

# SCD42

Breaking the size barrier in CO<sub>2</sub> sensing



### Features

- Photoacoustic sensor technology PASens®
- Smallest form factor: 10.1 x 10.1 x 6.5 mm<sup>3</sup>
- Reflow solderable for cost effective assembly
- Large output range: 0 ppm – 40'000 ppm
- Large supply voltage range: 2.4 – 5.5 V
- Accuracy: ± 75 ppm @ 400 ppm – 1'000 ppm
- Digital I<sup>2</sup>C interface
- Integrated temperature and humidity sensor

### Product Summary

The SCD42 is Sensirion's next generation miniature CO<sub>2</sub> sensor. This sensor builds on the photoacoustic sensing principle and Sensirion's patented PASens® and CMOSens® technology to offer high accuracy at an unmatched price and smallest form factor. SMD assembly allows cost- and space-effective integration of the sensor combined with maximal freedom of design. On-chip signal compensation is realized with the build-in SHT4x humidity and temperature sensor.

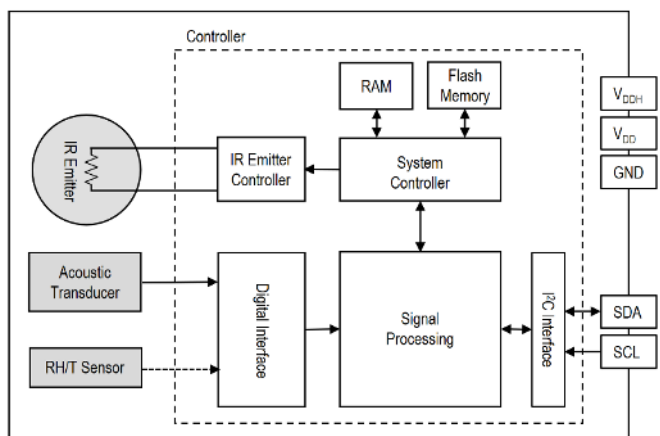
CO<sub>2</sub> is a key indicator for indoor air quality as high levels compromise humans' cognitive performance and well-being. The SCD42 enables smart ventilation systems to regulate ventilation in the most energy-efficient and human-friendly way. Moreover, indoor air quality monitors and other connected devices based on the SCD42 can help maintaining low CO<sub>2</sub> concentration for a healthy, productive environment.

### Device Overview

| Products   | Details                              |
|------------|--------------------------------------|
| SCD42-D-R2 | Accuracy: ± 75 ppm @ 400 – 1'000 ppm |

Full product list on page 20

### Functional Block Diagram



**Table of Contents**

**1 Sensor Performance .....3**

1.1 CO<sub>2</sub> Sensing Performance .....3

1.2 Humidity Sensing Performance .....3

1.3 Temperature Sensing Performance .....3

**2 Specifications .....4**

2.1 Electrical Specifications .....4

2.2 Absolute Maximum Ratings .....4

2.3 Interface Specifications .....5

2.4 Timing Specifications .....6

2.5 Material Contents .....6

**3 Digital Interface Description .....7**

3.1 Power-Up and Communication Start .....7

3.2 Data type & length .....7

3.3 Command Sequence Types .....7

3.4 SCD42 Command Overview .....8

3.5 Basic Commands .....9

3.6 On-Chip Output Signal Compensation .....10

3.7 Field Calibration .....13

3.8 Advanced Features .....15

3.9 Checksum Calculation .....17

**4 Mechanical specifications .....18**

4.1 Package Outline .....18

4.2 Land Pattern .....18

4.3 Tape & Reel Package .....19

4.4 Moisture Sensitivity Level .....19

4.5 Soldering Instructions .....20

4.6 Traceability .....20

**5 Ordering Information .....20**

**6 Revision History .....20**

# 1 Sensor Performance

## 1.1 CO<sub>2</sub> Sensing Performance

Default conditions of 25 °C, 50 % RH, ambient pressure 1013 mbar and 3.3 V supply voltage apply to values in the table below, unless otherwise stated.

| Parameter  | Conditions              | Value                        |
|--|-------------------------|------------------------------|
| CO <sub>2</sub> output range <sup>1</sup>  | -                       | 0 – 40'000 ppm               |
| SCD42 CO <sub>2</sub> measurement accuracy <sup>2</sup>  | 400 ppm – 1'000 ppm     | ± 75 ppm                     |
|  | 1'001 ppm – 2'000 ppm   | ± (40 ppm + 5% of reading)   |
| Repeatability  | Typical                 | ± 10 ppm                     |
| Response time <sup>3</sup>   | $\tau_{63\%}$ , typical | 60 s                         |
| Additional accuracy drift per year after five years of sensor operation and with automatic self-calibration algorithm enabled <sup>4</sup> | Typical                 | ± (5 ppm + 0.5 % of reading) |

**Table 1:** SCD42 CO<sub>2</sub> sensor specifications

## 1.2 Humidity Sensing Performance<sup>5</sup>

| Parameter                  | Conditions                      | Value             |
|----------------------------|---------------------------------|-------------------|
| Humidity measurement range | -                               | 0 %RH – 100 %RH   |
| Accuracy (typ.)            | 15 °C – 35 °C, 20 %RH – 65 %RH  | ± 6 % RH          |
|                            | -10 °C – 60 °C, 0 %RH – 100 %RH | ± 9 % RH          |
| Repeatability              | Typical                         | ± 0.4 %RH         |
| Response time <sup>3</sup> | $\tau_{63\%}$ , typical         | 90 s              |
| Accuracy drift             | -                               | < 0.25 %RH / year |

**Table 2:** SCD42 humidity sensor specifications

## 1.3 Temperature Sensing Performance<sup>5</sup>

| Parameter                     | Conditions              | Value            |
|-------------------------------|-------------------------|------------------|
| Temperature measurement range | -                       | - 10°C – 60°C    |
| Accuracy (typ.)               | 15 °C – 35 °C           | ± 0.8 °C         |
|                               | -10 °C – 60 °C          | ± 1.5 °C         |
| Repeatability                 | -                       | ± 0.1°C          |
| Response time <sup>3</sup>    | $\tau_{63\%}$ , typical | 120 s            |
| Accuracy drift                | -                       | < 0.03 °C / year |

**Table 3:** SCD42 temperature sensor specifications

<sup>1</sup> Exposure to CO<sub>2</sub> concentrations smaller than 400 ppm can affect the accuracy of the sensor if the automatic self-calibration (ASC) is on.

<sup>2</sup> Deviation to a high-precision reference. Accuracy is fulfilled by > 90% of the sensors after calibration. Rough handling, shipping and soldering reduces the accuracy of the sensor. Accuracy is restored with FRC or ASC recalibration features. Accuracy is based on tests with gas mixtures having a tolerance of ± 1.5%.

<sup>3</sup> Time for achieving 63% of a respective step function when operating the SCD42 Evaluation Kit with default measurement mode. Response time depends on design-in, signal update rate and environment of the sensor in the final application.

<sup>4</sup> For proper function of ASC field-calibration algorithm SCD42 has to be exposed to air with CO<sub>2</sub> concentration 400 ppm regularly. Maximum accuracy drift per year estimated from stress tests is ± (5 ppm + 2 % of reading). Higher drift values may occur if the sensor is not handled according to its handling instructions.

<sup>5</sup> Design-in of the SCD42 in final application, self-heating of the sensor and the environment impacts the accuracy of the RH/T sensor. To realize indicated specifications, the temperature-offset of the SCD42 inside the customer device must be set correctly (see chapter 3.6).

