

SCD4x



Features

-
-
-
-
-

Product Variants

-
-

Product Summary

Product Overview

Functional Block Diagram

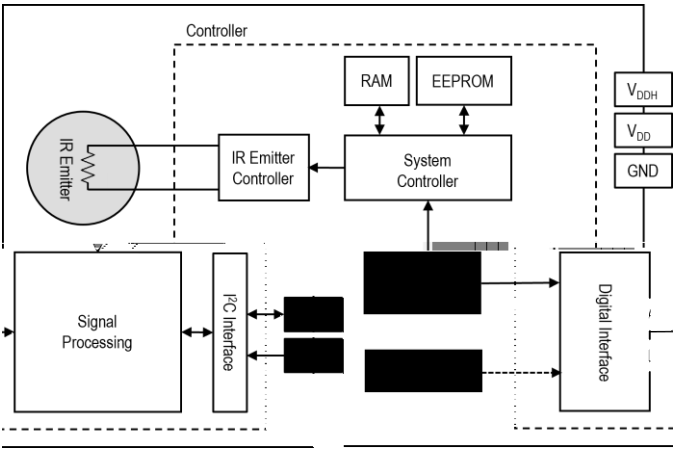


Table of Contents

- 1 Sensor Performance**

 - 2 Specifications**

 - 3 Digital Interface Description**

 - 4 Mechanical Specifications**

 - 5 Ordering Information**
 - 6 Revision History**
-

1 **Sensor Performance**
1.1 **CO₂ Sensing Performance**

Table 1

1.2 **Humidity Sensing Performance**

Table 2

Table 2

1.3 **Temperature Sensing Performance**

Table 3

Table 3

2 Specifications

2.1 Electrical Specifications

Table 4

2.2 Absolute Maximum Ratings

Table 5

Table 5

2.4 Timing Specifications

Table 7

| Parameter | Condition | Min. | Max. | Unit |
|-----------|-----------|------|------|------|
| | | | | |
| | | | | |
| | | | | |

Table 7

2.5 Material Contents

3 Digital Interface Description

3.1 Power-Up and Communication Start

3.2 Sensor I²C Address

Table 8

| SCD4x | Hex. Code |
|-------|-----------|
| | |

Table 8

3.3 Data Type & Length

3.4 Command Sequence Types

command"

"send command and fetch result"

read I²C sequences"

write I²C sequences"

