampling rate = 32x
0b110 (0x6)	oversampling rate = 64x
0b111 (0x7)	oversampling rate = 128x

- osr_p:(bit offset: 3) OSR_P selection

Value	Description
0b000 (0x0)	oversampling rate = 1x
0b001 (0x1)	oversampling rate = 2x
0b010 (0x2)	oversampling rate = 4x
0b011 (0x3)	oversampling rate = 8x
0b100 (0x4)	oversampling rate = 16x
0b101 (0x5)	oversampling rate = 32x
0b110 (0x6)	oversampling rate = 64x
0b111 (0x7)	oversampling rate = 128x

- press_en:(bit offset: 6) If set, enables sensor pressure measurements. Otherwise temperature only measurements is done.
- reserved_7:(bit offset: 7) reserved
- Note: the configured ODR might be invalid in combination with OSR configuration. This is observable with the ODR_-CONFIG.flag odr_is_valid. If configured ODR/OSR settings are invalid, default OSR settings will be used. The effective OSR settings for P/T can be read from osr_t_eff and osr_p_eff

7.34 Register (0x37) Output data rate (ODR) configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	1	1	1	0	0	0	0
Content	deep_dis	odr					pwr_mode	

- pwr_mode:(bit offset: 0) Power mode configuration The user can request a dedicated power mode by writing this field. A read returns the actual mode the device is in.
- Note: This bit is cleared again upon transition from FORCED to STANDBY mode.

Value	Description
0b00 (0x0)	Standby mode: no measurement ongoing
0b01 (0x1)	Normal mode: measurement in configured ODR grid
0b10 (0x2)	Forced mode: forced one-time measurement
0b11 (0x3)	Non-Stop mode: repetitive measurements without further duty-cycling

- odr:(bit offset: 2) ODR Selection
- Note: the configured ODR might be invalid in combination with OSR configuration This is observable with the flag odr_is_valid. If configured ODR/OSR settings are invalid, default OSR settings will be used. The effective OSR settings for P/T can be read from osr_t_eff and osr_p_eff

Value	Description
0x0	240.000 Hz (Error = 0.00)
0x1	218.537 Hz (Error = 0.67)
0x2	199.111 Hz (Error = 0.44)
0x3	179.200 Hz (Error = 0.44)
0x4	160.000 Hz (Error = 0.00)
0x5	149.333 Hz (Error = 0.44)
0x6	140.000 Hz (Error = 0.00)
0x7	129.855 Hz (Error = 0.11)
0x8	120.000 Hz (Error = 0.00)
0x9	110.164 Hz (Error = 0.15)
0xA	100.299 Hz (Error = 0.30)
0xB	89.600 Hz (Error = 0.44)
0xC	80.000 Hz (Error = 0.00)
0xD	70.000 Hz (Error = 0.00)
0xE	60.000 Hz (Error = 0.00)
0xF	50.056 Hz (Error = 0.11)
0x10	45.025 Hz (Error = 0.06)
0x11	40.000 Hz (Error = 0.00)
0x12	35.000 Hz (Error = 0.00)
0x13	30.000 Hz (Error = 0.00)
0x14	25.005 Hz (Error = 0.02)
0x15	20.000 Hz (Error = 0.00)
0x16	15.000 Hz (Error = 0.00)
0x17	10.000 Hz (Error = 0.00)
0x18	5.000 Hz (Error = 0.00)
0x19	4.000 Hz (Error = 0.00)
0x1A	3.000 Hz (Error = 0.00)
0x1B	2.000 Hz (Error = 0.00)
0x1C	1.000 Hz (Error = 0.00)
0x1D	0.500 Hz (Error = 0.00)
0x1E	0.250 Hz (Error = 0.00)
0x1F	0.125 Hz (Error = 0.00)

- deep_dis:(bit offset: 7) If asserted, disables the deep standby (Note: This field cannot be changed during an ongoing P/T conversion.)

