SenseAir® LP8 CO_2 sensor module
for integration into battery-powered applications

Sensor specification and integration guideline

IMPORTANT NOTE:

The SenseAir LP8 sensor module is **NOT** a stand-alone or autonomous device.

It's only capable of operating and meeting specification after successful integration with a host controller for handling power management and LP8-specific communication for operability.

Please contact SenseAir for alternatives to manage this electronic and software integration work by yourselves.

Legend and terminology

Terminology	Description
Active measurement period	The fraction of a full measurement period when EN_VBB is active for internal voltage regulation and LP8 sensor has power on VBB and VCAP and is available for or doing measurement and computations.
Full measurement period	The full measurement period including the non-active time in shutdown, when LP8 is powered off but lingering residual heat may be dissipating.
MCU	Microcontroller unit.
ADC	Analogue to Digital-Converter.
UART	Universal asynchronous receiver/transmitter.
RMS noise	Root mean square noise, within 1 standard deviation.

Standard Specifications

Measured gas	Carbon dioxide (CO_2)
Operating principle	Non-dispersive infrared (NDIR)
Operating environment range	**0 to 50°C, 0 to 85%$_{RH}$ (non-condensing)**
Calibrated CO_2 measurement range	0 to 2000 ppm
Extended CO_2 measurement range	2000 to 10’000 ppm
Accuracy CO_2 (calibrated range)	**± 50 ppm ± 3% of reading [1,2,3]**
Typical accuracy CO_2 (extended range)	± 10% of reading [1,2,3,4]
RMS noise CO_2	14 ppm @ 400 ppm 25 ppm @ 1000 ppm
CO_2 warm-up time	30–60 seconds
Measurement repeatability	Max. ± 1% of specified CO_2 concentration, ±10ppm @1000ppm
Accuracy temperature	**± 0.7°C (as measured on chipset)**
Power supply range	**2.9 to 5.5V**
UART and I/O interface voltage	2.5V (see pin descriptions)
Maximal peak-current Typical peak-current	140 mA (typically @ 0°C) 125 mA @ 25°C
Leakage current in shutdown	1 µA [5,6]
Charge per measurement	**3.6 mC (worst-case)**
Energy per measurement	11.9 mJ @ 3.3V
Avg. current w. 16s measurement period	225 µA [5,6]
60s measurement period	61 µA [5,6]
120s measurement period	31 µA [5,6]
Calibrated measurement period	**≥16 s**
Physical dimensions	8 mm x 33mm x 20mm
Estimated life-time expectancy	>15 years
Communication	Proprietary Modbus-extended functions (master-slave UART protocol over serial line)

Note 1: Accuracy is met at 10-40°C, 0-60%RH, after minimum three (3) performed Automatic Baseline Corrections, preferably spanning 8 days in-between, or a successful zero-calibration.
Note 2: Based on reading filtered CO2 measurement data in stable environments and in continuous operation by control mode
Note 3: Accuracy specification is referred to calibration gas mixtures with additional uncertainty of ±1%
Note 4: Extended range accuracy is not calibrated or guaranteed, it is extrapolated from calibrated range
Note 5: Resistor network for measuring VCAP voltage adds 12 µA @ 5.5V
Note 6: External super-capacitor leakage is not considered

Please note that:

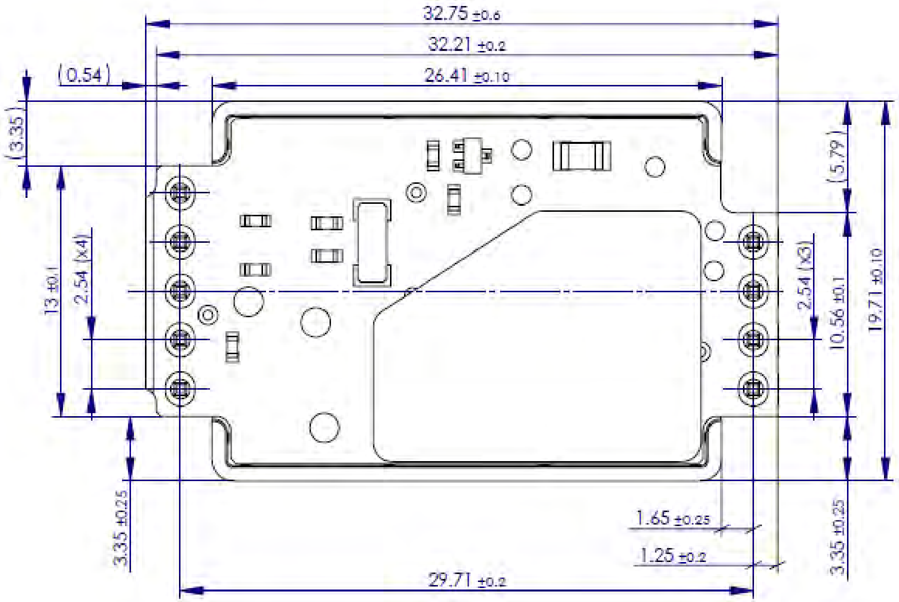
Final specification and accuracy is dependent on integration and operation mode and implementation of features run by the host controller.

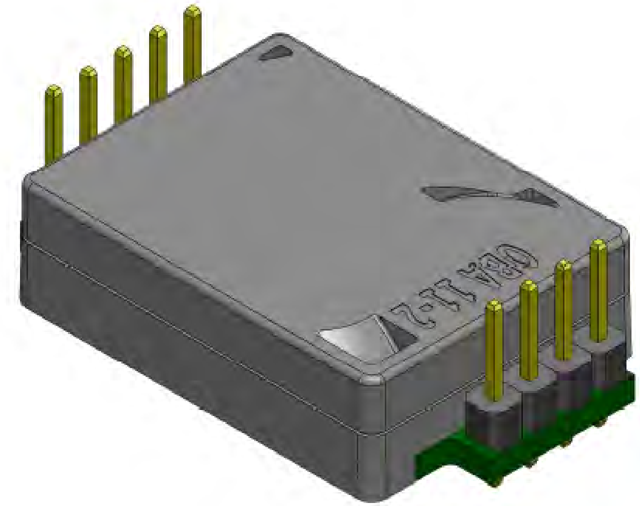
This integration guideline will try to highlight these dependences and details.

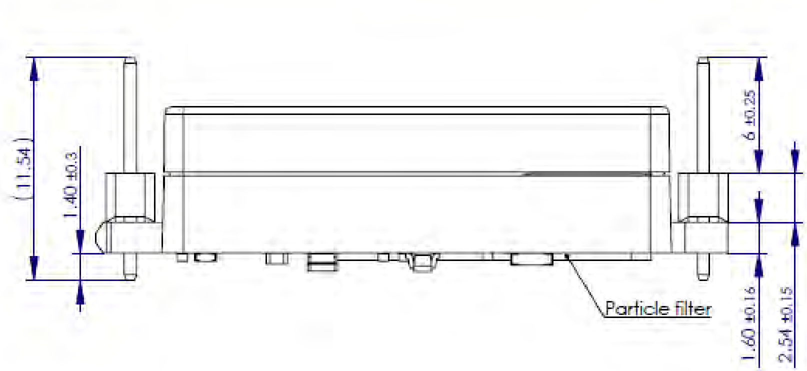
Worst-case charge per measurement:

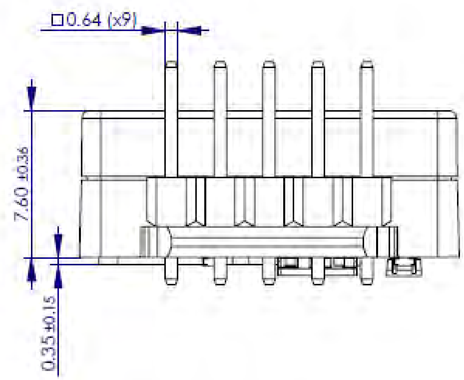
Total	3.6 mC
IR source (lamp)	2.4 mC
Electronics	1.2 mC

Physical dimensions

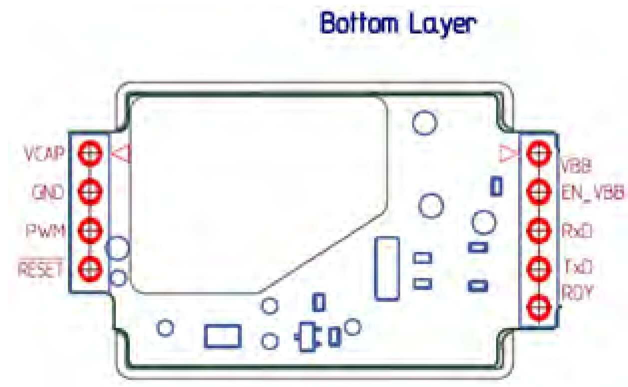








Pin descriptions



Note:
VCAP and EN_VBB may be connected directly with VBB for simplicity, as it is within the LP8 PCB-extended Factory Connector during factory calibration.

The LP8 sensor module is not supplied with this Factory Connector attached, it is cut-away in production system after calibration with following verification test.

Pin #	Pin name	Type	Max. voltage	Other specifications	Description
JP1 (4-pin header)					
1	VCAP	Power	6.5		**Lamp driver supply voltage.** LP8 monitors for low voltage errors using a 500kΩ resistor network connected to the MCU ADC.
2	GND	Power	-		**Ground.**
3	PWM	Output	3.6 [1,2]	$I_{PULL-UP}$ 10 to 80µA	**Unused.** Reserved for PWM functionality in potential future models and compatibility with legacy pin layouts.
4	!RESET	Input	2.5	$R_{PULL-UP}$ 10kΩ	RESET is used only in SenseAir's production system with continuous power supplied. For customer host-integration, **RESET shall be left unconnected or floating**. The host is expected to power cycle LP8 between every measurement. Pull-up resistor is connected to 2.5V
JP2 (5-pin header)					
1	VBB	Power	5.5		**Voltage regulator supply voltage** to LP8 microcontroller unit (MCU) and non-lamp driver electronics.
2	EN_VBB	Input	VBB	Disabled: ≤0.4V Enabled: ≥0.9V	Enable pin to activate the voltage regulator. When pin is in logic low state, and voltage regulator is disabled, LP8 draws maximum 2µA of leakage current through VBB.
3	RxD	Input	3.6	Logic low: ≤0.4V Logic high: ≥2.0V	Receive pin for UART communication to the LP8 sensor module MCU from host.
4	TxD	Output	3.6 [1,2]	$I_{PULL-UP}$ 10 to 80µA	Transmit pin for UART communication from the LP8 sensor module MCU to host.
5	RDY	Output	3.6 [1,2]		RDY signal is used to synchronize sensor states and communication readiness with a host system.

Note 1: Signals are configured as outputs and not allowed to be driven by another push-pull output.
Note 2: Values are referred to the periods when the outputs are set as weak pull-ups.

