CENELEC/WS NEST

Date: **2024-02**

prCWA XXXX:2024

Secretariat: UNE

Open protocol for CBRN sensor connectivity

Document type:
Document subtype:
Document stage:
Document language:

COVER PAGE

Contents

European foreword

This CEN/CENELEC Workshop Agreement has been developed in accordance with the CEN-CENELEC Guide 29 “CEN/CENELEC Workshop Agreements – A rapid prototyping to standardization” and with the relevant provisions of CEN/CENELEC Internal Regulations - Part 2. It was approved by a Workshop of representatives of interested parties on YYYY-MM-DD, the constitution of which was supported by CEN/CENELEC following the public call for participation made on 2023-03-23. However, this CEN/CENELEC Workshop Agreement does not necessarily include all relevant stakeholders.

The final text of this CEN/CENELEC Workshop Agreement was provided to CEN/CENELEC for publication on 2024-MM-DD.

Results incorporated in this CWA received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101018596.

The following organizations and individuals developed and approved this CEN/CENELEC Workshop Agreement:

- AMPER S & C IOT SL (S&C)/Alberto Fernández
- AMPER S & C IOT SL (S&C)/Narcis Avellana
- AMPER S & C IOT SL (S&C)/Adrián Cañadas
- AGENCIA ESTATAL CONSEJO SUPERIOR DE INVESTIGACIONES CIENTIFICAS (CSIC)/Manuel Lozano
- AGENCIA ESTATAL CONSEJO SUPERIOR DE INVESTIGACIONES CIENTIFICAS (CSIC)/Celeste Fleta
- AGENCIA ESTATAL CONSEJO SUPERIOR DE INVESTIGACIONES CIENTIFICAS (CSIC)/Esteve Amat
- AGENCIA ESTATAL CONSEJO SUPERIOR DE INVESTIGACIONES CIENTIFICAS (CSIC)/Sara Degli-Esposti
- WOEPAL GmbH (WOEPAL GmbH)/Natalie Schuetz, Chair
- WOEPAL GmbH (WOEPAL GmbH)/Moritz Zimmer
- WOEPAL GmbH (WOEPAL GmbH)/Stefan Palzer
- Asociación Española de Normalización (UNE)/Ana Mariblanca, Secretary
- TU Dortmund University/Álvaro Ortiz
- The Fraunhofer Institute for Communication, Information Processing and Ergonomics (FKIE)/ Frank Schneider
- The Fraunhofer Institute for Communication, Information Processing and Ergonomics (FKIE)/ Alexander Tiderko

- Agenzia nazionale per le nuove tecnologie, l'energia e lo sviluppo economico sostenibile (ENEA)/Roberto Chirico, RISEN (risen-h2020.eu)
- Particle/ Marco Manso, RISEN (risen-h2020.eu)
- Deutsches Institut für Normung (DIN)/Yusuf Yilmaz, RISEN (risen-h2020.eu)
- Deutsches Institut für Normung (DIN)/ Ulrike Schröder RISEN (risen-h2020.eu)

Attention is drawn to the possibility that some elements of this document may be subject to patent rights. CEN-CENELEC policy on patent rights is described in CEN-CENELEC Guide 8 “Guidelines for Implementation of the Common IPR Policy on Patent”. CEN/CENELEC shall not be held responsible for identifying any or all such patent rights.

Although the Workshop parties have made every effort to ensure the reliability and accuracy of technical and nontechnical descriptions, the Workshop is not able to guarantee, explicitly or implicitly, the correctness of this document. Anyone who applies this CEN/CENELEC Workshop Agreement shall be aware that neither the Workshop, nor CEN, can be held liable for damages or losses of any kind whatsoever. The use of this CEN/CENELEC Workshop Agreement does not relieve users of their responsibility for their own actions, and they apply this document at their own risk. The CEN/CENELEC Workshop Agreement should not be construed as legal advice authoritatively endorsed by CEN/CENELEC.

Introduction

NEST, an iNteropErable multidomain CBRN SysTem, is a research project founded by Research Executive Agency of the European Commission under Grant Agreement number 101018596.

NEST will design and implement a novel and unique standoff system with the capability to detect multiple threats amongst which are CBRN threats or pandemic viruses. As the day-to-day protection of commercial and transport facilities is the responsibility of the owners and operators, in close cooperation with local law enforcement, NEST will support owners, operators, and security staff by providing threat indications and warnings, and guidance for facility security by developing appropriate information-sharing and analysis mechanisms.

The system will rely on the simultaneous use of low-cost CBRN detectors embedded in one unique detection equipment. NEST will help in the early detection of CBRN threats in real time and provide complementary information useful for auditing or investigation purposes. These functionalities will be achieved by using an IoT platform capable of acquiring, processing, and merging data from internal and third-party services. Artificial intelligence will be applied to support decision process for securing facilities and for generating automatic alerts.

The existing and under-development standards relevant for the project were identified in the very early stages of the project to be considered during the project works. The relevant standardisation committees were addressed to inform their officers of NEST objectives and to evaluate and follow the best way to promote the transfer of the results of the project to future standards. However, there was neither existing no project dealing with the requirements for an open global detection system. Then the motivation for this project was to address this issue. Designers, manufacturers, integrators, and service providers will benefit of this CWA, because it would enable the compatibility of any kind of sensor with the system, in terms of connection, data transmission, mapping, communication protocols, etc.

Today, there exists an extensive number of detection systems, developed all over the world. Each system is specified for the detection of one or several agents and may be part of a larger system that integrates the information from all individual systems, processes the data and manages the visualization, notification, and operation. However, there are sometimes interoperability constraints, and some of the would-be technically optimal subsystems are incompatible with the overall system. This is a handicap both for the system designer, who cannot have the most efficient resources available, and for the manufacturers of sensors and detection systems, who cannot compete in certain projects because of these incompatibility issues. This document solves these interoperability constraints regarding the area of CBRN.

1 Scope

This document defines requirements for an open global system, that accepts any kind of detection subsystems. It will include sensor connection, data transmission, data management and compatibility requirements.

ISO/IEC/IECEE 21451-2:2010 applies to the connection and communication between smart transducer interfaces and a network capable of application processor, but it does not cover the communication via I^2C. This CWA is specially developed for the communication via I^2C.

This CWA lays the foundations:

- for the hardware-specific requirements for connecting CBRN sensors to a specially developed central communication unit, and
- for the digital communication flow between the sensors and this unit.

The intended users of this document are designers, manufacturers, integrators, and service providers from private and public companies.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 61076-2-106:2011 *Connectors for electronic equipment - Product requirements - Part 2-106: Circular connectors - Detail specification for connectors M 16 x 0,75 with screw-locking and degree of protection IP40 or IP65/67*

ETSI EN 300 220-1:2017 *Short Range Devices (SRD) operating in the frequency range 25 MHz to 1 000 MHz; Part 1: Technical characteristics and methods of measurement*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at https://www.iso.org/obp/

3.1
Sensor or sensor module
A device that measures one or more properties of one or more physical entities and outputs digital data that are transmitted via I^2C.

[SOURCE: IEV 741-02-09, modified]

3.2
Sensor HUB or HUB
A central functional interface that coordinates the data sent by the sensors and forwards it to the middleware.

[SOURCE: IEV 732-01-21, modified]

3.3
Middleware
Application-neutral computer programs providing services to application software in a way that the complexity of these application software and their infrastructure remains hidden.

Note 1 to entry: The term "middleware" also denotes a distribution platform, i.e. a protocol at a higher level than that of ordinary computer communication.

Note 2 to entry: In contrast to lower-level network services handling simple communication between computers, middleware supports the communication between processes.

Note 3 to entry: In the AAL context, AAL middleware provides components and services for AAL applications.

