

Arduino BLE Gadgets

BLE Communication Protocol

Table of Contents

1	BLE Services	2
1.1	Device Information Service.....	2
1.2	Data Logger Service	2
2	BLE Advertisement	4
3	Sensirion Sample Types.....	4
3.1	Sample type 0.....	5
3.2	Sample type 1.....	5
3.3	Sample type 2.....	5
3.4	Sample type 3.....	5
3.5	Sample type 4.....	6
3.6	Sample type 5.....	6
3.7	Sample type 6.....	6
3.8	Sample type 7.....	7
3.9	Sample type 8.....	7
3.10	Sample type 9.....	7
3.11	Sample type 10.....	8
3.12	Sample type 11.....	8
3.13	Sample type 12.....	8
3.14	Sample type 13.....	9
3.15	Sample type 14.....	9
3.16	Sample type 15.....	9
3.17	Sample type 16.....	10
3.18	Sample type 19.....	10
3.19	Sample type 20.....	10
3.20	Sample type 21.....	11
3.21	Sample type 22.....	11
3.22	Sample type 23.....	11
3.23	Sample type 24.....	12
3.24	Sample type 25.....	12
3.25	Sample type 26.....	12
3.26	Sample type 27.....	13
3.27	Sample type 28.....	13
3.28	Sample type 29.....	14
3.29	Sample type 30.....	14
3.30	Sample type 31.....	14
3.31	Sample type 32.....	15
3.32	Sample type 33.....	15
3.33	Sample type 34.....	15
3.34	Sample type 35.....	15
3.35	Sample type 36.....	16

1 BLE Services

The full characteristics UUID can be obtained by replacing the bold part of the service UUID with the given short UUID from the characteristics table.

1.1 Device Information Service

Service UUID 0000**180a**-0000-1000-8000-00805f9b34fb

Characteristics:

Description	UUID	Type	Expected value/format	Read	Write	Notify
Manufacturer name	2a29	String	"Sensirion"	●		
Model number	2a24	String	Device specific	●		
Serial number	2a25	String	Last 2 Bytes of BLE MAC address	●		
Firmware revision	2a26	String	"x.x.x"	●		
System ID	2a23	Bytes	BLE MAC address	●		

1.2 Battery Service

Service UUID 0000**180f**-0000-1000-8000-00805f9b34fb

Characteristics:

Description	UUID	Type	unit	Read	Write	Notify
Battery level	2a19	uint	%	●		

1.3 Data Logger Service

Service UUID 0000**8000**-B38D-4985-720E-0F993A68EE41

Characteristics:

Description	UUID	unit	Type	Read	Write	Notify
Logging interval	8001	ms	UInt32	●		
Available samples	8002	-	UInt16	●		
Requested samples	8003	-	UInt16	●	●	
Data transfer	8004	-	Byte(20)			●

