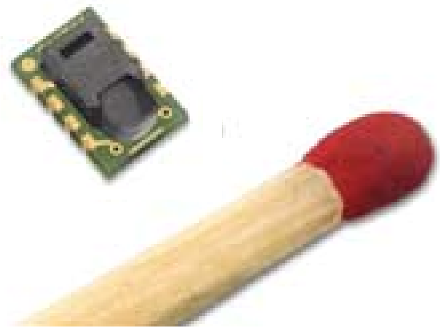
SENSIRION
THE SENSOR COMPANY

SHT1x
Humidity & Temperature Sensmitter

_ Relative humidity and temperature sensors
_ Dew point
_ Fully calibrated, digital output
_ No external components required
_ Ultra low power consumption
_ Surface mountable package
_ Excellent long-term stability
_ Small size
_ Automatic power down

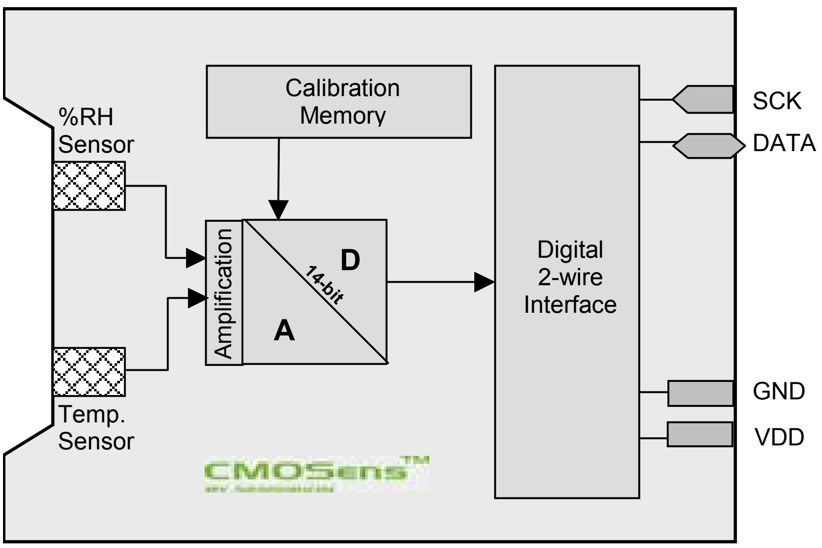


Preliminary Information March 2002

SHT1x Product Summary

The SHT1x is a single chip relative humidity and temperature multi sensor module comprising a calibrated digital output. Application of industrial CMOS processes with customized post processing (CMOSens™ technology) ensures highest reliability and excellent long term stability. The device includes two calibrated microsensors for relative humidity and temperature which are seamlessly coupled to a 14bit analog to digital converter and a serial interface circuit on the same chip. This results in superior signal quality, a fast response time and insensitivity to external disturbances (EMC) at a very competitive price.
Each sensor is calibrated in a precision humidity chamber and the calibration coefficients are programmed into the OTP memory. These coefficients are used internally during measurements to calibrate the signals from the sensors.
The 2-wire serial interface allows easy and fast system integration. Its tiny (7x5x3mm) size and low power consumption makes it the ultimate choice for even the most demanding applications including automotive, instrumentation, medical equipment, heating, ventilation and air conditioning systems (HVAC), portable consumer electronics and battery-operated controllers.
The device is supplied in a surface-mountable LCC type package. Other packaging options are available on request.

SHT1x Single Chip Relative Humidity and Temperature Sensor Module



1 Sensor Performance Specifications[1]

Parameter	Conditions	Min.	Typ.	Max.	Units
Humidity					
Resolution		0.5	0.03	0.03	% RH
		8	12	12	bit
Repeatability			±0.1		% RH
Accuracy[2] & Interchangeability		see figure 1			
Nonlinearity	10 - 90 %RH		±3		% RH
Range		0		100	% RH
Response time	1/e (63%) slowly moving air		4		s
Hysteresis			±1		% RH
Long term stability	Typical		< 1		% RH/yr
Temperature					
Resolution		0.04	0.01	0.01	°C
					°F
		12	14	14	bit
Repeatability			±0.1		°C
			±.2		°F
Accuracy		see figure 1			
Range		-40		123.8	°C
		-40		254.9	°F
Response Time	1/e (63%)	5		30	s

Table 1 Sensor Performance Specifications

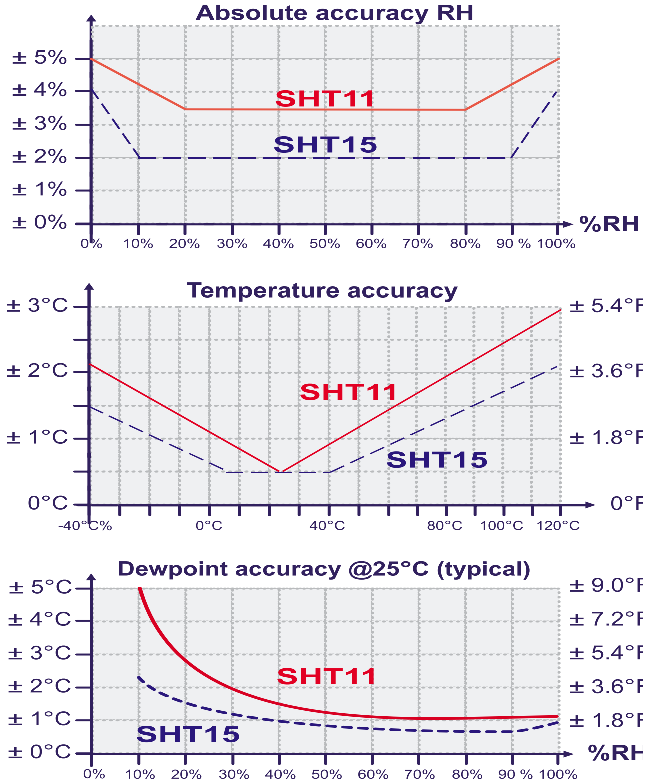


Figure 1 RH, Temperature and dewpoint accuracies

1.1 Converting the digital output to physical values

1.1.1 Humidity

To compensate for the non-linearity of the humidity sensor and to obtain the full accuracy it is recommended to convert the readout with the following formula:

