
LMDS200 -- LoRaWAN Microwave Radar Distance Sensor User Manual

last modified by Mengting Qiu

on 2024/03/07 08:41

Table of Contents

1. Introduction	3
1.1 What is LMDS200 Microwave Radar DistanceSensor	3
1.2 Features	4
1.3 Radar probe specification	4
1.4 Storage & Operation Temperature	5
1.5 Applications	5
1.6 Installation	5
1.7 Pin mapping and power on	6
2. Operation Mode	6
2.1 How it works	6
2.2 Example to use for LoRaWAN network	7
2.3 Uplink Payload	12
2.3.1 Device Status, FPORT=5	12
2.3.2 Sensor Configuration, FPORT=4	13
2.3.3 Distance, Uplink FPORT=2	13
2.3.4 Decoder in TTN V3	15
2.4 Show data on Datacake	15
3. Configure LMDS200 via AT Command or LoRaWAN Downlink	21
3.1 Set Transmit Interval Time(0x01)	22
3.2 Set Alarm Transmit Interval Time(0x0D)	22
3.3 Set Alarm Distance (0xA2)	23
3.4 Enter/Exit Alarm Mode (0xA3)	23
3.5 Set Interrupt Mode(0x06)	23
4. Battery & Power Consumption	24
5. FAQ	24
5.1 How to use AT Command to configure LMDS200	24
5.2 How to upgrade the firmware?	27
5.3 How to change the LoRa Frequency Bands/Region	27
6. Trouble Shooting	27
6.1 AT Command input doesn't work	27
7. Order Info	28
8. Packing Info	28
9. Support	28



Table of Contents:

- [1. Introduction](#)
 - [1.1 What is LMDS200 Microwave Radar DistanceSensor](#)
 - [1.2 Features](#)
 - [1.3 Radar probe specification](#)
 - [1.4 Storage & Operation Temperature](#)
 - [1.5 Applications](#)
 - [1.6 Installation](#)
 - [1.7 Pin mapping and power on](#)
- [2. Operation Mode](#)
 - [2.1 How it works](#)
 - [2.2 Example to use for LoRaWAN network](#)
 - [2.3 Uplink Payload](#)
 - [2.3.1 Device Status, FPORT=5](#)
 - [2.3.2 Sensor Configuration, FPORT=4](#)
 - [2.3.3 Distance, Uplink FPORT=2](#)
 - [2.3.4 Decoder in TTN V3](#)
 - [2.4 Show data on Datacake](#)
- [3. Configure LMDS200 via AT Command or LoRaWAN Downlink](#)
 - [3.1 Set Transmit Interval Time\(0x01\)](#)
 - [3.2 Set Alarm Transmit Interval Time\(0x0D\)](#)
 - [3.3 Set Alarm Distance \(0xA2\)](#)
 - [3.4 Enter/Exit Alarm Mode \(0xA3\)](#)
 - [3.5 Set Interrupt Mode\(0x06\)](#)
- [4. Battery & Power Consumption](#)
- [5. FAQ](#)
 - [5.1 How to use AT Command to configure LMDS200](#)
 - [5.2 How to upgrade the firmware?](#)
 - [5.3 How to change the LoRa Frequency Bands/Region](#)
- [6. Trouble Shooting](#)
 - [6.1 AT Command input doesn't work](#)
- [7. Order Info](#)
- [8. Packing Info](#)
- [9. Support](#)

1. Introduction

1.1 What is LMDS200 Microwave Radar DistanceSensor

The Dragino LMDS200 is a **LoRaWAN Microwave Radar distance sensor**. It uses **24Ghz Microwave** to detect the distance between sensor and different objects. Compare vs ultrasonic or Lidar measurement method,

Microwave Radar is **more reliable for condensation / dusty environment**. It can sense correct distance even there is water or thick dust on top of the sensor.

The LMDS200 can be applied to scenarios such as horizontal distance measurement, parking management system, object proximity and presence detection, intelligent trash can management system, robot obstacle avoidance, automatic control, sewer, etc.

LMDS200 can **measure two distances**: the closest object and next object behind the closest one.

LMDS200 supports **Alarm Feature**, user can set the LMDS200 to uplink data in a short interval when the distance is out of configured range.

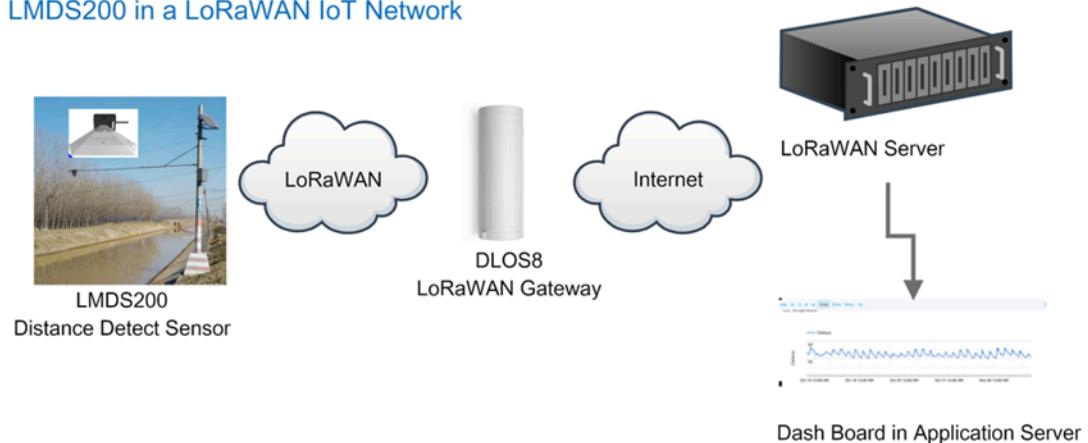
The LoRa wireless technology used in LMDS200 allows device to send data and reach extremely long ranges at low data-rates. It provides ultra-long range spread spectrum communication and high interference immunity whilst minimizing current consumption.

LMDS200 is powered by 8500mAh Li-SOCl2 battery, it is designed for long term use up to 5 years.

Each LMDS200 is pre-load with a set of unique keys for LoRaWAN registrations, register these keys to local LoRaWAN server and it will auto connect after power on.

*Battery life depends on how often to send data, please see [battery analyzer](#).

LMDS200 in a LoRaWAN IoT Network



1.2 Features

- LoRaWAN 1.0.3 Class A
- Ultra-low power consumption
- Microwave Radar for distance detection
- Short uplink interval for Distance Alarm
- Monitor Battery Level
- Bands: CN470/EU433/KR920/US915/EU868/AS923/AU915/IN865
- AT Commands to change parameters
- Uplink on periodically
- Downlink to change configure
- 8500mAh Battery for long term use
- Wall Mountable
- Outdoor Use

1.3 Radar probe specification

- Measuring Method: FMCW
- Frequency: 24.000 ~ 24.500 GHz

- Measurement output power: 6dBm
- Measure range: 0.5 ~ 20m
- Accuracy: $\pm 0.1\text{m}$
- Resolution: 0.01m
- Horizontal Angel: 78°
- Vertical Angel: 23°

1.4 Storage & Operation Temperature

-20°C to +85°C

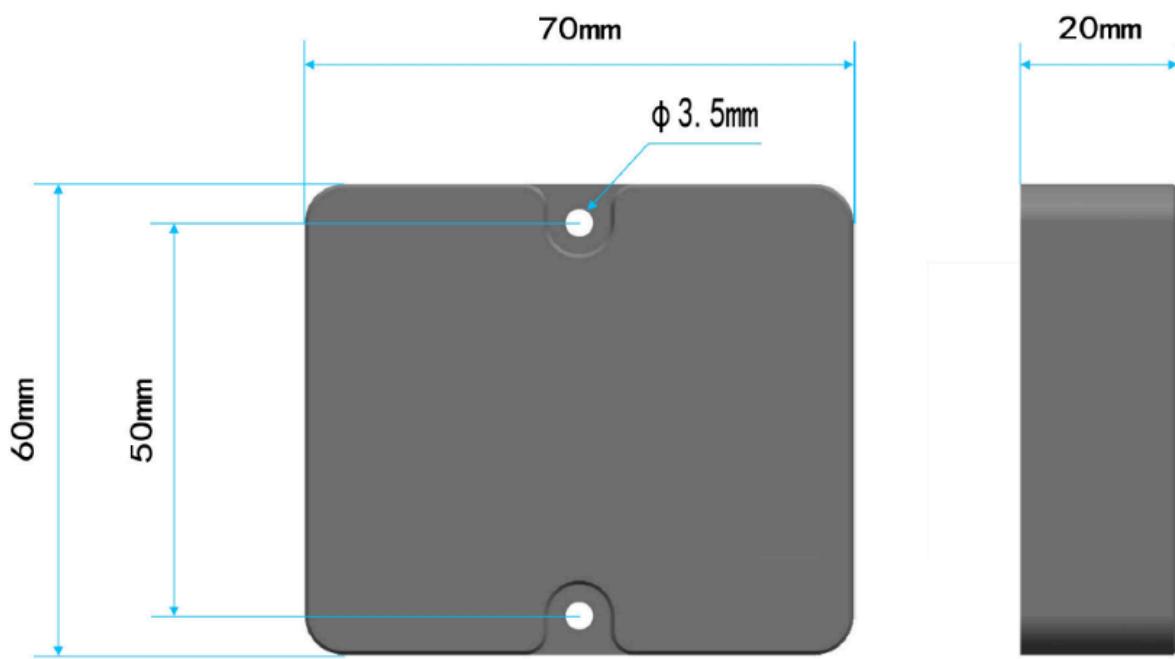
1.5 Applications

- Horizontal distance measurement
- Liquid level measurement
- Parking management system
- Object proximity and presence detection
- Intelligent trash can management system
- Robot obstacle avoidance
- Automatic control
- Sewer
- Bottom water level monitoring

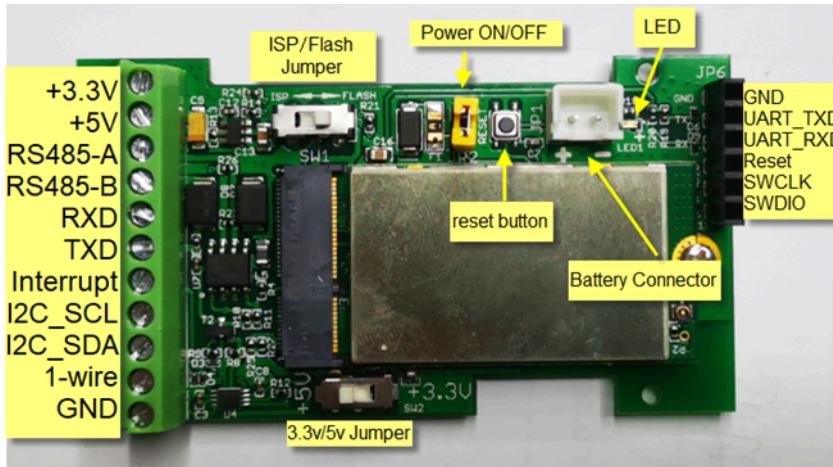
1.6 Installation

Sensor measure direction and angle is as below. When install the sensor, please make sure the sensor direct to object.





1.7 Pin mapping and power on



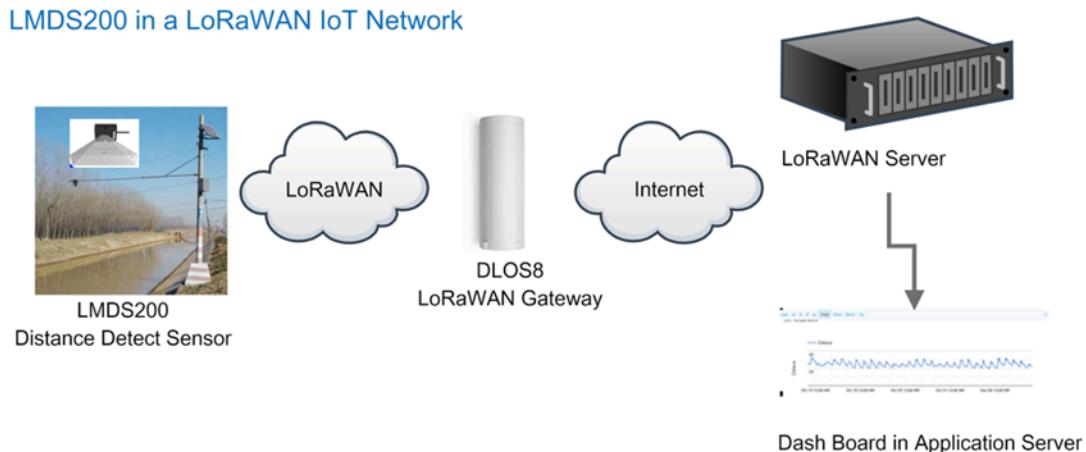
2. Operation Mode

2.1 How it works

Each LMDS200 is shipped with a worldwide unique set of OTAA keys. To use LMDS200 in a LoRaWAN network, user needs to input the OTAA keys in the LoRaWAN network server. So LMDS200 can join the LoRaWAN network and start to transmit sensor data.

2.2 Example to use for LoRaWAN network

Following is an example for how to join the [TTN v3 LoRaWAN Network](#). Below is the network structure; we use the [DLOS8](#) as a LoRaWAN gateway in this example.

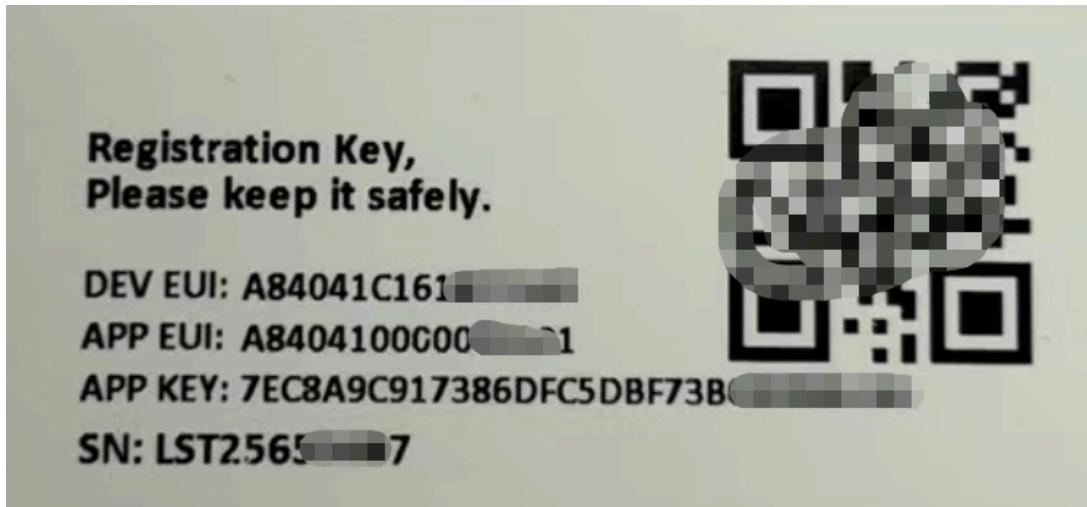


- In this user case, the LMDS200 is installed on top of river to detect the water level and send the level info to the LoRaWAN server. The LMDS200 will uplink different types of messages to the LoRaWAN server. See [Uplink payload](#) for detail.

Assume the LoRaWAN Gateway DLOS8 is already set to connect to the [TTN V3 network](#). We need to add the LMDS200 device in TTN V3:

Step 1: Create a device in TTN with the OTAA keys from LMDS200.

Each LMDS200 is shipped with a sticker with the default device keys, user can find this sticker in the box. it looks like below.



