Microcontroller Lab Assignment 3 & 4: Develop and deploy a firmware driver for I2C and SPI on STM32 microcontroller to communicate with Bosch temperature, pressure, and Humidity Sensor BME280

Computer Science and Engineering, University of Dhaka, Version 1.1

> Initial Version – please report errors Due Date: Within 3 Weeks

sac Bacc. Willing Wee

April 3, 2022

Contents

2]	1.2	General Objectives
2]	1.2	Working Objectives
	Des	ign Diagram
3]		
	Des	scription
•	3.1	GY-BME/P 280
		3.1.1 I2C Interface (Copied from Data sheet)
		3.1.2 I^2C Write
		3.1.3 I2C Read
		3.1.4 SPI Interface
		3.1.5 Register Address and Programming Hints
•	3.2	LCD Display (LCD 1602A)
		3.2.1 How to configure and use the LCD
		3.2.2 Initialize LCD
•	3.3	Experiment
٠	3.4	Report

1 Objectives

1.1 General Objectives

The objectives of the lab assignment are to understand and have hands-on training to build confidence and working knowledge in the use of SPI and I2C protocols and related peripherals. The expected

outcome of this assignment will expertise you to use I2C and SPI to interact with thousands of devices and sensors. To understand I2C and SPI, follow the course website lectures and see the listed videos.

1.2 Working Objectives

Develop SPI and I2C firmware driver to

- Read the current room (any area) temperature, pressure, and humidity from the BME 280 sensor
- Display the results on the LCD QAPASS 1602A.

2 Design Diagram



Figure 1: Building Control System

Use GPIO pins of STM32f446re as following

SPI2 GPIO pins	I2C1 GPIO Pins	QAPASS 1602A
• MOSI – PC1		Cl. CDIO D:
• MISO – PC2	• SDA – PB7	Choose GPIO Pins as required, avoid pins used for
• SCK – PC7	• SCL – PB8	special purpose on the development board
• NSS – PB9 (optional)		-

Note: Bottom line: you can display results on a computer terminal (UART) program; however, the use of LCD will increase your marks up to 20%

3 Description

GY-BME/P 280 is a temperature, humidity, and Pressure Sensor. This precision sensor can measure relative humidity from 0 to 100% with \pm 3% accuracy, barometric pressure from 300Pa to 1100 hPa with \pm 1 hPa absolute accuracy, and temperature from $-40^{\circ}C$ to $85^{\circ}C$ with \pm 1.0°C accuracy. The integrated circuit GY-BME/P 280 uses a Bosch BME280 Digital humidity, pressure, and temperature sensor. Datasheet of Bosch BME280 contains and addresses the detail of the I2C and SPI interface and register sets. The design comprises an LCD to illustrate the room temperature, pressure, and humidity.

3.1 GY-BME/P 280

BME280 is a temperature, pressure, and humidity sensor. Microcontroller can write configuration and read data (room temperature, air pressure, and humidity) from the sensor's registers. The sensor has 6-pins:

Pin Number/Name	Values/Description
VCC	1.2 to 3.6 volt
GND	Ground or '0' voltage
SCL	Clock for I2C and SPI
SDI	Serial Data (In – SPI)
SDO	Serial Data (Out – SPI)
CSB	Chip Select

3.1.1 I2C Interface (Copied from Data sheet)

The I^2C slave interface is compatible with Philips I^2C Specification version. All modes (standard, fast, high speed) are supported. SDA and SCL are not pure open-drain. The 7-bit device address is 111011x. The 6 MSB bits are fixed. The last bit is changeable by SDO value and can be changed during operation. Connecting SDO to GND results in slave address 1110110 (0x76); connecting it to V DDIO results in slave address 1110111 (0x77), which is the same as BMP280's I^2C address. The SDO pin cannot be left floating; if left floating, the I^C address will be undefined. i^2C interface uses the following pins:

PIN	Description
SCL	serial clock (SCL)
SDI	data (SDA)
SDO	Slave address LSB (GND = '0', V_{DDIO} = '1')

CSB must be connected to V_{DDIO} to select the I^2C interface. SDI is bi-directional with open-drain to GND must be externally connected to V_DDIO via a pull-up resistor. Refer to chapter 7 for connection instructions.

3.1.2 I^2C Write

Writing is done by sending the slave address in write mode (RW = '0'), resulting in slave address 111011X0 (the state of the SDO pin determines 'X.' Then, the master sends pairs of sensor register addresses and registers data. A stop condition ends the transaction. This is depicted in Fig. 2

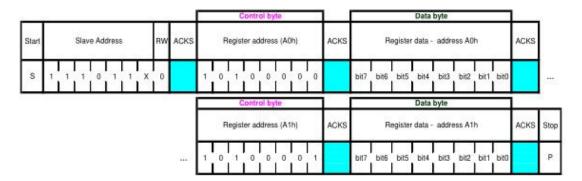


Figure 2: I2C multiple write

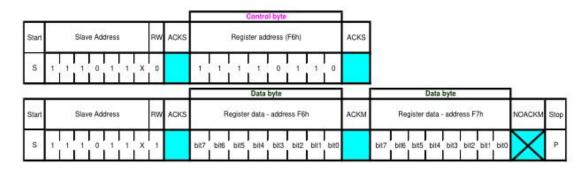


Figure 3: I2C multiple read

3.1.3 I2C Read

First, send the BME280 register address in write mode (slave address 111011X0) to read registers; then, send either a stop or a repeated start condition. After this, the slave is addressed in reading mode (RW = '1') at address 111011X1, after which the slave sends out data from auto-incremented register addresses until a NOACKM and stop condition occurs. This is depicted in Fig 3, where register 0xF6 and 0xF7 are read. Note: that NACK should be sent after the second last byte received. Than send STOP.

3.1.4 SPI Interface

See data sheet from page 34 and 35

3.1.5 Register Address and Programming Hints

Follow the following videos and BME data sheet (link given above). Note that pull-up register is available in STM32F446re. Configure after GPIO setting for open drain. BME280 Controller Tech Video – Click Here

3.2 LCD Display (LCD 1602A)

LCD 1602A can display 16 Characters in a line in two lines display. Character display front resolution 5x8 with 4-bit or 8-bit data input (D0-D7). Power supply V_{DD} 5V ($\pm 10\%$). Pin configuration:

3.2.1 How to configure and use the LCD

You need to configure and set the initial values for a typical data communication and control. The MCU can transfer data either in an 8-bit chunk or 4-bit nibble, and MCU uses Data pins D0-D7 for 8-bit or D4-D7 for 4-bit transfer. However, either configuration is valid to display. The only difference is that the 4-bit needs twice the transfer compared to the 8-bit transfer. However, it would be best to spend some time understanding the configuration. The configuration is straightforward, with absolutely no minor complexity. Use GPIOA port and pins from PA1 to PA7. Note that PA0 is reserved for reset/wakeup; therefore, you should avoid it. You must add sufficient delay between two instructions to be settled.

Data Bus: As shown in the above figure and table, an alphanumeric LCD has an 8-bit data bus referenced as D0-D7. It is an 8-bit data bus; we can send the data/cmd to LCD in bytes. It also provides the provision to send the data/cmd in chunks of 4-bit when there are a limited number of GPIO lines on the microcontroller. Reff: Explore Embedded

Pin Number	Label	Description
1	V_{ss}	0V
2	V_{dd}	5V
3	V_0	_
4	RS	1:Data Input; 0:Instruction Input
5	RW	1:Read; 0:Write
6	Е	Enable (1) LCD to accept data/instruction
7-14	D0-D7	Data Line (8-bit), D4-D7 for 4-bit input
15	A	+5V
16	K	GND

Table 1: LCD 1602 Pins

Register Select(RS): The LCD has two registers, namely a Data register and a Command register. To display data on the LCD needs to be written to the data register of the LCD. MCU can issue the Command to LCD by writing it to the Command register of LCD. It can use this signal to differentiate the data/cmd received by the LCD. If the RS signal is LOW, the LCD interprets the 8-bit info as Command, writes it Command register, and performs the action as per the Command. If the RS signal is high, the LCD interprets the 8-bit info as data and copies it to the data register. After that, the LCD decodes the data for generating the 5x7 pattern and finally displays it on the LCD. Reff: Explore Embedded

Read/Write(RW): This signal is used to write the data/cmd to LCD and reads the busy flag of LCD. The RW should be LOW for a write operation, and for the read operation, the R/W should be HIGH. Reff: Explore Embedded

Enable(EN): This pin enables trigger to LCD. After sending the data/cmd, Select the data/cmd register and select the Write operation. A HIGH-to-LOW pulse has to be sent on this enable pin which will latch the info into the LCD register and triggers the LCD to act accordingly. Reff: Explore Embedded

3.2.2 Initialize LCD

LCD 1602A can be initialize with the following sequence

- 1. Send command 0x30 Using 8-bit interface
- 2. Delay 20ms
- 3. Send command 0x30 8-bit interface
- 4. Delay 20ms
- 5. Send command 0x30 8-bit interface
- 6. Delay 20ms
- 7. Send Function set see Table 2 for more information
- 8. Display Clear command
- 9. Set entry mode command explained below

3.3 Experiment

To test your coding and configuration vary the temperature and humidity. Write the test results in the report.

3.4 Report

Write a report on I2C and SPI experience. Submit code and report to the online lab class.

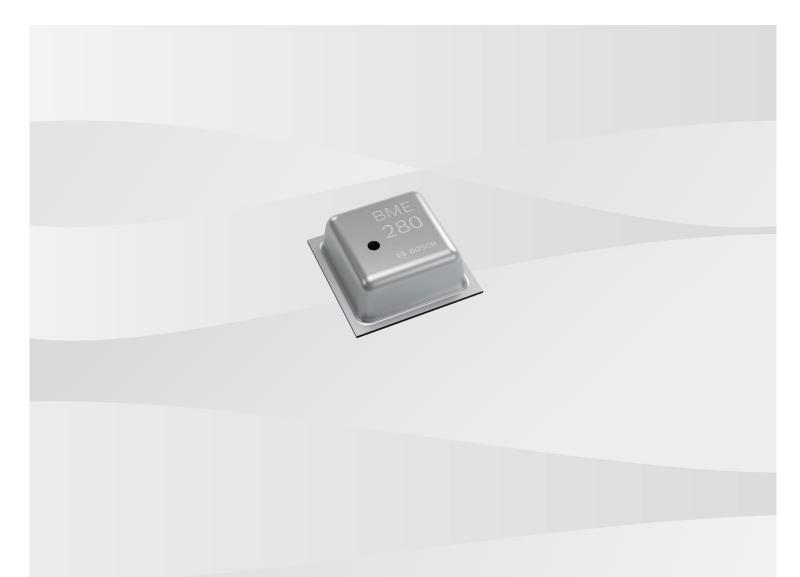
4 End Comments

For the real-life control system design for building management, you can try it with the actual sensor and device; however, you must follow the standard safety requirement.



BME280

Combined humidity and pressure sensor



BME280 - Data sheet

Document revision 1.6

Document release date September 2018

Document number BST-BME280-DS002-15

Technical reference code 0 273 141 185

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

BME280

Digital humidity, pressure and temperature sensor

Key features

Package
 2.5 mm x 2.5 mm x 0.93 mm metal lid LGA

Digital interface
 l²C (up to 3.4 MHz) and SPI (3 and 4 wire, up to 10 MHz)

Supply voltage
 V_{DD} main supply voltage range: 1.71 V to 3.6 V

 V_{DDIO} interface voltage range: 1.2 V to 3.6 V

Current consumption
 1.8 μA @ 1 Hz humidity and temperature
 2.8 μA @ 1 Hz pressure and temperature

3.6 µA @ 1 Hz humidity, pressure and temperature

0.1 µA in sleep mode

Operating range
 -40...+85 °C, 0...100 % rel. humidity, 300...1100 hPa

• Humidity sensor and pressure sensor can be independently enabled / disabled

Register and performance compatible to Bosch Sensortec BMP280 digital pressure sensor

• RoHS compliant, halogen-free, MSL1

Key parameters for humidity sensor

• Response time $(\tau_{63\%})$ 1 s

Accuracy tolerance
 Hysteresis
 ±3 % relative humidity
 ±1% relative humidity

Key parameters for pressure sensor

• RMS Noise 0.2 Pa, equiv. to 1.7 cm

Offset temperature coefficient ±1.5 Pa/K, equiv. to ±12.6 cm at 1 °C temperature change

Typical application

- Context awareness, e.g. skin detection, room change detection
- Fitness monitoring / well-being
 - Warning regarding dryness or high temperatures
 - Measurement of volume and air flow
- · Home automation control
 - control heating, venting, air conditioning (HVAC)
- · Internet of things
- GPS enhancement (e.g. time-to-first-fix improvement, dead reckoning, slope detection)
- Indoor navigation (change of floor detection, elevator detection)
- · Outdoor navigation, leisure and sports applications
- Weather forecast
- Vertical velocity indication (rise/sink speed)

Target devices

- Handsets such as mobile phones, tablet PCs, GPS devices
- · Navigation systems
- Gaming, e.g flying toys
- Camera (DSC, video)
- Home weather stations
- Flying toys
- Watches

General Description

The BME280 is as combined digital humidity, pressure and temperature sensor based on proven sensing principles. The sensor module is housed in an extremely compact metal-lid LGA package with a footprint of only 2.5×2.5 mm² with a height of 0.93 mm. Its small dimensions and its low power consumption allow the implementation in battery driven devices such as handsets, GPS modules or watches. The BME280 is register and performance compatible to the Bosch Sensortec BMP280 digital pressure sensor (see chapter 5.2 for details).

The BME280 achieves high performance in all applications requiring humidity and pressure measurement. These emerging applications of home automation control, in-door navigation, fitness as well as GPS refinement require a high accuracy and a low TCO at the same time.

The humidity sensor provides an extremely fast response time for fast context awareness applications and high overall accuracy over a wide temperature range.

