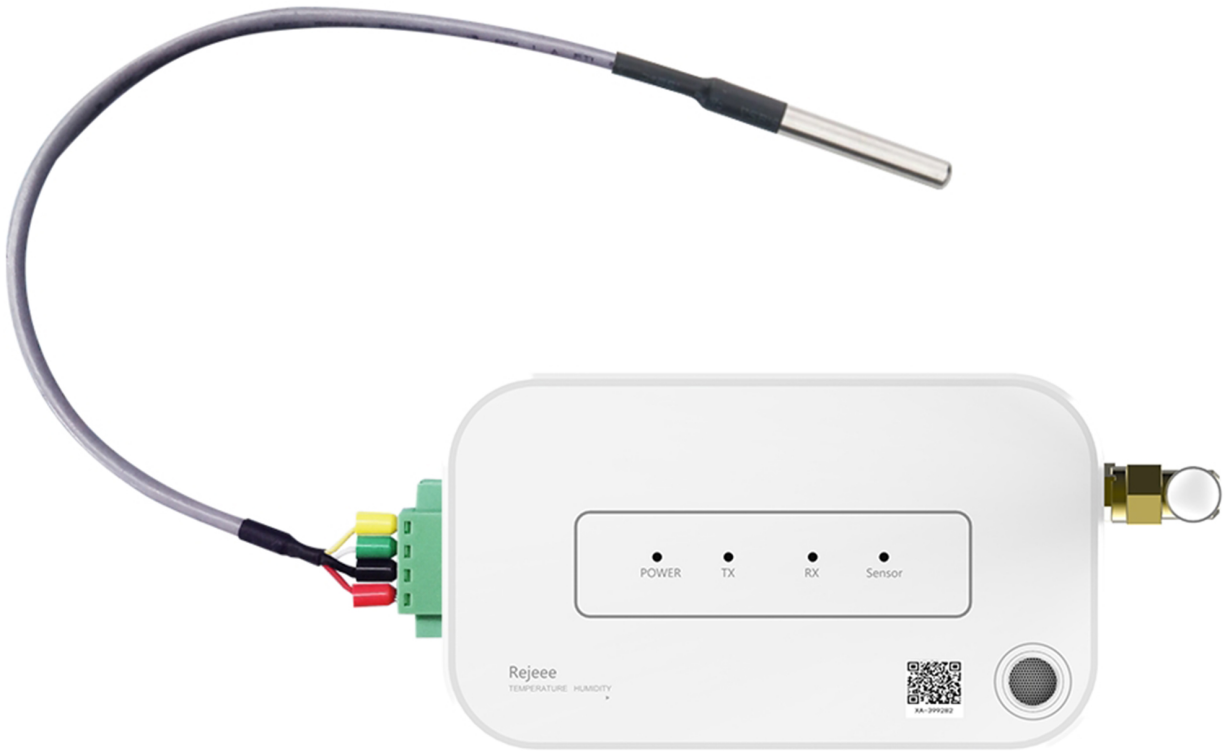
User Manual for SL111 Series, external temp/humi sensor

LoRaWAN Temperature and Humidity Sensor V1.0



Jiangsu Rejeee Intelligent Technology Co., Ltd
Address: No. 20,Xinghuo Road, Jiangbei District, Nanjing, China
Email: Jullie.zheng@rejeee.com
Tel: 0086 158 6180 7793
Web: http://www.rejeee.com