send I²C

Figure 2

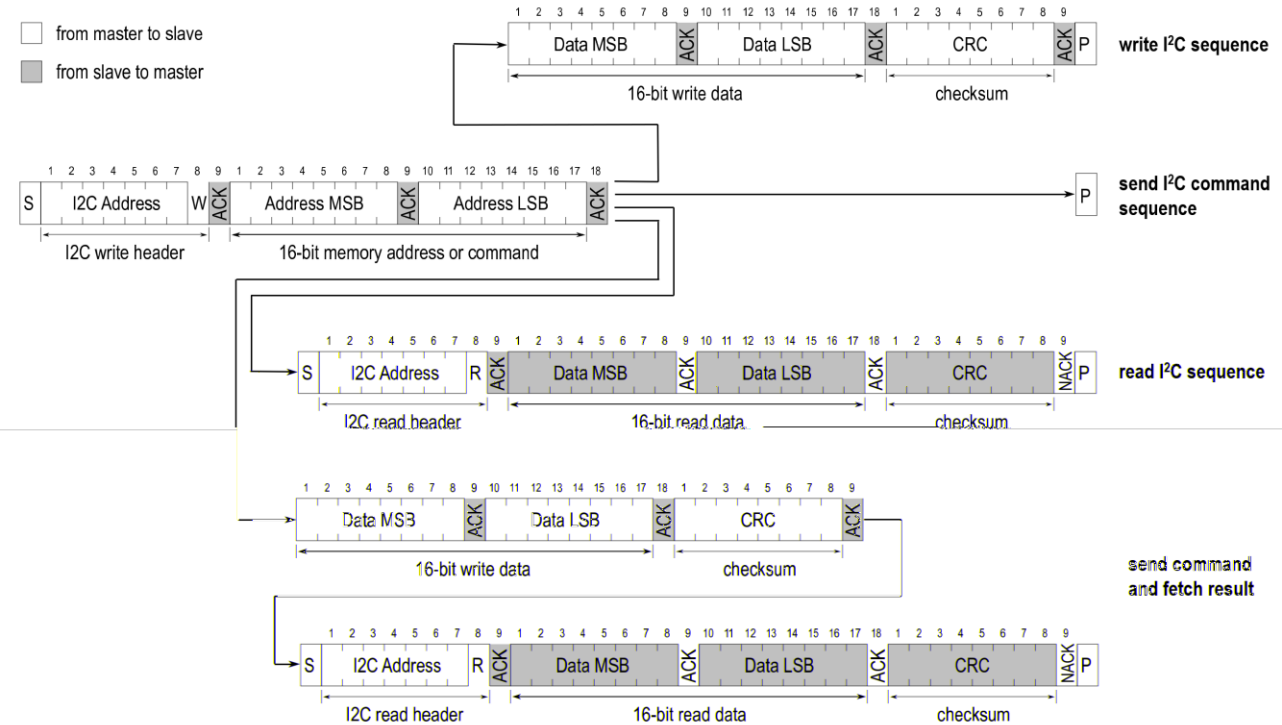


Figure 2: "write" "send" "read" "send command and fetch result"

"read" "send command and fetch results"

execution time

Table 9

Table 9

execution time

3.5 SCD4x Command Overview

Table 9.

| Domain | Command | Hex. Code | I ² C sequence type (see Section 3.4) | Execution | |
|--------|---------|--------------|---|--------------|-----------------|
| | | | | time [ms] | During meas. |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

Table 9

3.6 Basic Commands

- 1.
2. *start_periodic_measurement*
3. *read_measurement*
4. *stop_periodic_measurement*

get_data_ready_status, stop_periodic_measurement, set_ambient_pressure *read_measurement*
get_ambient_pressure.

3.6.1 *start_periodic_measurement*

Description

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> | | | | | |

Table 10

3.6.2 *read_measurement*

Description

get_data_ready_status

| Write | Input parameter: | | Response parameter: | | |
|------------------------|---------------------------|---------------|---------------------|--|---------------------------|
| | | | | | |
| | | | | $CO_2 \text{ [ppm]} = word[0]$ | |
| | | | | $T = -45 + 175 * \frac{word[1]}{2^{16} - 1}$ | |
| | | | | $RH = 100 * \frac{word[2]}{2^{16} - 1}$ | |
| Example: | | | | | |
| (hexadecimal) | Command | | | | |
| command execution time | | | | | |
| (hexadecimal) | CO ₂ = 500 ppm | CRC of 0x01f4 | Temp. = 25 °C | CRC of 0x6667 | RH = 37% CRC of 0x5eb9 |

Table 11

3.6.3 stop_periodic_measurement

Description

stop_periodic_measurement

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| <div>Example:</div> <div>(hexadecimal) Command</div> | | | | | |

Table 12

3.7 On-Chip Output Signal Compensation

stop_periodic_measurement

persist_settings

start_periodic_measurement

3.7.1 set_temperature_offset

Description

persist_settings

$T_{offset_previous}$ T_{SCD4x} $T_{Reference}$

get_temperature_offset

$T_{offset_actual} = T_{SCD4x} - T_{Reference} + T_{offset_previous}$

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | $word[0] = T_{offset}[^{\circ}C] * \frac{2^{16}-1}{175}$ | | | |
| <div>Example:</div> <div>(hexadecimal) Command $T_{offset} = 5.4\ ^{\circ}C$ CRC of 0x7e6</div> | | | | | |

Table 13

3.7.2 get_temperature_offset

| Write | Input parameter: | | Response parameter: | | |
|------------------------|-------------------------------------|---------------|---------------------|--|--|
| | | | | | |
| | | | | $T_{offset}[^{\circ}C] = word[0] * \frac{175}{2^{16}-1}$ | |
| Example: | | | | | |
| (hexadecimal) | Command | | | | |
| command execution time | | | | | |
| (hexadecimal) | $T_{offset} = 6.2\text{ }^{\circ}C$ | CRC of 0x0912 | | | |

Table 14

3.7.3 set_sensor_altitude

Description

persist_settings

| Write | Input parameter: | | Response parameter: | | |
|-----------------|------------------|---------------------------|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: | | | | | |
| (hexadecimal) | Command | Sensor altitude = 1'950 m | CRC of 0x079e | | |