The pressure sensor is an absolute barometric pressure sensor with extremely high accuracy and resolution and drastically lower noise than the Bosch Sensortec BMP180.

The integrated temperature sensor has been optimized for lowest noise and highest resolution. Its output is used for temperature compensation of the pressure and humidity sensors and can also be used for estimation of the ambient temperature.

The sensor provides both SPI and I²C interfaces and can be supplied using 1.71 to 3.6 V for the sensor supply V_{DD} and 1.2 to 3.6 V for the interface supply V_{DD} . Measurements can be triggered by the host or performed in regular intervals. When the sensor is disabled, current consumption drops to 0.1 μ A.

BME280 can be operated in three power modes (see chapter 3.3):

- sleep mode
- normal mode
- forced mode

In order to tailor data rate, noise, response time and current consumption to the needs of the user, a variety of oversampling modes, filter modes and data rates can be selected.

Please contact your regional Bosch Sensortec partner for more information about software packages.

Index of Contents

1.	Spec	ification		8
	1.1	General	electrical specification	8
	1.2	Humidity	parameter specification	9
	1.3	Pressure	e sensor specification	10
	1.4	Tempera	ature sensor specification	11
2.	Abso	olute max	rimum ratings	13
3.	Func	tional de	escription	14
	3.1	Block dia	agram	14
	3.2	Power m	nanagement	14
	3.3	Sensor r	modes	14
		3.3.1	Sensor mode transitions	15
		3.3.2	Sleep mode	15
		3.3.3	Forced mode	15
		3.3.4	Normal mode	16
	3.4	Measure	ement flow	17
		3.4.1	Humidity measurement	17
		3.4.2	Pressure measurement	17
		3.4.3	Temperature measurement	17
		3.4.4	IIR filter	18
	3.5	Recomm	nended modes of operation	19
		3.5.1	Weather monitoring	19
		3.5.2	Humidity sensing	19
		3.5.3	Indoor navigation	20
		3.5.4	Gaming	20
	3.6	Noise		21
4.	Data	readout.		23
	4.1	Data reg	gister shadowing	23

	4.2	Output co	ompensation	23
		4.2.1	Computational requirements	23
		4.2.2	Trimming parameter readout	24
		4.2.3	Compensation formulas	25
5.	Glob	al memor	y map and register description	26
	5.1	General	remarks	26
	5.2	Register	compatibility to BMP280	26
	5.3	Memory	map	26
	5.4	Register	description	27
		5.4.1	Register 0xD0 "id"	27
		5.4.2	Register 0xE0 "reset"	27
		5.4.3	Register 0xF2 "ctrl_hum"	27
		5.4.4	Register 0xF3 "status"	28
		5.4.5	Register 0xF4 "ctrl_meas"	28
		5.4.6	Register 0xF5 "config"	30
		5.4.7	Register 0xF70xF9 "press" (_msb, _lsb, _xlsb)	30
		5.4.8	Register 0xFA0xFC "temp" (_msb, _lsb, _xlsb)	31
		5.4.9	Register 0xFD0xFE "hum" (_msb, _lsb)	31
6.	Digit	al interfac	ces	32
	6.1	Interface	selection	32
	6.2	I ² C Interf	ace	32
		6.2.1	I ² C write	33
		6.2.2	I ² C read	33
	6.3	SPI inter	face	34
		6.3.1	SPI write	34
		6.3.2	SPI read	35
	6.4	Interface	parameter specification	35
		6.4.1	General interface parameters	35
		6.4.2	I ² C timings	35
		6.4.3	SPI timings	36

7.	Pin-o	out and connection diagram	38
	7.1	Pin-out	38
	7.2	Connection diagram I ² C	39
	7.3	Connection diagram 4-wire SPI	40
	7.4	Connection diagram 3-wire SPI	41
	7.5	Package dimensions	42
	7.6	Landing pattern recommendation	43
	7.7	7.7.1 Mass production devices	44
	7.8		
	7.9	Reconditioning Procedure	
	7.10	7.10.1 Dimensions	46
	7.11	Mounting and assembly recommendations	
	7.12	! Environmental safety	48
		7.12.1 RoHS	
		7.12.2 Halogen content	48
		7.12.3 Internal package structure	48
8.	Appe	endix A: Alternative compensation formulas	49
	8.1	Compensation formulas in double precision floating point	49
	8.2	Pressure compensation in 32 bit fixed point	50
9.	Арре	endix B: Measurement time and current calculation	51
	9.1	Measurement time	51
	9.2	Measurement rate in forced mode	51
	9.3	Measurement rate in normal mode	51
	9.4	Response time using IIR filter	52

	9.5	Current consumption	52
10.	Lega	ıl disclaimer	53
	10.1	1 Engineering samples	53
	10.2	2 Product use	53
	10.3	3 Application examples and hints	53
11.	Docu	ument history and modification	54

1. Specification

If not stated otherwise,

- All values are valid over the full voltage range
- All minimum/maximum values are given for the full accuracy temperature range
- ullet Minimum/maximum values of drifts, offsets and temperature coefficients are $\pm 3\sigma$ values over lifetime
- Typical values of currents and state machine timings are determined at 25 °C
- Minimum/maximum values of currents are determined using corner lots over complete temperature range
- Minimum/maximum values of state machine timings are determined using corner lots over 0...+65 °C temperature range

The specification tables are split into humidity, pressure, and temperature part of BME280.

1.1 General electrical specification

Table 1: Electrical parameter specification

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Supply Voltage Internal Domains	V_{DD}	ripple max. 50 mVpp	1.71	1.8	3.6	V
Supply Voltage I/O Domain	V_{DDIO}		1.2	1.8	3.6	V
Sleep current	I _{DDSL}			0.1	0.3	μΑ
Standby current (inactive period of normal mode)	IDDSB			0.2	0.5	μΑ
Current during humidity measurement	Іррн	Max value at 85 °C		340		μА
Current during pressure measurement	I _{DDP}	Max value at -40 °C		714		μА
Current during temperature measurement	I _{DDT}	Max value at 85 °C		350		μΑ
Start-up time	tstartup	Time to first communication after both $V_{DD} > 1.58 \text{ V}$ and $V_{DDIO} > 0.65 \text{ V}$			2	ms
Power supply rejection ratio (DC)	PSRR	full V _{DD} range			±0.01 ±5	%RH/V Pa/V
Standby time accuracy	$\Delta t_{ ext{standby}}$			±5	±25	%

Humidity parameter specification

Table 2: Humidity parameter specification

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Operating range ¹	R _H	For temperatures	-40	25	85	°C
		< 0 °C and > 60 °C see Figure 1	0		100	%RH
Supply current	Ідд,н	1 Hz forced mode, humidity and temperature		1.8	2.8	μΑ
Absolute accuracy tolerance	Ан	2080 %RH, 25 °C, including hysteresis		±3		%RH
Hysteresis ²	Нн	10→90→10 %RH, 25 °C		±1		%RH
Nonlinearity ³	NL _H	10→90 %RH, 25 °C		1		%RH
Response time to complete 63% of step ⁴	$ au_{63\%}$	90→0 or 0→90 %RH, 25°C		1		S
Resolution	R _H			0.008		%RH
Noise in humidity (RMS)	N _H	Highest oversampling, see chapter 3.6		0.02		%RH
Long term stability	ΔH_{stab}	1090 %RH, 25 °C		0.5		%RH/ year

¹ When exceeding the operating range (e.g. for soldering), humidity sensing performance is temporarily degraded and

reconditioning is recommended as described in section 7.8. Operating range only for non-condensing environment. 2 For hysteresis measurement the sequence $10 \rightarrow 30 \rightarrow 50 \rightarrow 70 \rightarrow 90 \rightarrow 70 \rightarrow 50 \rightarrow 30 \rightarrow 10$ %RH is used. The hysteresis is defined as the difference between measurements of the humidity up / down branch and the averaged curve of both branches

³ Non-linear contributions to the sensor data are corrected during the calculation of the relative humidity by the compensation formulas described in section 4.2.3.

⁴ The air-flow in direction to the vent-hole of the device has to be dimensioned in a way that a sufficient air exchange inside to outside will be possible. To observe effects on the response time-scale of the device an air-flow velocity of approx. 1 m/s is needed.

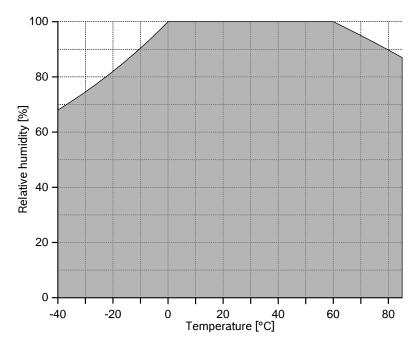


Figure 1: humidity sensor operating range

1.3 Pressure sensor specification

Table 3: Pressure parameter specification

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Operating temperature	T _A	operational	-40	25	+85	°C
range		full accuracy	0		+65	
Operating pressure range	Р	full accuracy	300		1100	hPa
Supply current	$I_{DD,LP}$	1 Hz forced mode, pressure and temperature, lowest power		2.8	4.2	μА

Temperature coefficient	TCOP	2565 °C, 900 hPa		±1.5		Pa/K
of offset ⁵				±12.6		cm/K
Absolute accuracy pressure	A ^P ext	A ^P _{ext} 3001100 hPa -200 °C		±1.7		hPa
	A _{P,full}	300 1100 hPa 0 65 °C		±1.0		hPa
	A ^P	1100 1250 hPa 25 40 °C		±1.5		hPa
Relative accuracy pressure V _{DD} = 3.3V	Arel	700 900hPa 25 40 °C		±0.12		hPa
Resolution of pressure output data	R _P	Highest oversampling		0.18		Pa
Noise in pressure	$N_{\text{P,fullBW}}$	Full bandwidth,		1.3		Pa
		highest oversampling See chapter 3.6		11		cm
	$N_{\text{P,filtered}}$	Reduced bandwidth,		0.2		Pa
		highest oversampling See chapter 3.6		1.7		cm
Solder drift		Minimum solder height 50µm	-0.5		+2.0	hPa
Long term stability ⁶	ΔP_{stab}	per year		±1.0		hPa
Possible sampling rate	f _{sample_P}	Lowest oversampling, see chapter 9.2	157	182		Hz

1.4 Temperature sensor specification

Table 4: Temperature parameter specification

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Operating range	Т	Operational	-40	25	85	°C
		Full accuracy	0		65	°C
Supply current	I _{DD,T}	1 Hz forced mode, temperature measurement only		1.0		μА
	A _{T,25}	25 °C		±0.5		°C

 $^{^{5}}$ When changing temperature by e.g. 10 $^{\circ}$ C at constant pressure / altitude, the measured pressure / altitude will change by (10 $^{\times}$ TCO_P).

 $^{^{6}}$ Long term stability is specified in the full accuracy operating pressure range 0 ... 65 $^{\circ}$ C

Absolute accuracy A temperature ⁷	A _{T,full}	065 °C	±1.0	°C
temperature	A _{T,ext} ⁸	-20 0 °C	±1.25	°C
	A _{T,ext} ⁹	-4020 °C	±1.5	°C
Output resolution	R _T	API output resolution	0.01	°C
RMS noise	N _T	Lowest oversampling	0.005	°C

 ⁷ Temperature measured by the internal temperature sensor. This temperature value depends on the PCB temperature, sensor element self-heating and ambient temperature and is typically above ambient temperature.
 ⁸ Target values & not guaranteed
 ⁹ Target values & not guaranteed

2. Absolute maximum ratings

The absolute maximum ratings are determined over complete temperature range using corner lots. The values are provided in Table 5.

Table 5: Absolute maximum ratings

Parameter	Condition	Min	Max	Unit
Voltage at any supply pin	V _{DD} and V _{DDIO} pin	-0.3	4.25	V
Voltage at any interface pin		-0.3	$V_{DDIO} + 0.3$	V
Storage temperature	≤ 65% RH	-45	+85	°C
Pressure		0	20 000	hPa
ESD	HBM, at any pin		±2	kV
	CDM		±500	V
	Machine model		±200	V
Condensation	No power supplied	Alle	owed	

3. Functional description

3.1 Block diagram

Figure 2 shows a simplified block diagram of the BME280:

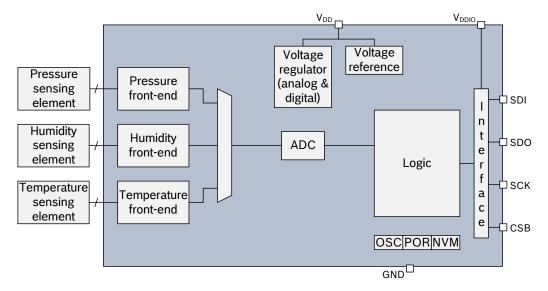


Figure 2: Block diagram of BME280

3.2 Power management

The BME280 has two distinct power supply pins

- V_{DD} is the main power supply for all internal analog and digital functional blocks
- V_{DDIO} is a separate power supply pin used for the supply of the digital interface

A power-on reset (POR) generator is built in; it resets the logic part and the register values after both V_{DD} and V_{DDIO} reach their minimum levels. There are no limitations on slope and sequence of raising the V_{DD} and V_{DDIO} levels. After powering up, the sensor settles in sleep mode (described in chapter 3.3.2).

It is prohibited to keep any interface pin (SDI, SDO, SCK or CSB) at a logical high level when V_{DDIO} is switched off. Such a configuration can permanently damage the device due an excessive current flow through the ESD protection diodes.

If V_{DDIO} is supplied, but V_{DD} is not, the interface pins are kept at a high-Z level. The bus can therefore already be used freely before the BME280 V_{DD} supply is established.

Resetting the sensor is possible by cycling V_{DD} level or by writing a soft reset command. Cycling the V_{DDIO} level will not cause a reset.