Electrical specifications

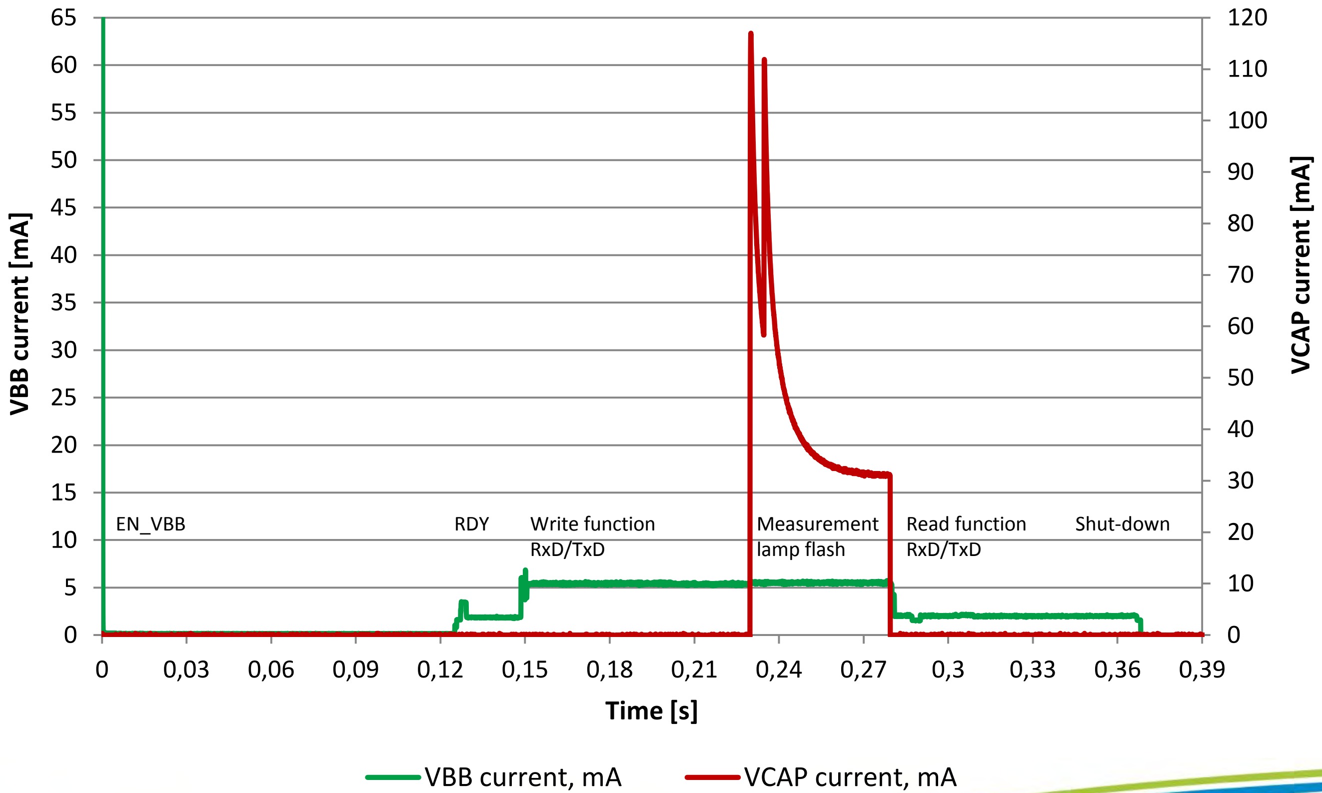
Parameter	Min	Typical	Max	Unit	Test conditions
Power supply voltage:					Lower than 2.8±3% V results in LP8 Error Status
VBB (sensor electronics)	2.9		5.5	V	
VCAP (lamp)	2.9		6.5	V	
Peak-current					When VBB = VCAP = 2.9 to 5.5V
VBB (sensor electronics) [1]		5.4	6	mA	Ambient temperature = 0 to 50°C
VCAP (lamp) [2]		119	129	mA	Ambient temperature = 25°C
VCAP (lamp) [2]			134	mA	Ambient temperature = 0°C [4]
Total (VBB + VCAP) [1,2]		125	140	mA	Ambient temperature = 0 to 50°C
Leakage-current while in shutdown, EN_VBB is off					
VBB (sensor electronics) [3]		1	2	µA	Ambient temperature = 25°C
VCAP (lamp) with 500kΩ resistor network		12	14	µA	Ambient temperature = 25°C, VCAP = 5.5V
VCAP (lamp) without voltage monitoring network [5]		0.1	0.2	µA	Ambient temperature = 25°C, VCAP = 5.5V
Electric charge per active measurement cycle					Ambient Temperature = 0 to 50°C, VBB = VCAP = 2.9 to 5.5V
VBB (sensor electronics)		1.1	1.2	mC	9600 baud rate
VBB (sensor electronics)		1.0	1.1	mC	19200 baud rate [5]
VCAP (lamp)		2.2	2.4	mC	

Note 1: Charging of 20 µF decoupling capacitance is not considered
Note 2: Charging of 220 nF decoupling capacitance is not considered
Note 3: Without pull-down resistor 100k on EN_VBB (as default, this is not mounted on LP8)
Note 4: Peak-current decreases with increasing temperatures
Note 5: Currently not available as purchasable option

Typical current consumption

An optimal active measurement period, the fraction with active voltage regulation and one write/read function each for communication, between application host and LP8 is less than 390 ms, when using a 9600 baud rate.
Typical values of peak-currents are, if the inrush-current spikes required for charging the decoupling capacitors are excluded;
VBB (electronics): 5.4 mA, **VCAP (lamp):** 119 mA, **Total:** 125 mA.

Typical LP8 current consumption;
Sequential measurement, 9600 baud rate, 25°C, 3.7V



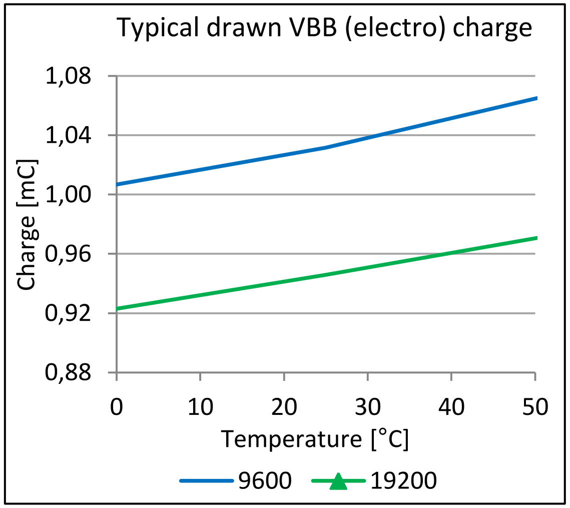
Measured the typically drawn total charge for the optimal communication cycle in +25°C and by 9600 baud rate

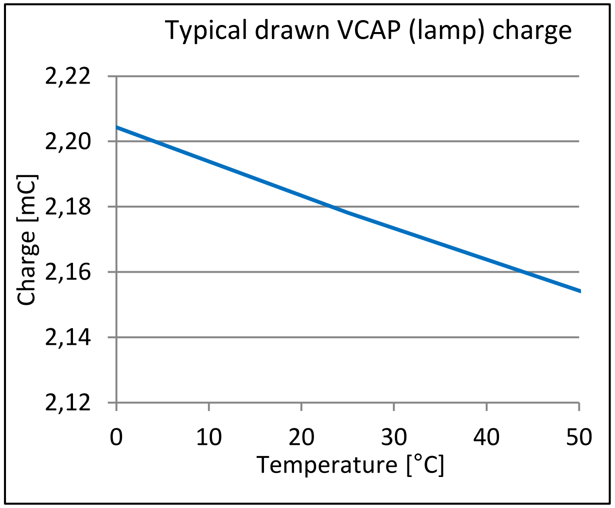
Pin	Charge [mC]
VBB (electronics)	1.03
VCAP (lamp)	2.18
Total	3.21

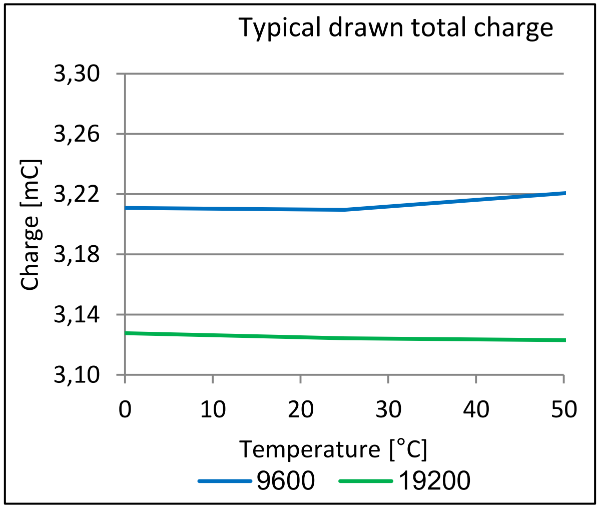
Useful for optimizing partial charging of super-capacitor

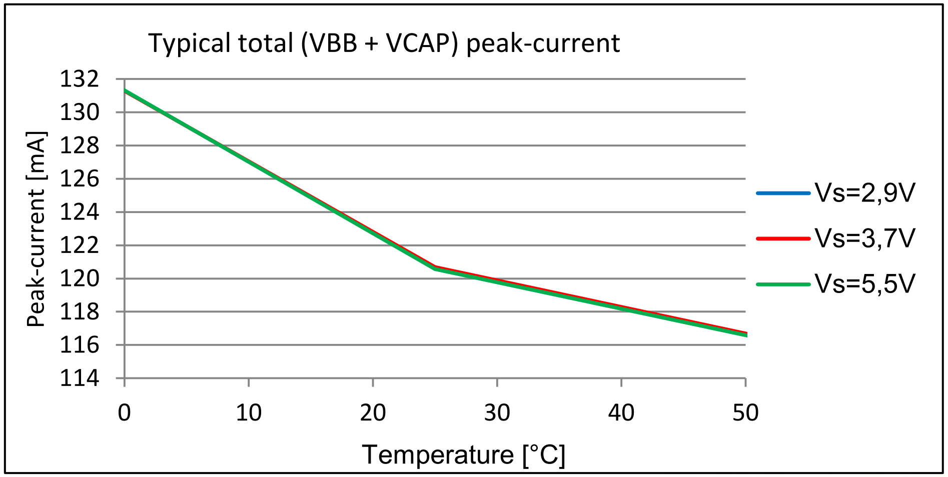
Temperature-dependence in consumption

The VBB and VCAP parameters' typical power consumption per 1 measurement cycle is characterized from 0°C to +50°C.
No significant dependence in the total drawn electric charge and peak-current in the full supply voltage range 2.9V to 5.5V.

