I²C communication guide

K20/K22/K30/K33/K45/K50 platforms

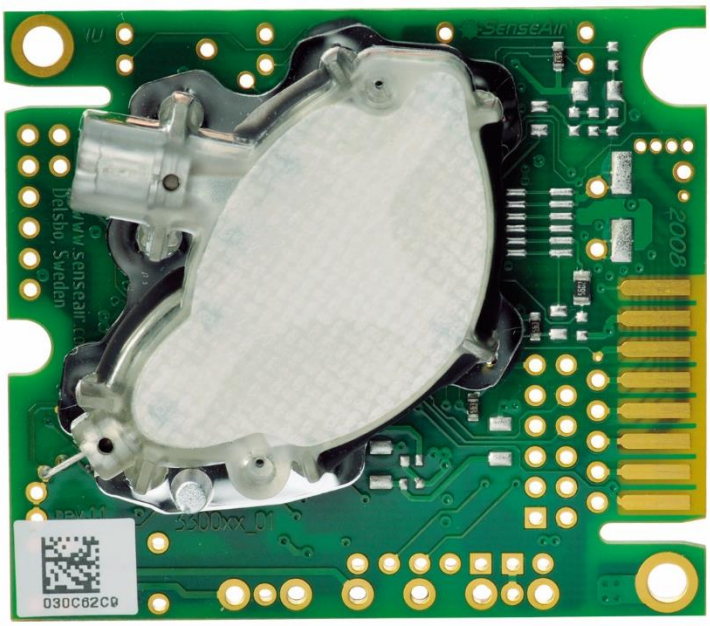


Table of contents:

1. Introduction 2
2. Electrical connection 2
3. Bus characteristics 7
4. Data transfer 12
5. Read/write RAM/EEPROM sequences 16
6. Limitations 19
7. Flowchart of controller operation and error handling 20
8. Sensor memory guide 23
9. Appendix A, Checksum calculation example: 27
10. Appendix B, I²C bus transaction examples 28
11. Appendix C, I²C bus terminals on factory connector 31

Senseair
an Asahi Kasei company

©2022 Senseair AB. All rights reserved.

1. Introduction

This document is a guide to how to communicate with CO_2 sensors in the K20/K22/K30/K33/K45/K50 platforms from Senseair AB using I^2C bus.

Communication is implemented using I^2C. Sensor operates in standard mode at rate up to 100kbit/s and serves as target only.

I^2C communication bus is described in "THE I^2C-BUS SPECIFICATION" (Philips Semiconductors, 2014). I^2C is a trademark of Philips Corps. The document can be downloaded at https://www.nxp.com/docs/en/user-guide/UM10204.pdf
One can find useful application information in AN10216 available from www.nxp.com.

Readers aimed to connect a sensor to a controller system fast should first read chapter 2, Electrical connection, then proceed with chapter 5, Read/write RAM/EEPROM sequences, chapter 6, Flowchart of controller operation and error handling, chapter 8, Sensor memory guide and after that look at the application examples in Appendix B.

2. Electrical connection

2.1. I^2C specification requirements

I^2C-bus employs 2 bi-directional lines SCL and SDA, connected to a positive supply voltage via a current source or pull-up resistors. The output stages of devices connected to the bus must have an open-drain or open-collector to perform the wired-AND function. See "THE I^2C-BUS SPECIFICATION" for more information on electrical connections, section 7 of that document describes how to choose pull-up resistors to meet required noise margin.

2.2. Electrical schematics and specification

The simplest connection to the sensor consists of 4 wires:

- Power "+", often referred as G+
- Power "-", sensor's ground, referred as G0 or GND.
- I^2C-bus bi-directional line SCL
- I^2C-bus bi-directional line SDA

Required logical levels on SCL and SDA lines depend on DVCC supplied on sensor's processor. DVCC is not the same for all models. If processor is powered by 3.3V, it does not tolerate 5V logical levels. See more details in 2.4.
If DVCC differs from I/O voltage of external processor, a level translator may be required. DVCC from the sensor is provided on the connector in order to support connection of a voltage translator. See more details in 2.4.

©2022 Senseair AB. All rights reserved.

The electrical schematic for the sensors in the K20/K22/K30/K33/K45/K50 series is shown in Figure 1. Both SCL and SDA lines have the processor's internal pull-up, which is connected in parallel with an external resistor. There is also a series resistor for each of the lines.

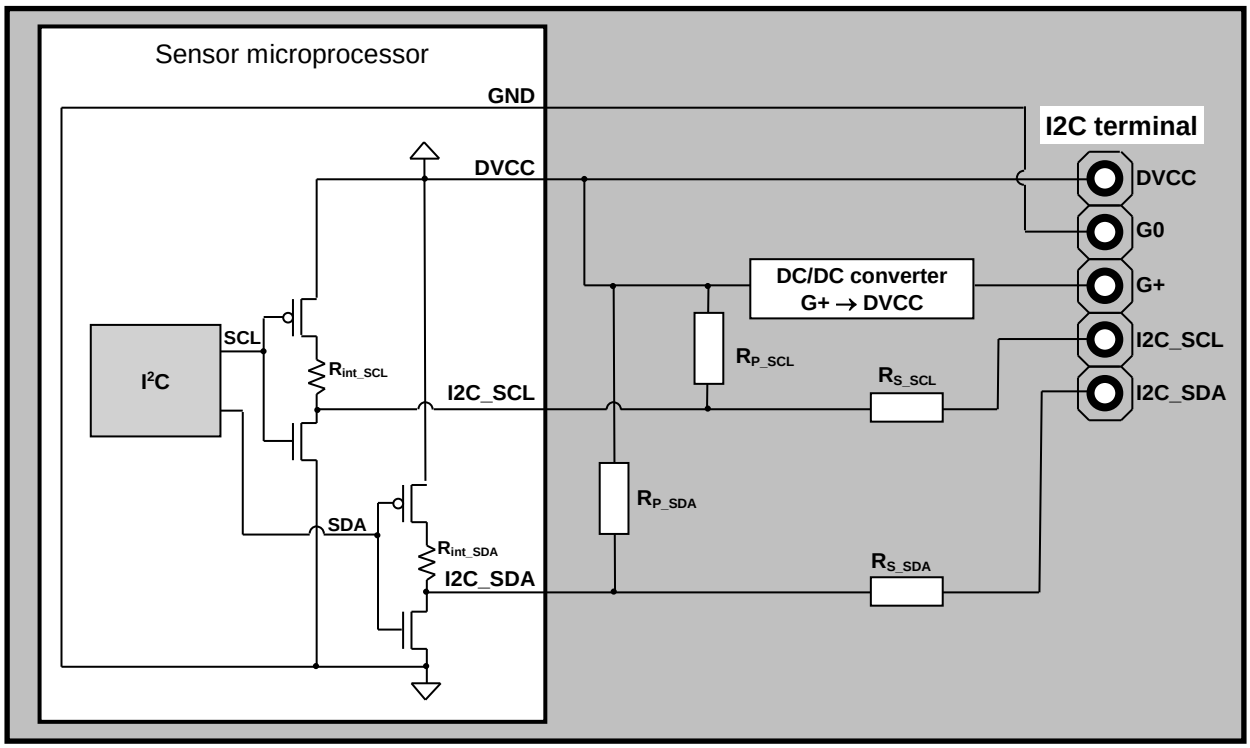


Figure 1. I²C sensor's internal connections

NOTE: K30 sensors have a 10R resistor between regulator output and DVCC pin on I2C terminal. This resistor is not shown on the drawing above.
Resistor values in Figure 1 in the K20/K22/K30/K33/K45/K50 series are summarised in Table 1. Processor's specification on input and output levels of general purpose pins used for I2C communication is quoted in

Resistor	Min	Typ	Max	Notes
Rp_scl (I2C SCL Pull-up)		56 kOhm		5% tolerance resistor
Rp_sda (I2C SDA Pull-up)		56 kOhm		5% tolerance resistor
Rs_scl (I2C SCL Series resistor)		56 Ohm		5% tolerance resistor
Rs_sda (I2C SDA Series resistor)		56 Ohm		5% tolerance resistor
SDA processor internal pull-up	4 kOhm	5.6 kOhm	8 kOhm	Specified in processor's data sheet, only applied if sensor drive the line

.

Resistor	Min	Typ	Max	Notes
Rp_scl (I²C SCL Pull-up)		56 kOhm		5% tolerance resistor
Rp_sda (I²C SDA Pull-up)		56 kOhm		5% tolerance resistor
Rs_scl (I²C SCL Series resistor)		56 Ohm		5% tolerance resistor
Rs_sda (I²C SDA Series resistor)		56 Ohm		5% tolerance resistor
SDA processor internal pull-up	4 kOhm	5.6 kOhm	8 kOhm	Specified in processor's data sheet, only applied if sensor drive the line

Table 1

©2022 Senseair AB. All rights reserved.

Table 3-5. 5V and 3.3V DC GPIO Specifications

Symbol	Description	Min	Typ	Max	Units	Notes
V_{OH}	High Output Level	Vdd - 1.0	–	–	V	IOH = 10 mA, Vdd = 4.75 to 5.25V (maximum 40 mA on even port pins (for example, P0[2], P1[4]), maximum 40 mA on odd port pins (for example, P0[3], P1[5])). 80 mA maximum combined IOH budget.
V_{OL}	Low Output Level	–	–	0.75	V	IOL = 25 mA, Vdd = 4.75 to 5.25V (maximum 100 mA on even port pins (for example, P0[2], P1[4]), maximum 100 mA on odd port pins (for example, P0[3], P1[5])). 150 mA maximum combined IOL budget.
V_{IL}	Input Low Level	–	–	0.8	V	Vdd = 3.0 to 5.25.
V_{IH}	Input High Level	2.1	–		V	Vdd = 3.0 to 5.25.
V_H	Input Hysterisis	–	60	–	mV	
I_{IL}	Input Leakage (Absolute Value)	–	1	–	nA	Gross tested to 1 µA.
C_{IN}	Capacitive Load on Pins as Input	–	3.5	10	pF	Package and pin dependent. Temp = 25°C.
C_{OUT}	Capacitive Load on Pins as Output	–	3.5	10	pF	Package and pin dependent. Temp = 25°C.

