

RS1xx LoRa Protocol

Sentrius™ RS1xx Sensor

Application Note

v2.10

1 INTRODUCTION

The goal of this document is to detail the messages sent between the RS1xx sensor and a LoRa network server.

2 CONVENTIONS

2.1 Data Formats

2.1.1 Data Types

Data types are based on the C language definitions for fixed point data types defined in *stdint.h* for C99 and newer compilers. These define whether a value is signed and how many bits are used to represent its value. For example, *int8_t* is a signed value represented in eight bits and *uint32_t* is a 32-bit unsigned value. The *_t* at the end designates a type defined (rather than native) data type.

2.1.2 Data Format

Temperature, humidity and voltage values are represented as two or four-byte values, depending upon the range of the connected sensor. The first byte, or pair of bytes, is the fractional portion of the value. The second byte, or pair of bytes, is the integer portion of the value. For example, a temperature of 27.43 degrees C has a fractional portion of 43 and an integer portion of 27. A temperature of -15.87 C would have a fractional portion of -87 and an integer portion of -15.

The following are the equations for temperature, humidity and voltage:

- $\text{Temp} = \text{Integer Portion} + (\text{Fractional Portion}/100)$
- $\text{Humidity} = \text{Integer Portion} + (\text{Fractional Portion}/100)$
- $\text{Voltage} = \text{Integer Portion} + (\text{Fractional Portion}/100)$

Note: Temperature and voltage use signed eight and sixteen-bit values, while humidity uses unsigned eight-bit values. However, because humidity cannot be negative, the sign bit is never set and there is no practical difference between using signed or unsigned values for humidity.

Refer to sections 8 and 9 for further details.

3 APPLICATION PORTS

Sensor uplink messages are sent on port 1. Downlink messages will be processed by the sensor on ports 0 through 223.

4 SERVER-TO-SENSOR MESSAGES

4.1 Message Options

All Server-to-Sensor messages have the same options byte format. The options byte is always at byte index 1.

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value	
		Sensor Response Options:						
		0 None						
		1 Send simple configuration next uplink						
		2 Send advanced configuration next uplink						
		3 Send firmware version next uplink						
[1]	Options	4 Send Open/Closed sensor config next uplink	enum	0x00	0x00	0x00	0x08	
		5 Send Battery Voltage next uplink						
		6 Reset sensor next uplink						
		7 Start sensor advertising next uplink						
		8 Send RTD configuration next uplink						

4.2 Message Name: Generic Data Retrieval Notification

Notification

Method Name: GenericDataRetrieval

Notes

Message Length = 2 bytes

Server Method: asGenericRequest

Direction: Downlink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	0x01	0x01	0x01	0x01	0x01
[1]	Options	See the Message Options section	enum	0x00	0x07	0x00	0x08

Example Message

Sent MSB First.

01 07

4.3 Message Name: Set UTC Notification

Notification

Method Name: SetUTC

Notes

Message Length = 8 bytes

Server Method: generateRTCDownlink

Direction: Downlink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	0x02	0x02	0x02	0x02	0x02
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2]	RTC_Year	Year (the current year less 2000)	uint8_t	0x0F	0x0F	0x0F	0x6C
[3]	RTC_Month	Month	uint8_t	0x01	0x01	0x01	0x0C
[4]	RTC_Day	Day	uint8_t	0x01	0x01	0x01	0x1F
[5]	RTC_Hours	Hours (24-hour format)	uint8_t	0x00	0x0E	0x00	0x17
[6]	RTC_Minutes	Minutes	uint8_t	0x00	0x0A	0x00	0x3B
[7]	RTC_Seconds	Seconds	uint8_t	0x00	0x01	0x00	0x3B

Example Message

Sent MSB First.

```
02 00 0F 01 01 0E 0A 01
```

4.4 Message Name: Sensor Config Notification

Notification

Method Name: Config

Notes

Message Length = 16 bytes

Server Method: asSetSensorParameters

Direction: Downlink

ReadSensorPeriod – Determines the time between individual temperature and humidity readings.

SensorAggregate – Determines how many readings are accumulated prior to sending a data packet to the gateway.

For example, if the *SensorAggregate* is set to 2 and the *ReadSensorPeriod* is set to 60 seconds, the data is sent over LoRa 2 x 60 seconds or every 120 seconds. Increasing the *SensorAggregate* value has a positive effect on battery life.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	0x03	0x03	0x03	0x03	0x03
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value				
[2]	Config_Battery Type	<table border="0"> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">1</td> <td>Zinc-Manganese Dioxide (Alkaline)</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">2</td> <td>Lithium/Iron Disulfide (Primary Lithium)</td> </tr> </table>	1	Zinc-Manganese Dioxide (Alkaline)	2	Lithium/Iron Disulfide (Primary Lithium)	uint8_t	0x01	0x01	0x01	0x02
1	Zinc-Manganese Dioxide (Alkaline)										
2	Lithium/Iron Disulfide (Primary Lithium)										
[3:4]	Config_Read SensorPeriod	Period in seconds to read the sensor <table border="0"> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">0</td> <td>Disabled</td> </tr> </table>	0	Disabled	uint16_t	0x0000	0x000A	0x0000	0xFFFF		
0	Disabled										
[5]	Config_Sensor Aggregate	Number of readings to aggregate before sending on LoRa	uint8_t	0x01	0x01	0x01	0x0B				
[6]	Config_Temp AlarmEnable	Enable temperature alarm	bool	0x00	0x01	0x00	0x01				
[7]	Config_Humidity AlarmEnable	Enable humidity alarm	bool	0x00	0x00	0x00	0x01				
[8]	Config_Temp AlarmLimitLow	Temperature alarm limit - low	int8_t	0xD8	0x00	0xD8	0x55				
[9]	Config_Temp AlarmLimitHigh	Temperature alarm limit - high	int8_t	0x32	0x32	0xD8	0x55				
[10]	Config_RH AlarmLimitLow	Humidity alarm limit - low	int8_t	0x1E	0x0A	0x00	0x64				
[11]	Config_RH AlarmLimitHigh	Humidity alarm limit - high	int8_t	0x50	0x50	0x00	0x64				
[12:13]	Config_LED_BLE	Flash period in seconds when in BLE connection <table border="0"> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">0</td> <td>No flash</td> </tr> </table>	0	No flash	uint16_t	0x000A	0x000A	0x0000	0xFFFF		
0	No flash										
[14:15]	Config_LED_LoRa	Flash period in seconds <table border="0"> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">0</td> <td>No flash</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">65535</td> <td>Tx/Rx debug mode Tx – green Rx – orange</td> </tr> </table>	0	No flash	65535	Tx/Rx debug mode Tx – green Rx – orange	uint16_t	0x000A	0x000A	0x0000	0xFFFF
0	No flash										
65535	Tx/Rx debug mode Tx – green Rx – orange										

Note: High Alarm Limits must be set to greater values than Low Alarm Limits.

