BLE RTD
Rev.0.5.10Mar21

taskit GmbH

Contents

# 1. Introduction

## 1.1 Scope of the Document

The scope of the present document is to describe a protocol specification that defines a 2.4 Ghz radio message format and a configuration service that defines a GATT configuration service to enable interoperability between hardware and application developers for taskit GmbH BLE RTD sensor interface.

This document presents the protocol specification and the configuration service for BLE RTD, Release 0.50.

## 1. Protocol Specification

The common frame PDU types and the individual service data byte layouts for the BLE RTD frame formats.

### 2.1 Common Elements

The core radio data type are defined in the standard as little-endian.

| Byte offset | Valu | Description |
|---|---|---|
| 0 1 2 3 4 | **e** | Length |
| 5 6 7 | 0x02 | GAP Advertising Flags |
| | 0x01 | General Discoverable Mode \| BR/EDR Not |
| | 0x06 | Supported Length |
| | 0x1B | Manufacturer Specific Advertisement Data |
| | 0xFF | 16-bit Company Identifier |
| | 0x7B | ... |
| | 0x01 | Frame Version |
| | 0x30 | |

#### 2.1.1 Length

The frame length in this version number is fixed at 27 bytes.

#### 2.1.2 Company Identifier

0x017B : taskit GmbH Company Identifier.

#### 2.1.3 Frame Version

Is the frame version number and defines the frame structure.

### 2.2 Frame Specification

All multi-byte values are little-endian.

| Byte offset | Field | Description |
|---|---|---|
| 8 9 10 11 | Frame | |
| 12 13 14 | Type Main | |
| 15 16 17 | Type | |
| 18 19 20 | Sub Type | 16-byte |
| 21 22 23 | UUID[15] | UUID ... |
| 24 25 26 | UUID[14] | |
| 27 28 29 | UUID[13] | |
| 30 | UUID[12] | |
| | UUID[11] | |
| | UUID[10] | |
| | UUID[9] | |
| | UUID[8] | |
| | UUID[7] | |
| | UUID[6] | |
| | UUID[5] | |
| | UUID[4] | |
| | UUID[3] | |
| | UUID[2] | |
| | UUID[1] | |
| | UUID[0] | 4-byte Data |
| | Data[3] | ... |
| | Data[2] | |
| | Data[1] | |
| | Data[0] | |

### 2.2.1 Frame Type

| **Frame Type** | Power | 0x0B |
|---|---|---|
| | **Byte** | **Description** |
| Measurement | **Value** | Broadcasts one or more measured values. |
| Alarm | 0x01 0x0A | Broadcasts an alarm depending on the sensor |

### 2.2.2 Main Type

Defines the type of the frame type.

| **Frame Type** | Power | | 0x01 0x0D 0x0E 0x0F |
|---|---|---|---|
| | **Main Type** | Diagnosti | 0x0D |
| Measurement | | c | **Description** |
| | Temperature | Voltage | Defines the type of measurement frame. |
| Alar | Voltage | **Byte** | |
| m | Low Power | **Value** | Defines the type of alarm frame. |

### 2.2.3 Sub Type

Defines the data type of the data field.

| **Main Type** | **Sub Type** | **Byte** | **4-byte Data Type** |
|---|---|---|---|
| | 1/100°C | **Value** | INT32 |
| Measurement | 1/100°C | 0x02 0x03 | FLOAT |
| | mV | 0x02 0x02 | UINT32 |
| Voltage | Sensor | | Bit Field |
| Diagnostic | Flags | | |

### 2.2.4 16-byte UUID

The 16-byte UUID is intended for cryptographic communication.

Without cryptographic procedures, the UUID corresponds to the device address.

### 2.2.5 4-byte Data

The 4-byte data depends on the frame, main and sub-type.

### 2.3 Scan Response Data

Sent as a scan response to a scan request in legacy advertising.

| Value | Description |
|---|---|
| 0X?? | «Complete Local Name» advertising data |
| 0x09 | type ... |
| 0x0A | Length |
| 0x02 | «Tx Power Level» advertising data type |
| 0x?? | |

Note:
If the device name is an empty string, the field «Complete Local Name» is ignored.
The maximum length of the device name is 20 bytes.

## 1. Configuration GATT Service

The BLE RTD Configuration Service runs as a GATT service on the sensor interface while it is connectable and allows configuration of the advertised data, the sensor calibration and the measuring intervals.

### 3.1 Service Definition

| | |
|---|---|
| Service UUID | ee8a**fff0**-b5be-11e3-9d09-0002a5d5c51b |
| Notes | Where not explicitly stated, data written and read is defined in terms of little-endian arrays of bytes. |

### 3.2 Characteristic

#### 3.2.1 Measuring Interval

| | |
|---|---|
| Characteristic UUID | ee8a**fff1**-b5be-11e3-9d09-0002a5d5c51b |
| Properties | uint16 (read) Note: little-endian<br>uint16 (write) Note: little-endian |
| Description | Advertising interval of the frame type measurement (0x01) depends on the measuring interval (in milliseconds). |
| Min/Max Value | 100/10000 |
| Default Value | 1000 |
| Return Codes | Invalid Attribute Length: attempt to write illegal values. |