## 2 Specifications

### 2.1 Electrical Specifications

| Parameter                                       | Symbol            | Conditions              | Min.                   | Typical | Max.                  | Units |
|---|-------------------|-------------------------|------------------------|---------|-----------------------|-------|
| Supply voltage DC <sup>6</sup>                  | V <sub>DD</sub>   |                         | 2.4                    | 3.3     | 5.5                   | V     |
| Voltage ripple peak to peak                     | V <sub>RPP</sub>  | Frequency ≥ 1 kHz       |                        |         | 30                    | mV    |
| Supply voltage variation <sup>7</sup>           | V <sub>Var</sub>  | Frequency < 1 kHz       |                        |         | ±5                    | mV    |
| Peak supply current <sup>8</sup>                | I <sub>peak</sub> | V <sub>DD</sub> = 3.3 V |                        | 175     | 205                   | mA    |
|   |                   | V <sub>DD</sub> = 5 V   |                        | 115     | 137                   | mA    |
| Average supply current for periodic measurement | I <sub>DD</sub>   | V <sub>DD</sub> = 3.3 V |                        | 15      | 18                    | mA    |
|   |                   | V <sub>DD</sub> = 5 V   |                        | 11      | 13                    | mA    |
| Input high level voltage                        | V <sub>IH</sub>   |                         | 0.65 x V <sub>DD</sub> |         | V <sub>DD</sub>       | -     |
| Input low level voltage                         | V <sub>IL</sub>   |                         |                        |         | 0.3 x V <sub>DD</sub> | -     |
| Output low level voltage                        | V <sub>OL</sub>   | 3 mA sink current       |                        |         | 0.66                  | V     |

**Table 4** SCD42 electrical specifications

### 2.2 Absolute Maximum Ratings

Stress levels beyond those listed in **Table 5** may cause permanent damage to the device. Exposure to minimum/maximum rating conditions for extended periods may affect sensor performance and reliability of the device.

| Parameter                                    | Conditions   | Value                             |
|--|--|-----------------------------------|
| Temperature operating conditions             |  | 0 – 50°C                          |
| Humidity operating conditions <sup>9</sup>   | Non-condensing   | 0 – 95 %RH                        |
| MSL Level <sup>10</sup>                      |  | 1                                 |
| DC supply voltage                            |  | - 0.3 V – 6.0 V                   |
| Max voltage on pins SDA, SCL, GND            |  | - 0.3 V to V <sub>DD</sub> +0.3 V |
| Input current on pins SDA, SCL, GND          |  | - 280 mA to 100 mA                |
| Short term storage temperature <sup>11</sup> |  | - 40°C – 70°C                     |
| Recommended storage temperature              |  | 10 °C – 30 °C                     |
| ESD HBM                                      |  | 2 kV                              |
| ESD CDM                                      |  | 500 V                             |
| Maintenance Interval                         | Maintenance free when ASC field-calibration algorithm <sup>12</sup> is used. | None                              |
| Sensor lifetime <sup>13</sup>                | Typical operating conditions   | > 10 years                        |

**Table 5:** SCD42 operation conditions, lifetime and maximum ratings

<sup>6</sup> Do not change supply voltage during operation

<sup>7</sup> Accuracy can be impaired by excess supply voltage variation caused by external load or temperature variations

<sup>8</sup> Power supply should be designed with respect to peak current.

<sup>9</sup> Accuracy can be reduced at relative humidity levels lower than 10 %.

<sup>10</sup> Exceeding MSL 1 storage conditions (30 °C, 85 % RH max.) prior to sensor assembly can lead to degraded sensor performance or sensor failure in the reflow soldering process.

<sup>11</sup> Short term storage refers to temporary conditions during e.g. transport.

<sup>12</sup> For proper function of ASC field-calibration algorithm the SCD42 has to be exposed to clean air with 400 ppm CO<sub>2</sub> concentration regularly.

<sup>13</sup> Sensor tested over simulated lifetime of > 10 years for indoor environment mission profile

### 2.3 Interface Specifications

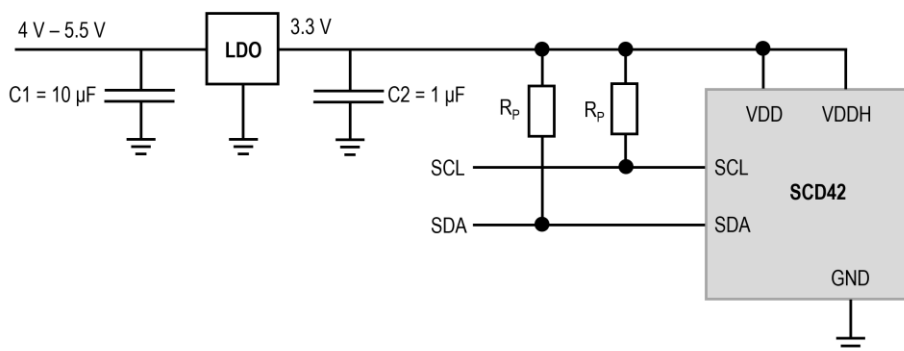
The SCD42 comes in an LGA package (**Table 6**). The package outline is schematically displayed in chapter 4.1. The landing pattern of the SCD42 can be found in chapter 4.2.

| Name | Comments  |
|------|---|
| VDD  | Supply voltage  |
| VDDH | Supply voltage IR source, must be connected to VDD on customer PCB          |
| GND  | Ground contact  |
| SDA  | I <sup>2</sup> C serial data, bidirectional                                 |
| SCL  | I <sup>2</sup> C serial clock   |
| DNC  | Do not connect, pads must be soldered to a floating pad on the customer PCB |

**Table 6** Pin assignment (top view). The notched corner of the protection membrane serves as a polarity mark to indicate pin 1 location.

VDD and VDDH are used to supply the sensor and must always be kept at the same voltage, i.e. should both be connected to the same power supply. The combined maximum current drawn on VDD and VDDH is indicated in **Table 4**. Care should be taken to choose a low noise power supply, which is adequately dimensioned for the relatively large peak currents. The power supply should be designed such that voltage variations caused by external influences remain below 5 mV. Consequently, no other components causing larger load variations should be supplied by the same voltage regulator. Sensirion recommends integrating a low-dropout regulator, LDO, such as the Toshiba TCR2EF33,LM(CT).

SCL is used to synchronize the I<sup>2</sup>C communication between the master (microcontroller) and the slave (sensor). The SDA pin is used to transfer data to and from the sensor. For safe communication, the timing specifications defined in the I<sup>2</sup>C manual<sup>14</sup> must be met. Both SCL and SDA lines should be connected to external pull-up resistors (e.g. R<sub>p</sub> = 10 kΩ, see **Figure 1**). To avoid signal contention, the microcontroller must only drive SDA and SCL low. For dimensioning resistor sizes please take bus capacity and communication frequency into account (see example in Section 7.1 of NXP's I<sup>2</sup>C Manual for more details<sup>14</sup>). It should be noted that pull-up resistors may be included in I/O circuits of microcontrollers.



**Figure 1:** Typical application circuit (for better clarity in the image, the positioning of the pins does not reflect the positions on the real sensor). VDD and VDDH must be connected to each other close to the sensor on the customer PCB.