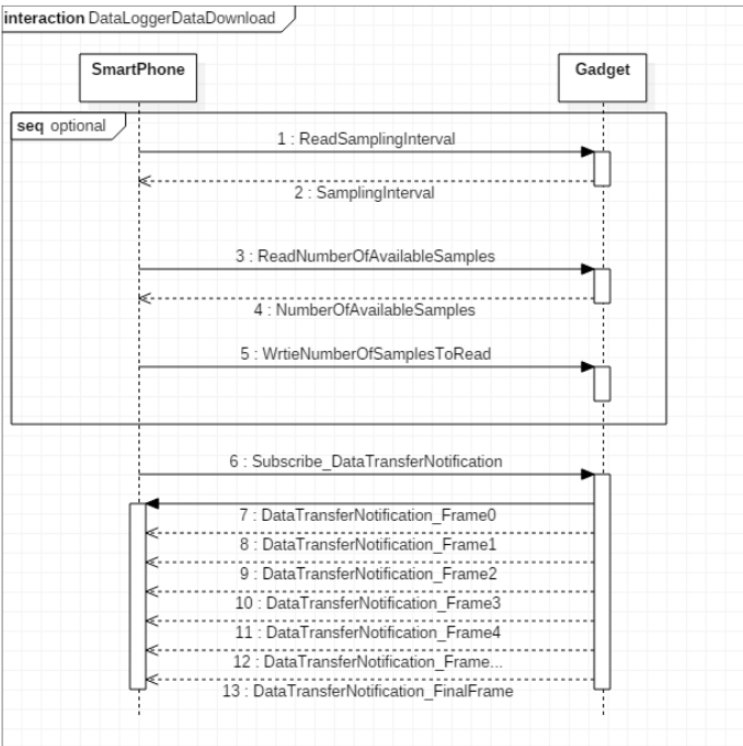
Usage Example

The steps 1 to 5 are optional, they are only required if not all samples should be transferred by the Data Logger. This is useful if older samples have already been read from the logger and only the missing new samples are to be transferred.

1. To be able to calculate how many samples we have missed; the sampling interval has to be readout.
2. The gadget returns the sampling interval in milliseconds.
3. Then we check how many samples are available at all.
4. The gadget returns the number of available samples.
5. Based on the retrieved information, we determine how many samples we want to read. If the requested number of samples is higher than the available number of samples, then all available samples are transmitted.

If the requested number of samples is less than the available number of samples, the oldest samples are omitted. If the requested number of samples is not specified, then all available samples are transmitted.

6. To start the data transfer, the Data Transfer notification must be subscribed.
7. The gadget starts sending notifications. The first notification contains the header with the information how the sampling data are structured. The following frames contains the sampling data.



Interaction with data logger service

Complete Data Logger Data Frame

Frame0: Header																			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
20 Byte Frame																			
Sequence Number	18 Byte Header																		
	16 Byte SampleTransferProtocol-Header																		
	Version	Protocol	Sample Type	Sampling Interval in ms			Age of the latest sample in ms			Number of Samples	unused								
0x0000	0x00	0x00	0x0009	0xXXXXXXXX			0XXXXXXXXX			0XXXXX	0x00	0x00		0x00	0x00				
Frame1: oldest samples																			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
20 Byte Frame																			
Sequence Number	18 Byte Data																		
	Sample: 0 (oldest)						Sample: 1						Sample: 2						
0x0001	Temperature-Ticks		Humidity-Ticks		CO2-Value		Temperature-Ticks		Humidity-Ticks		CO2-Value		Temperature-Ticks		Humidity-Ticks		CO2-Value		
Frame2: next samples																			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
20 Byte Frame																			
Sequence Number	18 Byte Data																		
	Sample: 3						Sample: 4						Sample: 5						
0x0002	Temperature-Ticks		Humidity-Ticks		CO2-Value		Temperature-Ticks		Humidity-Ticks		CO2-Value		Temperature-Ticks		Humidity-Ticks		CO2-Value		
Frame 3, 4, 5,...																			
FinalFrame: latest samples																			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
20 Byte Frame																			
Sequence Number	18 Byte Data																		
	Sample: [Number of Samples] - 1						unused												
0xXXXX	Temperature-Ticks		Humidity-Ticks		CO2-Value		0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Example 1: Example packets sent by a data logger service using sample type 9

For a complete list of the used sample types see section 3.

2 BLE Advertisement

We provide the newest sensor data in the manufacturer specific data of the advertising packet (see *Advertising Data Element: Manufacturer Specific Data* in the following data frame)

We also add the advertised name to the Complete Local Name part of the advertising packet (see *Advertising Data Element: Complete Local Name* in the following data frame)

Sensirion SCD4x CO2 Gadget																																																			
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																					
Adv. Data Element: Flags			Advertising Data Element: Manufacturer Specific Data														Advertising Data Element: Complete Local Name																																		
Length	AdvType	Data	Length	AdvType	Data												Length	AdvType	Data																																
0x02	0x01	0x06	0x0F	0xFF	Company Identifier												Manufacturer Data																																		
					0x0605				S. AdvT.		S. Type		Sample				0x0605				ASCII(M)		ASCII(Y)		ASCII(C)		ASCII(O)		ASCII(Z)																						
					0x06				0x05		0x00		0x08		Device ID		Temperature-Ticks		Humidity-Ticks		CO2		0x06		0x09		0x40		0x79		0x43		0x4F		0x32																
																								unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused

Sensirion SHT4x SmartGadget																																																							
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																									
Adv. Data Element: Flags			Advertising Data Element: Manufacturer Specific Data														Advertising Data Element: Complete Local Name																																						
Length	AdvType	Data	Length	AdvType	Data												Length	AdvType	Data																																				
0x02	0x01	0x06	0x0F	0xFF	Company Identifier												Manufacturer Data																																						
					0x0605				S. AdvT.		S. Type		Sample				0x0605				ASCII(S)		ASCII(m)		ASCII(a)		ASCII(Y)		ASCII(G)		ASCII(a)		ASCII(d)		ASCII(e)		ASCII(v)		ASCII(T)																
					0x06				0x05		0x00		0x08		Device ID		Temperature-Ticks		Humidity-Ticks		0x0C		0x09		0x53		0x6D		0x61		0x72		0x74		0x47		0x61		0x54		0x67		0x65		0x74										
																								unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused

DIY SFA30 BLE gadget																																																																																																																																																																																																																																																																																																																																																																																																																																					
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																																																																																																																																																																																																																																																																																																																																																																																																							
Adv. Data Element: Flags			Advertising Data Element: Manufacturer Specific Data														Advertising Data Element: Complete Local Name																																																																																																																																																																																																																																																																																																																																																																																																																				
Length	AdvType	Data	Length	AdvType	Data												Length	AdvType	Data																																																																																																																																																																																																																																																																																																																																																																																																																		
0x02	0x01	0x06	0x0F	0xFF	Company Identifier												Manufacturer Data																																																																																																																																																																																																																																																																																																																																																																																																																				
					0x0605				S. AdvT.		S. Type		Sample				0x0605				ASCII(S)		ASCII(m)		ASCII(a)		ASCII(Y)		ASCII(G)		ASCII(a)		ASCII(d)		ASCII(e)		ASCII(v)		ASCII(T)																																																																																																																																																																																																																																																																																																																																																																																														
					0x06				0x05		0x00		0x0E		Device ID		Temperature-Ticks		Humidity-Ticks		HCHO ticks		0x02		0x09		0x53		unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused	unused

Complete advertisement packets as example

3 Sensirion Sample Types

Sensirion BLE devices are using 2 different types of samples. Some of them are used by the data logger service, while the remaining are used by the BLE advertisement.

The data logger sample types include 2 bytes to identify it, while the ones used for advertisement have 1 byte describing the sample advertisement type (always 0x00) and a second byte to identify the sample type itself

To identify the sample type used by an example DIY gadget you can look in the code for the *DataProvider* definition:

```
DataProvider provider(lib, DataType::T_RH_VOC);
```

This line defines the data provider and the type of data it should expect. In this case: **T_H_VOC**.

The mapping to a sample type can be found in the *config.h* file of the *Arduino-ble-gadget* library. In this example we can see the following:

```
{T_RH_VOC,
  { .dataType = DataType::T_RH_VOC,
    .downloadType = 1,
    .sampleType = 3,
    .sampleSizeBytes = 6,
    .sampleCountPerPacket = 3,
```

Here we learn that the data type, used in this example, uses the sample type 1 for the download service (data logger service) and the sample type 3 for advertisement.

3.1 Sample type 0

Used by: Data logger service
 Measured values: Temperature, Humidity
 Devices using it: **SHT3x** based gadgets

Sample type		Sample data			
0	1	0	1	2	3
0x0000		Temperature ticks		Humidity ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$

3.2 Sample type 1

Used by: Data logger service
 Measured values: Temperature, Humidity, VOC
 Devices using it: **SHT3x** based gadgets

Sample type		Sample data					
0	1	0	1	2	3	4	5
0x0001		Temperature ticks		Humidity ticks		TVOC	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value

3.3 Sample type 2

Used by: Data logger service
 Measured values: Temperature, Humidity, VOC, VOC raw
 Devices using it: **AQ Minion** gadgets (based on SHT3x)

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x0002		Temperature ticks		Humidity ticks		TVOC		TVOC-raw	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- VOC-Raw = transmitted value