[SOURCE: IEV 871-05-09]

3.4
Gateway
Functional unit that connects two computer networks with different network architectures and protocols.

Note 1 – The computer networks may be local area networks, wide area networks, or other types of networks.

Note 2 – Examples of gateways are a LAN gateway, a mail gateway.

[SOURCE: IEV 732-01-17]

3.5
LoRa
A wireless modulation technique derived from Chirp Spread Spectrum (CSS) technology. It encodes information on radio waves using chirp pulses. [1]

3.6
LoRaWAN
A Media Access Control (MAC) layer protocol built on top of LoRa modulation. It is a software layer which defines how devices use the LoRa hardware, for example when they transmit, and the format of messages. [1]

3.7
Degree of protection of enclosure
IP
Numerical classification preceded by the symbol IP applied to the enclosure of equipment to provide:

protection of persons against contact with, or approach to, live parts and against contact with moving parts (other than smooth rotating shafts and the like) inside the enclosure,

protection of the equipment against ingress of solid foreign objects, and

where indicated by the classification, protection of the equipment against harmful ingress of water

Note 1 to entry: The conditions for other than rotating machines are specified in IEC 60529, Degrees of protection provided by enclosures (IP Code).

Note 2 to entry: The detailed test requirements for rotating electric machines are in IEC 60034-5, Rotating electrical machines – Part 5: Degrees of protection provided by the integral design of rotating electrical machines (IP code) – Classification.

Note 3 to entry: The enclosure which provides the degree of protection IP is not necessarily the same as the equipment enclosure providing the Type of Protection.

Note 4 to entry: An enclosure which provides the degree of protection required by one of the Types of Protection will have been subjected to other tests prior to the tests for degree of protection

[SOURCE: IEV 426-04-02]

3.8
Male contact
Pin contact
Contact member intended to make electric engagement on its outer surface for mating with the inner surface of another contact member.

[SOURCE: IEV 151-12-18]

3.9
Female contact
Socket contact
Ccontact member intended to make electric engagement on its inner surface for mating with the outer surface of another contact member

Note – In English, the term "socket contact" does not imply that socket contacts are always mounted in a socket nor that sockets have only socket contacts.

[SOURCE: IEV 151-12-17]

3.10
Twisted loom (pair)
Cabled assembly

Cable consisting of insulated conductors or single-core cables twisted together without a common covering

[SOURCE: IEV 461-04-06]

3.11
Master
Device which spontaneously sends information on a bus to a number of slave devices.

Note 1 to entry: The master can give a slave the right to transmit for one slave frame only within a limited time.

[SOURCE: IEV 811-37-62]

3.12
Slave
Device which receives information from the bus or sends information on the bus in response to a request (also called a poll) from the master.

[SOURCE: IEV 811-37-96]

3.13
Checksum
Frame verification byte

[SOURCE: ISO 17987-1:2016, 3.1.3]

Abbreviations

- CBRN – chemical, biological, radiological, and nuclear substances
- NEST - iNteropErable multidomain CBRN SysTem
- LoRaWAN ®– Long Range Wide Area Network
- LoRa ®– Long Range
- I^2C – Inter-Integrated Circuit
- IoT – Internet of Things
- RAM – Random Access Memory
- EEPROM – Electrically Erasable Programmable Read-Only Memory
- MSB – Most Significant Bit

4 Description of the developed system

The following chapters describe the developed system in detail and highlight why standardizing the connection of individual and stand-alone sensors to a data-processing system makes sense and is important.

4.1 General

The aim was to design and produce a system that detects various CBRN hazards. To realize this, the first step was to develop a central control unit, which facilitates communication with the individual CBRN sensors connected by cable.

Since there is no manufacturer who develops and distributes all the required sensor types, three partners each developed a new sensor prototype (chemical, biological and radiological) for the NEST project, with which the communication with the central control unit could be standardised.

As this specific project involves a newly developed system, the standardisation of the communication of the individual sensors with the control unit, which was planned in advance, could be implemented successfully. The overall system consists of a communication unit, the so-called Sensor-HUB, to which up to six independently operating sensors can be connected according to the current state. These sensors communicate with the HUB as slaves via I^2C. The HUB in turn sends the data via LoRaWAN to the middleware or platform above it.

In real applications, however, this is often not so easy to realize, as the sensors that are to be used differ in many aspects such as the power supply, communication interface, data rate, etc., making simultaneous data retrieval via a central control unit difficult, if not impossible. This is where the CWA comes in and tries to show how the connection and communication of individual sensors from different manufacturers can be standardised and thus simplified.

4.2 Platform, Sensor-HUB, and sensors

The HUB is the central communication unit with which up to six individual sensing modules can communicate. It manages the transfer of sensor data to the IoT platform through its onboard LoRa Module and can be powered with 12 V DC. The HUB needs an external LoRa gateway connected to the internet for the establishment of a bidirectional (LoRaWan class A) communication to the IoT platform. Figure 1 depicts a schematic overview of the HUB and the different communication units.

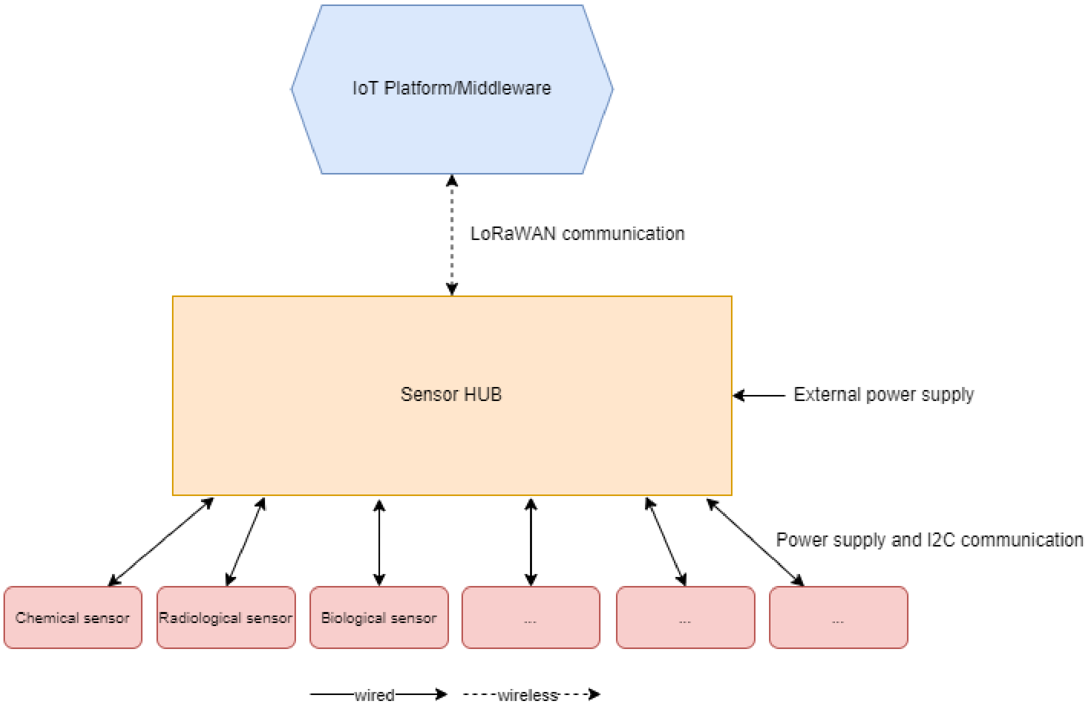

Figure 1: Schematic overview of the HUB and the communication units

With the aim of reducing the airtime transmission and comply with the European Telecommunications Standards Institute (ETSI) regulations the data transmitted via LoRaWAN should be reduced as much as possible. Depending on the country and the bands used ETSI allows a maximum duty cycle of 0.1, 1 or 10%. In Europe ETSI EN 300 220-1 is applied, and the EU863-870 band range is used which allows a 1% duty cycle of transmission airtime. Also, the airtime needed for a transmission is dependent on other factors such as the transmission bandwidth and the spread factor used by the LoRa (the Sensor-HUB uses DR3, SF9BW125).