<sup>14</sup> NXP's I<sup>2</sup>C-bus specification and user manual UM10204, Rev.6, 4 April 2014

## 2.4 Timing Specifications

**Table 7** list the timings of the ASIC part and does not reflect the availability or usefulness of the sensor readings. The SCD42 supports the I<sup>2</sup>C “standard-mode” as is described elsewhere (see footnote <sup>14</sup>).

| Parameter           | Condition                                     | Min. | Max. | Unit |
|---------------------|---|------|------|------|
| Power-up time       | After hard reset, $V_{DD} \geq 2.25\text{ V}$ | -    | 1000 | ms   |
| Soft reset time     | After re-initialization (i.e. reinit)         | -    | 1000 | ms   |
| SCL clock frequency | -   | 0    | 100  | kHz  |

**Table 7** System timing specifications.

## 2.5 Material Contents

The device is fully REACH and RoHS compliant.

### 3 Digital Interface Description

All SCD42 commands and data are mapped to a 16-bit address space.

| SCD42                    | Hex. Code |
|--------------------------|-----------|
| I <sup>2</sup> C address | 0x62      |

Table 8 I<sup>2</sup>C device address.

#### 3.1 Power-Up and Communication Start

The sensor starts powering-up after reaching the power-up threshold voltage  $V_{DD,Min} = 2.25\text{ V}$ . After reaching this threshold voltage, the sensor needs 1000 ms to enter the idle state. Once the idle state is entered it is ready to receive commands from the master.

Each transmission sequence begins with a START condition (S) and ends with a STOP condition (P) as described in the I<sup>2</sup>C-bus specification.

#### 3.2 Data type & length

Data sent to and received from the sensor consists of a sequence of 16-bit commands and/or 16-bit words (each to be interpreted as unsigned integer, most significant byte transmitted first). Each data word is immediately succeeded by an 8-bit CRC. In write direction it is mandatory to transmit the checksum. In read direction it is up to the master to decide if it wants to process the checksum (see chapter 3.9).

#### 3.3 Command Sequence Types

The SCD42 features four different I<sup>2</sup>C command sequence types: “read I<sup>2</sup>C sequences”, “write I<sup>2</sup>C sequences”, “send I<sup>2</sup>C command” and “send command and fetch result” sequences. Figure 2 illustrates how the I<sup>2</sup>C communication for the different sequence types is built-up.

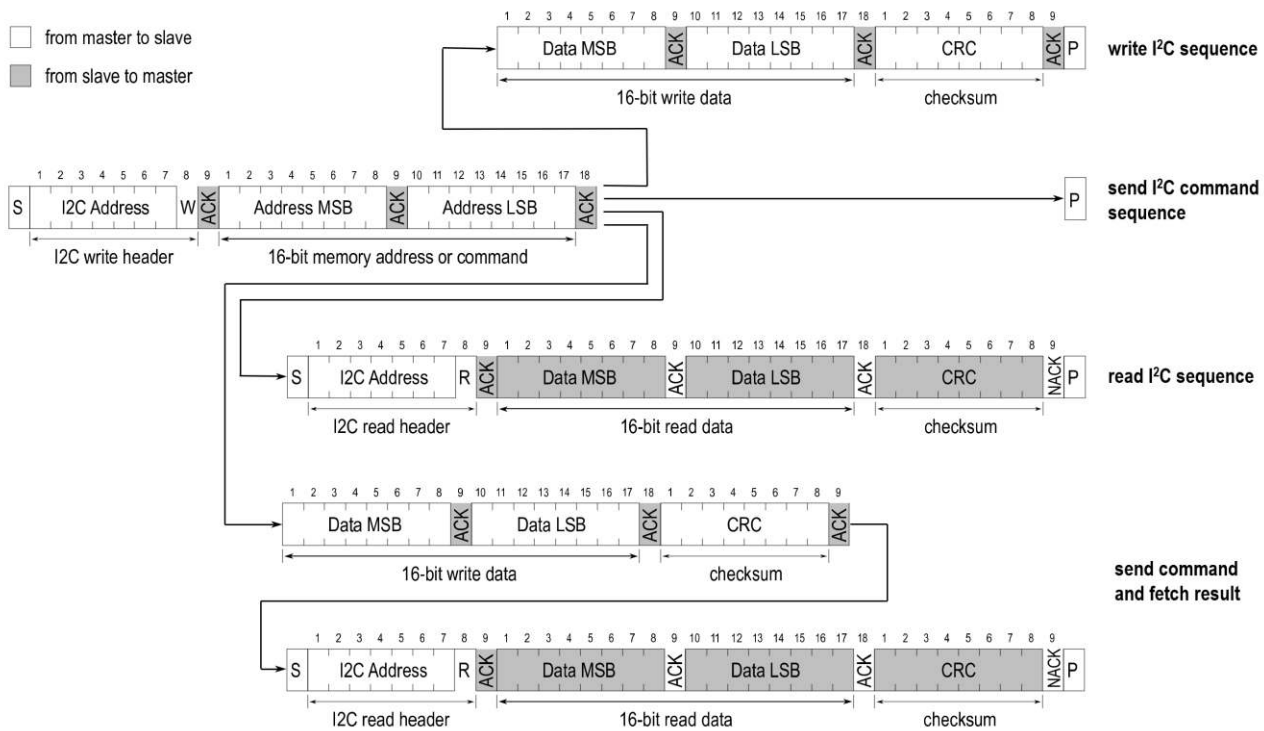


Figure 2: Command Sequence types: “write” sequence, “send command” sequence, “read” sequence, and “send command and fetch result” sequence.

For “read” or “send command and fetch results” sequences, after writing the address and/or data to the sensor and sending the ACK bit, the sensor needs the *execution time* (see Table 9) to respond to the I<sup>2</sup>C read header with an ACK bit. Hence, it is required to wait the command *execution time* before issuing the read header. Commands must not be sent while a previous command is being processed.

### 3.4 SCD42 Command Overview

**Table 9:** List of SCD42 sensor commands. Detailed description of SCD42 commands can be found further down. \*Column indicates whether command can be executed while a periodic measurement is running.

| Domain  | Command                                | Hex. Code | I <sup>2</sup> C sequence type (see chapter 3.3) | Execution |               |
|---|--|-----------|--|-----------|---------------|
|   |  |           |  | time [ms] | During meas.* |
| Basic Commands<br>Chapter 3.5                     | start_periodic_measurement             | 0x21b1    | send command                                     | -         | no            |
|   | read_measurement                       | 0xec05    | read   | 1         | yes           |
|   | stop_periodic_measurement              | 0x3f86    | send command                                     | 500       | yes           |
| On-chip output signal compensation<br>Chapter 3.6 | set_temperature_offset                 | 0x241d    | write  | 1         | no            |
|   | get_temperature_offset                 | 0x2318    | read   | 1         | no            |
|   | set_sensor_altitude                    | 0x2427    | write  | 1         | no            |
|   | get_sensor_altitude                    | 0x2322    | read   | 1         | no            |
|   | set_ambient_pressure                   | 0xe000    | write  | 1         | yes           |
| Field calibration<br>Chapter 3.7                  | perform_forced_recalibration           | 0x362f    | send command and fetch result                    | 400       | no            |
|   | set_automatic_self_calibration_enabled | 0x2416    | write  | 1         | no            |
|   | get_automatic_self_calibration_enabled | 0x2313    | read   | 1         | no            |
| Advanced features<br>Chapter 3.8                  | persist_settings                       | 0x3615    | send command                                     | 800       | no            |
|   | get_serial_number                      | 0x3682    | read   | 1         | no            |
|   | perform_self_test                      | 0x3639    | read   | 10000     | no            |
|   | perform_factory_reset                  | 0x3632    | send command                                     | 1200      | no            |
|   | reinit                                 | 0x3646    | send command                                     | 20        | no            |



### 3.5 Basic Commands

This section lists the basic SCD42 commands that are necessary to start a periodic measurement and subsequently read out the sensor outputs.