Example Message

```
Sent MSB First.
03 00 01 00 0A 01 01 00 00 32 0A 50 00 0A 00 0A
```

4.5 Message Name: Heater Control Notification

Notification

Method Name: HeaterControl

Notes

Message Length = 5 bytes

Server Method: asSetHeater

Direction: Downlink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	0x04	0x04	0x04	0x04	0x04
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2]	Options_Si702x_Heater_Setting	Setting that controls the power sent to the heating element. See si7021 datasheet for more details	uint8_t	0x00	0x02	0x00	0x0F
[3:4]	Options_Si702x_Heater_Time	Time in milliseconds for the unit to turn on the heating element 0 Heater disabled	uint16_t	0x0000	0x000A	0x0000	0xFFFF

Example Message

Sent MSB First.

```
04 00 02 00 0A
```

4.6 Message Name: Backoff Notification

Notification

Method Name: Backoff

Notes

Message Length = 4 bytes

Server Method: asRequestBackoff

Direction: Downlink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x05	0x05	0x05	0x05
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2:3]	BackoffPeriod	Amount of time in seconds to back off the sensor	uint16_t	0x0000	0x003C	0x0000	0xFFFF

Example Message

Sent MSB First.

```
05 00 00 3C
```

4.7 Message Name: FIFO Backlog Retrieval Notification

Notification

Method Name: BacklogRetrievalFIFO

Notes

Message Length = 6 bytes

Server Method: asRequestBacklogFIFO

Direction: Downlink

This command instructs the module to upload the stored logs and alarms oldest record first.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x06	0x06	0x06	0x06
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2:3]	Backlog_PullReqNum	Number of records to send Note: Alarm messages are sent first. 0 Cancel pull request	uint16_t	0x0000	0x0001	0x0000	0xFFFF
[4:5]	Backlog_PullReqPeriod	How often to send a backlog LoRa packet (seconds)	uint16_t	0x0000	0x000A	0x0000	0xFFFF

Example Message

Sent MSB First.

```
06 00 00 01 00 0A
```

4.8 Message Name: Format Log Flash Notification

Notification

Method Name: FormatLogFlash

Notes

Message Length = 2 bytes

Server Method: asFormatFlash

Direction: Downlink

This command erases all backlog and alarm records.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	0x07	0x07	0x07	0x07	0x07
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08

Example Message

Sent MSB First.

```
07 00
```

4.9 Message Name: Set Alkaline Thresholds Notification

Note: This message was deprecated and is no longer supported.

Notification

Method Name: SetAlkalineThresholds.

Notes

Message Length = 14 bytes

Server Method: asSetAlkalineThresholds

Direction: Downlink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x08	0x08	0x08	0x08
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2:3]	EightyPercent	80% threshold in mV	uint16_t	0x0A96	0x0A96	0x0000	0xFFFF
[4:5]	SixtyPercent	60% threshold in mV	uint16_t	0x0A14	0x0A14	0x0000	0xFFFF
[6:7]	FortyPercent	40% threshold in mV	uint16_t	0x099C	0x099C	0x0000	0xFFFF
[8:9]	TwentyPercent	20% threshold in mV	uint16_t	0x0898	0x0898	0x0000	0xFFFF
[10:11]	FivePercent	5% threshold in mV	uint16_t	0x076C	0x076C	0x0000	0xFFFF
[12:13]	Offset	Offset from 20 degrees C in mV	uint16_t	0x0064	0x0064	0x0000	0xFFFF

Example Message

Sent MSB First.

```
08 00 0A 96 0A 14 09 9C 08 98 07 6C 00 64
```

4.10 Message Name: Set Lithium Thresholds Notification

Note: This message was deprecated and is no longer supported.

Notification

Method Name: SetLithiumThresholds

Notes

Message Length = 14 bytes

Server Method: asSetLithiumThresholds

Direction: Downlink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	0x09	0x09	0x09	0x09	0x09
[1]	Options	See the Message Options section	0x00	0x00	0x00	0x00	0x08
[2:3]	EightyPercent	80% threshold in mV	uint16_t	0x0B86	0x0B86	0x0000	0xFFFF
[4:5]	SixtyPercent	60% threshold in mV	uint16_t	0x0B40	0x0B40	0x0000	0xFFFF
[6:7]	FortyPercent	40% threshold in mV	uint16_t	0x0AF0	0x0AF0	0x0000	0xFFFF
[8:9]	TwentyPercent	20% threshold in mV	uint16_t	0x0A5A	0x0A5A	0x0000	0xFFFF
[10:11]	FivePercent	5% threshold in mV.	uint16_t	0x0834	0x0834	0x0000	0xFFFF
[12:13]	Offset	Offset from 20 degrees C in mV	uint16_t	0x0096	0x0096	0x0000	0xFFFF

Example Message

Sent MSB First.

```
09 00 0B 86 0B 40 0A F0 0A 5A 08 34 00 96
```

4.11 Message Name: LIFO Backlog Retrieval Notification

Notification

Method Name: BacklogRetrievalLIFO

Notes

Message Length = 6 bytes

Server Method: *asRequestBacklogLIFO*

Direction: Downlink

This command instructs the module to upload the stored logs and alarms oldest record first.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type.	uint8_t	0x0A	0x0A	0x0A	0x0A
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2:3]	Backlog_PullReqNum	Number of records to send. Note: Alarm messages are sent first. 0 Cancel pull request	uint16_t	0x0000	0x0001	0x0000	0xFFFF
[4:5]	Backlog_PullReqPeriod	How often to send a backlog LoRa packet (seconds).	uint16_t	0x0000	0x000A	0x0000	0xFFFF

Example Message

Sent MSB First.

```
0A 00 00 01 00 0A
```


4.12 Message Name: Cancel Backlog Retrieval

Notification

Method Name: CancelBacklogRetrieval

Notes

Message Length = 2 bytes

Server Method: asCancelBacklogRetrieval

Direction: Downlink

This command instructs the module to cancel the uploading of the logs and alarms.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type.	uint8_t	0x0B	0x0B	0x0B	0x0B
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08

Example Message

```
Sent MSB First.
0B 00
```

4.13 Message Name: Open/Closed Config Notification

Notification

Method Name: OpenClosedConfig

Notes

Message Length = 10 bytes

Server Method: asSetOpenClosedConfig

Direction: Downlink

This command determines the configuration of the Open/Closed sensor.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value				
[0]	MsgType	Message type	uint8_t	0x0D	0x0D	0x0D	0x0D				
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08				
[2]	Operating Mode	The operating mode of the Open/Closed sensor <table border="1" style="margin-left: 20px;"> <tr> <td>0</td> <td>Door Sensor</td> </tr> <tr> <td>1</td> <td>RFU - Pushbutton</td> </tr> </table>	0	Door Sensor	1	RFU - Pushbutton	enum	0x00	0x00	0x00	0x01
0	Door Sensor										
1	RFU - Pushbutton										
[3]	LoRa Notification Options	Determines what Open/Closed sensor notifications are sent over LoRa BIT POSITION	bitfield	0x07	0x00	0x00	0x0F				

