

# UART Interface Description SEK-SVM40

Evaluation Kit for SGP40 – Indoor Air Quality Sensor for VOC Measurements



## Table of Contents

1 General Considerations.....	2
2 Physical Layer.....	2
3 SHDLC Frame Layer.....	3
4 UART Commands.....	6

## 1 General Considerations

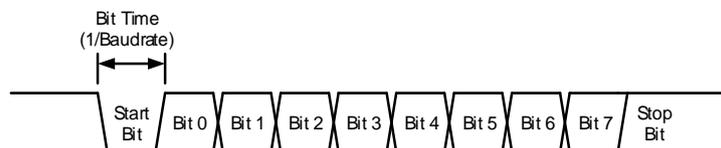
Settings as mentioned in **Table 1** have to be used when communicating with SVM40 sensor module via UART interface.

Parameter	Rating
Baud rate	115'200 bits s <sup>-1</sup>
Data bits	8
Parity	None
Stop bit	1

**Table 1** UART settings.

## 2 Physical Layer

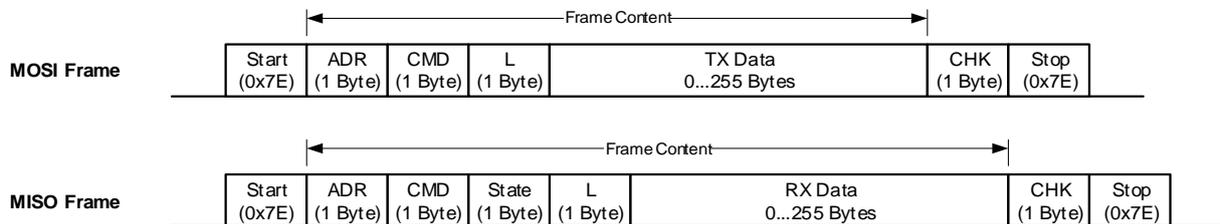
The SVM40 has separate RX and TX lines with unipolar logic levels. A transmitted byte looks as in **Figure 1**.



**Figure 1** Transmitted byte.

### 3 SHDLC Frame Layer

On top of the UART interface, the SVM40 uses the very powerful and easy-to-implement SHDLC<sup>1</sup> protocol. It is a serial communication protocol based on a master/slave architecture. The SVM40 acts as the slave device. Data is transferred in logical units called frames. Every transfer is initiated by the master sending a MOSI<sup>2</sup> frame. The slave will respond to the MOSI frame with a slave response, or MISO<sup>3</sup> frame. The two types of frames are shown in **Figure 2**.



**Figure 2** MOSI and MISO frames structure.

#### 3.1 Start/Stop Byte and Byte-Stuffing

The 0x7E character is sent at the beginning and at the end of the frame to signalize frame start and stop. If this byte 0x7E occurs anywhere else in the frame, it must be replaced by two other bytes (byte-stuffing). This also applies to the characters 0x7D, 0x11, and 0x13. Use **Table 2** for byte-stuffing.

Original data byte	Transferred data bytes
0x7E	0x7D, 0x5E
0x7D	0x7D, 0x5D
0x11	0x7D, 0x31
0x13	0x7D, 0x33

**Table 2** Reference table for byte-stuffing.

Example: Data to send = [0x43, 0x11, 0x7F] → Data transmitted = [0x43, 0x7D, 0x31, 0x7F].

#### 3.2 Address

The slave device address is always 0.

#### 3.3 Command

In the MOSI frame the command tells the device what to do with the transmitted data. In the MISO frame, the slave just returns the received command.

#### 3.4 Length

Length of the “TX Data” or “RX Data” field (before byte-stuffing).

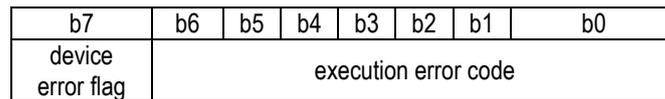
#### 3.5 State

The MISO frame contains a state byte, which allows the master to detect communication and execution errors.

<sup>1</sup> Sensirion High-Level Data Link Control.

<sup>2</sup> Master Out Slave In. Frame direction from master to slave.

<sup>3</sup> Master In Slave Out. Frame direction from slave to master.



**Figure 3** Status byte structure.

The first bit (b7) indicates that at least one of the error flags is set in the Device Status Register. The “Execution error code” signals all errors which occur while processing the frame or executing the command. The following table shows the error codes which can be reported from the device. Note that some of these errors are system internal errors which require additional knowledge to be understood. In case of a problem, they will help Sensirion to localize and solve the issue.