7.35 Register (0x38) Effective over-sampling rate (OSR) configuration

Bit	7	6	5	4	3	2	1	0
Read/Write	R	R	R	R	R	R	R	R
Reset Value	0	0	0	0	0	0	0	0
Content	odr_is_...	reserved_6	osr_p_eff			osr_t_eff		

- osr_t_eff:(bit offset: 0) OSR_T effective selection. Effectively selected OSR for temperature. Please refer to OSR_-CONFIG for encodings.
- osr_p_eff:(bit offset: 3) OSR_P effective selection. Effectively selected OSR for pressure. Please refer to OSR_-CONFIG for encodings.
- reserved_6:(bit offset: 6) reserved
- odr_is_valid:(bit offset: 7) If asserted, the ODR parametrization is valid (This is checked on every change in ODR and OSR configuration registers.)
- The values that are effective when the configured ODR might be invalid in combination with OSR configuration This is observable with the ODR_CONFIG.flag odr_is_valid

7.36 Register (0x7E) Command Register

Bit	7	6	5	4	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0	0	0	0	0
Content	cmd							

- cmd:(bit offset: 0) Available commands (Note: Register will always read as 0x00): no other values must be written to this register

Value	Description
0x0	reserved. No command.
0x5D	First CMD in the sequence 0x5D, 0xA0/0xA5 which enables the write/read of the NVM. If another CMD is sent within the sequence, the sequence for enabling the NVM prog mode is reset.
0x69	Last CMD in the sequence 0x73, 0xB4, 0x69, which enables the extended mode and makes the debug and test pages in the register map visible. If another CMD is sent within the sequence, the sequence for enabling the extended page mode is reset. The pages are changed using the register EXT_MODE. Disabling the extended mode is done by resetting the paging enable bit in the paging register
0x73	see extmode_en_last
0xA0	Last CMD in the sequence 0x5D, 0xA0 which enables the write of the NVM. If another CMD is sent within the sequence, the sequence for triggering the NVM programming is reset.
0xA5	Last CMD in the sequence 0x5D, 0xA5 which enables the read of the NVM. If another CMD is sent within the sequence, the sequence for triggering the NVM read is reset.
0xB4	see extmode_en_last
0xB6	Triggers a reset, all user configuration settings are overwritten with their default state. If this register is set using I2C, an ACK will NOT be transmitted to the host

8 Package[8]

8.1 BMP581 Package Outline Dimensions

8.1.1 Top View

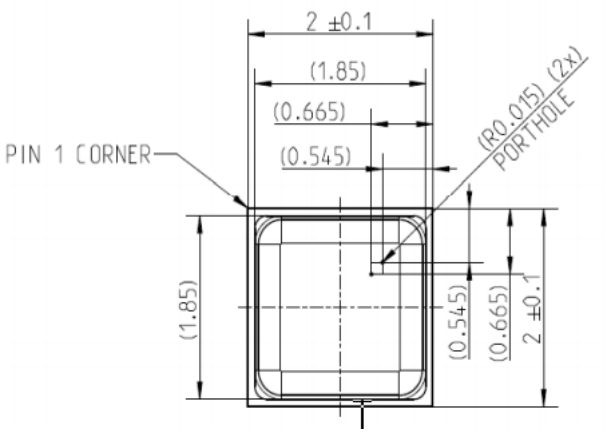


Figure 29: BMP581 top view

8.1.2 Bottom View

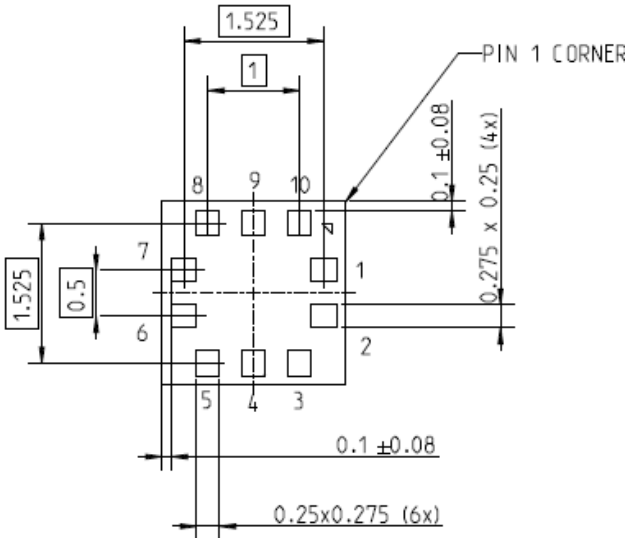


Figure 30: BMP581 bottom view

[8] UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN MILLIMETERS, TOLERANCES ±0.05

8.1.3 Side view

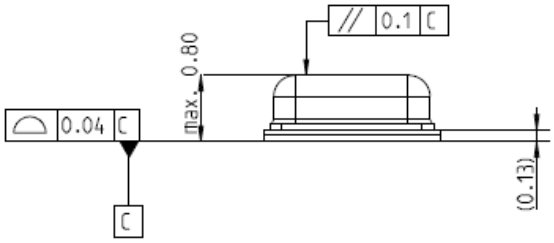


Figure 31: BMP581 side view

8.2 Landing pattern

It is recommended to use a land pattern with a size of Footprint +25 µm on each side. We recommend at least 200 µm distance between the pads. We do not recommend vias or traces under the BMP581. Furthermore, it is recommended that there is no solder mask under the sensor. The recommended horizontal clearance for the solder mask is 20 µm on each side. If the solder mask or other material underneath the sensor gets in contact with the sensor, there may be a negative impact on performance.