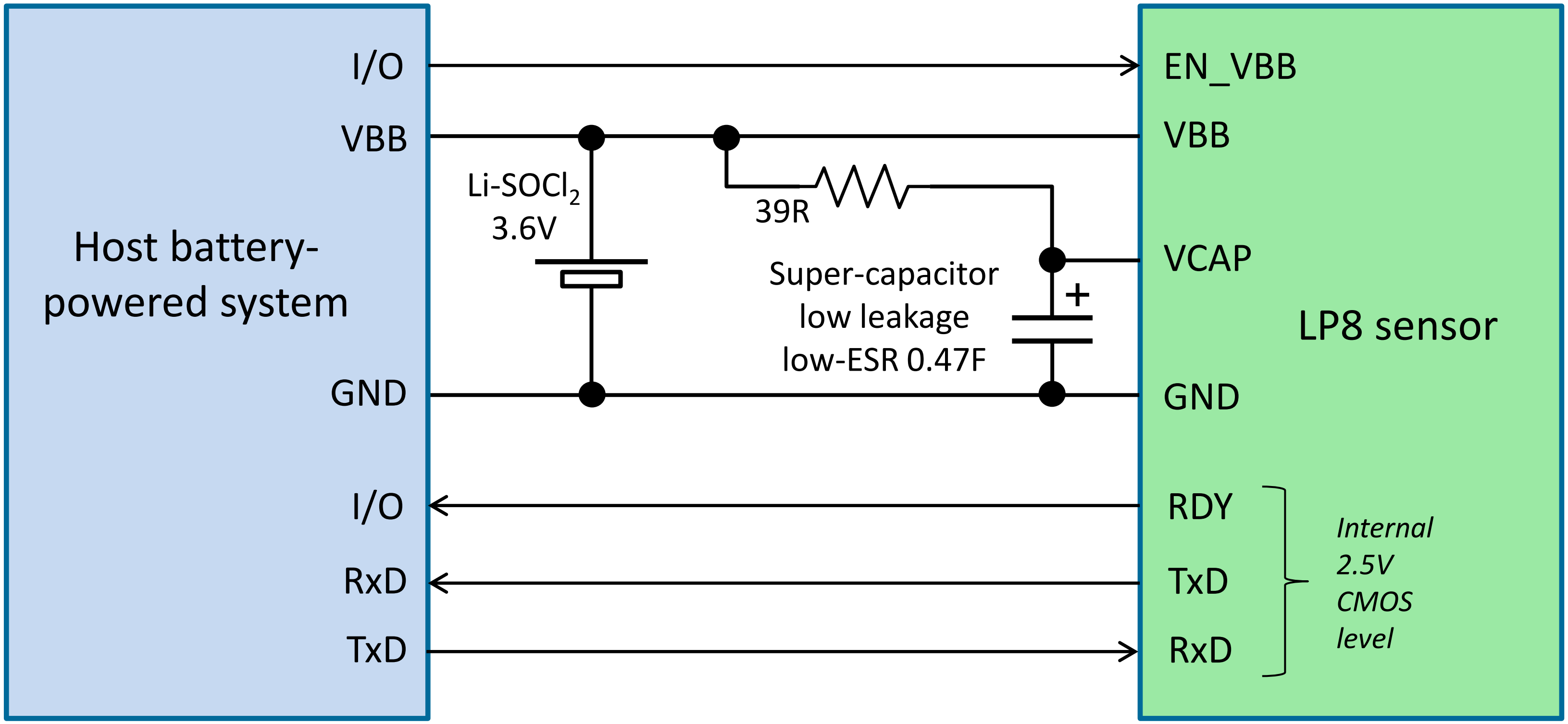








Simple host connection



- In some battery-powered systems, the current limiter can be simplified as a 5Ω resistor.
- Customer can apply low-leakage switches (for example TPS22907) to switch off both VCAP and VBB from leaking during the non-active measurement fraction, when only residual heat is dissipating, per each full measurement cycle.
- Suggested super-capacitor type is Eaton Bussman PM-5R0H474-R (0.47F 5V).
  It is specified as 8µA leakage-current @ 5V, 20^{o}C and 500mΩ ESR.
  VBB can also be supplied from super-capacitor.

Calculating average current consumption

$$I_{avg} = \frac{Q_{MCU} + Q_{lamp}}{T_{MEAS}} + I_{C\_leak} + I_{SHDN}$$

Where;

I_{avg} – LP8's average current consumption

T_{MEAS} – LP8's effective measurement period, as set and controlled by customer's integrated host application

Q_{MCU} – LP8's energy draw on VBB (electronic and microcontroller unit) per active measurement

Q_{lamp} – LP8's energy draw on VCAP (lamp) per active measurement

I_{C_leak} – The leakage-current of selected super-capacitor

I_{SHDN} – sum of leakage-currents by electronics and lamp driver while in sensor shutdown (if customer uses low-leakage switch for VBB and VCAP, this needs to be substituted)

An example:

Measurement period is 30 seconds, ambient temperature is 25°C, application is integrated and configured with a VCAP voltage monitoring resistor network showing 5.5V, super-capacitor leakage-current is stated as 8 µA.

$$I_{avg} = \frac{1030\ [\mu A \cdot s] + 2180[\mu A \cdot s]}{30\ [s]} + 12\ [\mu A] + 8\ [\mu A] = 127\ [\mu A]$$

Average current-consumption can most easily and drastically be reduced by:

- Increasing the host-controlled measurement period, and measure less frequently with less consumption by lamp
- Sync to measurement period and control partial charging of super-capacitor, to limit the available remaining energy that leak after active measurement, by an external low-leakage switch (for example TPS22907) and connect VBB and VCAP to the super-capacitor
- Or apply external low-leakage switches only for VBB and VCAP to lower internal LP8 shutdown leakage in non-active measurement
- Or apply a super-capacitor with lower leakage current

Low-power considerations in integration

- VCAP pin has an internal 500kΩ resistor-divider network connected to the LP8 MCU A/D-converter, it is used for measuring the effective voltage being supplied to the lamp driver. Monitoring that this voltage does not drop below threshold during active lamp pulse is crucial for calibrated CO_2 accuracy and intended function of sensor module. To reduce the impact of the resistor network, implement a low-leakage switch on VCAP during shutdown state to eliminated excess current consumed by the network between active measurements.

- The super-capacitor can successfully be kept only partially charged prior to a measurement. To keep equilibrium on the super-capacitor it should be supplied the same energy charge as is consumed by the LP8 during a single active measurement cycle, defined by worst-case earlier as 3.6 mC.

  For example:

  Battery power supply source holds a fixed 3.3V during discharge

  Desired voltage equilibrium on the super-capacitor to the LP8 is 3.1V (above the 2.9V threshold)

  A 100Ω resistor, under these circumstances, will provide (3.3V-3.1V)/100Ω = 2mA current
  This gives the time required to sufficiently charge the super-capacitor as 3.6mC / 2mA = 1.8 seconds

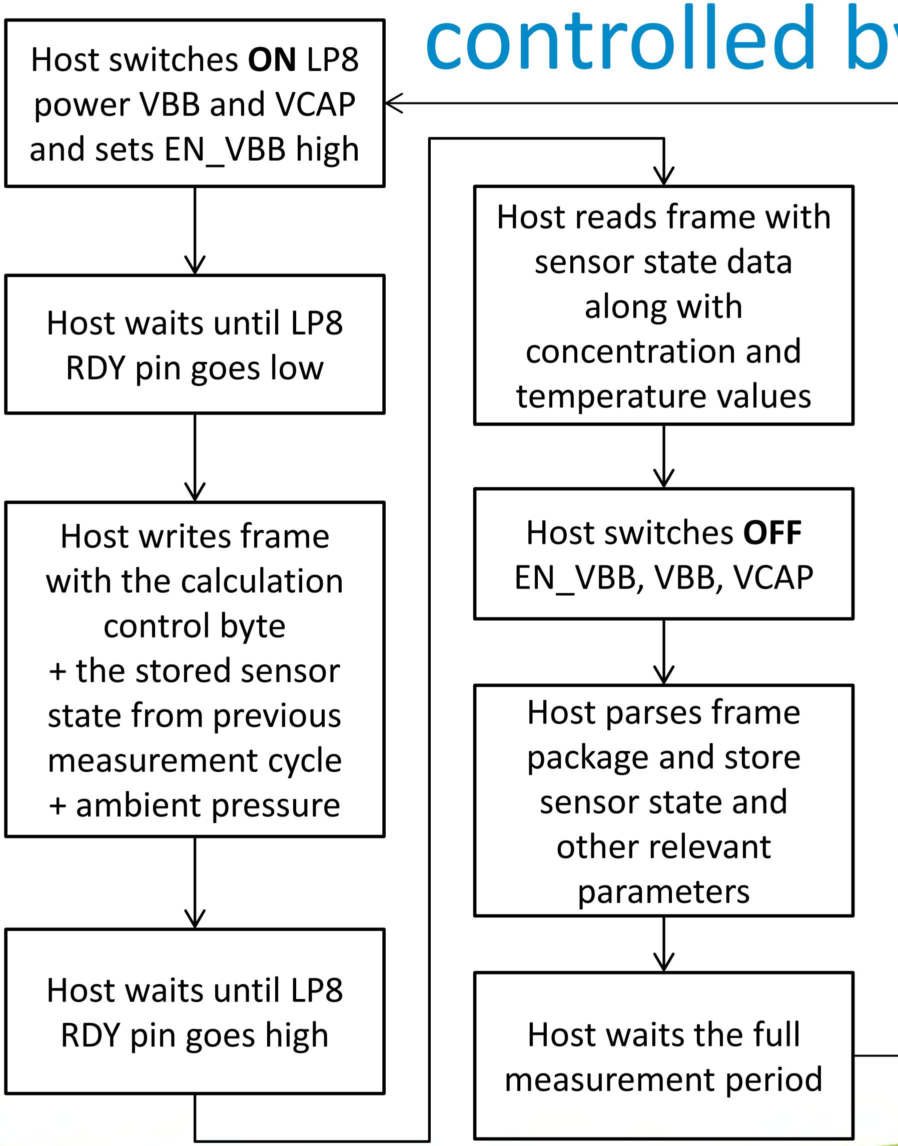
  To eliminate excess leakage-current, the super-capacitor can be decoupled from battery outside of this time

  - A current-source instead of a resistor reduces the time needed to charge the super-capacitor.

- The MCU integrated in application host shall set its I/O pins connected to LP8's TxD, RxD and RDY into high impedance states (Hi-Z) or logical low states when LP8 is in shutdown. The leakage-current on these I/O pins on the LP8 module in power-off state is not specified, or considered, if integrators fail to do this.

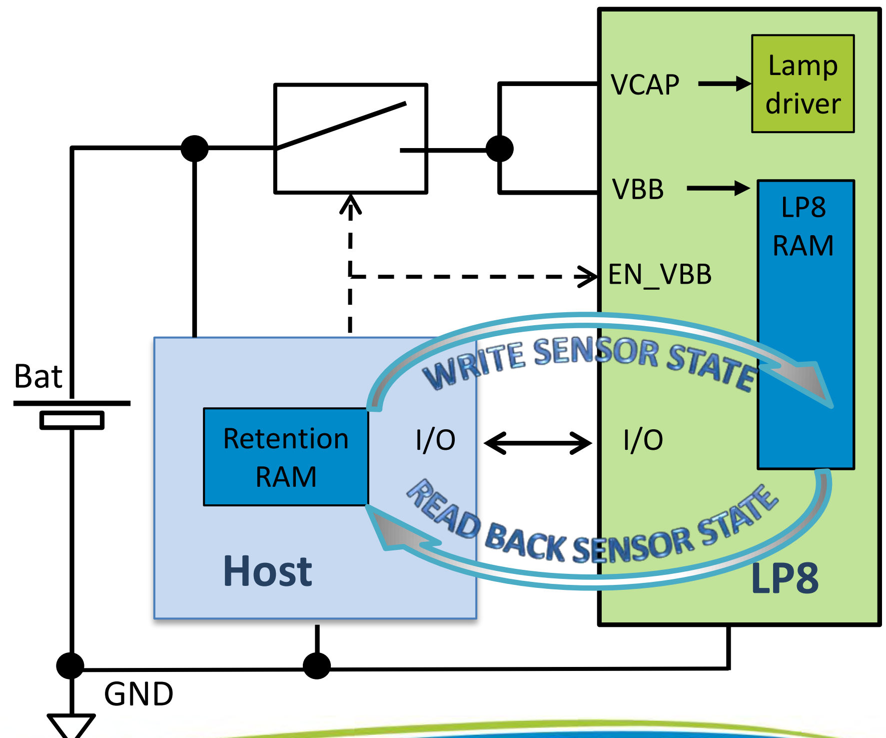
- Using external, host-controlled, switches on VBB and VCAP, with specified sub-mA leakage-current, can help reduce average current consumption further if such a time-controlled switch is not added prior to the super-capacitor.