3.3 Sensor modes

The BME280 offers three sensor modes: sleep mode, forced mode and normal mode. These can be selected using the *mode*[1:0] setting (see chapter 5.4.5). The available modes are:

- Sleep mode: no operation, all registers accessible, lowest power, selected after startup
- Forced mode: perform one measurement, store results and return to sleep mode
- Normal mode: perpetual cycling of measurements and inactive periods.

The modes will be explained in detail in chapters 3.3.2 (sleep mode), 3.3.3 (forced mode) and 3.3.4 (normal mode).

3.3.1 Sensor mode transitions

The supported mode transitions are shown in Figure 3. If the device is currently performing a measurement, execution of mode switching commands is delayed until the end of the currently running measurement period. Further mode change commands or other write commands to the register *ctrl_hum* are ignored until the mode change command has been executed. Mode transitions other than the ones shown below are tested for stability but do not represent recommended use of the device.

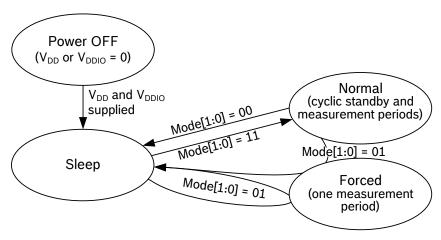


Figure 3: Sensor mode transition diagram

3.3.2 Sleep mode

Sleep mode is entered by default after power on reset. In sleep mode, no measurements are performed and power consumption (IDDSM) is at a minimum. All registers are accessible; Chip-ID and compensation coefficients can be read. There are no special restrictions on interface timings.

3.3.3 Forced mode

In forced mode, a single measurement is performed in accordance to the 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. This is similar to BMP180 operation. Using forced mode is recommended for applications which require low sampling rate or host-based synchronization. The timing diagram is shown below.

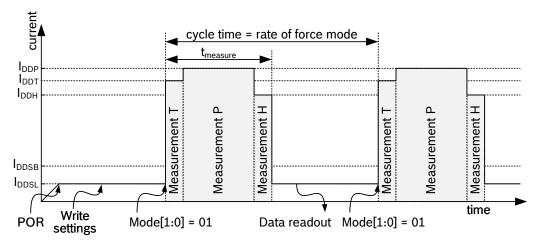


Figure 4: Forced mode timing diagram

3.3.4 Normal mode

Normal mode comprises an automated perpetual cycling between an (active) measurement period and an (inactive) standby period.

The measurements are performed in accordance to the selected measurement and filter options. The standby time is determined by the setting $t_sb[2:0]$ and can be set to between 0.5 and 1000 ms according to Table 27.

The total cycle time depends on the sum of the active time (see chapter 9) and standby time t_{standby}. The current in the standby period (I_{DDSB}) is slightly higher than in sleep mode. After setting the measurement and filter options and enabling normal mode, the last measurement results can always be obtained at the data registers without the need of further write accesses.

Using normal mode is recommended when using the IIR filter. This is useful for applications in which short-term disturbances (e.g. blowing into the sensor) should be filtered. The timing diagram is shown below:

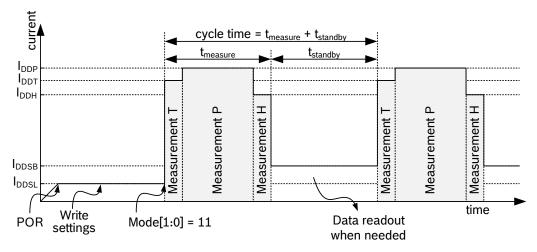


Figure 5: Normal mode timing diagram

3.4 Measurement flow

The BME280 measurement period consists of a temperature, pressure and humidity measurement with selectable oversampling. After the measurement period, the pressure and temperature data can be passed through an optional IIR filter, which removes short-term fluctuations in pressure (e.g. caused by slamming a door). For humidity, such a filter is not needed and has not been implemented. The flow is depicted in the diagram below.

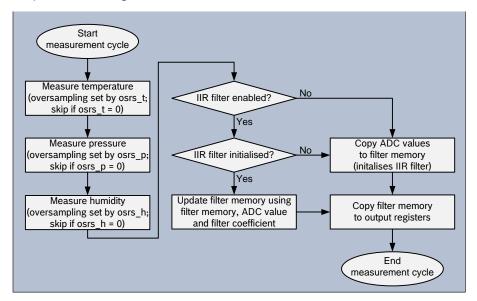


Figure 6: BME280 measurement cycle

The individual blocks of the diagram above will be detailed in the following subchapters.

3.4.1 Humidity measurement

The humidity measurement can be enabled or skipped. When enabled, several oversampling options exist. The humidity measurement is controlled by the *osrs_h*[2:0] setting, which is detailed in chapter 5.4.3. For the humidity measurement, oversampling is possible to reduce the noise. The resolution of the humidity measurement is fixed at 16 bit ADC output.

3.4.2 Pressure measurement

Pressure measurement can be enabled or skipped. When enabled, several oversampling options exist. The pressure measurement is controlled by the $osrs_p[2:0]$ setting which is detailed in chapter 5.4.5. For the pressure measurement, oversampling is possible to reduce the noise. The resolution of the pressure data depends on the IIR filter (see chapter 3.4.4) and the oversampling setting (see chapter 5.4.5):

- When the IIR filter is enabled, the pressure resolution is 20 bit.
- When the IIR filter is disabled, the pressure resolution is 16 + (osrs_p 1) bit, e.g. 18 bit when osrs p is set to '3'.

3.4.3 Temperature measurement

Temperature measurement can be enabled or skipped. Skipping the measurement could be useful to measure pressure extremely rapidly. When enabled, several oversampling options exist. The temperature measurement is controlled by the *osrs_t*[2:0] setting which is detailed in chapter 5.4.5. For the temperature measurement, oversampling is possible to reduce the noise.

The resolution of the temperature data depends on the IIR filter (see chapter 3.4.4) and the oversampling setting (see chapter 5.4.5):

When the IIR filter is enabled, the temperature resolution is 20 bit.

• When the IIR filter is disabled, the temperature resolution is $16 + (osrs_t - 1)$ bit, e.g. 18 bit when $osrs_t$ is set to '3'.

3.4.4 IIR filter

The humidity value inside the sensor does not fluctuate rapidly and does not require low pass filtering. However, the environmental pressure is subject to many short-term changes, caused e.g. by slamming of a door or window, or wind blowing into the sensor. To suppress these disturbances in the output data without causing additional interface traffic and processor work load, the BME280 features an internal IIR filter. It effectively reduces the bandwidth of the temperature and pressure output signals¹⁰ and increases the resolution of the pressure and temperature output data to 20 bit. The output of a next measurement step is filtered using the following formula:

$$\textit{data_filtered} = \frac{\textit{data_filtered_old} \cdot (\textit{filter_coefficient} - 1) + \textit{data_ADC}}{\textit{filter_coefficient}}$$

Data_filtered_old is the data coming from the current filter memory, and data_ADC is the data coming from current ADC acquisition. Data_filtered is the new value of filter memory and the value that will be sent to the output registers.

The IIR filter can be configured to different filter coefficients, which slows down the response to the sensor inputs. Note that the response time with enabled IIR filter depends on the number of samples generated, which means that the data output rate must be known to calculate the actual response time. For register configuration, please refer to Table 28. A sample response time calculation is shown in chapter 9.4.

Table 6: filter settings

Filter coefficient	Samples to reach ≥75 % of step response
Filter off	1
2	2
4	5
8	11
16	22

In order to find a suitable setting for *filter*, please consult chapter 3.5.

When writing to the register *filter*, the filter is reset. The next ADC values will the pass through the filter unchanged and become the initial memory values for the filter. If temperature or pressure measurements are skipped, the corresponding filter memory will be kept unchanged even though the output registers are set to 0x80000. When the previously skipped measurement is re-enabled, the output will be filtered using the filter memory from the last time when the measurement was not skipped. If this is not desired, please write to the *filter* register in order to re-initialize the filter.

-

¹⁰ Since the BME280 does not sample continuously, filtering can suffer from signals with a frequency higher than the sampling rate of the sensor. E.g. environmental fluctuations caused by windows being opened and closed might have a frequency <5 Hz. Consequently, a sampling rate of ODR = 10 Hz is sufficient to obey the Nyquist theorem.

The step response (e.g. response to in sudden change in height) of the different filter settings is displayed in Figure 7.

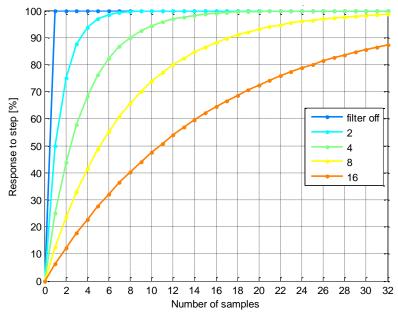


Figure 7: Step response at different IIR filter settings

3.5 Recommended modes of operation

The different oversampling options, filter settings and sensor modes result in a large number of possible settings. In this chapter, a number of settings recommended for various scenarios are presented.

3.5.1 Weather monitoring

Description: Only a very low data rate is needed. Power consumption is minimal. Noise of pressure values is of no concern. Humidity, pressure and temperature are monitored.

Table 7: Settings and performance for weather monitoring

Suggested settings for weather monitoring			
Sensor mode	forced mode, 1 sample / minute		
Oversampling settings	pressure ×1, temperature ×1, humidity ×1		
IIR filter settings	filter off		
Performance for suggested settings			
Current consumption	0.16 μΑ		
RMS Noise	3.3 Pa / 30 cm, 0.07 %RH		
Data output rate 1/60 Hz			

3.5.2 Humidity sensing

Description: A low data rate is needed. Power consumption is minimal. Forced mode is used to minimize power consumption and to synchronize readout, but using normal mode would also be possible.

Table 8: Settings and performance for humidity sensing

Suggested settings for weather monitoring			
Sensor mode	forced mode, 1 sample / second		
Oversampling settings	pressure ×0, temperature ×1, humidity ×1		
IIR filter settings	filter off		
Performance for suggested settings			
Current consumption	2.9 μΑ		
RMS Noise 0.07 %RH			
Data output rate	1 Hz		

3.5.3 Indoor navigation

Lowest possible altitude noise is needed. A very low bandwidth is preferred. Increased power consumption is tolerated. Humidity is measured to help detect room changes. This setting is suggested for the Android settings 'SENSOR_DELAY_NORMAL' and 'SENSOR_DELAY_UI'.

Table 9: Settings and performance for indoor navigation

Suggested settings for indoor navigation			
Sensor mode	normal mode, t _{standby} = 0.5 ms		
Oversampling settings pressure ×16, temperature ×2, humidity ×1			
IIR filter settings	filter coefficient 16		
Performance for suggested settings			
Current consumption	633 μΑ		
RMS Noise	0.2 Pa / 1.7 cm		
Data output rate 25Hz			
Filter bandwidth	0.53 Hz		
Response time (75%)	0.9 s		

3.5.4 Gaming

Low altitude noise is needed. The required bandwidth is ~2 Hz in order to respond quickly to altitude changes (e.g. be able to dodge a flying monster in a game). Increased power consumption is tolerated. Humidity sensor is disabled. This setting is suggested for the Android settings 'SENSOR_DELAY_GAMING' and 'SENSOR_DELAY_FASTEST'.

Table 10: Settings and performance for gaming

Suggested settings for gaming			
Sensor mode normal mode, t _{standby} = 0.5 ms			
Oversampling settings	pressure ×4, temperature ×1, humidity ×0		
IIR filter settings	filter coefficient 16		
Performance for suggested settings			
Current consumption	581 μΑ		
RMS Noise	0.3 Pa / 2.5 cm		
Data output rate 83 Hz			
Filter bandwidth 1.75 Hz			
Response time (75%) 0.3 s			

3.6 Noise

The noise depends on the oversampling and, for pressure and temperature, on the filter setting used. The stated values were determined in a controlled environment and are based on the average standard deviation of 32 consecutive measurement points taken at highest sampling speed. This is needed in order to exclude long term drifts from the noise measurement. The noise depends both on humidity/pressure oversampling and temperature oversampling, since the temperature value is used for humidity/pressure temperature compensation. The oversampling combinations use below results in an optimal power to noise ratio.

Table 11: Noise and current for humidity

Humidity / temperature oversampling setting	Typical RMS noise in humidity [%RH] at 25 °C	Typ. current [µA] at 1 Hz forced mode, 25 °C, humidity and temperature measurement, incl.
×1/×1	0.07	1.8
×2/×1	0.05	2.5
×4 / ×1	0.04	3.8
×8/×1	0.03	6.5
×16 / ×1	0.02	11.7

Table 12: Noise and current for pressure

Typical RMS noise in pressure [Pa] at 25 °C						Typ. current [µA] at 1 Hz forced
Pressure / temperature	IIR filter coefficient			icient	mode, 25 °C, pressure and temperature measurement, incl.	
oversampling setting	off	2	4	8	16	IDDSM
×1/×1	3.3	1.9	1.2	0.9	0.4	2.8
×2/×1	2.6	1.5	1.0	0.6	0.4	4.2
×4/×1	2.1	1.2	8.0	0.5	0.3	7.1
×8/×1	1.6	1.0	0.6	0.4	0.2	12.8
×16 / ×2	1.3	0.8	0.5	0.4	0.2	24.9

Table 13: Temperature dependence of pressure noise

RMS noise at different temperatures				
Temperature	Typical change in noise compared to 25 °C			
-10 °C	+25 %			
25 °C	±0 %			
75 °C	-5 %			

Table 14: Noise in temperature

Temperature oversampling setting	Typical RMS noise in temperature [°C] at 25 °C
×1	0.005
×2	0.004
×4	0.003
×8	0.003
×16	0.002

4. Data readout

To read out data after a conversion, it is strongly recommended to use a burst read and not address every register individually. This will prevent a possible mix-up of bytes belonging to different measurements and reduce interface traffic. Note that in I²C mode, even when pressure was not measured, reading the unused registers is faster than reading temperature and humidity data separately.