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
		0 Open notifications					
		1 Closed notifications					
		2 Resend notifications					
		3 Cancel notifications					
[4:5]	Open Dwell Time	Sets the duration in seconds after which an Open sensor status is annunciated	uint16_t	0x0005	0x0005	0x0005	0x03FF
[6:7]	Closed Dwell Time	Sets the duration in seconds after which a Closed sensor status is annunciated	uint16_t	0x0005	0x0005	0x0005	0x03FF
[8]	Resend Interval	Defines the duration in minutes after which status messages are resent	uint8_t	0x01	0x01	0x01	0xFF
[9]	Debounce Adjust	Defines the duration in milliseconds that the Open/Closed sensor status must remain consistent before the status is changed	uint8_t	0x00	0x10	0x00	0xFF

Example Message

Sent MSB First.

```
0D 00 00 00 00 05 00 05 01 10
```

4.14 Message Name: RTD Sensor Config Notification

Notification

Method Name: RTDConfig

Notes

Message Length = 15 bytes.

Server Method: *asSetRTDSensorParameters*

Direction: Downlink

ReadSensorPeriod determines the time between individual temperature and humidity readings. The *SensorAggregate* determines how many readings are accumulated prior to sending a data packet to the gateway. For example, if the *SensorAggregate* is set to 2 and the *ReadSensorPeriod* is set to 60 seconds, the data is sent over LoRa 2 x 60 seconds or every 120 seconds. Increasing the *SensorAggregate* value has a positive effect on battery life.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	0x0E	0x0E	0x0E	0x0E	0x0E
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2]	Config_Battery Type	1 Zinc-Manganese Dioxide (Alkaline)	uint8_t	0x01	0x01	0x01	0x02
		2 Lithium/Iron Disulfide (Primary Lithium)					

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[3:4]	Config_Read SensorPeriod	Period in seconds to read the sensor 0 Disabled	uint16_t	0x000A	0x000A	0x0000	0xFFFF
[5]	Config_Sensor Aggregate	Number of readings to aggregate before sending on LoRa	uint8_t	0x01	0x01	0x01	0x0B
[6]	Config_Temp AlarmEnable	Enable temperature alarm	bool	0x00	0x01	0x00	0x01
[7:8]	Config_Temp AlarmLimitLow	Temperature alarm limit - low	int16_t	0xFF38	0x0000	0xFF38	0x0352
[9:10]	Config_Temp AlarmLimitHigh	Temperature alarm limit - high	int16_t	0x00FA	0x0032	0xFF38	0x0352
[11:12]	Config_LED_ BLE	Flash period in seconds when in BLE connection 0 No flash	uint16_t	0x000A	0x000A	0x0000	0xFFFF
[13:14]	Config_LED_ LoRa	Flash period in seconds 0 No flash 65535 Tx/Rx debug mode Tx – green Rx – orange	uint16_t	0x000A	0x000A	0x0000	0xFFFF

Note: High Alarm Limits must be set to greater values than Low Alarm Limits.

Example Message

```
Sent MSB First.
0E 00 01 00 0A 01 01 00 00 00 32 00 0A 00 0A
```

4.15 Message Name: FIFO Targeted Backlog Retrieval Notification

Notification

Method Name: TargetedBacklogRetrievalFIFO

Notes

Message Length = 16 bytes

Server Method: *asRequestTargetedBacklogFIFO*

Direction: Downlink

This command instructs the module to upload the stored logs and alarms oldest record first within the range of the passed timestamps. Note that backlogs retrieved using this command are not removed from memory and must be read with either the FIFO or LIFO Backlog Retrieval Notification commands to achieve this.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x0F	0x0F	0x0F	0x0F
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[2]	StartYear	Start timestamp year	uint8_t	0x0F	0x0F	0x0F	0x6C
[3]	StartMonth	Start timestamp month	uint8_t	0x01	0x01	0x01	0x0C
[4]	StartDay	Start timestamp day	uint8_t	0x01	0x02	0x01	0x1F
[5]	StartHour	Start timestamp hour	uint8_t	0x00	0x03	0x00	0x17
[6]	StartMinute	Start timestamp minute	uint8_t	0x00	0x04	0x00	0x3B
[7]	StartSecond	Start timestamp second	uint8_t	0x00	0x05	0x00	0x3B
[8]	EndYear	End timestamp year	uint8_t	0x0F	0x10	0x0F	0x6C
[9]	EndMonth	End timestamp month	uint8_t	0x01	0x01	0x01	0x0C
[10]	EndDay	End timestamp day	uint8_t	0x01	0x02	0x01	0x1F
[11]	EndHour	End timestamp hour	uint8_t	0x00	0x03	0x00	0x17
[12]	EndMinute	End timestamp minute	uint8_t	0x00	0x04	0x00	0x3B
[13]	EndSecond	End timestamp second	uint8_t	0x00	0x05	0x00	0x3B
[14:15]	Backlog_PullReqPeriod	How often to send a backlog LoRa packet (seconds)	uint16_t	0x0000	0x000A	0x0000	0xFFFF

Example Message

Sent MSB First.

```
0F 00 0F 01 02 03 04 05 10 01 02 03 04 05 00 0A
```

4.16 Message Name: LIFO Targeted Backlog Retrieval Notification

Notification

Method Name: TargetedBacklogRetrievalLIFO

Notes

Message Length = 16 bytes

Server Method: *asRequestTargetedBacklogLIFO*

Direction: Downlink

This command instructs the module to upload the stored logs and alarms newest record first within the range of the passed timestamps. Note that backlogs retrieved using this command are not removed from memory and must be read with either the FIFO or LIFO Backlog Retrieval Notification commands to achieve this.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x10	0x10	0x10	0x10
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x08
[2]	StartYear	Start timestamp year	uint8_t	0x0F	0x0F	0x0F	0x6C
[3]	StartMonth	Start timestamp month	uint8_t	0x01	0x01	0x01	0x0C
[4]	StartDay	Start timestamp day	uint8_t	0x01	0x02	0x01	0x1F
[5]	StartHour	Start timestamp hour	uint8_t	0x00	0x03	0x00	0x17

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[6]	StartMinute	Start timestamp minute	uint8_t	0x00	0x04	0x00	0x3B
[7]	StartSecond	Start timestamp second	uint8_t	0x00	0x05	0x00	0x3B
[8]	EndYear	End timestamp year	uint8_t	0x0F	0x10	0x0F	0x6C
[9]	EndMonth	End timestamp month	uint8_t	0x01	0x01	0x01	0x0C
[10]	EndDay	End timestamp day	uint8_t	0x01	0x02	0x01	0x1F
[11]	EndHour	End timestamp hour	uint8_t	0x00	0x03	0x00	0x17
[12]	EndMinute	End timestamp minute	uint8_t	0x00	0x04	0x00	0x3B
[13]	EndSecond	End timestamp second	uint8_t	0x00	0x05	0x00	0x3B
[14:15]	Backlog_ PullReqPeriod	How often to send a backlog LoRa packet (seconds)	uint16_t	0x0000	0x000A	0x0000	0xFFFF