1. General Information

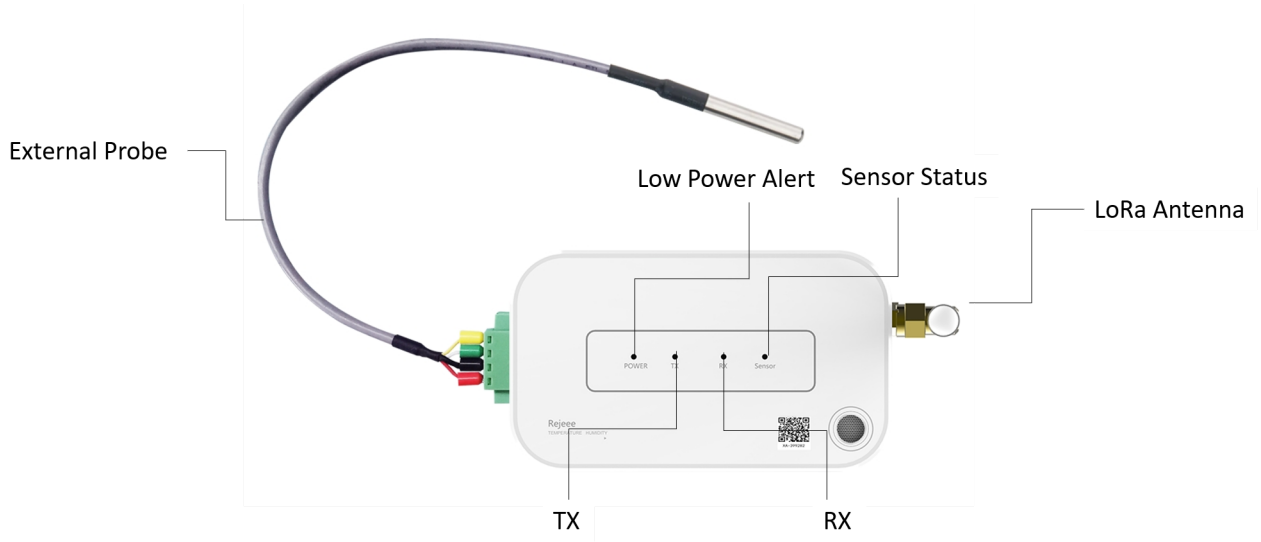
SL111 is long range low power temperature and humidity sensor based on Semtech SX1262/SX1268.

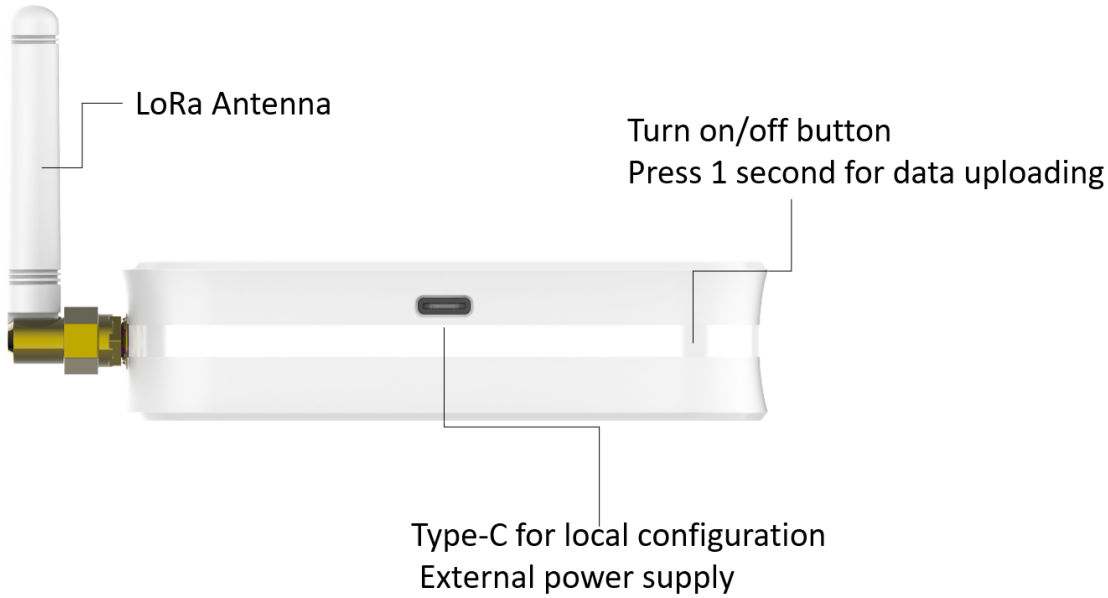
Sensor Type	Product Number
Built-in and external SHT30	SL111CN, SL111EU,SL111US,SL111AS

1.1 Main features:

Sensirion High Sensitivity Sensor
Type-C for Local Configuration
Internal Battery Up to 5 Years(LCP=5s，LFT=600s @SF9)
LoRa Long Range Low Power
Both built in and external SHT 30