Data readout is done by starting a burst read from 0xF7 to 0xFC (temperature and pressure) or from 0xF7 to 0xFE (temperature, pressure and humidity). The data are read out in an unsigned 20-bit format both for pressure and for temperature and in an unsigned 16-bit format for humidity. It is strongly recommended to use the BME280 API, available from Bosch Sensortec, for readout and compensation. For details on memory map and interfaces, please consult chapters 5 and 6 respectively.

After the uncompensated values for pressure, temperature and humidity 'ut', 'up' and 'uh' have been read, the actual humidity, pressure and temperature needs to be calculated using the compensation parameters stored in the device. The procedure is elaborated in chapter 4.2.

4.1 Data register shadowing

In normal mode, the timing of measurements is not necessarily synchronized to the readout by the user. 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. The end of the burst read is marked by the rising edge of CSB pin in SPI case or by the recognition of a stop condition in I²C case. After the end of the burst read, all user data registers are updated at once.

4.2 Output compensation

The BME280 output consists of the ADC output values. However, each sensing element behaves differently. Therefore, the actual pressure and temperature must be calculated using a set of calibration parameters. In this chapter, the method to read out the trimming values will be given. The recommended calculation uses fixed point arithmetic and is given in chapter 4.2.3. In high-level languages like Matlab™ or LabVIEW™, fixed-point code may not be well supported. In this case the floating-point code in appendix 8.1 can be used as an alternative. For 8-bit micro controllers, the variable size may be limited. In this case a simplified 32 bit integer code with reduced accuracy is given in appendix 8.2.

4.2.1 Computational requirements

In the table below an overview is given for the number of clock cycles needed for compensation on a 32 bit Cortex-M3 micro controller with GCC optimization level -O2. This controller does not feature a floating point unit, thus all floating-point calculations are emulated. Floating point is only recommended for PC application, where an FPU is present and these calculations are performed drastically faster.

Table 15: Computational requirements for compensation formulas

Compensation of	Number of clocks (ARM Cortex-M3)				
	32 bit integer	64 bit integer	Double precision		
Humidity	~83	-	~2900 11		
Temperature	~46	-	~2400 11		
Pressure	~112 12	~1400	~5400 11		

4.2.2 Trimming parameter readout

The trimming parameters are programmed into the devices' non-volatile memory (NVM) during production and cannot be altered by the customer. Each compensation word is a 16-bit signed or unsigned integer value stored in two's complement. As the memory is organized into 8-bit words, two words must always be combined in order to represent the compensation word. The 8-bit registers are named calib00...calib41 and are stored at memory addresses 0x88...0xA1 and 0xE1...0xE7. The corresponding compensation words are named dig_T# for temperature compensation related values, dig_P# for pressure related values and dig_H# for humidity related values. The mapping is seen in Table 16.

Table 16: Compensation parameter storage, naming and data type

Register Address	Register content	Data type
0x88 / 0x89	dig_T1 [7:0] / [15:8]	unsigned short
0x8A / 0x8B	dig_T2 [7:0] / [15:8]	signed short
0x8C / 0x8D	dig_T3 [7:0] / [15:8]	signed short
0x8E / 0x8F	dig_P1 [7:0] / [15:8]	unsigned short
0x90 / 0x91	dig_P2 [7:0] / [15:8]	signed short
0x92 / 0x93	dig_P3 [7:0] / [15:8]	signed short
0x94 / 0x95	dig_P4 [7:0] / [15:8]	signed short
0x96 / 0x97	dig_P5 [7:0] / [15:8]	signed short
0x98 / 0x99	dig_P6 [7:0] / [15:8]	signed short
0x9A/0x9B	dig_P7 [7:0] / [15:8]	signed short
0x9C / 0x9D	dig_P8 [7:0] / [15:8]	signed short
0x9E / 0x9F	dig_P9 [7:0] / [15:8]	signed short
0xA1	dig_H1 [7:0]	unsigned char
0xE1/0xE2	dig_H2 [7:0] / [15:8]	signed short
0xE3	dig_H3 [7:0]	unsigned char
0xE4 / 0xE5[3:0]	dig_H4 [11:4] / [3:0]	signed short
0xE5[7:4] / 0xE6	dig_H5 [3:0] / [11:4]	signed short

¹¹ Use only recommended for high-level programming languages like Matlab™ or LabVIEW™

¹² Use only recommended for 8-bit micro controllers

0xE7 dig_H6 signed char

4.2.3 Compensation formulas

Please note that it is strongly advised to use the API available from Bosch Sensortec to perform readout and compensation. If this is not wanted, the code below can be applied at the user's risk. Both pressure and temperature values are expected to be received in 20 bit format, positive, stored in a 32 bit signed integer. Humidity is expected to be received in 16 bit format, positive, stored in a 32 bit signed integer.

The variable t_fine (signed 32 bit) carries a fine resolution temperature value over to the pressure and humidity compensation formula and could be implemented as a global variable.

The data type "BME280_S32_t" should define a 32 bit signed integer variable type and can usually be defined as "long signed int".

The data type "BME280_U32_t" should define a 32 bit unsigned integer variable type and can usually be defined as "long unsigned int".

For best possible calculation accuracy in pressure, 64 bit integer support is needed. If this is not possible on your platform, please see appendix 8.2 for a 32 bit alternative.

The data type "BME280_S64_t" should define a 64 bit signed integer variable type, which on most supporting platforms can be defined as "long long signed int". The revision of the code is rev.1.1.

```
// Returns temperature in DegC, resolution is 0.01 DegC. Output value of "5123" equals 51.23
// t fine carries fine temperature as global value
BME280_S32_t t_fine;
BME280_S32_t BME280_compensate_T_int32(BME280_S32_t adc_T)
   BME280 S32 t var1, var2, T;
   var1 = (((adc T)>3) - ((BME280 S32 t)dig T1<<1))) * ((BME280 S32 t)dig T2)) >> 11;
   var2 = (((((adc_T)>4) - ((BME280_S32_t)dig_T1))) * ((adc_T)>4) - ((BME280_S32_t)dig_T1)))
>> 12) *
      ((BME280_S32_t)dig_T3)) >> 14;
   t fine = var1 + var2;
   T = (t fine * 5 + 128) >> 8;
   return T;
// Returns pressure in Pa as unsigned 32 bit integer in Q24.8 format (24 integer bits and 8
fractional bits).
// Output value of "24674867" represents 24674867/256 = 96386.2 Pa = 963.862 hPa
BME280 U32 t BME280 compensate P int64(BME280 S32 t adc P)
   BME280_S64_t var1, var2, p;
   var1 = ((BME280_S64_t)t_fine) - 128000;
   var2 = var1 * var1 * (BME280_S64_t)dig_P6;
   var2 = var2 + ((var1*(BME280_S64_t)dig_P5)<<17);</pre>
   var2 = var2 + (((BME280_S64_t)dig_P4)<<35);
   var1 = ((var1 * var1 * (BME280_S64_t)dig_P3)>>8) + ((var1 * (BME280_S64_t)dig_P2)<<12);
   var1 = (((((BME280_S64_t)1) << 47) + var1)) * ((BME280_S64_t) dig_P1) >> 33;
   if (var1 == 0)
   {
      return 0; // avoid exception caused by division by zero
   p = 1048576 - adc P;
   p = (((p << 31) - var2) * 3125) / var1;
   var1 = (((BME280_S64_t)dig_P9) * (p>>13) * (p>>13)) >> 25;
var2 = (((BME280_S64_t)dig_P8) * p) >> 19;
   p = ((p + var1 + var2) >> 8) + (((BME280 S64 t)dig P7) << 4);
   return (BME280_U32_t)p;
\//\ Returns humidity in %RH as unsigned 32 bit integer in Q22.10 format (22 integer and 10
fractional bits).
// Output value of "47445" represents 47445/1024 = 46.333 %RH
BME280_U32_t bme280_compensate_H_int32(BME280_S32_t adc_H)
   BME280 S32 t v x1 u32r;
   v \times 1 u \cdot 32r = (t fine - ((BME280 S32 t) 76800));
```

5. Global memory map and register description

5.1 General remarks

The entire communication with the device is performed by reading from and writing to registers. Registers have a width of 8 bits. There are several registers which are reserved; they should not be written to and no specific value is guaranteed when they are read. For details on the interface, consult chapter 6.

5.2 Register compatibility to BMP280

The BME280 is downward register compatible to the BMP280, which means that the pressure and temperature control and readout is identical to BMP280. However, the following exceptions have to be considered:

Table 17: Register incompatibilities between BMP280 and BME280

Register	Bits	Content	BMP280	BME280
0xD0 "id"	7:0	chip_id	Read value is 0x56 / 0x57 (samples) 0x58 (mass production)	Read value is 0x60
0xF5 "config"	7:5	t_sb	'110': 2000 ms '111': 4000 ms	'110': 10 ms '111': 20 ms
0xF70xF9 "press"	19:0	press	Resolution (1620 bit) depends only on osrs_p	Without filter, resolution depends on osrs_p; when using filter, resolution is always 20 bit
0xFA0xFC "temp"	19:0	temp	Resolution (1620 bit) only depends on <i>osrs_t</i>	Without filter, resolution depends on osrs_t; when using filter, resolution is always 20 bit

5.3 Memory map

The memory map is given in Table 18 below. Reserved registers are not shown.

Table 18: Memory map

Register Name	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Reset state
hum_lsb	0xFE				hum_l:	sb<7:0>				0x00
hum_msb	0xFD				hum_m	isb<7:0>				0x80
temp_xlsb	0xFC		temp_x	lsb<7:4>		0	0	0	0	0x00
temp_lsb	0xFB				temp_l	sb<7:0>				0x00
temp_msb	0xFA				temp_m	nsb<7:0>				0x80
press_xlsb	0xF9		press_x	lsb<7:4>		0	0	0	0	0x00
press_lsb	0xF8		press_lsb<7:0>					0x00		
press_msb	0xF7		press_msb<7:0>				0x80			
config	0xF5		t_sb[2:0] filter[2:0] spi3w_en[0]			spi3w_en[0]	0x00			
ctrl_meas	0xF4		osrs_t[2:0]			le[1:0]	0x00			
status	0xF3					measuring[0]			im_update[0]	0x00
ctrl_hum	0xF2		osrs_h[2:0]				0x00			
calib26calib41	0xE10xF0		calibration data			individual				
reset	0xE0		reset[7:0]			0x00				
id	0xD0		chip_id[7:0]			0x60				
calib00calib25	0x880xA1				calibrat	tion data				individual

Registers:	Reserved registers	Calibration data	Control registers	Data registers	Status registers	Chip ID	Reset
Туре:	do not change	read only	read / write	read only	read only	read only	write only

5.4 Register description

5.4.1 Register 0xD0 "id"

The "id" register contains the chip identification number chip_id[7:0], which is 0x60. This number can be read as soon as the device finished the power-on-reset.

5.4.2 Register 0xE0 "reset"

The "reset" register contains the soft reset word reset[7:0]. If the value 0xB6 is written to the register, the device is reset using the complete power-on-reset procedure. Writing other values than 0xB6 has no effect. The readout value is always 0x00.

5.4.3 Register 0xF2 "ctrl hum"

The "ctrl_hum" register sets the humidity data acquisition options of the device. Changes to this register only become effective after a write operation to "ctrl_meas".

Table 19: Register 0xF2 "ctrl_hum"

Register 0xF2 "ctrl_hum"	Name	Description
Bit 2, 1, 0	osrs_h[2:0]	Controls oversampling of humidity data. See Table 20 for settings and chapter 3.4.1 for details.

Table 20: register settings osrs_h

osrs_h[2:0]	Humidity oversampling
000	Skipped (output set to 0x8000)
001	oversampling ×1
010	oversampling ×2
011	oversampling ×4
100	oversampling ×8
101, others	oversampling ×16

5.4.4 Register 0xF3 "status"

The "status" register contains two bits which indicate the status of the device.

Table 21: Register 0xF3 "status"

Register 0xF3 "status"	Name	Description
Bit 3	measuring[0]	Automatically set to '1' whenever a conversion is running and back to '0' when the results have been transferred to the data registers.
Bit 0	im_update[0]	Automatically set to '1' when the NVM data are being copied to image registers and back to '0' when the copying is done. The data are copied at power-on-reset and before every conversion.

5.4.5 Register 0xF4 "ctrl_meas"

The "ctrl_meas" register sets the pressure and temperature data acquisition options of the device. The register needs to be written after changing "ctrl_hum" for the changes to become effective.

Table 22: Register 0xF4 "ctrl_meas"

Register 0xF4 "ctrl_meas"	Name	Description
Bit 7, 6, 5	osrs_t[2:0]	Controls oversampling of temperature data. See Table 24 for settings and chapter 3.4.3 for details.
Bit 4, 3, 2	osrs_p[2:0]	Controls oversampling of pressure data. See Table 23 for settings and chapter 3.4.2 for details.
Bit 1, 0	mode[1:0]	Controls the sensor mode of the device. See Table 25 for settings and chapter 3.3 for details.

Table 23: register settings osrs_p

osrs_p[2:0]	Pressure oversampling
000	Skipped (output set to 0x80000)
001	oversampling ×1
010	oversampling ×2
011	oversampling ×4
100	oversampling ×8
101, others	oversampling ×16

Table 24: register settings osrs_t

osrs_t[2:0]	Temperature oversampling
000	Skipped (output set to 0x80000)
001	oversampling ×1
010	oversampling ×2
011	oversampling ×4
100	oversampling ×8
101, others	oversampling ×16

Table 25: register settings mode

mode[1:0]	Mode
00	Sleep mode
01 and 10	Forced mode
11	Normal mode

5.4.6 Register 0xF5 "config"

The "config" register sets the rate, filter and interface options of the device. Writes to the "config" register in normal mode may be ignored. In sleep mode writes are not ignored.