Users can enter these keys in the LoRaWAN Server portal. Below is the TTN V3 screenshot:

Add APP EUI in the application.

The screenshot shows the 'Add application' page of the The Things Stack Community Edition. The top navigation bar includes tabs for Overview, Applications (which is highlighted), Gateways, and Orgs. The main form fields are:

- Owner***: davidhuang
- Application ID***: my-new-application
- Application name**: My new application
- Description**: Description for my new application

An optional note below the form states: "Optional application description; can also be used to save notes about the application". At the bottom left of the form is a blue button labeled "Create application".

The screenshot shows a LoRaWAN management interface with two main sections: a device details page at the top and a registration form below it.

Device Details Page:

- Header:** CCC ID: 123, 4 End devices, 2 Collaborators, 2 API keys, Created 95 days ago.
- General Information:** Application ID: 123, Created at: Feb 2, 2021 11:12:30, Last updated at: Apr 30, 2021 11:00:33.
- Activity Log:** Shows a list of live data messages with timestamps and descriptions, such as "Forward data message to Application Server".
- Buttons:** Search by ID, Import end devices, + Add end device.

Registration Form:

- Section 1: Select the end device**
 - From The LoRaWAN Device Repository (selected)
 - Manually

Brand* dropdown: Dragino Technology Co.,...
Model* dropdown: Type to search...
Cannot find your exact end device? link
- Section 2: Enter registration data**

Please choose an end device first to...
Register end device button

A red arrow points from the "Cannot find your exact end device?" link to the dropdown menu, which lists various device models: LBT1, LDDS20, LDDS75, LDS01, LGT92, LHT65, LSE01, and LSN50-V2. A red arrow also points from the "Please choose an end device first to..." message to the "Register end device" button.

2. Enter registration data

Frequency plan ⓘ *

Select...

The frequency plan used by the end device

AppEUI ⓘ *

..... 88

The AppEUI uniquely identifies the owner of the end device. If no AppEUI is provided by the device manufacturer (usually for development), it can be filled with zeros.

You can also choose to create the device manually.

Register end device

From The LoRaWAN Device Repository [Manually](#)

Preparation

Activation mode *

Over the air activation (OTAA)
 Activation by personalization (ABP)
 Multicast
 Do not configure activation

LoRaWAN version ⓘ *

Select...

Network Server address

eu1.cloud.thethings.network

Application Server address

eu1.cloud.thethings.network

External Join Server ⓘ

Add APP KEY and DEV EUI

Register end device

From The LoRaWAN Device Repository Manually 1

Frequency plan 2

LoRaWAN version 3

Regional Parameters version 4

Show advanced activation, LoRaWAN class and cluster settings 5

DevEUI 6
 0/50 used

AppEUI 7

AppKey 8

End device ID 9

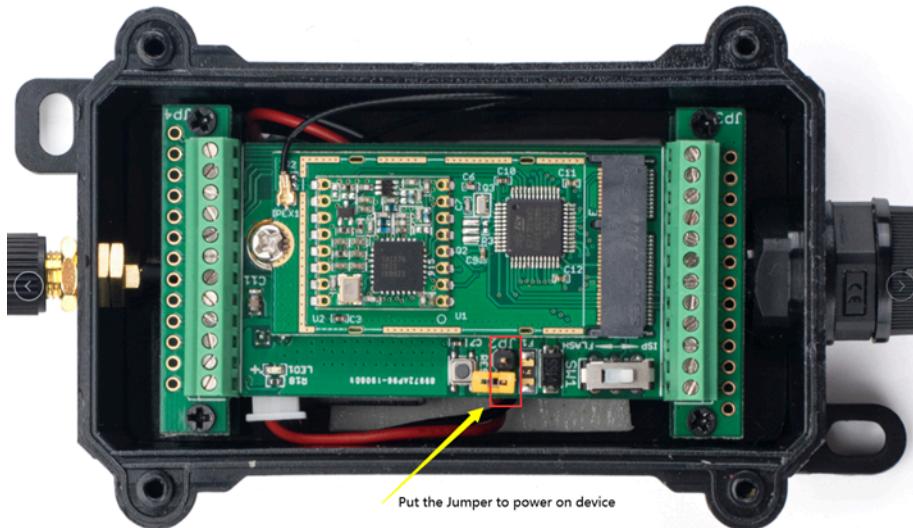
This value is automatically prefilled using the DevEUI

After registration

View registered end device
 Register another end device of this type

Step 2: Power on LMDS200

Put a Jumper on JP2 to power on the device. (The Switch must be in FLASH position).



Put the jumper to power on LMDS200 and it will auto-join to the TTN V3 network. After join success, it will start to upload sensor data to TTN V3 and the user can see it in the panel.

2.3 Uplink Payload

Uplink payloads have two types:

- Distance Value: Use FPORT=2
- Other control commands: Use other FPORT fields.

The application server should parse the correct value based on FPORT settings.

2.3.1 Device Status, FPORT=5

Include device configure status. Once LMDS200 Joined the network, it will uplink this message to the server.

Users can also use the downlink command (0x26 01) to ask LMDS200 to resend Device Status.

Device Status (FPORT=5)					
Size(bytes)	1	2	1	1	2
Value	Sensor Model	Firmware Version	Frequency Band	Sub-band	BAT

lmds200
ID: lmds200

↑ 1 ↓ n/a • Last activity 6 seconds ago

Overview Live data Messaging Location Payload formatters Claiming General settings

Time	Type	Data preview
16:32:27	Fail to send webhook	Error:undefined:undefined
↑ 15:32:27	Forward uplink data message	Payload: { Bat: 3.304, Fre_band: "EU868", Sensor_model: "LMDS200", Sub_band: 0, Ver: 100 } 0C 01 00 01 00 0C E8 FPort: 5
↑ 15:32:22	Forward join-accept message	
GD 15:32:20	Accept join-request	

- **Sensor Model:** For LMDS200, this value is 0x0C
- **Firmware Version:** 0x0100, Means: v1.0.0 version
- **Frequency Band:**

0x01: EU868

0x02: US915

0x03: IN865

0x04: AU915

0x05: KZ865

0x06: RU864

0x07: AS923

0x08: AS923-1

0x09: AS923-2

0x0a: AS923-3

0x0b: CN470

0x0c: EU433

0x0d: KR920

0x0e: MA869

- Sub-Band:**

- AU915 and US915:value 0x00 ~ 0x08
- CN470: value 0x0B ~ 0x0C
- Other Bands: Always 0x00

- Battery Info:**

Check the battery voltage.

Ex1: 0xB45 = 2885mV

Ex2: 0xB49 = 2889mV

2.3.2 Sensor Configuration, FPORT=4

LMDS200 will only send this command after getting the downlink command (0x26 02) from the server.

Sensor Configuration FPORT=4				
Size(bytes)	3	1	4	1
Value	TDC (unit:sec)	ATDC (unit:min)	Alarm Settings	Interrupt Settings

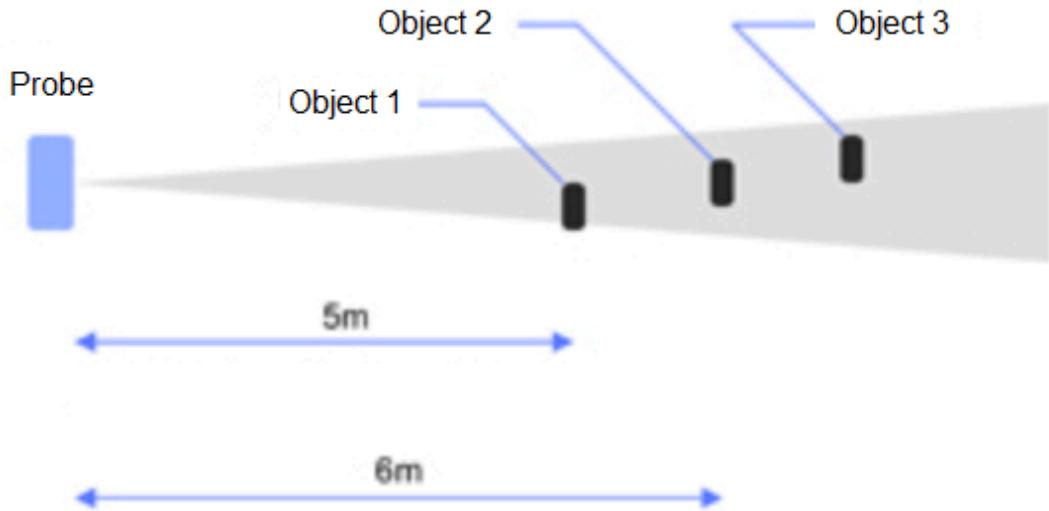
2.3.3 Distance, Uplink FPORT=2

LMDS200 will send this uplink **after** Device Status once join the LoRaWAN network successfully. And LMDS200 will:

- periodically send this uplink every 1 hour (TDC time), this interval [can be changed](#).
- periodically send this uplink every 1 minute in Alarm Mode.
- send this uplink while there is [interrupt event](#).

Uplink Payload totals 11 bytes.

Distance Value, FPORT=2				
Size(bytes)	2	2	2	1
Value	BAT	Object1 Distance	Object2 Distance	Status & Alarm
Status & Alarm field				
Size(bit)	6	1	1	1
Value	DALARM Counter	Distance Alarm 0: Normal Value 1: Distance Alarm	Interrupt Alarm 0: No Alarm 1: external Interrupt Alarm	



Object1 Distance:

Distance between sensor probe to the first object. (unit: cm)

For example, if the data you get from the register is 0x00 0x73, the distance between the sensor and the measured object is **0073(H) = 115 (D) = 115 cm.**

Notice: There are two special values for object 1 distance:

- 0x0001: Probe not detected
- 0x0002: Reading Invalid (exceed the valid range of the probe)

Object2 Distance:

Distance between sensor probe to the second object. (unit: cm)

DALARM Counter : Alarm Counter.

↑ 14:30:18	Forward uplink data message	Payload: { Bat: 3.328, DALARM_count: 19, Distance_alarm: 1, Interrupt_alarm: 0, dis1: 115, dis2: 164 }	8D 00 00 73 00 A4 4E	FP
① 14:29:18	Fail to send webhook	Error:undefined:undefined		
① 14:29:18	Fail to send webhook	Error:undefined:undefined		
① 14:29:18	Fail to send webhook	Error:undefined:undefined		
① 14:29:18	Fail to send webhook	Error:undefined:undefined		
↑ 14:29:18	Forward uplink data message	Payload: { Bat: 3.328, DALARM_count: 18, Distance_alarm: 1, Interrupt_alarm: 0, dis1: 115, dis2: 164 }	8D 00 00 73 00 A4 4A	FP
① 14:28:18	Fail to send webhook	Error:undefined:undefined		
① 14:28:18	Fail to send webhook	Error:undefined:undefined		
① 14:28:18	Fail to send webhook	Error:undefined:undefined		
① 14:28:18	Fail to send webhook	Error:undefined:undefined		
↑ 14:28:18	Forward uplink data message	Payload: { Bat: 3.33, DALARM_count: 17, Distance_alarm: 1, Interrupt_alarm: 0, dis1: 115, dis2: 164 }	8D 02 00 73 00 A4 46	FPo

2.3.4 Decoder in TTN V3

The screenshot shows the TTN V3 web interface with the 'Payload formatters' tab selected. On the left sidebar, 'End devices' is selected. In the main area, under 'Uplink', it says 'These payload formatters are executed on uplink messages from this end device and take precedence over application level payload formatters.' Below this, 'Formatter type' is set to 'Javascript'. The 'Formatter parameter' section contains the following code:

```
1 function decodeUplink(input) {  
2     return {  
3         data: [  
4             bytes: input.bytes  
5         ],  
6         warnings: [],  
7         errors: []  
8     }  
9 }
```

A red box highlights the first two lines of the code. At the bottom right is a 'Save changes' button.

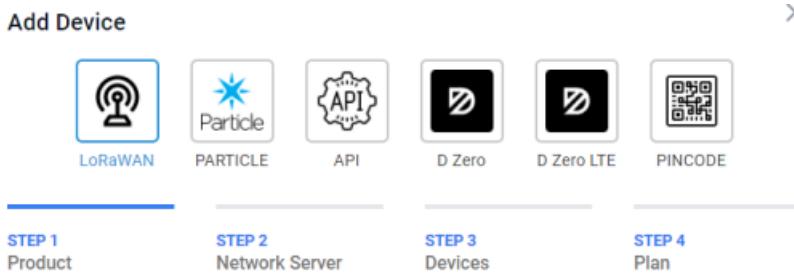
Please check the decoder from this link: <https://github.com/dragino/dragino-end-node-decoder>

2.4 Show data on Datacake

Datacake IoT platform provides a human-friendly interface to show the sensor data, once we have sensor data in TTN V3, we can use Datacake to connect to TTN V3 and see the data in Datacake. Below are the steps:

Step 1: Link TTNV3 to [DATACAKE](#)

Step 2: Configure LMDS200 in Datacake



Datacake Product

You can add devices to an existing product on Datacake, create a new empty product or start with one of the templates. Products allow you to share the same configuration (fields, dashboard and more) between devices.

The screenshot shows the 'New Product' creation screen. It has three options: 'New Product from template' (Create new product from a template), 'Existing Product' (Add devices to an existing product), and 'New Product' (Create new empty product). The 'New Product' option is highlighted with a blue box and a red arrow labeled '1' pointing to it. Below this is a section for 'New Product' with a 'Product Name' input field containing 'LDS03A', a red arrow labeled '2' pointing to it, and a 'Next' button at the bottom right with a red arrow labeled '3' pointing to it.

Network Server

Please choose the LoRaWAN Network Server that your devices are connected to.

The screenshot shows a list of network servers:

- The Things Stack V3**
TTN V3 / Things Industries
Uplinks Downlinks
- The Things Network V2**
The old Things Network
Uplinks Downlinks
- helium** Helium
Uplinks Downlinks
- LORIOT**
Uplinks Downlinks
- kerlink** Kerlink Wanesy
Uplinks

Showing 1 to 5 of 8 results

Previous Next

Add Devices

Enter one or more LoRaWAN Device EUIs and the names they will have on Datacake.

New: You can now upload a CSV file with either one column (just the device's DevEUI) or two columns (DevEUI and Name), which will populate the form below.