Error code		Description
Dec.	Hex.	
0	0x00	No error. Processing and execution of the command was successful.
1	0x01	Wrong data length for this command (too much or little data)
2	0x02	Unknown command
3	0x03	No access right for command
4	0x04	Illegal command parameter or parameter out of allowed range
40	0x28	Internal function argument out of range
67	0x43	Command not allowed in current state

**Table 3** Reference table for error codes.

### 3.6 Data

The data has a usable size of [0...255] bytes (original data before byte-stuffing). The meaning of the data content depends on the command. The data in the frames is transmitted in big-endian order (MSB first). If subcommands are used for the command the first byte will be used to define the subcommand.

### 3.7 Checksum

The checksum is built before byte-stuffing and checked after removing stuffed bytes from the frame. The checksum is defined as follows:

1. Sum all bytes between start and stop (without start and stop bytes).
2. Take the least significant byte of the result and invert it. This will be the checksum.

For a MOSI frame use Address, Command, Length, and Data to calculate the checksum.

For a MISO frame use Address, Command, State, Length, and Data to calculate the checksum.

Example: start measurement (MOSI frame without start/stop and without byte-stuffing).

ADR	CMD	L	TX data	CHK
0x02	0x43	0x04	[0x64, 0xA0, 0x22, 0xFC]	0x94

The checksum is calculated as shown in **Table 4**.

Parameter	Hex. code
ADR	0x02
CMD	0x43
L	0x04
Data 0	0x64
Data 1	0xA0
Data 2	0x22
Data 3	0xFC
Sum of all bytes	0x26B
LSB of sum	0x6B
<b>Inverted (= checksum)</b>	<b>0x94</b>

**Table 4** Checksum calculation.

## 4 UART Commands

Table 5 shows an overview of the available UART commands.

Command	Send command during
<a href="#"><u>svm40_start_measurement</u></a>	idle mode
<a href="#"><u>svm40_get_signals</u></a>	measure mode
<a href="#"><u>svm40_get_raw_signals</u></a>	measure mode
<a href="#"><u>svm40_stop_measurement</u></a>	measure mode
<a href="#"><u>svm40_get_temperature_offset</u></a>	both
<a href="#"><u>svm40_set_temperature_offset</u></a>	idle mode
<a href="#"><u>svm40_get_voc_parameters</u></a>	both
<a href="#"><u>svm40_set_voc_parameters</u></a>	idle mode
<a href="#"><u>svm40_store_input_parameters</u></a>	both
<a href="#"><u>svm40_get_voc_states</u></a>	measure mode
<a href="#"><u>svm40_set_voc_states</u></a>	idle mode
<a href="#"><u>svm40_get_version</u></a>	both
<a href="#"><u>svm40_device_reset</u></a>	both

Table 5 UART commands available for SVM40.

### 4.1 Start Measurement

<b>Command</b>	<i>svm40_start_measurement</i>	
<b>Description</b>	This command triggers the operation mode of all sensors. It must be called once prior to the <i>svm40_get_signals</i> or <i>svm40_get_raw_signals</i> commands, respectively. Per default the SVM40 starts in idle mode after powering up.	
<b>Command hex. code</b>	0x00	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x00
<b>MISO Data</b>	None	

Table 6 Description of the UART start measurement command.

Data	Hex. code
MOSI	0x7E 0x00 0x00 0x01 0x00 0xFE 0x7E
MISO	0x7E 0x00 0x00 0x00 0x00 0xFF 0x7E

**Table 7** Example frames for UART start measurement command.

## 4.2 Get Signals

<b>Command</b>	<i>svm40_get_signals</i>	
<b>Description</b>	This command reads out VOC Index, relative humidity, and temperature. It returns 6 bytes.	
<b>Command hex. code</b>	0x03	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x0A
<b>MISO Data</b>	<b>Byte number</b>	<b>Description</b>
	0, 1	int16 provides the VOC Index (no unit) with a scaling factor of 10, e.g., an output of +250 corresponds to a VOC Index of +25.0.
	2, 3	int16 provides the relative humidity (in % RH) compensated for the temperature offset with a scaling factor of 100, e.g., an output of +2'500 corresponds to +25.00 % RH.
	4, 5	int16 provides the temperature (in °C) with a scaling factor of 200, e.g., an output of +5'000 corresponds to +25.00 °C.