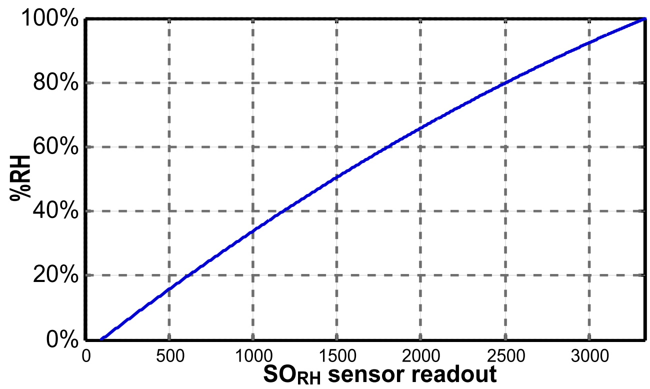
$$RH_{linear} = c_1 + c_2 \bullet SO_{RH} + c_3 \bullet SO_{RH}^2$$

where SO_{RH} is the sensor readout for RH and with

$c_1 = -4$ $c_2 = 0.0405$ $c_3 = -2.8 * 10^{-6}$ for 12bit SO_{RH}

$c_1 = -4$ $c_2 = 0.648$ $c_3 = -7.2 * 10^{-4}$ for 8bit SO_{RH}

For simplified, less computation intense conversion formulas see application note "RH Non-Linearity Compensation".



For temperatures significantly different from 25°C (~77°F) the temperature coefficient of the RH sensor should be considered:

$$RH_{true} = (T_{°C} - 25) \bullet (t_1 + t_2 \bullet SO_{RH}) + RH_{linear}$$

with $t_1 = 0.01$; $t_2 = 0.00008$, $t_2 = 0.00128$ for 8bit SO_{RH}

This equals ~0.12%RH / °C @ 50%RH

1.1.2 Temperature

The temperature sensor is very linear by design. Use the following formula to convert from digital readout to temperature:

Temperature = $d_1 + d_2 * SO_T$

Use the appropriate table entries for 5V or 3V.

	Celsius		Fahrenheit	
SO_T	**d_1**	**d_2**	**d_1**	**d_2**
14bit 5V	-40	0.01	-40	0.018
12bit 5V	-40	0.04	-40	0.072
14bit 3V	-38.4	0.0098	-37.1	0.0176
12bit 3V	-38.4	0.0392	-37.1	0.0704

This equals a voltage dependency of ~ -0.2°C/V @ 25°C

1.1.3 Dewpoint

See application note "Dewpoint calculation" for more information.

(1) For operation within normal operation range as described in Chapter 3
(2) Not including non-linearity
(3) The default measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit through the status register.

2 Serial Interface

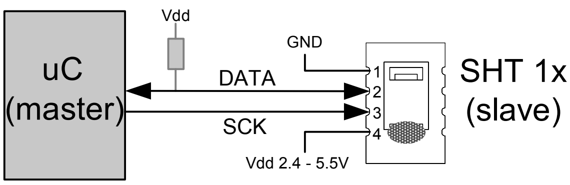


Figure 2 Typical application circuit

Pin	Name	Comment
1	GND	Ground
2	DATA	Serial data bidirectional
3	SCK	Serial clock input
4	VDD	Supply 2.4 – 5.5V

Table 2 SHT1x Pin Description

2.1 Power Pins

The SHT1x requires a voltage supply between 2.4V and 5.5V. After powerup the device requires 11ms to reach its "sleep" state. No commands should be sent before that time. Power supply pins (VDD, GND) may be decoupled with a 100 nF capacitor.

2.2 **I/O Pins** (Bidirectional 2-wire Interface)

See Table 5 for a detailed IO characteristics.

2.2.1 Serial clock input (SCK)

The SCK is used to synchronize the communication between a master and the SHT1x.

2.2.2 Serial data (DATA)

The DATA tristate pin is used to transfer data in and out of the device. Data must be updated on this pin after the falling edge and is valid on the rising edge of the serial clock SCK. An external pull-up resistor is required to pull the signal high. (See Figure 2). Pull-up resistors are often included in I/O circuits of microcontrollers

2.2.3 Command sequence

To initiate a transmission a "Transmission Start" sequence has to be issued. It consists of a lowering of the DATA line while SCK is high, followed by a low pulse on SCK and raising DATA again while SCK is still high.

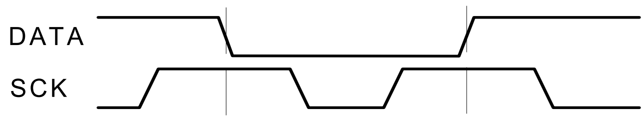


Figure 3 "Transmission Start" sequence

The subsequent command sequence consists of three address bits (only "000" is currently supported) and five command bits. The proper reception of a command by the SHT1x is indicated by pulling the ack bit low on the DATA pin.
See 2.2.5 "Measurement Sequence" for an application of the command sequence

2.2.4 Connection reset sequence

If communication with the SHT1x is lost the following signal sequence will reset its serial interface:
While leaving DATA high toggle SCK 9 or more times. This must be followed by a "Transmission Start" sequence preceding the next command.

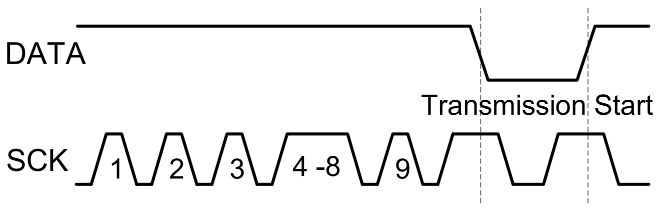
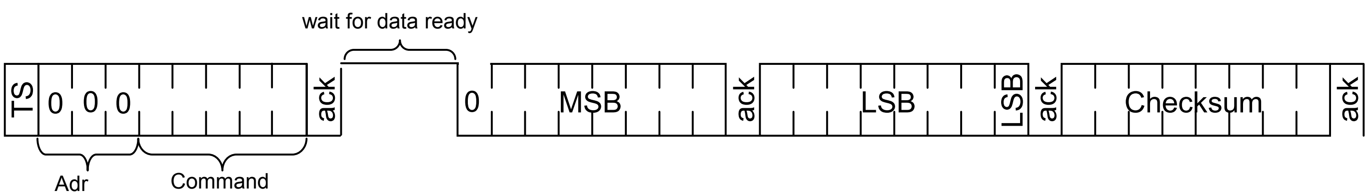


Figure 4 Connection reset sequence

Command	Code	Description
Reserved	0000x	Reserved
Measure Temperature	**00011**	**Temperature measurement**
Measure Humidity	**00101**	**Humidity measurement**
Status Register Read	00111	Read access to the status register (see application note)
Status Register Write	00110	Write access to the status register (see application note)
Reserved	0101x-1110x	Reserved
Soft reset	**11110**	**resets the chip, clears the status register to default values** wait 11ms before next command

Table 3 SHT1x list of commands



2.2.5 *Measurement sequence (T and RH)*

After issuing a measurement command ('00000101' for RH, '00000011' for Temperature) the controller has to wait for the measurement to complete. This takes approximately 11/55/210ms for a 8/12/14bit measurement. The exact time varies by up to ±15% with the speed of the internal oscillator. To signal the completion of a measurement, the SHT1x pulls down the data line ② and the controller must restart SCK. Two bytes of measurement data and one byte of CRC checksum will then be transmitted. The uC must acknowledge each byte by pulling the DATA line low. All values are MSB first, right justified. (e.g. the 5th SCK is MSB for a 12bit value).