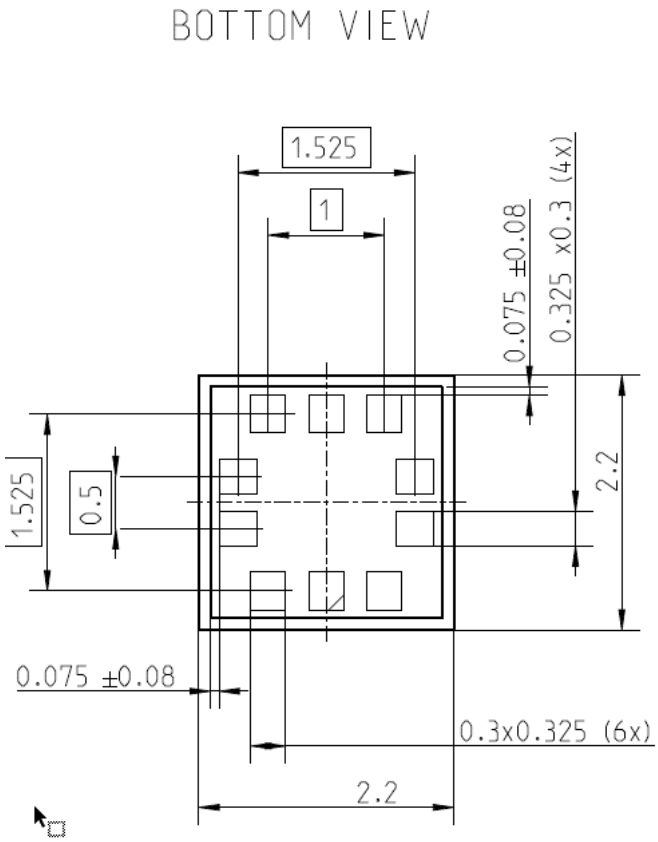


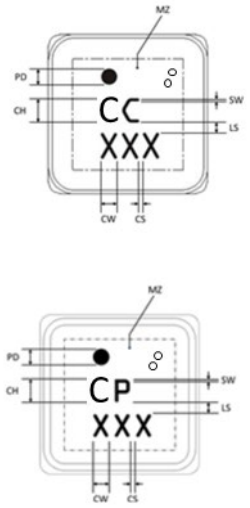
Figure 32: BMP581 landing pattern (bottom view)

8.3 Device Marking

The BMP581 device lid shows the following laser-marking:

8.3.1 Mass Production Devices

Table 29: Package markings

Marking	Name	Symbol	Description
	Product number	C	1 digit (alphanumeric), fixed; for identification of device type (“C” = BMP581)
	Date Code	n/a	no date code
	Supply Chain ID	C or P	1 digit (alphanumeric), fixed; For identification of supplier and location
	Lot Counter (Trace Code)	XXX	3 digits (alphanumeric 0–Z), variable, no reset; 36^3 = 46656 sublots/supplier/device type

8.3.2 Engineering Samples

Table 30: Marking of engineering samples

Position	Name	Symbol	Remark
Upper Line	Eng. Sample ID	CE or PE	Packaging house ID
Lower Line left	Major revision ID	F, A, or C	single alphanumeric digit
Lower Line middle	Minor revision ID	0,1,2…	single numeric digit
Lower Line right	Sublot ID	A,B,C,…	single alphanumeric digit
Corner of pin 1	Pin 1 Orientation Marker	●	Solid circle with diameter of 200 µm

8.4 Moisture Sensitivity Level and Soldering

8.4.1 MSL and device storage

The BMP581 is classified as MSL 1 (moisture sensitivity level) according to IPC/JEDEC standards J-STD-020E and JSTD-033D.

To ensure good solder-ability, the devices shall be stored at room temperature (20°C) before.

The soldering process can lead to an offset shift. The physical origin of this shift is not material aging but mechanical hysteresis frozen in by the soldering temperature cycle. Thus the shift is reversible.
Multiple reflow cycles will not add up in multiple offset shifts. The device is in the same condition after every solder reflow cycle.
Manual unsoldering can lead to further offset shift, especially if the soldering temperature and / or soldering time is above the given values of 260°C and 40 sec.
Avoid contact of the device with liquids or small particles.