Continuous LP8 operation, controlled by simple host

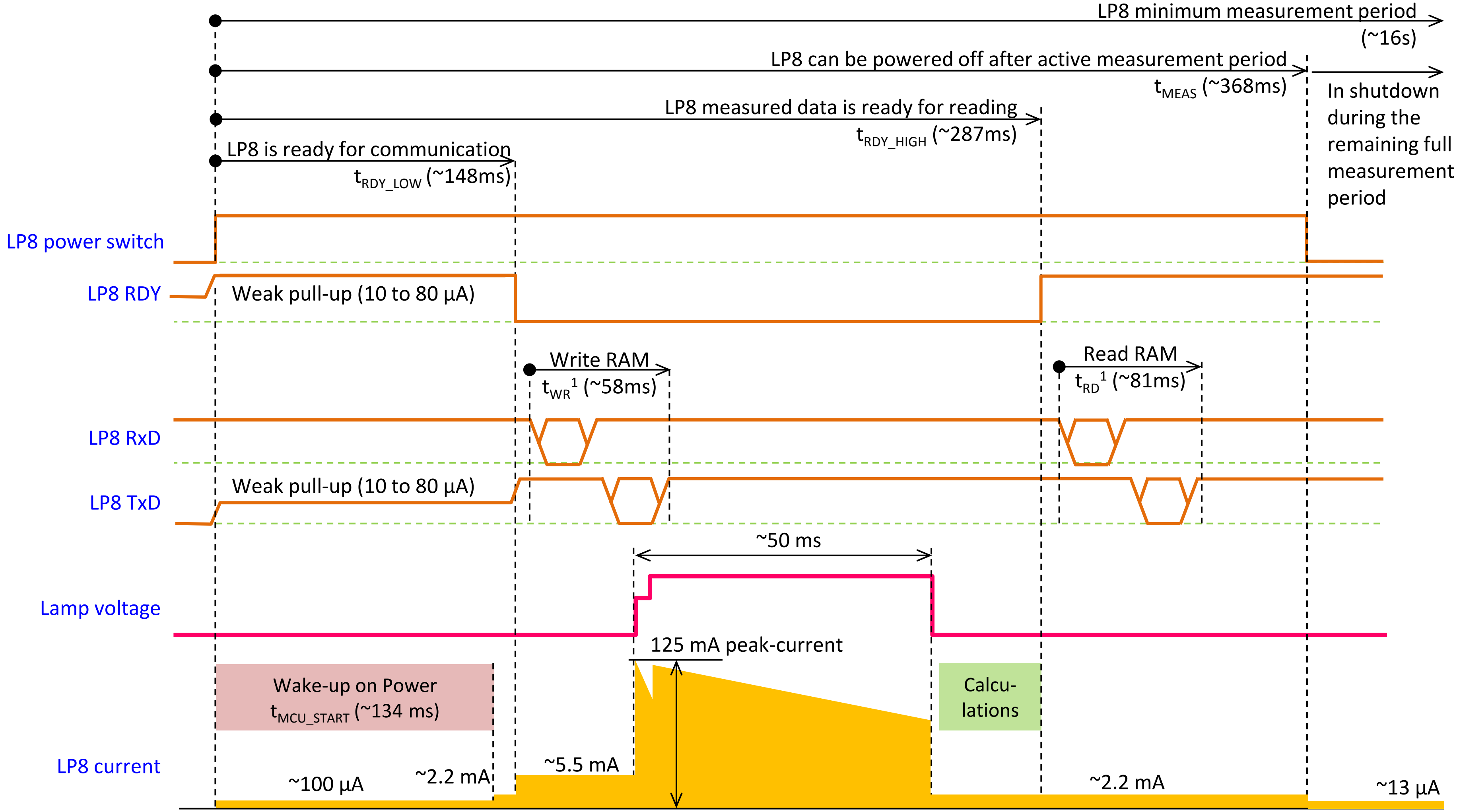


Measurement period of the sensor is determined by customer host system and may vary and change in operation dynamically without degrading measurement accuracy.

Minimum allowed measurement period is 16 seconds (below 16 seconds accuracy is not guaranteed) due to residual heat.



Typical measurement timing diagram
(no calibration)



Note 1: Typical values for 9600 baud rate

Timing parameters

<table>
<tr><th rowspan="2">Parameter
[ms]</th><th colspan="3">FW Rev 1.08 and higher</th><th rowspan="2">Calculation Control command</th><th rowspan="2">Test conditions</th></tr>
<tr><th>Min</th><th>Typical</th><th>Max</th></tr>
<tr><td>t_{MCU_START}</td><td></td><td>134 [2]</td><td></td><td>-</td><td>$T_{ambient}$ = 25°C
VBB = VCAP = 3.7V</td></tr>
<tr><td rowspan="2">t_{RDY_LOW}</td><td>136</td><td>148</td><td>157</td><td>Initial/Sequential measurement</td><td rowspan="4">t_{MCU_START} = 125 to 140ms [2]
$T_{ambient}$ = 25°C
VBB = VCAP = 3.7V</td></tr>
<tr><td>136</td><td>148</td><td>157</td><td>Zero, Background calibrations, ABC</td></tr>
<tr><td rowspan="2">t_{RDY_HIGH}</td><td>221 [1]</td><td>287</td><td>300</td><td>Initial/Sequential measurement</td></tr>
<tr><td>221 [1]</td><td>360</td><td>372</td><td>Zero, Background calibrations, ABC</td></tr>
<tr><td rowspan="2">t_{MEAS}</td><td>294 [1] $+t_{HOST}$</td><td>368 $+t_{HOST}$</td><td>389 $+t_{HOST}$</td><td>Initial/Sequential measurement</td><td rowspan="2">t_{MCU_START} = 125 to 140ms [2]
$T_{ambient}$ = 25°C
VBB = VCAP = 3.7V
LabVIEW host emulation on PC, 9600 baud</td></tr>
<tr><td>294 [1] $+t_{HOST}$</td><td>441 $+t_{HOST}$</td><td>461 $+t_{HOST}$</td><td>Zero, Background calibrations, ABC</td></tr>
<tr><td>t_{WR}</td><td>52</td><td>58</td><td>64</td><td>-</td><td>Host writes 26 bytes;
LabVIEW host emulation on PC, 9600 baud</td></tr>
<tr><td>t_{RD}</td><td>73</td><td>81</td><td>89</td><td>-</td><td>Host reads 44 bytes;
LabVIEW host emulation on PC, 9600 baud</td></tr>
</table>

Note 1: Minimum value assume that an error occurred where FW skips measurement lamp pulse
Note 2: Typical value is specified by the MCU producer

Host polling of RDY pin

Always polling on RDY pin

Consider the weak pull-up (10 to 80µA) on RDY pin need to satisfy logic high level for host before RDY can switch to go low

1. Turn on VBB
2. Initial delay: 4 ms [1]
3. Wait until RDY switches to logic low
   LP8 error if RDY doesn't switch within 210 ms
4. UART "Write RAM"
5. Wait until RDY switches to logic high
   LP8 error if RDY doesn't switch within 250 ms
6. UART "Read RAM"
7. Turn off VBB

Note 1: Initial delay is needed when turning on VBB regulator, for external switches and establishing pull-up on RDY. If a slow external switch is used then it will be necessary to increase this delay.

Polling only for RDY pin to switch into logic high level

The initial weak pull-up state on RDY is desired to be omitted from polling

1. Turn on VBB
2. Delay for switch and timeout error: 210 ms
3. UART "Write RAM"
4. Wait until RDY switches into logic high
   LP8 error if RDY doesn't switch within 250 ms
5. UART "Read RAM"
6. Turn off VBB

No polling on RDY pin, worst-case delay in timer

RDY pin is not used by host and skipped integration can save the host an I/O pin

1. Turn on VBB
2. Delay for switch and timeout error: 210 ms
3. UART "Write RAM"
4. Delay for switch and timeout error: 250 ms
5. UART "Read RAM"
6. Turn off VBB

FW Rev.	Typical active measurement period cycle time T_{MEAS} (between power ON and OFF) for the Sequential Measurement function command, [ms]		
	Polling on RDY pin	Polling only for RDY pin to go high	No polling on RDY pin
≥1.08	368	368	630

Power-efficient communication cycles

In order to make optimal and power-efficient communication with the LP8 sensor module from the application host; it is required that the host unit only sends 1 write function command followed by 1 read function per each active measurement period in accordance with the LP8 RDY pin changing to low/high. These single write and read function commands must then incorporate and sequence all the information that needs to be communicated within their respective data package frames.

The application host should immediately, after it has received the successful read function PDU response, shutdown the power to the LP8 sensor module.

The application host unit should then activate any and all external low-leakage switches, to limit the leakage current while LP8 is in shutdown, and utilize internal power-saving features to refrain from busy-waiting, while potential residual heat in LP8 is dissipating or waiting for set sampling period to pass.

The application host unit can skip implementing an I/O pin to actively listen to the LP8 RDY signal change, for less effectively synchronization of communication readiness, before sending the Write and subsequent Read function commands; This sync functionality is then to be enabled by a host-internal timer and set with worst-case delay timings. The circuit is then closed and leaking power excessively before reaching the cyclic shutdown and off state.

These worst-case hard timing delays, if implemented, can be derived from the earlier timing diagram on pages 14-15

Modbus serial line communication

Default SenseAir MODBUS UART settings:

- Device address – 0x68 or 0xFE
- Baud rate – 9600
- Parity bit – No parity bit
- Stop bits – 2 bits

MODBUS ADU (Application Data Unit) package frame (X + 4 bytes)			
Device address (1 byte)	Function Code (1 byte)	Function Data (X bytes)	CRC (2 bytes, Low byte first)
	MODBUS PDU package frame (X + 1 byte)		

Function Code 65 (0x41)
Write to LP8 RAM

Request PDU [1]

Function code	1 byte	0x41
Starting address, high byte [2]	1 byte	0x00
Starting address, low byte	1 byte	Starting-pointer in LP8 RAM
Number of bytes to write	1 byte	Y (in hexadecimal value)
Data to write to LP8	Y bytes	

Response PDU [1]

Responded function code	1 byte	0x41

Error Response PDU [1]

Responded function code	1 byte	0xC1
Error code	1 byte	Specific error flag bits are set

Note 1: The full MODBUS ADU package frame is not written out.
Note 2: The high byte in RAM memory address (the page) will always be 0x00 in the current RAM memory map version.