Communication terminates after the acknowledge bit of the CRC data. If CRC-8 Checksum is not used the controller may terminate the communication after the measurement data LSB by keeping ack high.

The SHT 11 automatically returns to sleep mode after the measurement and communication have finished.

Warning: To keep heat up of the SHT1x below 0.1°C it should not be active for more than 15% of the time (e.g. max. 3 measurements / second for 12bit accuracy).

2.2.6 *CRC-8 Checksum Calculation*

Please consult application note "CRC-8 Checksum Calculation" for information on how to calculate the CRC.

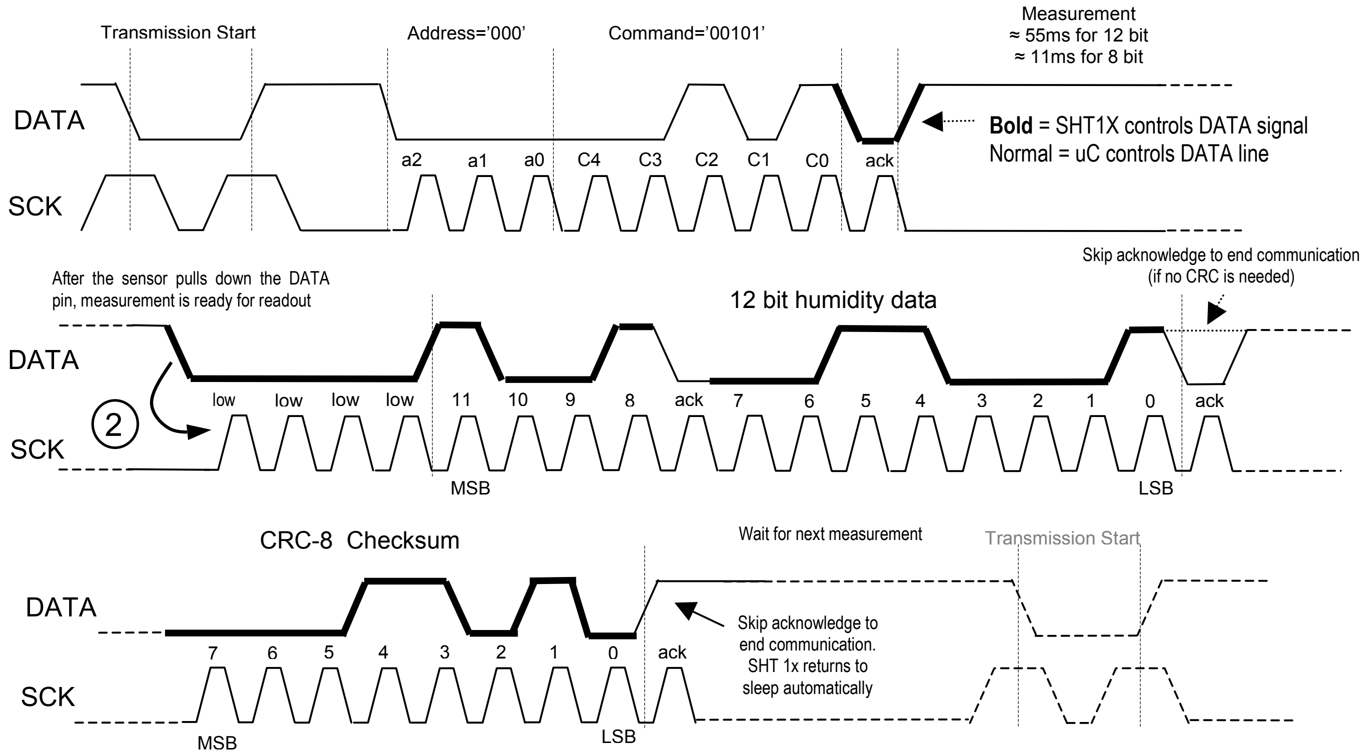


Figure 5 Example RH measurement sequence for value "0000'1001 ' 0011'0001"= 2353 = 75.79%RH

2.3 Status Register

Some of the advanced functions of the SHT1x are available through the status register. The following section gives a brief overview of these features. Please consult application note "Status Register" for more information.

2.3.1 *Heater*

An on chip heating element can be switched on. It will increase the temperature of the sensor by approximately 5°C. Power consumption will increase by 8mA @ 5V.

Applications:

- By comparing temperature and humidity values before and after switching on the heater, proper functionality of both sensors can be verified.
- In high RH environments heating the sensor element will avoid condensation.

Warning: The built-in calibration is not correct while the SHT1x is heated!

2.3.2 *End Of Life (EOL)*

The SHT1x End of Life (EOL) function detects VDD voltages below 2.47V. Accuracy is ±0.05V

2.3.3 *Measurement resolution*

The default measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit. This is especially useful in high speed or extreme low power applications.

Please consult application note "Status Register" for more information on how to access and use these features.

3 Specifications SHT1x

3.1 Absolute Maximum Ratings

Ambient Storage Temperature: -40°C to 120°C

3.2 Operating Conditions

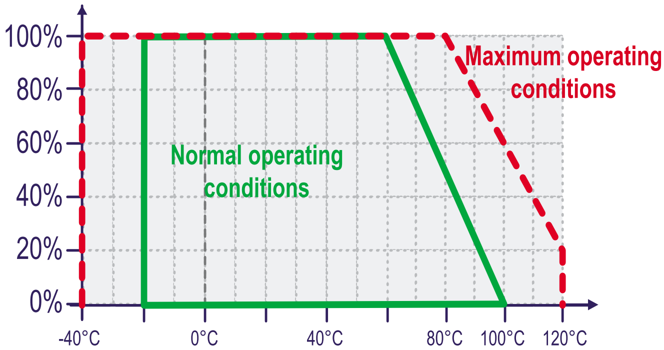


Figure 6 Recommended operating conditions

Temperatures outside the recommended range may temporarily offset the RH signal by up to +3%RH. The sensor will slowly return to calibration conditions but heating the device up to 90°C at <5%RH for 24h will reverse the effect of high RH, high temperature environments promptly. Prolonged exposure to extreme conditions may accelerate ageing of the sensor.