**Table 8** Description of the UART get signals command.

Data	Hex. code
MOSI	0x7E 0x00 0x03 0x01 0x0A 0xF1 0x7E
MISO	0x7E 0x00 0x03 0x00 0x06 0x00 0x00 0x00 0x00 0x00 0x00 0xF6 0x7E

**Table 9** Example frames for UART get signals command.

### 4.3 Get Raw Signals

<b>Command</b>	<i>svm40_get_raw_signals</i>	
<b>Description</b>	This command reads out VOC Index, relative humidity, and temperature (like <i>svm40_get_signals</i> ) and additionally the raw signal of SGP40 (proportional to the logarithm of the resistance of the MOX layer) as well as relative humidity and temperature which are not compensated for temperature offset. It returns 12 bytes.	
<b>Command hex. code</b>	0x03	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x0B
<b>MISO Data</b>	<b>Byte number</b>	<b>Description</b>
	0, 1	int16 provides the VOC Index (no unit) with a scaling factor of 10, e.g., an output of +250 corresponds to a VOC Index of +25.0.
	2, 3	int16 provides the relative humidity (in % RH) compensated for the temperature offset with a scaling factor of 100, e.g., an output of +2'500 corresponds to +25.00 % RH.
	4, 5	int16 provides the temperature (in °C) with a scaling factor of 200, e.g., an output of +5'000 corresponds to +25.00 °C.
	6, 7	uint16 directly provides the raw signal (in ticks) of the SGP40 without scaling.
	8, 9	int16 provides the uncompensated relative humidity (in % RH) with a scaling factor of 100, e.g., an output of +2'500 corresponds to +25.00 % RH.
	10, 11	int16 provides the uncompensated temperature (in °C) with a scaling factor of 200, e.g., an output of +5'000 corresponds to +25.00 °C.

**Table 10** Description of the UART get raw signals command.

<b>Data</b>	<b>Hex. code</b>
MOSI	0x7E 0x00 0x03 0x01 0x0B 0xF0 0x7E
MISO	0x7E 0x00 0x03 0x00 0x0C 0x00 0x00 0x0A 0xFE 0x16 0xBD 0x73 0x5A 0x0E 0x83 0x12 0xD5 0xD0 0x7E

**Table 11** Example frames for UART get signals command.

## 4.4 Stop Measurement

<b>Command</b>	<i>svm40_stop_measurement</i>
<b>Description</b>	This command stops the operation mode of all sensors and returns the SVM40 to idle mode.
<b>Command hex. code</b>	0x01
<b>Max. response time</b>	50 ms
<b>Post processing time</b>	0 ms
<b>MOSI Data</b>	None
<b>MISO Data</b>	None

**Table 12** Description of the UART stop measurement command.

Data	Hex. code
MOSI	0x7E 0x00 0x01 0x00 0xFE 0x7E
MISO	0x7E 0x00 0x01 0x00 0x00 0xFE 0x7E

**Table 13** Example frames for UART stop measurement command.

## 4.5 Get Temperature Offset for RHT Measurements

<b>Command</b>	<i>svm40_get_temperature_offset</i>	
<b>Description</b>	This command reads out the current temperature offset used for the compensation of RHT measurements by returning 2 bytes.	
<b>Command hex. code</b>	0x60	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x01
<b>MISO Data</b>	<b>Byte number</b>	<b>Description</b>
	0, 1	int16 provides the temperature offset (in °C) with a scaling factor of 200, e.g., an output of +400 corresponds to +2.00 °C. Default is 0 °C.

**Table 14** Description of the UART get temperature offset command.

Data	Hex. code
MOSI	0x7E 0x00 0x60 0x01 0x01 0x9D 0x7E
MISO	0x7E 0x00 0x60 0x00 0x02 0x00 0x00 0x9D 0x7E

**Table 15** Example frames for UART get temperature offset command. Result based on default settings.

#### 4.6 Set Temperature Offset for RHT Measurements

<b>Command</b>	<i>svm40_set_temperature_offset</i>	
<b>Description</b>	This command sets the temperature offset used for the compensation of subsequent RHT measurements when sent together with 2 bytes.	
<b>Command hex. code</b>	0x60	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x81
	1, 2	int16 provides the temperature offset (in °C) with a scaling factor of 200, e.g., an output of +400 corresponds to +2.00 °C. Default is 0 °C.
<b>MISO Data</b>	None	

**Table 16** Description of the UART set temperature offset command.

<b>Data</b>	<b>Hex. code</b>
MOSI	0x7E 0x00 0x60 0x05 0x81 0x00 0x00 0x00 0x00 0x19 0x7E
MISO	0x7E 0x00 0x60 0x00 0x00 0x9F 0x7E