Function Code 68 (0x44)
Read from LP8 RAM

Request PDU [1]

Function code	1 byte	0x44
Starting address, high byte [2]	1 byte	0x00
Starting address, low byte	1 byte	Starting-pointer in LP8 RAM
Number of bytes to read	1 byte	Z (in hexadecimal value)

Response PDU [1]

Responded function code	1 byte	0x44
Number of bytes read	1 byte	Z (in hexadecimal value)
Data read from LP8	Z bytes	

Error Response PDU [1]

Responded function code	1 byte	0xC4
Error code	1 byte	Specific error flag bits are set

LP8 RAM memory map

LP8 RAM memory address space and parameters dedicated to the communication with application host:

Page 00	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8x	Calculation control	**LP8's Sensor State from previous cycle has to be included in the write function data and written back to LP8 from host in each new measurement cycle, and then read out again to host post-measurement with its updated values.**													Reserved	
9x	**Sensor State byte space continuation.**								Host Pressure (S16, 0.1 hPa)		CO2_Conc (S16, ppm)		CO2_Conc_Pres. (S16, ppm)		Space_Temp (S16, 0.01°C)	
Ax	VCAP1 (S16, mV)		VCAP2 (S16, mV)		Error Status3	Error Status2	Error Status1	Error Status0	Filtered_CO2 (S16, ppm)		Filt_CO2_Pres. (S16, ppm)		*Unused byte space*			

Optimally-sized function data package frames regarding most accurate and time-efficient operation

Write function data to send from application host to the LP8 to make a measurement

Write these 26 bytes (number of bytes 0x1A) to the LP8 RAM memory addresses **0x0080** to **0x0099**

Calculation control	Sensor State	Host Pressure (S16, 0.1 hPa)
0x80	0x81 – 0x98	0x99

Read function data to read back to the application host from LP8 after a new measurement is done

Read these 44 bytes (number of bytes 0x2C) from the LP8 RAM memory addresses **0x0080** to **0x00AB**

Calc. control	Sensor State	Host Pres.	Measurement data, voltage diagnostics and Error Status
0x80	0x81 – 0x98		0x9A – 0xAB

Parameter list and descriptions

Parameter	Byte Length	Starting Address	Format	Unit	Description
Calculation Control	1	0x80	Bit structure	N/A	Writing this parameter selects the operation mode and measurement function for this measurement period for the LP8 sensor module.
Sensor State	23	0x81	Structure	N/A	Proprietary structure for LP8 internal data; such as input to the noise-suppression filter algorithm and calibration references. This has to be passed and saved in the host retention memory during shutdown for use in the next LP8 measurement.
Host Pressure	2	0x98	S16	10 Pa = 0.1 hPa	If application host is measuring ambient pressure, writing this current value to LP8 will update internal pressure correction algorithms and allow reading out pressure corrected measurement values. If ambient pressure is not measured by host, then this parameter can be skipped as LP8 will assume normal sea level pressure of 10124 (1012.4 hPa).
CO2_Conc	2	0x9A	S16	ppm	CO2 concentration without pressure-correction or any noise-suppression filtering.
CO2_Conc_Pres	2	0x9C	S16	ppm	Pressure-corrected CO2 concentration value without noise-suppression filtering.
Filtered_CO2	2	0xA8	S16	ppm	Noise-filtered CO2 concentration without pressure-correction.
Filt_CO2_Pres.	2	0xAA	S16	ppm	Pressure-corrected filtered concentration value.
Space_Temp	2	0x9E	S16	0.01 °C	The measured LP8 sensor temperature from NTC.
VCAP1	2	0xA0	U16	mV	VCAP voltage measured by sensor prior lamp pulse, indicator of battery voltage
VCAP2	2	0xA2	U16	mV	VCAP voltage measured by sensor during lamp pulse, to ensure sufficient energy
Error Status	4	0xA4	Bit structure	N/A	Error bit structure

U16 – unsigned integer, 16 bits
S16 – signed integer, 16 bits

Calculation Control: Measurements

Calculation Control Byte	Operating mode for this measurement period	Description
0x10	Initial measurement	Used only when a previous sensor state is unavailable; e.g. after a fresh boot-up or after previous recalibrations where the noise-suppression filters have been reset.
0x20	Sequential measurement	The default operating mode, reusing previous sensor states for noise suppression and passing background calibration references.

It's possible for limited-operation and troubleshooting to continuously write 0x10 to the Calculation Control Byte in every new measurement period. But then the sensor, by only running "Initial Measurement" operation, will be less accurate as it will always overwrite filters with "new" initial measurements and never effectively use noise-suppression filtering from the previous sensor states. See page 35 for code example.

Calculation Control: Zero-calibrations

Calculation Control Byte	Operating mode for this measurement period	Description
0x40	Zero-calibration: with noisy non-filtered CO_2	Requires a stable ambient environment free from any CO_2, i.e. concentration is 0 ppm. Makes a new measurement and sets a new baseline offset from the noisy and non-filtered internal signal reference.
0x41	Zero-calibration: with noise-filtered CO_2	Requires a stable ambient environment free from any CO_2, i.e. concentration is 0 ppm. Makes a new measurement and sets a new baseline offset from the noise-filtered internal signal reference from the new and old measurement.
0x42	Zero-calibration: with noisy non-filtered CO_2 + reset noise-suppression filters	Requires a stable ambient environment free from any CO_2, i.e. concentration is 0 ppm. Makes a new measurement and sets a new baseline offset from the noisy and non-filtered internal signal reference. It then overwrites the internal historic noise parameter references set after this measurement to be the same as the current measured raw value.
0x43	Zero-calibration: with noise-filtered CO_2 + reset noise-suppression filters	Requires a stable ambient environment free from any CO_2, i.e. concentration is 0 ppm. Makes a new measurement and sets a new baseline offset from the noise-filtered internal signal reference. It then overwrites the internal historic noise parameter references set after this measurement to be the same as the current measured raw value.

Zero-calibrations are the most accurate recalibration routine, and are not at all affected performance-wise by having an available pressure sensor on host for accurate pressure-compensated references.

Not resetting the noise-suppression filters after a zero-calibration may cause the subsequent next few measurements, when again LP8 sensor module is exposed to normal application-intended air mixtures, to be extra inaccurate. It is applying the signal references for the noise-filtering from the sensor state data taken from the previous zero-point environment. This additional inaccuracy is temporal and would eventually diminish and properly readjust by itself with enough new samples. But if speed of measurement accuracy is of importance, then resetting the noise-suppression filters is a faster process.

A zero ppm environment is most easily created by flushing the optical cell of the LP8 sensor module and filling up an encapsulating enclosure with nitrogen gas, N_2, displacing all previous air volume concentrations. Another less reliable or accurate zero reference point can be created by scrubbing an airflow using e.g. Soda lime.

Calculation Control: Background-calibrations

Calculation Control Byte	Operating mode for this measurement period	Description
0x50	Background-calibration: with noisy non-filtered CO_2	Requires a stable ambient environment in fresh air, i.e. concentration is near-400 ppm. LP8 makes a new measurement and sets a new baseline offset by the difference between the noisy and non-filtered (but pressure-compensated) internal signal reference and the predefined fresh air reference.
0x51	Background-calibration: with noise-filtered CO_2	Requires a stable ambient environment in fresh air, i.e. concentration is near-400 ppm. LP8 makes a new measurement and sets a new baseline offset from the difference between the noise-filtered (but pressure-compensated) internal signal reference and the predefined fresh air reference.
0x52	Background-calibration: with noisy non-filtered CO_2 + reset noise-suppression filters	Requires a stable ambient environment in fresh air, i.e. concentration is near-400 ppm. LP8 makes a new measurement and sets a new baseline offset from the difference between the noisy and non-filtered (but pressure-compensated) internal signal reference and the predefined fresh air reference. It then overwrites the internal historic noise parameter references set after this measurement to be the same as the current measured raw value.
0x53	Background-calibration: with noise-filtered CO_2 + reset noise-suppression filters	Requires a stable ambient environment in fresh air, i.e. concentration is near-400 ppm. LP8 makes a new measurement and sets a new baseline offset by the difference between the noise-filtered (pressure-compensated) internal signal reference and the predefined fresh air reference. It then overwrites the internal historic noise parameter references set after this measurement to be the same as the current measured raw value.

A "fresh air" background environment is assumed to be 400 ppm by normal ambient atmospheric pressure by sea level. It can be referenced in a crude way by placing the sensor in direct proximity to outdoor air, free of combustion sources and human presence, preferably during either by open window or fresh air inlets or similar. Calibration gas by exactly 400ppm can be purchased and used, but by the same means purchasing nitrogen gas for zero-calibration will generally be cheaper and produce better results.

Calculation Control: ABC

Calculation Control Byte	Operating mode for this measurement period	Description
0x70	Automatic Baseline Correction, ABC: w. noise-filtered CO2	Informs the LP8 to perform an ABC calibration based on the stored baseline reference continuously passed back and forth within the sensor states. A new offset is set based from the stored reference value in Sensor State and its assumed correlation to 400ppm.
0x72	Automatic Baseline Correction, ABC: w. noise-filtered CO2 + reset noise-suppression filters	Informs the LP8 to perform an ABC calibration based on the stored baseline reference continuously passed back and forth within the sensor states. A new offset is set based from the stored reference value in Sensor State and its assumed correlation to 400ppm. It then overwrites the internal historic noise parameter reference set after this measurement to be the same as the current measured raw value.

The Automatic Baseline Correction algorithm is a proprietary SenseAir method for referencing to “fresh air” as the lowest, but required stable, CO2-equivalent internal signal the sensor has measured during a set time period. This time period is decided by the application host, as the host decides when to write this Calculation Control value into the LP8 sensor module, but it is recommended to be something like an 8 day period as to catch low-occupancy and other lower-emission time periods and favourable outdoor wind-directions and similar which can plausibly and routinely expose the sensor to the most true fresh air environment.

If such an environment can never be expected to occur, either by sensor locality or ever-presence of CO2 emission sources, or exposure to even lower concentrations than the natural fresh air baseline, then ABC recalibration can’t be used.

In each new measurement period, the sensor will compare its internal signal references stored in sensor state to the new ones made in the current measurement, and if new values show a lower CO2-equivalent raw signal while also in a stable environment, the reference is updated with these new values.

The ABC algorithm also has a limit on how much it is allowed to change the baseline correction offset with, per each ABC cycle, meaning that self-calibrating to adjust to bigger drifts or signal changes may take more than one ABC cycle.

The LP8 noise-suppression filter explained

The noise-suppression filter algorithm is a digital and dynamic IIR-filter, comparing the raw signal reference from the previous sensor state with the signal obtained in the new measurement sequence. It then sets a fractional allowance for the signal step change to pass through into final presented CO2-calculations.

- If the signal change is “big”, implying there is an actual change in the ambient environment, the noise-suppression filter will still limit ½ (50%) of this change to make up the presented CO2-concentration value.
- If the signal step change is “small”, implying only natural noise causing deviation in an otherwise stable ambient environment, it will supress all but ⅙ (16.67%) of this change to make up the presented CO2-concentration value.

The historic nature and dynamic selection, depending on the amplitude of change, of the fractional noise-suppression in this IIR-filter make a trade-off between fast response time when needed and better accuracy in stable conditions.

However, the lagging response time by the noise-suppression filter and the required passing of the sensor state feedback (used for the historic part of the IIR filter) may not fit all applications, and the application host need to be programmed for when such mode of operation is useful and hence which CO2-concentration parameter from the LP8 RAM memory to use, especially for recalibrations.

- If the measurement period of LP8 is set very long, and the normal ambient environment is naturally very prone to changes over such set period, then using the noise-suppression filter may give rise to more inaccurate CO2-concentrations than just living with the full impact of raw signal noise by the sensor.

One option to speed up and improve accuracy in recalibrations, and decrease the lagging effect from the noise-suppression filters on response time and accuracy, is to temporally make the measurement period the fastest that it can be, minimally 16s, during these events. Hence the previous sensor state reference is more likely to still be a valid representation of also the current environment and noise-suppression can be set higher for more accurate signal reference as baseline correction.

LP8 sensor recalibration explained

The LP8 sensor module works as a passive slave device and totally rely on host commands applied through the "Calculation Control" byte for its functionality. This includes timing and exact function for when and how to apply CO_2 recalibration for continuous accurate measurements. The differences between the four types of calibration used in LP8 are:

1) **Zero calibration:** function which forcefully assumes the current environment is 0 ppm (for instance by Nitrogen gas)
2) **Background calibration:** function which forcefully assumes the current environment to be 400 ppm fresh air environment
   a) For all above, using unfiltered concentration – LP8 uses current unfiltered measurement value to set recalibration offset
   b) For all above, using filtered concentration – LP8 uses calculated noise-suppressed value to set recalibration offset (sensor should have been exposed to this stable fresh air environment for more than 40 measurement periods)
3) **ABC (Automatic Baseline Correction):** uses an internal stored signal reference (together with accompanying parameters) of the lowest concentration value, treated as 400 ppm, as a recalibration target. This is collected and compared continuously from the last "Initial state", "ABC", "Background- or Zero calibration" commands written into the "Calculation Control" byte.