The typical communication sequence between the I<sup>2</sup>C master (e.g., a microcontroller) and the SCD42 sensor is as follows:

1. The sensor is powered up
2. The I<sup>2</sup>C master sends a *start\_periodic\_measurement* command. Signal update interval is 5 seconds.
3. The I<sup>2</sup>C master periodically reads out data with the *read\_measurement* sequence.
4. To put the sensor back to idle mode, the I<sup>2</sup>C master sends a *stop periodic measurement* command.

While a periodic measurement is running, no other commands must be issued with the exception of *read\_measurement*, *get\_data\_ready\_status*, *stop\_periodic\_measurement* and *set\_ambient\_pressure*.

#### 3.5.1 start\_periodic\_measurement

**Description:** start periodic measurement, signal update interval is 5 seconds.

**Table 10:** start\_periodic\_measurement I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                     | Input parameter: - |                   | Response parameter: - |                   | Max. command duration [ms] |
|--|--------------------|-------------------|-----------------------|-------------------|----------------------------|
|  | length [bytes]     | signal conversion | length [bytes]        | signal conversion |                            |
| 0x21b1                                     | -                  | -                 | -                     | -                 | not applicable             |
| <b>Example:</b> start periodic measurement |                    |                   |                       |                   |                            |
| Write<br>(hexadecimal)                     | 0x21b1<br>Command  |                   |                       |                   |                            |

#### 3.5.2 read\_measurement

**Description:** read sensor output. The measurement data can only be read out once per signal update interval as the buffer is emptied upon read-out. If no data is available in the buffer, the sensor returns a NACK. To avoid a NACK response, the *get\_data\_ready\_status* can be issued to check data status (see chapter **Error! Reference source not found.** for further details). The I<sup>2</sup>C master can abort the read transfer with a NACK followed by a STOP condition after any data byte if the user is not interested in subsequent data.

**Table 11:** read\_measurement I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                                       | Input parameter: -                  |                       |                         | Response parameter: CO <sub>2</sub> , Temperature, Relative Humidity |                     | Max. command duration [ms] |
|--|-------------------------------------|-----------------------|-------------------------|--|---------------------|----------------------------|
|  | length [bytes]                      | signal conversion     | length [bytes]          | signal conversion  |                     |                            |
| 0xec05   | -                                   | -                     | 3                       | CO <sub>2</sub> [ppm] = word[0]                                      | 1                   |                            |
|  |                                     |                       | 3                       | $T = -45 + 175 * \text{word}[1] / 2^{16}$                            |                     |                            |
|  |                                     |                       | 3                       | $RH = 100 * \text{word}[2] / 2^{16}$                                 |                     |                            |
| <b>Example:</b> read sensor output (500 ppm, 25 °C, 37 % RH) |                                     |                       |                         |  |                     |                            |
| Write<br>(hexadecimal)                                       | 0xec05<br>Command                   |                       |                         |  |                     |                            |
| Wait   | 1 ms <i>command execution time</i>  |                       |                         |  |                     |                            |
| Response<br>(hexadecimal)                                    | 0x01f4<br>CO <sub>2</sub> = 500 ppm | 0x7b<br>CRC of 0x01f4 | 0x6667<br>Temp. = 25 °C | 0xa2<br>CRC of 0x6667  | 0x5eb9<br>RH = 37 % | 0x3c<br>CRC of 0x5eb9      |

### 3.5.3 stop\_periodic\_measurement

**Description:** stop periodic measurement to change the sensor configuration or to save power. Note that the sensor will only respond to other commands after waiting 500 ms after issuing the *stop\_periodic\_measurement* command.

**Table 12:** stop\_periodic\_measurement I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                    | Input parameter: - |                   | Response parameter: - |                   | Max. command duration [ms] |
|---|--------------------|-------------------|-----------------------|-------------------|----------------------------|
|   | length [bytes]     | signal conversion | length [bytes]        | signal conversion |                            |
| 0x3f86                                    | -                  | -                 | -                     | -                 | 500                        |
| <b>Example:</b> stop periodic measurement |                    |                   |                       |                   |                            |
| Write<br>(hexadecimal)                    | 0x3f86<br>Command  |                   |                       |                   |                            |

### 3.6 On-Chip Output Signal Compensation

The SCD42 features on-chip signal compensation to counteract pressure and temperature effects. Feeding the SCD42 with the pressure or altitude enables highest accuracy of the CO<sub>2</sub> output signal across a large pressure range. Setting the temperature offset improves the accuracy of the relative humidity and temperature output signal. Note that the temperature offset does not impact the accuracy of the CO<sub>2</sub> output.

To change or read sensor settings, the SCD42 must be in idle mode. A typical sequence between the I<sup>2</sup>C master and the SCD42 is described as follows:

1. If the sensor is operated in a periodic measurement mode, the I<sup>2</sup>C master sends a *stop\_periodic\_measurement* command.
2. The I<sup>2</sup>C master sends one or several commands to get or set the sensor settings.
3. If configurations shall be preserved after power-cycle events, the *persist\_settings* command must be sent (see chapter 3.8.1)
4. The I<sup>2</sup>C master sends a start measurement command to set the sensor in the operating mode again.

#### 3.6.1 set\_temperature\_offset

**Description:** The temperature offset has no influence on the SCD42 CO<sub>2</sub> accuracy. Setting the temperature offset of the SCD42 inside the customer device correctly allows the user to leverage the RH and T output signal. Note that the temperature offset can depend on various factors such as the SCD42 measurement mode, self-heating of close components, the ambient temperature and air flow. Thus, the SCD42 temperature offset should be determined inside the customer device under its typical operation conditions (including the operation mode to be used in the application) and in thermal equilibrium. Per default, the temperature offset is set to 4° C. To save the setting to the EEPROM, the *persist setting* (see chapter 3.8.1) command must be issued. Equation (1) shows how the characteristic temperature offset can be obtained.

$$T_{offset\_actual} = T_{SCD40} - T_{Reference} + T_{offset\_previous} \tag{1}$$

**Table 13:** set\_temperature\_offset I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                           | Input parameter: Offset temperature |  | Response parameter: - |                   | Max. command duration [ms] |
|--|-------------------------------------|--|-----------------------|-------------------|----------------------------|
|  | length [bytes]                      | signal conversion  | length [bytes]        | signal conversion |                            |
| 0x241d   | 3                                   | word[0] = T <sub>offset</sub> [°C] * 2 <sup>16</sup> / 175 | -                     | -                 | 1                          |
| <b>Example:</b> set temperature offset to 5.4 °C |                                     |  |                       |                   |                            |
| Write<br>(hexadecimal)                           | 0x241d<br>Command                   | 0x07e6<br>T <sub>offset</sub> = 5.4 °C                     | 0x48<br>CRC of 0x7e6  |                   |                            |

### 3.6.2 get\_temperature\_offset

**Table 14:** get\_temperature\_offset I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                       | Input parameter: -   |                        | Response parameter: Offset temperature |  | Max. command duration [ms] |
|--|--|------------------------|--|--|----------------------------|
|  | length [bytes]   | signal conversion      | length [bytes]                         | signal conversion  |                            |
| 0x2318                                       | -  | -                      | 3                                      | $T_{\text{offset}} [^{\circ}\text{C}] = 175 * \text{word}[0] / 2^{16}$ | 1                          |
| <b>Example:</b> temperature offset is 6.2 °C |  |                        |  |  |                            |
| Write<br>(hexadecimal)                       | 0x2318<br>Command  |                        |  |  |                            |
| Wait   | 1 ms   | command execution time |  |  |                            |
| Response<br>(hexadecimal)                    | 0x0912<br>$T_{\text{offset}} = 6.2 \text{ }^{\circ}\text{C}$ | 0x63<br>CRC of 0x0912  |  |  |                            |

### 3.6.3 set\_sensor\_altitude

**Description:** Reading and writing of the sensor altitude must be done while the SCD42 is in idle mode. Typically, the sensor altitude is set once after device installation. To save the setting to the EEPROM, the *persist setting* (see chapter 3.8.1) command must be issued. Per default, the sensor altitude is set to 0 meter above sea-level.