8.4.2 Reflow Solder profile

The device has been tested for soldering according to J-STD-002E with Pb-free soldering. The minimum height of the solder after reflow shall be at least 25µm. This is required for a good mechanical decoupling between sensor device and the printed circuit board (PCB). When designing the solder paste silk print opening window, avoid excess solder paste to allow good reflow.

The device has been tested for a total of up to 3 reflow soldering cycles.

This could be a situation where a PCB is mounted with devices from both sides (i.e. 2 reflow cycles necessary) and where in the next step an additional re-work cycle could be required (1 reflow).

8.5 Environmental Safety

8.5.1 RoHS

The BMP581 sensor meets the requirements of the EC restriction of hazardous substances (RoHS) directive, see also: Directive 2015/863 (amending Annex II to Directive 2011/65/EU) of the European Parliament and of the Council on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

8.5.2 Halogen content

The BMP581 is halogen-free. For more details on the analysis results please contact your Bosch Sensortec representative.

8.6 Internal Package Structure

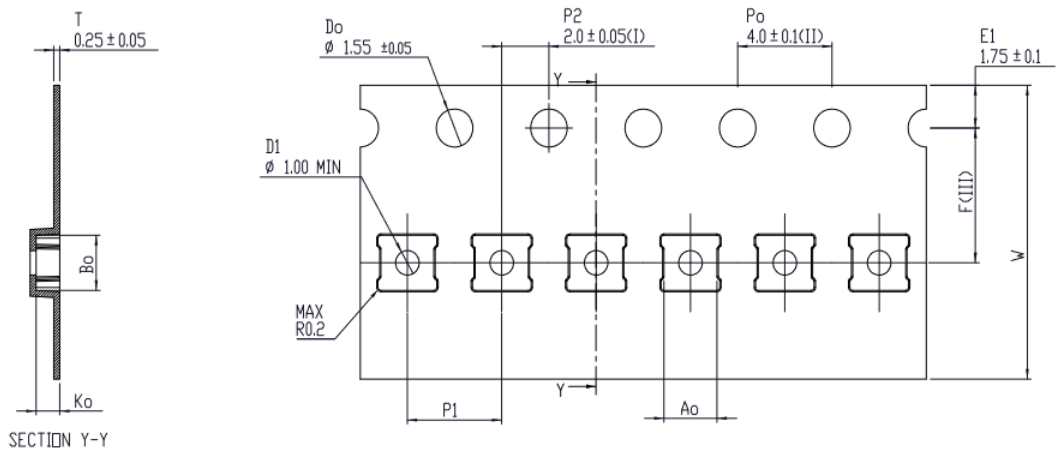
Within the scope of Bosch Sensortec's ambition to improve its products and secure the mass product supply, Bosch Sensortec qualifies additional sources (e.g. 2nd source) for the LGA package of the BMP581.

While Bosch Sensortec took care that all of the technical packages parameters are described above are 100% identical for all sources, there can be differences in the chemical content and the internal structural between the different package sources.

However, as secured by the extensive product qualification process of Bosch Sensortec, this has no impact to the usage or to the quality of the BMP581 product.

8.7 Tape and reel specification

8.7.1 Dimensions



Ao	2.20 +/- 0.05
Bo	2.20 +/- 0.05
Ko	0.95 +/- 0.05
F	5.50 +/- 0.05
P 1	4.00 +/- 0.1
W	12.00 +/- 0.3

Figure 33: Tape and Reel dimensions

Quantity per reel: 10 kpcs.

8.7.2 Orientation within the reel

The orientation of the sensor placement inside the tape on reel can be found below.