Table 2

2.3.I²C connection on different sensor models

The I²C Bus pins are available at I2C terminals on most of the sensor platforms (see Figure 2, 3 and 4).

K33

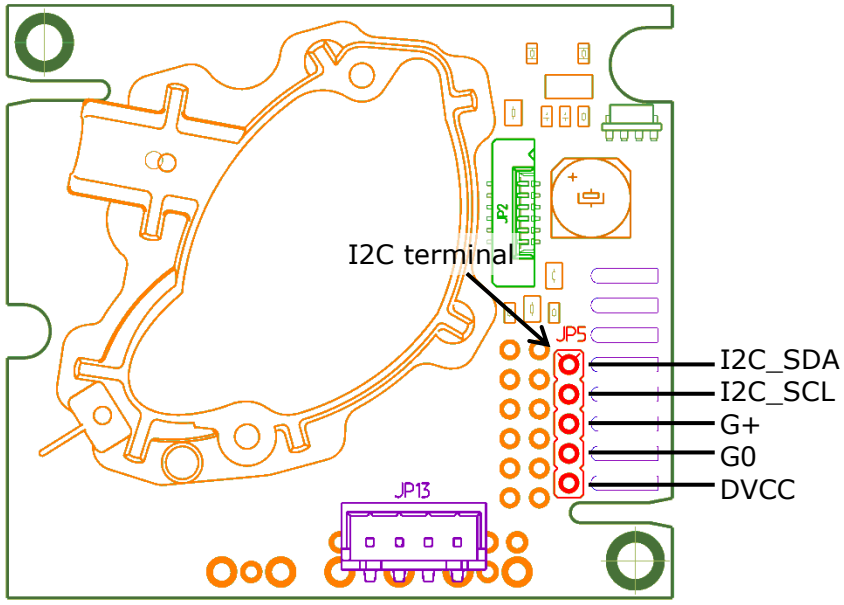


Figure 2. I²C terminal on K33 sensor

©2022 Senseair AB. All rights reserved.

K22

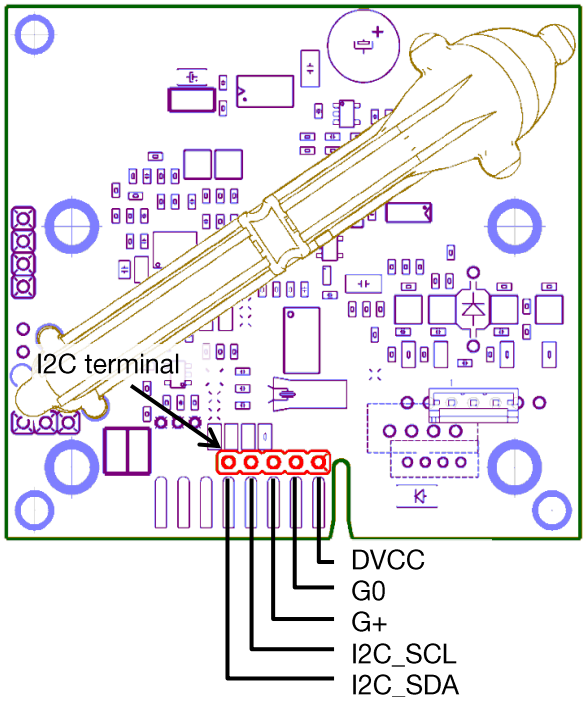


Figure 3. I2C terminal on K22 sensor

K30

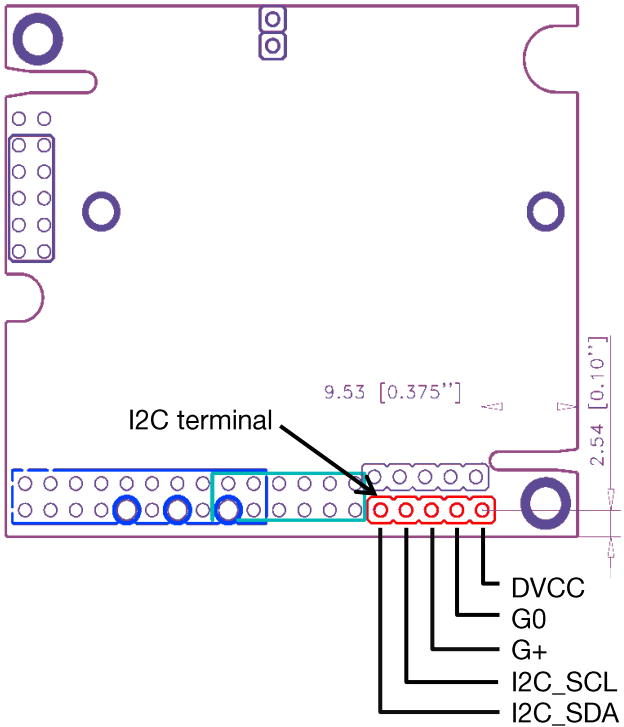


Figure 4. I²C terminal on K30 sensor

I2C bus pins are available on factory connector as well. Refer Appendix C, I2C bus terminals on factory connector for details.

©2022 Senseair AB. All rights reserved.

2.4. DVCC and supply voltages

DVCC voltage is different for different models in the K20/K22/K30/K33/K45/K50 series, as described in Table 3. This needs to be considered when interfacing the sensors since it might be necessary to use level conversion circuitry in some cases. Descriptions of such circuitry and other examples are available from the internet (e.g. https://www.maximintegrated.com/en/design/technical-documents/app-notes/1/1159.html)

Sensor type	DVCC voltage	Supply voltage	Notes
K20-4B	3.3 V	4.5-7.0 VDC	
K20-4BP	3.3 V	4.5-7.0 VDC	
K20-FH	5.0 V	5.0 VDC	Supply voltage is fed to the DVCC
K20-FHP	5.0 V	5.0 VDC	Supply voltage is fed to the DVCC
K20-PO	3.3 V	1.9-3.5 VDC	
K20-POP	3.3 V	1.9-3.5 VDC	
K22-4B	3.3 V	4.5-12.0 VDC	
K22-FH	5.0 V	5.0 VDC	Supply voltage is fed to the DVCC
K22-LN	3.3 V	7.5-12.0 VDC	
K22-PWM	3.3 V	4.5-12.0 VDC	
K22-OC	3.3 V	4.5-12.0 VDC	
K30	3.3 V	4.5-14.0 VDC	
K30-LN	3.3 V	7.2-14.0 VDC	
K33	3.3 V	4.5-14.0 VDC	
K45	5.0 V	24.0 VAC/VDC	
K50	5.0 V	24.0 VAC/VDC	

Table 3 DVCC and power supply for different sensor types

DVCC can deliver a small amount of current to drive external circuitry. Maximum allowed current available depends on model. As a general rule, current shouldn't exceed 12 mA.

©2022 Senseair AB. All rights reserved.

3. Bus characteristics

3.1. Functional description

The sensors in the K20/K22/K30/K33/K45/K50 series use the I²C Bus for communication with other systems. Sensor acts as a target device on the I²C bus, which means that it must be controlled by an I²C controller device. It can operate both as receiver and transmitter, but the controller device generates clock signals on SCL line and controller device determines which mode is activated.
The physical layer of the I²C Bus is implemented in a dedicated hardware block in the sensor processor. The I²C hardware block is configured and controlled by sensor firmware.

Parameter	Value	Notes
Controller/target mode	Target	Sensor will never initiate communication; it only replies to controller
Data rate	100kbits/s	
Addressing mode	7bit	
Address range	0x00 to 0x7F	Address is configured in EEPROM. Default address after production is 0x68 “Any sensor” address is 0x7F. Sensor answers on data transfers with this address disregarding configured sensor address.

Table 4 I²C parameters for K20/K22/K30/K33/K45/K50 sensors

The following bus conditions are defined; see “THE I²C-BUS SPECIFICATION”:

3.2. Bus not busy

Both data SDA and clock SCL lines remain high.

3.3. Start condition

A high-to-low transition of the SDA line while the clock SCL is high, determines a Start condition. All commands must be preceded by a Start condition.

3.4. Stop condition

A low-to-high transition of the SDA line while the clock SCL line is high determines a Stop condition. All operations must end with a Stop condition.

3.5. Data valid

The state of the data SDA line represents valid data when, after a Start condition, the data SDA line is stable for the duration of the high period of the clock signal.
The data on the SDA line must be changed during the low period of the clock signal.
Each data transfer is initiated with a Start condition and terminated with Stop condition.

©2022 Senseair AB. All rights reserved.

3.6. Acknowledge

Each **receiving** device, when addressed, is obliged to generate an Acknowledge signal after the reception of each byte. The controller device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during acknowledge clock pulse. Sensor doesn't generate any acknowledge bits if it is busy with some internal task, for example, measurement. It is not an error and controller shall try communication with sensor again. Acknowledgement is shown in Figure 5.