**Table 15:** set\_sensor\_altitude I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                                | Input parameter: Sensor altitude |                                     | Response parameter: - |                   | Max. command duration [ms] |
|---|----------------------------------|-------------------------------------|-----------------------|-------------------|----------------------------|
|   | length [bytes]                   | signal conversion                   | length [bytes]        | signal conversion |                            |
| 0x2427  | 3                                | word[0] = Sensor altitude [m]       | -                     | -                 | 1                          |
| <b>Example:</b> set sensor altitude to 1'950 m.a.s.l. |                                  |                                     |                       |                   |                            |
| Write<br>(hexadecimal)                                | 0x2427<br>Command                | 0x079e<br>Sensor altitude = 1'950 m | 0x09<br>CRC of 0x79e  |                   |                            |

### 3.6.4 get\_sensor\_altitude

**Table 16:** get\_sensor\_altitude I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                            | Input parameter: -                  |                        | Response parameter: Sensor altitude |                               | Max. command duration [ms] |
|---|-------------------------------------|------------------------|-------------------------------------|-------------------------------|----------------------------|
|   | length [bytes]                      | signal conversion      | length [bytes]                      | signal conversion             |                            |
| 0x2322  | -                                   | -                      | 3                                   | Sensor altitude [m] = word[0] | 1                          |
| <b>Example:</b> sensor altitude is 1'100 m.a.s.l. |                                     |                        |                                     |                               |                            |
| Write<br>(hexadecimal)                            | 0x2322<br>Command                   |                        |                                     |                               |                            |
| Wait  | 1 ms                                | command execution time |                                     |                               |                            |
| Response<br>(hexadecimal)                         | 0x044c<br>Sensor altitude = 1'100 m | 0x42<br>CRC of 0x044c  |                                     |                               |                            |

### 3.6.5 set\_ambient\_pressure

**Description:** The *set\_ambient\_pressure* command can be sent during periodic measurements to enable continuous pressure compensation. Note that setting an ambient pressure using *set\_ambient\_pressure* overrides any pressure compensation based on a previously set sensor altitude.

**Table 17:** set\_ambient\_pressure I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                            | Input parameter: Ambient pressure |  | Response parameter: -        |                   | Max. command<br>duration [ms] |
|---|-----------------------------------|--|------------------------------|-------------------|-------------------------------|
|   | length [bytes]                    | signal conversion                      | length [bytes]               | signal conversion |                               |
| 0xe000  | 3                                 | word[0] = ambient P [Pa] /<br>100      | -                            | -                 | 1                             |
| <b>Example:</b> set ambient pressure to 98'700 Pa |                                   |  |                              |                   |                               |
| Write<br>(hexadecimal)                            | 0xe000<br><i>Command</i>          | 0x03db<br><i>Ambient P = 98'700 Pa</i> | 0x42<br><i>CRC of 0x03db</i> |                   |                               |

### 3.7 Field Calibration

To realize high initial and long-term accuracy, the SCD42 includes two field calibration features. Forced recalibration (FRC) enables restoring highest accuracy with the assistance of a CO<sub>2</sub> reference value immediately. Typically, FRC is applied to compensate for drifts originating from the sensor assembly process or other extensive stresses. Automatic self-calibration (ASC) ensures highest long-term stability of the SCD42 without the need of manual action steps from the user. The automatic self-calibration algorithm assumes that the sensor is exposed to the atmospheric CO<sub>2</sub> concentration of 400 ppm at least once per week.

#### 3.7.1 perform\_forced\_recalibration

**Description:** To successfully conduct an accurate forced recalibration, the following steps need to be carried out:

1. Operate the SCD42 in the operation mode later used in normal sensor operation for > 3 minutes in an environment with homogenous and constant CO<sub>2</sub> concentration.
2. Issue *stop\_periodic\_measurement*. Wait 500 ms for the stop command to complete.
3. Subsequently issue the *perform\_forced\_recalibration* command and optionally read out the FRC correction (i.e. the magnitude of the correction) after waiting for 400 ms for the command to complete.
  - A return value of 0xffff indicates that the forced recalibration has failed.

Note that the sensor will fail to perform a forced recalibration if it was not operated before sending the command. Please make sure that the sensor is operated at the voltage desired for the application when applying the forced recalibration sequence.

**Table 18:** perform\_forced\_recalibration I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)   | Input parameter: Target CO <sub>2</sub> concentration |   | Response parameter: FRC-correction |  | Max.<br>command<br>duration [ms] |
|--|---|---|------------------------------------|--|----------------------------------|
|  | length [bytes]  | signal conversion                                     | length [bytes]                     | signal conversion  |                                  |
| 0x362f   | 3   | word[0] = Target concentration [ppm CO <sub>2</sub> ] | 3                                  | FRC correction [ppm CO <sub>2</sub> ] = word[0] – 0x8000<br>word[0] = 0xffff in case of failed FRC | 400                              |
| <b>Example:</b> perform forced recalibration, reference CO <sub>2</sub> concentration is 480 ppm |   |   |                                    |  |                                  |
| Write<br>(hexadecimal)   | 0x362f<br><i>Command</i>                              | 0x01e0<br><i>Input: 480 ppm</i>                       | 0xb4<br><i>CRC of 0x01e0</i>       |  |                                  |
| Wait   | 400 ms  | <i>command execution time</i>                         |                                    |  |                                  |
| Response<br>(hexadecimal)  | 0x7fce<br><i>Response: - 50 ppm</i>                   | 0x7b<br><i>CRC of 0x7fce</i>                          |                                    |  |                                  |

### 3.7.2 set\_automatic\_self\_calibration\_enabled

**Description:** Set the current state (enabled / disabled) of the automatic self-calibration. By default, ASC is enabled. To save the setting to the EEPROM, the *persist\_setting* (see chapter 3.8.1) command must be issued.

**Table 19:** set\_automatic\_self\_calibration\_enabled I<sup>2</sup>C sequence description.

| Write<br>(hexadecimal)   | Input parameter: ASC enabled |   | Response parameter: - |                   | Max. command duration [ms] |
|--|------------------------------|---|-----------------------|-------------------|----------------------------|
|  | length [bytes]               | signal conversion                                       | length [bytes]        | signal conversion |                            |
| 0x2416   | 3                            | word[0] = 1 → ASC enabled<br>word[0] = 0 → ASC disabled | -                     | -                 | 1                          |
| <b>Example:</b> set automatic self-calibration status: enabled |                              |   |                       |                   |                            |
| Write<br>(hexadecimal)   | 0x2416<br>Command            | 0x0001<br>ASC enabled                                   | 0xB0<br>CRC of 0x0001 |                   |                            |

