Client-Server Communication Model

Between sensor and base station

Author: Simon Hoffmann

Last edited: 09.11.2022

Contents

[1. Packet format 3](#_Toc118993494)

[2. Packet types 4](#_Toc118993495)

[3. Send Data 5](#_Toc118993496)

[3.1 Water Sensor 5](#_Toc118993497)

[3.2 Smoke Detector 5](#_Toc118993498)

[4. Communication examples 6](#_Toc118993499)

[4.1 Connection establishment 6](#_Toc118993500)

[4.1.1 New Connection 6](#_Toc118993501)

[4.1.2 Existing Connection 6](#_Toc118993502)

[4.2 Disconnect 7](#_Toc118993503)

[4.3 Connection available (PING) 7](#_Toc118993504)

[4.4 Sending Data 8](#_Toc118993505)

# Packet format

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **idSensor** | **idBase** | **packetID** | **packetType** | **Data** | **End identifier** |
| uint8\_t | uint8\_t | uint8\_t | uint8\_t | variable | ~ |

idSensor - unique id for sensor

isBase - unique id for Base Station

packetID - unique packet id for when a message is answered

packetType - The type of data being sent

Data - Packet data

End identifier - Indicates end of package

# Packet types

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Packettype** | **idSensor** | **idBase** | **packetID** | **packetType** | **Data** |
| CONNECT | 0 | 0 | n | 1 | Sensor Type\* |
| CONNACK | 0 | uint8\_t | n | 2 | Id for Sensor+ |
|  |  |  |  | 3 |  |
| ACK | uint8\_t | uint8\_t | m | 4 | - |
| SEND | uint8\_t | uint8\_t | k | 5 | See [Send Data](#_Send_Data) |
| PING | uint8\_t | uint8\_t | i | 6 | - |
| DISCONNECT | uint8\_t | uint8\_t | g | 7 | - |

+Id for Sensor (uint8\_t)

\*Sensor Type (uint8\_t):

1 – Water sensor

2 – Smoke detector

3 –

# Send Data

First part is uint8\_t with type of sensor

## 3.1 Water Sensor

The Raspberry pi Pico have temperature sensors, the conversion must be done base station side.

|  |  |  |  |
| --- | --- | --- | --- |
| Sensor type (uint8\_t) | Humidity (uint16\_t) | Submerged (uint8\_t) | Temperature (uint16\_t) |
| 1 | n | 0 or 1 | n |

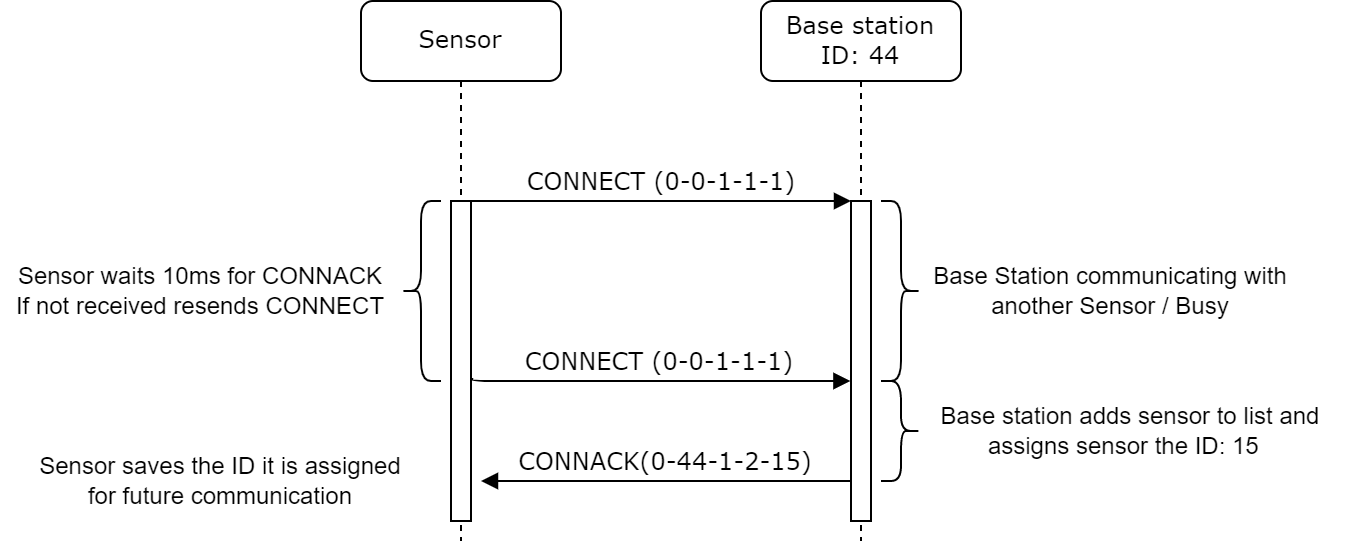
## 3.2 Smoke Detector

|  |  |  |  |
| --- | --- | --- | --- |
| Sensor type (uint8\_t) | Smoke / CO2 detection | CO detection | Temperature (uint16\_t) |
| 2 | Not sure if can discern the difference | Not sure if can discern the difference | n |