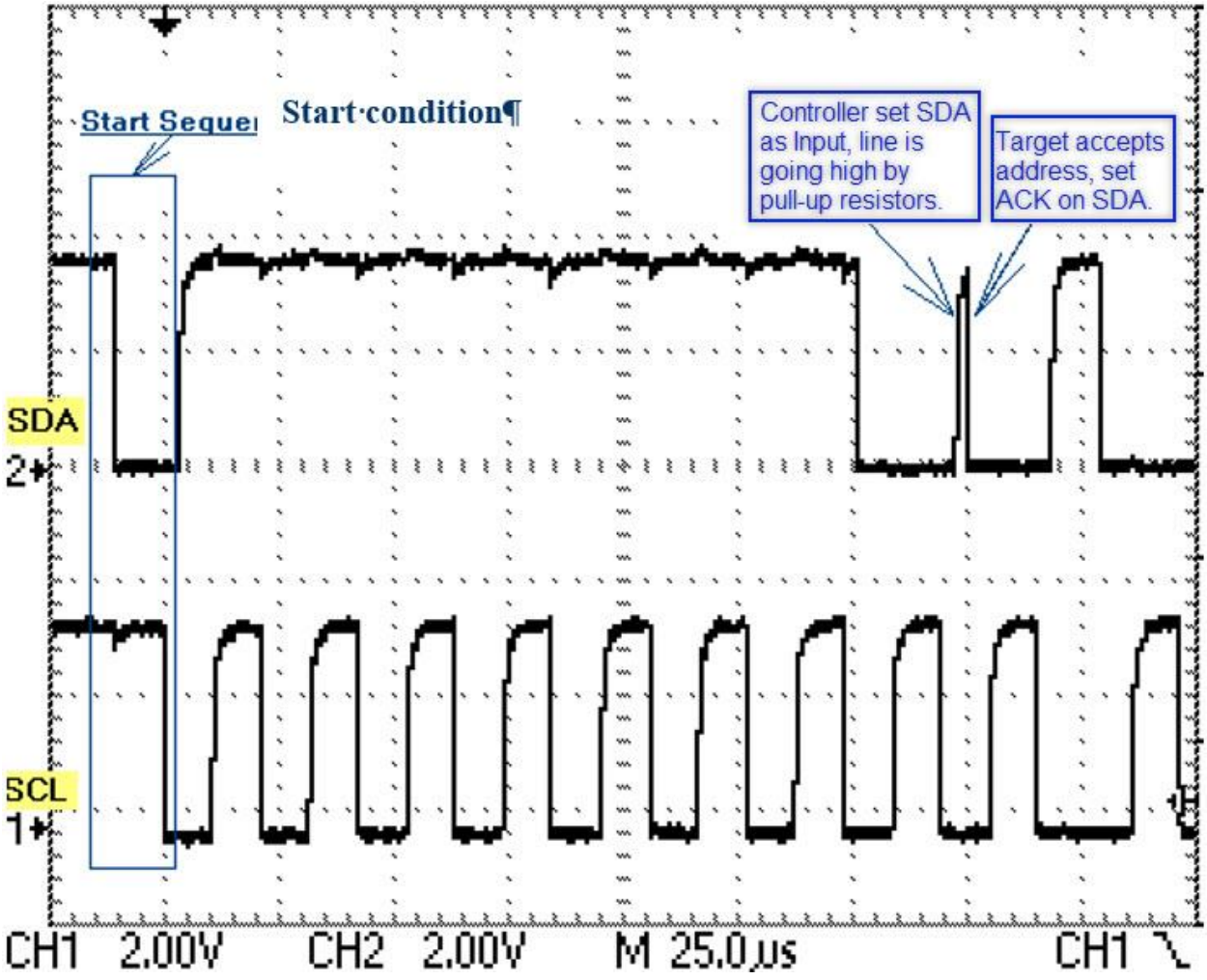


Figure 5. Target acknowledges I²C-address from Controller. I²C address = 0x7F, "any sensor" address. Write command sent with Bit0 = 0.

©2022 Senseair AB. All rights reserved.

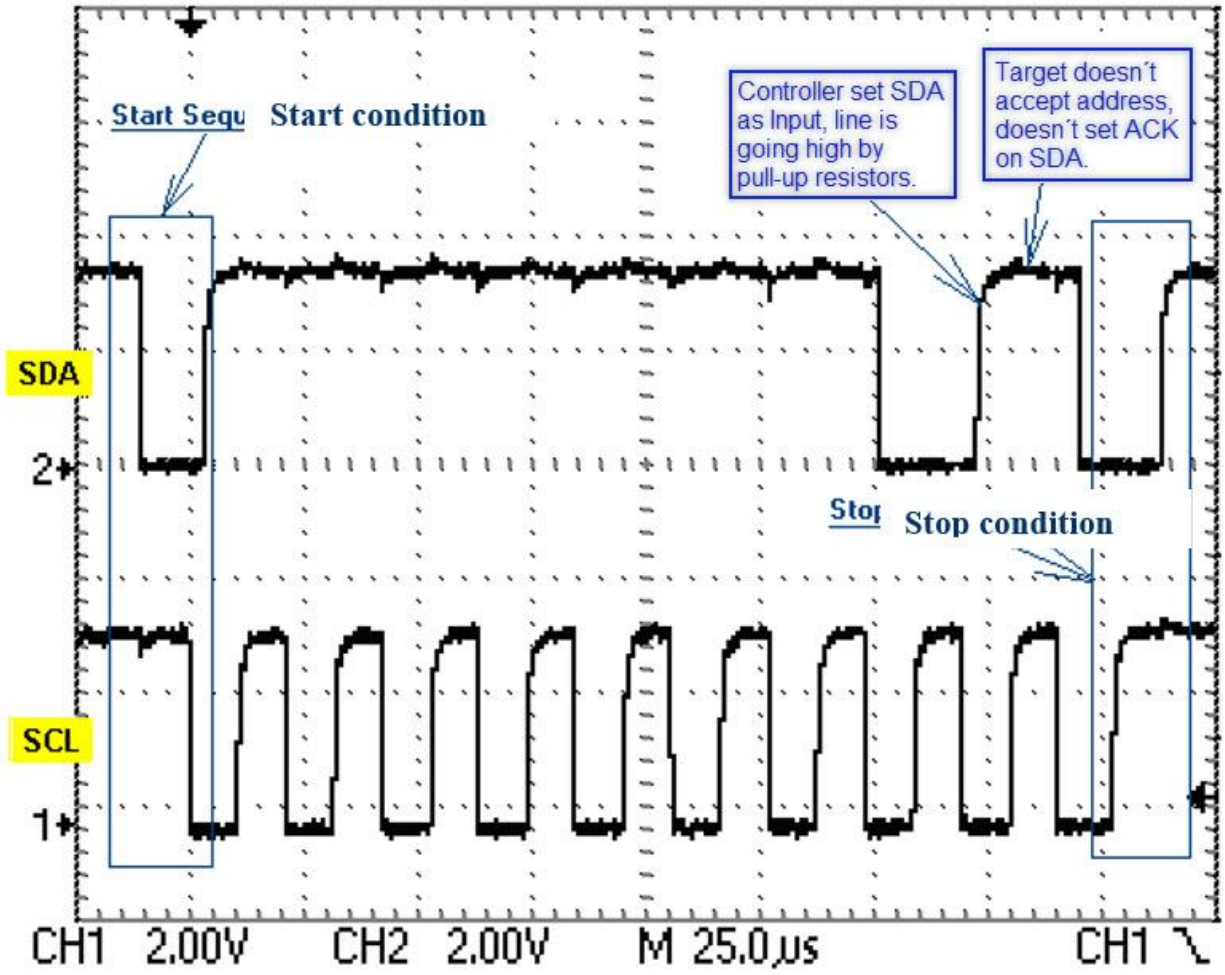


Figure 6. Target doesn't acknowledge I²C-address from Controller. I²C address = 0x7F, "any sensor" address.

©2022 Senseair AB. All rights reserved.

3.7. Use of the clock synchronising mechanism as a handshake

"THE I²C-BUS SPECIFICATION", page 10 and 13, defines a special mechanism to allow target to execute slow operations.

"If target can't receive or transmit another complete byte of data until it has performed some other function… it can hold the clock line SCL Low to force controller into wait state. Data transfer then continues when the target is ready for another byte of data and release clock line SCL."

This mechanism is used by sensor to hold already started communication during measurement, see Figure 7. Controller must support this part of I²C specification.

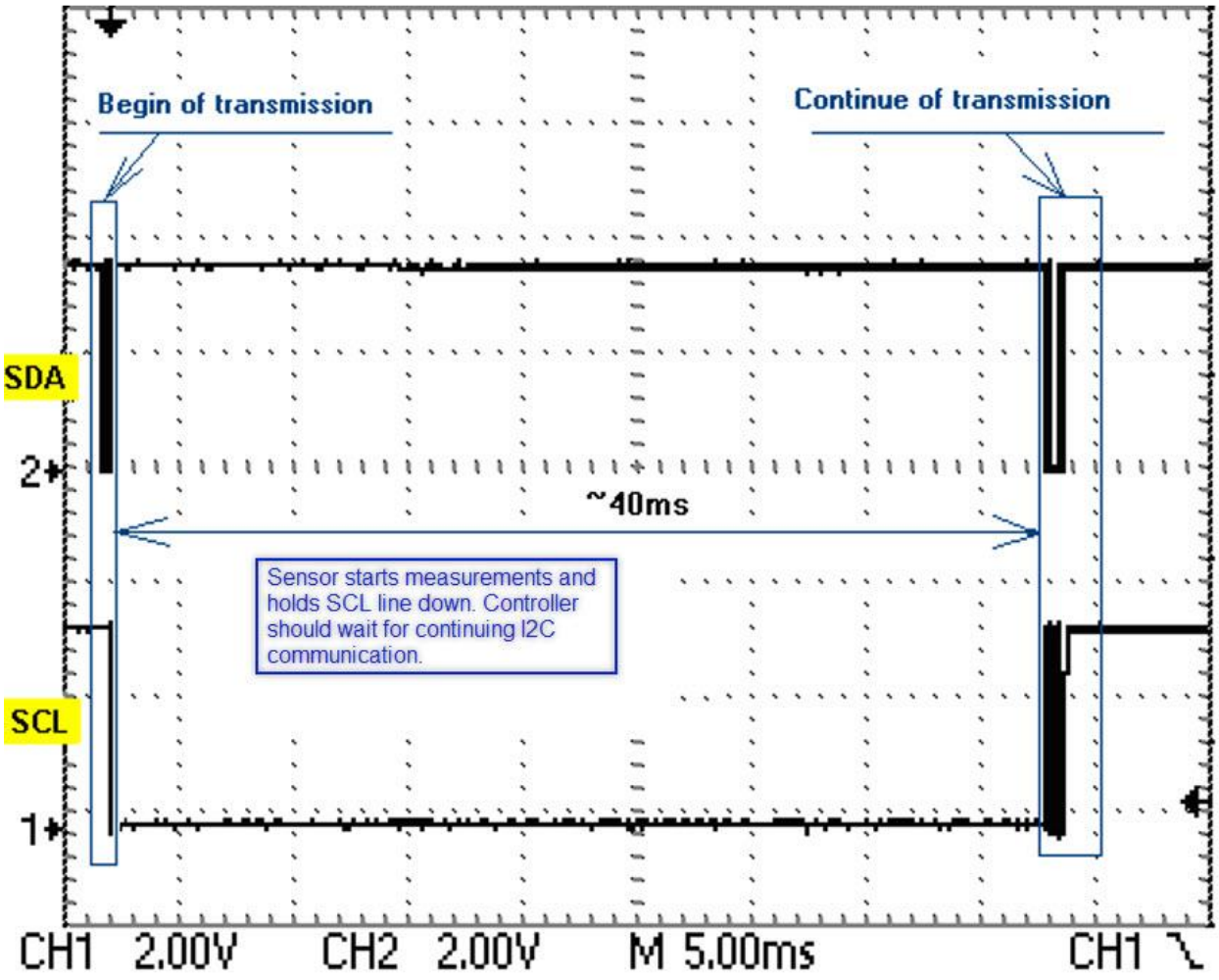


Figure 7. Controller starts transmission, then Sensor (Target) makes measurements and holds line SCL down. After finishing of measurements, Controller continues transmission. As an option, Controller can send Stop Sequence and communicate with another device on bus.