3.3 Special Conditions

Extensive tests were performed in various environments. Please contact us for complete qualification results.

3.4 Electrical Specifications[1]

3.4.1 ESD (Electrostatic Discharge)

ESD immunity is qualified according to MIL STD 883E, method 3015 (Human Body Model at ±2kV)).
Latch-up immunity is provided at a force current of ±100 mA with T_{amb}=80°C according to JEDEC 17.

3.4.2 DC Characteristics

VDD=5V, Temperature= 25°C unless otherwise noted

Parameter	Conditions	Min.	Typ.	Max.	Units
Power supply DC		2.4	5	5.5	V
Supply current	measuring		550		µA
	average	2[2]	28[3]		µA
	sleep		0.3	1	µA
Low level output voltage		0		20%	Vdd
High level output voltage		75%		100%	Vdd
Low level input voltage	Negative going	0		20%	Vdd
High level input voltage	Positive going	80%		100%	Vdd
Input current on pads				1	µA
Output peak current	on			4	mA
	Tristated (off)		10		µA

Table 4 SHT1x DC Characteristics

3.4.3 I/O Characteristics

	Parameter	Conditions	Min	Typ.	Max.	Unit
F_{SCK}	SCK frequency	VDD > 4.5 V			10	MHz
		VDD < 4.5 V			1	MHz
T_{RFO}	DATA fall time	Output load 5 pF	3.5	10	20	ns
		Output load 100 pF	30	40	200	ns
T_{CLH}	SCK high time		100			ns
T_{CLL}	SCK low time		100			ns
T_V	DATA valid from			50		ns
T_{HO}	Output hold time		0	10		ns
T_R/T_F	SCK rise/fall time				200	ns

Table 5 SHT1x I/O Signals Characteristics

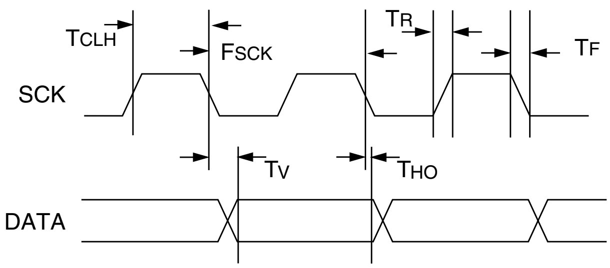


Figure 7 Timing Diagram

(1) Parameters are periodically sampled and not 100% tested
(2) With one measurement of 8 bit accuracy without OTP reload per second
(3) With one measurement of 12bit accuracy per second

4 Physical Dimensions and Mounting Information

4.1 Package type

The device is supplied in a surface-mountable LCC type package. The sensors housing consists of a Liquid Crystal Polymer (LCP) cap with epoxy glob top on a standard 0.8mm FR4 substrate.
Device size is 7.62 x 5.08 x 2.5 mm. Weight 100mg
Other packaging options are available on request.

4.2 Mounting Recommendations

The relative humidity of a gas strongly depends on its temperature. It is therefore essential to keep the sensor at the same temperature as the air of which the humidity is to be measured.
If the SHT1x shares a PCB with heating electronic components it should be mounted below the heat source and the housing must remain well ventilated. To reduce heat conduction copper layers between the SHT1x and the rest of the PCB should be minimized and a slit may be milled in between.

Prolonged direct exposure of the SHT1x to strong light or UV radiation should be avoided.

4.3 Wiring considerations and signal integrity

Carrying the SCK and DATA signal parallel and in close proximity (e.g. in wires) for more than 10cm may result in crosstalk and loss of communication. This may be resolved by routing VDD and/or GND between the two signals.

4.4 Soldering Information

The SHT1x can be soldered using standard reflow ovens at maximum 225°C for 20 seconds. For manual soldering contact time must be limited to 5 seconds at up to 350°C
Please consult the application note "Soldering procedure" for detailed instructions.

4.5 Delivery Conditions

The SHT1x will be delivered in standard IC tubes by max. 80 pieces per tube. Other delivery options may be available on request.

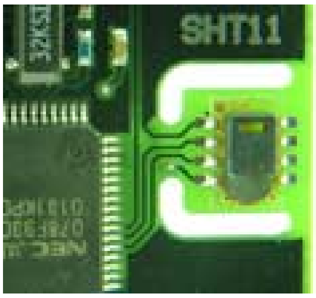
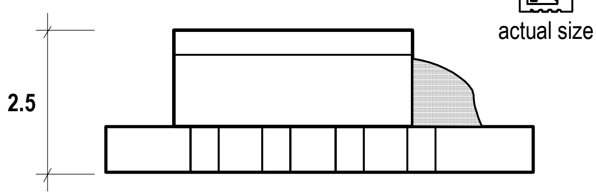
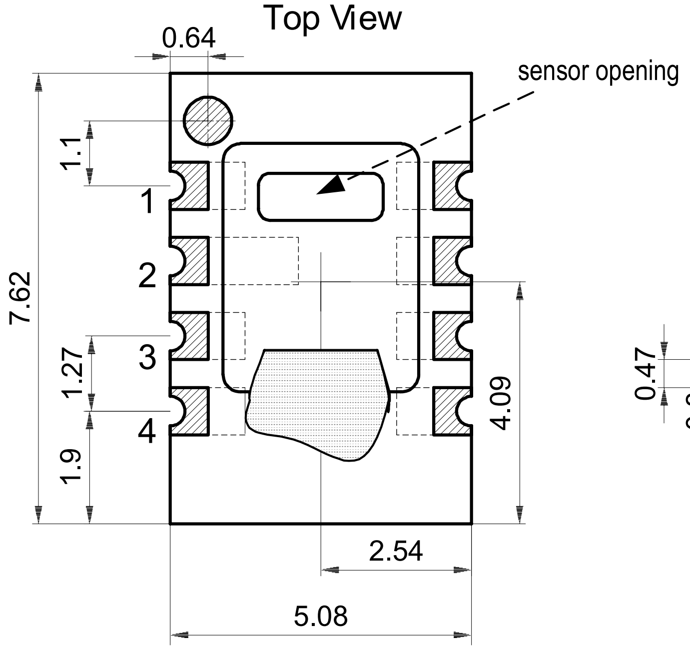