ⓘ Drag and drop a .csv file here or click to choose one

DEVEUI	NAME
99 55 66 33 22 44 11 46 8 bytes	LDS03A
+ Add another device 1 2 3	

[Back](#)[Next](#)

DATACAKE

Fleet > LDS03A

LDS03A

Serial Number 9955663322441140	Last update Never
Dashboard	History
Downlinks	Debug
Rules	Permissions

[Configuration](#) Configuration

General Configuration

Device Name	<input type="text" value="LDS03A"/>
-------------	-------------------------------------

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

Payload Decoder

When your devices sends data, the payload will be passed to the payload decoder, alongside the event's name. The payload decoder then transforms it to measurements.

```

1> function onEvent(e){}
2> var ss=(bytes[0]<0x20)?'TRUE':'FALSE'
3> var bb=(bytes[1]<0x20)?'OPEN':'CLOSE'
4> var dd=(bytes[2]<0x20)?'00000000000000000000000000000000':bytes[2]+bytes[3]+bytes[4].toString('0')
5> var ee=bytes[4]+bytes[5]+bytes[6]+bytes[7]+bytes[8].toString('0')
6> var ff=bytes[9]+bytes[10]+bytes[11]+bytes[12]+bytes[13]+bytes[14].toString('0')
7> var gg=bytes[15]+bytes[16]+bytes[17]+bytes[18]+bytes[19]+bytes[20].toString('0')
8> var hh=bytes[21]+bytes[22]+bytes[23]+bytes[24]+bytes[25]+bytes[26].toString('0')
9> var ii=bytes[27]+bytes[28]+bytes[29]+bytes[30]+bytes[31]+bytes[32].toString('0')
10> var jj=bytes[33]+bytes[34]+bytes[35]+bytes[36]+bytes[37]+bytes[38].toString('0')
11>
12> function getif(c_num){
13>   if(parseInt(c_num)<10)
14>     c_num = "0" + c_num
15>
16>   return c_num
17>
18> function getDate(str){
19>   var c_date
20>   if(str > 9999999999)
21>     c_date = new Date(parseInt(str))
22>   else
23>     c_date = new Date(parseInt(str * 1000))
24>
25>   var c_year = c_date.getFullYear(),
26>   c_month = c_date.getMonth(),
27>   c_day = c_date.getDate(),
28>   c_hour = c_date.getHours(),
29>   c_min = c_date.getMinutes(),
30>   c_sec = c_date.getSeconds()
31>
32>   var c_time = c_year + '-' + getif(c_month) + '-' + getif(c_day) + ' ' + getif(c_hour) + ':' + getif(c_min) + ':' + getif(c_sec)
33>
34>   return c_time
35>
36>

```

Payload

Port: 1

Try Decoder

Output

console.log Output

Recognized measurements

1 → Red arrow pointing to the 'Try Decoder' button.

2 → Red arrow pointing to the 'Save' button.

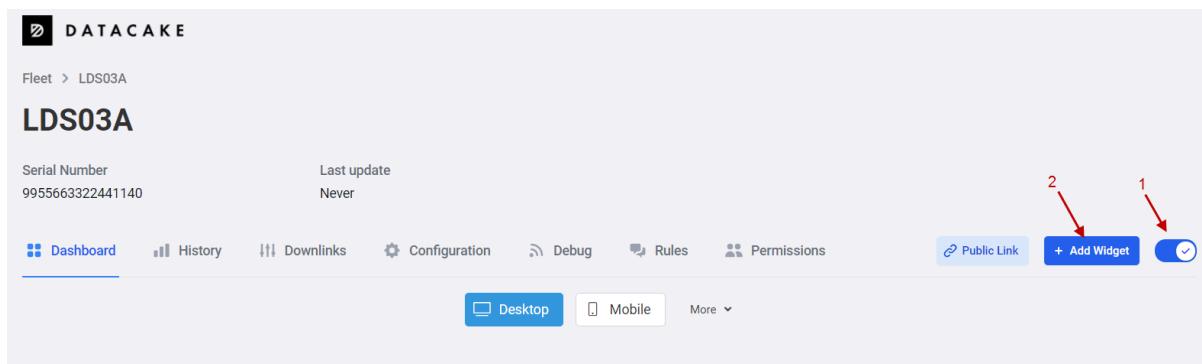
Fields

Fields describe the data the device will store.

NAME	2	IDENTIFIER	TYPE	CURRENT VALUE	LAST UPDATE	
BAT		BAT	Float	0	5 minutes ago	⋮
DOOR_OPEN_TIMES		DOOR_OPEN_TIMES	Float	0	3 minutes ago	⋮
LAST_DOOR_OPEN_DURATION		LAST_DOOR_OPEN_DURATION	Float	0	3 minutes ago	⋮
ALARM		ALARM	Boolean	False	a few seconds ago	⋮
DOOR_OPEN_STATUS		DOOR_OPEN_STATUS	Boolean	False	a few seconds ago	⋮

1 → Red box around the first row (BAT).

2 → Red number above the second column.



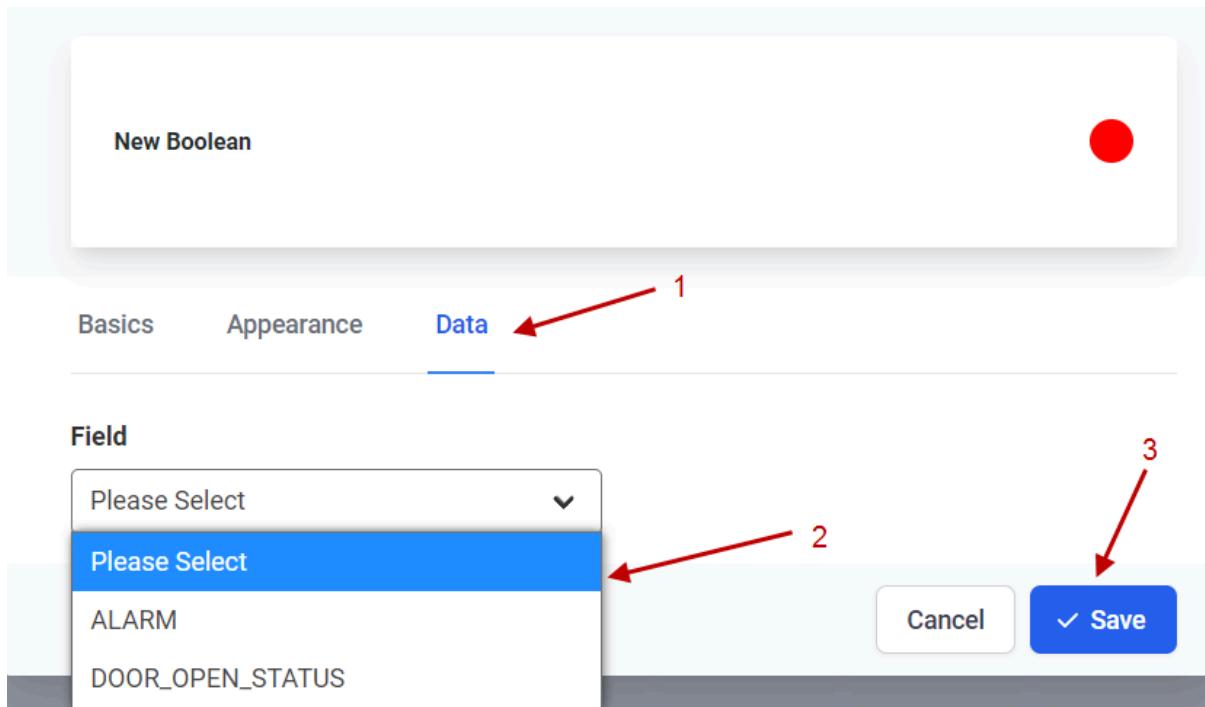
The screenshot shows the 'Data' tab configuration dialog. At the top, tabs for Basics, Data (selected), Appearance, Gauge, and Timeframe are visible. Below the tabs, a section labeled 'Field' contains a dropdown menu with the following options:

- Please Select
- BAT
- DOOR_OPEN_TIMES
- LAST_DOOR_OPEN_DURATION

A red arrow labeled '1' points to the 'Data' tab. A red arrow labeled '2' points to the 'BAT' option in the dropdown menu. A red arrow labeled '3' points to the 'Save' button at the bottom right of the dialog.

Below the field selection, there's a 'Decimal Places' input field containing the value '2'. At the bottom right of the dialog are 'Cancel' and 'Save' buttons, with the 'Save' button highlighted by a red arrow labeled '3'.

Edit Boolean Widget



DATACAKE

Fleet > LDS03A

LDS03A

Serial Number: 9955663322441140 Last update: Fri Nov 12 2021 16:51:53 GMT+0800

Dashboard History Downlinks Configuration Debug Rules Permissions Public Link Add Widget

Desktop Mobile More

BAT	0
DOOR_OPEN_TIMES	1
LAST_DOOR_OPEN_DURATION	0
DOOR_OPEN_STATUS	1 (red dot)
ALARM	0 (red dot)

3. Configure LMDS200 via AT Command or LoRaWAN Downlink

Use can configure LMDS200 via AT Command or LoRaWAN Downlink.

- AT Command Connection: See [FAQ](#).
- LoRaWAN Downlink instruction for different platforms: [IoT LoRaWAN Server](#)

There are two kinds of commands to configure LMDS200, they are:

- [General Commands](#).

These commands are to configure:

- General system settings like: uplink interval.
- LoRaWAN protocol & radio related command.

They are same for all Dragino Device which support DLWS-005 LoRaWAN Stack(**Note****). These commands can be found on the wiki: [End Device AT Commands and Downlink Command](#)

- [Commands special design for LMDS200](#)

These commands only valid for LMDS200, as below:

3.1 Set Transmit Interval Time(0x01)

Feature: Change LoRaWAN End Node Transmit Interval.

AT Command: AT+TDC

Command Example	Function	Response
AT+TDC=?	Show current transmit Interval	30000 OK the interval is 30000ms = 30s
AT+TDC=60000	Set Transmit Interval	OK Set transmit interval to 60000ms = 60 seconds

Downlink Command: 0x01

Format: Command Code (0x01) followed by 3 bytes time value.

If the downlink payload=0100003C, it means set the END Node's Transmit Interval to 0x00003C=60(S), while type code is 01.

- Example 1: Downlink Payload: 0100001E // Set Transmit Interval (TDC) = 30 seconds
- Example 2: Downlink Payload: 0100003C // Set Transmit Interval (TDC) = 60 seconds

3.2 Set Alarm Transmit Interval Time(0x0D)

Feature: Change LoRaWAN End Node Alarm Transmit Interval. Default Value: 1 minute

AT Command: AT+ATDC

Command Example	Function	Response
AT+ATDC=?	Show current alarm transmit Interval	1 OK the interval is 1 minute
AT+ATDC=1	Set Alarm Transmit Interval	OK Set alarm transmit interval to 1 minute

Downlink Command: 0x0D

Format: Command Code (0x0D) followed by 1 byte for time value.

If the downlink payload=0D02, it means set the END Node's Alarm Transmit Interval to 2 minutes, while type code is 0D.

3.3 Set Alarm Distance (0xA2)

LMDS200 supports Alarm Feature, when LMDS200 detect the distance exceed the alarm settings, LMDS200 will Enter Alarm Mode and use the [ATDC Interval](#) (default is 1 minute) to uplink Distance value. Alarm mode will last for 60 uplinks (default 1 hour) and can be close by downlink command 0xA300.

Note: Alarm mode only valid for the Distance 1(the Closest Object)

User can set Alarm Distance to enable/disable Alarm Mode.

AT+ALARMC=AABBCCDD

- AABB: Hex value for Alarm low threshold, CCDD: Hex value for Alarm high threshold
- When 0xAABB=0, and 0xCCDD≠0, Alarm trigger when higher than max
- When 0xAABB≠0, and 0xCCDD =0xFFFF, Alarm trigger when lower than min
- When 0xAABB≠0 and 0xCCDD≠0, Alarm trigger when higher than max or lower than min

Example:

AT+ALARMC=006400C8 // Alarm when < 100 or higher than 200.

Downlink Payload:

0x(A2 00 01 00 00) // Same as AT+ALARMC=00010000

3.4 Enter/Exit Alarm Mode (0xA3)

Feature: Enter/Exit Alarm mode:

AT Command: AT+ALARM=0(Exit Alarm Mode or AT+ALARM=1 (Enter Alarm Mode)

Downlink Command:

0xA3 00 // Exit Alarm Mode.

0xA3 01 // Enter Alarm Mode. Enter Alarm mode will use ATDC interval in the next 59 uplinks.

3.5 Set Interrupt Mode(0x06)

Feature, Set Interrupt mode for GPIO_EXIT.

AT Command: AT+INTMOD

Command Example	Function	Response
AT+INTMOD=?	Show current interrupt mode	0 OK the mode is 0 = No interruption
AT+INTMOD=2	Set the interrupt triggering mode 0. (Disable Interrupt), 1. (Trigger by rising and falling edge) 2. (Trigger by falling edge) 3. (Trigger by rising edge)	OK

Downlink Command: 0x06

Format: Command Code (0x06) followed by 3 bytes.

This means that the interrupt mode of the end node is set to 0x000003=3 (rising edge trigger), and the type code is 06.

- Example 1: Downlink Payload: 06000000 // Turn off interrupt mode
- Example 2: Downlink Payload: 06000003 // Set the interrupt mode to rising edge trigger

4. Battery & Power Consumption

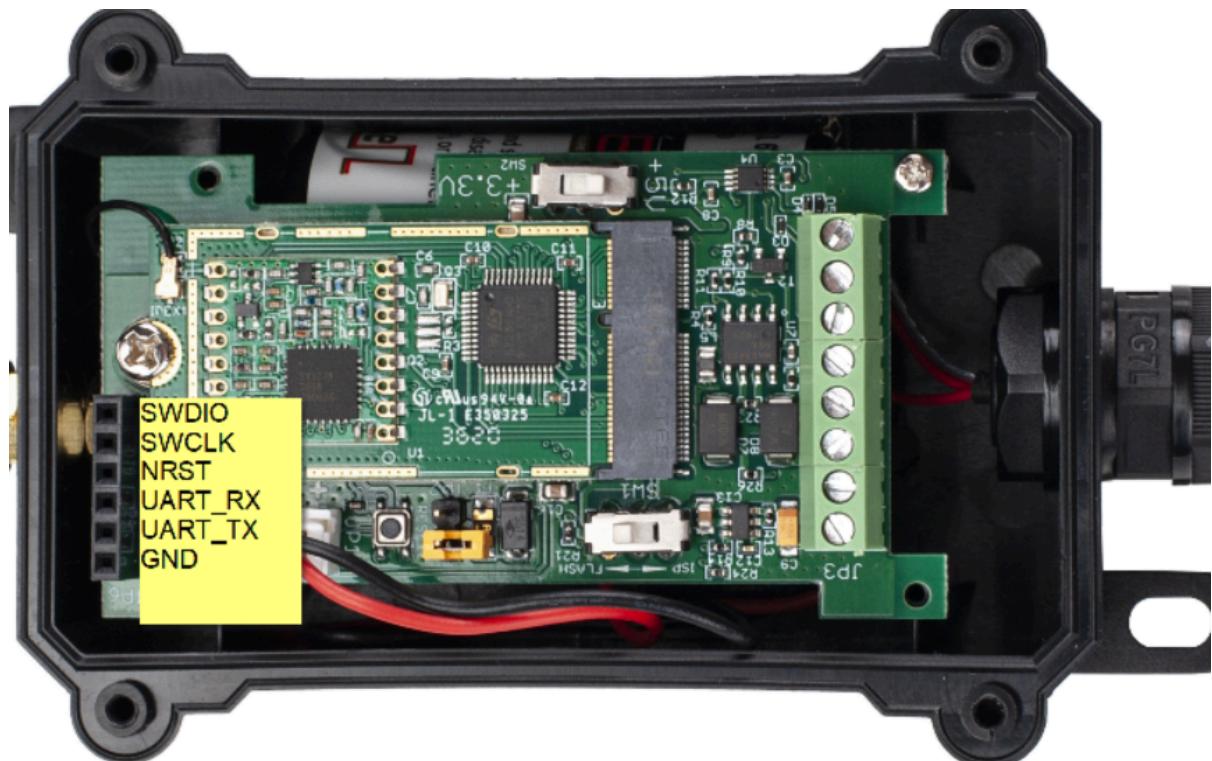
LMDS200 uses ER26500 + SPC1520 battery pack. See below link for detail information about the battery info and how to replace.

[Battery Info & Power Consumption Analyze](#).