Starting Background calibration requires either a representative "fresh air" background environment or by exposure to a calibration gas mixture of 400 ppm CO_2 in Nitrogen. A crude "fresh air" environment can be achieved by placing the sensor in direct proximity to outdoor air, free of combustion sources and human presence, like an open window or fresh air inlets or similar.
Starting Zero calibration requires placing the sensor in an encapsulated enclosure, e.g. an ESD-safe plastic bag, and flushing it with nitrogen gas. It is the most accurate recalibration routine, and is not affected in performance by having an available pressure sensor on host for accurate pressure-compensated references.

LP8 filtered and unfiltered trade-offs

Concentration parameters	
Unfiltered parameters: CO2_Conc, CO2_Conc_Pres	Natural signal noise in readings, but response time is immediate and only limited by gas diffusion into the optical cell.
Filtered parameters: Filtered_CO2, Filt_CO2_Pres	Only small noise in readings, slower resulting response time

Important: Noise-suppression filtering is performed on raw signals, and the filtered and unfiltered parameters are always calculated from the raw signals independently and in parallel.

Zero-/Background and External reference calibration *	
Using unfiltered recalibration option	Worse accuracy for recalibration, but a representative calibration environment is allowed and needed to persist for only one single measurement period
Using filtered recalibration option	Good accuracy for recalibration offset, but a representative calibration environment must persist for more than 40 measurement periods to ensure a proper filtered reading

* ABC function commands only ever use filtered parameter as baseline , because only a low reading which is also taken in a stable environment is accepted to replace the current fresh air baseline reference.

Sensor program algorithm flow of measurement cycle:

Measuring and acquiring raw signals → Filtering [or resets filter memory if "Initial Measurement" or an Error type occurred] → [Perform recalibration offset] → [Reset filters for the commands with codes 32,33,42,43,52,53,73] → Calculating Concentrations

Reset filters means that LP8 updates its raw-signal filter memory with the measured value from the actual cycle

LP8 resulting RMS noise in measurements

Calculation Control, 0x80	Description of Calculation Control flow command	Concentration read in the measurement	
		CO2_Conc_Pres	Filt_CO2_Pres
0x10	Initial measurement	Measured value	Measured value
0x20	Sequential measurement	Measured value	Measured value + Δnoise
0x40	Zero calibration using unfiltered data	0 ppm	0 ppm + Δnoise
0x41	Zero calibration using filtered data	0 ppm + Δnoise	0 ppm
0x42	Zero calibration using unfiltered data + reset filters	0 ppm	
0x43	Zero calibration using filtered data + reset filters	0 ppm + Δnoise	
0x50	Background calibration using unfiltered data	400 ppm	400 ppm + Δnoise
0x51	Background calibration using filtered data	400 ppm + Δnoise	400 ppm
0x52	Background calibration using unfiltered data + reset filters	400 ppm	
0x53	Background calibration using filtered data + reset filters	400 ppm + Δnoise	

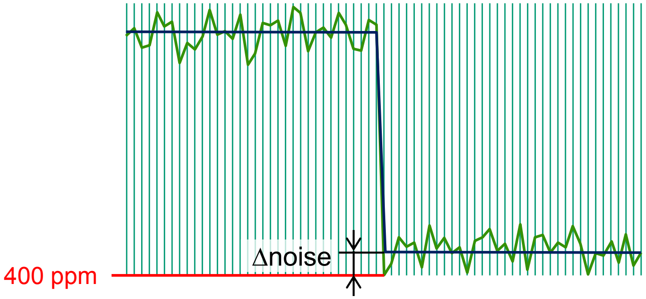
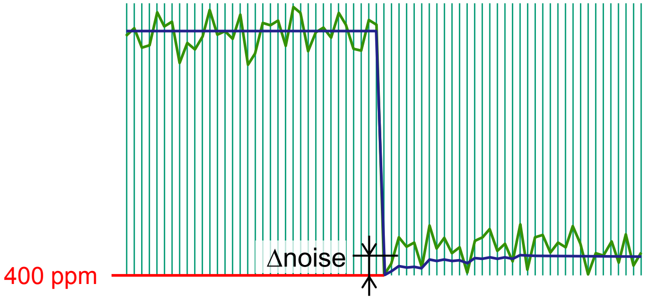
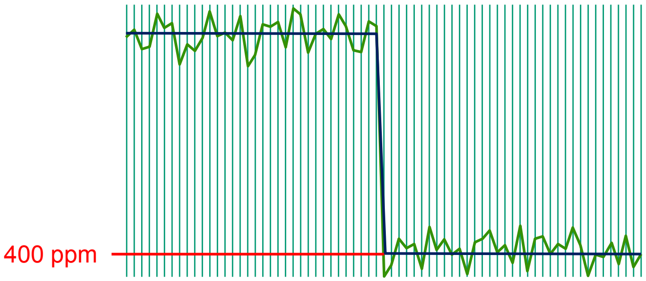
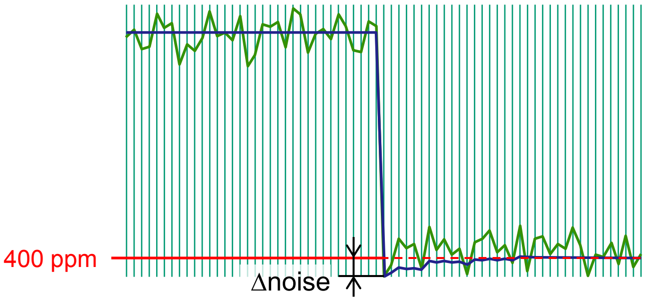
Δnoise – signed difference between unfiltered concentration and filtered one. Refer the RMS noise specifications on page 4 of the document where values are given for two ambient concentrations.

The above table is valid only if no error occurs and calibrations are performed under valid conditions, i.e. residual Nitrogen has been purged if a zero calibration was performed, concentration is close to 400 ppm if background calibration was performed.

LP8 resetting filters explained

Filtered and unfiltered concentrations on example of using Background Calibration function commands

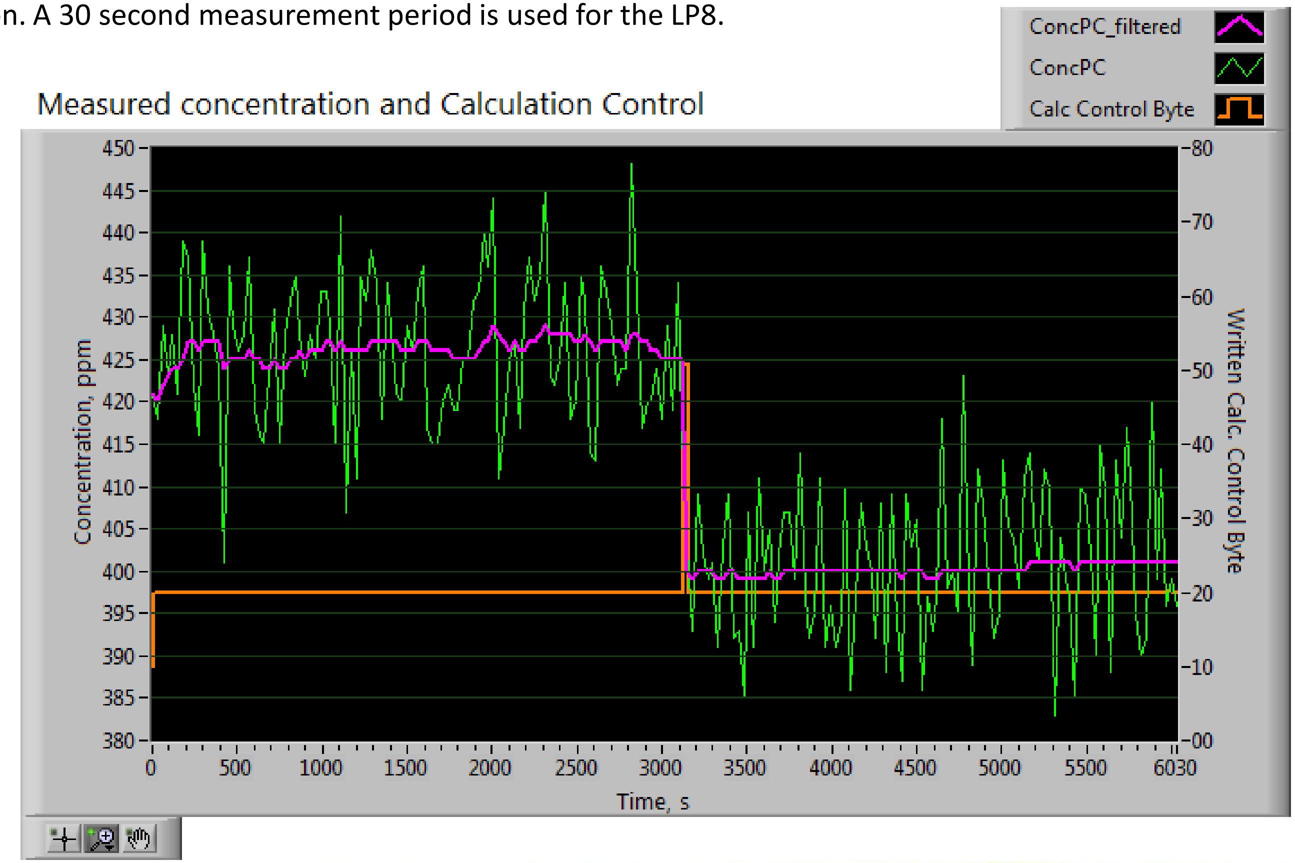
In the example below sensor shows higher concentration level then it is expected in 400 ppm environment and its accuracy is corrected by applying Background Calibration command to the Calculation Control. **Green line** is CO2_Conc_Pres, **Blue line** is Filt_Conc_Pres

Background calibration	”… with no resetting of filters”	“+ resetting of filters”
Using unfiltered data: Possible to under-/overshoot with the offset to what is the assumed fresh air background due to noise		
Using filtered data: More correctly set the offset to the actual fresh air background, but requires more samples in stable environment		

LP8 Background Calibration demonstrated

An example of filtered and unfiltered concentration behavior when implementing Background Calibration command (0x51) in the Calculation Control.