3.8. Bus protocol

A. Data transfer may be initiated only when the bus is not busy.
B. Data transfer may be initiated even when line SDA is kept low by the sensor. This protocol is defined for low power sensors with sleep mode.

©2022 Senseair AB. All rights reserved.

3.9. Timing diagram

There is timing diagram of I2C communication by specification:

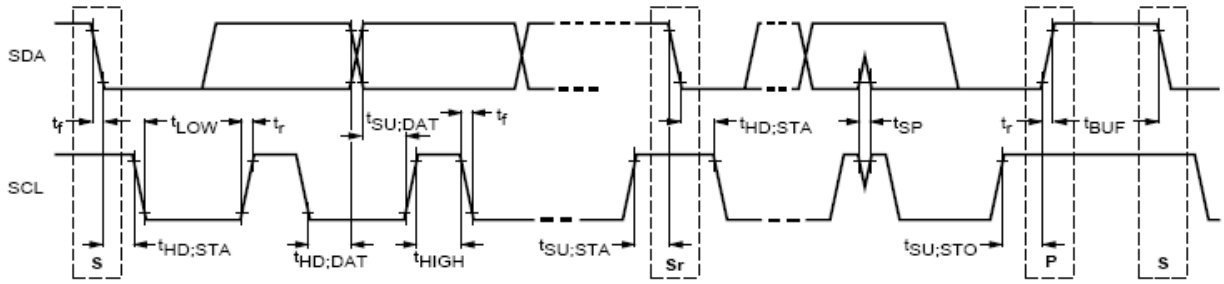


Figure 8

Parameter	Symbol	Min	Max	Unit
SCL clock frequency	f_{SCL}	0	100	kHz
Hold time (repeated) START condition. After this period, the first clock pulse is generated	$t_{HD;STA}$	4.0	—	µs
LOW period of the SCL clock	t_{LOW}	4.7	—	µs
HIGH period of the SCL clock	t_{HIGH}	4.0	—	µs
Set-up time for a repeated START condition	$t_{SU;STA}$	4.7	—	µs
Data hold time for I2C-bus devices	$t_{HD;DAT}$	0 (1)	3.45 (2)	µs
Data set-up time	$t_{SU;DAT}$	250	—	ns
Rise time of both SDA and SCL signals	t_r	—	1000	ns
Fall time of both SDA and SCL signals	t_f	—	300	ns
Set-up time for STOP condition	$t_{SU;STO}$	4.0	—	µs
Bus free time between a STOP and START condition.	t_{BUF}	4.7	—	µs
Capacitive load for each bus line	C_b	—	400	pF
Noise margin at the LOW level for each connected device (including hysteresis)	V_{nL}	0.1	—	V_{DD}
Noise margin at the HIGH level for each connected device (including hysteresis)	V_{nH}	0.2	—	V_{DD}

Table 5

Notes:

1. A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the VIHmin of the SCL signal) to bridge the undefined region of the falling edge of SCL.
2. The maximum $t_{HD;DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.

©2022 Senseair AB. All rights reserved.

4. Data transfer

4.1. Communication session

Any communication session with sensor consists of 2 data transfer slots with one wait slot between them. The first data transfer, named Request, contains command and eventual data to write to the sensor. The wait time slot is required to let the sensor to recognise and execute command and prepare response.

The second data transfer, named Response, contains check of completion of command execution and reads data from sensor. The “complete” bit has to be checked to verify that the command was successfully executed. Controller may need to repeat attempts to read response until a valid frame with “complete” bit set to one is received or timeout is occurred. Busy sensor may be indicated by absence of acknowledge bit as well.

This basic scenario of communication session is illustrated on Figure 9.

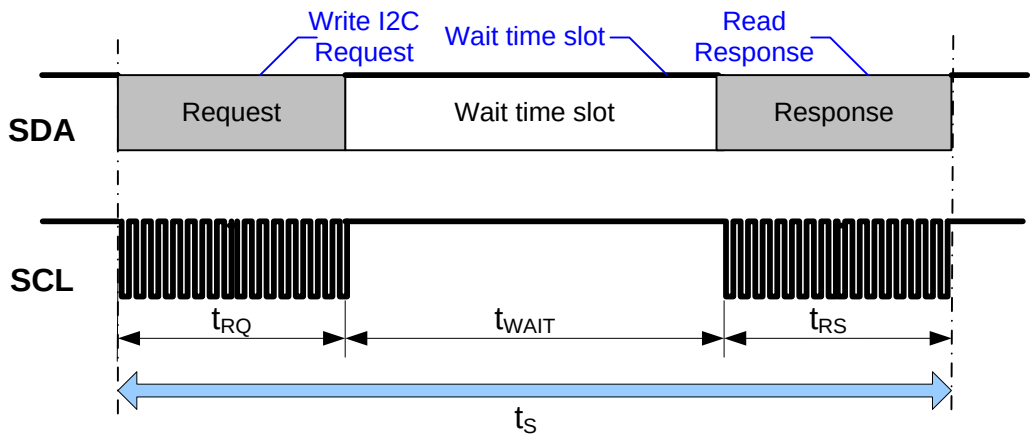


Figure 9. Communication timing diagram

Low power sensor models may require some time for wake up. They doesn’t answer on the first request and it’s worth to provide pulse to wake up them as it’s shown on Figure 10

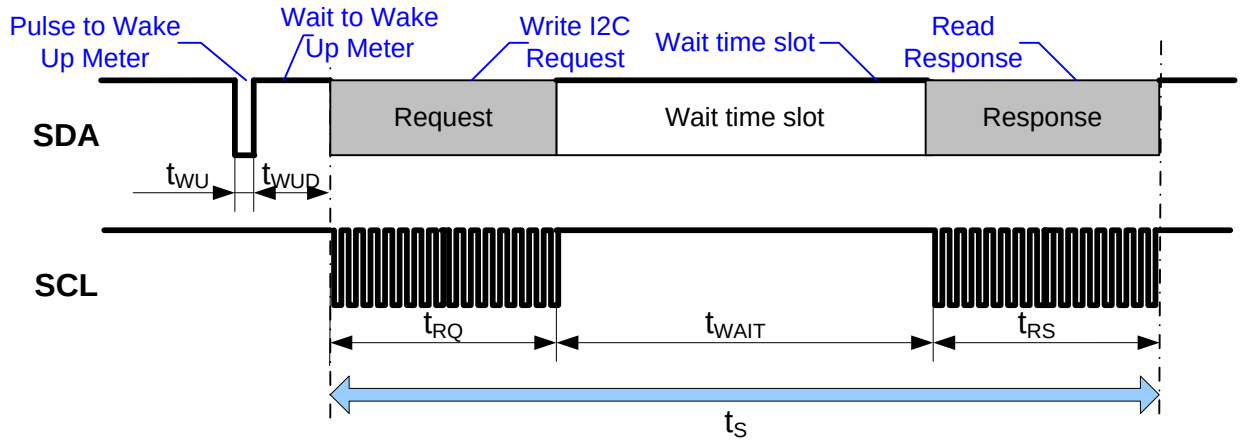


Figure 10. Communication timing diagram for low power sensors (K20-4B, K22-4B and K33 BLG/ELG)

©2022 Senseair AB. All rights reserved.

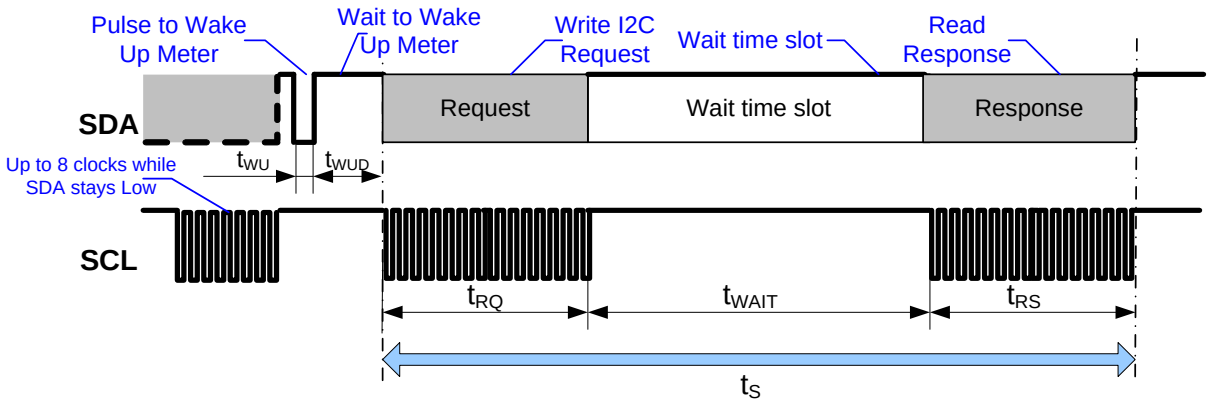


Figure 11. “Save” communication timing diagram with recovery from the situation when sensor’s I2C hardware holds line SDA low