Table 15

3.7.4 get_sensor_altitude

Description

get_sensor_altitude

set_sensor_altitude

| Write | Input parameter: | | Response parameter: | | |
|------------------------|--|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: | | | | | |
| (hexadecimal) | Command | | | | |
| command execution time | | | | | |
| (hexadecimal) | Sensor altitude = 1'100 m CRC of 0x044c | | | | |

Table 16

3.7.5 set_ambient_pressure

Description *set_ambient_pressure*

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> <i>Ambient P = 98'700 Pa</i> <i>CRC of 0x03db</i> | | | | | |

Table 17

3.7.6 get_ambient_pressure

Description *get_ambient_pressure*
 set_ambient_pressure

| Write | Input parameter: | | Response parameter: A | | |
|---|------------------|--|-----------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> <i>command execution time</i> <i>(hexadecimal)</i> <i>Ambient P = 98'700 Pa</i> <i>CRC of 0x03db</i> | | | | | |

Table 18

3.8 Field Calibration

3.8.1 perform_forced_rec calibration

Description

stop_periodic_measurement
perform_forced_rec calibration

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| <div>Example:</div> <div><div>(hexadecimal)</div><div>Command</div><div>Input: 480 ppm</div><div>CRC of 0x01e0</div></div> <div>command execution time</div> | | | | | |

3.8.2 set_automatic_self_calibration_enabled

Description

persist_settings

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> <i>ASC enabled</i> <i>CRC of 0x0001</i> | | | | | |

Table 20

3.8.3 get_automatic_self_calibration_enabled

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> <i>command execution time</i> <i>(hexadecimal)</i> <i>ASC disabled</i> <i>CRC of 0x0000</i> | | | | | |

Table 21

3.8.4 set_automatic_self_calibration_target

Description

persist_settings

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> <i>ASC target = 435 ppm</i> <i>CRC of 0x01b3</i> | | | | | |

Table 22

3.8.5 get_automatic_self_calibration_target

Description

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal) Command</i> <i>command execution time</i> <i>(hexadecimal) ASC target is 420 ppm CRC of 0x01a4</i> | | | | | |

Table 23

3.9 Low Power Periodic Measurement Mode

start_low_power_periodic_measurement
read_measurement

get_data_ready_status

read_measurement_command

3.9.1 start_low_power_periodic_measurement

Description

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal) Command</i> | | | | | |

Table 24

3.9.2 **get_data_ready_status**

Description

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: | | | | | |
| <i>(hexadecimal)</i> <i>Command</i> | | | | | |
| <i>command execution time</i> | | | | | |
| <i>(hexadecimal)</i> <i>Least significant 11 bits are 0 → data not ready</i> <i>CRC of 0x8000</i> | | | | | |

Table 25

3.10 **Advanced Features**

3.10.1 **persist_settings**

Description

persist_settings

persist_settings

| Write | Input parameter: | | Response parameter: | | |
|-------------------------------------|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: | | | | | |
| <i>(hexadecimal)</i> <i>Command</i> | | | | | |

Table 26

3.10.2 get_serial_number

Description

get_serial_number

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|----------------------|---------------------|----------------------|----------------------|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> <i>command execution time</i> | | | | | |
| <i>(hexadecimal)</i> | <i>word[0]</i> | <i>CRC of 0xf896</i> | <i>word[1]</i> | <i>CRC of 0x9f07</i> | <i>word[2]</i> |
| | | | | | <i>CRC of 0x3bbe</i> |

Table 27

3.10.3 perform_self_test

Description

perform_self_test

| Write | Input parameter: | | Response parameter: | | |
|---|--------------------------------|--|----------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> <i>command execution time</i> | | | | | |
| <i>(hexadecimal)</i> | <i>No malfunction detected</i> | | <i>CRC of 0x0000</i> | | |

Table 28

3.10.4 perform_factory_reset

Description

perform_factory_reset

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <i>(hexadecimal)</i> <i>Command</i> | | | | | |

Table 29

3.10.5 **reinit**

Description *reinit*
 reinit *reinit*

| Write | Input parameter: | | Response parameter: | | |
|-------------------------------------|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: | | | | | |
| <i>(hexadecimal)</i> <i>Command</i> | | | | | |