Table 26: Register 0xF5 "config"

Register 0xF5 "config"	Name	Description
Bit 7, 6, 5	t_sb[2:0]	Controls inactive duration t _{standby} in normal mode. See Table 27 for settings and chapter 3.3.4 for details.
Bit 4, 3, 2	filter[2:0]	Controls the time constant of the IIR filter. See Table 27 for settings and chapter 3.4.4 for details.
Bit 0	spi3w_en[0]	Enables 3-wire SPI interface when set to '1'. See chapter 6.3 for details.

Table 27: t_sb settings

t_sb[2:0]	t _{standby} [ms]
000	0.5
001	62.5
010	125
011	250
100	500
101	1000
110	10
111	20

Table 28: filter settings

filter[2:0]	Filter coefficient
000	Filter off
001	2
010	4
011	8
100, others	16

5.4.7 Register 0xF7...0xF9 "press" (_msb, _lsb, _xlsb)

The "press" register contains the raw pressure measurement output data up[19:0]. For details on how to read out the pressure and temperature information from the device, please consult chapter 4.

Table 29: Register 0xF7 ... 0xF9 "press"

Register 0xF70xF9 "press"	Name	Description
0xF7	press_msb[7:0]	Contains the MSB part up[19:12] of the raw pressure measurement output data.
0xF8	press_lsb[7:0]	Contains the LSB part up[11:4] of the raw pressure measurement output data.
0xF9 (bit 7, 6, 5, 4)	press_xlsb[3:0]	Contains the XLSB part up[3:0] of the raw pressure measurement output data. Contents depend on temperature resolution.

5.4.8 Register 0xFA...0xFC "temp" (_msb, _lsb, _xlsb)

The "temp" register contains the raw temperature measurement output data ut[19:0]. For details on how to read out the pressure and temperature information from the device, please consult chapter 4.

Table 30: Register 0xFA ... 0xFC "temp"

Register 0xFA0xFC "temp"	Name	Description
0xFA	temp_msb[7:0]	Contains the MSB part ut[19:12] of the raw temperature measurement output data.
0xFB	temp_lsb[7:0]	Contains the LSB part ut[11:4] of the raw temperature measurement output data.
0xFC (bit 7, 6, 5, 4)	temp_xlsb[3:0]	Contains the XLSB part ut[3:0] of the raw temperature measurement output data. Contents depend on pressure resolution.

5.4.9 Register 0xFD...0xFE "hum" (_msb, _lsb)

The "temp" register contains the raw temperature measurement output data ut[19:0]. For details on how to read out the pressure and temperature information from the device, please consult chapter 4.

Table 31: Register 0xFD ... 0xFE "hum"

Register 0xFD0xFE "hum"	Name	Description
0xFD	hum_msb[7:0]	Contains the MSB part uh[15:8] of the raw humidity measurement output data.
0xFE	temp_lsb[7:0]	Contains the LSB part uh[7:0] of the raw humidity measurement output data.

6. Digital interfaces

The BME280 supports the I²C and SPI digital interfaces; it acts as a slave for both protocols. The I²C interface supports the Standard, Fast and High Speed modes. The SPI interface supports both SPI mode '00' (CPOL = CPHA = '0') and mode '11' (CPOL = CPHA = '1') in 4-wire and 3-wire configuration.

The following transactions are supported:

- Single byte write
- multiple byte write (using pairs of register addresses and register data)
- single byte read
- multiple byte read (using a single register address which is auto-incremented)

6.1 Interface selection

Interface selection is done automatically based on CSB (chip select) status. If CSB is connected to V_{DDIO} , the I²C interface is active. If CSB is pulled down, the SPI interface is activated. After CSB has been pulled down once (regardless of whether any clock cycle occurred), the I²C interface is disabled until the next power-on-reset. This is done in order to avoid inadvertently decoding SPI traffic to another slave as I²C data. Since the device startup is deferred until both V_{DD} and V_{DDIO} are established, there is no risk of incorrect protocol detection because of the power-up sequence used. However, if I²C is to be used and CSB is not directly connected to V_{DDIO} but is instead connected to a programmable pin, it must be ensured that this pin already outputs the V_{DDIO} level during power-on-reset of the device. If this is not the case, the device will be locked in SPI mode and not respond to I²C commands.

6.2 I²C Interface

The I²C slave interface is compatible with Philips I²C Specification version 2.1. For detailed timings, please review Table 33. All modes (standard, fast, high speed) are supported. SDA and SCL are not pure open-drain. Both pads contain ESD protection diodes to VDDIO and GND. As the devices does not perform clock stretching, the SCL structure is a high-Z input without drain capability.

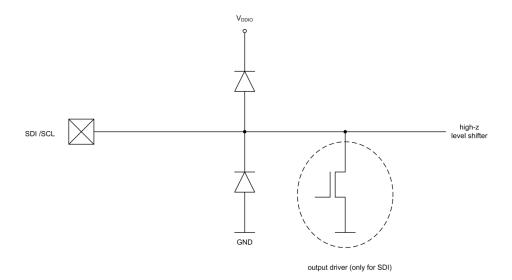


Figure 8: SDI/SCK ESD drawing

The 7-bit device address is 111011x. The 6 MSB bits are fixed. The last bit is changeable by SDO value and can be changed during operation. Connecting SDO to GND results in slave address 1110110 (0x76); connection it to V_{DDIO} results in slave address 1110111 (0x77), which is the same as

BMP280's I²C address. The SDO pin cannot be left floating; if left floating, the I²C address will be undefined.

The I²C interface uses the following pins:

SCK: serial clock (SCL)SDI: data (SDA)

• SDO: Slave address LSB (GND = '0', VDDIO = '1')

CSB must be connected to V_{DDIO} to select I²C interface. SDI is bi-directional with open drain to GND: it must be externally connected to V_{DDIO} via a pull up resistor. Refer to chapter 7 for connection instructions.

The following abbreviations will be used in the I²C protocol figures:

S Start
 P Stop

ACKS Acknowledge by slave
 ACKM Acknowledge by master
 NACKM Not acknowledge by master

6.2.1 I2C write

Writing is done by sending the slave address in write mode (RW = '0'), resulting in slave address 111011X0 ('X' is determined by state of SDO pin. Then the master sends pairs of register addresses and register data. The transaction is ended by a stop condition. This is depicted in Figure 9.

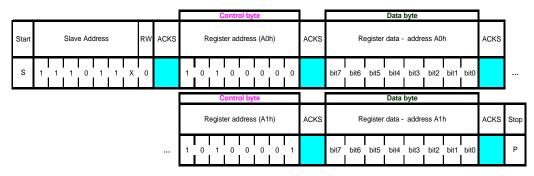


Figure 9: I2C multiple byte write (not auto-incremented)

6.2.2 I2C read

To be able to read registers, first the register address must be sent in write mode (slave address 111011X0). Then either a stop or a repeated start condition must be generated. After this the slave is addressed in read mode (RW = '1') at address 111011X1, after which the slave sends out data from auto-incremented register addresses until a NOACKM and stop condition occurs. This is depicted in Figure 10, where register 0xF6 and 0xF7 are read.

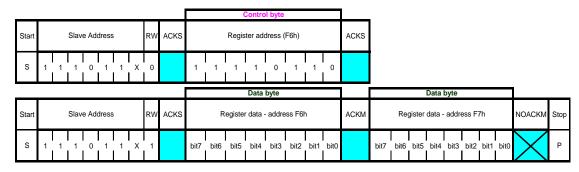


Figure 10: I2C multiple byte read

6.3 SPI interface

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 '1' to the register spi3w_en. The pad SDI is used as a data pad in 3-wire mode.

The SPI interface uses the following pins:

- CSB: chip select, active low
- SCK: serial clock
- SDI: serial data input; data input/output in 3-wire mode
- SDO: serial data output; hi-Z in 3-wire mode

Refer to chapter 7 for connection instructions.

CSB is active low and has an integrated pull-up resistor. Data on SDI 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 Figure 11. For timing details, please review Table 34.

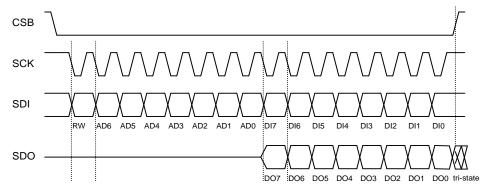


Figure 11: SPI protocol (shown for mode '11' in 4-wire configuration)

In SPI mode, only 7 bits of the register addresses are used; the MSB of register address is not used and replaced by a read/write bit (RW = '0' for write and RW = '1' for read).

Example: address 0xF7 is accessed by using SPI register address 0x77. For write access, the byte 0x77 is transferred, for read access, the byte 0xF7 is transferred.

6.3.1 SPI write

Writing is done by lowering CSB and sending pairs control bytes and register data. The control bytes consist of the SPI register address (= full register address without bit 7) and the write command (bit7 = RW = '0'). Several pairs can be written without raising CSB. The transaction is ended by a raising CSB. The SPI write protocol is depicted in Figure 12.

					Con	rol b	yte					Data byte					Control byte				Data byte														
Start	RW			Re	egiste	er ado	Iress	(F4h)				Data re	egister	addres	s F4h			RW		Re	gister	addre	ess (F	5h)				Data r	egister	- adress	s F5h			Stop
CSB		Г					Т		Т										î i																CSB
=	0		1 .	1	· 1	. 0	1		o .	0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	0	1	1	1	0	1	0	1	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	=
0		ı	- 1		ı	1	1		- 1				l	ı		l	l	1										ı		l	I 1		1 1		1

Figure 12: SPI multiple byte write (not auto-incremented)

6.3.2 SPI read

Reading is done by lowering CSB and first sending one control byte. The control bytes consist of the SPI register address (= full register address without bit 7) and the read command (bit 7 = RW = '1'). After writing the control byte, data is sent out of the SDO pin (SDI in 3-wire mode); the register address is automatically incremented. The SPI read protocol is depicted in Figure 13.

				Contr	ol byt	e				Data byte				Data byte											
Start	RW		Re	gister	addre	ess (F	6h)				Data re	gister -	addres	ss F6h				[Oata re	gister	- addre	ess F7	h		Stop
CSB = 0	1	1	1	1 	0	1 	1	0	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	CSB = 1

Figure 13: SPI multiple byte read

6.4 Interface parameter specification

6.4.1 General interface parameters

The general interface parameters are given in Table 32 below.

Table 32: interface parameters

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Input low level	V_{il_si}	V _{DDIO} =1.2 V to 3. 6V			20	%V _{DDI}
						0
Input high level	V_{ih_si}	V _{DDIO} =1.2 V to 3.6 V	80			%V _{DDI}
						0
Output low level I ² C	V _{ol_SDI}	V _{DDIO} =1.62 V, I _{ol} =3 mA			20	%V _{DDI}
						0
Output low level I ² C	V _{ol_SDI_1.2}	V _{DDIO} =1.20 V, I _{ol} =3 mA			23	%V _{DDI}
						0
Output low level SPI	V _{ol_SDO}	V _{DDIO} =1.62 V, I _{ol} =1 mA			20	%V _{DDI}
						О
Output low level SPI	$V_{ol_SDO_1.2}$	V _{DDIO} =1.20 V, I _{ol} =1 mA			23	%V _{DDI}
						0
Output high level	V_{oh}	V _{DDIO} =1.62 V, I _{oh} =1 mA	80			%V _{DDI}
		(SDO, SDI)				О
Output high level	V _{oh_1.2}	V_{DDIO} =1.20 V, I_{oh} =1 mA	60			$%V_{DDI}$
		(SDO, SDI)				0
Pull-up resistor	R _{pull}	Internal CSB pull-up	70	120	190	kΩ
		resistance to V _{DDIO}				
I ² C bus load capacitor	Сь	On SDI and SCK			400	pF

6.4.2 I2C timings

For I2C timings, the following abbreviations are used:

- "S&F mode" = standard and fast mode
- "HS mode" = high speed mode
- C_b = bus capacitance on SDA line

All other naming refers to I²C specification 2.1 (January 2000).

The I²C timing diagram is in Figure 14. The corresponding values are given in Table 33.

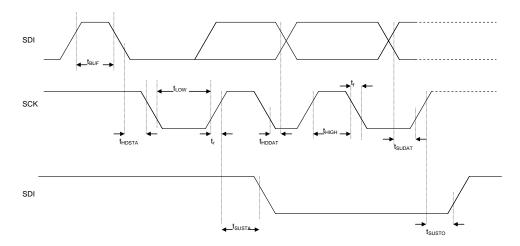


Figure 14: I2C timing diagram

Table 33: I2C timings

Parameter	Symbol	Condition	Min	Тур	Max	Unit
SDI setup time	tsu;dat	S&F Mode	160			ns
		HS mode	30			ns
SDI hold time	thd;dat	S&F Mode, C _b ≤100 pF	80			ns
		S&F Mode, C _b ≤400 pF	90			ns
		HS mode, C _b ≤100 pF	18		115	ns
		HS mode, C _b ≤400 pF	24		150	ns
SCK low pulse	t _{LOW}	HS mode, Cb≤100 pF	160			ns
		$V_{DDIO} = 1.62 \text{ V}$				
SCK low pulse	tLOW	HS mode, C _b ≤100 pF	210			ns
		$V_{DDIO} = 1.2 V$				

The above-mentioned I²C specific timings correspond to the following internal added delays:

- Input delay between SDI and SCK inputs: SDI is more delayed than SCK by typically 100 ns in Standard and Fast Modes and by typically 20 ns in High Speed Mode.
- Output delay from SCK falling edge to SDI output propagation is typically 140 ns in Standard and Fast Modes and typically 70 ns in High Speed Mode.

6.4.3 SPI timings

The SPI timing diagram is in Figure 15, while the corresponding values are given in Table 34. All timings apply both to 4- and 3-wire SPI.