Figure 8 Mounting example

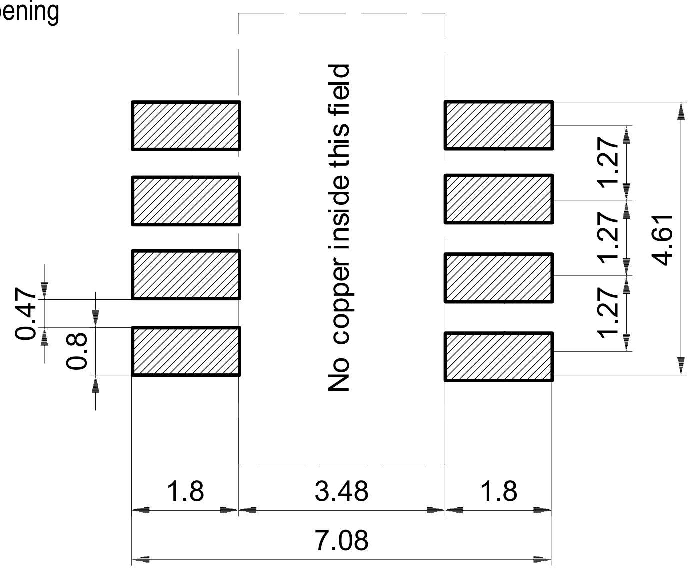
Side View



Top View



Recommended PCB Footprint



all measurements in mm

5 Revision history

Date	Page	Changes
February 2002	1-9	First public release
February (2) 2002	4	Corrected CRC information to match application note
March 2002	2	Extended SHT11 3.5 accuracy range to 20%-80%
	8	Added image of mounting example
	2	Changed coefficients of temperature conversion formula
		Various small modifications

The latest version of this document and all application notes can be found at:
www.sensirion.com/en/download/humiditysensor/SHT11.htm

6 Important Notices

The warranty for each SENSIRION AG product comes in the form of a written warranty which governs sale and use of such product. Such warranty is contained in the printed terms and conditions under which such product is sold, or in a separate written warranty supplied with the product. Please refer to such written warranty with respect to its applicability to certain applications of such product.

These products may be subject to restrictions on use. Please contact SENSIRION AG for a list of the current additional restrictions on these products. By purchasing these products, the purchaser of these products agrees to comply with such restrictions. Please contact SENSIRION AG for clarification of any restrictions described herein.
SENSIRION AG reserves the right, without further notice, to change the SENSIRION SHT1x Relative Humidity and Temperature Sensor product specifications and/or information in this document and to improve reliability, functions and design.

SENSIRION AG assumes no responsibility or liability for any use of SENSIRION SHT1x product. Application examples and alternative uses of the SENSIRION SHT1x are for illustration purposes only and SENSIRION AG makes no representation or warranty that such applications shall be suitable for the use specified.

7 Caution

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

Headquarters and Sales Office

SENSIRION AG
Eggbühlstr. 14
P.O. Box
CH-8052 Zürich
Switzerland

Phone: + 41 (0)1 306 40 00
Fax: + 41 (0)1 306 40 30
e-mail: info@sensirion.com
http://www.sensirion.com/

SHTxx
Humidity & Temperature
Sensmitter

Application Note
Status Register

1 Introduction

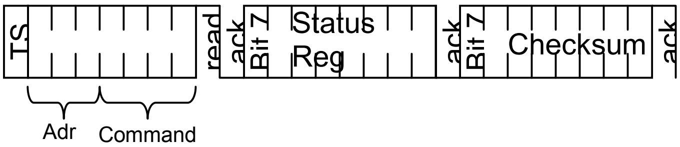
Some of the advanced functions of the SHTxx are available through the status register.
This document describes the required communication and the features available through the status register.

2 Revision History

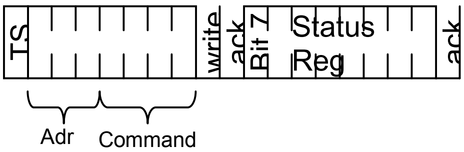
August 27, 2001	C2	URO	Revision 0.9 (Preliminary)
October 20, 2001	C2	URO	Revision 1.00 changed to new CI
November 12, 2001	C2	URO	Revision 1.10 added status register bit for resolution
November 22, 2001	C2	URO	Revision 1.11 corrected polarity of resolution bit
January 24,2002	C1	URO	Revision 1.2 default values bit 4-7, EOL paragraph, small typos

3 The Status Register

3.1.1 Status Register read



3.1.2 Status Register write



3.1.3 Status Register

Bit	Type	Description	Default	
7		reserved	0	
6	R	End of Life (low voltage detection)	X	
5		reserved	0	
4		reserved	0	
3		For Testing only, do not use	0	
2	R/W	Heater	0	off
1	R/W	no reload from OTP	0	reload
0	R/W	'1' = 8bit RH / 12bit Temperature resolution '0' = 12bit RH / 14bit Temperature resolution	0	12bit RH 14bit Temp.

3.1.4 Heater

An on chip heating element can be switched on. It will increase the temperature of the sensor by approximately 5°C. Power consumption will increase by 8mA @ 5V.
Applications:

- By comparing temperature and humidity values before and after switching on the heater, proper functionality of both sensors can be verified.
- In high RH environments heating the sensor element will avoid condensation.

Warning: The built-in calibration is not correct while the SHT11 is heated!

3.1.5 End Of Life (EOL, low voltage detector)

The SHT11 End of Life (EOL) function detects VDD voltages below 2.45V. Accuracy is ±0.1V
Warning: This bit is only updated during a measurement.

3.1.6 Calibration reload before measurement

To save power and gain speed the OTP reload before every measurement may be bypassed. This saves ~8.2ms from each measurement time.
Explanation:
In rare ESD environments the SHT11 may temporarily lose the calibration data from the volatile memory. Default is therefore to reread it from the OTP memory before every measurement.

3.1.7 Measurement resolution

The measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit. This is especially useful in high speed or extreme low power applications
"0" is 12/14bit "1" is 8/12bit.