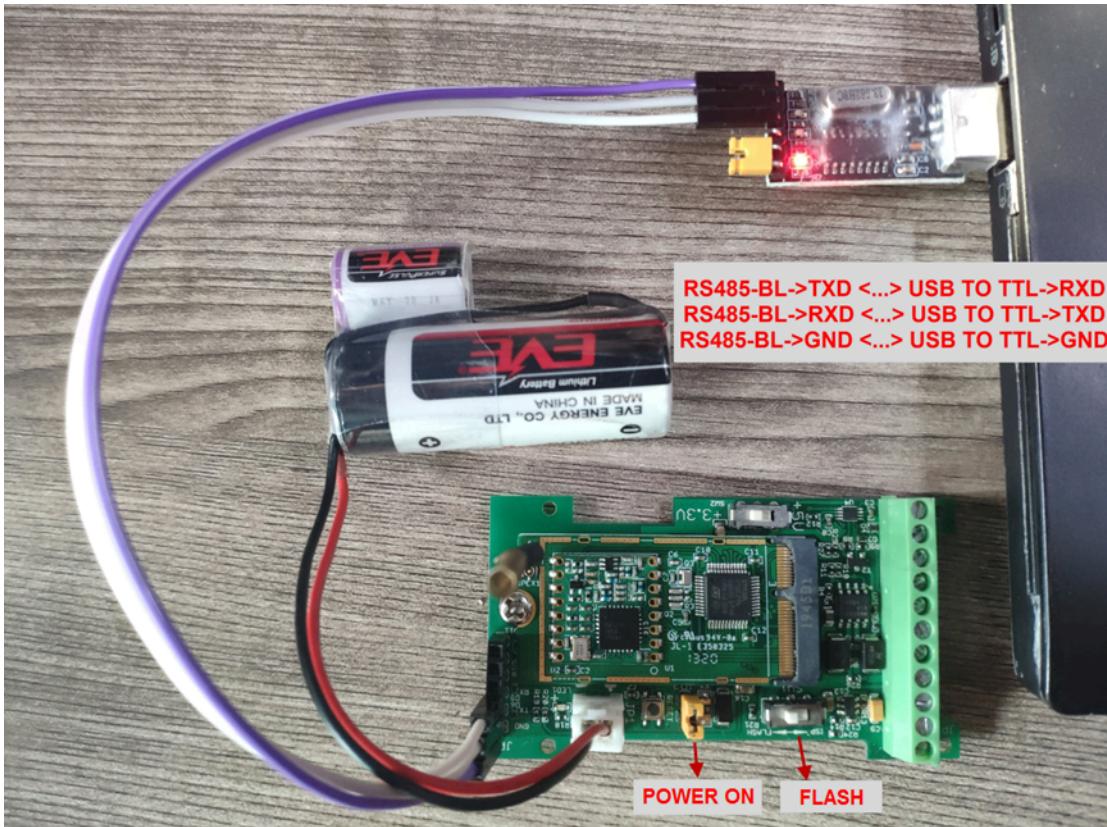
5. FAQ

5.1 How to use AT Command to configure LMDS200

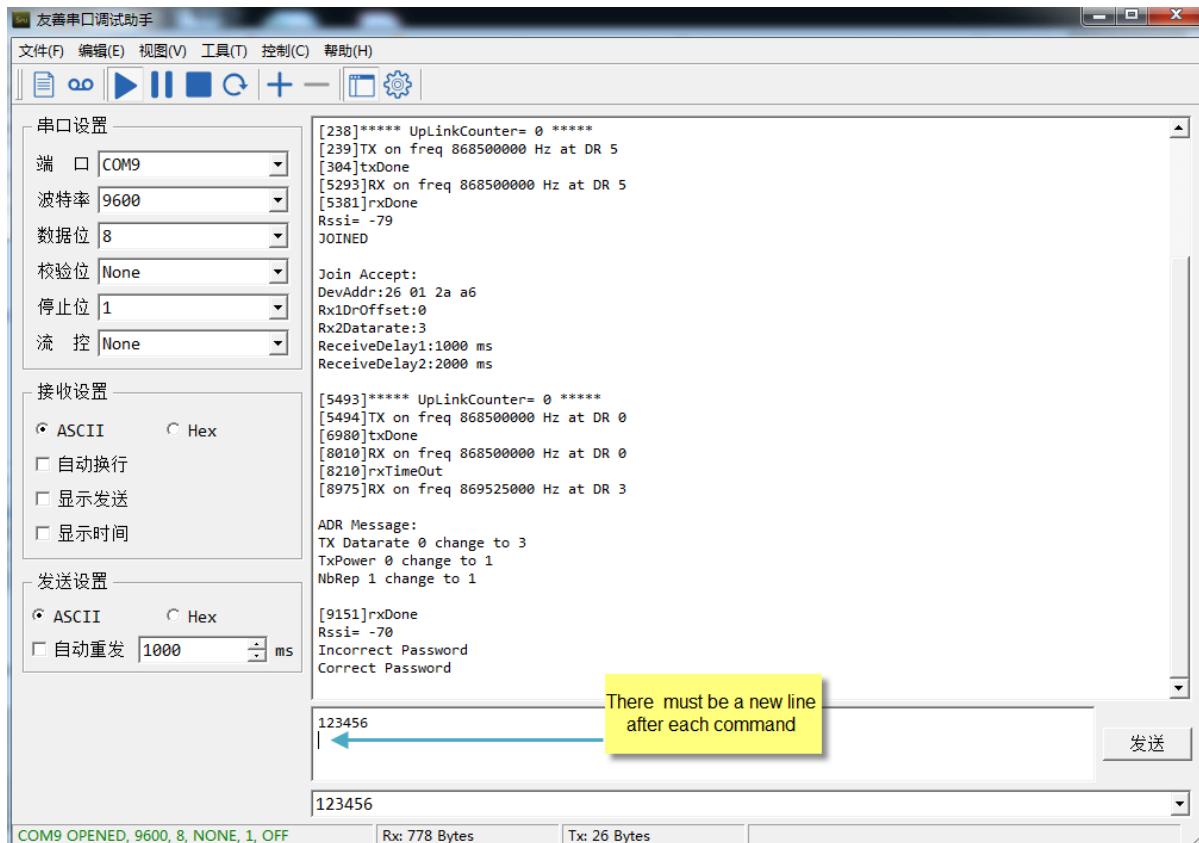
LMDS200 UART connection photo



LMDS200 supports AT Command set. User can use a USB to TTL adapter plus the 3.5mm Program Cable to connect to LMDS200 to use AT command, as below.



In the PC, you need to set the serial baud rate to **9600** to access the serial console for LMDS200. LMDS200 will output system info once power on as below:



5.2 How to upgrade the firmware?

A new firmware might be available for:

- Support new features
- For bug fix
- Change LoRaWAN bands.

Instruction for how to upgrade: <http://wiki.dragino.com/xwiki/bin/view/Main/Firmware%20Upgrade%20Instruction%20for%20STM32%20base%20products/#H2.HardwareUpgradeMethodSupportList>

Firmware location: https://www.dragino.com/downloads/index.php?dir=LoRa_End_Node/LMDS200/Firmware/

5.3 How to change the LoRa Frequency Bands/Region

You can follow the instructions for [how to upgrade image](#).
When downloading the images, choose the required image file for download.

6. Trouble Shooting

6.1 AT Command input doesn't work

In the case if user can see the console output but can't type input to the device. Please check if you already include the **ENTER** while sending out the command. Some serial tool doesn't send **ENTER** while press the send key, user need to add ENTER in their string.

7. Order Info

Part Number : **LMDS200-XX**

XX: The default frequency band

- **AS923** : LoRaWAN AS923 band
- **AU915** : LoRaWAN AU915 band
- **EU433** : LoRaWAN EU433 band
- **EU868** : LoRaWAN EU868 band
- **KR920** : LoRaWAN KR920 band
- **US915** : LoRaWAN US915 band
- **IN865** : LoRaWAN IN865 band
- **CN470** : LoRaWAN CN470 band

8. Packing Info

Package Includes:

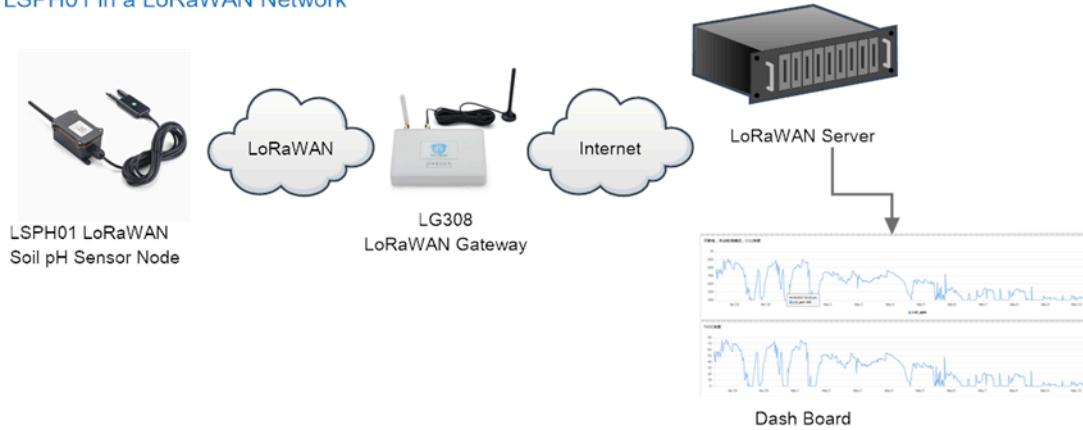
LMDS200 LoRaWAN Microwave Radar Distance Sensor x 1

9. Support

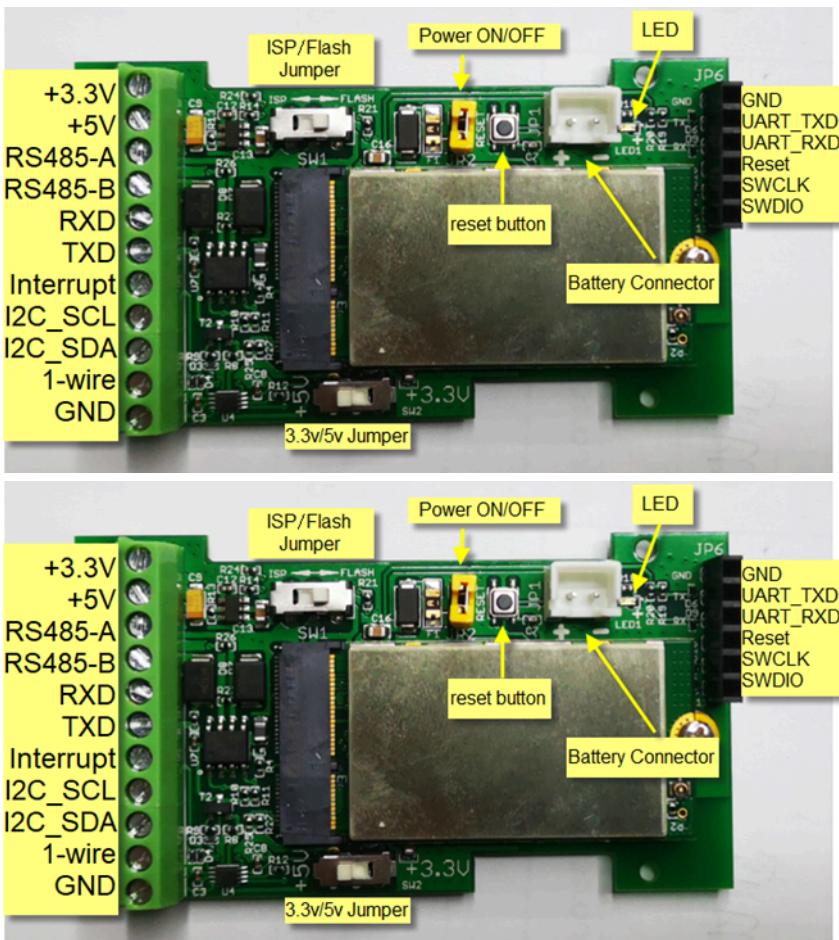
- Support is provided Monday to Friday, from 09:00 to 18:00 GMT+8. Due to different timezones we cannot offer live support. However, your questions will be answered as soon as possible in the before-mentioned schedule.
- Provide as much information as possible regarding your enquiry (product models, accurately describe your problem and steps to replicate it etc) and send a mail to support@dragino.com.



LSPH01 in a LoRaWAN Network



User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
-- LoRaWAN Microwave Radar Distance Sensor User Manual



Register end device

From The LoRaWAN Device Repository [Manually](#)

Preparation

Activation mode *

Over the air activation (OTAA)
 Activation by personalization (ABP)
 Multicast
 Do not configure activation

LoRaWAN version ⓘ *

MAC V1.0.3 

Network Server address

eu1.cloud.thethings.network

Application Server address

eu1.cloud.thethings.network

External Join Server ⓘ

Enabled

Join Server address

eu1.cloud.thethings.network

Start 

Register end device

From The LoRaWAN Device Repository [Manually](#)

1 Basic settings End device ID's, Name and Description

2 Network layer settings Frequency plan, regional parameters, end device class and session keys.

3 Join settings Root keys, NetID and kek labels.

End device ID ⓘ *

lsnPk01

AppEUI ⓘ * 

DevEUI ⓘ * 

End device name

LSNPK01

End device description

Description for my new end device

Optional end device description; can also be used to save notes about the end device

Network layer settings >

Register end device

From The LoRaWAN Device Repository [Manually](#)

Basic settings End device ID's, Name and Description

Network layer settings Frequency plan, regional parameters, end device class and session keys.

Join settings Root keys, NetID and kek labels.

Frequency plan *
Europe 863-870 MHz (SF12 for RX2)

LoRaWAN version
MAC V1.0.3

Regional Parameters version *
PHY V1.0.3 REVA

LoRaWAN class capabilities
 Supports class B
 Supports class C

Advanced settings

[Basic settings](#) [Join settings](#)

Register end device

From The LoRaWAN Device Repository [Manually](#)

Basic settings End device ID's, Name and Description

Network layer settings Frequency plan, regional parameters, end device class and session keys.

Join settings Root keys, NetID and kek labels.

Root keys

AppKey *
BD 72 1D AC F3 CC AB 67 72 8D 7A F5 4D DF 30 8B

Advanced settings

[Network layer settings](#) [Add end device](#)

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

The screenshot shows the configuration of payload formatters for an end device in The Things Stack Community Edition. The left sidebar includes sections for Overview, End devices, Live data, Payload formatters (selected), Integrations, Collaborators, API keys, and General settings. The main area displays two log entries for uplink messages, followed by a configuration panel for payload formatters. The 'Payload formatters' tab is selected, and the 'Uplink' tab is active. The configuration panel shows the following details:

- Formatter type:** Javascript (selected over Use application payload formatter, None, GRPC service, CayenneLPP, Repository).
- Formatter parameter:**

```
1 function decodeUplink(input) {
2     return {
3         data: [
4             bytes: input.bytes
5         ],
6         warnings: [],
7         errors: []
8     };
9 }
```
- Save changes** button at the bottom.

Below this, another screenshot shows the 'Choose webhook template' section, with the 'Webhooks' option highlighted in the sidebar. The available templates include Ubidots, Datacake, TagoIO, Akenza Core, ThingSpeak, Qubitro, and thethings.io.

Add custom webhook

Template information



Datacake

Send data to Datacake via TTI adapter

[About Datacake](#) | [Documentation](#)

Template settings

Webhook ID *

my-new-datacake-webhook

Token *

(empty field)

Datacake API Token

Create datacake webhook

Add Device



LoRaWAN



PARTICLE



API



D Zero



D Zero LTE



PINCODE

STEP 1
Product

STEP 2
Network Server

STEP 3
Devices

STEP 4
Plan

Datacake Product

You can add devices to an existing product on Datacake, create a new empty product or start with one of the templates. Products allow you to share the same configuration (fields, dashboard and more) between devices.