**Table 17** Example frames for UART set temperature offset command based on default settings.

#### 4.7 Get Parameters of VOC Algorithm

<b>Command</b>	<i>svm40_get_voc_parameters</i>	
<b>Description</b>	This command reads out the current four parameters used for the VOC Algorithm returning 8 bytes.	
<b>Command hex. code</b>	0x60	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x08
<b>MISO Data</b>	<b>Byte number</b>	<b>Description</b>
	0, 1	int16 directly provides VOC Index (no unit) value representing the average conditions. Default is VOC Index = 100.
	2, 3	int16 directly provides learning time (in h) which is used by the VOC Algorithm to estimate its states from the history. Events longer than approx. twice the learning time will be forgotten. Default is 12 h.
	4, 5	int16 directly provides maximum gating duration (in min). During this period, the estimator of the VOC Algorithm states is frozen when the VOC Index is very high. Default is 180 min. 0 disables this feature.
	6, 7	int16 directly provides initial standard deviation (no unit) used during start-up of the sensor. During start-up period, a lower value boosts VOC events while a higher value decreases VOC events. Default is 50.

**Table 18** Description of the UART get voc parameters command.

<b>Data</b>	<b>Hex. code</b>
MOSI	0x7E 0x00 0x60 0x01 0x08 0x96 0x7E
MISO	0x7E 0x00 0x60 0x00 0x08 0x00 0x64 0x00 0x0C 0x00 0xB4 0x00 0x32 0x41 0x7E

**Table 19** Example frames for UART get voc parameters command returning default parameters.

#### 4.8 Set Parameters of VOC Algorithm

<b>Command</b>	<i>svm40_set_voc_parameters</i>	
<b>Description</b>	This command sets the four parameters used for the VOC Algorithm when sent together with 8 bytes.	
<b>Command hex. code</b>	0x60	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x88
	1, 2	int16 directly provides VOC Index (no unit) value representing the average conditions. Default is VOC Index = 100.
	3, 4	int16 directly provides learning time (in h) which is used by the VOC Algorithm to estimate its states from the history. Events longer than approx. twice the learning time will be forgotten. Default is 12 h.
	5, 6	int16 directly provides maximum gating duration (in min). During this period, the estimator of the VOC Algorithm states is frozen when the VOC Index is very high. Default is 180 min. 0 disables this feature.
	7, 8	int16 directly provides initial standard deviation (no unit) used during start-up of the sensor. During start-up period, a lower value boosts VOC events while a higher value decreases VOC events. Default is 50.
<b>MISO Data</b>	None	

**Table 20** Description of the UART set voc parameters command.

<b>Data</b>	<b>Hex. code</b>
MOSI	0x7E 0x00 0x60 0x09 0x88 0x00 0x64 0x00 0x0C 0x00 0xB4 0x00 0x32 0xB8 0x7E
MISO	0x7E 0x00 0x60 0x00 0x00 0x9F 0x7E

**Table 21** Example frames for UART set voc parameters command based on default parameters.

#### 4.9 Store Input Parameters to Non-Volatile Memory

<b>Command</b>	<i>svm40_store_input_parameters</i>	
<b>Description</b>	This command stores all parameters previously sent to the slave via the <i>svm40_set_temperature_offset</i> and/or the <i>svm40_set_voc_parameters</i> commands to the non-volatile memory of SVM40. These parameters will not be erased during reset and will be used by the corresponding algorithms after start-up. To reset the storage to factory settings the master has to set all parameters to the default values followed by a subsequent call of the <i>svm40_store_input_parameters</i> command.	
<b>Command hex. code</b>	0x60	
<b>Max. response time</b>	500 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x80
<b>MISO Data</b>	None	

**Table 22** Description of the UART store input parameters command.

Data	Hex. code
MOSI	0x7E 0x00 0x60 0x01 0x80 0x1E 0x7E
MISO	0x7E 0x00 0x60 0x00 0x00 0x9F 0x7E

**Table 23** Example frames for UART store input parameters command.

#### 4.10 Get States of VOC Algorithm

<b>Command</b>	<i>svm40_get_voc_states</i>	
<b>Description</b>	This command reads out the states of VOC Algorithm by returning 8 bytes. These values can be used to set the states (using the <i>svm40_set_voc_states</i> command) after resuming sensor operation, e.g., after a short interruption by skipping the initial learning phase of the VOC Algorithm.	
<b>Command hex. code</b>	0x61	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x08
<b>MISO Data</b>	<b>Byte number</b>	<b>Description</b>
	0 ... 7	uint8[8] array of 8 bytes providing the states of the VOC Algorithm.