Table 30

3.10.6 **get_sensor_variant**

Description

| Write | Input parameter: | | Response parameter: | | |
|-------------------------------------|--------------------------|--------------|---------------------|---------------|--|
| | | | | | |
| | | | | | |
| <i>(hexadecimal)</i> <i>Command</i> | | | | | |
| <i>command execution time</i> | | | | | |
| Example: | | | | | |
| — | | | | | |
| <i>(hexadecimal)</i> | <i>Product version =</i> | <i>SCD41</i> | <i>CRC of</i> | <i>0x1440</i> | |
| Example: | | | | | |
| — | | | | | |
| <i>(hexadecimal)</i> | <i>Product version =</i> | <i>SCD40</i> | <i>CRC of</i> | <i>0x0440</i> | |

Table 31

3.11 Single Shot Measurement Mode (SCD41 Only)

wake_up

measure_single_shot

read_measurement

power_down

max. command duration

max. command duration

power_down

perform_factory_reset

power_down/wake_up

3.11.1 measure_single_shot

Description

read_measurement

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| <div>Example:</div> <div>(hexadecimal) Command</div> | | | | | |

Table 32

3.11.2 **measure_single_shot_rht_only**

Description

read_measurement

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: (hexadecimal) Command | | | | | |

Table 33

3.11.3 **power_down**

Description

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: (hexadecimal) Command | | | | | |

Table 34

3.11.4 **wake_up**

Description

wake_up

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: (hexadecimal) Command | | | | | |

Table 35:

3.11.5 set_automatic_self_calibration_initial_period

Description

persist_settings

| Write | Input parameter: | | Response parameter: | | |
|--|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: | | | | | |
| <div>(hexadecimal) Command Initial period CRC of 0x004c</div> <div>76 hours</div> | | | | | |

Table 36

3.11.6 get_automatic_self_calibration_initial_period

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: | | | | | |
| <div>(hexadecimal) Command</div> <div>command execution time</div> | | | | | |
| <div>(hexadecimal) 76 hours CRC of 0x004c</div> | | | | | |

Table 37

3.11.7 set_automatic_self_calibration_standard_period

Description

persist_settings

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <div> <div>(hexadecimal)</div> <div>Command</div> <div>Standard period 156 hours</div> <div>CRC of 0x009c</div> </div> | | | | | |

Table 38

3.11.8 get_automatic_self_calibration_standard_period

| Write | Input parameter: | | Response parameter: | | |
|---|------------------|--|---------------------|--|--|
| | | | | | |
| | | | | | |
| Example: <div> <div>(hexadecimal)</div> <div>Command</div> <div>command execution time</div> </div> <div> <div>(hexadecimal)</div> <div>Standard period 156 hours</div> <div>CRC of 0x009c</div> </div> | | | | | |

Table 39

3.12 Checksum Calculation

Table 40

| Property | Value | Example code (C/C++) |
|----------|-------|--|
| | | <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 < count; ++current_byte) { crc ^= (data[current_byte]); for (crc_bit = 8; crc_bit > 0; --crc_bit) { if (crc & 0x80) crc = (crc << 1) ^ CRC8_POLYNOMIAL; else crc = (crc << 1); } } return crc; }</pre> |