3.2 Digital state machine

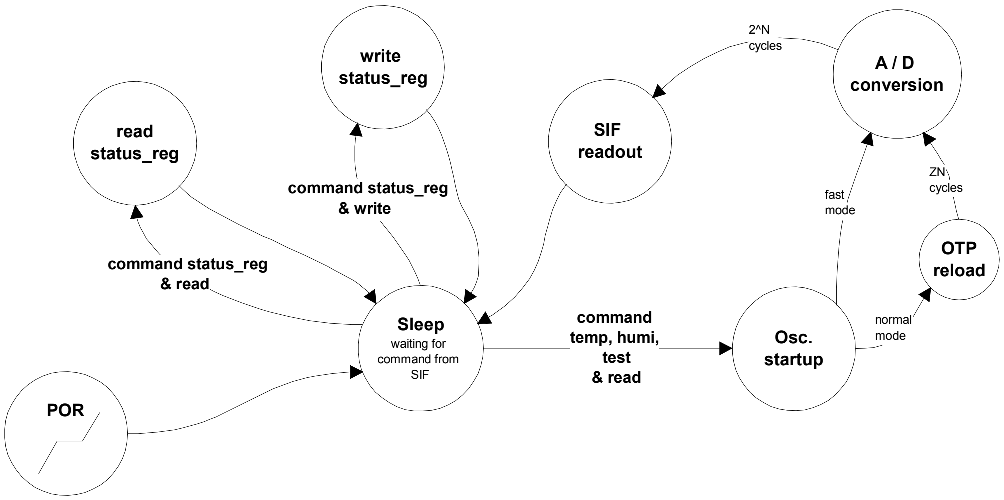


Figure 1 Digital Finite State Machine State Diagram

Headquarters and Sales Office

SENSIRION AG
Eggbühlstr. 14
P.O. Box
CH-8052 Zürich
Switzerland

Phone: + 41 (0)1 306 40 00
Fax: + 41 (0)1 306 40 30
e-mail: info@sensirion.com
http://www.sensirion.com/

SHTxx
Humidity & Temperature
Sensmitter

Application Note
CRC

1 Introduction

A CRC checksum is calculated over the whole transmission. If a CRC mismatch is detected, the SHTxx should be reset (command "00011110") and the measurement should be repeated.

2 Theory

CRC stands for Cyclic Redundancy Check. It is one of the most effective error detection schemes and requires a minimal amount of hardware.
For in-depth information on CRC we recommend the comprehensive: "A painless guide to CRC error detection algorithms" available at: http://www.repairfaq.org/filipg/LINK/F_crc_v3.html

The polynomial used in the SHTxx is: $x^8+ x^5+ x^4$. The types of errors that are detectable with this polynomial are:
1. Any odd number of errors anywhere within the transmission.
2. All double-bit errors anywhere within the transmission.
3. Any cluster of errors that can be contained within an 8-bit "window" (1-8 bits incorrect).
4. Most larger clusters of errors.

The CRC register initializes with the value of the lower nibble of the status register ("0000'$s_3s_2s_1s_0$", default "00000000"). It covers the whole transmission (command and response bytes) without the acknowledge bits. See the datasheet SHT11 on page 4 for an example of CRC readout.
The receiver can perform the CRC calculation upon the first part of the original message and then compare the result with the received CRC- 8. If a CRC mismatch is detected, the SHTxx should be reset (command "00011110") and the measurement should be repeated.

This application note will cover two methods for checking the CRC. The first "Bitwise" is more suited for hardware or lowlevel implementation while the later "Bytewise" is the preferred method for more powerful microcontroller solutions.

2.1 Bitwise

With the bitwise method, the receiver copies the structure of the CRC generator in hard- or software.
An algorithm to calculate this could look like this:

1) Initialise CRC Register to low nibble of status register (reversed ($s_0s_1s_2s_3$'0000))
2) Compare each (transmitted and received) bit with bit 7
3) If the same: shift CRC register, bit0='0'
   else: shift CRC register and then invert bit4 and bit5, bit0='1' (see figure 1)
4) receive new bit and go to 2)
5) The CRC value retrieved from the SHTxx must be reversed (bit 0 = bit 7, bit 1=bit 6 … bit 7 = bit 0) and can then be compared to the final CRC value.[2]

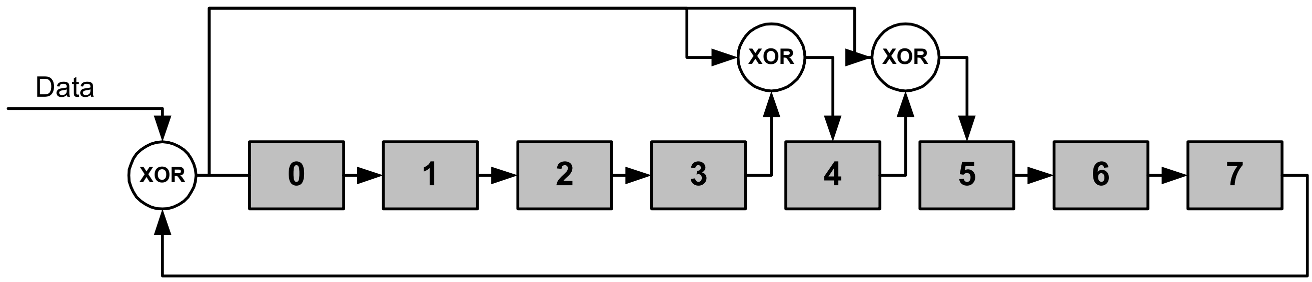


Figure 1 Internal structure of the SHTxx CRC-8 generator

2.1.1 Example for bitwise

Example 2: RH Measurement (as example in datasheet)

Input bit's	CRC Value 0011'0001	0x	dec	Comment
	0000'0000			Start with contents of status register[1]
0	0000'0000	00	0	1st bit of command
0	0000'0000	00	0	2nd bit of command
0	0000'0000	00	0	…
0	0000'0000	00	0	
0	0000'0000	00	0	
1	0011'0001			CRC EXOR polynom
0	0110'0010			
1	**1111'0101**	**F5**	**245**	**CRC after command**
0	1101'1011			1st byte (MSB) of measurement
0	1000'0111			
0	0011'1111			
0	0111'1110			
1	1100'1101			
0	1010'1011			
0	0110'0111			
1	**1111'1111**	**FF**	**255**	**CRC value**
0	1100'1111			2nd byte (LSB) of measurement
0	1010'1111			
1	0101'1110			
1	1000'1101			
0	0010'1011			
0	0101'0110			
0	1010'1100			
1	**0101'1000**	**58**	**88**	**Final CRC value**