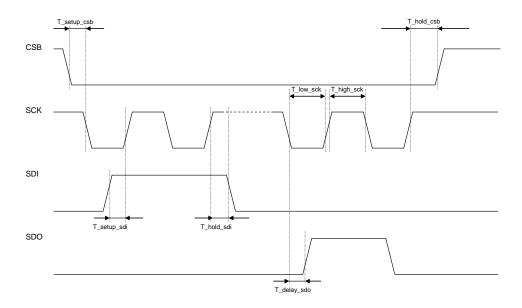


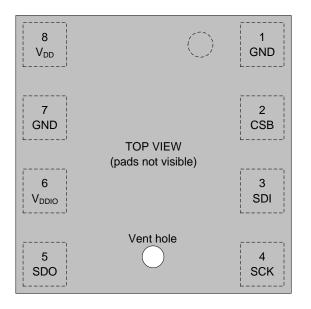
Figure 15: SPI timing diagram

Table 34: SPI timings

Parameter	Symbol	Condition	Min	Тур	Max	Unit
SPI clock input frequency	F_spi		0		10	MHz
SCK low pulse	T_low_sck		20			ns
SCK high pulse	T_high_sck		20			ns
SDI setup time	T_setup_sdi		20			ns
SDI hold time	T_hold_sdi		20			ns
SDO output delay	T_delay_sdo	25 pF load, V _{DDIO} =1.6 V min			30	ns
SDO output delay	T_delay_sdo	25 pF load, V _{DDIO} =1.2 V min			40	ns
CSB setup time	T_setup_csb		20			ns
CSB hold time	T_hold_csb		20			ns

7. Pin-out and connection diagram

7.1 Pin-out



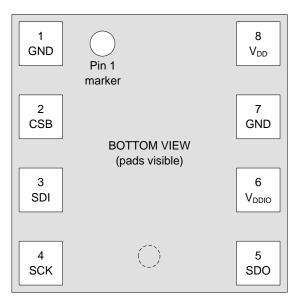


Figure 16: Pin-out top and bottom view

Note: The pin numbering of BME280 is performed in the untypical clockwise direction when seen in top view and counter-clockwise when seen in bottom view.

Table 35: Pin description

Pin	Name	I/O Type	Description		Connect to	
				SPI 4W	SPI 3W	I ² C
1	GND	Supply	Ground		GND	
2	CSB	ln	Chip select	CSB	CSB	V _{DDIO}
3	SDI	In/Out	Serial data input	SDI	SDI/SDO	SDA
4	SCK	ln	Serial clock input	SCK	SCK	SCL
5	SDO	In/Out	Serial data output	SDO	DNC	GND for default address
6	V _{DDIO}	Supply	Digital / Interface supply		V _{DDIO}	
7	GND	Supply	Ground		GND	
8	V_{DD}	Supply	Analog supply		V_{DD}	

7.2 Connection diagram I²C

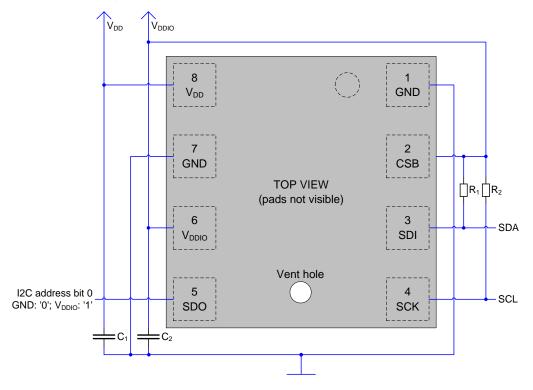


Figure 17: I2C connection diagram

Notes:

- The recommended value for C1, C2 is 100 nF
- The value for the pull-up resistors R_1 , R_2 should be based on the interface timing and the bus load; a normal value is $4.7~k\Omega$
- A direct connection between CSB and V_{DDIO} is required

7.3 Connection diagram 4-wire SPI

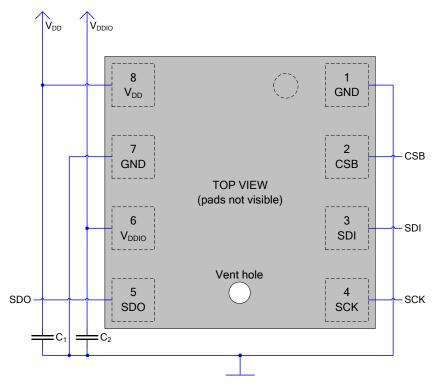


Figure 18: 4-wire SPI connection diagram

Note: The recommended value for C_1 , C_2 is 100 nF

7.4 Connection diagram 3-wire SPI

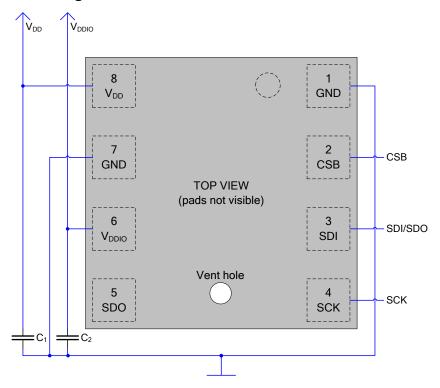


Figure 19: 3-wire SPI connection diagram

Note: The recommended value for C_1 , C_2 is 100 nF

7.5 Package dimensions

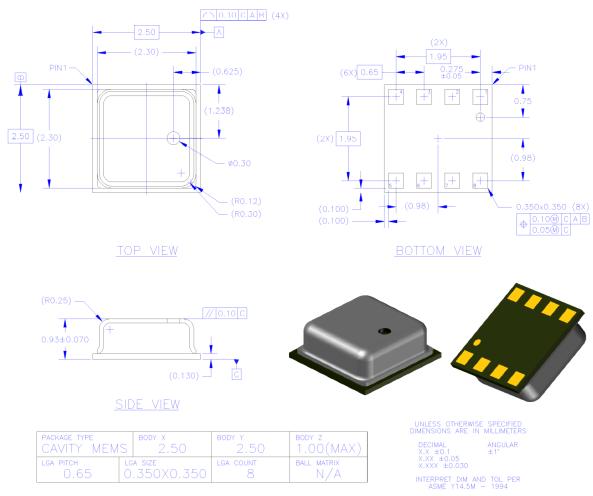


Figure 20: Package dimensions for top, bottom and side view

7.6 Landing pattern recommendation

For the design of the landing pattern, the following dimensioning is recommended:

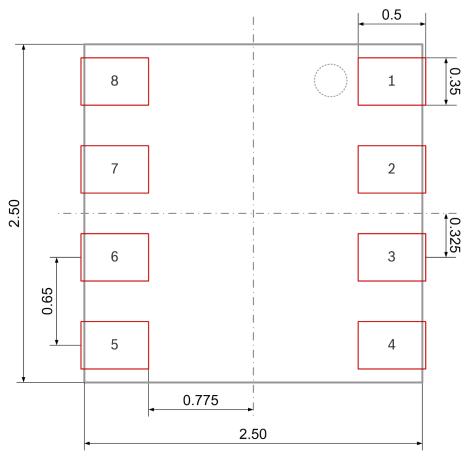


Figure 21: Recommended landing pattern (top view)

Note: red areas demark exposed PCB metal pads.

- In case of a solder mask defined (SMD) PCB process, the land dimensions should be defined by solder mask openings. The underlying metal pads are larger than these openings.
- In case of a non solder mask defined (NSMD) PCB process, the land dimensions should be defined in the metal layer. The mask openings are larger than these metal pads.

7.7 Marking

7.7.1 Mass production devices

Table 36: Marking of mass production parts

Marking	Symbol	Description
5 6 7 8	ccc	Lot counter: 3 alphanumeric digits, variable to generate mass production trace-code
Vent hole O TL Pin 1 marker	Т	Product number: 1 alphanumeric digit, fixed to identify product type, T = "U" "U" is associated with the product BME280 (part number 0 273 141 185)
4 3 2 1	L	<u>Sub-contractor ID</u> : 1 alphanumeric digit, variable to identify sub-contractor (L = "P")

7.7.2 Engineering samples

Table 37: Marking of engineering samples

Marking	Symbol	Description
	XX	Sample ID: 2 alphanumeric digits, variable to generate trace-code
Vent hole Vent Pin 1	N	Eng. Sample ID: 1 alphanumeric digit, fixed to identify engineering sample, N = " * " or "e" or "E"
4 3 2 1 marker	CC	Counter ID: 2 alphanumeric digits, variable to generate trace-code

7.8 Soldering guidelines and reconditioning recommendations

The moisture sensitivity level of the BME280 sensors corresponds to JEDEC Level 1, see also:

- IPC/JEDEC J-STD-020C "Joint Industry Standard: Moisture/Reflow Sensitivity Classification for non-hermetic Solid State Surface Mount Devices"
- IPC/JEDEC J-STD-033A "Joint Industry Standard: Handling, Packing, Shipping and Use of Moisture/Reflow Sensitive Surface Mount Devices".

The sensor fulfils the lead-free soldering requirements of the above-mentioned IPC/JEDEC standard, i.e. reflow soldering with a peak temperature up to 260°C. The minimum height of the solder after reflow shall be at least 50µm. This is required for good mechanical decoupling between the sensor device and the printed circuit board (PCB).

Profile Feature	Pb-Free Assembly
Average Ramp-Up Rate (Ts _{max} to Tp)	3° C/second max.
Preheat - Temperature Min (Ts _{min}) - Temperature Max (Ts _{max}) - Time (ts _{min} to ts _{max})	150 °C 200 °C 60-180 seconds
Time maintained above: - Temperature (T _L) - Time (t _L)	217 °C 60-150 seconds
Peak/Classification Temperature (Tp)	260 °C
Time within 5 °C of actual Peak Temperature (tp)	20-40 seconds
Ramp-Down Rate	6 °C/second max.
Time 25 °C to Peak Temperature	8 minutes max.

Note 1: All temperatures refer to topside of the package, measured on the package body surface.

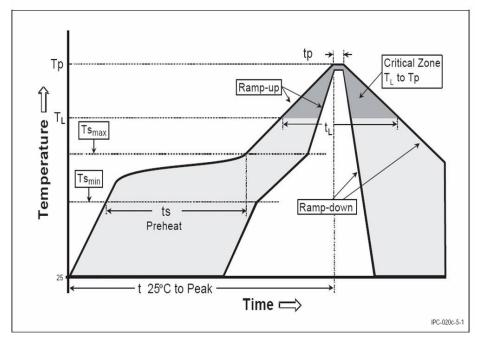


Figure 22: Soldering profile

7.9 Reconditioning Procedure

After exposing the device to operating conditions, which exceed the limits specified in section 1.2, e.g. after reflow, the humidity sensor may possess an additional offset. Therefore the following reconditioning procedure is mandatory to restore the calibration state:

1. Dry-Baking: 120 °C at <5% rH for 2 h

2. Re-Hydration: 70 °C at 75% rH for 6 h

or alternatively

Dry-Baking:
 Re-Hydration:
 120 °C at <5% rH for 2 h
 Re-Tydration:
 Re-Hydration:

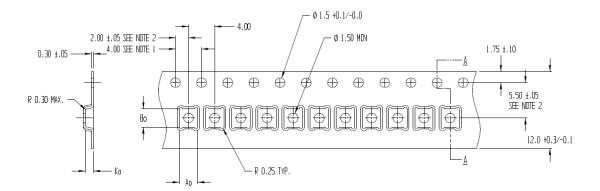
or alternatively after solder reflow only

1. Do not perform Dry-Baking

2. Ambient Re-Hydration: ~25 °C at >40% rH for >5d

7.10 Tape and reel specification

7.10.1 Dimensions



An = 2.81 Bn = 2.85 Kn = 1.20

Figure 23: Tape and Reel dimensions

Quantity per reel: 10 kpcs.

7.10.2 Orientation within the reel

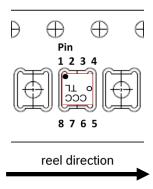


Figure 24: Orientation within tape

7.11 Mounting and assembly recommendations

In order to achieve the specified performance for you design, the following recommendations and the "Handling, soldering & mounting instructions BME280" should be taken into consideration when mounting a pressure sensor on a printed-circuit board (PCB):

- The clearance above the metal lid shall be 0.1mm at minimum.
- For the device housing appropriate venting needs to be provided in case the ambient pressure shall be measured.
- Liquids shall not come into direct contact with the device.
- During operation the sensor chip is sensitive to light, which can influence the accuracy of the measurement (photo-current of silicon). The position of the vent hole minimizes the light exposure of the sensor chip. Nevertheless, Bosch Sensortec recommends avoiding the exposure of BME280 to strong light sources.
- Soldering may not be done using vapor phase processes since the sensor will be damaged by the liquids used in these processes.

7.12 Environmental safety

7.12.1 RoHS

The BME280 sensor meets the requirements of the EC restriction of hazardous substances (RoHS) directive, see also:

Directive 2011/65/EU of the European Parliament and of the Council of 8 June 2011 on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

7.12.2 Halogen content

The BME280 is halogen-free. For more details on the analysis results please contact your Bosch Sensortec representative.

7.12.3 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 package of the BME280.

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 BME280 product.

8. Appendix A: Alternative compensation formulas

8.1 Compensation formulas in double precision floating point

Please note that it is strongly advised to use the API available from Bosch Sensortec to perform readout and compensation. If this is not wanted, the code below can be applied at the user's risk. Both pressure and temperature values are expected to be received in 20 bit format, positive, stored in a 32 bit signed integer. Humidity is expected to be received in 16 bit format, positive, stored in a 32 bit signed integer.

The variable t_fine (signed 32 bit) carries a fine resolution temperature value over to the pressure compensation formula and could be implemented as a global variable.