**Table 24** Description of the UART get voc states command.

Data	Hex. code
MOSI	0x7E 0x00 0x61 0x01 0x08 0x95 0x7E
MISO	0x7E 0x00 0x61 0x00 0x08 0x00 0x00 0x00 0x00 0x00 0x32 0x00 0x00 0x64 0x7E

**Table 25** Example frames for UART get voc states command.

#### 4.11 Set States of VOC Algorithm

<b>Command</b>	<i>svm40_set_voc_states</i>	
<b>Description</b>	This command sets the states of the VOC Algorithm when sent together with 8 bytes, which were retrieved by the <i>svm40_get_voc_states</i> command before. This can be used when resuming sensor operation, e.g., after a short interruption by skipping the initial learning phase of the VOC Algorithm.	
<b>Command hex. code</b>	0x61	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 subcommand, must be set to 0x88
	1 ... 8	uint8[8] array of 8 bytes setting the states of the VOC Algorithm.
<b>MISO Data</b>	None	

**Table 26** Description of the UART set voc states command.

Data	Hex. code
MOSI	0x7E 0x00 0x61 0x09 0x88 0x00 0x00 0x00 0x00 0x00 0x32 0x00 0x00 0xDB 0x7E
MISO	0x7E 0x00 0x61 0x00 0x00 0x9E 0x7E

**Table 27** Example frames for UART set voc states command.

## 4.12 Get Version of Device

<b>Command</b>	<i>svm40_get_version</i>	
<b>Description</b>	This command returns information on the hardware, firmware, and protocol in 7 bytes.	
<b>Command hex. code</b>	0xD1	
<b>Max. response time</b>	50 ms	
<b>Post processing time</b>	0 ms	
<b>MOSI Data</b>	None	
<b>MISO Data</b>	<b>Byte number</b>	<b>Description</b>
	0	uint8 provides the major version number of the firmware.
	1	uint8 provides the minor version number of the firmware.
	2	bool provides the debug state of the firmware.
	3	uint8 provides the major version number of the hardware.
	4	uint8 provides the minor version number of the hardware.
	5	uint8 provides the major version number of the protocol.
	6	uint8 provides the minor version number of the protocol.

**Table 28** Description of the UART get version command.

<b>Data</b>	<b>Hex. code</b>
MOSI	0x7E 0x00 0xD1 0x00 0x2E 0x7E
MISO	0x7E 0x00 0xD1 0x00 0x07 0x02 0x02 0x00 0x02 0x00 0x01 0x00 0x20 0x7E

**Table 29** Example frames for UART get version command.

## 4.13 Device Reset

<b>Command</b>	<i>svm40_device_reset</i>
<b>Description</b>	This command performs a reset of the device and restarts the SVM40 in idle mode. Prior to executing the reset, the device will acknowledge the call. If the command is sent with broadcast, the reset is executed directly after the reception of the command. All previously set parameters sent by <i>svm40_set_temperature_offset</i> , <i>svm40_set_voc_parameters</i> , and <i>svm40_set_states</i> commands will be lost. The temperature offset and the VOC Algorithm parameters can be stored to the non-volatile memory of SVM40 by calling the <i>svm40_store_input_parameters</i> command.
<b>Command hex. code</b>	0xD3
<b>Max. response time</b>	50 ms
<b>Post processing time</b>	100 ms
<b>MOSI Data</b>	None
<b>MISO Data</b>	None

**Table 30** Description of the UART device reset command.

Data	Hex. code
MOSI	0x7E 0x00 0xD3 0x00 0x2C 0x7E
MISO	0x7E 0x00 0xD3 0x00 0x00 0x2C 0x7E

**Table 31** Example frames for UART device reset command.

## Revision History

Date	Version	Page(s)	Changes
December, 2020	1.0	All	Initial release
March, 2021	1.1	14	Amended typo

## Headquarters and Subsidiaries

### **Sensirion AG**

Laubisruetistr. 50  
CH-8712 Staefa ZH  
Switzerland

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

### **Sensirion Taiwan Co. Ltd**

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

### **Sensirion Inc., USA**

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

### **Sensirion Japan Co. Ltd.**

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

### **Sensirion Korea Co. Ltd.**

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

### **Sensirion China Co. Ltd.**

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

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