4.2.Timing specification

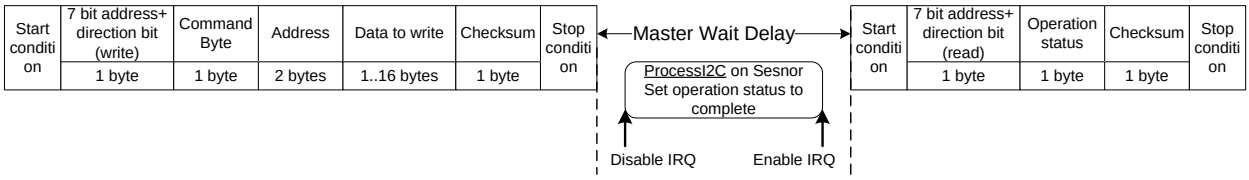
Time parameters shown on Figure 9, Figure 10 and Figure 11 are specified in Table 6

Symbol	Description	Min	Typical	Max	Notes
t_{RQ}	Time of Request	-	-	120 ms	Must be checked by I2C-Controller
t_{WAIT}	Wait time (delay)	1 ms (TBD)	20 ms	(TBD)	
t_{RS}	Time of response	-	-	120 ms	Must be checked by I2C-Controller
t_S	Total Session time	-	~40 ms	160 ms	Must be checked by I2C-Controller
t_{WU}	Time of Pulse for Wake-up	TBD	300 µs	TBD	
t_{WUD}	Delay for Wake-up	TBD	1 ms	TBD	

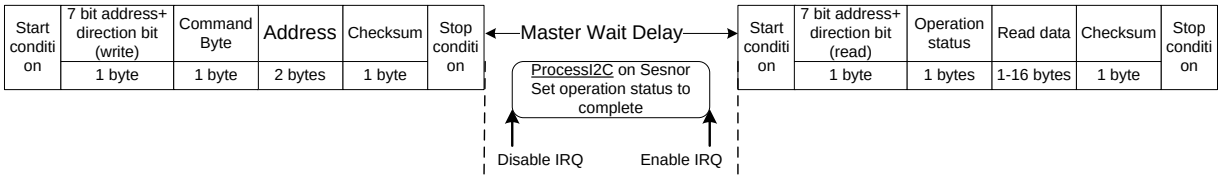
Table 6

Overview of data packages in communication session

Write RAM/EEPROM



Read RAM/EEPROM



©2022 Senseair AB. All rights reserved.

4.3. Details of bit fields

Byte position	Bits in byte	Field	Value	Interpretation
0	7:1	7-bit I²C-Target address	0x00 to 0x7F	I²C address. Default Sensor Address is 0x68. On Point-to-Point connections Controller also can use address 0x7F – "any sensor".
	0	direction bit	0 or 1	Read/Write bit encoding Read = 1, Write = 0.
1	7:4	Command	1,2,3,4	0x1 - Write RAM 0x2 - Read RAM 0x3 - Write EE (1)(2) 0x4 - Read EE (1) 0x0, 0x5..0xF – Reserved for future use.
	3:0	Number of Data bytes	0..15	The number of bytes to read/write. Value range from 1 to 15; 0 means 16 bytes. E.g. for write 1 byte to sensor RAM this byte should be 0x11 (Command = WriteRAM, NrOfBytes = 1)
2:3		RAM Address, MSB first		Address in sensor's RAM.
4:(3+N)		N bytes of Data (N=1..16)		Data for Writing (N = 1..16 bytes) This field is present only for commands "Write RAM" and "Write EEPROM". For commands "Read RAM" and "Read EEPROM" N=0.
4+N		Checksum		Arithmetic sum of the bytes sent (not including first byte with address and direction bit). See Appendix A for code example.

Table 7 Encoding of fields Request

Byte position	Bits in byte	Field	Value	Interpretation
0	7:1	7-bit I²C-target address	0x68	I²C address. Default Sensor Address is 0x68. On Point-to-Point connections Controller also can use address 0x7F – "any sensor".
	0	direction bit	0 or 1	Read/Write bit encoding Read = 1, Write = 0.
1	7:4	Command	1, 2, 3, 4	0x1 - Write RAM 0x2 - Read RAM 0x3 - Write EE (1)(2) 0x4 - Read EE (1) 0x0, 0x5...0xF – Reserved for future use.
	3:1	Reserved	000	Reserved for future use, should be 000 for compatibility.
	0	Complete/ Incomplete bit	**1** (Completed)	Complete/Incomplete bit. 1-Complete, 0-Incomplete.
2:(1+N)		N bytes of Data (N=1..16)		Data read (N = 1..16 bytes) This field is present only for commands "Read RAM" and "Read EEPROM". For command "Write RAM" and "Write EEPROM" N=0.
2+N		Checksum		Arithmetic sum of the bytes sent (not including first byte with address and direction bit). See Appendix A for code example.

©2022 Senseair AB. All rights reserved.

Table 8 Response (Request Completed)

Byte position	Bits in byte	Field	Value	Interpretation
0	7:1	7-bit I2C-Target address	0x68	I²C address. Default Sensor Address is 0x68. On Point-to-Point connections Controller also can use address 0x7F — “any sensor”.
	0	direction bit	0 or 1	Read/Write bit encoding Read = 1, Write = 0.
1	7:4	Command	1, 2, 3, 4	0x1 - Write RAM 0x2 - Read RAM 0x3 - Write EE (1)(2) 0x4 - Read EE (1) 0x0, 0x5...0xF – Reserved for future use.
	3:2	Reserved	00	Reserved for future use, should be 00 for compatibility.
	1	Invalid Data	0 or 1	See Note 2
	0	Complete/ Incomplete bit	**0** (Incomplete)	Complete/Incomplete bit. 1-Complete, 0-Incomplete.
2:(N)		N bytes of Data (N=1..16)		N = expected number of Data bytes (1..16) for commands “Read RAM” and “Read EEPROM”. (3)
2+N		Checksum		Arithmetic sum of the bytes sent (not including first byte with address and direction bit). See Appendix A for code example.

Table 9 Response (Request Uncompleted)

Notes:

1. The commands ReadEEPROM and WriteEEPROM are legal only for sensors with external EEPROM! (K22 with EEPROM/K30/K33/K45/K50)

2. External EEPROM has page size of 16 bytes. If data in command “Write EEPROM” crosses bound of the EEPROM page, bit “Write error” is set and write command is not executed.

3. I²C-target sends Operation status byte with flag “Incomplete” until I²C-Controller finishes transaction. I²C-Target cannot process Request when communication is active; hence I²C-Controller have to send Stop Condition if it gets status “Incomplete”. In current implementation of communication, Controller should read all expected bytes of Response, independent of flag “Complete/Incomplete”.

©2022 Senseair AB. All rights reserved.

5. Read/write RAM/EEPROM sequences

5.1. RAM and EEPROM

There are four command sequences supported, “Write RAM”, “Read RAM”, “Write EEPROM” and “Read EEPROM”.
Note that Read/Write EEPROM commands are supported only by sensors with external EEPROM. K20 and K22 sensors without EEPROM don’t respond on EEPROM commands.

Sensor type	EEPROM	Size	Notes
K20-4B	NO		Virtual EEPROM available
K20-4BP	NO		Virtual EEPROM available
K20-FH	NO		Virtual EEPROM available
K20-FHP	NO		Virtual EEPROM available
K20-PO	NO		Virtual EEPROM available
K20-POP	NO		Virtual EEPROM available
K22-4B	NO		Virtual EEPROM available
K22-FH	NO		Virtual EEPROM available
K22-LN	YES	128B	
K22-PWM	YES	128B	
K22-OC	YES	128B	
K30	YES	128B	
K30-LN	YES	128B	
K33	YES	8KB	
K33 BLG/ELG	YES	64KB	
K45	YES	16KB	
K50	YES	8KB	

Table 10

©2022 Senseair AB. All rights reserved.

5.2. Write RAM

Controller writes up to 16 bytes to Sensor's RAM.

Request:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Number of bytes Low nibble	RAM address	Data to write	check sum	I²C Stop condition
	Sensor addr.	0 (write)	0x1	0..0xF				
	1 byte		1 byte		2 bytes	1..16 bytes	1 byte	

"Write Complete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	checksum	I²C Stop condition
	Sensor addr.	1 (read)	0x11			
	1 byte		1 byte		1 byte	

"Write Incomplete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	checksum	I²C Stop condition
	Sensor addr.	1 (read)	0x10			
	1 byte		1 byte		1 byte	

5.3. Read RAM

Controller reads up to 16 bytes from Sensor's RAM.

Request:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Number of bytes Low nibble	RAM address	checksum	I²C Stop condition
	Sensor addr.	0 (write)	0x2	0..0xF			
	1 byte		1 byte		2 bytes	1 byte	

"Read Complete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	Read Data	checksum	I²C Stop condition
	Sensor addr.	1 (read)	0x21				
	1 byte		1 byte		1..16 bytes	1 byte	

"Read Incomplete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	All other bytes	I²C Stop condition
	Sensor addr.	1 (read)	0x20		0x20	
	1 byte		1 byte			

©2022 Senseair AB. All rights reserved.

5.4. Write EEPROM

Controller writes up to 16 bytes to Sensor's external EEPROM (available only in K22/K30/K33/K45/K50 versions)

Request:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Number of bytes Low nibble	EEPROM address	Data to write	check sum	I²C Stop condition
	Sensor addr.	0 (write)	0x3	0..0xF				
	1 byte		1 byte		2 bytes	1..16 bytes	1 byte	

"Write Complete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	checksum	I²C Stop condition
	Sensor addr.	1 (read)	0x31			
	1 byte		1 byte		1 byte	

"Write Incomplete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	checksum	I²C Stop condition
	Sensor addr.	1 (read)	0x30			
	1 byte		1 byte		1 byte	

5.5. Read EEPROM

Controller reads up to 16 bytes from Sensor's external EEPROM (available only in K22/K30/K33/K45/K50 versions)

Request:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Number of bytes Low nibble	EEPROM address	checksum	I²C Stop condition
	Sensor addr.	0 (write)	0x4	0..0xF			
	1 byte		1 byte		2 bytes	1 byte	

"Read Complete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	Read Data	checksum	I²C Stop condition
	Sensor addr.	1 (read)	0x41				
	1 byte		1 byte		1..16 bytes	1 byte	

"Read Incomplete" Response:

I²C Start condition	7-bit I²C Address	Read/ Write bit	Command high nibble	Status bit	All other bytes	I²C Stop condition
	Sensor addr.	1 (read)	0x40		0x40	
	1 byte		1 byte			

©2022 Senseair AB. All rights reserved.