The data type "BME280_S32_t" should define a 32 bit signed integer variable type and could usually be defined as "long signed int". The revision of the code is rev. 1.1 (pressure and temperature) and rev. 1.0 (humidity).

Compensating the measurement value with double precision gives the best possible accuracy but is only recommended for PC applications.

```
// Returns temperature in DegC, double precision. Output value of "51.23" equals 51.23 DegC.
// t fine carries fine temperature as global value
BME280 S32_t t_fine;
double BME280_compensate_T_double(BME280_S32_t adc_T)
   double var1, var2, T;
   var1 = (((double) adc T)/16384.0 - ((double) dig T1)/1024.0) * ((double) dig T2);
   var2 = ((((double)adc_T)/131072.0 - ((double)dig_T1)/8192.0)
   (((double)adc_T)/131072.0 - ((double) dig_T1)/8192.0)) * ((double)dig_T3);
t_fine = (BME280_S32_t)(var1 + var2);
   T = (var1 + var2) / 5120.0;
   return T;
// Returns pressure in Pa as double. Output value of "96386.2" equals 96386.2 Pa = 963.862 hPa
double BME280 compensate P double (BME280 S32 t adc P)
   double var1, var2, p;
   var1 = ((double)t_fine/2.0) - 64000.0;
   var2 = var1 * var1 * ((double)dig_P6) / 32768.0;
   var2 = var2 + var1 * ((double)dig P5) * 2.0;
   var2 = (var2/4.0) + (((double)dig_P4) * 65536.0);
   var1 = (((double) dig_P3) * var1 * var1 / 524288.0 + ((double) dig_P2) * var1) / 524288.0;
var1 = (1.0 + var1 / 32768.0)*((double) dig_P1);
   if (var1 == 0.0)
      return 0; // avoid exception caused by division by zero
   p = 1048576.0 - (double) adc P;
   p = (p - (var2 / 4096.0)) * 6250.0 / var1;
  var1 = ((double) dig_P9) * p * p / 2147483648.0;
var2 = p * ((double) dig_P8) / 32768.0;
   p = p + (var1 + var2 + ((double)dig P7)) / 16.0;
   return p;
// Returns humidity in %rH as as double. Output value of "46.332" represents
double bme280 compensate_H_double(BME280_S32_t adc_H);
   double var H;
   var_H = (((double)t_fine) - 76800.0);
   var_H = (adc_H - (((double)dig_H4) * 64.0 + ((double)dig_H5) / 16384.0 *
      var_H)) * (((double)dig_H2) / 65536.0 * (1.0 + ((double)dig_H6) /
      67108864.0 * var H *
      (1.0 + ((double)\overline{dig} H3) / 67108864.0 * var H)));
   var_H = var_H * (1.0 - ((double)dig_H1) * var_H / 524288.0);
   if (var H > 100.0)
      var_H = 100.0;
   else i\bar{f} (var H < 0.0)
      var H = 0.0;
return var_H;
```

8.2 Pressure compensation in 32 bit fixed point

Please note that it is strongly advised to use the API available from Bosch Sensortec to perform readout and compensation. If this is not wanted, the code below can be applied at the user's risk. Both pressure and temperature values are expected to be received in 20 bit format, positive, stored in a 32 bit signed integer.

The variable t_fine (signed 32 bit) carries a fine resolution temperature value over to the pressure compensation formula and could be implemented as a global variable.

The data type "BME280_S32_t" should define a 32 bit signed integer variable type and can usually be defined as "long signed int".

The data type "BME280_U32_t" should define a 32 bit unsigned integer variable type and can usually be defined as "long unsigned int".

Compensating the pressure value with 32 bit integer has an accuracy of typically 1 Pa (1-sigma). At high filter levels this adds a significant amount of noise to the output values and reduces their resolution.

```
// Returns temperature in DegC, resolution is 0.01 DegC. Output value of "5123" equals 51.23
// t fine carries fine temperature as global value
BME280 S32 t t_fine;
BME280_S32_t BME280_compensate_T_int32(BME280_S32_t adc_T)
   BME280 S32 t var1, var2, T;
   var1 = ((((adc_T>>3) - ((BME280_S32_t)dig_T1<<1))) * ((BME280_S32_t)dig_T2)) >> 11;
var2 = (((((adc_T>>4) - ((BME280_S32_t)dig_T1)) * ((adc_T>>4) - ((BME280_S32_t)dig_T1)))
      ((BME280 S32 t)dig T3)) >> 14;
   t_fine = var1 + var2;
   T = (t fine * 5 + 128) >> 8;
   return T;
// Returns pressure in Pa as unsigned 32 bit integer. Output value of "96386" equals 96386 Pa
= 963.86 \text{ hPa}
BME280 U32 t BME280 compensate P int32(BME280 S32 t adc P)
   BME280_S32_t var1, var2;
   BME280 U32_t p;
   var1 = (((BME280_S32_t)t_fine)>>1) - (BME280_S32_t)64000;
var2 = (((var1>>2) * (var1>>2)) >> 11) * ((BME280_S32_t)dig_P6);
   var2 = var2 + ((var1*((BME280 S32 t)dig P5))<<1);</pre>
   var2 = (var2>>2) + (((BME280_S32_t)dig_P4) << 16);
   var1 = (((dig_P3 * (((var1>>2) * (var1>>2)) >> 13 )) >> 3) + ((((BME280_S32_t)dig_P2) *
var1)>>1))>>18;
   var1 = ((((32768+var1))*((BME280 S32 t)dig P1))>>15);
   if (var1 == 0)
      return 0; // avoid exception caused by division by zero
   p = (((BME280\_U32\_t)(((BME280\_S32\_t)1048576)-adc\_P)-(var2>>12)))*3125;
   if (p < 0x80000000)
      p = (p << 1) / ((BME280 U32 t)var1);
   else
      p = (p / (BME280 U32 t) var1) * 2;
   var1 = (((BME280_S32_t)dig_P9) * ((BME280_S32_t)(((p>>3) * (p>>3))>>13)))>>12;
   var2 = (((BME280_S32_t)(p>>2)) * ((BME280_S32_t)dig_P8))>>13;
   p = (BME280\_U32\_t)((BME280\_S32\_t)p + ((var1 + var2 + dig\_P7) >> 4));
   return p;
```

9. Appendix B: Measurement time and current calculation

In this chapter, formulas are given to calculate measurement rate, filter bandwidth and current consumption in different settings.

9.1 Measurement time

The active measurement time depends on the selected values for humidity, temperature and pressure oversampling and can be calculated in milliseconds using the formulas below.

$$t_{measure,typ} = 1 + [2 \cdot T_oversampling]_{osrs_t \neq 0} + [2 \cdot P_oversampling + 0.5]_{osrs_p \neq 0} + [2 \cdot H_oversampling + 0.5]_{osrs_h \neq 0}$$

$$t_{measure,max} = 1.25 + [2.3 \cdot T_oversampling]_{osrs_t \neq 0} + [2.3 \cdot P_oversampling + 0.575]_{osrs_p \neq 0} + [2.3 \cdot H_oversampling + 0.575]_{osrs_p \neq 0}$$

For example, using temperature oversampling ×1, pressure oversampling ×4 and no humidity measurement, the measurement time is:

$$t_{measure,typ} = 1 + [2 \cdot 1] + [2 \cdot 4 + 0.5] + [0] = 11.5 \text{ ms}$$

$$t_{measure,max} = 1.25 + [2.3 \cdot 1] + [2.3 \cdot 4 + 0.575] + [0] = 13.325 \text{ ms}$$

9.2 Measurement rate in forced mode

In forced mode, the measurement rate depends on the rate at which it is forced by the master. The highest possible frequency in Hz can be calculated as:

$$ODR_{max,forced} = \frac{1000}{t_{measure}}$$

If measurements are forced faster than they can be executed, the data rate saturates at the attainable data rate. For the example above with 11.5 ms measurement time, the typically achievable output data rate would be:

$$ODR_{max,forced} = \frac{1000}{11.5} = 87 \text{ Hz}$$

9.3 Measurement rate in normal mode

The measurement rate in normal mode depends on the measurement time and the standby time and can be calculated in Hz using the following formula:

$$ODR_{normal_mode} = \frac{1000}{t_{measure} + t_{standby}}$$

The accuracy of $t_{standby}$ is described in the specification parameter $\Delta t_{standby}$. For the example above with 11.5 ms measurement time, setting normal mode with a standby time of 62.5 ms would result in a data rate of:

$$ODR_{normal_mode} = \frac{1000}{11.5 + 62.5} = 13.51 \text{ Hz}$$

9.4 Response time using IIR filter

When using the IIR filter, the response time of the sensor depends on the selected filter coefficient and the data rate used. It can be calculated using the following formula:

$$t_{response, 75\%} = \frac{1000 \cdot n_{samples, 75\%}}{ODR}$$

For the example above with a data rate of 13.51 Hz, the user could select a filter coefficient of 8. According to Table 6, the number of samples needed to reach 75% of a step response using this filter setting is 11. The response time with filter is therefore:

$$t_{response, 75\%} = \frac{1000 \cdot 11}{13.51} = 814 \text{ ms}$$

9.5 Current consumption

The current consumption depends on the selected oversampling settings, the measurement rate and the sensor mode, but not on the IIR filter setting. It can be calculated as:

$$\begin{split} I_{DD,forced} &= I_{DDSL} \cdot (1 - t_{measure} \cdot ODR) + \frac{ODR}{1000} \\ & \cdot \left(205 + I_{DDT} \cdot [2 \cdot T_oversampling]_{osrs_t \neq 0} + I_{DDP} \cdot [2 \cdot P_oversampling + 0.5]_{osrs_p \neq 0} \\ & + I_{DDH} \cdot [2 \cdot H_oversampling + 0.5]_{osrs_h \neq 0} \right) \end{split}$$

$$\begin{split} I_{DD,normal} &= I_{DDSB} \cdot (1 - t_{measure} \cdot ODR) + \frac{ODR}{1000} \\ & \cdot \left(205 + I_{DDT} \cdot [2 \cdot T_oversampling]_{osrs_t \neq 0} + I_{DDP} \cdot [2 \cdot P_oversampling + 0.5]_{osrs_p \neq 0} \\ & + I_{DDH} \cdot [2 \cdot H_oversampling + 0.5]_{osrs_h \neq 0} \right) \end{split}$$

Note that the only difference between forced and normal mode current consumption is that the current for the inactive time is either IDDSL or IDDSB. For the example above, the current would be

$$\begin{split} I_{DD,normal} &= 0.2 \cdot (1 - 0.0115 \cdot 13.51) + \frac{13.51}{1000} (205 + 350 \cdot [2 \cdot 1] + 714 \cdot [2 \cdot 4 + 0.5] + [0]) \\ &= 0.2 \cdot (0.845) + \frac{13.51}{1000} (205 + 700 + 6069 + 0) \\ &= 0.2 + 94.2 = 94.4 \, \mu \text{A} \end{split}$$

10. Legal disclaimer

10.1 Engineering samples

Engineering Samples are marked with an asterisk (*) 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.

10.2 Product use

Bosch Sensortec products are developed for the consumer goods industry. They are not designed or approved for use in military applications, life-support appliances, safety-critical automotive applications and devices or systems where malfunctions of these products can reasonably be expected to result in personal injury. They may only be used within the parameters of this product data sheet.

The resale and/or use of products are at the Purchaser's own risk and the Purchaser's own responsibility.

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 inform Bosch Sensortec without delay of any security relevant incidents.

10.3 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.

11. Document history and modification

Rev. No	Page	Description of modification/changes	Date
0.1		Document creation	2012-11-06
1.0		Final datasheet	2014-11-12
1.1	48	Updated RoHS directive to 2011/65/EU effective 8 June 2011	2015-05-07
1.2	2, 3	Adjusted target devices, applications	2015-10-15
1.4		Minor corrections	2018-01-17
1.5		Template update	2018-09-17



Bosch Sensortec GmbH

Gerhard-Kindler-Straße 9 72770 Reutlingen / Germany

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

Modifications reserved
Preliminary - specifications subject to change without notice
Document number: BST-BME280-DS002-15
Revision_1.6_092018

Specification

for

LCD Module

1602A-1 (V1.2)

1. 0 FEATURES

• Display Mode: STN, BLUB

• Display Formate: 16 Character x 2 Line

• Viewing Direction: 6 O'Clock

• Input Data: 4-Bits or 8-Bits interface available

• Display Font : 5 x 8 Dots

• Power Supply : Single Power Supply (5V±10%)

Driving Scheme: 1/16Duty,1/5Bias
BACKLIGHT (SIDE): LED (WHITE)

2.0 ABSOLUTE MAXIMUM

Item	Symbol	Min.	Max.	Unit
Power Supply for logic	Vdd	-0.3	+7.0	V
Power supply for LCD Drive	Vlcd	Vdd-10.0	Vdd+0.3	V
Input Voltage	Vi	-0.3	Vdd+0.3	V
Operating Temperature	Ta	0	+50	$^{\circ}$
Storage Temperature	Tstg	-10	+60	$^{\circ}$

3.0ELECTRICAL CHARACTERISTICS

 $(Ta=25^{\circ}C;Vdd=3.0V\pm10\%,otherwise specified)$

Item	Symbol	Test Condition	Min.	Тур.	Max.	Unit
Power Supply for Logic	Vdd		4.7	5.0	5.5	V
Operating Voltage for LCD	Vdd-Vo			5.0		V
Input High voltage	Vih		2.2		Vdd	V
Input Low voltage	Vil		-0.3		0.6	V
Output High voltage	Voh	-Ioh=0.2mA	2.4			V
Output Low voltage	Vol	Iol=1.2mA			0.4	V
Power supply current	Idd	Vdd=3.0v		1.1		mA