Example Message

Sent MSB First.

```
10 00 0F 01 02 03 04 05 10 01 02 03 04 05 00 0A
```

5 SENSOR-TO-SERVER MESSAGES

5.1 Sensor-to-Server Message Options

All Sensor-to-Server messages have the same options byte format. The options byte is always at byte index 1

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value	
		BIT POSITION						
		0	Sensor request for server time					
[1]	Options	1	Sensor configuration error					
		2	Sensor alarm flag					
		3	Sensor reset flag					
		4	Sensor fault flag					
			bitfield	0x00	0x00	0x00	0x10	

5.2 Message Name: Send Temp RH Data Notification

Notification

Method Name: SendTempRHData

Notes

Message Length = 11 bytes

Server Method: asTempAndRhData

Direction: Uplink

Note: There is no timestamp in this message. The server must assume the time it receives this message is the timestamp and this should almost always be within a few seconds of the actual time.

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor aggregate count in this case.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x01	0x01	0x01	0x01
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2]	Humidity Fractional	Fractional portion of humidity measurement in %	uint8_t	0x1E	0x1E	0x00	0x64
[3]	Humidity Integer	Integer portion of humidity measurement in %	uint8_t	0x01	0x01	0x00	0xFF
[4]	Temp Fractional	Fractional portion of temperature measurement in C	int8_t	0x41	0x41	0x00	0xFF

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[5]	Temp Integer	Integer portion of temperature measurement in C	int8_t	0x19	0x19	0xD8	0x55
[6]	Battery Capacity	Index for percentage of battery capacity remaining 0 0-5% 1 5-20% 2 20-40% 3 40-60% 4 60-80% 5 80-100%	uint8_t	0x05	0x02	0x00	0x05
[7:8]	AlarmMsg Count	Number of backlog alarm messages in sensor FLASH	uint16_t	0x0000	0x0000	0x0000	0x0FFF
[9:10]	BacklogMsg Count	Number of backlog non-alarm messages in sensor FLASH	uint16_t	0x0000	0x0000	0x0000	0x0FFF

Note: Refer to section 8 for details of the encoding scheme for 8-bit temperature and humidity data.

Example Message

```
Sent MSB First.
01 00 1E 01 41 19 02 00 00 00 00
```

5.3 Message Name: Send Temp and RH Aggregated Data Notification

Notification

Method Name: SendTempRHAggregatedData

Notes

Server Method: *asTempAndRhAggregate*

Direction: *Uplink*

The length of this message is variable.

MsgType, Options, Msg Counts, NumberReadings and Timestamp = 11 bytes

One Temperature and Humidity Reading = 4 bytes

Message Length = NumReading * 4 + 11. Since NumReadings max is 10: 10 x 4 = 40. 40 + 11 = 51.

It is important that this packet be 51 bytes or less since the max EU packet across datarates is 51 bytes in length.

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor configuration.

The timestamp is the time of the last sensor reading. The server must use the sensor read period parameter (part of the device configuration) to calculate the timestamps of the remaining data.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value	
[0]	MsgType	Message type.	Uint8_t	0x02	0x02	0x02	0x02	
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10	
[2]	AlarmMsg Count	Number of backlog alarm messages in sensor FLASH.	Uint8_t	0x00	0x00	0x00	0xFF	
[3:4]	BacklogMsg Count	Number of backlog non-alarm messages in sensor FLASH	uint16_t	0x0000	0x0000	0x0000	0x0FFF	
[5]	Battery Capacity	Index for percentage of battery capacity remaining	uint8_t	0x05	0x02	0x00	0x05	
		0						0-5%
		1						5-20%
		2						20-40%
		3						40-60%
		4						60-80%
5	80-100%							
[6]	Number Readings	Number of sensor readings in packet.	Uint8_t	-	0x02	0x01	0x0B	
[7:10]	Timestamp	Seconds since Jan 1 2015	uint32_t	-	0x0000 0064	0x0000 0000	0xFFFF FFFF	
[11]	Humidity Fractional	Fractional portion of humidity measurement in %	uint8_t	-	0x01	0x9D	0x64	
[12]	Humidity Integer	Integer portion of humidity measurement in %	uint8_t	-	0x1E	0x00	0x64	
[13]	Temp Fractional	Fractional portion of temperature measurement in C	int8_t	-	0x19	0x9D	0x63	
[14]	Temp Integer	Integer portion of temperature measurement in C	int8_t	-	0x41	0xD8	0x55	
[15]	Humidity Fractional	Fractional portion of humidity measurement in %	uint8_t	-	0x03	0x1E	0x64	
[16]	Humidity Integer	Integer portion of humidity measurement in %	uint8_t	-	0x1D	0x01	0xFF	
[17]	Temp Fractional	Fractional portion of temperature measurement in C	int8_t	-	0x19	0x19	0xFF	
[18]	Temp Integer	Integer portion of temperature measurement in C	int8_t	-	0x41	0xD8	0x55	

Note: Refer to section 8 for details of the encoding scheme for 8-bit temperature and humidity data.

Example Message

Sent MSB First.

```
02 00 00 00 00 00 05 02 00 00 00 64 01 1E 19 41 03 1D 19 41
```

For RTD sensors, 8-bit temperature and humidity data is replaced with 16-bit temperature data.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value	
[0]	MsgType	Message type.	uint8_t	0x02	0x02	0x02	0x02	
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10	
[2]	AlarmMsg Count	Number of backlog alarm messages in sensor FLASH.	uint8_t	0x00	0x01	0x00	0xFF	
[3:4]	BacklogMsg Count	Number of backlog non-alarm messages in sensor FLASH	uint16_t	0x0000	0x0123	0x0000	0x0FFF	
[5]	Battery Capacity	Index for percentage of battery capacity remaining		uint8_t	0x05	0x02	0x00	0x05
		0	0-5%					
		1	5-20%					
		2	20-40%					
		3	40-60%					
		4	60-80%					
[6]	Number Readings	Number of sensor readings in packet.	uint8_t	-	0x02	0x01	0x0B	
[7:10]	Timestamp	Seconds since Jan 1 2015	uint32_t	-	0x0000 0064	0x0000 0000	0xFFFF FFFF	
[11:12]	Temperature Fractional	Fractional portion of temperature measurement in C	int16_t	-	0x0001	0xFF9D	0x0063	
[13:14]	Temperature Integer	Integer portion of temperature measurement in C	int16_t	-	0x0019	0xFF38	0x0352	
[15:16]	Temperature Fractional	Fractional portion of temperature measurement in C	int16_t	-	0x0003	0xFF9D	0x0063	
[17:18]	Temperature Integer	Integer portion of temperature measurement in C	int16_t	-	0x0019	0xFF38	0x0352	

Note: Refer to section 9 for details of the encoding scheme for 16-bit temperature data.