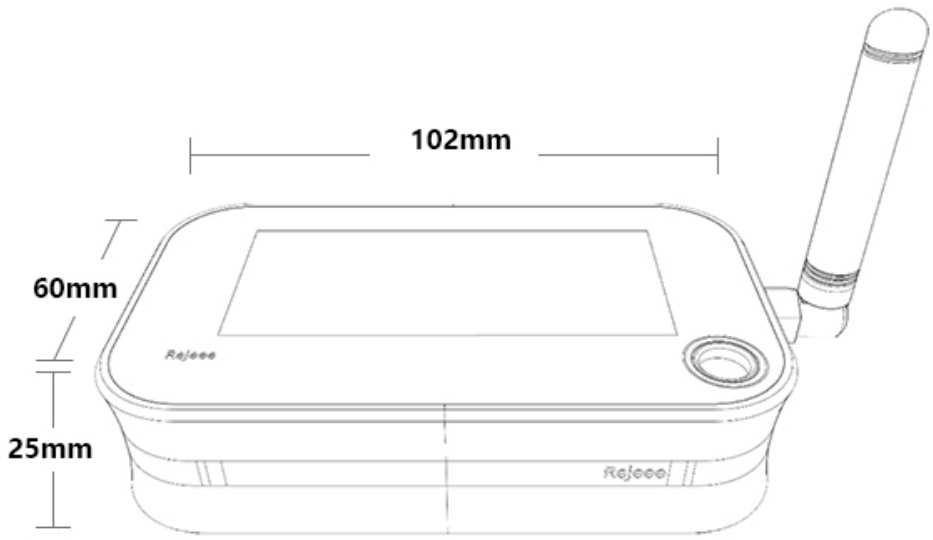
1.2 Details





Parameters	Feature
CPU	STM32L151
Wireless	LoRaWAN(SX1262/SX1268)
Encryption	AES128
Battery	Built-in Li-battery (Changeable, and No Recharge)
Battery Capacity	5400mAh
Working Temperature	-45℃~+ 85℃
Working Humidity	0-100%RH
Communication	Half duplex
Accuracy	Temperature : ±0.3℃, Humidity：±3%RH
Lifespan	5 Years(Every 10 Minutes for data uploading @SF9)
Data Speed	300bps-62.5k bps
Size	102mm*60mm*25mm
TX Power	22dBm Max
RX Sensitivity	-140 dBm
Frequency	SX1268: CN470 SX1262: EU868 / US915 / AS923