From this point to describe the communication from the Platform (Gateway) to the Sensor-HUB the terms "Down", "Downlink" and for the other way round, communication from Sensor-HUB to the Platform (Gateway) the terms "Up", "Uplink" will be used.

There are three types of so called "Uplink-Messages". A normal transmission message when no sensor alarms are triggered will only include the actual time stamp of the system and the information about which sensors are active (attached and working), will be called "Alive-Message".

If any alarm is triggered the Sensor-HUB will immediately transmit an "Alarm-Message", which will include at least the alarm field, and any other appropriate data according to the established measurement protocol.

To be able to retrieve any information of the sensor system or to configure important parameters, a "Downlink-Message" can be sent to the Sensor-HUB, which then replies with a "Data-Message" with the requested data.

The HUB also provides geo-localization, system time and additional data (complementing individual sensor modules) like pressure, temperature, and humidity.

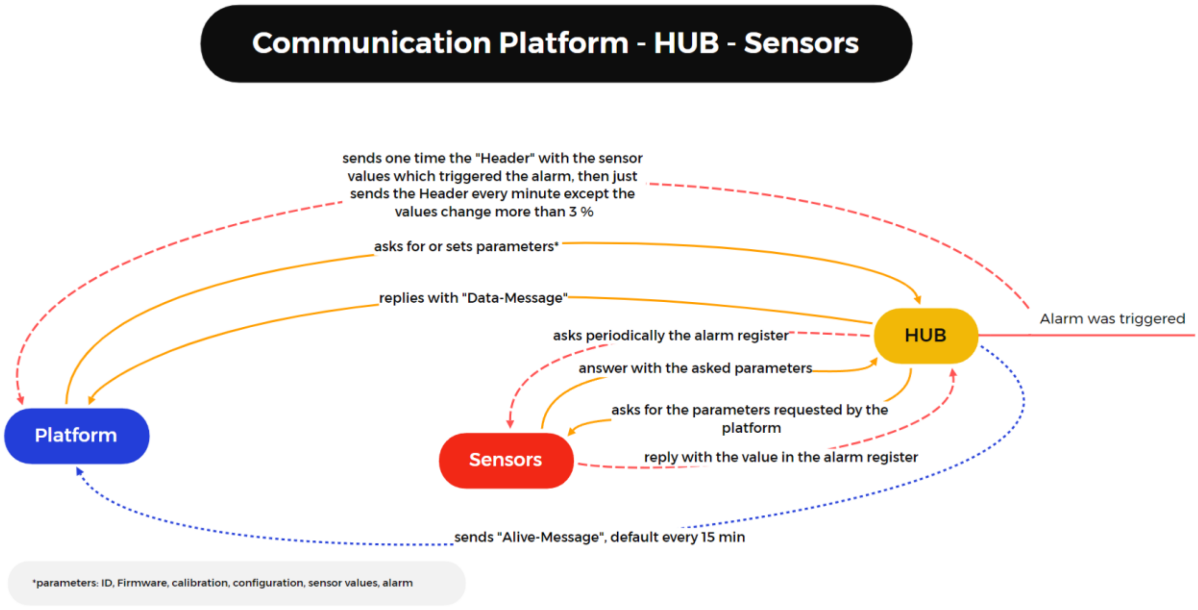

Figure 2: Communication flow between platform, HUB and sensors

Figure 2 depicts the communication between the platform, the HUB and the sensors. There is one loop between the HUB and the sensors, which runs continuously. Here the HUB asks periodically the alarm register of each sensor. If an alarm was triggered, the HUB sends the "Header" with the sensor values which triggered the alarm to the platform one time, then it just sends the Header every minute except the values change more than 3%. In this case the sensor values are sent again. If no alarm is triggered, the HUB sends an "Alive-Message" with default every 15 min. Additionally the platform can ask for or set parameters either of the HUB or the sensors. In this case the HUB replies with a "Data-Message". For further information about the Header and the different kinds of implemented messages, refer to the Annex A (LoRa communication protocol).

The communication between the HUB and the sensors and the power supply of all sensor modules is done via hardware connection with each of the sensors getting its own 5 V power supply from the HUB. The communication protocol is based on the I^2C standard with a 5 V logic. The HUB is the I^2C master and initiates communication with the sensor modules (slave devices). Each connected sensor has a specific address that is known in advance. There are defined commands for reading and writing register addresses in both RAM and EEPROM of the slave. The commands for the individual readable and writable values are implemented in advance in the HUB, thus the connection of an unknown sensor is not possible. To guarantee the correctness of the data during transmission, there are two verification mechanisms. On the one hand, the command sent by the HUB is repeated, including an error check, and on the other hand, a checksum of all sent bytes (except address and direction bit) is also sent. This checksum is sent and checked both by the HUB to the sensor and vice versa. Figure 3 is an example of the communication flow for reading values of a sensor module (on the left) and writing values to the sensor (on the right). The operation status includes the command sent by the HUB and the error check.

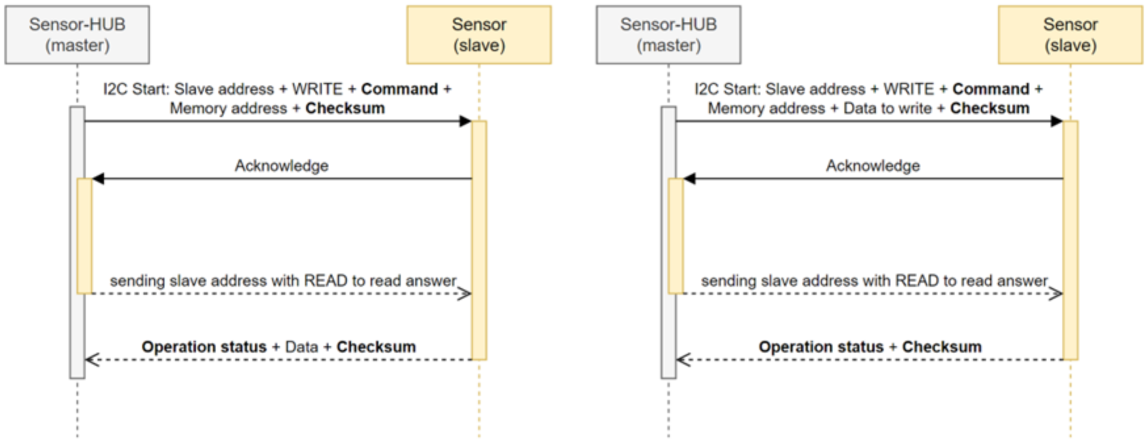

Figure 3: Example of an I^{2}C communication flow

The HUB and the sensor modules are connected via circular connectors with threaded joint M16 according to EN 61076-2-106 which are used in industrial data transfer applications. This IEC is an international standard for round connectors with screw-locking mating interfaces. It provides a uniform basis for the design, manufacture, and testing of round connectors in industrial applications, ensuring interoperability, reliability, and safety. These connectors are designed for input/output and sensor/actuator interfaces with IP67/68 protection. The HUB has the male connector and the cable the female one.-The connectors used have six pins, of which only four are required. Two for the I^{2}C communication and two for power supply purposes. The remaining ones are reserved and could potentially provide 12 V, an additional alarm signal or something else. Figure 4 on the left depicts the male connector of the HUB and the sensors (e.g. Lumberg 0314-06) and on the right the female connector for the cable (e.g. Lumberg 0322-06).