Table 40

4 Mechanical Specifications

4.1 Package Outline

Figure 3

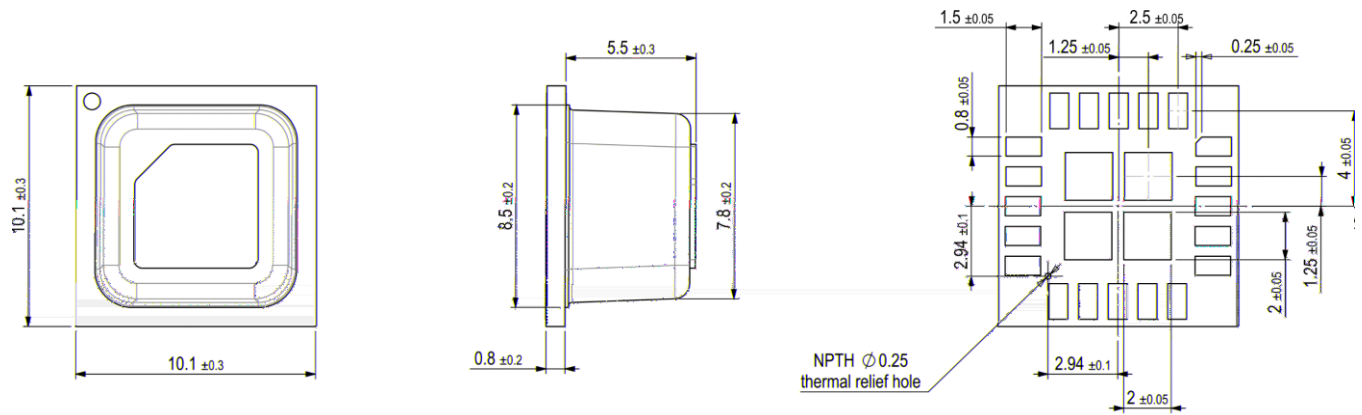


Figure 3

4.2 Land Pattern Recommendation

Figure 4

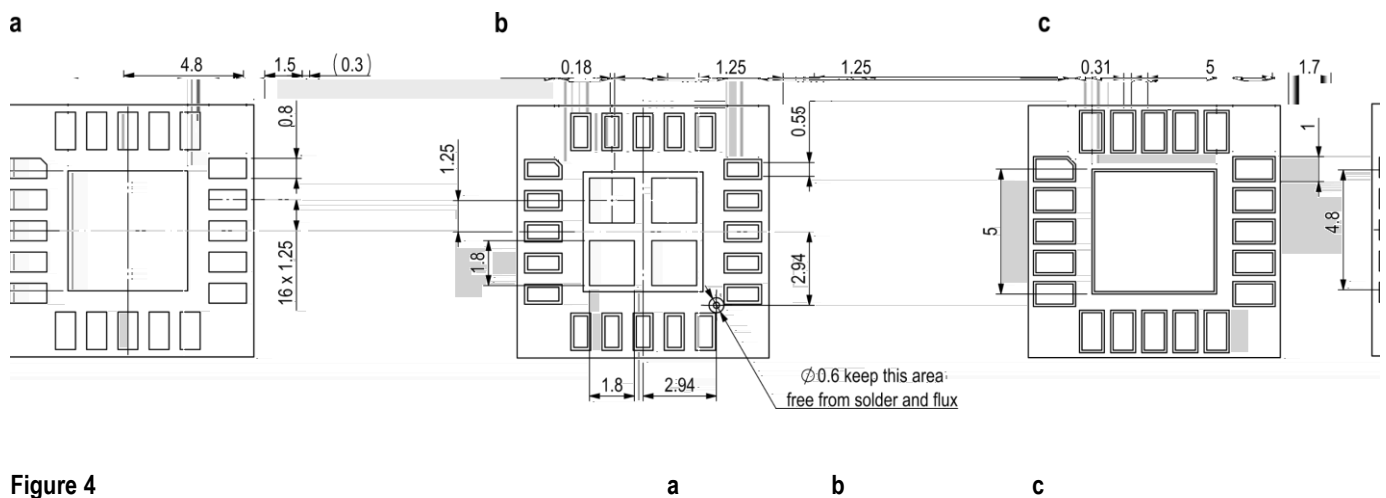


Figure 4

4.3 Tape & Reel Package
Figure 5

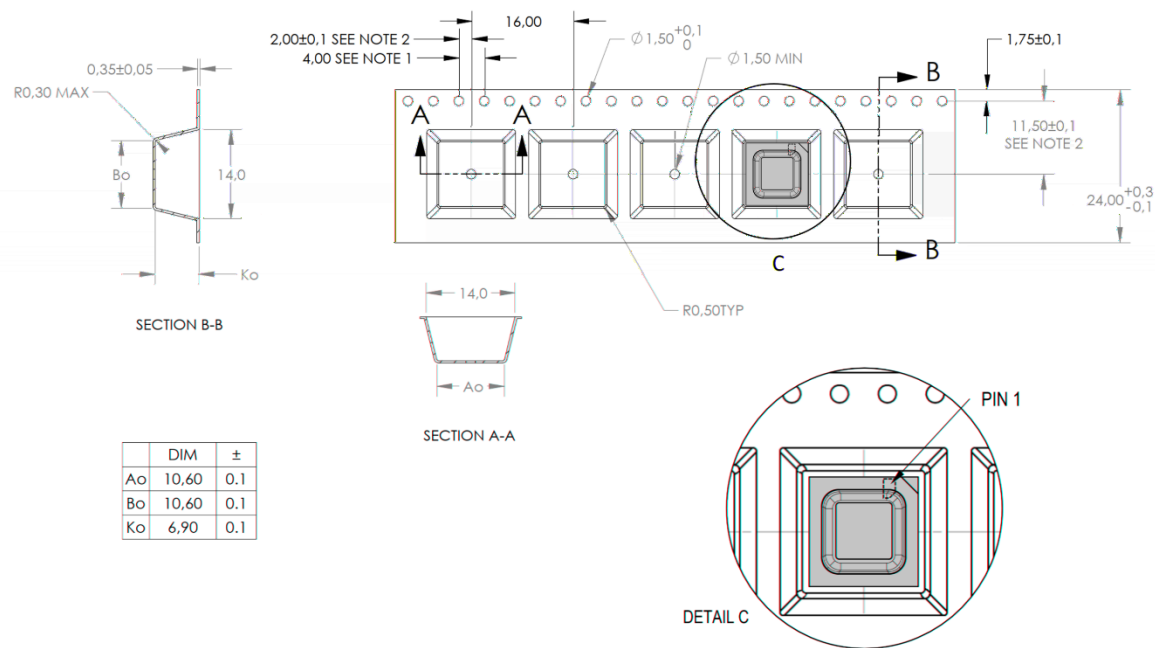


Figure 5

4.4 Moisture Sensitivity Level

4.5 Soldering Instructions

Table 41

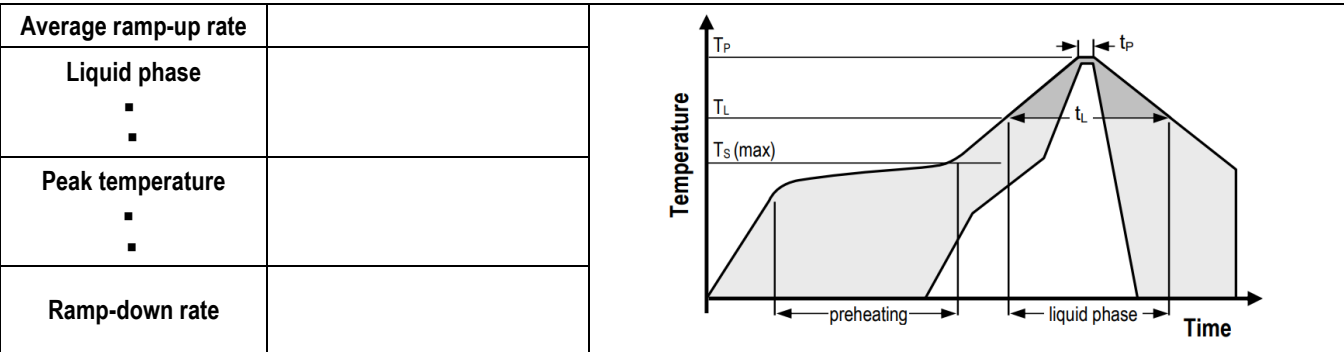


Table 41

4.6 Traceability and Identification

Figure 6

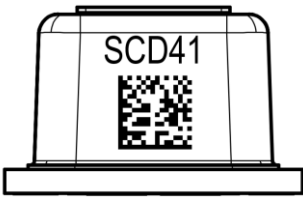


Figure 6:

5 Ordering Information

Table 42

| Part Name | Description | Ordering quantity (pcs) | Product Number |
|-----------|-------------|-------------------------|----------------|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Table 42

5.1 Historical Information

Table 43

| Period Active | Product Number | Note |
|---------------|----------------|------|
| | | |
| | | |

Table 43:

6 Revision History

[illegible]

| | | | |
|--|--|--|--|
| | | | |
|--|--|--|--|

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.

ESD Precautions

Warranty

-
-
-
-

Headquarters and Subsidiaries

| | | |
|--------------------------|--------------------------|--------------------------|
| Sensirion AG | Sensirion Inc., USA | Sensirion Korea Co. Ltd. |
| | <div></div> <div></div> | <div></div> <div></div> |
| | Sensirion Japan Co. Ltd. | Sensirion China Co. Ltd. |
| <div></div> <div></div> | <div></div> <div></div> | <div></div> <div></div> |
| Sensirion Taiwan Co. Ltd | | |
| <div></div> <div></div> | <div></div> | |