1.3 Size: 102mm*60mm*25mm

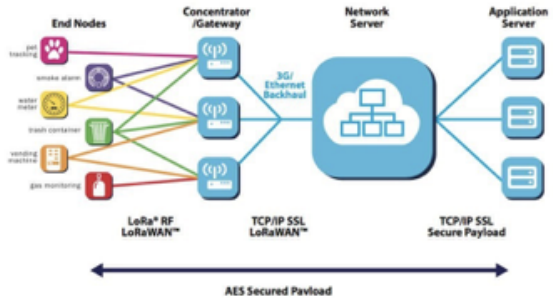


1.4 Installation:

Hang on the wall

2. Connect to LoRaWAN Network

LoRaWAN Network Structure

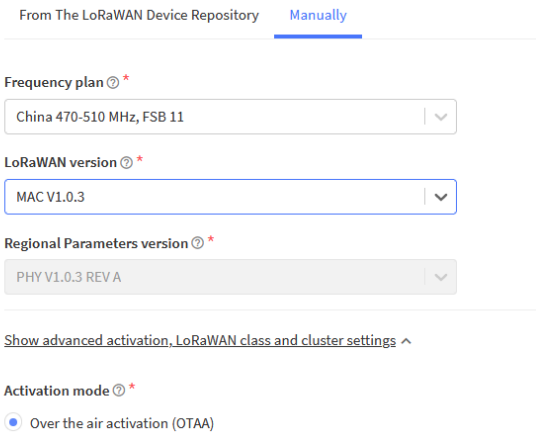
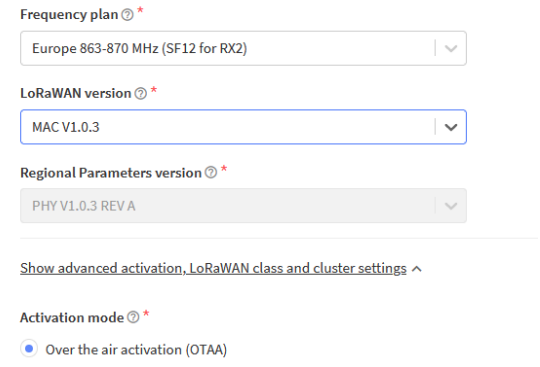
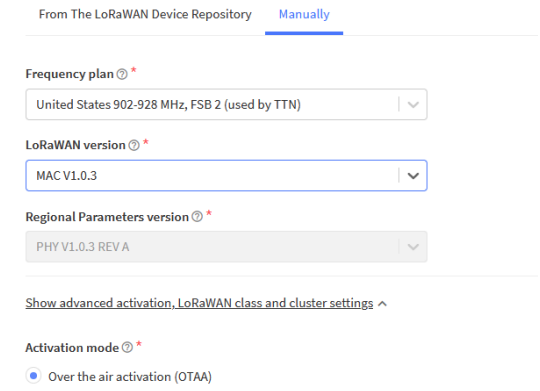
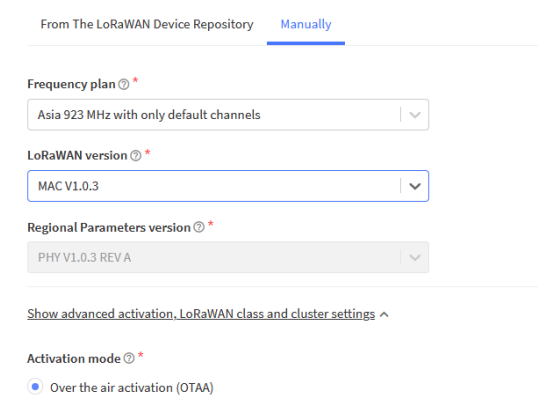


SL111 temperature and humidity sensor is based on standard LoRaWAN Class A, so you can connect to any LoRaWAN network as below:

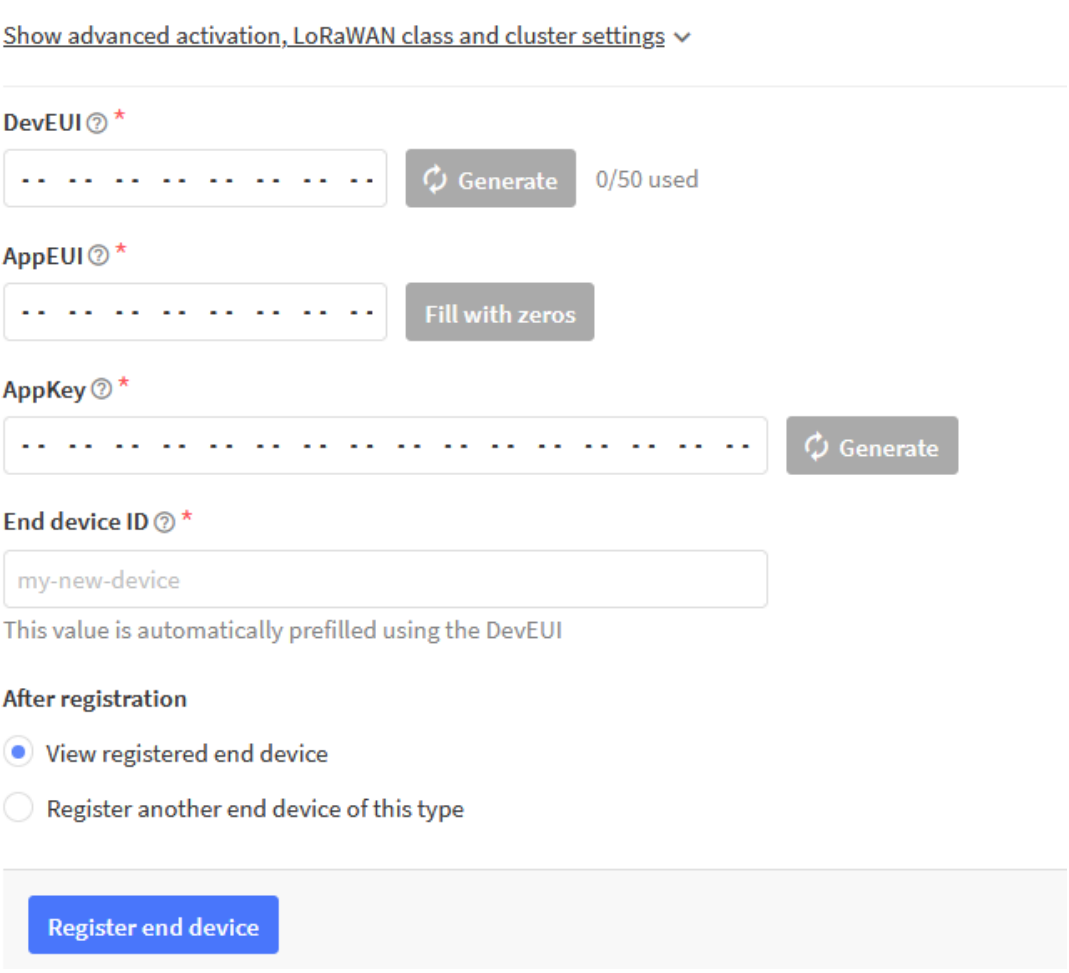
SL111 sensor data uplink format with LoRaWAN OTAA, OTAA parameter as below:
AppEUI: CACBB80000000001
AppKey: 11223344556677889900AABBCCDDEEFF
DevEUI: Customer can find DevEUI on the product, also you can read DevEUI through Rejeee SensorTool
SensorTool