New Product from template

Create new product from a template

Existing Product

Add devices to an existing product

New Product

Create new empty product

New Product

If your device is not available as a template, you can start with an empty device. You will have to create the device definition (fields, dashboard) and provide the payload decoder in the device's configuration.

Product Name

LSPH01|

Next

Add Device



LoRaWAN



PARTICLE



API



D Zero



D Zero LTE



PINCODE

STEP 1
Product

STEP 2
Network Server

STEP 3
Devices

STEP 4
Plan

Network Server

Please choose the LoRaWAN Network Server that your devices are connected to.

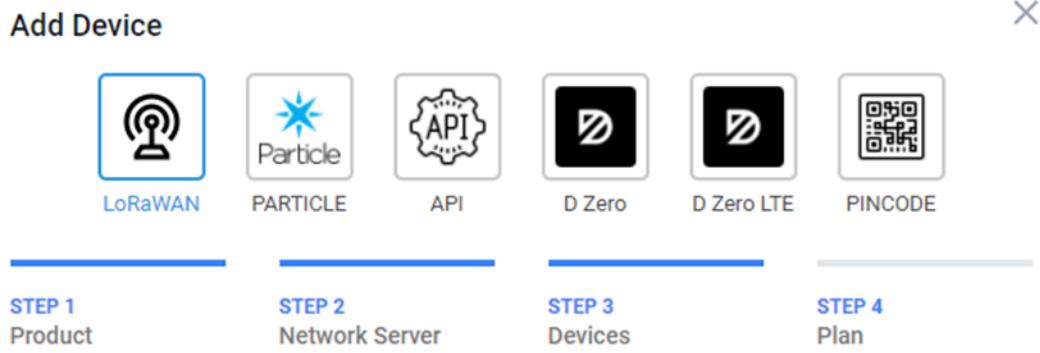
	The Things Stack V3 TTN V3 / Things Industries	Uplinks	Downlinks	
<input type="radio"/>		The Things Network V2 The old Things Network	Uplinks	Downlinks
<input type="radio"/>		helium Helium	Uplinks	Downlinks
<input type="radio"/>		LORIOT	Uplinks	Downlinks
<input type="radio"/>		Kerlink Wanesy	Uplinks	

Showing 1 to 5 of 8 results

[Previous](#) [Next](#)

[Back](#)

[Next](#)



Add Devices

Enter one or more LoRaWAN Device EUIs and the names they will have on Datacake.

DEVEUI	NAME
00 00 00 00 00 00 00 6E 8 bytes	LSPH01

+ Add another device

Consistent with DEUI on TTN

Back Next

LSPH01

Location	Serial Number	Last update	Product Slug
-	0000000000000066	-	@ lsp01

Dashboard History Downlinks Configuration Debug Rules Permissions

General Configuration

Name: LSPH01

Location:

Tags: You can use tags to organize your devices and create filters on the dashboard

Arctan

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

Payload Decoder

Product-wide setting

When your devices sends data, the payload will be passed to the payload decoder, alongside the event's name. The payload decoder then transforms it to measurements.

```

1 // (array) or bytes to an object or variant.
2 var value=bytes[0]<0 | bytes[1]) & 0x3FFF;
3 var batV=value/1000; //Battery,units:V
4
5 value=bytes[1]<0 | bytes[2];
6 if((bytes[1] & 0x00) == 0)
7 {
8     y=(value<0 | bytes[2]);
9     var temp_051820=(value/10).toFixed(2); //051820,temperature
10    var temp_051820=(value/10).toFixed(2); //051820,temperature
11    value=bytes[2]<0 | bytes[3];
12    var PH=(value/10).toFixed(2);
13    value=bytes[3]<0 | bytes[4];
14    value=bytes[4]<0 | bytes[5];
15    value=bytes[5]<0 | bytes[6];
16    if((value & 0x0000)>>15 === 0)
17    {
18        temp=(value/10).toFixed(2); //temp_Soil,temperature
19        else if((value & 0x0000)>>15 === 1)
20        {
21            temp=(value>>0xFFFF)/10).toFixed(2);
22        }
23    }
24    var l_flag = bytes[6];
25    var met_type = bytes[10];
26    return {
27        "field": "BAT",
28        "value": batV
29    },
30    {
31        "field": "PH",
32        "value": PH
33    },
34    {
35        "field": "TEMP",
36        "value": temp
37    };
38 }

```

Payload

Payload

Port

1

 Try Decoder

Output

Rawdata from Output

Derivatized measurements

 Add Field

Fields

Name	Identifier	Type	Current Value	More
BAT	 BAT	Float	0	
PH	 PH	Float	0	
temp	 TEMP	Float	0	

LSPH01

Location

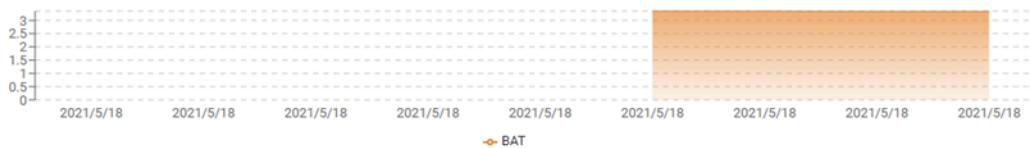
Serial Number
0000000000000066

Last update
Tue May 18 2021 16:40:14 GMT+0800

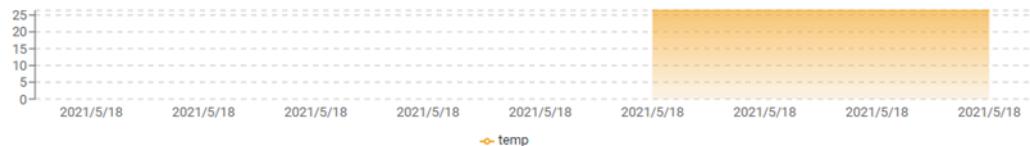
Product Slug
 lsp01

[Dashboard](#) [History](#) [Downlinks](#) [Configuration](#) [Debug](#) [Rules](#) [Permissions](#)

BAT
a few seconds ago



TEMP
a few seconds ago

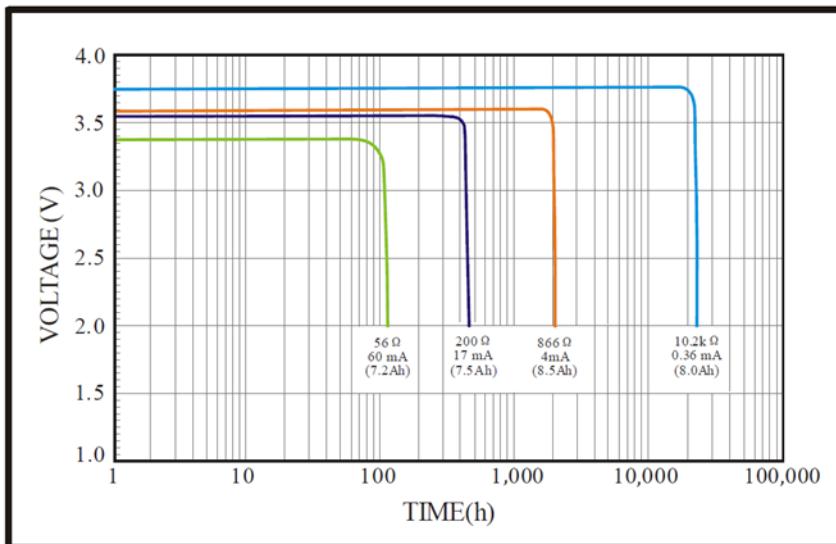


PH
a few seconds ago

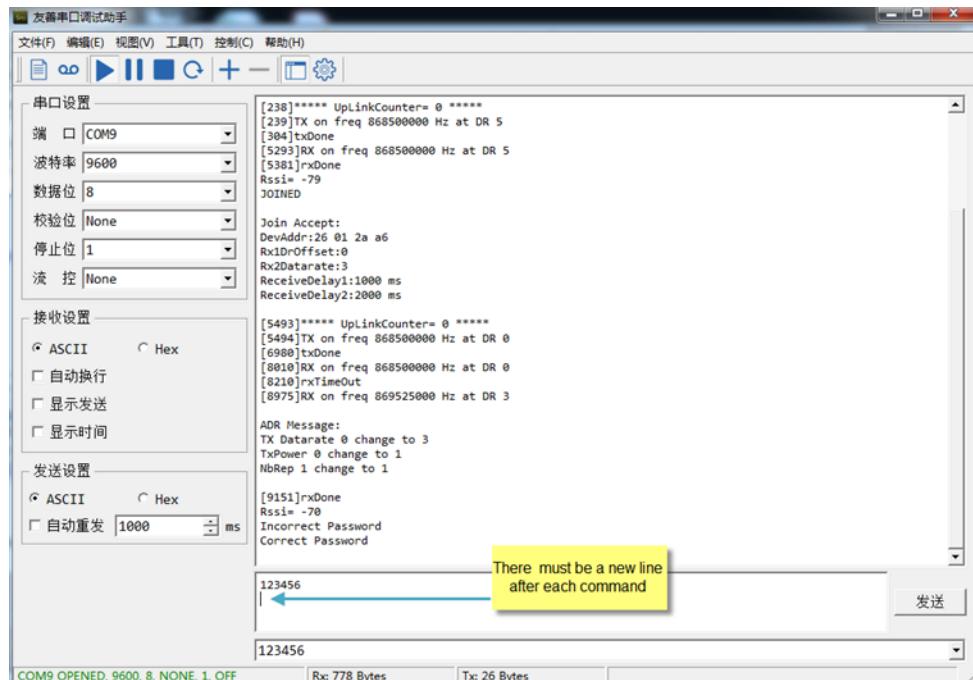
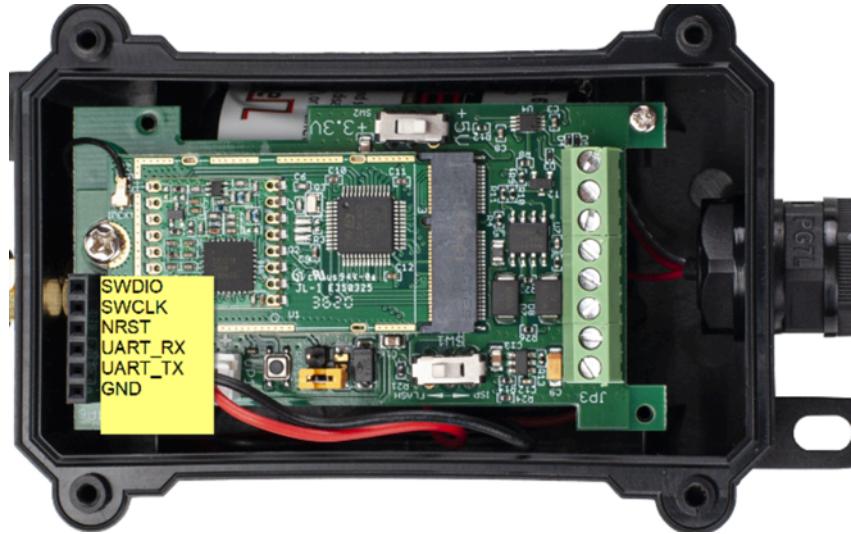




1. Typical discharge profile at +20 °C(Typical value)

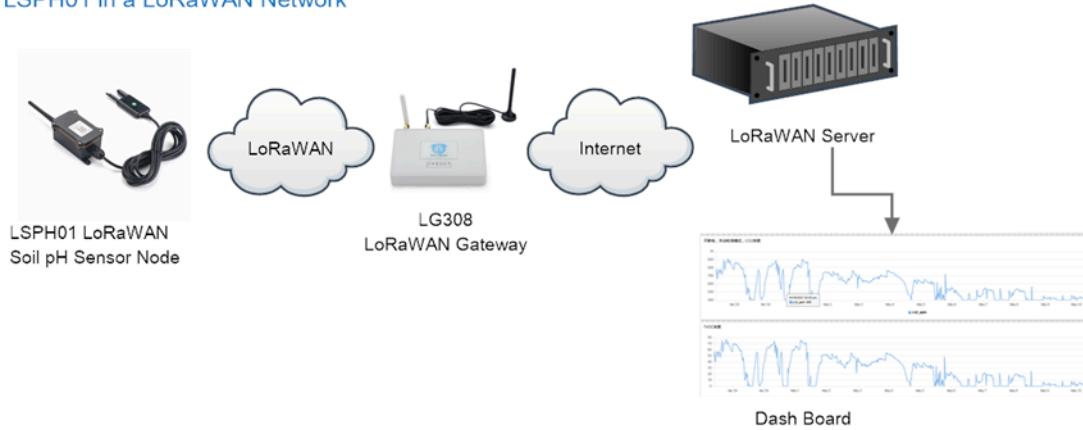


Battery Life Calculator										
How to use:		Note: do not modify the formula in the table.								
1. After selecting the product number and model, then select the TDC unit, and finally enter the TDC, you can get the predicted battery life										
Explanation of abbreviations: WD ->Watchdog TX ->Transmit RX ->Receive										
Product	battery capacity(mah)									
DRAGINO LD901_LoRaWAN_Door_Sensor	240									
UNIT	TDC (Uplink Interval)	Work Mode								
min	20	Mod-1								
	Sleep power (mW*ms)	Sampling power (mW*ms)	TX power (mW*ms)	RX1 power (mW*ms)	RX2 power (mW*ms)	Watchdog power (mW*ms)	Average power (mA)	Detect power (mA*s)	Life expectancy (yrs)	
EU868	DR8_SF7_125K_14dB	8400	427.16444	7367.8546	880.58080	4097.083	752.1706667	0.018268045	0	1.5
	DR8_SF8_125K_14dB	8400	427.16444	13210.2547	950.09441	4097.083	752.1706667	0.021102523	0	1.2
	DR8_SF9_125K_14dB	8400	427.16444	23652.008	1008.0390	4097.083	752.1706667	0.019887736	0	0.8
	DR8_SF10_125K_14dB	8400	427.16444	42246.125	1461.4876	4097.083	752.1706667	0.047792297	0	0.6
	DR8_SF11_125K_14dB	8400	427.16444	94011.4	2210.4828	4097.083	752.1706667	0.091509995	0	0.3
	DR8_SF12_125K_14dB	8400	427.16444	168081	4097.083	4097.083	752.1706667	0.154625138	0	0.2
US915	DR8_SF7_125K_20dB	8400	427.16444	8445.476	682.61080	1587.115	752.1706667	0.016008176	0	1.6
	DR8_SF8_125K_20dB	8400	427.16444	15170.395	931.6491	1587.115	752.1706667	0.022797298	0	1.2
	DR8_SF9_125K_20dB	8400	427.16444	27254.383	1411.388	1587.115	752.1706667	0.032794012	0	0.8
	DR8_SF10_125K_20dB	8400	427.16444	48355.12	995.2343	1587.115	752.1706667	0.050735363	0	0.5