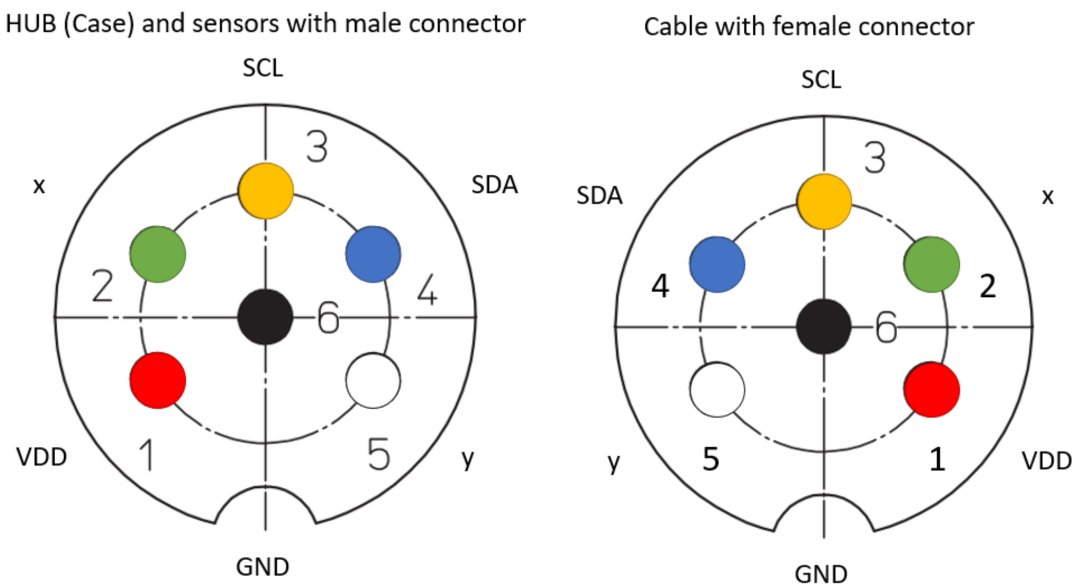

Figure 4: Pin assignment of male and female connectors

The colours correspond to the colour of the used core of the connected cable (see Figure 5 e.g. U3Z500 USB-cable 8 x 0.08 mm²). A twisted pair is used for the I²C lines to ensure that the signal transmission is as loss-free and interference-free as possible.

Figure 5: Example of a cable with the shields and twisted pairs

4.3 Necessity of standardised communication and hardware compatibility

With the system presented, it was already clear in advance which sensor types were to be connected. Nevertheless, care was taken during the design to ensure that communication was as uniform as practically. The system described was developed for a very specialised area, that of CBRN detection. To ensure the early detection of hazards, it is essential to guarantee loss-free and interference-free data transmission. For this reason, data transmission by cable from the sensors to the HUB was chosen. In this area, as in some others, it makes sense to standardise the hardware implementation of data transmission so that every newly developed sensor type is compatible with the existing detection system.

5 Component requirements

The development of the system described in the previous chapter has resulted in the following requirements for the individual components in order to provide a system that is as open as possible for the connection of a wide variety of sensors.

5.1 Hardware design of the sensors

- Power supply
  - 5 V
- Current consumption
  - Maximum of 500 mA
- Connector
  - Lumberg 0314-06
  - Connection: see chapter 4.2 Fig. 4
- Memory
  - RAM
  - EEPROM

5.2 Software implementation of the sensors

The sensor module must use the I²C Bus standard with 5 V logic for communication with the HUB. The sensor module acts as a slave device, controlled by a master I²C device. The master on the bus generates clock signals on SCL line and will determine if the sensor module should act as a receiver or transmitter on the bus. The I^2C bus physical layer is implemented accordingly with the standard I^2C communication protocol (according to the document UM10204 Rev 7.0 October 2021 [2]) to support data rates of 100 kbits/s.

The device address can be configured in EEPROM, see I^2C address chapter.

5.2.1 Write RAM/EEPROM

(grey)	Send by Slave
	Send by Host/Master

5.2.1.1 Write request

START condition	7 bit address + W/R bit (Write)	Command + N° data bytes	Memory address	Data to write	Checksum	STOP condition
	1 byte	1 byte	1 byte	1 to 31 bytes	1 byte	

Start	A6	A5	A4	A3	A2	A1	A0	0	ACK	C3	C2	C1	C0	D3	D2	D1	D0	ACK	…
	Slave address [6:0]							W		Command [2:0]			N° Data bytes [4:0]						…

…	A7	A6	A5	A4	A3	A2	A1	A0	ACK	D7	D6	D5	D4	D3	D2	D1	D0	ACK	…
…	Memory address [7:0]									Data to write 1 [7:0]									…

…	D7	D6	D5	D4	D3	D2	D1	D0	ACK	D7	D6	D5	D4	D3	D2	D1	D0	ACK	Stop
…	Data to write N [7:0]									Checksum [7:0]									

5.2.1.2 Write response

START condition	7 bit address + W/R bit (Read)	Operation status	Checksum	STOP condition

	1 byte	1 byte	1 byte	

Star t	A 6	A 5	A 4	A 3	A 2	A 1	A 0	1	AC K	S 7	S 6	S 5	S 4	S 3	S 2	S 1	S0	AC K	…
	Slave address [6:0]							R		Operation status [7:0]									…

…	D7	D6	D5	D4	D3	D2	D1	D0	NACK	Stop
…	Checksum [7:0]									

5.2.2 Read RAM/EEPROM

5.2.2.1 Read request

START condition	7 bit address + W/R bit (Write)	Command + N° data bytes	Memory address	Checksum	STOP condition
	1 byte	1 byte	1 byte	1 byte	

Star t	A 6	A 5	A 4	A 3	A 2	A 1	A 0	0	AC K	C 3	C 2	C1	C 0	D3	D 2	D 1	D0	AC K	…
	Slave address [6:0]							W		Command [2:0]			N° Data bytes [4:0]						…

…	A 7	A 6	A 5	A 4	A 3	A 2	A 1	A 0	A C K	D 7	D 6	D 5	D 4	D 3	D 2	D 1	D 0	A C K	St o p
…	Memory address [7:0]									Checksum [7:0]									

5.2.2.2 Read response

START condition	7 bit address + W/R bit (Read)	Operation status	Data to read	Checksum	STOP condition
	1 byte	1 byte	1 to 31 bytes	1 byte	

Star t	A 6	A 5	A 4	A 3	A 2	A 1	A 0	1	AC K	S 7	S 6	S 5	S 4	S 3	S 2	S 1	S0	AC K	…
	Slave address [6:0]							R		Operation status [7:0]									…

…	D7	D6	D5	D4	D3	D2	D1	D0	ACK	…	D7	D6	D5	D4	D3	D2	D1	D0	ACK	…
…	Data to read 1 [7:0]									…	Data to read N [7:0]									…

…	D7	D6	D5	D4	D3	D2	D1	D0	NACK	Stop
…	Checksum [7:0]									

5.2.3 Bit fields details for write and read request

Byte position	Bit position in byte	Value	Field description
0	7:1	0x00 to 0x7F	I²C address. Default address: see datasheet.
	0	0 or 1	Read/Write direction bit. Read = 1; Write = 0
1	7:5	0b001x xxxx to 0b100x xxxx	Command code: 0b001x xxxx – Write RAM 0b010x xxxx – Read RAM 0b011x xxxx – Write EEPROM 0b100x xxxx – Read EEPROM Remaining values reserved for future use.
	4:0	0bxxx0 0000 to 0bxxx1 1111	Number of bytes to read/write. Range from 1 to 31. 0bxxx0 0000 means 1, 0bxxx1 1111 means 31.
2			Memory address from sensor. MSB first.
3:(2+N)			Data to be written (N=1 to 31 bytes) <u>Field only present for Writing command request</u>. For Read command request this field is not present.
3+N			Arithmetic sum of the bytes sent without first byte (address and direction bit). ***(see example code provided)***