#### 3.2.2 Calibration Offset

| | |
|---|---|
| Characteristic UUID | ee8a**fff2**-b5be-11e3-9d09-0002a5d5c51b |
| Properties | int32 (read) Note: little-endian<br>int32 (write) Note: little-endian |
| Description | The value written in 1/100°C is added to the calculated RTD value in 1/100°C. |
| Default Value | 0.00 |
| Return Codes | Invalid Attribute Length: attempt to write illegal values. |

#### 3.2.3 Calibration Slope

| | |
|---|---|
| Characteristic UUID | ee8a**fff3**-b5be-11e3-9d09-0002a5d5c51b |
| Properties | int32 (read) Note: little-endian<br>int32 (write) Note: little-endian |
| Description | The value written in 1/100% is multiplied by the calculated RTD value in 1/100°C. |
| Default Value | 1.00 |
| Return Codes | Invalid Attribute Length: attempt to write illegal values. |

### 3.2.4 Measured Value

| Characteristic UUID | ee8a**fff4**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | int32 (read) Note: little-endian<br>int32 (notify) Note: little-endian |
| Description | Read: returns the last measured RTD value in 1/100°C.<br>Notify: returns the currently measured RTD value in 1/100°C. |
| Return Codes | Write Not Permitted: on any write attempt. |

### 3.2.5 Sensor Type

| Characteristic UUID | ee8a**fff5**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | uint8 (read)<br>uint8 (write) |
| Description | 0: PT100<br>1: PT500 Note: Unavailable<br>2: PT1000 |
| Default value | 2: PT1000 |
| Return Codes | Invalid Attribute Length: attempt to write illegal values. |

### 3.2.6 Device Name

| Characteristic UUID | ee8a**fff6**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | utf8 string (read)<br>utf8 string (write) |
| Description | The device name is returned in the active scan response with the «Complete Local Name» GAP advertising data type (0x09).<br><br>Write this characteristic with an empty string to ignore the device name in active scan. |
| Length | ≤ 20 bytes |
| Default | "BLERTD" |
| value | Invalid Attribute Length: attempt to write illegal values. |

Return Codes

### 3.2.7 Store Config

| Characteristic UUID | ee8a**fff7**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | uint8 (write) |
| Description | All changes to the characteristic values of the configuration service are first stored if a 1 is written to the store config characteristic, otherwise the changed values will be lost after a restart. |
| Possible values | 0: Factory Reset<br>1: Store Config |
| Return Codes | Invalid Attribute Length: attempt to write illegal values. |

### 3.2.8 PHY

| Characteristic UUID | ee8a**fff8**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | uint8 (read)<br>uint8 (write) |
| Possible values | 1: 1M PHY (Legacy)<br>2: Coded S=2 PHY (Long range)<br>3: 1M PHY \| Coded S=2 PHY |
| Default Value | 1: 1M PHY (Legacy) |
| Return Codes | Invalid Attribute Length: attempt to write illegal values. |

### 3.2.9 Pairing Passcode

| Characteristic UUID | ee8a**fff9**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | uint32 (write) Note: little-endian |
| Description | The pairing process using an association method that supports MITM (Man in the Middle) protection. |
| Possible values | 0 to 999999 |
| Default Value | 0: Pairing Disabled |
| Return Codes | Invalid Attribute Length: attempt to write illegal values. |

### 3.2.10 Radio TxPower

| Characteristic UUID | ee8a**fffa**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | uint8 (read)<br>uint8 (write) |
| Description | Sets the RF transmitter output power.<br><br>The TxPower is returned in the active scan response with the «Tx Power Level» GAP advertising data type (0x0A). |
| Possible values | 0: -21 dBm<br>1: -18 dBm<br>2: -15 dBm<br>3: -12 dBm<br>4: -9 dBm<br>5: -6 dBm<br>6: -3 dBm<br>7: 0 dBm<br>8: 1 dBm<br>9: 2 dBm<br>10: 3 dBm<br>11: 4 dBm<br>12: 5 dBm<br>7: 0 dBm |
| Default Value | Invalid Attribute Length: attempt to write illegal values. |
| Return Codes | |

### 3.2.11 Battery Level

| Characteristic UUID | ee8a**fffb**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | uint16 (read) Note: little-endian<br>uint16 (notify) Note: little-endian |
| Description | Battery level is the current battery voltage in millivolts.<br><br>Read: returns the last measured battery voltage.<br>Notify: returns the currently measured battery |
| Return Codes | voltage. Write Not Permitted: on any write attempt. |

### 3.2.12 Sensor Diagnostic

| Characteristic UUID | ee8a**fffc**-b5be-11e3-9d09-0002a5d5c51b |
|---|---|
| Properties | uint8 (read)<br>uint8 (notify) |
| Description | Allowing the detection of open and short-circuit conditions and out-of-range signals. |
| Return Codes | Write Not Permitted: on any write attempt. |

**Sensor Diagnostic Flags**

| Bit | Description |
|---|---|
| 6 5 | An open-circuit or short-circuit condition is detected. |
| [4:3 | A power-on reset because the last time the register was |
| ] | read. The modulator was over-ranged. |