6. Limitations

6.1. Important notice:

Sensor's I²C communication is not implemented as a hardware block but it is implemented in software instead. It imposes its limitations on the performance and compliance with I2C general specification. Limitations include:

1. There is a dead time in communication when sensor performs measurement.
2. Communication with sensors networked on the same I2C bus may be complicated by the fact that there is no hardware detection of the address. The consequence is the response of all sensors on the first byte with address by ACK bit. Read more about ACK bit in chapter Acknowledge.
3. Moreover, if one of networked on the same I2C bus sensors is measuring, it will keep SCL line low until measurement finish. See chapter "Use of the clock synchronising mechanism as a handshake" for more detailed description.

6.2. Sensor dead time:

The sensor is equipped with only a small microcontroller that handles both measurements and communication, since the quality of measurements are of highest priority there are some limitation to the way communication is handled. While the sensor is actively measuring CO_2, communication is switched off (i.e. I²C interrupts are disabled) to not disturb the measurements. This will have the effect that there will be a short period during which the sensor will not respond to I²C Bus activities. This period is short and the impact on communication is very limited. However, when designing an I²C system where the sensor is a component this situation needs to be handled correctly. It may be necessary to implement checks and actions if a command sent to the sensor has failed to execute correctly. The sensor protocol has therefore a mandatory check of status after each command, with 1 bit (complete/not complete) that will tell if the command has been executed by the sensor.

©2022 Senseair AB. All rights reserved.

7. Flowchart of controller operation and error handling

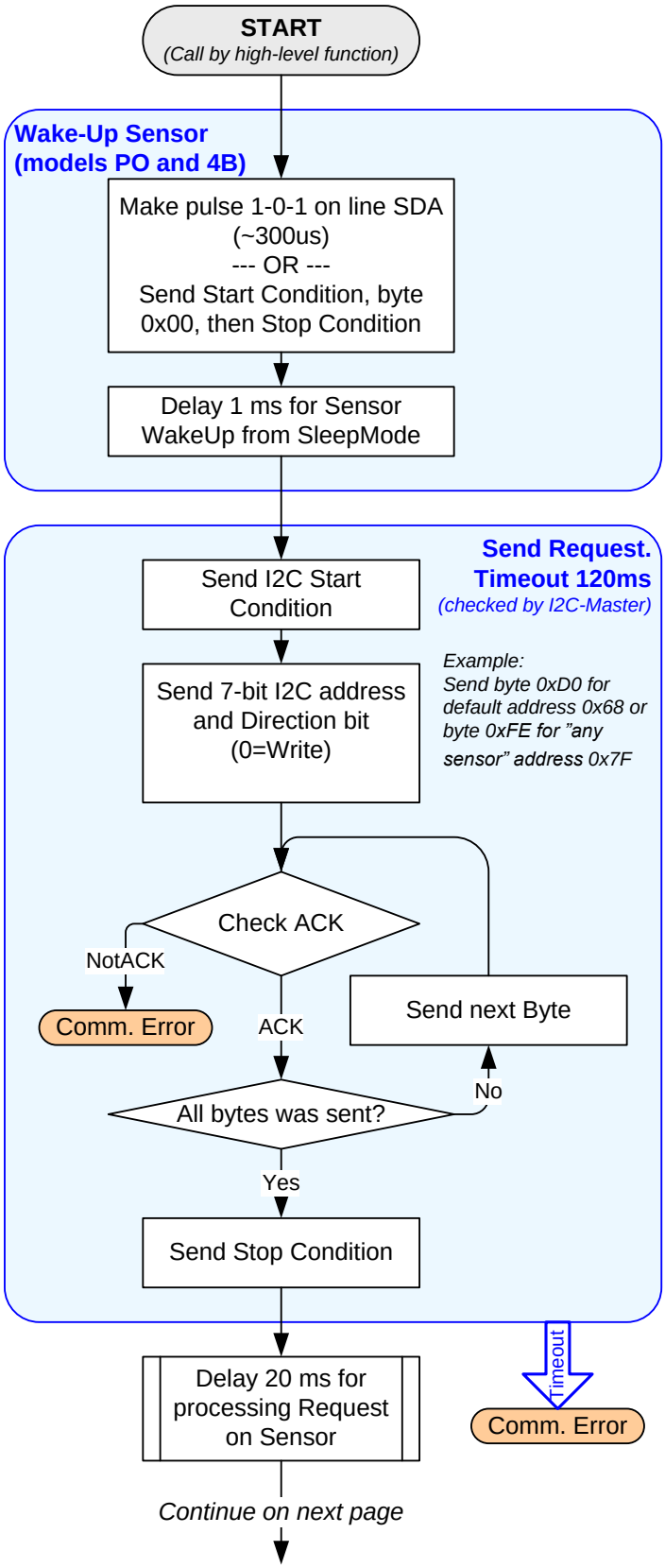
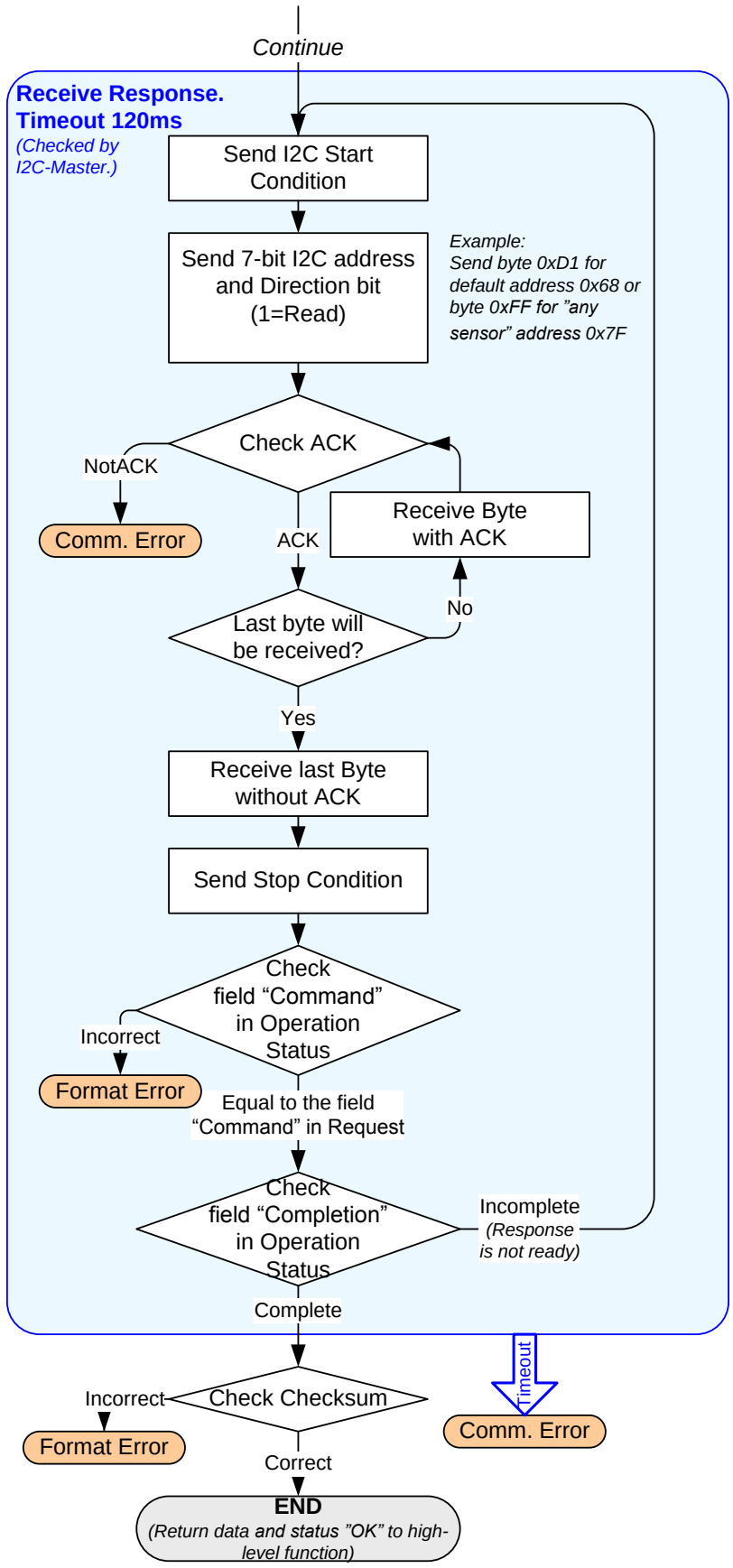


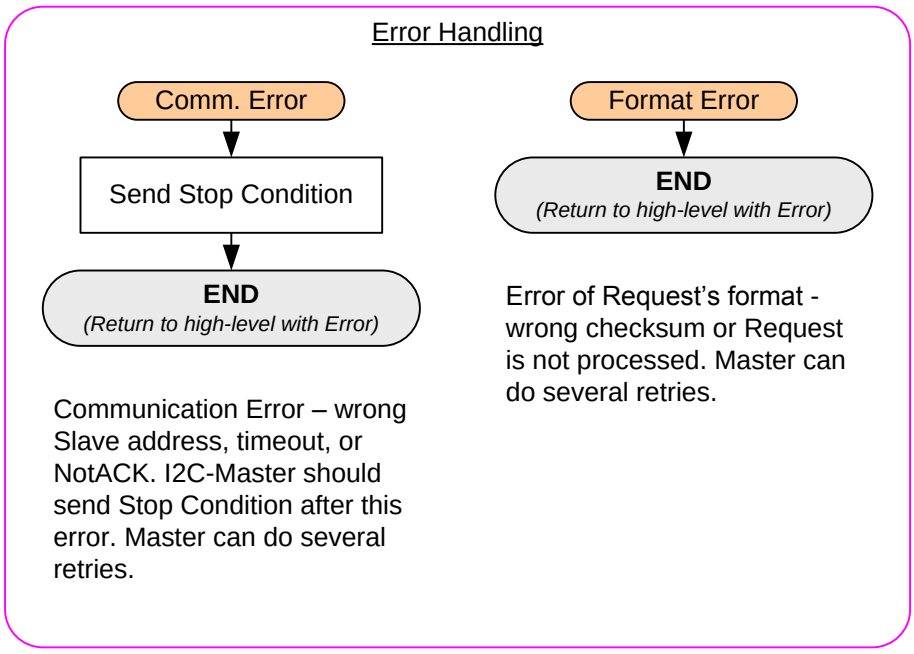
Figure 12.