5.2.4 Bit fields details for write and read response

Byte position	Bit position in byte	Value	Field description
0	7:1	0x00 to 0x7F	I²C address. Default address: see datasheet.
	0	0 or 1	Read/Write direction bit. Read = 1; Write = 0
1	7:5	0b001x xxxx to 0b100x xxxx	Command code: 0b001x xxxx – Write RAM 0b010x xxxx – Read RAM 0b011x xxxx – Write EEPROM 0b100x xxxx – Read EEPROM

			Remaining values reserved for future use.
	4:0	0bxxx0 0000 to 0bxxx1 1111	Operation Status. ERRORCHECK. 0bxxx0 0000 indicates No Errors, complete and/or valid data. Otherwise, error list must be checked.
2:(1+N)			Data to be read (N=1 to 31 bytes) Field only present for Reading command responses. For Write command responses this field is not present.
2+N			Arithmetic sum of the bytes sent without first byte (address and direction bit). ***(see example code provided)***

5.2.5 Variables and parameters details for the sensor

Variable/Parameter	**Magnitude**	**Type**	**Size**	**Write/ Read**
I^2C Address	N/A. ***See I²C address chapter***	Unsigned int	1 byte	W/R
Firmware Version	N/A. ***See product information chapter***	Unsigned int	1 byte	R
Sensor ID	N/A. ***See product information chapter***	Unsigned int	2 bytes	R
Production information	N/A. ***See product information chapter***	Unsigned int	4 bytes	R
Measured values	N/A.	Signed int	2 bytes	R
Thresholds	N/A.	Signed int	2 bytes	W/R
Calibration parameters	N/A.	Unsigned int	2 bytes	W/R
Configuration	N/A.	Unsigned int	2 bytes	W/R
MeasureTime Interval	Deciseconds (ds)	Unsigned int	2 bytes	W/R
Alarm	N/A.	Unsigned int	1 byte	R
Reset	N/A.	Unsigned int	1 byte	W
Status	N/A.	Unsigned int	1 byte	R

5.2.6 Example Code for Checksum calculation

Arithmetic sum of the bytes sent (not including first byte with address and direction bit).

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

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

5.2.7 Memory map definition for SRAM and EEPROM

5.2.7.1 Sensor Module SRAM Memory Map

Variable	*Address*		*1 byte*	*Address*		*1 byte*
Reserved	0	0x00				
Alarm	1	0x01	-	-	-	AL (7:0)
Reset		-	-	2	0x02	R (7:0)
Sensor Value 1	3	0x03	S1 (15:8)	4	0x04	S1 (7:0)
Sensor Value 2	5	0x05	S2 (15:8)	6	0x06	S2 (7:0)
Sensor Value 3	7	0x07	S3 (15:8)	8	0x08	S3 (7:0)
Sensor Value 4	9	0x09	S3 (15:8)	10	0x0A	S3 (7:0)
Sensor Value 5	11	0x0B	S3 (15:8)	12	0x0C	S3 (7:0)
Sensor Value 6	13	0x0D	S3 (15:8)	14	0x0E	S3 (7:0)
Sensor Threshold 1	15	0x0F	ST1 (15:8)	16	0x10	ST1 (7:0)
Sensor Threshold 2	17	0x11	ST1 (15:8)	18	0x12	ST1 (7:0)
Sensor Threshold 3	19	0x13	ST2 (15:8)	20	0x14	ST2 (7:0)
Sensor Threshold 4	21	0x15	ST2 (15:8)	22	0x16	ST2 (7:0)
Sensor Threshold 5	23	0x17	ST3 (15:8)	24	0x18	ST3 (7:0)
Sensor Threshold 6	25	0x19	ST3 (15:8)	26	0x1A	ST3 (7:0)
Measure Time interval	27	0x1B	MT (15:8)	28	0x1C	MT (7:0)
Open to industry	29	0x1D	*Reserved*	30	0x1E	*Reserved*
Open to industry	31	0x1F	*Reserved*	32	0x20	*Reserved*

5.2.7.2 Sensor Module EEPROM Memory Map

Variable	*Address*		*1 byte*	*Address*		*1 byte*
Reserved	0	0x00				
Sensor ID	1	0x01	ID (15:8)	2	0x02	ID (7:0)
Firmware Version	-	-	-	3	0x03	FW (7:0)
Production Information 1	4	0x04	PI (15:8)	5	0x05	PI1(7:0)
Production Information 2	6	0x06	PI2(15:8)	7	0x07	PI2(7:0)
I^2C Address	8	0x08	-	9	0x09	ADRS (7:0)
Calibration Parameter 1	10	0x0A	CP1 (15:8)	11	0x0B	CP1 (7:0)
Calibration Parameter 2	12	0x0C	CP1 (15:8)	13	0x0D	CP1 (7:0)
Calibration Parameter 3	14	0x0E	CP2 (15:8)	15	0x0F	CP2 (7:0)
Calibration Parameter 4	16	0x10	CP2 (15:8)	17	0x11	CP2 (7:0)
Calibration Parameter 5	18	0x12	CP3 (15:8)	19	0x13	CP3 (7:0)
Calibration Parameter 6	20	0x14	CP3 (15:8)	21	0x15	CP3 (7:0)
Configuration	22	0x16	CF (15:8)	23	0x17	CF (7:0)
Open to industry	24	0x18	*Reserved*	25	0x19	*Reserved*
Open to industry	26	0x1A	*Reserved*	27	0x1B	*Reserved*

5.2.7.3 Sensor ID definition

	Manufacturer						Freely choosable									
Bits	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

The sensor ID consists of 2 bytes of which the first 6 bits are for registered manufacturers. These are listed in a separate table. If a manufacturer is not yet registered, the first 6 bits contain zeros. The rest of the bits (7-16) can be freely coded, but the explanation must be also noted in a table.

5.2.7.4 Error check control definition

In every Write or Read Response an Operation Status is sent. It is in the 4 less significant bits of the Byte 1. A zero value (0x0) indicates no error. Each bit indicates one error cause, multiple bits can be active at once.

Bit number	Error description
0	Checksum from received request fail
1	N° bytes received not match with expected
2	*open to industry*

3	*open to industry*
4	*open to industry*

5.2.7.5 Calibration Parameters definition and functionality

If the calibration parameters are used for a sensor, they must be stored and described in the sensor's data sheet.

5.2.7.6 Production Information of the Sensor

The production information of the sensor is defined in a four bytes parameter, divided in PI1 and PI2.

For PI1, the first byte indicates the fabrication year and month date, the first 4 bits indicate the year (start counting at 2020), and the next 4 bits indicate the month. The second byte indicates in the same way the main version and the last one the sub-version.

On the figure below an example for clarification is shown for a sensor with version 1.2, produced in March 2023.

For PI2, the first byte indicates sensor vendor ID identifier, while the second byte indicates sensor model identifier. PI2 is intended to be used by upper software layers to identify vendor and sensor model so to retrieve from its local database (or vendor database) metadata necessary to automatically render data structures, communication channels and commands so to create digital model of sensor within the backend and frontend of the software. It is out of the scope of this CWA how this is performed.