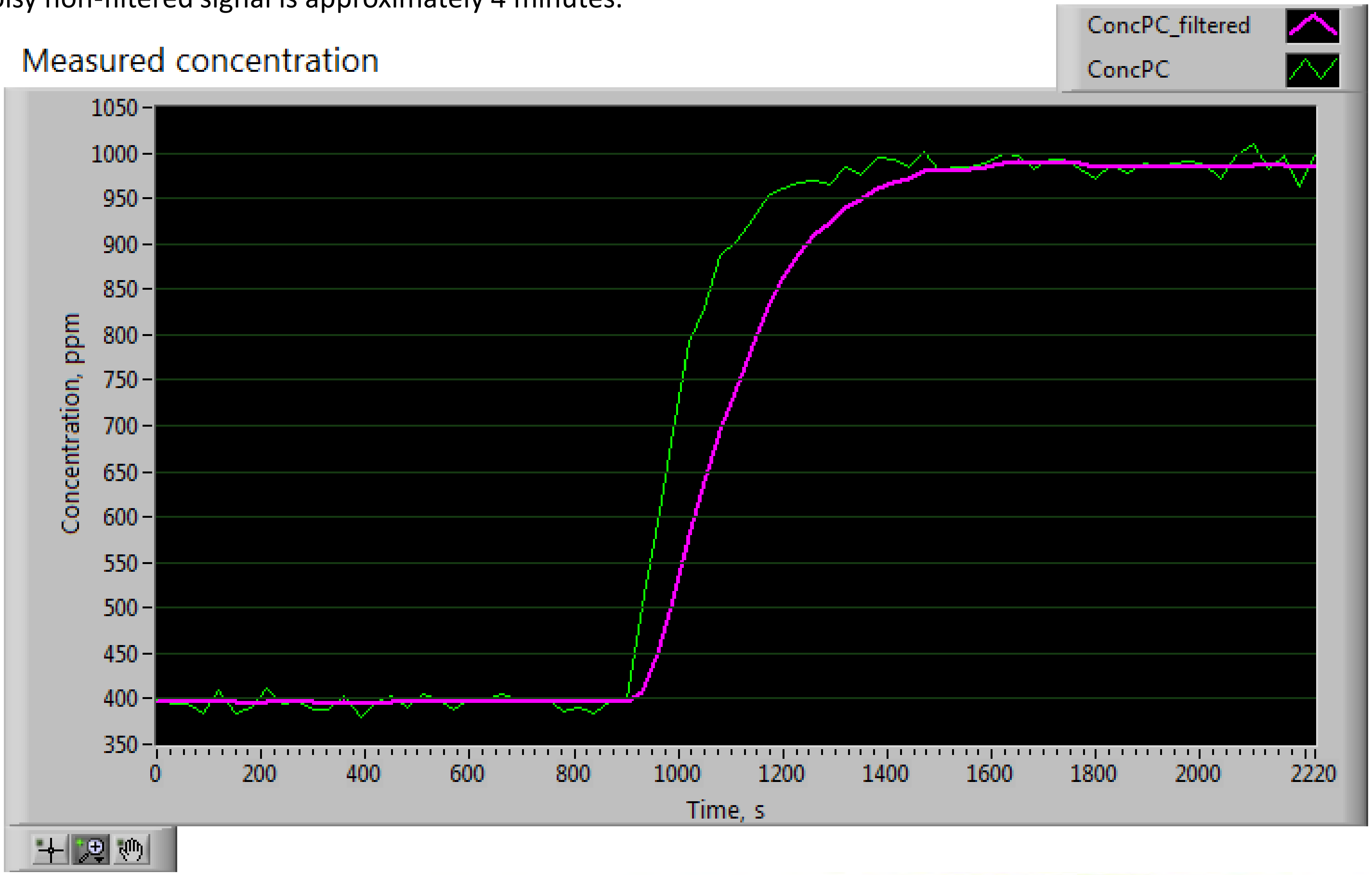
A true 400 ppm environment is created by calibration gas and the sensor initially show a drift roughly 25 ppm higher, which is to be corrected by the Background Calibration. A 30 second measurement period is used for the LP8.



Ex: LP8 sensor module response time, 16s

The stable gas concentration in an enclosure with LP8 sensor module is changed from 400 ppm to 1000 ppm.
Gas flow rate is ~1.5L/min, the enclosure volume is ~1L, so the concentration change step response is affected by this factors as well.

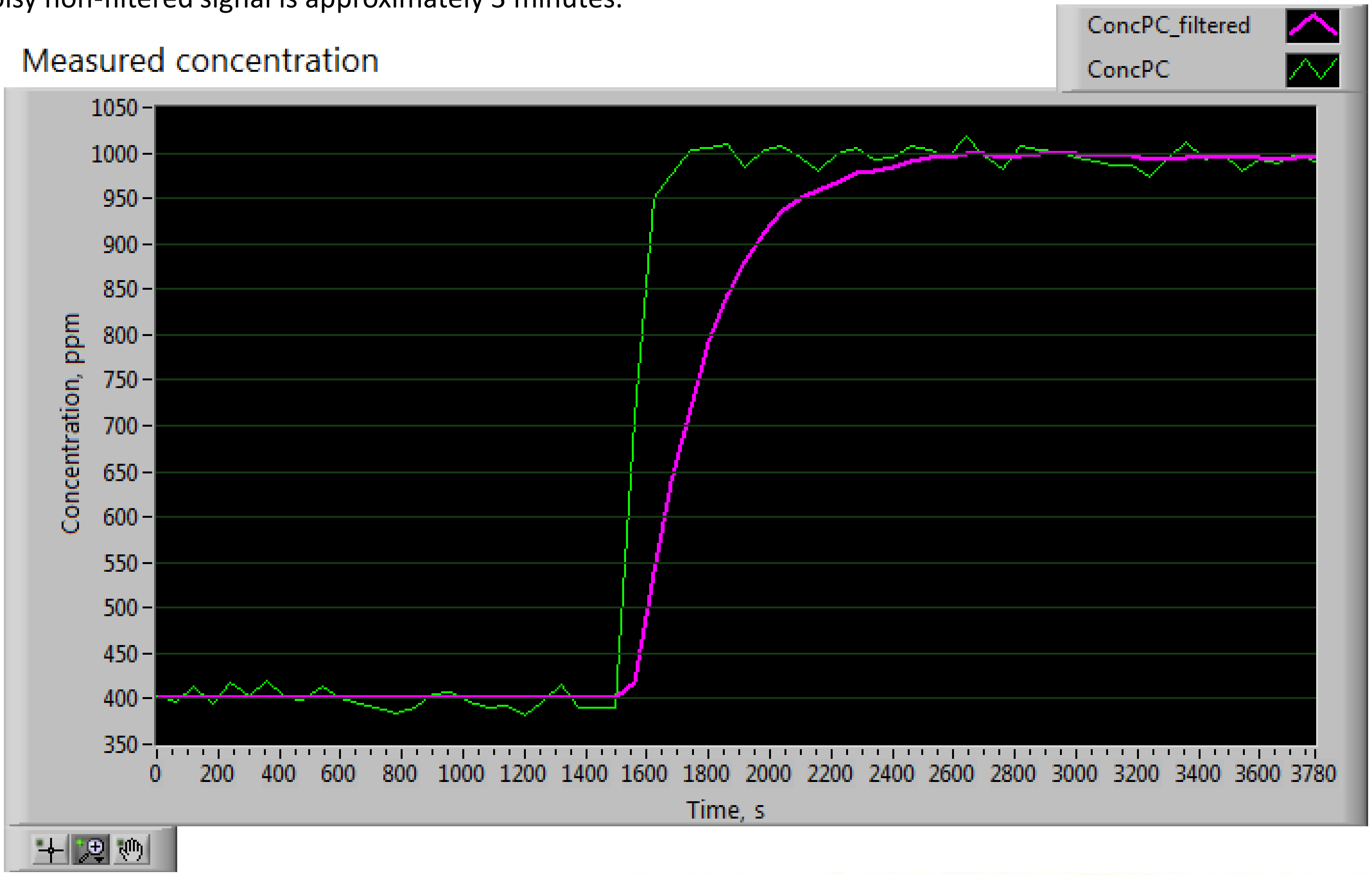
- Measurement period is controlled by application host to be 16 seconds.
- Noise-suppressed filtered signal settles to 90% of the step response in ca 7 minutes.
- The settling time of the noisy non-filtered signal is approximately 4 minutes.



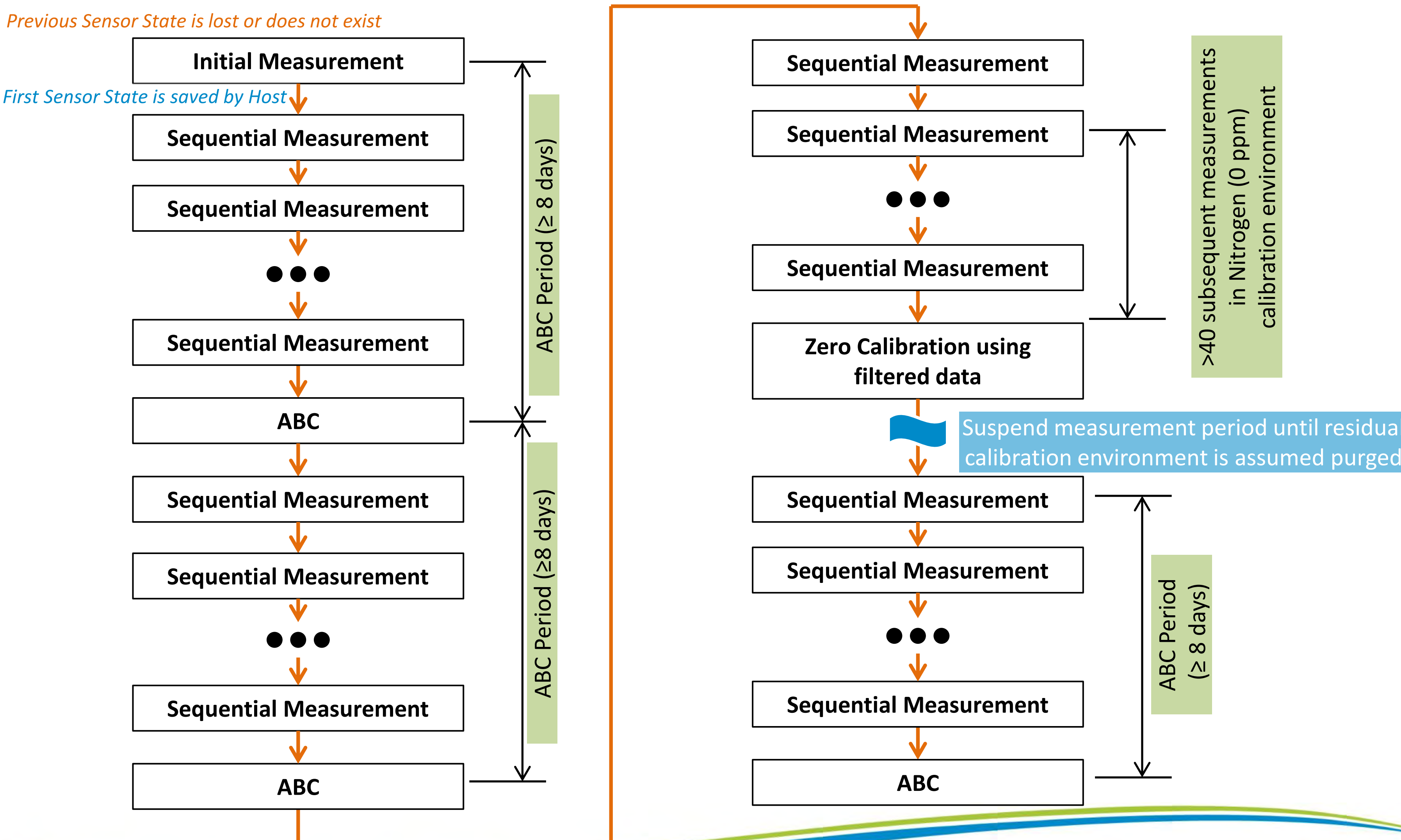
Ex: LP8 sensor module response time, 60s

The stable gas concentration in an enclosure with LP8 sensor module is changed from 400 ppm to 1000 ppm.
Gas flow rate is ~1.5L/min, the enclosure volume is ~1L, so the concentration change step response is affected by this factors as well.

- Measurement period is controlled by application host to be 60 seconds (1 minute).
- Noise-suppressed filtered signal settles to 90% of the step response in ca 10 minutes.
- The settling time of the noisy non-filtered signal is approximately 3 minutes.