Example 1: readout of status register containing 0x40

Input bit's	CRC Value 0011'0001	0x	dec	Comment
	0000'0000			Start with contents of status register[1]
0	0000'0000	00	0	1st bit of command
0	0000'0000	00	0	2nd bit of command
0	0000'0000	00	0	…
0	0000'0000	00	0	
0	0000'0000	00	0	
1	0011'0001			CRC EXOR polynom
1	0101'0011			
1	**1001'0111**	**97**	**151**	**CRC after command**
0	0001'1111			1st bit (MSB) of status register
1	0000'1111			
0	0001'1110			
0	0011'1100			
0	0111'1000			
0	1111'0000			
0	1101'0001			
0	**1001'0011**	**93**	**147**	**Final CRC value**

(1) Low nibble only, whole byte reversed ($s_0s_1s_2s_3$'0000)
(2) This is different to other CRC implementations

2.2 Bytewise

With this implementation the CRC data is stored in a 256 byte lookup table.
Perform the following operations:

1. Initialize the CRC register with the value of the lower nibble of the value of the status register (reversed ($s_0s_1s_2s_3$'0000)). (default '00000000' = 0)
2. XOR each (transmitted and received) byte with the previous CRC value. The result is the new byte that you need to calculate the CRC value from.
3. Use this value as the index to the table to obtain the new CRC value.
4. Repeat from 2.) until you have passed all bytes through the process.
5. The last byte retrieved from the table is the final CRC value.
6. The CRC value retrieved from the SHTxx must be reversed (bit 0 = bit 7, bit 1=bit 6 … bit 7 = bit 0) and can then be compared to the final CRC value.(2)

2.2.1 256 byte CRC Lookup table

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**
0	49	98	83	196	245	166	151	185	136	219	234	125	76	31	46	67	114	33	16	135	182	229	212	250	203	152	169	62	15	92	109
32	**33**	**34**	**35**	**36**	**37**	**38**	**39**	**40**	**41**	**42**	**43**	**44**	**45**	**46**	**47**	**48**	**49**	**50**	**51**	**52**	**53**	**54**	**55**	**56**	**57**	**58**	**59**	**60**	**61**	**62**	**63**
134	183	228	213	66	115	32	17	63	14	93	108	251	202	153	168	197	244	167	150	1	48	99	82	124	77	30	47	184	137	218	235
64	**65**	**66**	**67**	**68**	**69**	**70**	**71**	**72**	**73**	**74**	**75**	**76**	**77**	**78**	**79**	**80**	**81**	**82**	**83**	**84**	**85**	**86**	**87**	**88**	**89**	**90**	**91**	**92**	**93**	**94**	**95**
61	12	95	110	249	200	155	170	132	181	230	215	64	113	34	19	126	79	28	45	186	139	216	233	199	246	165	148	3	50	97	80
96	**97**	**98**	**99**	**100**	**101**	**102**	**103**	**104**	**105**	**106**	**107**	**108**	**109**	**110**	**111**	**112**	**113**	**114**	**115**	**116**	**117**	**118**	**119**	**120**	**121**	**122**	**123**	**124**	**125**	**126**	**127**
187	138	217	232	127	78	29	44	2	51	96	81	198	247	164	149	248	201	154	171	60	13	94	111	65	112	35	18	133	180	231	214
128	**129**	**130**	**131**	**132**	**133**	**134**	**135**	**136**	**137**	**138**	**139**	**140**	**141**	**142**	**143**	**144**	**145**	**146**	**147**	**148**	**149**	**150**	**151**	**152**	**153**	**154**	**155**	**156**	**157**	**158**	**159**
122	75	24	41	190	143	220	237	195	242	161	144	7	54	101	84	57	8	91	106	253	204	159	174	128	177	226	211	68	117	38	23
160	**161**	**162**	**163**	**164**	**165**	**166**	**167**	**168**	**169**	**170**	**171**	**172**	**173**	**174**	**175**	**176**	**177**	**178**	**179**	**180**	**181**	**182**	**183**	**184**	**185**	**186**	**187**	**188**	**189**	**190**	**191**
252	205	158	175	56	9	90	107	69	116	39	22	129	176	227	210	191	142	221	236	123	74	25	40	6	55	100	85	194	243	160	145
192	**193**	**194**	**195**	**196**	**197**	**198**	**199**	**200**	**201**	**202**	**203**	**204**	**205**	**206**	**207**	**208**	**209**	**210**	**211**	**212**	**213**	**214**	**215**	**216**	**217**	**218**	**219**	**220**	**221**	**222**	**223**
71	118	37	20	131	178	225	208	254	207	156	173	58	11	88	105	4	53	102	87	192	241	162	147	189	140	223	238	121	72	27	42
224	**225**	**226**	**227**	**228**	**229**	**230**	**231**	**232**	**233**	**234**	**235**	**236**	**237**	**238**	**239**	**240**	**241**	**242**	**243**	**244**	**245**	**246**	**247**	**248**	**249**	**250**	**251**	**252**	**253**	**254**	**255**
193	240	163	146	5	52	103	86	120	73	26	43	188	141	222	239	130	179	224	209	70	119	36	21	59	10	89	104	255	206	157	172

(2) This is different to other CRC implementations

2.2.2 Code example for lookup table

The following procedure calculates the CRC-8. The result accumulates in the variable CRC.

```
Var
CRC : Byte;
Procedure calc_CRC(X: Byte);

Const
CRC_Table : Array[0..255] of Byte = (
0,   49, 98, 83, 196,245,166,151,185,136,219,234,125,76, 31, 46, 67, 114,33, 16,
135,182,229,212,250,203,152,169,62, 15, 92, 109,134,183,228,213,66, 115,32, 17,
63, 14, 93, 108,251,202,153,168,197,244,167,150,1,  48, 99, 82, 124,77, 30, 47,
184,137,218,235,61, 12, 95, 110,249,200,155,170,132,181,230,215,64, 113,34, 19,
126,79, 28, 45, 186,139,216,233,199,246,165,148,3,  50, 97, 80, 187,138,217,232,
127,78, 29, 44, 2,  51, 96, 81, 198,247,164,149,248,201,154,171,60, 13, 94, 111,
65, 112,35, 18, 133,180,231,214,122,75, 24, 41, 190,143,220,237,195,242,161,144,
7,  54, 101,84, 57, 8,  91, 106,253,204,159,174,128,177,226,211,68, 117,38, 23,
252,205,158,175,56, 9,  90, 107,69, 116,39, 22, 129,176,227,210,191,142,221,236,
123,74, 25, 40, 6,  55, 100,85, 194,243,160,145,71, 118,37, 20, 131,178,225,208,
254,207,156,173,58, 11, 88, 105,4,  53, 102,87, 192,241,162,147,189,140,223,238,
121,72, 27, 42, 193,240,163,146,5,  52, 103,86, 120,73, 26, 43, 188,141,222,239,
130,179,224,209,70, 119,36, 21, 59, 10, 89, 104,255,206,157,172);

Begin
CRC := CRC_Table[X xor CRC];
End;
```