Production information “PI1“ address 0x02 (15:0) two bytes parameter			
PI1 (15:8) Address 0x02		PI1 (7:0) Address 0x03	
Year (from 2020)	Month	Main version	Sub-version
0 0 1 1	0 0 1 1	0 0 0 1	0 0 1 0
Year 2023	Month March	Hardware version 1	Sensor version 2
Production information “PI2“ address 0x04 (15:0) two bytes parameter			
PI2 (15:8) Adress 0x04		PI2 (7:0) Address 0x05	
Vendor_ID		Model_ID	

5.2.7.7 I^2C Address specifications

The I^2C address is configured in the EEPROM of the sensor. Different I^2C address ranges are assigned to the different sensor categories (C, B & RN). The table can be expanded to include other sensor categories, such as environmental sensors.
At the moment, up to 16 sub-categories can be distinguished per sensor category via the I^2C address.

Sensor module category	**Address Bit 0**	**Address Bit 1**	**Address Bit 2**	**Address Bit 3**	**Address Bit 4**	**Address Bit 5**	**Address Bit 6**	**Address Bit 7**
Chemical sensor modules	0	0	1	x	x	x	x	x
Biological sensor modules	0	1	0	x	x	x	x	x
Radiological and nuclear modules	0	1	1	x	x	x	x	x

Modification of the I^2C sensor address should be possible to avoid address conflicts between the sensors.

Annex A (Informative)
LoRaWAN Communication Protocol

This annex will define the radio communication protocol between the sensors systems and the platform which is the LoRaWAN communication between Sensor-HUB and the Platform (Gateway).

With the aim of reducing the airtime transmission and comply with the European Telecommunications Standards Institute (ETSI) regulations the data transmitted should be reduced as much as possible. Depending on the country and the bands used ETSI allows a maximum duty cycle of 0.1, 1 or 10%. In Europe the EU863-870 band range is used which allows a 1% duty cycle of transmission airtime. Also, the airtime needed for a transmission is dependent on other factors such as the transmission bandwidth and the spread factor used by the LoRa (the Sensor-HUB uses DR3, SF9BW125).

From this point to describe the communication from the Platform (Gateway) to the Sensor-HUB the terms “Down”, “Downlink” and for the other way round, communication from Sensor-HUB to the Platform (Gateway) the terms “Up”, “Uplink” will be used.

There are three types of so called **“Uplink-Messages”**. A normal transmission message when no sensor alarms are triggered will only include the actual time stamp of the system and the information about which sensors are active (attached and working), will be called **“Alive-Message”**.

If any alarm is triggered the Sensor-HUB will immediately transmit an **“Alarm-Message”**, which will include at least the alarm field, and any other appropriate data according to the established measurement protocol.

To be able to retrieve any information of the sensor system or to configure important parameters, a **“Downlink-Message”** can be sent to the Sensor-HUB, which then replies with a **“Data-Message”** with the requested data.

A short overview over the communication between Platform, HUB and sensors can be found in the appendix.

Downlink message and payload definition

The “Downlink-Message” as shown in figure 1 will be used to write to or request fields of the sensor system. It consists of a “Command” field which will define the sensor and variable to write to or read from. In case of a write packet the corresponding “Data” field information will follow. The number of bytes of the data field will depend on the definition of the variables/fields from each sensor and command definition. In case of a read request no “Data” bytes will be sent. See table 1 and 2 for read and write command definitions. Table 1 also includes a column called CSP (configuration sensor parameter) which is related to the “Read all Configuration Sensor Parameters” command. When this command is sent, the Sensor-HUB will send per module a “Data-Message” with every parameter which is checked in the CSP column. Table 3 includes a column called DBG (debug). If a variable is checked, it is possible to display the related debug parameters in the platform.

COMMAND	DATA (Only if W/R → “1”)

Write → "1" Read → "0"	Sensor & Fields to be sent/read	Data sent accordingly to the sensor and fields definition
W/R [7]	Command [6:0]	
byte 0		byte 1…N

Figure 1.- Downlink message payload definition.

Read Commands

Read Commands (W/R → "0")					
Command	**Sensor**	**Variable**	**Size**	**Byte Description***	**CSP**
0000000	HUB	Alarm	0 byte	Sensors alarm status sent in "Data-message" Header without extra data	
0000001		Alive send rate	2 bytes	TSR (0…1)	X
0000010		ID & Firmware	2 bytes	ID (0); FW (1)	X
0000011		Read all Configuration Sensor Parameters	0 byte	Use to ask for all the configuration parameters (CSP marked)	
0000100	GPS	Latitude & Longitude	8 bytes	Lat (0…3); Lon (4…7)	
0000101		Latitude & Longitude thresholds	8 bytes	LatT (0…3); LonT (4…7)	X
0000110	Environmental	Temperature, Pressure & Humidity	7 bytes	Temp (0…1); Pres (2…4) Humid (5…6)	
0000111		Temperature, Pressure & Humidity thresholds	7 bytes	TempT (0…1); PresT (2…4) HumiT (5…6)	X
0001000	MOX	ID & Firmware	2 bytes	ID (0); FW (1)	X
0001001		Calibration parameters	12 bytes	CP1 (0…1); CP2 (2…3) CP3 (4…5); CP4 (6…7) CP5 (8…9); CP6 (10…11)	X
0001010		Configuration parameters	17 bytes	T1(0); T2(1); T3(2); HT1(3…4); HT2(5…6); HT3(7…8); BL1(9…10); BL2(11…12); BL3(13…14); MT (15…16)	X
0001011		Gas & sensor values	12 bytes	G1 (0…1); G2 (2…3) G3 (4…5); S1 (6…7) S2 (8…9); S3 (10…11)	
0001100		Gas & sensor threshold values	12 bytes	GT1 (0…1); GT2 (2…3) GT3 (4…5); ST1 (6…7) ST2 (8…9); ST3 (10…11)	X
0011000	RAMAN	ID & Firmware	2 bytes	ID (0); FW (1)	X
0011001		Calibration parameters	12 bytes	CP1 (0…1); CP2 (2…3) CP3 (4…5); CP4 (6…7) CP5 (8…9); CP6 (10…11)	X
0011010		Configuration parameters	4 bytes	IT (0…1); MT (2…3)	X
0011011		Gas values	24 bytes	G1 (0…3); G2 (4…7) G3 (8…11); G4 (12…15) G5 (16…19); G6 (20…23)	

0011100		Gas threshold values	24 bytes	GT1 (0…3); GT2 (4…7) GT3 (8…11); GT4 (12…15) GT5 (16…19); GT6 (20…23)	X
0100000	Biological	IDs & Firmware	10 bytes	ID (0…3); DI (4…7); FW (8…9)	X
0100001		Read debugging parameters	12 bytes	ST(0); ER(1); IP1(2…3); IP2(4…5); IP3(6…7); IP4(8…9); IP5(10…11)	X
0100010		Read detection threshold	4 bytes	DT(0…3)	X
0100011		Detection values	4 bytes	DV(0…3)	
0100100		TimeStamp	4 bytes	TS(0…3)	X
0110000	Radiological	Sensor Identification	3 bytes	FW(0); IDS(1); TS(2)	X
0110001		Sensor communication	4 bytes	RT(0); ST (1); ER (2); AL(3)	X
0110010		Integration time	2 bytes	T(0…1)	X
0110011		Temperature	1 bytes	TP(0)	X
0110100		Threshold parameters	10 bytes	THDEF(0…1); THMIN(2…3); THMAX(4…5); TH(6…7) TH_COUNT(8…9)	X
0110101		Radiation values	4 bytes	V(0…1); VAL(2…3);	
**** For type, size and magnitude description see Table 3.***					

Table 1.- Read command definition.