You can also choose ABP, for ABP you can find the parameter as below:
APP KEY/APP EUI: 11223344556677889900AABBCCDDEEFF
DevEUI: Customer can find DevEUI on the product, also you can read DevEUI through Rejeee SensorTool
SensorTool

For example: Connecting to The Things Network, please make sure choose manually connect and OTAA as below:

Sensor	LoRaWAN
SL111CN	
SL111EU	
SL111US	
SL111AS	

2.1 Set ID and Key



Data analysis example for JavaScript:

```
function decodeUplink(input) {
    var obj = {};
    var warnings = [];
    var len = input.bytes ? input.bytes.length : 0;
    var offset = 0, dtype = 0, dlen = 0;

    do {
        dtype = input.bytes[offset++];
        if (0xFF == dtype) {
            /* 0xFF is ACK from Device */
            obj.ackcmd = input.bytes[offset++];
            obj.ackstatus = input.bytes[offset++];
        } else if (0x00 === dtype) {
            /* first is device information(0x00) */
            obj.battery = (input.bytes[offset++] & 0x1F);
            obj.res = input.bytes[offset++];
        } else if (0x01 == dtype) {
            offset += 8;
        } else if (0x02 == dtype) {
            offset += 8;
        } else if (0x03 == dtype) {
            offset += 2;
        } else if (0x04 == dtype) {
            /* temperature sensor, value unit is 0.1 */
```

```
            obj.temperature = (((input.bytes[offset] & 0x80 ? input.bytes[offset] - 0x100 :
input.bytes[offset]) << 8) + input.bytes[offset + 1]) / 10;
            offset += 2;
        } else if (0x05 == dtype) {
            /* humidity sensor, value unit is 1 %RH */
            obj.humidity = input.bytes[offset++];
        } else if (0x06 == dtype) {
            obj.oxygen = input.bytes[offset++];
        } else if (0x07 == dtype) {
            offset += 4;
        } else if (0x08 == dtype) {
            /* ignore */
            offset += 4;
        } else if (0x09 == dtype) {
            /* ignore */
            offset += 1;
        } else if (0x14 == dtype) {
            /* Mutil-temperature sensor, value unit 0.1 */
            dlen = input.bytes[offset++];
            if (dlen >= 2) {
                obj.temperature1 = (((input.bytes[offset] & 0x80 ? input.bytes[offset] - 0x100
: input.bytes[offset]) << 8) + input.bytes[offset + 1]) / 10;
                offset += 2;
                dlen -= 2;
            }

            if (dlen >= 2) {
                obj.temperature2 = (((input.bytes[offset] & 0x80 ? input.bytes[offset] - 0x100
: input.bytes[offset]) << 8) + input.bytes[offset + 1]) / 10;
                offset += 2;
                dlen -= 2;
            }
            if (dlen > 0) {
                offset += dlen;
            }
        } else if (0x15 == dtype) {
            /* Mutil-humidity sensor, value unit 1 %RH */
            dlen = input.bytes[offset++];
            if (dlen >= 1) {
                obj.humidity1 = input.bytes[offset++];
                dlen -= 1;
            }

            if (dlen >= 1) {
                obj.humidity2 = input.bytes[offset++];
                dlen -= 1;
            }

            if (dlen > 0) {
                offset += dlen;
            }
        } else {
            /* ignore all > 0x10 */
            if(dtype > 0x10){
                dlen = input.bytes[offset++];
                if (dlen > 0) {
```

```
                    offset += dlen;
                }
            }
        }
    } while (len > offset);

    return {
        data: obj,
        warnings: warnings
    };
}
```