### 3.7.3 get\_automatic\_self\_calibration\_enabled

**Table 20:** get\_automatic\_self\_calibration\_enabled I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)   | Input parameter: -     |                               | Response parameter: ASC enabled |   | Max. command duration [ms] |
|--|------------------------|-------------------------------|---------------------------------|---|----------------------------|
|  | length [bytes]         | signal conversion             | length [bytes]                  | signal conversion                                       |                            |
| 0x2313   | -                      | -                             | 3                               | word[0] = 1 → ASC enabled<br>word[0] = 0 → ASC disabled | 1                          |
| <b>Example:</b> read automatic self-calibration status: disabled |                        |                               |                                 |   |                            |
| Write<br>(hexadecimal)   | 0x2313<br>Command      |                               |                                 |   |                            |
| Wait   | 1 ms                   | <i>command execution time</i> |                                 |   |                            |
| Response<br>(hexadecimal)  | 0x0000<br>ASC disabled | 0x81<br>CRC of 0x0000         |                                 |   |                            |

### 3.8 Advanced Features

#### 3.8.1 persist\_settings

**Description:** Configuration settings such as the temperature offset, sensor altitude and the ASC enabled/disabled parameter are by default stored in the volatile memory (RAM) only and will be lost after a power-cycle. The *persist\_settings* command stores the current configuration in the EEPROM of the SCD42, making them persistent across power-cycling. To avoid unnecessary wear of the EEPROM, the *persist\_settings* command should only be sent when persistence is required and if actual changes to the configuration have been made. The EEPROM is guaranteed to endure at least 2000 write cycles before failure. Note that field calibration history (i.e. FRC and ASC, see chapter 3.7) is automatically stored in a separate EEPROM dimensioned for the specified sensor lifetime.

**Table 21:** persist\_settings I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)           | Input parameter: - |                   | Response parameter: - |                   | Max. command duration [ms] |
|----------------------------------|--------------------|-------------------|-----------------------|-------------------|----------------------------|
|                                  | length [bytes]     | signal conversion | length [bytes]        | signal conversion |                            |
| 0x3615                           | -                  | -                 | -                     | -                 | 800                        |
| <b>Example:</b> persist settings |                    |                   |                       |                   |                            |
| Write<br>(hexadecimal)           | 0x3615<br>Command  |                   |                       |                   |                            |

#### 3.8.2 get\_serial\_number

**Description:** Reading out the serial number can be used to identify the chip and to verify the presence of the sensor. The get serial number command returns 3 words, and every word is followed by an 8-bit CRC checksum. Together, the 3 words constitute a unique serial number with a length of 48 bits (big endian format).

**Table 22:** get\_serial\_number I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                               | Input parameter: -                 |                              |                          | Response parameter: serial number                       |                          | Max. command duration [ms]   |
|--|------------------------------------|------------------------------|--------------------------|---|--------------------------|------------------------------|
|  | length [bytes]                     | signal conversion            | length [bytes]           | signal conversion                                       |                          |                              |
| 0x3682   | -                                  | -                            | 9                        | Serial number = word[0] << 32   word[1] << 16   word[2] | 1                        |                              |
| <b>Example:</b> serial number is 273'325'796'834'238 |                                    |                              |                          |   |                          |                              |
| Write<br>(hexadecimal)                               | 0x3682<br>Command                  |                              |                          |   |                          |                              |
| Wait   | 1 ms <i>command execution time</i> |                              |                          |   |                          |                              |
| Response<br>(hexadecimal)                            | 0xf896<br><i>word[0]</i>           | 0x31<br><i>CRC of 0xf896</i> | 0x9f07<br><i>word[1]</i> | 0xc2<br><i>CRC of 0x9f07</i>                            | 0x3bbe<br><i>word[2]</i> | 0x89<br><i>CRC of 0x3bbe</i> |

### 3.8.3 perform\_self\_test

**Description:** The perform\_self\_test feature can be used as an end-of-line test to check sensor functionality and the customer power supply to the sensor.

**Table 23:** perform\_self\_test I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                                     | Input parameter: - |                        | Response parameter: sensor status        |   | Max. command duration [ms] |
|--|--------------------|------------------------|--|---|----------------------------|
|  | length [bytes]     | signal conversion      | length [bytes]                           | signal conversion   |                            |
| 0x3639   | -                  | -                      | 3  | word[0] = 0 → no malfunction detected<br>word[0] ≠ 0 → malfunction detected | 10000                      |
| <b>Example:</b> perform self-test, no malfunction detected |                    |                        |  |   |                            |
| Write<br>(hexadecimal)                                     | 0x3639<br>Command  |                        |  |   |                            |
| Wait   | 10000 ms           | command execution time |  |   |                            |
| Response<br>(hexadecimal)                                  | 0x0000             | 0x81                   | No malfunction detected<br>CRC of 0x0000 |   |                            |

### 3.8.4 perform\_factory\_reset

**Description:** The perform\_factory\_reset command resets all configuration settings stored in the EEPROM and erases the FRC and ASC algorithm history.

**Table 24:** perform\_factory\_reset I<sup>2</sup>C sequence description

| Write<br>(hexadecimal)                | Input parameter: - |                   | Response parameter: - |                   | Max. command duration [ms] |
|---------------------------------------|--------------------|-------------------|-----------------------|-------------------|----------------------------|
|                                       | length [bytes]     | signal conversion | length [bytes]        | signal conversion |                            |
| 0x3632                                | -                  | -                 | -                     | -                 | 1200                       |
| <b>Example:</b> perform factory reset |                    |                   |                       |                   |                            |
| Write<br>(hexadecimal)                | 0x3632<br>Command  |                   |                       |                   |                            |

### 3.8.5 reinit

**Description:** The reinit command reinitializes the sensor by reloading user settings from EEPROM. Before sending the reinit command, the stop measurement command must be issued. If the reinit command does not trigger the desired re-initialization, a power-cycle should be applied to the SCD42.

**Table 25:** reinit I<sup>2</sup>C sequence description

| Write<br>(hexadecimal) | Input parameter: -                  |                   | Response parameter: - |                   | Max. command duration [ms] |
|------------------------|-------------------------------------|-------------------|-----------------------|-------------------|----------------------------|
|                        | length [bytes]                      | signal conversion | length [bytes]        | signal conversion |                            |
| 0x3646                 | -                                   | -                 | -                     | -                 | 20                         |
| <b>Example:</b> reinit |                                     |                   |                       |                   |                            |
| Write<br>(hexadecimal) | 0x3646<br>Command re-initialization |                   |                       |                   |                            |



### 3.9 Checksum Calculation

The 8-bit CRC checksum transmitted after each data word is generated by a CRC algorithm. Its properties are displayed in **Table 26**. The CRC covers the contents of the two previously transmitted data bytes. To calculate the checksum only these two previously transmitted data bytes are used. Note that command words are not followed by CRC.

| Property       | Value                          | Example code (C/C++)   |
|----------------|--------------------------------|--|
| Name           | CRC-8                          | <pre> #define CRC8_POLYNOMIAL 0x31 #define CRC8_INIT 0xFF  uint8_t sensirion_common_generate_crc(const uint8_t* data, uint16_t count) {     uint16_t current_byte;     uint8_t crc = CRC8_INIT;     uint8_t crc_bit;     /* calculates 8-Bit checksum with given polynomial */     for (current_byte = 0; current_byte &lt; count; ++current_byte) {         crc ^= (data[current_byte]);         for (crc_bit = 8; crc_bit &gt; 0; --crc_bit) {             if (crc &amp; 0x80)                 crc = (crc &lt;&lt; 1) ^ CRC8_POLYNOMIAL;             else                 crc = (crc &lt;&lt; 1);         }     }     return crc; } </pre> |
| Width          | 8 bit                          |  |
| Protected Data | read and/or write data         |  |
| Polynomial     | 0x31 ( $x^8 + x^5 + x^4 + 1$ ) |  |
| Initialization | 0xFF                           |  |
| Reflect input  | False                          |  |
| Reflect output | False                          |  |
| Final XOR      | 0x00                           |  |
| Examples       | CRC (0xBEEF) = 0x92            |  |

**Table 26** I<sup>2</sup>C CRC properties.

## 4 Mechanical specifications

### 4.1 Package Outline

Figure 3 schematically displays the package outline. The notched corner of the protection membrane serves as a polarity mark to indicate pin 1 location. Nominal dimensions and tolerances are listed in Table 27.