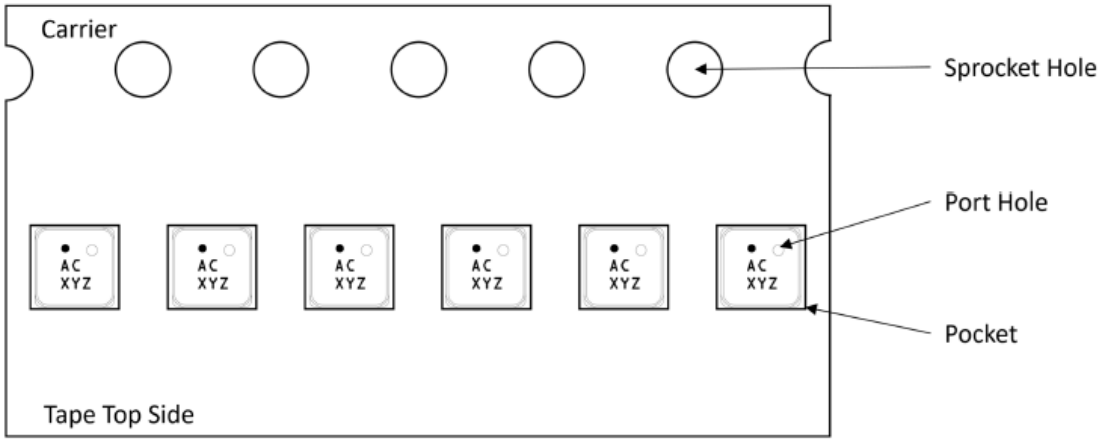


Figure 34: Orientation within tape

9 Legal disclaimer

i. Engineering samples

Engineering Samples are marked with an asterisk (*), (E) or (e). Samples may vary from the valid technical specifications of the product series contained in this data sheet. They are therefore not intended or fit for resale to third parties or for use in end products. Their sole purpose is internal client testing. The testing of an engineering sample may in no way replace the testing of a product series. Bosch Sensortec assumes no liability for the use of engineering samples. The Purchaser shall indemnify Bosch Sensortec from all claims arising from the use of engineering samples.

ii. Product use

Bosch Sensortec products are developed for the consumer goods industry. They may only be used within the parameters of this product data sheet. They are not fit for use in life-sustaining or safety-critical systems. Safety-critical systems are those for which a malfunction is expected to lead to bodily harm, death or severe property damage. In addition, they shall not be used directly or indirectly for military purposes (including but not limited to nuclear, chemical or biological proliferation of weapons or development of missile technology), nuclear power, deep sea or space applications (including but not limited to satellite technology).

Bosch Sensortec products are released on the basis of the legal and normative requirements relevant to the Bosch Sensortec product for use in the following geographical target market: BE, BG, DK, DE, EE, FI, FR, GR, IE, IT, HR, LV, LT, LU, MT, NL, AT, PL, PT, RO, SE, SK, SI, ES, CZ, HU, CY, US, CN, JP, KR, TW. If you need further information or have further requirements, please contact your local sales contact.

The resale and/or use of Bosch Sensortec products are at the purchaser's own risk and his own responsibility. The examination of fitness for the intended use is the sole responsibility of the purchaser.

The purchaser shall indemnify Bosch Sensortec from all third party claims arising from any product use not covered by the parameters of this product data sheet or not approved by Bosch Sensortec and reimburse Bosch Sensortec for all costs in connection with such claims.

The purchaser accepts the responsibility to monitor the market for the purchased products, particularly with regard to product safety, and to inform Bosch Sensortec without delay of all safety-critical incidents.

iii. Application examples and hints

With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Bosch Sensortec hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights or copyrights of any third party. The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. They are provided for illustrative purposes only and no evaluation regarding infringement of intellectual property rights or copyrights or regarding functionality, performance or error has been made.

10 Document history and modification

Rev. No	Chapter	Description of modification/changes	Date
1.0	6	Final official datasheet Changed reset value for address 0x14 and 0x13	September 2021
1.1	5.2 1 4.8.1 3 1 4.9	Updated I²C and I3C timing specifications wrt. 1.2V Updated I²C and I3C timings in Table 4 Corrected register names and added footnotes New chapter quick start guide Added absolute accuracy from -40 to +85°C Changed storage temperature from -40 to 125°C Added solder drift after 5 times reflow Final test (good / bad part)	November 2021
1.2		New product picture	February 2022
1.3	8.2	Added landing pattern	April 2022
	1	Removed max ODR in forced mode	
1.4	7.22	Modified register description	July 2022
	7.27		
	7.36		
	7.32		
	2	Current consumption low power mode modified to high resolution in the table	
1.5	Table 3	Changed tolerances for footprint	April 2023
	8.3	Removed details for laser marking spec	
	8	BMP581 drawing update	
1.6	6.2.3 and 6.2.4	Corrected CSB connection	May 2023
1.7	6.2	Changed description to VDDIO only	October 2023
	8.3	Changed laser marking	
1.8	all	Updated table numbers	December 2023
1.9	2	Comment to maximum ratings	March 2024
1.10	1	Updated sales part number	November 2024
1.11	8.2	Updated landing pattern drawing	February 2025

Bosch Sensortec GmbH
Gerhard-Kindler-Straße 9
72770 Reutlingen / Germany

contact@bosch-sensortec.com
www.bosch-sensortec.com

Document number: BST-BMP581-DS004-11