3. Wireless LoraWAN Sensor Data Format

LoRaWAN Format:
Picture as below, FRMPayload is sensor data.

PHYPayload:

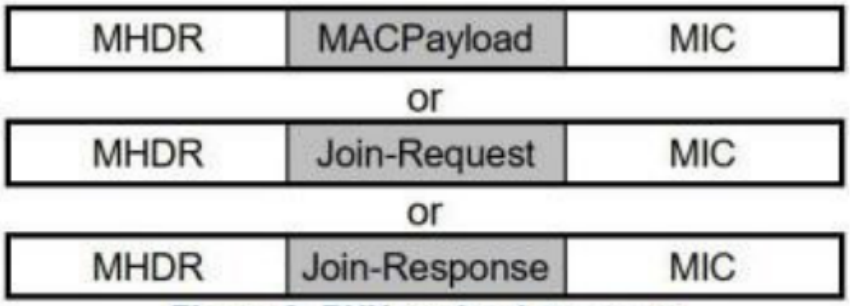


Figure 6: PHY payload structure

MACPayload:

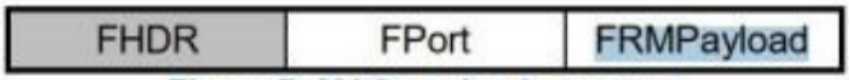


Figure 7: MAC payload structure

MHDR	FHDR	FPort	FRMPayload=Sensor Data(Message)			MIC
			Data 1	...	Data N	4 Bytes
			Type+Data N Bytes	Type+Data N Bytes	Type+Data N Bytes	

4. Sensor Data Definition

4.1 Multi-temperature (0x14)

Adapt N-way temperature according to length, and if N is 1, the basic temperature type 0x04 can be directly used.
If N>1 channel temperature needs to be transmitted, merge similar data items in the following order.

Type	Length	Value	Value	Value
1 Byte	1 Byte	int16_t	...	int16_t
0x14	2*N	No.1 temp	...	No. N temp

4.2 Multi-humidity (0x15)

According to the length, N channels of humidity can be adapted. If N is 1, the basic humidity type 0x05 can be directly used.
If N>1 channel of humidity needs to be transmitted, merge similar data items in the following order.

Type	Length	Value	Value	Value
1 Byte	1 Byte	uint8_t	...	uint8_t
0x15	N	No. 1 humi	...	No. N humi

FRMPayload Example: 007F101404033C033A15024445
00 7F10 Device Information, 7F is battery level = 31， which means battery is 100%, we measure battery level from 0-31, which means 0-100%. Version is 0x10, which means Rejeee temp/humi sensor.
033C and 033A is temperature, is 0x033C = 828 = 82.8℉, 0x033A = 826 =82.6℉. ℉ is only for US market, for other market sensor will send ℃ by default. The first temp is from built-in sensor, and the second one is from external sensor.
44 and 45 is humidity, is 0x44 = 68 = 68 %RH, is 0x45 = 69 = 69 %RH. The first humi is from built-in sensor and the second one is from external sensor.

5. Local Configuration:

Note: Factory reset data uploading is every 10 mins, customers can change data uploading frequency as below:
Connect sensor with a USB-C cable to computer for local configuration, through local configuration, you can change the packet frequency. Refer [SensorTool Manual](http://doc.rejeee.com/web/#/32?page_id=339