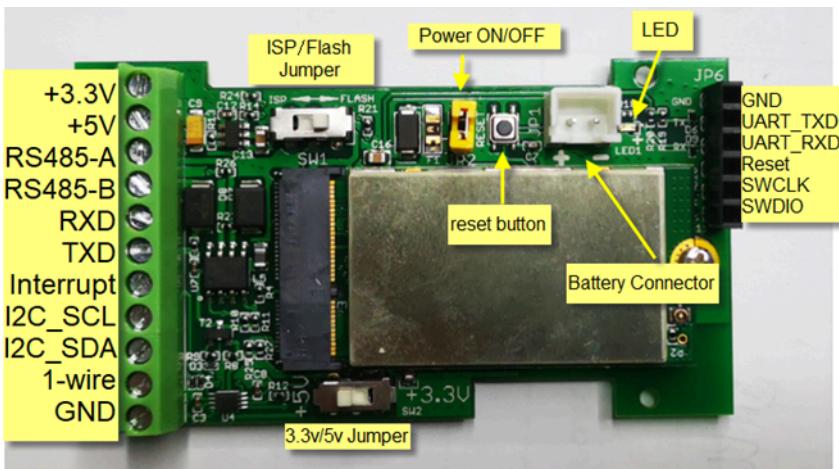




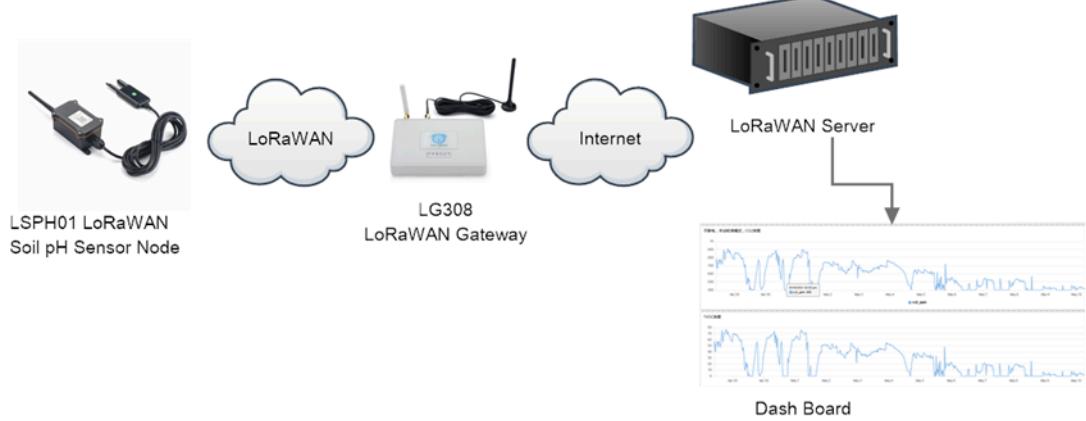
LSPH01 in a LoRaWAN Network



User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
-- LoRaWAN Microwave Radar Distance Sensor User Manual



LSPH01 in a LoRaWAN Network



Register end device

From The LoRaWAN Device Repository [Manually](#)

Preparation

Activation mode *

Over the air activation (OTAA)
 Activation by personalization (ABP)
 Multicast
 Do not configure activation

LoRaWAN version ⓘ *

MAC V1.0.3 

Network Server address

eu1.cloud.thethings.network

Application Server address

eu1.cloud.thethings.network

External Join Server ⓘ

Enabled

Join Server address

eu1.cloud.thethings.network

Start 

Register end device

From The LoRaWAN Device Repository [Manually](#)

1 Basic settings End device ID's, Name and Description

2 Network layer settings Frequency plan, regional parameters, end device class and session keys.

3 Join settings Root keys, NetID and kek labels.

End device ID ⓘ *

lsnPk01

AppEUI ⓘ * 

DevEUI ⓘ * 

End device name

LSNPK01

End device description

Description for my new end device

Optional end device description; can also be used to save notes about the end device

Network layer settings >

Register end device

From The LoRaWAN Device Repository [Manually](#)

Basic settings End device ID's, Name and Description

Network layer settings Frequency plan, regional parameters, end device class and session keys.

Join settings Root keys, NetID and kek labels.

Frequency plan *
Europe 863-870 MHz (SF12 for RX2)

LoRaWAN version
MAC V1.0.3

Regional Parameters version *
PHY V1.0.3 REVA

LoRaWAN class capabilities
 Supports class B
 Supports class C

Advanced settings ▾

[Basic settings](#) [Join settings](#)

Register end device

From The LoRaWAN Device Repository [Manually](#)

Basic settings End device ID's, Name and Description

Network layer settings Frequency plan, regional parameters, end device class and session keys.

Join settings Root keys, NetID and kek labels.

Root keys

AppKey *
BD 72 1D AC F3 CC AB 67 72 8D 7A F5 4D DF 30 8B

Advanced settings ▾

[Network layer settings](#) [Add end device](#)

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

The screenshot shows the configuration of payload formatters for an end device in The Things Stack Community Edition. The left sidebar includes sections for Overview, End devices, Live data, Payload formatters (selected), Integrations, Collaborators, API keys, and General settings. The main area displays a log of messages and a configuration panel for payload formatters. The configuration panel has tabs for Overview, Live data, Messaging, Location, Payload formatters (selected), Claiming, and General settings. It shows an uplink message with a Javascript formatter selected. The formatter code is:

```

1 function decodeUplink(input) {
2   return {
3     data: [
4       bytes: input.bytes
5     ],
6     warnings: [],
7     errors: []
8   }
9 }

```

A red box highlights the 'Javascript' radio button. A blue box highlights the 'Payload formatters' tab. A red arrow points to the 'Webhooks' option in the sidebar under the 'Payload formatters' section.

Add custom webhook

Template information



Datacake

Send data to Datacake via TTI adapter

[About Datacake](#) | [Documentation](#)

Template settings

Webhook ID *

my-new-datacake-webhook

Token *

(empty field)

Datacake API Token

Create datacake webhook

Add Device



LoRaWAN



PARTICLE



API



D Zero



D Zero LTE



PINCODE

STEP 1
Product

STEP 2
Network Server

STEP 3
Devices

STEP 4
Plan

Datacake Product

You can add devices to an existing product on Datacake, create a new empty product or start with one of the templates. Products allow you to share the same configuration (fields, dashboard and more) between devices.

New Product from template

Create new product from a template

Existing Product

Add devices to an existing product

New Product

Create new empty product

New Product

If your device is not available as a template, you can start with an empty device. You will have to create the device definition (fields, dashboard) and provide the payload decoder in the device's configuration.

Product Name

LSPH01|

Next

Add Device



LoRaWAN



PARTICLE



API



D Zero



D Zero LTE



PINCODE

STEP 1
Product

STEP 2
Network Server

STEP 3
Devices

STEP 4
Plan

Network Server

Please choose the LoRaWAN Network Server that your devices are connected to.

	The Things Stack V3 TTN V3 / Things Industries	Uplinks	Downlinks	
<input type="radio"/>		The Things Network V2 The old Things Network	Uplinks	Downlinks
<input type="radio"/>		helium Helium	Uplinks	Downlinks
<input type="radio"/>		LORIOT	Uplinks	Downlinks
<input type="radio"/>		Kerlink Wanesy	Uplinks	

Showing 1 to 5 of 8 results

[Previous](#) [Next](#)

[Back](#)

[Next](#)

Add Device

LoRaWAN PARTICLE API D Zero D Zero LTE PINCODE

STEP 1 Product STEP 2 Network Server STEP 3 Devices STEP 4 Plan

Add Devices

Enter one or more LoRaWAN Device EUIs and the names they will have on Datacake.

DEVEUI	NAME
00 00 00 00 00 00 00 6E 8 bytes	LSPH01

+ Add another device

Consistent with DEUI on TTN

Back

Next

LSPH01

Location Serial Number Last update Product Slug
0000000000000066 - - ⚒ lsp01

Dashboard History Downlinks Configuration Debug Rules Permissions

General Configuration

Name: LSPH01

Location:

Tags: You can use tags to organize your devices and create filters on the dashboard

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

Payload Decoder

Product-wide setting

When your devices sends data, the payload will be passed to the payload decoder, alongside the event's name. The payload decoder then transforms it to measurements.

```

1 // (array) or bytes to an object or variant.
2 var value=bytes[0]<0 | bytes[1]) & 0x3FFF;
3 var batV=value/1000; //Battery,units:V
4
5 value=bytes[1]<0 | bytes[2];
6 if((bytes[1] & 0x00) == 0)
7   y=(value < 0 ? -value : value);
8 else
9   y=(value < 0 ? -value : value)/0518020;
10 var temp_0518020=(value/10).toFixed(2)/0518020,temperature
11
12 value=bytes[1]<0 | bytes[3];
13 var PH=(value/10).toFixed(2);
14
15 value=bytes[1]<0 | bytes[7];
16 if((value & 0x0000)>>15 === 0)
17   temp=(value/10).toFixed(2); //temp_Soil,temperature
18 else if((value & 0x0000)>>15 === 1)
19   temp=(value>>0xFFFF)/10).toFixed(2);
20
21 var l_flag = bytes[1];
22 var met_type = bytes[10];
23
24+
25+
26+
27+
28+
29+
30+
31+
32+
33+
34+
35+
36+
37+
38+

```

Payload

Payload

Port

1

Try Decoder

Output

Measurement Output

Normalized measurements

Add Field

Fields

Name	Identifier	Type	Current Value	More
BAT	BAT	Float	0	More
PH	PH	Float	0	More
temp	TEMP	Float	0	More

LSPH01

Location

Serial Number
0000000000000066

Last update
Tue May 18 2021 16:40:14 GMT+0800

Product Slug
lsp01

Dashboard History Downlinks Configuration Debug Rules Permissions

BAT
a few seconds ago



TEMP
a few seconds ago

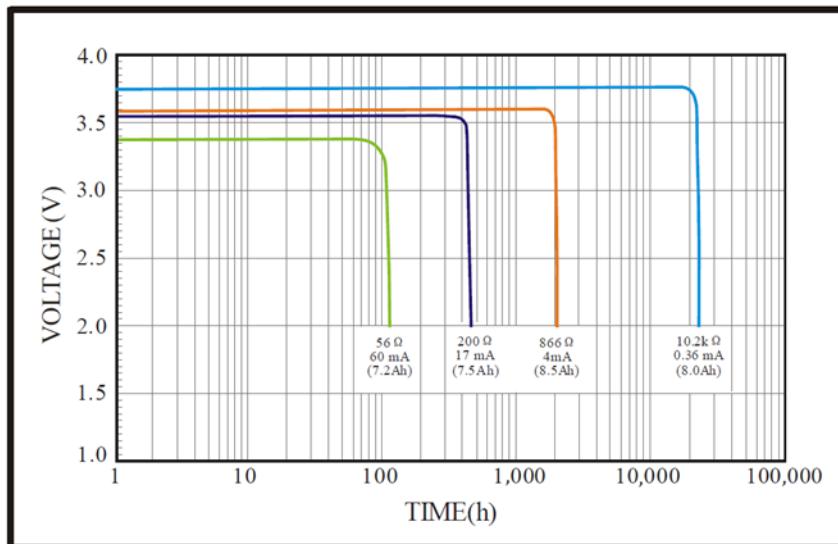


PH
a few seconds ago

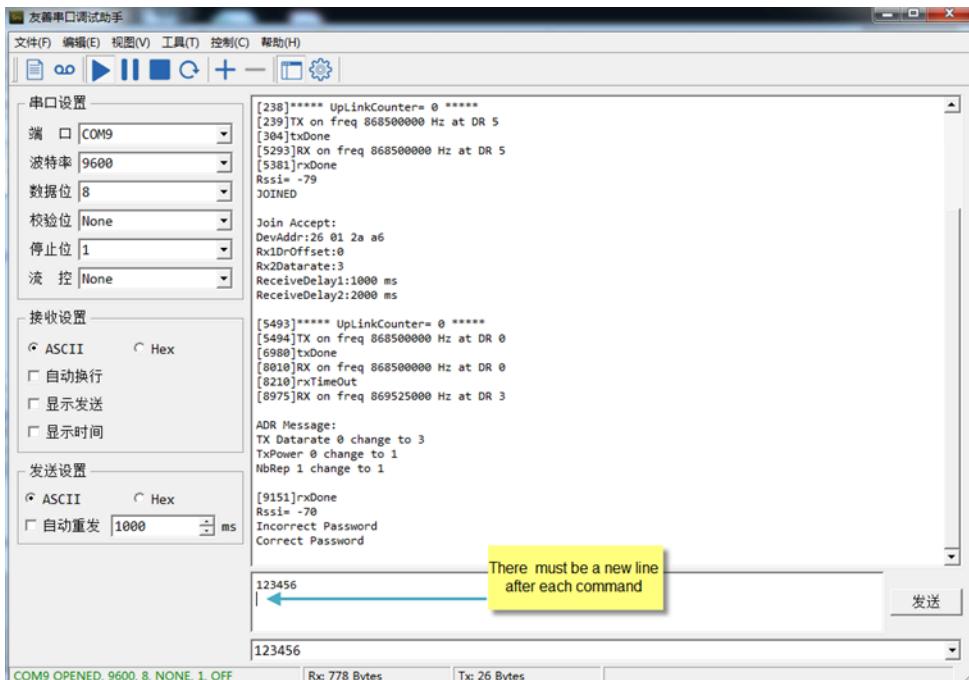


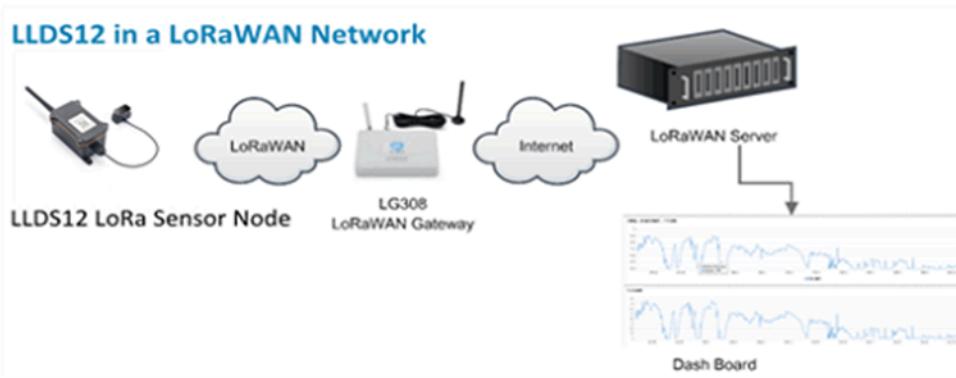


1. Typical discharge profile at +20 °C(Typical value)



Battery Life Calculator											
Product		battery capacity(mah)	Time (ms)								
UNIT		TDC (Uplink Interval)	Work Mode	W	D	sleep	R	X	R	X	
min	max	mod-1									
			Sleep power (mA*s/m)	Sampling power (mA*s/m)	TX power (mA*s/m)	RX power (mA*s/m)	RX2 power (mA*s/m)	Watchdog power (mA*s/average power (mA))	Detect power (mA*s)	Life expectancy (y)	
EU868	DR1_SF7_125K_14dB	8400	427.16444	7367.85446	880.58080	4097.083	752.1706667	0.018268045	0	1.5	
	DR4_SF8_125K_14dB	8400	427.16444	13210.25446	950.09441	4097.083	752.1706667	0.021102523	0	1.2	
	DR3_SF9_125K_14dB	8400	427.16444	23652.00446	1008.03930	4097.083	752.1706667	0.019887736	0	0.8	
	DR2_SF10_125K_14dB	8400	427.16444	42246.125	1461.4876	4097.083	752.1706667	0.047792297	0	0.6	
	DR1_SF11_125K_14dB	8400	427.16444	94011.4	2210.4828	4097.083	752.1706667	0.091509995	0	0.3	
	DR0_SF12_125K_14dB	8400	427.16444	164801	4097.083	4097.083	752.1706667	0.154625138	0	0.2	
US915	DR1_SF7_325K_20dB	8400	427.16444	8445.476	682.63080	1587.115	752.1706667	0.016008176	0	1.6	
	DR2_SF8_325K_20dB	8400	427.16444	15570.395	931.6491	1587.115	752.1706667	0.022797250	0	1.2	
	DR1_SF9_325K_20dB	8400	427.16444	27254.383	941.388	1587.115	752.1706667	0.032794012	0	0.8	
	DR0_SF10_325K_20dB	8400	427.16444	8835.12	995.2343	1587.115	752.1706667	0.050735363	0	0.5	