Example Message

Sent MSB First.

```
02 00 01 01 23 02 02 00 00 00 64 00 01 00 19 00 03 00 19
```

5.4 Message Name: Send BackLog Message Notification

Notification

Method Name: SendBackLogMessage

Notes

Message Length = 10 bytes.

Server Method: asBacklogTempAndRh

Direction: Uplink

The timestamp is the time the sensor data was sampled.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x03	0x03	0x03	0x03
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2:5]	Timestamp	Seconds since Jan 1 2015	uint32_t	-	0x00000064	0x00000000	0xFFFFFFF
[6]	Humidity Fractional	Fractional value of humidity measurement in %	uint8_t	-	0x1E	0x00	0x63
[7]	Humidity Integer	Integer value of humidity measurement in %	uint8_t	-	0x01	0x00	0x64
[8]	Temperature Fractional	Fractional value of temperature measurement in C	int8_t	-	0x00	0x9D	0x63
[9]	Temperature Integer	Integer value of temperature measurement in C	int8_t	-	0x19	0xD8	0x55

Note: Refer to section 8 for details of the encoding scheme used for 8-bit temperature and humidity data.

Example Message

```
Sent MSB First.
03 00 00 00 00 64 1E 01 00 19
```

For RTD sensors, 8-bit temperature and humidity data is replaced with 16-bit temperature data.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x03	0x03	0x03	0x03
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2:5]	Timestamp	Seconds since Jan 1 2015	uint32_t	-	0x00000064	0x00000000	0xFFFFFFF
[6:7]	Temperature Fractional	Fractional portion of temperature measurement in C	int16_t	-	0x0062	0xFF9D	0x0063
[8:9]	Temperature Integer	Integer portion of temperature measurement in C	int16_t	-	0x0100	0xFF38	0x0352

Note: Refer to section 9 for details of the encoding scheme for 16-bit temperature data.

Example Message

```
Sent MSB First.
03 00 00 00 00 64 00 62 01 00
```

5.5 Message Name: Send BackLog Messages Notification

Notification

Method Name: SendBackLogMessages

Notes

Server Method: *asBacklogTempAndRhAggregate*

Direction: *Uplink*

The length of this message is variable.

The MsgType, NumberReadings, and Options = 3 bytes

The Timestamp, Temperature, and Humidity = 8 bytes

Message Length = NumReading * 8 + 3. Since NumReadings max is 6: 6 x 8 = 48. 48 + 3 = 51

It is important that this packet be 51 bytes or less since the max EU packet across datarates is 51 bytes in length.

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor configuration.

The *Timestamp* is the time the sensor data was sampled.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x04	0x04	0x04	0x04
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2]	Num Reading	Number of sensor-reading backlogs in packet	uint8_t	0x01	0x01	0x01	0x06
[3:6]	Timestamp	Seconds since Jan 1 2015	uint32_t	-	0x0000 0064	0x0000 0000	0xFFFF FFFF
[7]	Humidity Fractional	Fractional value of humidity measurement in %	uint8_t	-	0x1E	0x00	0x64
[8]	Humidity Integer	Integer value of humidity measurement in %	uint8_t	-	0x01	0x00	0x64
[9]	Temp Fractional	Fractional value of temperature measurement in C	int8_t	-	0x41	0x9D	0x64
[10]	Temp Integer	Integer value of temperature measurement in C	int8_t	-	0x19	0xD8	0x7D

Note: Refer to section 8 for details of the encoding scheme for 8-bit temperature and humidity data.

Example Message

Sent MSB First.

```
04 00 01 00 00 00 64 1E 01 41 19
```

For RTD sensors, 8-bit temperature and humidity data is replaced with 16-bit temperature data.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x04	0x04	0x04	0x04
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2]	Num Reading	Number of sensor reading backlogs in packet	uint8_t	0x01	0x01	0x01	0x06
[3:6]	Timestamp	Seconds since Jan 1 2015	uint32_t	-	0x0000 0064	0x0000 0000	0xFFFF FFFF
[7:8]	Temperature Fractional	Fractional portion of temperature measurement in C	int16_t	-	0x0010	0xFF9D	0x0063
[9:10]	Temperature Integer	Integer portion of temperature measurement in C	int16_t	-	0x0100	0xFF38	0x0352

Note: Refer to section 9 for details of the encoding scheme used for 16-bit temperature data.

Example Message

```
Sent MSB First.
04 00 01 00 00 00 64 00 10 01 00
```

5.6 Message Name: Send Sensor Config Simple Notification

Notification

Method Name: SendSensorConfigSimple

Notes

Message Length = 8 bytes

Server Method: asSensorConfigSimple

Direction: Uplink

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor configuration.

The *SensorAggregate* setting aggregates or collects sensor data every *ReadSensorPeriod* and only sends them over LoRa when the *SensorAggregate* number is reached. For example, if the *SensorAggregate* is set to 2 and the *ReadSensorPeriod* is set to 60 seconds, the data is sent over LoRa 2 x 60 seconds or every 120 seconds. This has a positive effect on battery life.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x05	0x05	0x05	0x05
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[2]	Battery Type	1	Zinc-Manganese Dioxide (Alkaline)	uint8_t	0x01	0x01	0x01
		2	Lithium/Iron Disulfide (Primary Lithium)				
[3:4]	ReadSensor Period	Period in seconds to read the sensor 0 Disabled	uint16_t	0x003C	0x003C	0x001E	0xFFFF
[5]	Sensor Aggregate	Number of readings to aggregate before sending on LoRa	uint8_t	0x01	0x01	0x01	0x0B
[6]	TempAlarm Enabled	False Disabled True Enabled	bool	0	0	0	1
[7]	HumidityAlarm Enabled	False Disabled True Enabled	bool	0	0	0	1

Note: Before enabling alarms, confirm the High Alarm Limits are set to greater values than Low Alarm Limits. These can be set via the sensor config advanced notification defined in section 5.7

Example Message

Sent MSB First.

```
05 00 01 00 00 01 00 00
```

5.7 Message Name: Send Sensor Config Advanced Notification

Notification

Method Name: SendSensorConfigAdvanced

Notes

Message Length = 16 bytes.

Server Method: asSensorConfigAdvanced

Direction: Uplink

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor configuration.