# Communication examples

## 4.1 Connection establishment

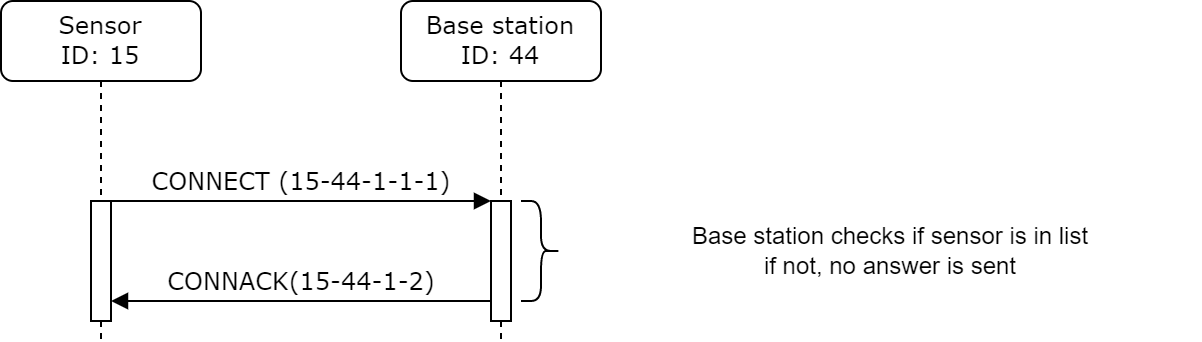
### 4.1.1 New Connection

Sensor has no ID (Default: 0) meaning it needs one assigned

If sensor doesn’t receive a CONNACK after 10 tries it sleeps for 1min.

### 4.1.2 Existing Connection

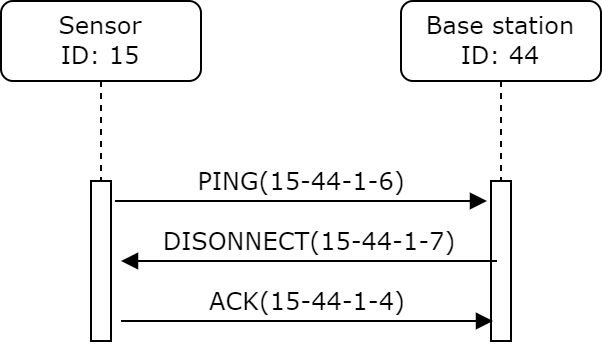
Sensor has ID: 15



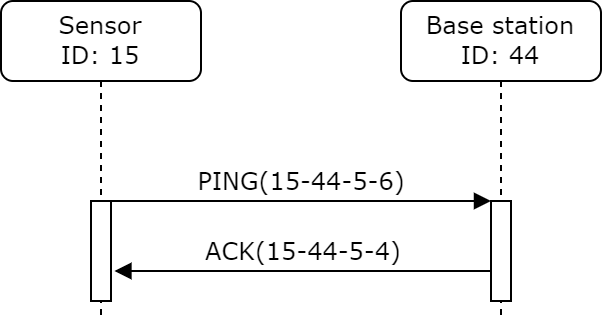
If sensor doesn’t receive a CONNACK after 10 tries it sleeps for 1min.

## 4.2 Disconnect

Disconnect can only be done upon receiving a PING Packet, because the Sensor RF Module is not constantly active. The Ping happens every minute, allowing a relatively timely disconnect.



## 4.3 Connection available (PING)

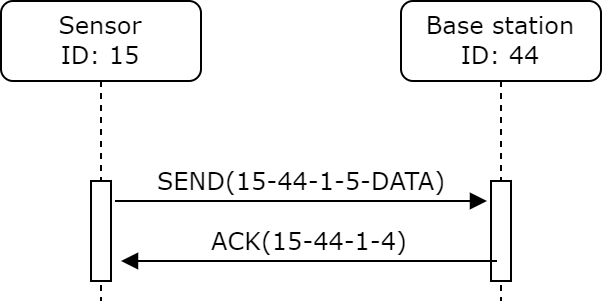


The sensor must send a PING Packet ever 1min to the Base station to indicate it is still active. If the base station does not receive a packet after 2min, it sets the sensor status to offline/error.

If the sensor does not get an ACK/DISCONNECT after 10 tries, it must retry connecting to the station.

This must only happen if nothing was sent in the 1min span.

## 4.4 Sending Data



Data is sent until an acknowledge is received (every 10ms up to 10 tries)

Base station sends ACK back to sensor.

If the sensor does not get an ACK/DISCONNECT after 10 tries, it must retry connecting to the station.

Base station must ignore duplicate messages (with previous and current states maybe)