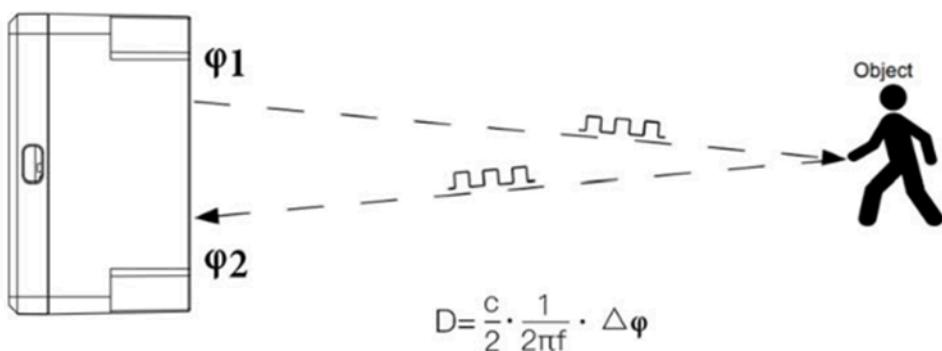
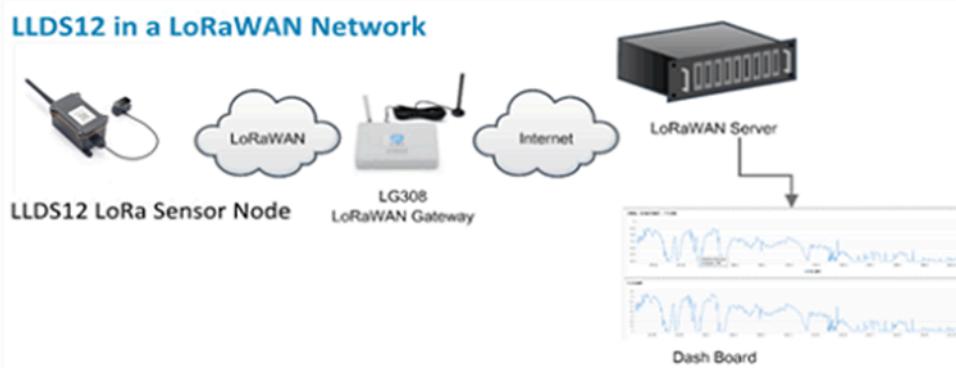
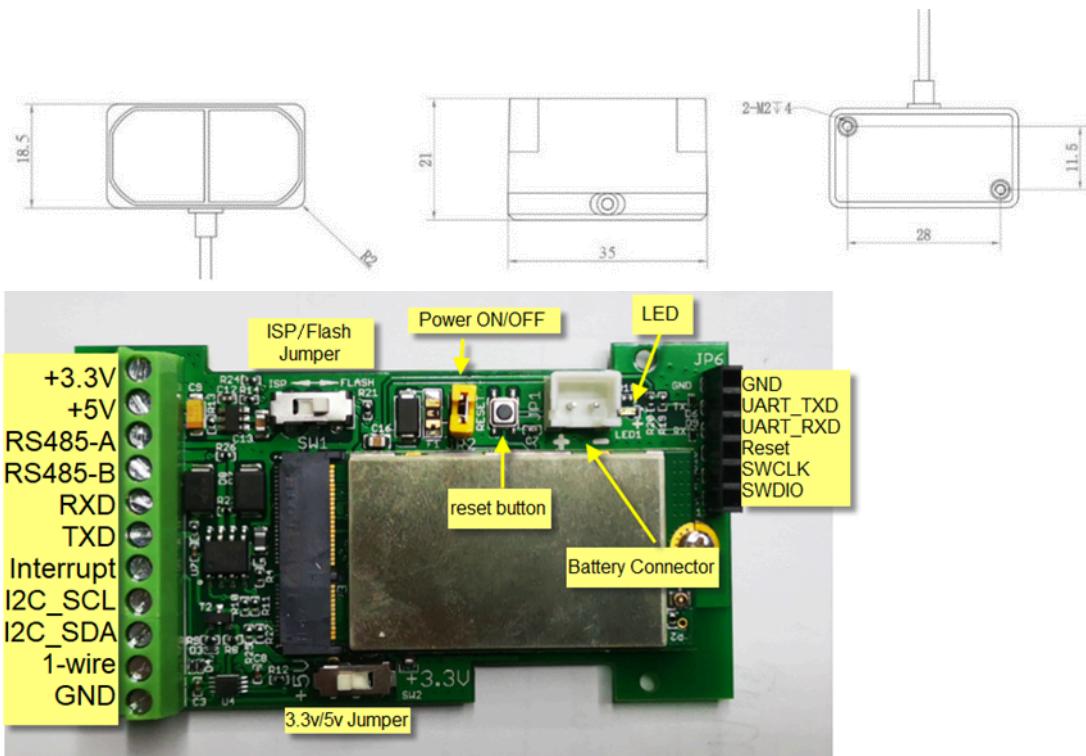


Figure 1 Schematics of TOF Principle

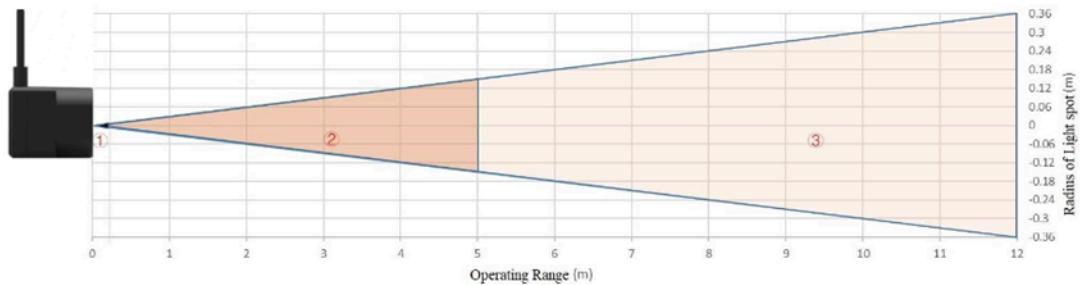


Figure 2 Schematic diagram of size of light spot

$$d = D \cdot \tan\beta$$

Table 2 the Minimum side length of effective detection corresponding to Detecting Range

Detecting range	1m	2m	3m	4m	5m	6m	7m	8m	9m	10m	11m	12m
Minimum side length	6cm	12cm	18cm	24cm	30cm	36cm	42cm	48cm	54cm	60cm	66cm	72cm

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
-- LoRaWAN Microwave Radar Distance Sensor User Manual

The screenshot shows the Datacake interface for managing IoT devices. At the top, a modal window titled 'Add Device' is open, showing options for 'LoRaWAN', 'PARTICLE', 'API', 'D Zero', 'D Zero LTE', and 'PINCODE'. The 'LoRaWAN' option is selected. Below the modal, the main interface displays a list of devices: 'aq4', 'LAQ4-template', 'dd75-test', 'ht65-one', 'ht65-test', 'ht65-two', 'sn50', 'LSN50-template', 'LSPH01-template', 'JT22222', and 'RS485-LN'. A message indicates 'to 11 of 11 results'. On the right, there is a '50 per page' dropdown.

Add Device Wizard:

- Step 1 Product:** LoRaWAN
- Step 2 Network Server:** Network Server
- Step 3 Devices:** Devices
- Step 4 Plan:** Plan

Add Devices: Enter one or more LoRaWAN Device EUIs and the names they will have on Datacake.

DEVEUI	NAME
21 21 24 25 35 34 14 8 bytes	LLDS12

+ Add another device

Back Next

Device Configuration Page:

Location	Serial Number	Last update	Product Slug
-	A8404166A18219CF	Mon May 31 2021 19:27:51 GMT+0800	dragino-dds75-6

Dashboard new Legacy Dashboard History Downlinks Configuration Debug Rules Permissions

General Configuration:

- Name: Idd75-test
- Location: (empty input field)
- Tags: You can use tags to organize your devices and create filters on the dashboard
Add tag
- Metadata: Metadata is displayed on the device overview and can be used in dashboards
Add metadata

A red arrow points from the 'Configuration' tab in the navigation bar down to the 'General Configuration' section.

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200 -- LoRaWAN Microwave Radar Distance Sensor User Manual

Payload Decoder

When your devices sends data, the payload will be passed to the payload decoder, alongside the event's name. The payload decoder then transforms it to measurements.

```
1 * Function Decoder(bytes, port) {
2   // Decode an uplink message from a buffer
3   // (array) of bytes to an object of fields.
4   var value=bytes[0]<>0 | bytes[1] & 0x0FFF;
5   var battValue=bytes[0]<>0 | bytes[1] & 0x0FFF;
6   value=bytes[2]<>0 | bytes[3];
7   var hum=(value<>0)<>0;
8   var temp=(value<>0)<>0;
9   var temp_D51880=(value<>10).toFixed(2); //D51880,temperature
10  var temp_D51880=(value<>10).toFixed(2); //D51880,temperature
11  value=bytes[4]<>0 | bytes[5];
12  var hum=(value<>10).toFixed(2);
13  var temp=(value<>10).toFixed(2);
14  value=bytes[6]<>0 | bytes[7];
15  var temp=(value<>10).toFixed(2);
16  var temp=(value<>10).toFixed(2);
17
18  var l_flag = bytes[8];
19
20  return [
21    {
22      #field: "BATTERY",
23      value: battV
24    },
25    {
26      #field: "LEAF_MOISTURE",
27      value: hum
28    },
29    {
30      #field: "LEAF_TEMPERATURE",
31      value: temp
32    }
33  ];
34 }
```

Payload

Output

Port: 1

Try Decoder

Recognized measurements

Location: A8404166A18219CF

Serial Number: A8404166A18219CF

Last update: Mon May 31 2021 19:27:51 GMT+0800

Product Slug: dragino-ldds75-6

Dashboard **new** Legacy Dashboard History Downlinks Configuration Debug Rules Permissions

We have introduced a new and more powerful way to create dashboards. Try out the new dashboard builder by clicking the first Dashboard tab above.

Distance

2,799 mm

Last Update: 19 minutes ago

Battery Voltage

3 Volt

Last Update: 19 minutes ago

Sensor Status

Sensor OK

Last Update: 19 minutes ago

Trend



Page 57 / 105 - last modified by Mengting Qiu on 2024/03/07 08:41

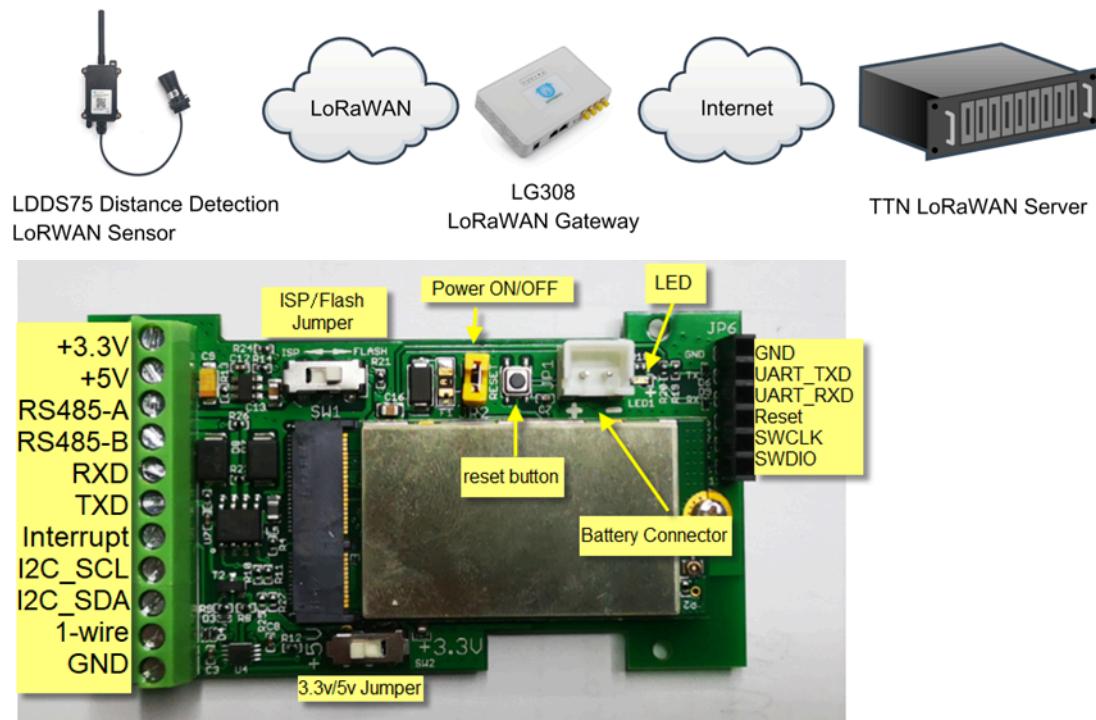
User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200 -- LoRaWAN Microwave Radar Distance Sensor User Manual

The screenshot shows a dashboard interface for managing LoRaWAN end nodes. At the top, it displays basic node information: Location (Location), Serial Number (A8404166A18219CF), Last update (Mon May 31 2021 19:27:51 GMT+0800), and Product Slug (dragino-ldds75-6). Below this is a navigation bar with tabs: Dashboard (new), Legacy Dashboard, History, Downlinks, Configuration, Debug, Rules, and Permissions. A message通知 states: "We have introduced a new and more powerful way to create dashboards. Try out the new dashboard builder by clicking the first Dashboard tab above." The main content area contains several cards:

- Distance:** Shows a value of 2,799 mm last updated 19 minutes ago.
- Battery Voltage:** Shows a value of 3 Volt last updated 19 minutes ago.
- Sensor Status:** Shows "Sensor OK" last updated 19 minutes ago.
- Trend:** A chart showing a single data point at approximately 2,799 mm.
- LLDS12 Node 1 (ID: 121212):** Shows the last seen time (2 minutes ago), a log entry for an uplink data message at 18:28:19, and a log entry for an accept join-request at 18:18:01. It also shows a "Live data" section with a table of recent messages and a "Data preview" section showing raw hex and ASCII data.
- LLDS12 Node 2 (ID: 121212):** Similar to Node 1, showing the last seen time (2 minutes ago), log entries for uplink data messages and accept join-requests, and a "Live data" section with a table of recent messages and a "Data preview" section showing raw hex and ASCII data.



LDDS75 in a LoRaWAN Network



LDDS75 in a LoRaWAN Network



The screenshot shows the TTN LoRaWAN Server interface. At the top, there are two message logs for an end device. The first log shows an uplink message with a payload containing distance data. The second log shows a duplicate uplink message with the same payload. Below the logs, there is a table of messages with columns for time, type, DevAddr, MAC payload, FPort, SNR, RSSI, Bandwidth, and Raw payload.

On the left, a sidebar navigation includes Overview, End devices (selected), Live data, Payload formatters (highlighted with a red box), Integrations, Collaborators, API keys, and General settings. The End devices section lists the end device used in the logs.