The *SensorAggregate* setting aggregates or collects sensor data every *ReadSensorPeriod* and only sends them over LoRa when the *SensorAggregate* number is reached. For example, if the *SensorAggregate* is set to 2 and the *ReadSensorPeriod* is set to 60 seconds, the data is sent over LoRa 2 x 60 seconds or every 120 seconds. This has a positive effect on battery life.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x06	0x06	0x06	0x06
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2]	Battery Type	1 Zinc-Manganese Dioxide (Alkaline)	uint8_t	0x01	0x01	0x01	0x02

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
		2 Lithium/Iron Disulfide (Primary Lithium)					
[3:4]	ReadSensor Period	Period in seconds to read the sensor 0 Disabled, sensor is not read	uint16_t	0x003C	0x003C	0x0000	0xFFFF
[5]	Sensor Aggregate	Number of readings to aggregate before sending on LoRa	uint8_t	0x01	0x01	0x01	0x0B
[6]	TempAlarms Enabled	True Temperature alarms enabled False Temperature alarms disabled	bool	0	0	0	1
[7]	HumidityAlarms Enabled	True Humidity alarms enabled False Humidity alarms disabled	bool	0	0	0	1
[8]	TempAlarm LimitLow	Temperature alarm limit – low in C	int8_t	0x00	0x00	0xD8	0x55
[9]	TempAlarm LimitHigh	Temperature alarm limit – high in C	int8_t	0x00	0x32	0xD8	0x55
[10]	HumidityAlarm LimitLow	Humidity alarm limit - low in %	int8_t	0x0A	0x0A	0x00	0x64
[11]	HumidityAlarm LimitHigh	Humidity alarm limit - high in %	int8_t	0x50	0x50	0x00	0x64
[12:13]	LED_BLE	Flash period in seconds when in BLE connection 0 No flash	uint16_t	0x000A	0x000A	0x0000	0xFFFF
[14:15]	LED_Heartbeat	Flash period in seconds 0 No flash 65535 LED in LoRa Tx/Rx mode Tx – green Rx – orange	uint16_t	0x000A	0x000A	0x0000	0xFFFF

Note: High Alarm Limits must be set to greater values than Low Alarm Limits.

Example Message

Sent MSB First.

```
06 00 01 00 00 01 00 00 00 32 0A 50 00 0A 00 0A
```

5.8 Message Name: Send FW Version Notification

Notification

Method Name: SendFWVersion

Notes

Message Length = 11 bytes.

Server Method: asFwVersion

Direction: Uplink

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor configuration.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x07	0x07	0x07	0x07
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2]	Year	Version year	0x00	0x11	0x00	0x00	0xFF
[3]	Month	Version month	0x00	0x01	0x01	0x01	0x0C
[4]	Day	Version day	0x00	0x01	0x01	0x01	0x1F
[5]	Version Major	Version major	0x00	0x01	0x00	0x01	0xFF
[6]	Version Minor	Version minor	0x00	0x00	0x00	0x00	0xFF
[7:10]	Part Number	Part number of firmware	uint32_t	0x0000 0000	0x0049 3E6F	0x0000 0000	0xFFFF FFFF

Example Message

Sent MSB First.

```
07 00 00 01 01 00 00 00 49 3E 6F
```

5.9 Message Name: Send Open Closed Config

Notification

Method Name: SendOpenClosedConfig

Notes

Message Length = 10 bytes.

Server Method: asSendOpenClosedConfig

Direction: Uplink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value								
[0]	MsgType	Message type	uint8_t	0x08	0x08	0x08	0x08								
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x10								
[2]	Operating Mode	The operating mode of the Open/Closed sensor <table border="1"> <tr> <td>0</td> <td>Door Sensor</td> </tr> <tr> <td>1</td> <td>RFU – Pushbutton</td> </tr> </table>	0	Door Sensor	1	RFU – Pushbutton	enum	0x00	0x00	0x00	0x01				
0	Door Sensor														
1	RFU – Pushbutton														
[3]	LoRa Notification Options	Determines what Open/Closed sensor notifications are sent over LoRa BIT POSITION <table border="1"> <tr> <td>0</td> <td>Open notifications</td> </tr> <tr> <td>1</td> <td>Closed notifications</td> </tr> <tr> <td>2</td> <td>Resend notifications</td> </tr> <tr> <td>3</td> <td>Cancel notifications</td> </tr> </table>	0	Open notifications	1	Closed notifications	2	Resend notifications	3	Cancel notifications	bitfield	0x07	0x00	0x00	0x0F
0	Open notifications														
1	Closed notifications														
2	Resend notifications														
3	Cancel notifications														
[4:5]	Open Dwell Time	Sets the duration in seconds after which an Open sensor status is annunciated	uint16_t	0x0005	0x0005	0x0005	0x03FF								
[6:7]	Closed Dwell Time	Sets the duration in seconds after which a Closed sensor status is annunciated	uint16_t	0x0005	0x0005	0x0005	0x03FF								
[8]	Resend Interval	Defines the duration in minutes after which status messages are resent	uint8_t	0x01	0x01	0x01	0xFF								
[9]	Debounce Adjust	Defines the duration in milliseconds that the Open/Closed sensor status must remain consistent before the status is changed	uint8_t	0x00	0x10	0x00	0xFF								

Example Message

Sent MSB First.

```
08 00 00 00 00 05 00 05 01 10
```

5.10 Message Name: Send Open Closed Status

Notification

Method Name: SendOpenClosedStatus

Notes

Message Length = 6 bytes.

Server Method: asSendOpenClosedStatus

Direction: Uplink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x09	0x09	0x09	0x09
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x10
[2]	Operating Mode	The operating mode of the Open/Closed sensor 0 Door Sensor 1 RFU – Pushbutton	enum	0x00	0x00	0x00	0x01
[3]	State	The state of the Open/Closed sensor. 0 Closed 1 Open	enum	0x00	0x00	0x00	0x01
[4]	Alert Cancellation	Indicates whether this is an alert cancellation 0 No 1 Yes	enum	0x00	0x00	0x00	0x01
[5]	Counter	Indicates the number of messages that have been sent since this state was entered	uint8_t	0x00	0x05	0x00	0xFF

Example Message

Sent MSB First.

09 00 00 00 00 05

5.11 Message Name: Send Battery Voltage

Notification

Method Name: SendBatteryVoltage

Notes

Message Length = 4 bytes.

Server Method: asSendBatteryVoltage

Direction: Uplink

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x0A	0x0A	0x0A	0x0A
[1]	Options	See the Message Options section	enum	0x00	0x00	0x00	0x10
[2]	Voltage Fractional	Fractional part of the last measured battery voltage	int8_t	0x00	0x0A	0x00	0x63

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[3]	Voltage Integer	Integer part of the last measured battery voltage	int8_t	0x00	0x03	0x00	0x03

Note: Refer to section 8 for details of the encoding scheme used for 8-bit voltage data.

Example Message

Sent MSB First.

0A 00 0A 03

5.12 Message Name: Send RTD Data Notification

Notification

Method Name: SendRTDData

Notes

Message Length = 11 bytes

Server Method: asRTDData

Direction: Uplink

Note: There is no timestamp in this message. The server must assume the time it receives this message is the timestamp and this should almost always be within a few seconds of the actual time.