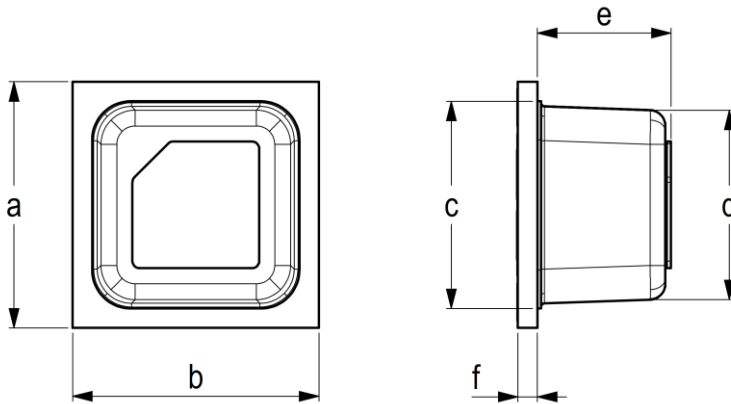


Figure 3: Packaging outline drawing of the SCD42: (left) top view and (right) side view. Nominal dimensions and tolerances are listed in millimeters.

| Dimension      | a     | b     | c     | d     | e     | f     |
|----------------|-------|-------|-------|-------|-------|-------|
| Nominal [mm]   | 10.1  | 10.1  | 8.5   | 7.8   | 5.5   | 0.8   |
| Tolerance [mm] | ± 0.3 | ± 0.3 | ± 0.2 | ± 0.2 | ± 0.3 | ± 0.2 |

Table 27: Nominal dimensions and tolerances SCD42 (all in mm). The weight of the sensor is approx. 0.6 g.

Note that the white protection membrane on top of the sensor must not be removed or tampered with to ensure proper sensor operation.

### 4.2 Land Pattern

Recommended land pattern, solder paste and solder mask are shown in Figure 4. These are recommendations only and not specifications. The exact mask geometries, distances and stencil thicknesses must be adapted to the customer soldering processes.

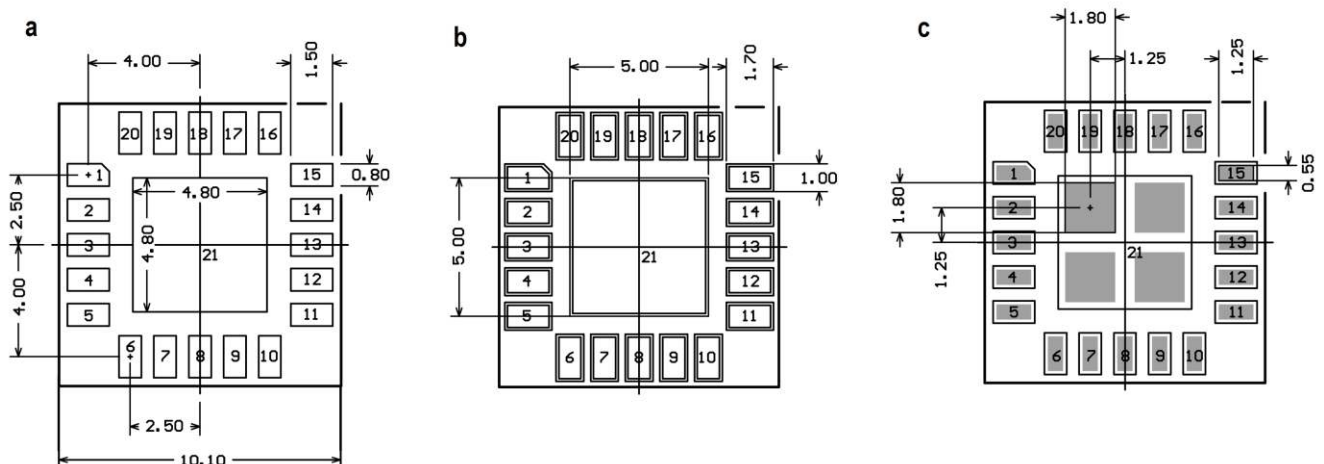
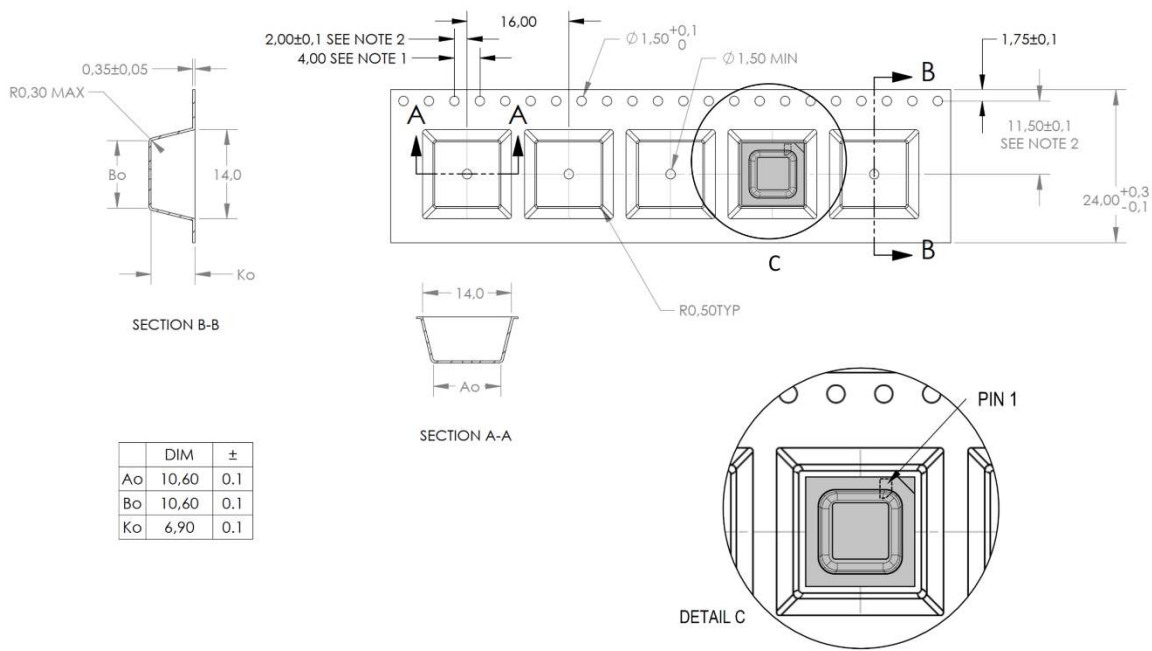


Figure 4: SCD42 footprint (top view): landing pads (a), solder mask (b) and solder paste (c).

### 4.3 Tape & Reel Package



**Figure 5:** Technical drawing of the packaging tape with sensor orientation in tape. Header tape is to the right and trailer tape to the left on this drawing. Dimensions are given in millimeters.

### 4.4 Moisture Sensitivity Level

Sensirion SCD42 sensors shall be treated according to Moisture Sensitivity Level 1 (MSL1) as described in IPC/JEDEC J-STD-033B1. Exposure to moisture levels or temperatures before sensor assembly, which exceed the limits as stated in this document, can result in yield loss and sensor performance degradation.