Write Commands

Write Commands (W/R → "1")				
Command	**Sensor**	**Variable**	**Size**	**Byte Description***
0000000	HUB	Reset	1 byte	RST(0)
0000001		Alive send rate	2 bytes	TSR (0…1)
0000010		Reserved	1 byte	Reserved
0000011		Reserved	1 byte	Reserved
0000101	GPS	Latitude & Longitude thresholds	8 bytes	LatT (0…3); LonT (4…7)
0000111	Environmental	Temperature, Pressure & Humidity thresholds	7 bytes	TempT (0…1); PresT (2…4) HumiT (5…6)
0001001	MOX	Calibration parameters	12 bytes	CP1 (0…1); CP2 (2…3) CP3 (4…5); CP4 (6…7) CP5 (8…9); CP6 (10…11)
0001010		Configuration parameters	17 bytes	T1(0); T2(1); T3(2); HT1(3…4); HT2(5…6); HT3(7…8); BL1(9…10); BL2(11…12); BL3(13…14); MT (15…16)
0001100		Gas & sensor threshold values	12 bytes	GT1 (0…1); GT2 (2…3) GT3 (4…5); ST1 (6…7) ST2 (8…9); ST3 (10…11)
0011001	RAMAN	Calibration parameters	12 bytes	CP1 (0…1); CP2 (2…3) CP3 (4…5); CP4 (6…7)

				CP5 (8…9); CP6 (10…11)
0011010		Configuration parameters	4 bytes	IT (0…1); MT (2…3)
0011100		Gas threshold values	24 bytes	GT1 (0…3); GT2 (4…7) GT3 (8…11); GT4 (12…15) GT5 (16…19); GT6 (20…23)
0100010	Biological	Detection thresholds	4bytes	DT(0…3);
0100100		Time Stamp	4 bytes	TS(0…3)
0100001		Internal parameters 1-5	10 bytes	IP1(0…1); IP2(2…3); IP3(4…5); IP4(6…7); IP5(8…9);
0110010	Radiological	Integration Time	2 bytes	T(0…1)
0110100		Threshold parameters	4 bytes	TH (0…1); THCOUNT(2…3)
**** For Type and magnitude description see Table 3.***				

Table 2.- Write command definition.

Variables and their magnitudes

Sensor	Descriptor	Variable	Type	Magnitude	Default values	Min/Max	DBG	W/R
HUB	ID	ID	UINT	N/A	0	0 / 255		R
	FW	Firmware version	UINT	N/A	0	0 / 255		R
	TSR	Alive send rate interval	UINT	Minutes (m)	15	1 / 7000		W/R
	RST	Reset (To Restore HUB and Sensors parameters to default values)	UINT	N/A	0	0 / 255		W
Environmental	Temp	Temperature	INT	1e-2 °C	2500	-4000 / 8500		R
	TempT	Temperature threshold	UINT	1e-2 °C	200	1 / 8500		W/R
	Pres	Pressure	UINT	1e-2 hPa	101300	10 / 200000		R
	PresT	Pressure threshold	UINT	1e-2 hPa	100	10 / 2000		W/R
	Humi	Humidity	UINT	1e-3 %	20000	0 / 65535		R
	HumiT	Humidity threshold	UINT	1e-3 %	5000	0 / 65535		W/R
GPS	Lat	Latitude	INT	(1e-7) Decimal Degrees	0	-90e7 / 90e7		R

	LatT	Latitude threshold	INT	(1e-7) Decimal Degrees	10e-7	-90e7 / 90e7		W/R
	Lon	Longitude	INT	(1e-7) Decimal Degrees	0	-180e7 / 180e7		R
	LonT	Longitude threshold	INT	(1e-7) Decimal Degrees	10e-7	-180e7 / 180e7		W/R
MOX	ID	Identification number	UINT	N/A	0	0 / 255		R
	FW	Firmware version	UINT	N/A	0	0 / 255		R
	CP1 … CP6	Calibration parameter	UINT	N/A	0	0 / 65535	X	W/R
	T1 … T3	Heater Temperature	UINT	4 °C	63	0 / 255		W/R
	HT1 … HT3	Heating time	UINT	ms	100	1 / 65535		W/R
	BL1 … BL3	Base line	UINT	N/A	0	0 / 65535	X	W/R
	MT	Measure time interval	UINT	0.1 s	10	1 / 65535		W/R
	G1 … G3	Gas value	UINT	2 ppb	0	0 / 65535		R
	GT1 … GT3	Gas threshold	UINT	2 ppb	100	0 / 65535		W/R
	S1 … S3	Sensor value	UINT	N/A	0	0 / 65535	X	R
	ST1 … ST3	Sensor threshold	UINT	N/A	50	0 / 65535	X	W/R
RAMAN	ID	Identification number	UINT	N/A	0	0 / 255		R
	FW	Firmware version	UINT	N/A	0	0 / 255		R
	CP1 … CP6	Calibration parameter	UINT	N/A	0	0 / 65535	X	W/R
	IT	Integration time	UINT	ms	10	0 / 65500		W/R
	MT	Measure time interval	UINT	0.1 s	10	1 / 65530		W/R
	G1 … G6	Gas value	UINT	ppb	0	0 / 4.29e9		R
	GT1 … GT6	Gas threshold	UINT	ppb	100	0 / 4.29e9		W/R
Biological	ID	Sensor chip ID	UINT	N/A	0	0 / 4294967295		R
	DI	Device ID	UINT	N/A	0	0 / 4294967295		R

	FW	Firmware version	UINT	N/A	0	0 / 4294967295		R
	ST	Status	UINT	N/A	0	0 / 255	X	R
	ER	Error Fields	UINT	N/A	0	0 / 255	X	R
	IP1 … IP5	Internal Parameters	UINT	N/A	0	0 / 65500		R/W
	DT	Detection Threshold	UINT	Number of particles/protein molecules (1e-2 µg/ml)	120	0 / 10000		W/R
	DV	Detected Values	UINT	Number of particles/protein molecules (1e-2 µg/ml)	100	0 / 10000		W/R
Radiological	FW	Firmware version	UINT	N/A	0	0 / 255		R
	IDS	Sensor ID	UINT	N/A	0	0 / 255		R
	TS	Sensor type	UINT	N/A	0	0 / 255		R
	RT	Reset	UINT	N/A	0	0 / 255		R
	ST	Status	UINT	N/A	0	0 / 255	X	R
	ER	Error Field	UINT	N/A	0	0 / 255	X	R
	AL	Alarm	UINT	N/A	0	0 / 255		R
	TP	Internal temperature	INT	°C	20	-128 / 127	X	R
	T	Integration Time	UINT	Seconds (S)	1	1 / 3600		W/R
	THDEF	Detection threshold Default	UINT	cps	40	0 / 65000	X	R
	THMIN	Detection threshold MIN	UINT	cps	10	0 / 65000	X	R
	THMAX	Detection Threshold MAX	UINT	cps	1000	0 / 65000		R
	TH	Detection Threshold	UINT	cps	40	0 / 65000		W/R
	TH_COUNT	Detection Threshold COUNT	UINT	cps	100	0 / 65000		W/R
	V	Measure Value	UINT	cps	0	0 / 65000		R

	VAL	Measure Value AL	UIN T	cps	0	0 / 65000		R

Table 3.- Variable types and magnitudes definition.

For HUB, Environmental, GPS, MOX and RAMAN variables increments should be made in the magnitudes expressed. For example, for MOX Heater Temperature (T) only values starting from 0 °C in 4°C steps till 1024°C are possible.

Reset of the system

For the HUB Reset command the data byte will indicate if the complete system, a single sensor or a group of sensors should restore the variables to default values. See figure 2 for bit description and examples, where "1" means to restart and "0" means make nothing.

Sensor	Reserved	HUB	Environmental	GPS	Biological	Radiological	MOX	RAMAN
HUB "RST" byte	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Example reset only HUB parameters (0x40)	0	1	0	0	0	0	0	0
Example reset HUB, GPS, and Radiological parameters (0x54)	0	1	0	1	0	1	0	0
Example reset all parameters (0x7F)	0	1	1	1	1	1	1	1

Figure 2.- Reset bit register definition. If bit is set to "1" it will indicate that this sensor should be reset to default values.