©2022 Senseair AB. All rights reserved.



©2022 Senseair AB. All rights reserved.

Figure 13. Continues on next page



©2022 Senseair AB. All rights reserved.

8. Sensor memory guide

8.1. RAM/EEPROM write precautions

The Read/Write RAM/EEPROM gives access to the whole memory area of the microprocessor and external EEPROM. The user must take care when writing to not overwrite any RAM/EEPROM location that could compromise the execution of the sensor firmware such as calibration data, stack variables and other local variables.

8.2. Important variables

Memory maps are product specific, but CO_2 value keeps its location.
It is located in RAM at address 0x08 (high byte) and 0x09 (low byte).
Refer examples in Appendix B

Variable	Address	Format	Notes
Space CO_2	0x08 in RAM	2 bytes signed integer Most significant byte at lower address	Negative readings are possible, in particular at nitrogen / zero gas flow test.
Space temperature	0x12 in RAM	2 bytes signed integer, two decimals (value xxxx = xx.xx°C) Most significant byte at lower address	Only K33 BLG/ELG and K45
RH	0x14 in RAM	2 bytes signed integer, two decimals (value xxxx = xx.xx%) Most significant byte at lower address	Only K33 BLG/ELG
Error Status	0x1E in RAM	1 Byte, bit field	Error discovered during self test is indicated by setting one of the bits in the Error Status byte to 1. Normal operation without errors is indicated by Error Status = 0. Details about error status are product specific and may be found in product technical documentation.

Table 11 Measured value and status

Variable	Product	Memory map identification	Address	Notes
MeterControl	K22	<=0x0A	0x3E, EEPROM	Only on sensors with external EEPROM
	K30	<=0x27	0x3E, EEPROM	
	K33	<=0x2B <=0x5D	0x03, EEPROM	
	K50	<=0x2D	0x03, EEPROM	

Table 12 Meter Control byte

See information on use of meter control in the following bellow chapters.

©2022 Senseair AB. All rights reserved.

8.3.Sensor and firmware identifications

Variable	Address	Format	Notes
Firmware Type	0x62 RAM	Byte unsigned	
Revision Main	0x63 RAM	Byte unsigned	
Revision sub	0x64 RAM	Byte unsigned	

Table 13 Firmware identification bytes

Variable	Address	Format	Notes
Sensor Type Id	0x2C RAM	3 bytes unsigned word, Most significant byte first	This parameter is copied to the RAM from EEPROM at power up.
Sensor's serial number	0x28 RAM	4 bytes unsigned word, Most significant byte first	This parameter is copied to the RAM from EEPROM at power up.
Memory map identification	0x2F RAM	Byte unsigned	This parameter is written to RAM by firmware at power up.

Table 14 Sensor type, number and memory map identification bytes

Variable	Address	Format	Notes
I^2C address	0x20 in RAM 0x00 in EEPROM	Byte unsigned	Sensor address is stored in EEPROM and is copied to RAM at sensor start. In order to change I^2C address it's necessary to write value to EEPROM and power down and then power up the sensor

Table 15 I^2C address

8.4.ABC parameters and control

Variable	Address	Format	Notes
ABC Period	0x40	2 bytes unsigned word. MS byte at lower address	Units: 1 hour ABC is disabled by writing 0 to the ABC period.
ABC enable / disable	Bit 1 in MeterControl in EEPROM[1]	Bit, 1 – disabled 0 – enabled	Disable: read MeterControl byte, Set bit 1 to 1 (OR 0x02) and write to MeterControl Enable: read MeterControl byte, Reset bit 1 to 0 (AND 0xFD) and write to MeterControl

Table 16 (not valid for K45)

Note:

1. The sensor has to be restarted (power cycle off and on) after MeterControl in EEPROM has been changed in order to activate the new configuration.

©2022 Senseair AB. All rights reserved.

8.5. Fractional filter parameters

Sensor's firmware takes average of several signal measurements to provide more accurate and low noise in output carbon dioxide value.

This digital filter is implemented as fractional filter algorithm, where new filtered value is calculated as sum of old filtered value and fraction of difference between old filtered value and measured momentary value of the signal. Fraction is always represented as division by 2^n, where n is named as "Frac" parameter.

In order to get advantages of filtering with high value of "Frac" parameter and at the same time improve response on the fast change of CO_2 concentration, the algorithm of calculation of "Frac" parameter is utilised. It decreases value of "Frac" parameter when signal changes fast and it increases "Frac" parameter when signal doesn't trend to change. This algorithm is named as "Dynamical frac" algorithm.

Both fractional filter and dynamical frac algorithms can be enabled and disabled. In some later versions of firmware it is possible to configure minimum and maximum limits for "Frac" parameters change by dynamical frac algorithm.

Variable	Address	Format	Notes
Fractional algorithm enable / disable	Bit 2 in MeterControl EEPROM[1]	Bit, 1 – disabled 0 – enabled	Disable: read MeterControl byte, Set bit 2 to 1 (OR 0x04) and write to MeterControl Enable: read MeterControl byte, Reset bit 2 to 0 (AND 0xFB) and write to MeterControl
Dynamical frac algorithm enable / disable	Bit 3 in MeterControl EEPROM[1]	Bit, 1 – disabled 0 – enabled	Disable: read MeterControl byte, Set bit 3 to 1 (OR 0x08) and write to MeterControl Enable: read MeterControl byte, Reset bit 2 to 0 (AND 0xFB) and write to MeterControl
DefaultFrac	0x4A in EEPROM[1]	Unsigned	Value in 1..8 range

Table 17 (not valid for K45)

Note:

1. The sensor has to be restarted (power cycle off and on) after MeterControl in EEPROM has been changed in order to activate the new configuration.

©2022 Senseair AB. All rights reserved.

8.6. Signal and calibration parameters

Variable	Address	Format	Notes
Zero	0x38 in EEPROM[1,2]	2 bytes unsigned word. MS byte at lower address	Calibration parameter individually measured for each sensor at calibration at factory. Shouldn't be changed. Read only
ZeroTrim	0x48 in EEPROM[1,2]	2 bytes Signed word. MS byte at lower address	Calibration correction parameter. It is added to Zero calibration parameter during CO2 concentration calculation. May be both positive and negative. This parameter is to be used for sensor calibration in the field.

Table 18 (not valid for K45)

Note:

1. The sensor has to be restarted (power cycle off and on) after MeterControl in EEPROM has been changed in order to activate the new configuration.
2. For K33 sensors with memory map 0x5D: Zero – 0xBE, ZeroTrim – 0x4A

8.7. Commands available through SCR

The Special Command Register (SCR), gives access to commands other than ReadRAM and WriteRAM, these commands are called by writing the appropriate command code to RAM address 0x60 (see memory map)

The user must take care when writing to not overwrite any EEPROM location that could compromise the execution of the sensor firmware such as configuration and calibration data. The EEPROM (virtual EEPROM) accessed by SCR register is a part of sensor Flash and has nothing to do with external EEPROM in K22/K30/K33/K45/K50 sensors.

Command	Code	Function	Notes
ReadEEPROM	0x1	Copies first page of EEPROM contents to RAM so it can be read through a WriteRAM command	
WriteEEPROM	0x2	Copies RAM contents to first page of EEPROM	

Table 19 SCR Commands

©2022 Senseair AB. All rights reserved.

9. Appendix A, Checksum calculation example:

```
typedef unsigned char BYTE;


BYTE CheckSum(BYTE * buf, BYTE count) {
    BYTE sum=0;
    while (count>0) {
      sum += *buf;
      buf++;
      count--;
    }
    return sum;
}
```

Let buf point to the first byte after the “7-bit address+Direction bit” field. Byte counter count should be set to number of bytes sent excluding checksum byte.

©2022 Senseair AB. All rights reserved.

10. Appendix B, I²C bus transaction examples

10.1. Example: Reading of CO2 value from sensor

Note that some models, e.g. Senseair K33 ICB, has a different scale factor on the ppm reading. The reading on these models is divided by 10 (i.e. when ambient CO_2 level is 400 ppm the sensor will transmit the number 40).

To read the current CO2 concentration from the sensor we need to read memory locations 0x08 (hi byte) and 0x09 (low byte).

To do this we need to send a sequence of two I2C frames: first we send an I2C write frame containing the sensor address, command number and how many bytes to read, RAM address to read from, and a checksum. Then we send an I2C read frame to read the status, data and checksum. See chapter 2 for details.

In our case we want to read 2 bytes starting from address 0x08. This will give us data from address 0x08 and 0x09, which contains current CO2 reading. The sensor address is 0x68 (default factory setting, configurable in EEPROM).

So, the first frame should look like:

Start | 0xD0 | 0x22 | 0x00 | 0x08 | 0x2A | Stop

a. 0xD0 is Sensor address and read/write bit. 0x68 shifted one bit to left and R/W bit is 0 (Write).
b. 0x22 is command number 2 (ReadRAM), and 2 bytes to read
c. Checksum 0x2A is calculated as sum of byte 2, 3 and 4.

The next frame will read the actual data:

Start | 0xD1 | <4 bytes read from sensor> | Stop

d. The 1:st byte from the sensor will contain operation status, where bit 0 tells us if the read command was successfully executed.
e. The 2:nd and 3:rd byte will contain CO2 value hi byte and CO2 value low byte.
f. The 4:th byte contains checksum

10.2. Example: Start background and zero calibration with I²C commands

In K30, K33 and K50 meters it is possible to start zero and background calibrations with I2C commands.

Background calibration for K30 meters look like this:

Start | 0xD0 | 0x12 | 0x00 | 0x67 | 0x7C | 0x06| 0xFB | Stop

a. 0xD0 is Sensor address and read/write bit. 0x68 shifted one bit to left and R/W bit is 0 (Write).
b. 0x12 is command number 1 (WriteRAM), and 2 bytes to write
c. 0x7C06 is background calibration command
d. Checksum 0xFB is calculated as sum of byte 2-6.