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor aggregate count in this case.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value	
[0]	MsgType	Message type	uint8_t	0x0B	0x0B	0x0B	0x0B	
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x01	0x00	0x10	
[2:3]	Fractional Temperature	Fractional portion of temperature measurement in C	int16_t	-	0x0000	0xFF9D	0x0063	
[4:5]	Integer Temperature	Integer portion of temperature measurement in C	int16_t	-	0x0010	0xFF38	0x0352	
[6]	Battery Capacity	Index for percentage of battery capacity remaining	uint8_t	0x05	0x02	0x00	0x05	
		0						0-5%
		1						5-20%
		2						20-40%
		3						40-60%
		4						60-80%
5	80-100%							

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[7:8]	AlarmMsg Count	Number of backlog alarm messages in sensor FLASH	uint16_t	0x0000	0x0000	0x0000	0x0FFF
[9:10]	BacklogMsg Count	Number of backlog non-alarm messages in sensor FLASH	uint16_t	0x0000	0x0000	0x0000	0x0FFF

Note: Refer to section 9 for details of the encoding scheme used for 16-bit temperature data.

Example Message

Sent MSB First.

```
0B 01 00 00 00 10 02 00 00 00 00
```

5.13 Message Name: Send RTD Config Advanced Notification

Notification

Method Name: SendRTDConfig

Notes

Message Length = 15 bytes

Server Method: asRTDConfig

Direction: Uplink

A *Sensor Configuration Error* in the Options byte means that the sensor is operating in North America at DR0, where the maximum LoRa payload is 11-bytes, but it is configured with an aggregate count > 1, which results in a payload greater than 11-bytes. In this case, the sensor stores the data it can't send to FLASH. The server should change the sensor configuration.

The *SensorAggregate* setting aggregates or collects sensor data every *ReadSensorPeriod* and only sends them over LoRa when the *SensorAggregate* number is reached. For example, if the *SensorAggregate* is set to 2 and the *ReadSensorPeriod* is set to 60 seconds, the data is sent over LoRa 2 x 60 seconds or every 120 seconds. This has a positive effect on battery life.

Parameters

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
[0]	MsgType	Message type	uint8_t	0x0C	0x0C	0x0C	0x0C
[1]	Options	See the Sensor-To-Server Message Options section	bitfield	0x00	0x00	0x00	0x10
[2]	Battery Type	1	Zinc-Manganese Dioxide (Alkaline)	uint8_t	0x01	0x01	0x02
		2	Lithium/Iron Disulfide (Primary Lithium)				
[3:4]	ReadSensor Period	Period in seconds to read the sensor		uint16_t	0x003C	0x003C	0x0000
		0	Disabled, sensor is not read				
[5]	Sensor Aggregate	Number of readings to aggregate before sending on LoRa	uint8_t	0x01	0x01	0x01	0x0B
[6]	TempAlarms	True	Temperature alarms enabled	bool	0	0	1

Byte Index	Name	Description	Type	Default Value	Example Value	Min Value	Max Value
	Enabled	False	Temperature alarms disabled				
[7:8]	TempAlarm LimitLow	Temperature alarm limit – low in C	int16_t	0xFF38	0x0000	0xFF38	0x0352
[9:10]	TempAlarm LimitHigh	Temperature alarm limit – high in C	int16_t	0x00FA	0x0032	0xFF38	0x0352
[11:12]	LED_BLE	Flash period in seconds when in BLE connection 0 No flash	uint16_t	0x000A	0x000A	0x0000	0xFFFF
[13:14]	LED_Heartbeat	Flash period in seconds 0 No flash 65535 LED in LoRa Tx/Rx mode Tx – green Rx – orange	uint16_t	0x000A	0x000A	0x0000	0xFFFF

Note: High Alarm Limits must be set to greater values than Low Alarm Limits.

Example Message

```
Sent MSB First.
0C 00 01 00 3C 01 00 00 00 00 32 00 0A 00 0A
```

6 APPENDIX A – UPLINK MESSAGE REVISION HISTORY

Firmware Version	Message Name	Message Identifier	Notes
3.1	SendTempRHData	0x01	Initial production release
	SendTempRHAggregatedData	0x02	
	SendBackLogMessage	0x03	
	SendBackLogMessages	0x04	
	SendSensorConfigSimple	0x05	
	SendSensorConfigAdvanced	0x06	
	SendFWVersion	0x07	
4.2	All	-	Added Sensor Request for Server Time bit Added Sensor Reset bit
5.7	SendOpenClosedConfig	0x08	Added for Open/Closed sensor type support
	SendOpenClosedStatus	0x09	
6.0	All	-	Added Sensor Fault bit
	SendTempRHAggregatedData	0x02	Extended for RTD sensor type support
	SendBackLogMessage	0x03	
	SendBackLogMessages	0x04	Added for battery voltage measurement Added for RTD sensor type support
	SendBatteryVoltage	0x0A	
	SendRTDDData	0x0B	
6.1	SendRTDConfig	0x0C	Added for targeted backlog retrieval
	TargetedBacklogRetrievalFIFO	0x0F	
	TargetedBacklogRetrievalLIFO	0x10	

7 APPENDIX B – DOWNLINK MESSAGE REVISION HISTORY

Firmware Version	Message Name	Message Identifier	Notes
3.1	GenericDataRetrieval	0x01	Initial production release
	SetUTC	0x02	
	Config	0x03	
	HeaterControl	0x04	
	Backoff	0x05	
	BacklogRetrievalFIFO	0x06	
	FormatLogFlash	0x07	
	SetAlkalineThresholds	0x08	
	SetLithiumThresholds	0x09	
4.5	EnterBootloader	0xFF	Added Enter Bootloader downlink command
4.7	BacklogRetrievalLIFO	0x0A	Added for better interaction with backlog data
	CancelBacklogRetrieval	0x0B	
5.7	All	-	Send Open/Closed Sensor Config bit
	OpenClosedConfig	0x0D	Added for Open/Closed sensor support
6.0	All	-	Added Sensor Reset bit
			Added Sensor Advertise bit
			Added Battery Voltage Request bit
	SetAlkalineThresholds	0x08	Messages deprecated
	SetLithiumThresholds	0x09	
	RTDConfig	0x0E	Added for RTD sensor support
6.1	TargetedBacklogRetrievalFIFO	0x0F	Added for targeted backlog retrieval
	TargetedBacklogRetrievalLIFO	0x10	

8 APPENDIX C – DECODING 8-BIT DATA

Data returned by Internal Temperature/Humidity Sensor, External Temperature Sensor and Open/Closed sensor variants is returned in 8-bit format. The data is returned in two bytes, with one representing the integer portion of the data, and the other the fractional portion of the data. Each byte needs to be decoded separately then the fractional data divided by 100.