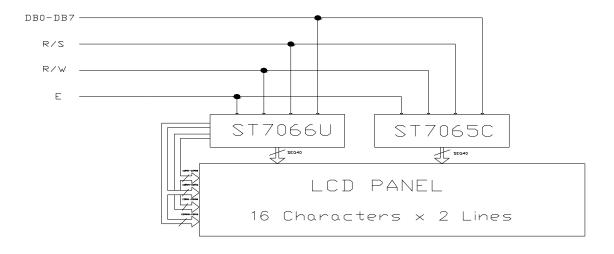
4.0 MECHANICAL PARAMETERS

Item	Description	Unit
PCB Dimension	80.0*36.0*1.6	mm
View Dimension	69.5*14.5	mm

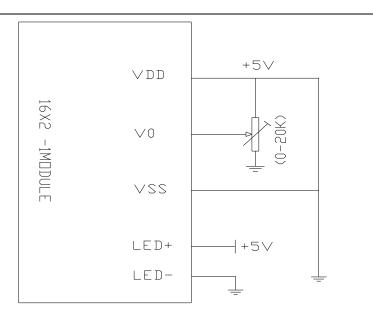
5. 0 PIN ASSIGNMENT

No.	Symbol	Level		Function						
1	Vss		0V							
2	Vdd		+5V	Power Supply						
3	V0		for LCD							
4	RS	H/L	Register Select: H:Data Input L:Instruction Inpu							
5	R/W	H/L	HRead LWrite							
6	Е	H,H-L		Enable Signal						
7	DB0	H/L								
8	DB1	H/L								
9	DB2	H/L	Data bu	s used in 8 bit transfer						
10	DB3	H/L								
11	DB4	H/L								
12	DB5	H/L	Data bus fo	or both 4 and 8 bit transfer						
13	DB6	H/L								
14	DB7	H/L								
15	BLA		BL	ACKLIGHT +5V						
16	BLK		BL	ACKLIGHT 0V-						

6.0 BLOCK DIAGRAM



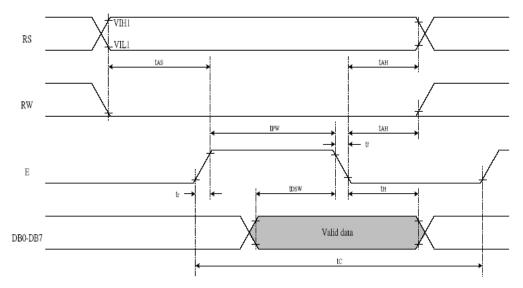
7.0 POWER SUPPLY BLOCK DIAGRAM



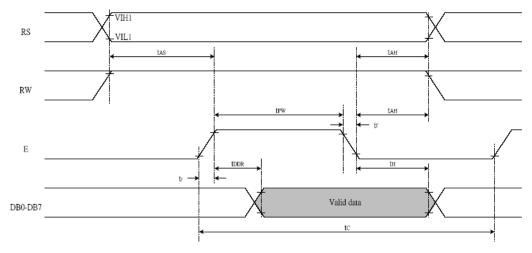
8.0 TIMING CHARACTERISTICS

	Write Mod	e (Writing data from MPU t	o ST706	6U)		
T _C	Enable Cycle Time	Pin E	1200	-	-	ns
T _{PW}	Enable Pulse Width	Pin E	140	-	-	ns
T_R, T_F	Enable Rise/Fall Time	Pin E	,	-	25	ns
T _{AS}	Address Setup Time	Pins: RS,RW,E	0	1	1	ns
T_AH	Address Hold Time	Pins: RS,RW,E	10	•	·	ns
T_{DSW}	Data Setup Time	Pins: DB0 - DB7	40	ı	ı	ns
T_H	Data Hold Time	Pins: DB0 - DB7	10	ı	ı	ns
	Read Mode	(Reading Data from ST70	66U to N	IPU)		
T _C	Enable Cycle Time	Pin E	1200	ı	ı	ns
T_{PW}	Enable Pulse Width	Pin E	140	ı	ı	ns
T_R, T_F	Enable Rise/Fall Time	Pin E	,	ı	25	ns
T _{AS}	Address Setup Time	Pins: RS,RW,E	0	1	1	ns
T _{AH}	Address Hold Time	Pins: RS,RW,E	10	-	-	ns
T _{DDR}	Data Setup Time	Pins: DB0 - DB7	1	-	100	ns
T _H	Data Hold Time	Pins: DB0 - DB7	10	-	-	ns

Writing data from MPU to ST7066U



Reading data from ST7066U to MPU



9. 0 Display control instruction

The display control instructions control the internal state of the ST7066U-0A. Instruction is received from MPU to ST7066U-0A for the display control. The following table shows various instructions.

There are four categories of instructions that:

- Designate ST7066U functions, such as display format, data length, etc.
- Set internal RAM addresses
- Perform data transfer with internal RAM
- Others

Instruction Table:

nstruction rad				Inst	ructi	on C	ode)				Description
Instruction	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Time (270KHz)
Clear Display	0	0	0	0	0	0	0	0	0	1	Write "20H" to DDRAM. and set DDRAM address to "00H" from AC	1.52 ms
Return Home	0	0	0	0	0	0	0	0	1	х	Set DDRAM address to "00H" from AC and return cursor to its original position if shifted. The contents of DDRAM are not changed.	1.52 ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	Ø	Sets cursor move direction and specifies display shift. These operations are performed during data write and read.	37 us
Display ON/OFF	0	0	0	0	0	0	1	D	O	В	D=1:entire display on C=1:cursor on B=1:cursor position on	37 us
Cursor or Display Shift	0	0	0	0	0	1	S/C	R/L	х	х	Set cursor moving and display shift control bit, and the direction, without changing DDRAM data.	37 us
Function Set	0	0	0	0	1	DL	N	F	х	х	DL:interface data is 8/4 bits N:number of line is 2/1 F:font size is 5x11/5x8	37 us
Set CGRAM address	0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0	Set CGRAM address in address counter	37 us
Set DDRAM address	0	0	1	AC6	AC5	AC4	AC3	AC2	AC1	AC0	address counter	37 us
Read Busy flag and address	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Whether during internal operation or not can be known by reading BF. The contents of address counte can also be read.	0 us
Write data to RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Write data into internal RAM (DDRAM/CGRAM)	37 us
Read data from RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Read data from internal RAM (DDRAM/CGRAM)	37 us

Note

Be sure the ST7066U is not in the busy state (BF = 0) before sending an instruction from the MPU to the ST7066U. If an instruction is sent without checking the busy flag, the time between the first instruction and next instruction will take much longer than the instruction time itself. Refer to Instruction Table for the list of each instruction execution time.

■ Instruction Description

Clear Display

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0

Code 0 0 0 0 0 0 0 0 0 1

Clear all the display data by writing "20H" (space code) to all DDRAM address, and set DDRAM address to "00H" into AC (address counter). Return cursor to the original status, namely, bring the cursor to the left edge on first line of the display. Make entry mode increment (I/D = "1").

Return Home

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0

Code 0 0 0 0 0 0 0 0 1 x

Return Home is cursor return home instruction. Set DDRAM address to "00H" into the address counter. Return cursor to its original site and return display to its original status, if shifted. Contents of DDRAM does not change.

Entry Mode Set

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0

Code 0 0 0 0 0 0 0 1 1 1/D S

Set the moving direction of cursor and display.

> I/D : Increment / decrement of DDRAM address (cursor or blink)

When I/D = "High", cursor/blink moves to right and DDRAM address is increased by 1.

When I/D = "Low", cursor/blink moves to left and DDRAM address is decreased by 1.

* CGRAM operates the same as DDRAM, when read from or write to CGRAM.

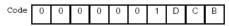
S: Shift of entire display

When DDRAM read (CGRAM read/write) operation or S = "Low", shift of entire display is not performed. If S = "High" and DDRAM write operation, shift of entire display is performed according to I/D value (I/D = "1" : shift left, I/D = "0" : shift right).

s	I/D	Description
Н	Н	Shift the display to the left
Н	L	Shift the display to the right

Display ON/OFF

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0



Control display/cursor/blink ON/OFF 1 bit register.

> D: Display ON/OFF control bit

When D = "High", entire display is turned on.

When D = "Low", display is turned off, but display data is remained in DDRAM.

> C: Cursor ON/OFF control bit

When C = "High", cursor is turned on.

When C = "Low", cursor is disappeared in current display, but I/D register remains its data.

➤ B: Cursor Blink ON/OFF control bit

When B = "High", cursor blink is on, that performs alternate between all the high data and display character at the cursor position.

When B = "Low", blink is off.

Cursor or Display Shift

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0

Code	0	0	0	0	0	1	S/C	R/I	x	×
		~		_	_	٠.	0,0		^	,,

Without writing or reading of display data, shift right/left cursor position or display. This instruction is used to correct or search display data. During 2-line mode display, cursor moves to the 2nd line after 40th digit of 1st line. Note that display shift is performed simultaneously in all the line. When displayed data is shifted repeatedly, each line shifted individually. When display shift is performed, the contents of address counter are not changed.

S/C	R/L	Description	AC Value
L	L	Shift cursor to the left	AC=AC-1
L	Н	Shift cursor to the right	AC=AC+1
Н	L	Shift display to the left. Cursor follows the display shift	AC=AC
Н	Н	Shift display to the right. Cursor follows the display shift	AC=AC

Function Set

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0

Code 0 0 0 0 1 DL N F x x

DL : Interface data length control bit

When DL = "High", it means 8-bit bus mode with MPU.

When DL = "Low", it means 4-bit bus mode with MPU. So to speak, DL is a signal to select 8-bit or 4-bit bus mode.

When 4-bit bus mode, it needs to transfer 4-bit data by two times.

N: Display line number control bit

When N = "Low", it means 1-line display mode.

When N = "High", 2-line display mode is set.

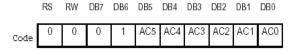
F: Display font type control bit

When F = "Low", it means 5 x 8 dots format display mode

When F = "High", 5 x11 dots format display mode.

N	F	No. of Display Lines	Character Font	Duty Factor
L	L	1	5x8	1/8
L	Н	1	5x11	1/11
Н	Х	2	5x8	1/16

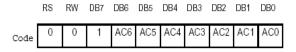
Set CGRAM Address



Set CGRAM address to AC.

This instruction makes CGRAM data available from MPU.

Set DDRAM Address



Set DDRAM address to AC.

This instruction makes DDRAM data available from MPU.

When 1-line display mode (N = 0), DDRAM address is from "00H" to "4FH".

In 2-line display mode (N = 1), DDRAM address in the 1st line is from "00H" to "27H", and DDRAM address in the 2nd line is from "40H" to "67H".

Read Busy Flag and Address

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0

Code 0 1 BF AC6 AC5 AC4 AC3 AC2 AC1 AC0

When BF = "High", indicates that the internal operation is being processed. So during this time the next instruction cannot be accepted.

The address Counter (AC) stores DDRAM/CGRAM addresses, transferred from IR.

After writing into (reading from) DDRAM/CGRAM, AC is automatically increased (decreased) by 1.

Write Data to CGRAM or DDRAM

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0



Write binary 8-bit data to DDRAM/CGRAM.

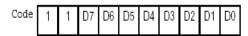
The selection of RAM from DDRAM, CGRAM, is set by the previous address set instruction

: DDRAM address set, CGRAM address set. RAM set instruction can also determine the AC direction to RAM.

After write operation, the address is automatically increased/decreased by 1, according to the entry mode.

Read Data from CGRAM or DDRAM

RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0



Read binary 8-bit data from DDRAM/CGRAM.

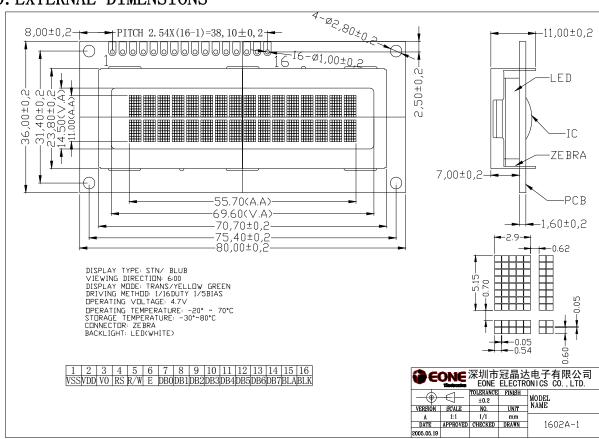
The selection of RAM is set by the previous address set instruction. If address set instruction of RAM is not performed before this instruction, the data that read first is invalid, because the direction of AC is not determined. If you read RAM data several times without RAM address set instruction before read operation, you can get correct RAM data from the second, but the first data would be incorrect, because there is no time margin to transfer RAM data.

In case of DDRAM read operation, cursor shift instruction plays the same role as DDRAM address set instruction: it also transfer RAM data to output data register. After read operation address counter is automatically increased/decreased by 1 according to the entry mode. After CGRAM read operation, display shift may not be executed correctly.

* In case of RAM write operation, after this AC is increased/decreased by 1 like read operation. In this time, AC indicates the next address position, but you can read only the previous data by read instruction.

67-64 63-60	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
	CG RAM (1)															
001	(2)															
1010	(3)															
011	(4)															
100	(5)															
)101	(6)															
)110	6															
)111	(8)															
1000	(1)															
1001	(2)															
1010	(3)															
1011	(4)															
1100	(5)															
1101	(6)															
1110	(7)															
1111	(8)															

9. EXTERNAL DIMENSIONS



Command	Code	Description	execution time
Clear Display	0x01		$82\mu S$
Display on/off	$0x0Y\rightarrow[DCB]$	Y=1DCB (Binary), D:Display on(1)/off(0), C:Cursor on(1)/off(0),	$37\mu S$
Cursor/Display Shift	0x0[1 S/C R/L * *]	B:Cursor Blink on(1), off(0) See the data sheet	$37\mu S$
Function Sets	For detail	See the LCD 1602A data sheet	

Table 2: Command and control function for LCD Display (See data sheet for detail)