©2022 Senseair AB. All rights reserved.

Write complete response:

Start | 0xD1 | Command and status | C-sum | Stop

e. 0xD1 is Sensor address and read/write bit. 0x68 shifted one bit to left and R/W bit is 1 (Read).
f. 0x11 means write RAM command – status ok, 0x10 means write RAM command – status not ok
g. Checksum

Zero calibration for K30:

Start | 0xD0 | 0x12 | 0x00 | 0x67 | 0x7C | 0x07| 0xFC | Stop

h. 0xD0 is Sensor address and read/write bit. 0x68 shifted one bit to left and R/W bit is 0 (Write).
i. 0x12 is command number 1 (WriteRAM), and 2 bytes to write
j. 0x7C07 is zero calibration command
k. Checksum 0xFC is calculated as sum of byte 2-6.

Write complete response is identical as in background calibration example (ok, or not ok)

In K50 meters with memory map 8 or lower the zero and background commands are identical to K30 meters. In K50 and K33-ICB meters with memory maps higher than 8 the address to write to is moved (from 0x67, 0x68) to 0x32, 0x33 and in K33BLG/ELG to 0x42, 0x43.

10.3. Example: Reading of SpaceTemp from K33 BLG/ELG and K45

Space temperature can be read from memory locations 0x12 (hi byte) and 0x13 (low byte).

To do this a sequence of two I2C frames has to be sent to the sensor: first send a I2C write frame containing the sensor address, command number and how many bytes to read, RAM address to read from, and a checksum. Then send a I2C read frame to read the status, data and checksum. See chapter 2 for details.

In this case 2 bytes starting from address 0x12 should be read. This will give data from address 0x12 and 0x13, which contains current space temperature. The sensor address is 0x68 (default factory setting, configurable in EEPROM).

So, the first frame should look like:

Start | 0xD0 | 0x22 | 0x00 | 0x12 | 0x34 | Stop

g. 0xD0 is Sensor address and read/write bit. 0x68 shifted one bit to left and R/W bit is 0 (Write).
h. 0x22 is command number 2 (ReadRAM), and 2 bytes to read
i. Checksum 0x34 is calculated as sum of byte 2, 3 and 4.

The next frame will read the actual data:

Start | 0xD1 | <4 bytes read from sensor> | Stop

j. The 1:st byte from the sensor will contain operation status, where bit 0 tells us if the read command was successfully executed.
k. The 2:nd and 3:rd byte will contain space temp value hi byte and space temp value low byte.
l. The 4:th byte contains checksum

©2022 Senseair AB. All rights reserved.

10.4. Example: Reading of RH from K33 BLG/ELG

RH can be read from memory locations 0x14 (hi byte) and 0x15 (low byte).

To do this a sequence of two I2C frames has to be sent to the sensor: first send a I2C write frame containing the sensor address, command number and how many bytes to read, RAM address to read from, and a checksum. Then send a I2C read frame to read the status, data and checksum. See chapter 2 for details.

In this case 2 bytes starting from address 0x14 should be read. This will give data from address 0x14 and 0x15, which contains current space temperature. The sensor address is 0x68 (default factory setting, configurable in EEPROM).

So, the first frame should look like:

Start | 0xD0 | 0x22 | 0x00 | 0x14 | 0x36 | Stop

m. 0xD0 is Sensor address and read/write bit. 0x68 shifted one bit to left and R/W bit is 0 (Write).
n. 0x22 is command number 2 (ReadRAM), and 2 bytes to read
o. Checksum 0x36 is calculated as sum of byte 2, 3 and 4.

The next frame will read the actual data:

Start | 0xD1 | <4 bytes read from sensor> | Stop

p. The 1:st byte from the sensor will contain operation status, where bit 0 tells us if the read command was successfully executed.
q. The 2:nd and 3:rd byte will contain space temp value hi byte and space temp value low byte.
r. The 4:th byte contains checksum

To read mixing ratio instead of RH change RAM start address to 0x16 and increase C-sum to 0x38.

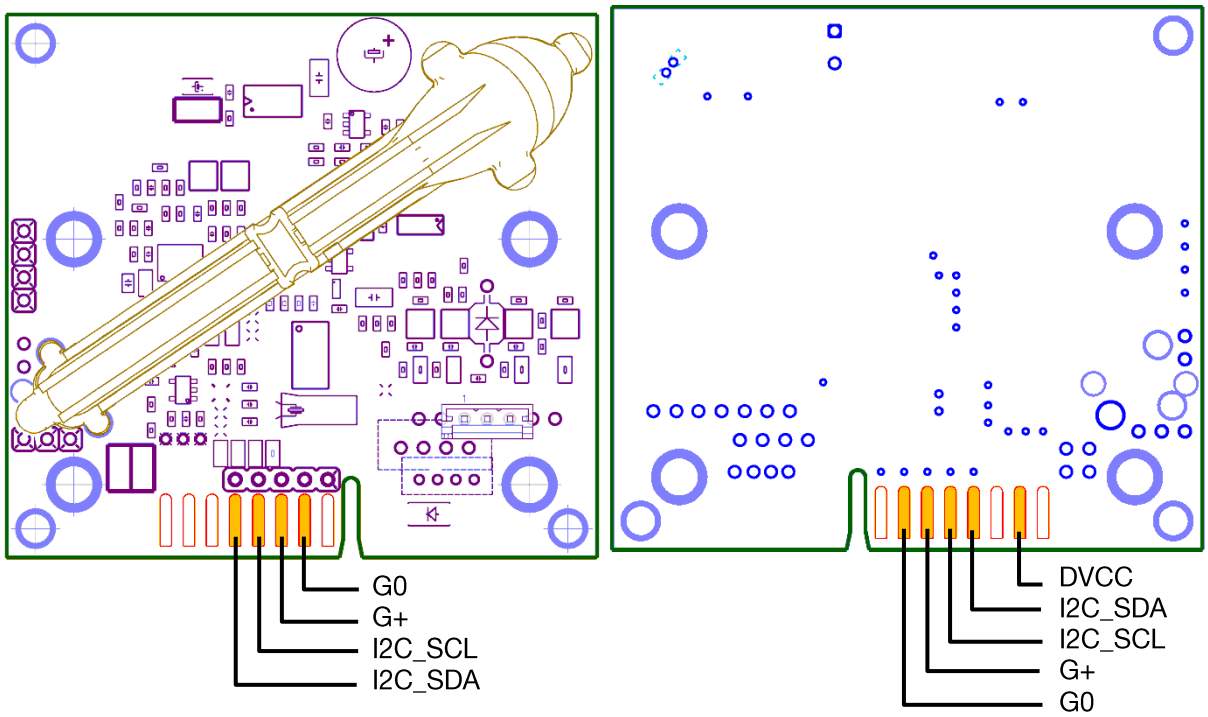
©2022 Senseair AB. All rights reserved.

11. Appendix C, I²C bus terminals on factory connector

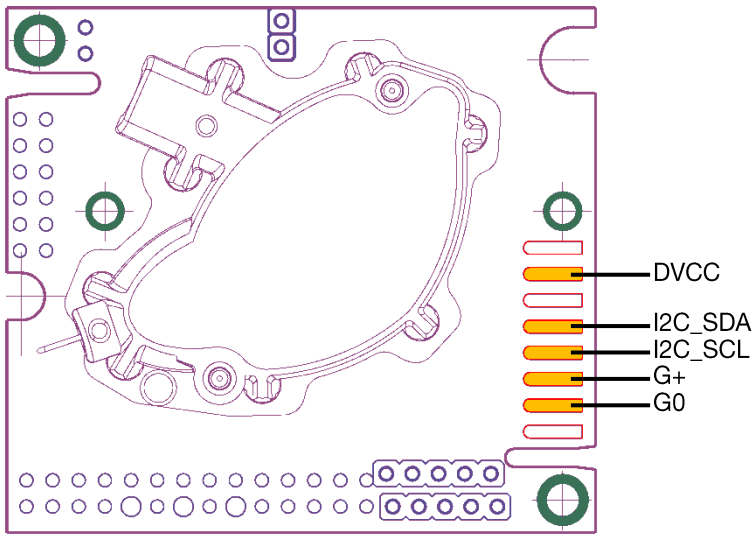
The I²C Bus pins are available at the factory edge connector. Please remember that Senseair is not supposed to keep position or pinout of the factory connector. Figures bellow provides information for test and debug of the system and not for use for system connection.

Please contact Senseair if you are going to use factory edge connector in your system.

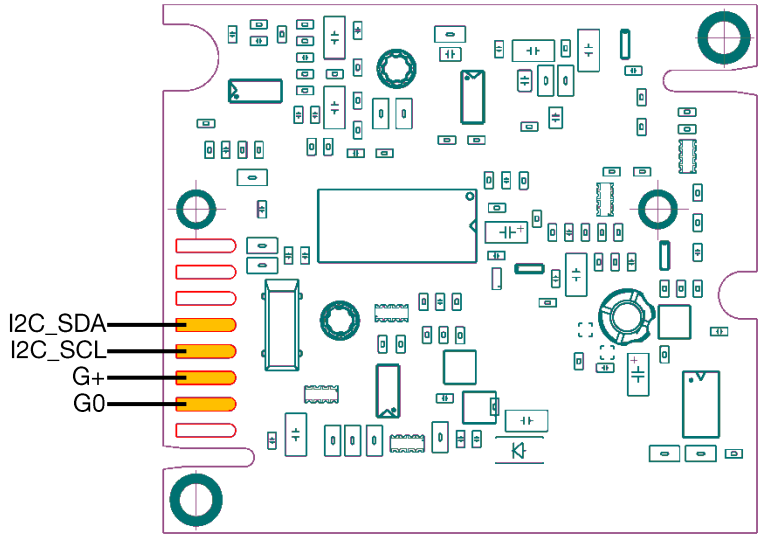
11.1. K22 hardware



11.2. K30 hardware



©2022 Senseair AB. All rights reserved.



www.senseair.com

©2022 Senseair AB. All rights reserved.