The sum of the two gives the resultant value.

This is shown in [Figure 1](#), where YY is the decoded integer data value, and ZZ the decoded fractional data value, divided by 100.

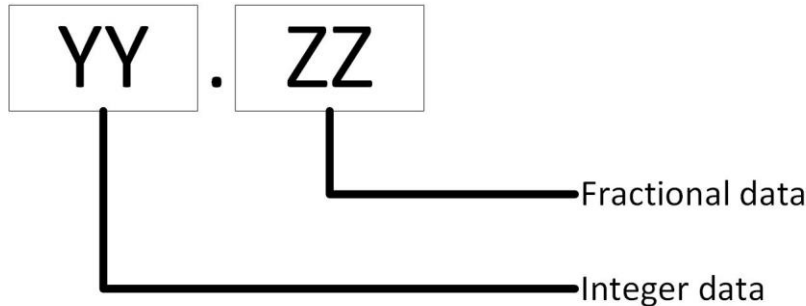


Figure 1 – 8-bit data integer and fractional data

A fractional value of 1 ($1/100 = 0.01$) and integer value of 1 would decode as 1.01.

8.1 Unsigned 8-bit data

For voltage and humidity data, unsigned 8-bit data is used due to these data types never becoming negative values.

Note: Unsigned 8-bit data can express between the values of 0 to 255. When converting this data, care must be taken to ensure signed types are not used during the conversion.

8.2 Signed 8-bit data

For temperature data, signed 8-bit data is used to allow negative values to be expressed. In this case, both integer and fractional data are returned from the sensor as negative values.

Note: Signed 8-bit data can express between the values of -128 to 127. Care should be taken when decoding the data to ensure signed 8-bit types are used.

9 APPENDIX D – DECODING 16-BIT DATA

Data returned by RTD sensor variants is returned in signed 16-bit format. The data is returned in four bytes, with two bytes representing the integer portion of the data, and two bytes the fractional portion of the data. For both integer and fractional data, the data is split into MSB (Most Significant Byte) and LSB (Least Significant Byte) parts.

When decoding the data, each MSB part must be multiplied by 256, then added to the LSB to give the final 16-bit value.

The final fractional value must then be divided by 100 and added to the final integer value to give the signed 16-bit temperature value.

This is shown in [Figure 2](#), where WW is the MSB of the integer data multiplied by 256.

YY, the MSB of the fractional data, has been multiplied by 256, then added to the LSB of the fractional data and the result divided by 100.

The final 16-bit temperature value is the result of the sum of the integer and fractional values.

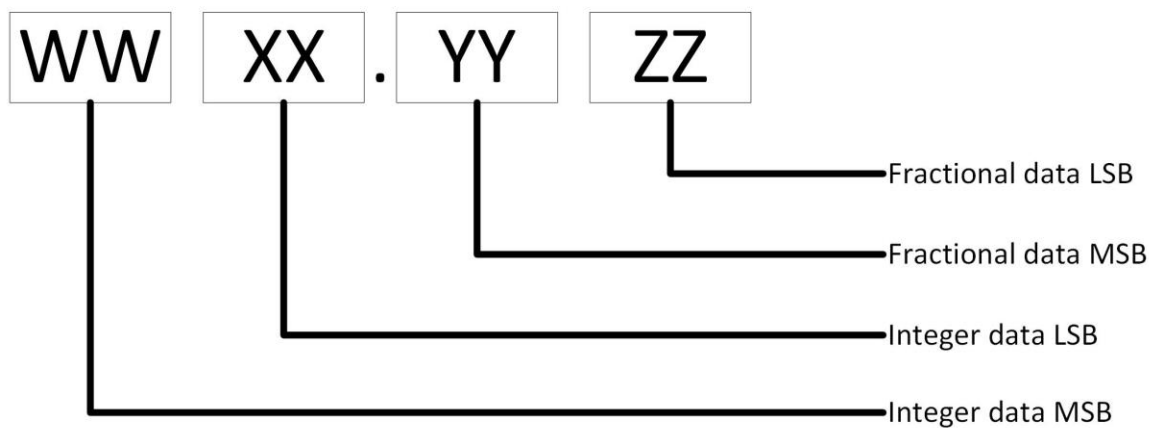


Figure 2 – 16-bit data integer and fractional data

For example, a value of -256.25 would be returned in hexadecimal format with an integer data MSB of 0xFF, an integer data LSB of 0x0, a fractional data MSB of 0xFF and a fractional data LSB of 0xE7.

The integer and fractional MSB values are both multiplied by 256, giving 0xFF00 and 0xFF00.

The adjusted integer MSB is added to the integer LSB, giving 0xFF00 (-256).

The adjusted fractional MSB is added to the fractional LSB, giving 0xFEE7 (-25).

The sum of the fractional MSB and LSB is divided by 100, giving -0.25.

These are then added together, resulting in -256.25.

10 REVISION HISTORY

Version	Date	Notes	Contributor(s)	Approver
1.0	17 Dec 2017	Initial Release		Jonathan Kaye
2.0	16 March 2018	Updated to latest messages		Chris Hofmeister
2.1	11 April 2018	Corrected message type values, put them in numeric order		Mark Monson
2.2	3 August 2018	Updated to new template. Updates to BatteryCapacity.		Seokwoo Yoon
2.3	5 Sept. 2018	Updated options byte in uplink messages for version 4.2 FW		Chris Hofmeister
2.4	25 Jan 2019	Updated <i>seconds</i> to <i>milliseconds</i> for Options_Si702x	Mark Monson	Jonathan Kaye
2.5	15 Aug 2019	Updated formatting and aggregate uplink	Mark Monson	Jonathan Kaye
2.6	9 October 2019	Updated content of Send Temp and RH Aggregated Data Notification and Send Sensor Config Simple Notification	Greg Leach	Chris Boorman
2.7	16 April 2020	Added the following: <ul style="list-style-type: none"> ▪ Details of Open/Closed sensor uplink and downlink messages. ▪ Details of Battery Voltage request uplink message ▪ Details of Reset downlink option ▪ Details of Sensor Fault uplink option ▪ Detail to Set UTC downlink message ▪ Details of 16-bit temperature values ▪ Uplink Message Revision History ▪ Downlink Message Revision History 	Greg Leach	Chris Boorman
2.8	22 June 2020	Added details of Targeted Backlog retrieval commands	Greg Leach	Chris Boorman
2.9	24 July 2020	Added appendices detailing decoding of temperature, humidity and voltage data	Greg Leach	Chris Boorman
2.10	5 August 2020	Added the following: <ul style="list-style-type: none"> • Updated RTD sensor type related command details • Added footnotes referring to decoding schemes for temperature, humidity and voltage data to appropriate command descriptions • Added footnotes describing the required values for Alarm Limits to appropriate command descriptions • Corrected example value for Battery Type for Sensor Config Notification command description 	Greg Leach	Chris Boorman