3 Revision history

Date	Revision	Changes
December 30, 2001	0.9 (Preliminary)	Initial revision
February 18, 2001	1.01	
February 27, 2001	1.02	corrected bug in CRC register init. (byte must be reversed)
May 16, 2002	1.03	emphasize that command to SHTxx is also in CRC

The latest version of this document and all application notes can be found at:
www.sensirion.com/en/download/humiditysensor/SHT11.htm

Headquarters and Sales Office

SENSIRION AG
Eggbühlstr. 14
P.O. Box
CH-8052 Zürich
Switzerland

Phone: + 41 (0)1 306 40 00
Fax: + 41 (0)1 306 40 30
e-mail: info@sensirion.com
http://www.sensirion.com/

SENSIRION
THE SENSOR COMPANY

SHTxx
Humidity & Temperature
Sensmitter

Application Note
Compensation of RH non-Linearity

1 Introduction

The SHTxx devices show a small non-linearity of the humidity sensor.
This application note describes various ways to compensate it in the attached microcontroller.

2 Revision History

October 20, 2001	C2	URO	Revision 0.9 (Preliminary)
February 10, 2002	C2	URO	Revision 1.0 modified to final coefficients

3 Implementation

If the formula on page 2 of the SHT1x datasheet is to complex and therefore too computation intense, the follow calculations may provide simplified alternatives.

The examples are based on a 8 bit humidity readout. 12 bit readouts can be converted with similar formulas but with a slightly more complex calculation.

Type of calculation	Inaccuracy due to non-linearity (10-90%RH)	Complexity of calculation
linear	± 2.2% RH	Simple (8bit subtract, right shift)
2 * linear	± 0.8% RH	Quite simple (8bit multi, 16bit add/subtract)
Polynomial 2nd order	± 0.1% RH	Floating point multiplications

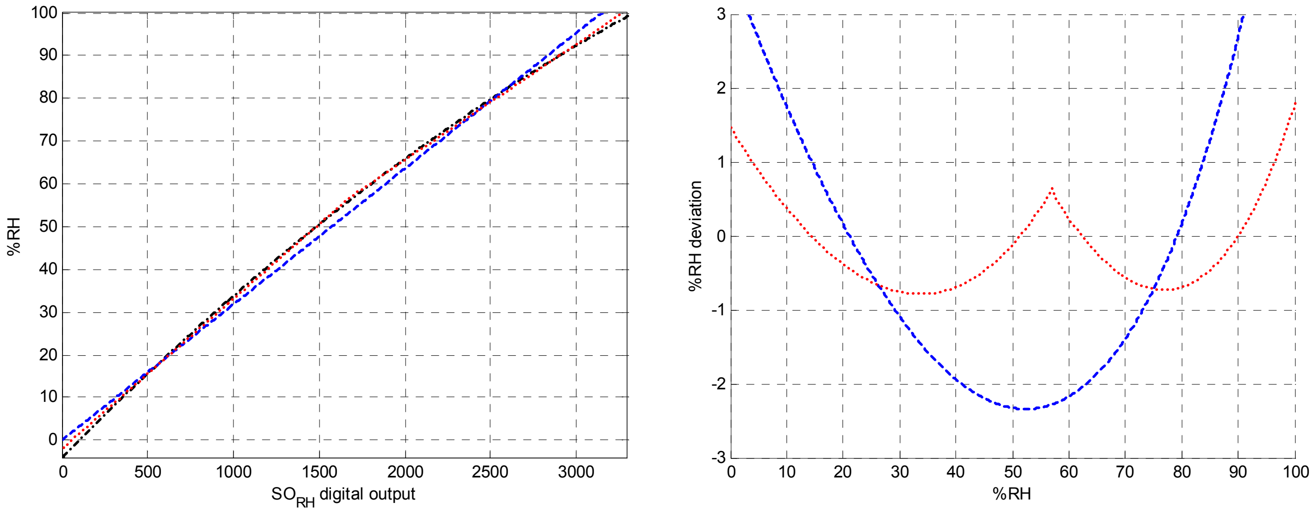


Figure 1 Inaccuracy due to non-linearity, original(from datasheet, black, dash-dotted), linear (blue, dashed) , 2* linear (red, dotted)

3.1 Linear

The most basic conversion formula from sensor output to %RH is:

$$RH_{simple} = c_1 + c_2 \bullet SO_{RH}$$

with $c_1 = 0.5$; $c_2 = 0.5$

3.2 2* linear

For improved accuracy with minimal calculation complexity the following calculation is recommended:
$RH_{real} = (a*SO + b) / 256$
Where SO denotes the 8 bit humidity sensor output signal.

Validity	a	b
$0 \le SO \le 107$	143	-512
$108 \le SO \le 255$	111	2893

With the above values the calculation can be done with a single 8 bit multiplication followed by a 16bit addition / subtraction.

Sample Code:

```
u16  result;            // 16Bit unsigned for the result
u08  sensor_out;        // 8Bit unsigned for the sensoroutput

sensor_out = readSensor8(); // read 8 bit humidity value from SHTxx

If ( sensor_out <= 107 )
{
   result = mult8Bit( 143, sensor_out );       // result = a * sensor_out
   result < 512 ? result = 512;                // check for underflow
   result = result - 512                       // result = result + b
}
else
{
   result = mult8Bit( 111, sensor_out );       // result = a * sensor_out
   result = result + 2893                      // result = result + b
   result > 25600 ? result = 25600;            // check for overflow (optional)
}

//8 MSB's are 0-100%RH integers, 8 LSB's are remainder

result = result >> 8  // result = result / 256
```

3.3 Polynomial 2nd order

Please consult the Datasheet for formula and coefficients.

Headquarters and Sales Office

SENSIRION AG	Phone:	+ 41 (0)1 306 40 00
Eggbühlstr. 14	Fax:	+ 41 (0)1 306 40 30
P.O. Box	e-mail:	info@sensirion.com
CH-8052 Zürich	http://www.sensirion.com/	
Switzerland		