A "Reset" button is available at the HUB to force a software reset of the HUB system. Furthermore a "User define" button is available at the HUB to force an Alive-Message to be sent to the Platform.

The firmware version of the Sensor-HUB and the different sensors will indicate in the most significant part of the byte the major firmware version and in the less significant one the minor firmware version. For the case of the biological sensor which has two bytes for this variable, the most significant byte will indicate the major, and the less significant one the Minor firmware version. For example, a firmware version "V1.05" will correspond with "0x15" (Hexadecimal) or "0001 0101" (binary) for the one-byte firmware version.

Uplink messages and payload definition

The simple message will be the Alive-Message which will only transmit information about the sensors that are alive (present and working), and the current time of the system. This will also be the "HEADER" of an Uplink message. As in figure 3 is shown, the bit [39] "Type" will indicate if it is an Alive-Message or a Data- or Alarm-Message. The next 6 bits [38:33] ("Active/Alarm sensor") will be the active indicator for each possible sensor attached. If it is an Alive-Message (bit not set), then the next bit will indicate, if the sensor attached is "on" ("0") or "off" ("1"). If it is a Data-Message (bit set), the next bit will indicate if the status of the sensor attached is "ok" ("0") or if an alarm was triggered ("1") (see table 4 for definition). The bits [32:0] are used to transfer the time stamp of the system. The GPS (position) and environmental sensor (temperature, humidity and pressure) will be always active because they are included in the HUB system hardware. In normal condition the HUB should have all the sensors attached. If the communication with any of them fails or any alarm is triggered it will activate the corresponding bit (see table 4). The GPS bit will indicate if a valid GPS signal is available ("0" GPS signal is available, "1" no signal). A valid GPS signal will synchronize the system internal real clock used for the time stamp.

<table>
<tr><td></td><td colspan="22">HEADER</td></tr>
<tr><td></td><td>Type [39]</td><td colspan="3">Active / Alarm sensor [38:33]</td><td colspan="18">System time stamp [32:0]</td></tr>
<tr><td></td><td>0→Alive
1→Data</td><td colspan="3">0→ ON/OK
1→OFF/Alarm</td><td colspan="3">year</td><td colspan="3">month</td><td colspan="3">day</td><td colspan="3">hour</td><td colspan="3">minutes</td><td colspan="3">seconds</td></tr>
<tr><td>bit</td><td>39</td><td>38</td><td>…</td><td>33</td><td>32</td><td>…</td><td>26</td><td>25</td><td>…</td><td>22</td><td>21</td><td>…</td><td>17</td><td>16</td><td>…</td><td>12</td><td>11</td><td>…</td><td>6</td><td>5</td><td>…</td><td>0</td></tr>
<tr><td>byte</td><td colspan="5">0</td><td colspan="4">1</td><td colspan="5">2</td><td colspan="4">3</td><td colspan="4">4</td></tr>
</table>

Figure 3.- Uplink Message Header payload definition (MSB first).

Sensor	Active/Alarm sensors field bit assigned
Biological	33
Radiological	34
Metal Oxide	35
Optical	36
GPS	37
Reserved	38

Table 4.- Active sensor field bit assignment.

An Alive-Message will be sent regularly to the platform to inform that the system is alive/active and if all the sensors are connected and working. This Alive-Message default send rate is 15 minutes but could be modified by the platform.

If any alarm is triggered the first Alarm-Message sent to the platform will include the Header with the specific data variables of the corresponding sensor. The HUB will continue sending Alarm-Messages every minute until the alarm is no longer active, but the following messages will only consist of the Header with no data so long the data don't change more than 3% of the first sent values.

To transmit data to the platform the "Data-Message" is used. The bit Type [39] of the Header is set to "1" to indicate that it is a data packet. The Alarm bits [38:33] will indicate the alarm status of the sensors. If extra data is sent either because of a previous platform request or because the measurement protocol is so defined, after the Header a data part will be added to the payload of the message. As in figure 4 is shown, the data payload part consists of a command byte [7:0] which will define the sensor and data variables that are sent, it is followed by the corresponding data bytes accordingly to the command definition (see table 1 and 2).

More than one data load could be sent concatenated in the same uplink message.

	HEADER	DATA		
		Sensor & Fields sent		Data sent accordingly to the sensor and fields definition
bit		Reserved [7]	Command [6:0]	Data
byte	0…4	5		6...N

Figure 4.- Uplink Message Data payload definition (MSB first).

<u>Examples for message types</u>

1.- Alive-Message sent from HUB to Platform with MOX & Radiological sensors active, RAMAN and Biological sensors inactive/not present.

Uplink message consists of only the Header (5 bytes long) with bits:

Type = 0 → Alive Message

Active sensor = 000110 → MOX and Radiological sensors have communication problems.

	HEADER											
	Type	Active sensors						Time stamp				
bit	0	0	0	0	1	1	0	x	---	---	---	---
byte	0								1	2	3	4

Alive message sent from Sensor-HUB to Platform.

2.- Data message sent from Sensor-HUB to Platform when alarm is triggered in biological sensor.

In this case an Alarm-Message is sent. The uplink message consists of only the Header (5 bytes long) with bits:

Type = 1 → Data Message

Alarm sensor = 000001 → Biological Sensor Alarm

	HEADER											
	Type	Alarm sensors						Time stamp				
bit	1	0	0	0	0	0	1	x	---	---	---	---
byte	0								1	2	3	4

Data message sent from Sensor-HUB to Platform when an alarm is triggered.

3.- Platform requests some sensor parameters. In this example the environmental sensor values of temperature, pressure and humidity

Platform sends a Downlink message with only the Header (1 byte long) with bits:

W/R = 0 → Read request

Command = 0000101 → Environmental temperature, pressure and humidity request

	HEADER							
	W/R	Command						
bit	0	0	0	0	0	1	0	1
byte	0							

Downlink message from platform to Sensor-HUB to request Environmental sensor values.

The Sensor-HUB will respond with an Uplink Data-Message consisting of the Header (5 bytes long) and the data part (8 bytes) as follows:

	HEADER												DATA														
	Type	Alarm sensor						Time Stamp					Sensor & Fields sent								Data						
														Command							T		P			H	
bit	1	0	0	0	0	0	0	-	-	-	-	-	x	0	0	0	0	1	0	1	-	-	-	-	-	-	-
byte	0								1	2	3	4	5								6	7	8	9	10	11	12

Response Data-message to an environmental sensor values request.

Type = 1 → Data Message

Alarm sensor = 000000 → No alarm, all sensors are ok. If any alarm is active or triggered at this moment would be here indicated as well.

Command = 0000101 → Environmental temperature, pressure and humidity request

Data bytes 6...12 will contain the corresponding sensor values.

4.- Platform sends a Write request to the HUB system to set configuration parameters of the MOX sensor, new threshold gas values shall be set. The Downlink Data-Message consists of the Header (1 byte) and the data part corresponding to the data values to be written (12 bytes) as follows:

	HEADER								DATA											
	W/R	Command							GT1		GT2		GT3		ST1		ST2		ST3	
bit	1	0	0	0	1	0	1	0	-	-	-	-	-	-	-	-	-	-	-	-
byte	0								1	2	3	4	5	6	7	8	9	10	11	12

W/R = 1 → Write request

Command = 0001010 → Gas & sensor threshold values

Data bytes 1…12 will contain the corresponding new threshold values to be written to the sensor.

Bibliography

[1] https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan

[2] https://www.nxp.com/docs/en/user-guide/UM10204.pdf