3.4 Sample type 3

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, VOC, VOC raw
 Devices using it: **AQ Minion** gadgets (based on SHT3x)

S. Adv type	S. type	Sample data									
		0	1	2	3	4	5	6	7	8	9
0x00	0x03	Device ID		Temperature ticks		Humidity ticks		TVOC		TVOC-raw	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- VOC-Raw = transmitted value

3.5 Sample type 4

Used by: BLE Advertisement
 Measured values: Temperature, Humidity
 Devices using it: **SHT3x** based gadgets

S. Adv type	S. type	Sample data					
		0	1	2	3	4	5
0x00	0x04	Device ID		Temperature ticks		Humidity ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$

3.6 Sample type 5

Used by: Data logger service
 Measured values: Temperature, Humidity
 Devices using it: **SHT4x** based gadgets

Sample type		Sample data			
0	1	0	1	2	3
0x0005		Temperature ticks		Humidity ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = -6 + (125.0 * \text{ticks}) / (2^{16} - 1)$

3.7 Sample type 6

Used by: BLE Advertisement
 Measured values: Temperature, Humidity
 Devices using it: **SHT4x** based gadgets

S. Adv type	S. type	Sample data					
		0	1	2	3	4	5
0x00	0x06	Device ID		Temperature ticks		Humidity ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$

- $RH = -6 + (125.0 * ticks) / (2^{16} - 1)$

3.8 Sample type 7

Used by: Data logger service
 Measured values: Temperature, Humidity, CO2
 Devices using it: Sensirion MyCO2

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x0007		Temperature ticks		Humidity ticks		CO2		<i>reserved</i>	

Conversion:

- $T = -45 + ((175.0 * ticks) / (2^{16} - 1))$
- $RH = (100.0 * ticks) / (2^{16} - 1)$
- CO2 = transmitted value

3.9 Sample type 8

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, CO2
 Devices using it: Sensirion MyCO2

		Sample data									
S. Adv type	S. type	0	1	2	3	4	5	6	7	8	9
0x00	0x08	Device ID		Temperature ticks		Humidity ticks		CO2		<i>reserved</i>	

Conversion:

- $T = -45 + ((175.0 * ticks) / (2^{16} - 1))$
- $RH = (100.0 * ticks) / (2^{16} - 1)$
- CO2 = transmitted value

3.10 Sample type 9

Used by: Data logger service
 Measured values: Temperature, Humidity, CO2
 Devices using it:

Sample type		Sample data					
0	1	0	1	2	3	4	5
0x0002		Temperature ticks		Humidity ticks		CO2	

Conversion:

- $T = -45 + ((175.0 * ticks) / (2^{16} - 1))$
- $RH = (100.0 * ticks) / (2^{16} - 1)$
- CO2 = transmitted value

3.11 Sample type 10

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, CO2
 Devices using it:

S. Adv type	S. type	Sample data							
		0	1	2	3	4	5	6	7
0x00	0x0A	Device ID		Temperature ticks		Humidity ticks		CO2	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value

3.12 Sample type 11

Used by: Data logger service
 Measured values: Temperature, Humidity, CO2, PM2.5
 Devices using it: Environmental Gadget (SHT3x based)

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x000B		Temperature ticks		Humidity ticks		CO2		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- $PM2.5 = (1000.0 * \text{ticks}) / (2^{16} - 1)$

3.13 Sample type 12

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, CO2, PM2.5
 Devices using it: Environmental Gadget (SHT3x based)

S. Adv type	S. type	Sample data									
		0	1	2	3	4	5	6	7	8	9
0x00	0x0C	Device ID		Temperature ticks		Humidity ticks		CO2		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value

- PM2.5 = $(1000.0 * \text{ticks}) / (2^{16} - 1)$

3.14 Sample type 13

Used by: Data logger service
 Measured values: Temperature, Humidity, Formaldehyde
 Devices using it:

Sample type		Sample data					
0	1	0	1	2	3	4	5
0x000D		Temperature ticks		Humidity ticks		HCHO ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- $HCHO = \text{ticks} / 5$

3.15 Sample type 14

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, Formaldehyde
 Devices using it:

		Sample data							
S. Adv type	S. type	0	1	2	3	4	5	6	7
0x00	0x0E	Device ID		Temperature ticks		Humidity ticks		HCHO ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- $HCHO = \text{ticks} / 5$

3.16 Sample type 15

Used by: Data logger service
 Measured values: Temperature, Humidity, VOC, PM2.5
 Devices using it: SEN Gadget

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x000F		Temperature ticks		Humidity ticks		VOC		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- $PM2.5 = (1000.0 * \text{ticks}) / (2^{16} - 1)$

3.17 Sample type 16

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, VOC, PM2.5
 Devices using it: SEN Gadget

S. Adv type	S. type	Sample data									
		0	1	2	3	4	5	6	7	8	9
0x00	0x0C	Device ID		Temperature ticks		Humidity ticks		VOC		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- $PM2.5 = (1000.0 * \text{ticks}) / (2^{16} - 1)$

3.18 Sample type 19

Used by: Data logger service
 Measured values: Temperature, Humidity, CO2, VOC, PM2.5, Formaldehyde
 Devices using it: Monstera Deliciosa gadget

S. type	Sample data											
	0	1	2	3	4	5	6	7	8	9	10	11
0x0013	Temperature ticks		Humidity ticks		CO2		VOC		PM2.5 ticks		HCHO ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- VOC = transmitted value
- $PM2.5 = (1000.0 * \text{ticks}) / (2^{16} - 1)$
- $HCHO = \text{ticks} / 5$

3.19 Sample type 20

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, CO2, VOC, PM2.5, Formaldehyde
 Devices using it: Monstera Deliciosa gadget

S. Adv type	S. type	Sample data													
		0	1	2	3	4	5	6	7	8	9	10	11	12	13
0x00	0x14	Device ID		Temperature ticks		Humidity ticks		CO2		VOC		PM2.5 ticks		HCHO ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$

- CO2 = transmitted value
- VOC = transmitted value
- PM2.5 = (1000.0 * ticks) / (2¹⁶ - 1)
- HCHO = ticks / 5

3.20 Sample type 21

Used by: Data logger service
 Measured values: Temperature, Humidity, VOC, NOx
 Devices using it: SVM41 DIY Gadget

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x0015		Temperature ticks		Humidity ticks		VOC		NOx	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- NOx = transmitted value

3.21 Sample type 22

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, VOC, NOx
 Devices using it: SVM41 DIY Gadget

S. Adv type	S. type	Sample data									
		0	1	2	3	4	5	6	7	8	9
0x00	0x16	Device ID		Temperature ticks		Humidity ticks		VOC		NOx	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- NOx = transmitted value

3.22 Sample type 23

Used by: Data logger service
 Measured values: Temperature, Humidity, VOC, NOx, PM2.5
 Devices using it: SEN55 DIY Gadget

Sample type		Sample data									
0	1	0	1	2	3	4	5	6	7	8	9
0x0017		Temperature ticks		Humidity ticks		VOC		NOx		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$

- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- NOx = transmitted value
- $PM2.5 = \text{ticks} / 10$

3.23 Sample type 24

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, VOC, NOx, PM2.5
 Devices using it: SEN55 DIY Gadget

S. Adv type	S. type	Sample data											
		0	1	2	3	4	5	6	7	8	9	10	11
0x00	0x18	Device ID		Temperature ticks		Humidity ticks		VOC		NOx		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- NOx = transmitted value
- $PM2.5 = \text{ticks} / 10$

3.24 Sample type 25

Used by: Data logger service
 Measured values: Temperature, Humidity, CO2, VOC, NOx, PM2.5
 Devices using it:

S. type		Sample data											
0	1	0	1	2	3	4	5	6	7	8	9	10	11
0x0019		Temperature ticks		Humidity ticks		CO2		VOC		NOx		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- VOC = transmitted value
- NOx = transmitted value
- $PM2.5 = \text{ticks} / 10.0$

3.25 Sample type 26

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, CO2, VOC, NOx, PM2.5
 Devices using it:

S. Adv type	S. type	Sample data													
		0	1	2	3	4	5	6	7	8	9	10	11	12	13
0x00	0x1A	Device ID		Temperature ticks		Humidity ticks		CO2		VOC		NOx		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- VOC = transmitted value
- NOx = transmitted value
- PM2.5 = ticks / 10.0

3.26 Sample type 27

Used by: Data logger service
 Measured values: Temperature, Humidity, CO2, PM2.5
 Devices using it: Environmental Gadget (SHT3x based)

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x001B		Temperature ticks		Humidity ticks		CO2		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- PM2.5 = ticks / 10.0

3.27 Sample type 28

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, CO2, PM2.5
 Devices using it: Environmental Gadget (SHT3x based)

S. Adv type	S. type	Sample data									
		0	1	2	3	4	5	6	7	8	9
0x00	0x1C	Device ID		Temperature ticks		Humidity ticks		CO2		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- PM2.5 = ticks / 10.0

3.28 Sample type 29

Used by: Data logger service
 Measured values: Temperature, Humidity, VOC, PM2.5
 Devices using it: SEN Gadget

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x001D		Temperature ticks		Humidity ticks		VOC		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- $PM2.5 = \text{ticks} / 10.0$

3.29 Sample type 30

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, VOC, PM2.5
 Devices using it: SEN Gadget

		Sample data									
S. Adv type	S. type	0	1	2	3	4	5	6	7	8	9
0x00	0x1E	Device ID		Temperature ticks		Humidity ticks		VOC		PM2.5 ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- VOC = transmitted value
- $PM2.5 = \text{ticks} / 10.0$

3.30 Sample type 31

Used by: Data logger service
 Measured values: Temperature, Humidity, CO2, VOC, PM2.5, Formaldehyde
 Devices using it: Monstera Deliciosa gadget

S. type		Sample data											
0	1	0	1	2	3	4	5	6	7	8	9	10	11
0x001F		Temperature ticks		Humidity ticks		CO2		VOC		PM2.5 ticks		HCHO ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- VOC = transmitted value
- $PM2.5 = \text{ticks} / 10.0$
- $HCHO = \text{ticks} / 5.0$

3.31 Sample type 32

Used by: BLE Advertisement
 Measured values: Temperature, Humidity, CO2, VOC, PM2.5, Formaldehyde
 Devices using it: Monstera Deliciosa gadget

		Sample data													
S. Adv type	S. type	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0x00	0x20	Device ID		Temperature ticks		Humidity ticks		CO2		VOC		PM2.5 ticks		HCHO ticks	

Conversion:

- $T = -45 + ((175.0 * \text{ticks}) / (2^{16} - 1))$
- $RH = (100.0 * \text{ticks}) / (2^{16} - 1)$
- CO2 = transmitted value
- VOC = transmitted value
- PM2.5 = ticks / 10.0
- HCHO = ticks / 5.0

3.32 Sample type 33

Used by: Data logger service
 Measured values: PM1.0, PM2.5, PM4.0, PM10
 Devices using it: SEN50-based gadgets sending only PM values

Sample type		Sample data							
0	1	0	1	2	3	4	5	6	7
0x0021		PM1.0 ticks		PM2.5 ticks		PM4.0 ticks		PM10 ticks	

Conversion:

- PM = ticks / 10.0

3.33 Sample type 34

Used by: BLE Advertisement
 Measured values: PM1.0, PM2.5, PM4.0, PM10
 Devices using it: SEN50-based gadgets sending only PM values

S. Adv type	S. type	Sample data									
		0	1	2	3	4	5	6	7	8	9
0x00	0x22	Device ID		PM1.0 ticks		PM2.5 ticks		PM4.0 ticks		PM10 ticks	

Conversion:

- PM = ticks / 10.0

3.34 Sample type 35

Used by: Data logger service
 Measured values: CO2
 Devices using it: CO2 gadgets

Sample type		Sample data	
0	1	0	1
0x0023		CO2	

Conversion:

- CO2 = transmitted value

3.35 Sample type 36

Used by: BLE Advertisement

Measured values: CO2

Devices using it: CO2 gadgets

		Sample data			
S. Adv type	S. type	0	1	2	3
0x00	0x04	Device ID		CO2	

Conversion:

- CO2 = transmitted value