Ex: Continuous Operation Sequence



Ex: Communication package frames in the LP8 measurement cycle

Bytes	Description	Corresponding sensor addresses
<CC>	Calculation Control, 1 byte	0x80
<Any1><Any2>…<Any23>	Any “don’t care” values, 23 bytes	0x81 to 0x97
<SS1><SS2>… <SS23>	Sensor State, 23 bytes	0x81 to 0x97
<PP_H><PP_L>	Host pressure value, 2 bytes The value 10124 (0x278C) is default pressure compensation in the sensor	0x98 to 0x99
<D1><D2>…<D18>	Measured data and sensor status, 18 bytes	0x9A to 0xAB
<CRC_L><CRC_H>	CRC, 2 bytes	

LP8 pressure compensation

1) If host is equipped with a pressure sensor it may write the current ambient pressure value to the addresses 0x98 and 0x99. LP8 pressure compensation defaults to using 10124 if host omits writing to the addresses.
2) If host is not equipped with a pressure sensor there are two options:
    a) Continuously write static 10124 to the Host Pressure, 0x98 and 0x99, in every write to RAM.
    b) Recommended: simply skip it by writing a shorter 24 bytes frame only to the addresses 0x80 to 0x97.

Initial Measurement w. host pressure (previous Sensor State is lost or does not exist)

1) Host powers up sensor
2) Host waits until RDY signal is set low
3) Host writes command "*Write 26 bytes starting from the address 0x0080, Calculation Control = 0x10*":

   <FE> <41> <00> <80> <1A> <10> <Any1> <Any2> ……. <Any23> <PP_H> <PP_L> <CRC_L> <CRC_H>
4) Host reads response if no communication error occurs:

   <FE> <41> <81> <E0>
5) Host waits until RDY signal is set high
6) Host writes command "*Read 44 bytes starting from the address 0x0080*":

   <FE> <44> <00> <80> <2C> <CRC_L> <CRC_H>
7) Host reads response if no communication error occurs:

   <FE> <44> <2C> <00> <SS1> <SS2> …… <SS23> <PP_H> <PP_L> <D1> <D2> …… <D18> <CRC_L> <CRC_H>
8) Host powers down sensor

Sequential Measurement, ABC, … w. host pressure (Sensor State is saved from the previous measurement)

1) Host powers up sensor
2) Host waits until RDY signal is set low
3) Host writes command "*Write 26 bytes starting from the address 0x0080, Calculation Control = CC*":

   <FE> <41> <00> <80> <1A> <CC> <SS1> <SS2> ……. <SS23> <PP_H> <PP_L> <CRC_L> <CRC_H>
4) Host reads response if no communication error occurs:

   <FE> <41> <81> <E0>
5) Host waits until RDY signal is set high
6) Host writes command "*Read 44 bytes starting from the address 0x0080*":

   <FE> <44> <00> <80> <2C> <CRC_L> <CRC_H>
7) Host reads response if no communication error occurs:

   <FE> <44> <2C> <00> <SS1> <SS2> …… <SS23> <PP_H> <PP_L> <D1> <D2> …… <D18> <CRC_L> <CRC_H>
8) Host powers down sensor

Initial Measurement w/o. host pressure (previous Sensor State is lost or does not exist)

1) Host powers up sensor
2) Host waits until RDY signal is set low
3) Host writes command "*Write 24 bytes starting from the address 0x0080, Calculation Control = 0x10*":
   <FE> <41> <00> <80> <18> <10> <Any1> <Any2> ……. <Any23> <CRC_L> <CRC_H>
4) Host reads response if no communication error occurs:
   <FE> <41> <81> <E0>
5) Host waits until RDY signal is set high
6) Host writes command "*Read 44 bytes starting from the address 0x0080*":
   <FE> <44> <00> <80> <2C> <CRC_L> <CRC_H>
7) Host reads response if no communication error occurs:
   <FE> <44> <2C> <00> <SS1> <SS2> …… <SS23> <PP_H> <PP_L> <D1> <D2> …… <D18> <CRC_L> <CRC_H>
8) Host powers down sensor

Sequential Measurement, ABC, … w/o. host pressure (Sensor State is saved from the previous measurement)

1) Host powers up sensor
2) Host waits until RDY signal is set low
3) Host writes command "*Write 24 bytes starting from the address 0x0080, Calculation Control = CC*":
   <FE> <41> <00> <80> <18> <CC> <SS1> <SS2> ……. <SS23> <CRC_L> <CRC_H>
4) Host reads response if no communication error occurs:
   <FE> <41> <81> <E0>
5) Host waits until RDY signal is set high
6) Host writes command "*Read 44 bytes starting from the address 0x0080*":
   <FE> <44> <00> <80> <2C> <CRC_L> <CRC_H>
7) Host reads response if no communication error occurs:
   <FE> <44> <2C> <00> <SS1> <SS2> …… <SS23> <PP_H> <PP_L> <D1> <D2> …… <D18> <CRC_L> <CRC_H>
8) Host powers down sensor

The most efficient initialization alternative when the host also doesn't have a pressure sensor, simply skipping writing any of the "don't care" values to buffer the LP8 Sensor State:

Initial Measurement w/o. host pressure (previous Sensor State is lost or does not exist) – efficient option

1) Host powers up sensor
2) Host waits until RDY signal is set low
3) Host writes command "*Write 1 byte starting from the address 0x0080, Calculation Control = 0x10*":
   <FE> <41> <00> <80> <01> <10> <28> <7E>
4) Host reads response if no communication error occurs:
   <FE> <41> <81> <E0>
5) Host waits until RDY signal is set high
6) Host writes command "*Read 44 bytes starting from the address 0x0080*":
   <FE> <44> <00> <80> <2C> <CRC_L> <CRC_H>
7) Host reads response if no communication error occurs:
   <FE> <44> <2C> <00> <SS1> <SS2> …… <SS23> <PP_H> <PP_L> <D1> <D2> …… <D18> <CRC_L> <CRC_H>
8) Host powers down sensor

Initial Measurement, Ext. ref. calibration,… w/o. host pressure (Sensor State is saved from previous measurement)

1) Host powers up sensor
2) Host waits until RDY signal is set low
3) Host writes command "*Write 24 bytes starting from the address 0x0080, Calculation Control = CC*":
   <FE> <41> <00> <80> <18> <CC> <SS1> <SS2> ……. <SS23> <CRC_L> <CRC_H>
4) Host reads response if no communication error occurs:
   <FE> <41> <81> <E0>
5) Host waits until RDY signal is set high
6) Host writes command "*Read 44 bytes starting from the address 0x0080*":
   <FE> <44> <00> <80> <2C> <CRC_L> <CRC_H>
7) Host reads response if no communication error occurs:
   <FE> <44> <2C> <00> <SS1> <SS2> …… <SS23> <PP_H> <PP_L> <D1> <D2> …… <D18> <CRC_L> <CRC_H>
8) Host powers down sensor

Static Modbus Response PDU

For easier error-handling, the host may simply compare the static LP8 response PDU from any successful write function request.

In full ADU package frame format, by addressing using Modbus device address FE “any sensor” this response always looks like this;

<Device address>	<Function Code>	<CRC Low>	<CRC High>
0xFE	0x41	0x81	0xE0

Error Handling

ErrorStatus structure

Error Status	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
ErrorStatus0	WarmUp	Memory	OutOfRange	SelfDiag	Calibration Error	AlgError	Reserved	FatalError
ErrorStatus1	Parameters override bits				DetectSig Error	ADC Error	VCAP2 low	VCAP1 low
ErrorStatus2	Reserved				Noisy non-filtered concentration channel OOR bits			
ErrorStatus3	Reserved				Noise-suppressed filtered concentration channel OOR bits			

Error Handling

ErrorStatus0 byte description

Bit	Bit Name	Error Description	Suggested Action
0	FatalError	Fatal Error The bit is a joint bit for different error sources when sensor can not provide correct operation, among them: • Configuration EEPROM parameters are out of range or corrupted • Virtual EEPROM memory read/write error • Error in VCAP measurements	Switch off/on sensor power and start with “Initial Measurement” in the Calculation Control byte. Contact local distributor.
2	AlgError	Algorithm Error Configuration EEPROM parameters are out of range or corrupted	Switch off/on sensor power and start with “Initial Measurement” in the Calculation Control byte. Contact local distributor.
3	Calibration Error	Calibration Calculation Error Out of range error at Zero-/Background calibration and ABC	Repeat recalibration or wait until next ABC event.
4	SelfDiag	Self Diagnostics Error Hardware error is detected or important EEPROM parameters are corrupted	Contact local distributor.
5	OutOfRange	Out Of Range Error (OOR) Indicates an error which occurs at different stages of concentration calculation algorithm. Resets automatically after source of error disappears.	Try sensor in fresh air. Perform sensor zero or background calibration. Check sensor temperature readings.
6	Memory	Memory Error Virtual EEPROM read/write error: page checksum error during read or write verification, FLASH operation error.	Contact local distributor.
7	WarmUp	WarmUp bit Bit is only used in SenseAir production System and is not set in normal operation by customers mode	-

Error Handling

ErrorStatus1 byte description

Bit	Bit Name	Error Description	Suggested Action
0	VCAP1 low	VCAP1 voltage low Voltage measured prior lamp pulse is below preset threshold. The threshold is 2.8V±3%.	Check battery. Sensor supply voltage is below specified operational limit of 2.9V.
1	VCAP2 low	VCAP2 voltage low Average voltage measured at the beginning of lamp pulse (during inrush steps) is below preset threshold. The threshold is 2.7V±3%.	Equivalent series resistance of the sensor power supply source (a battery or super-capacitor) is not enough to provide low-voltage drop during 125mA lamp inrush step.
2	ADC Error	ADC Error MCU ADC out-of-range error has occurred.	Switch off/on sensor power and apply “initial measurement” to the Calculation Control byte. Contact local distributor.
3	Reserved		
4-7	Parameters override bits	This bits indicate which parameter is forced to a predefined value in the debug mode. Should not appear during normal operation.	-

Error Handling

ErrorStatus2 and ErrorStatus3 byte description

Bit	Bit Name	Error Description	Suggested Action
0	Signal baseline	The raw signal from emitter is out of specification	Contact local distributor.
1	Temperature	The temperature effect on the thermopile is out of specification	Contact local distributor.
2	Table	The signal conversion to CO_2 concentration value doesn't find a match	Contact local distributor.
3	Pressure	The host pressure used for compensation is out of specification	Contact local distributor.

The bits 3-0 in the **ErrorStatus2** (non-filtered data) and **ErrorStatus3** (noise-suppression filtered data) bytes decode on what algorithm stage an ”Out Of Range Error” (OOR) has occurred respectively.

ErrorStatus2	Unfiltered Concentration Errors							
					Pressure correct OOR	Table entry OOR	TempComp OOR	IR_Signal OOR
ErrorStatus3	Filtered Concentration Errors							
					Pressure correct OOR	Table entry OOR	TempComp OOR	IR_Signal OOR

LP8 Adapter Board

LP8 Adapter Board facilitates easy evaluation of LP8 sensor by connecting it to a PC or host MCU system:

- Power options: USB 5V or an external power supply
- The same connector can be used either for connecting sensor to a PC using FTDI-cable or connecting to a host MCU system for quick prototyping
- A switch for VCAP
- A footprint for optional super-capacitor (ex. Eaton Bussman PM-5R0H474-R 0.47F 5V)
- Shunt resistors for measuring current: 1Ω on VCAP and 20Ω on VBB
- Factory Connector for use with other SenseAir sensor models
- Use SenseAir's UIP5 PC software for simple communication to LP8
- Please contact SenseAir for, UMA2183 LP8 Adapter User Guide, for more information on this adaptor