### 4.5 Soldering Instructions

For soldering, standard reflow soldering ovens may be used. The sensors are designed to withstand soldering profile based on IPC/JEDEC J-STD-020 with a maximum peak temperature of 235°C during up to 30 sec for Pb-free assembly in IR/Convection reflow ovens.

Note that due to the comparably large size of the SCD42 sensor significant temperature differences across the sensor element can occur during reflow soldering. Specifically, the temperature within the sensor cap can be higher than the temperature measured at the pad using usual temperature monitoring methods. Care must be taken that a temperature of 235° C is not exceeded at any time in any part of the sensor.

Do not apply any board wash process step subsequently to the reflow soldering. Note that the dust cover on top of the cap must not be removed or wetted with any liquid. Finally, the SCD42 is not compatible with vapor phase reflow soldering.

Minor temporary accuracy deviations of the CO<sub>2</sub> reading can result from the reflow soldering of the SCD42. Full sensor accuracy is restored after at most five days after the soldering process, independently on whether the sensor is operated or not.

|                      |  |  |
|----------------------|--|--|
| Average ramp-up rate | < 3 °C / second  |  |
| Liquid phase         | <ul style="list-style-type: none"> <li>▪ T<sub>L</sub> &lt; 220 °C</li> <li>▪ t<sub>L</sub> &lt; 60 seconds</li> </ul> |  |
| Peak temperature     | <ul style="list-style-type: none"> <li>▪ T<sub>P</sub> ≤ 235 °C</li> <li>▪ t<sub>P</sub> &lt; 30 seconds</li> </ul>    |  |
| Ramp-down rate       | < 4 °C / seconds for temperature > T <sub>L</sub>  |  |

**Table 28** Soldering profile parameter

### 4.6 Traceability

All SCD42 sensors have a distinct electronic serial number for identification and traceability (see chapter 3.8.2). The serial number can be decoded by Sensirion only and allows for tracking through production, calibration, and testing.

## 5 Ordering Information

Use the part names and product numbers shown in the following table when ordering the SCD42 CO<sub>2</sub> sensor. For the latest product information and local distributors, visit <http://www.sensirion.com/>.

| Part Name        | Description  | Ordering quantity (pcs) | Product Number |
|------------------|--|-------------------------|----------------|
| SCD42-D-R2       | SCD42 CO2 sensor SMD component as reel, I2C                  | 600 sensors per reel    | 3.000.634      |
| SEK-SCD41-Sensor | SEK-SCD41-Sensor set; SCD41 on development board with cables | 1                       | 3.000.455      |
| SEK-SensorBridge | Sensor Bridge to connect SEK-SCD41-Sensor to computer        | 1                       | 3.000.124      |

**Table 29** SCD42 ordering options

## 6 Revision History

| Date           | Version | Page(s) | Changes  |
|----------------|---------|---------|--|
| September 2021 | 0.9     | all     | Preliminary version  |
| September 2021 | 0.91    | 1, 20   | Minor revisions (page 1), addition of product number (page 20)   |
| October 2021   | 0.92    | 4       | Moisture sensitivity level (MSL) changed to 1  |
| March 2022     | 1       | 3, 20   | Clarification on additional sensor accuracy drift (Table 1)<br>Addition of temporary accuracy deviation after reflow soldering process (4.5) |

## Important Notices

### Warning, Personal Injury

**Do not use this product as safety or emergency stop devices or in any other application where failure of the product could result in personal injury. Do not use this product for applications other than its intended and authorized use. Before installing, handling, using or servicing this product, please consult the data sheet and application notes. Failure to comply with these instructions could result in death or serious injury.**

If the Buyer shall purchase or use SENSIRION products for any unintended or unauthorized application, Buyer shall defend, indemnify and hold harmless SENSIRION and its officers, employees, subsidiaries, affiliates and distributors against all claims, costs, damages and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if SENSIRION shall be allegedly negligent with respect to the design or the manufacture of the product.

### ESD Precautions

The inherent design of this component causes it to be sensitive to electrostatic discharge (ESD). To prevent ESD-induced damage and/or degradation, take customary and statutory ESD precautions when handling this product.

### Warranty

SENSIRION warrants solely to the original purchaser of this product for a period of 12 months (one year) from the date of delivery that this product shall be of the quality, material and workmanship defined in SENSIRION's published specifications of the product. Within such period, if proven to be defective, SENSIRION shall repair and/or replace this product, in SENSIRION's discretion, free of charge to the Buyer, provided that:

- notice in writing describing the defects shall be given to SENSIRION within fourteen (14) days after their appearance;
- such defects shall be found, to SENSIRION's reasonable satisfaction, to have arisen from SENSIRION's faulty design, material, or workmanship;
- the defective product shall be returned to SENSIRION's factory at the Buyer's expense; and
- the warranty period for any repaired or replaced product shall be limited to the unexpired portion of the original period.

This warranty does not apply to any equipment which has not been installed and used within the specifications recommended by SENSIRION for the intended and proper use of the equipment. EXCEPT FOR THE WARRANTIES EXPRESSLY SET FORTH HEREIN, SENSIRION MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THE PRODUCT. ANY AND ALL WARRANTIES, INCLUDING WITHOUT LIMITATION, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, ARE EXPRESSLY EXCLUDED AND DECLINED.

SENSIRION is only liable for defects of this product arising under the conditions of operation provided for in the data sheet and proper use of the goods. SENSIRION explicitly disclaims all warranties, express or implied, for any period during which the goods are operated or stored not in accordance with the technical specifications.

SENSIRION does not assume any liability arising out of any application or use of any product or circuit and specifically disclaims any and all liability, including without limitation consequential or incidental damages. All operating parameters, including without limitation recommended parameters, must be validated for each customer's applications by customer's technical experts. Recommended parameters can and do vary in different applications.

SENSIRION reserves the right, without further notice, (i) to change the product specifications and/or the information in this document and (ii) to improve reliability, functions and design of this product.

Copyright © 2021, by SENSIRION. CMOSens® is a trademark of Sensirion. All rights reserved

## Headquarters and Subsidiaries

### Sensirion AG

Laubisruetistr. 50  
CH-8712 Staefa ZH  
Switzerland

phone: +41 44 306 40 00  
fax: +41 44 306 40 30  
[info@sensirion.com](mailto:info@sensirion.com)  
[www.sensirion.com](http://www.sensirion.com)

### Sensirion Taiwan Co. Ltd

phone: +886 3 5506701  
[info@sensirion.com](mailto:info@sensirion.com)  
[www.sensirion.com](http://www.sensirion.com)

### Sensirion Inc., USA

phone: +1 312 690 5858  
[info-us@sensirion.com](mailto:info-us@sensirion.com)  
[www.sensirion.com](http://www.sensirion.com)

### Sensirion Japan Co. Ltd.

phone: +81 3 3444 4940  
[info-jp@sensirion.com](mailto:info-jp@sensirion.com)  
[www.sensirion.com/jp](http://www.sensirion.com/jp)

### Sensirion Korea Co. Ltd.

phone: +82 31 337 7700~3  
[info-kr@sensirion.com](mailto:info-kr@sensirion.com)  
[www.sensirion.com/kr](http://www.sensirion.com/kr)

### Sensirion China Co. Ltd.

phone: +86 755 8252 1501  
[info-cn@sensirion.com](mailto:info-cn@sensirion.com)  
[www.sensirion.com/cn](http://www.sensirion.com/cn)

To find your local representative, please visit  
[www.sensirion.com/distributors](http://www.sensirion.com/distributors)