In the main content area, under the End devices tab, the Uplink tab is selected. A note says "These payload formatters are executed on uplink messages from this end device and take precedence over application level payload formatters." Below this, the Formatter type section shows "Javascript" selected (highlighted with a red box). The Formatter parameter section contains a code editor with the following JavaScript function:

```

1 function decodeUplink(input) {
2     return {
3         data: {
4             bytes: input.bytes
5         },
6         warnings: [],
7         errors: []
8     };
9 }

```

A "Save changes" button is located at the bottom right of the payload formatter configuration panel.

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
-- LoRaWAN Microwave Radar Distance Sensor User Manual

Complex configuration and setup.

Search All Manufacturers

Dragino LSE01
Dragino 

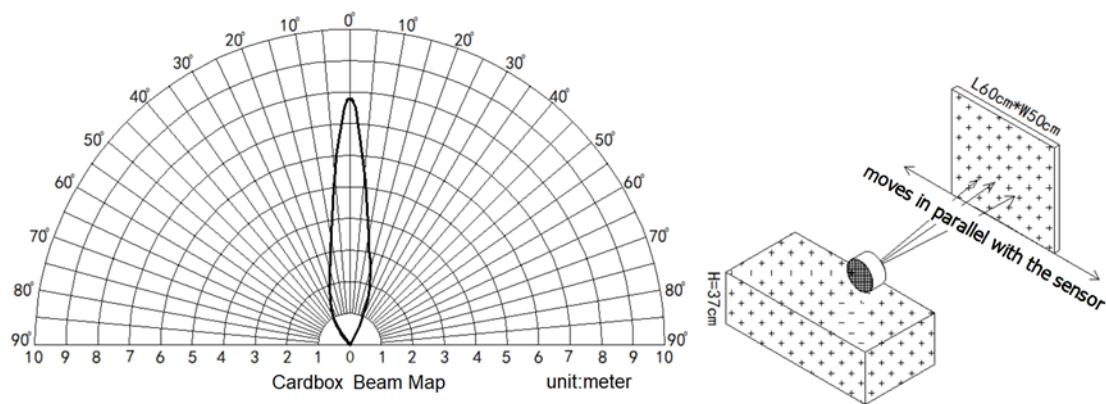
Dragino LT-22222-L
Dragino 

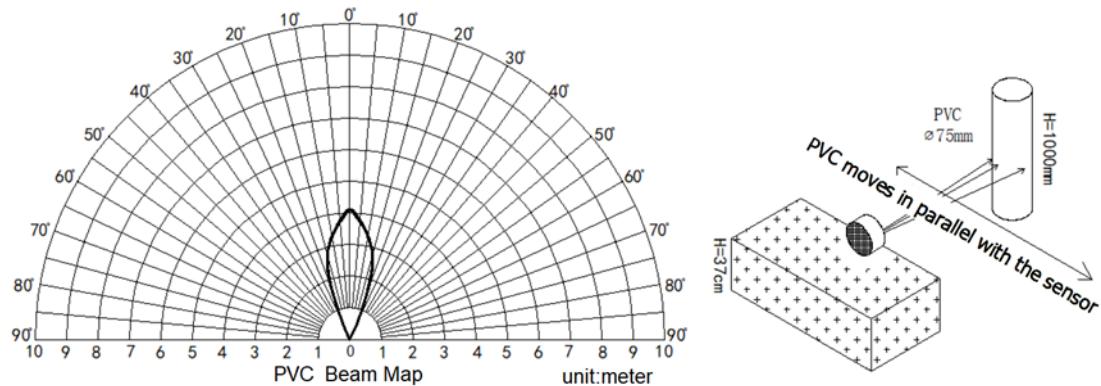
Dragino LWL01
Dragino 

ESP32-Paxcounter
cyberman54 

Elysys ELT-2
Elysys.se 

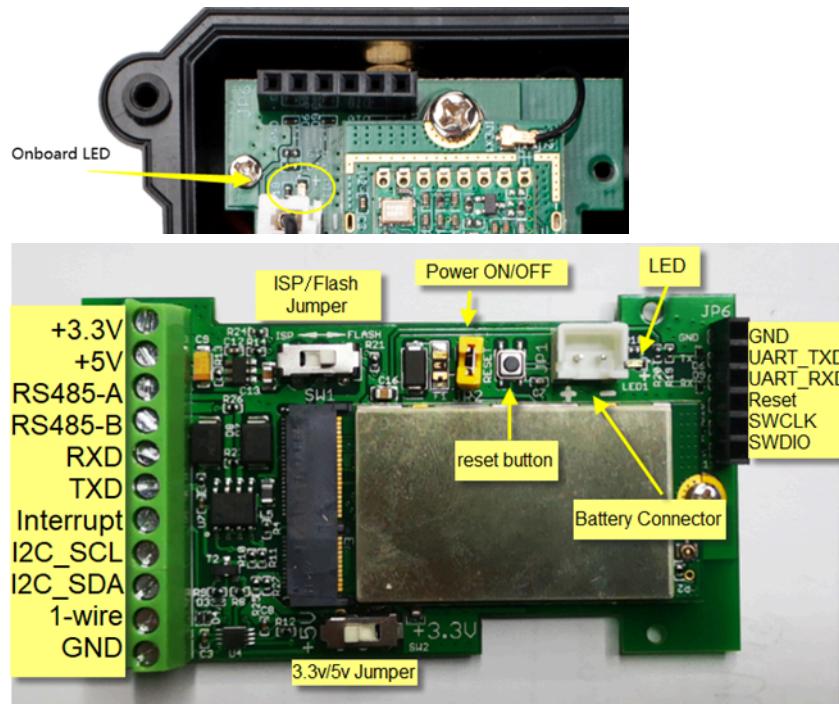
Showing 26 to 30 of 79 results Previous Next





LDDS20 in a LoRaWAN Network





LDDS20 in a LoRaWAN Network

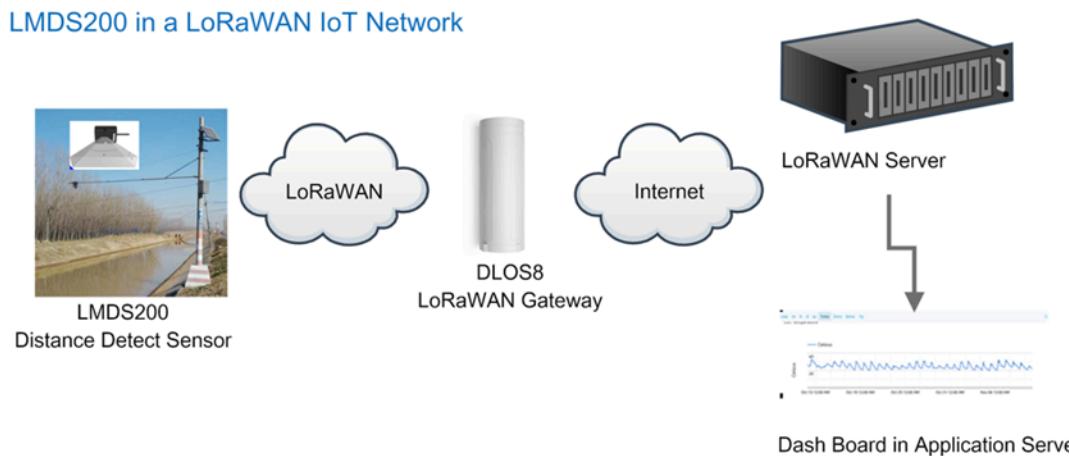




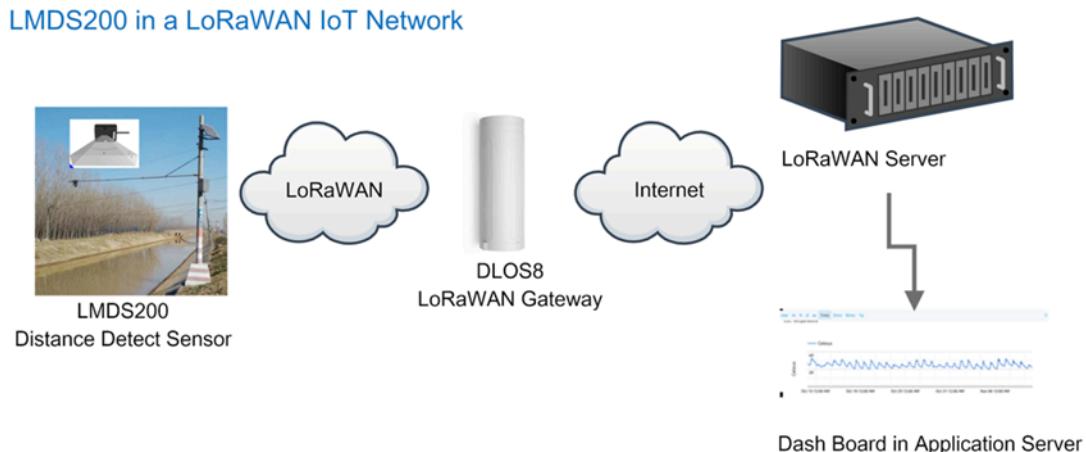




LMDS200 in a LoRaWAN IoT Network



LMDS200 in a LoRaWAN IoT Network



Register end device

From The LoRaWAN Device Repository Manually 1

Frequency plan ⓘ *
 2

LoRaWAN version ⓘ *
 3

Regional Parameters version ⓘ *

Show advanced activation, LoRaWAN class and cluster settings ▾

DevEUI ⓘ *
 0/50 used 4

AppEUI ⓘ *
 5

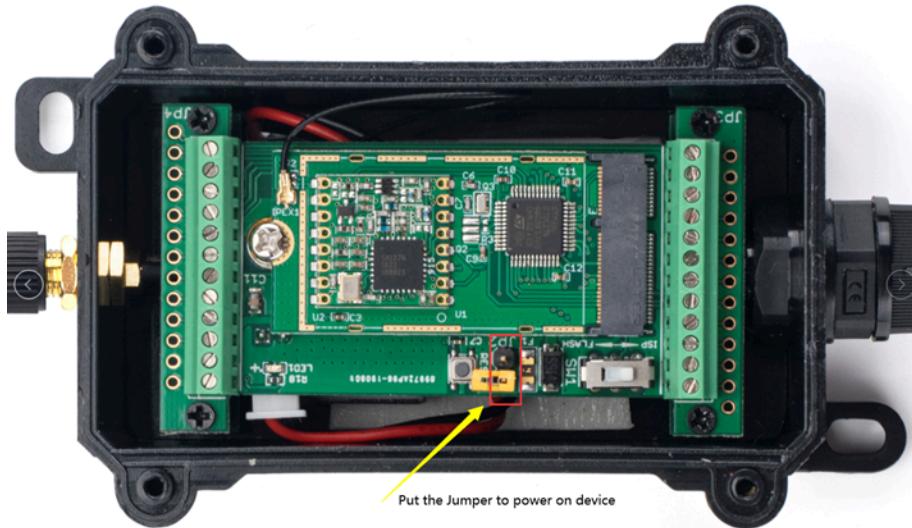
AppKey ⓘ *
 6

End device ID ⓘ *
 7
This value is automatically prefilled using the DevEUI

After registration

View registered end device
 Register another end device of this type

8



User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

The screenshot shows the Datacake Product creation interface. At the top, there's a header for 'lmds200' with an ID of 'lmds200'. Below it, a 'Live data' tab is selected, showing a 'Data preview' section with two log entries: 'Fail to send webhook' and 'Forward uplink data message'. The 'Data preview' section shows a payload with fields like ATDC, Alarm_max, Alarm_min, Interrupt, TDC, and FPort.

Below the preview, there's a 'Add Device' section with a 'LoRaWAN' icon and several other icons for Particle, API, D Zero, D Zero LTE, and PINCODE. A navigation bar at the bottom shows 'STEP 1 Product', 'STEP 2 Network Server', 'STEP 3 Devices', and 'STEP 4 Plan'.

Datacake Product

You can add devices to an existing product on Datacake, create a new empty product or start with one of the templates. Products allow you to share the same configuration (fields, dashboard and more) between devices.

New Product from template
Create new product from a template

Existing Product
Add devices to an existing product

New Product
Create new empty product

New Product 1
If your device is not available as a template, you can start with an empty device. You will have to create the device definition (fields, dashboard) and provide the payload decoder in the device's configuration.

Product Name 2
LDS03A

Next 3

Network Server

Please choose the LoRaWAN Network Server that your devices are connected to.

<input checked="" type="radio"/>  The Things Stack V3 TTN V3 / Things Industries	Uplinks	Downlinks
<input type="radio"/>  The Things Network V2 The old Things Network	Uplinks	Downlinks
<input type="radio"/>  Helium	Uplinks	Downlinks
<input type="radio"/>  LORIOT	Uplinks	Downlinks
<input type="radio"/>  kerlink	Kerlink Wanesy	Uplinks

Showing 1 to 5 of 8 results

Back 2
Next 3

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
-- LoRaWAN Microwave Radar Distance Sensor User Manual

Add Devices

Enter one or more LoRaWAN Device EUIs and the names they will have on Datacake.

New: You can now upload a CSV file with either one column (just the device's DevEUI) or two columns (DevEUI and Name), which will populate the form below.

 Drag and drop a .csv file here or click to choose one

DEVEUI	NAME
99 55 66 33 22 44 11 4f 8 bytes	LDS03A
+ Add another device	

1 2 3

[Back](#) [Next](#)

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
 -- LoRaWAN Microwave Radar Distance Sensor User Manual

Battery Life Calculator										
How to use:										
1. Please do not modify the formula in the table.		2. After selecting the product number and mode, then select the TDC unit, and finally enter the TDC, you can get the predicted battery life.								
3. Explanation of abbreviations : WD → Waiting TX → Transmit RX → Receive										
Product	battery capacity(mAh)									
LMDS200_LoraWAN_Door_Sensor	240									
UNIT	TDC (Uplink Interval)	Work Mode								
min	20									
ACQUA TIME (ms)										
Sleep power (mA/m/s)	Sampling power (mA/m/s)	Tx power (mA/m/s)	RX1 power (mA/m/s)	RX2 power (mA/m/s)	Waiting power (mA/m/s)	Average power (mA)	Detect power (mA/s)	Life expectancy (yrs)		
EU868	DRS_SF7_125K_14dB	8400	427.16444	7367.8548	886.58489	4097.083	757.1706667	0.018706085	0	1.5
	DRS_SF8_125K_14dB	8400	427.16444	13710.2528	950.09481	4097.083	757.1706667	0.02192523	0	1.2
	DRS_SF9_125K_14dB	8400	427.16444	23652.008	1068.0336	4097.083	757.1706667	0.03198736	0	0.8
	DR2_SF10_125K_14dB	8400	427.16444	42244.175	1461.4876	4097.083	757.1706667	0.047792297	0	0.6
	DR1_SF11_125K_14dB	8400	427.16444	94013.4	2270.4829	4097.083	757.1706667	0.095150995	0	0.3
	DR9_SF12_125K_14dB	8400	427.16444	168081	4097.083	4097.083	757.1706667	0.354625338	0	0.2
US915	DR3_SF7_125K_20dB	8400	427.16444	8442.476	682.61089	1587.115	757.1706667	0.016908376	0	1.6
	DR2_SF8_125K_20dB	8400	427.16444	15110.785	913.6491	1587.115	757.1706667	0.022707736	0	1.2
	DR1_SF9_125K_20dB	8400	427.16444	27544.383	941.388	1587.115	757.1706667	0.03279472	0	0.8
	DR9_SF10_125K_20dB	8400	427.16444	48851.32	995.2243	1587.115	757.1706667	0.050235383	0	0.5





pH buffer solution	AT Command to calibrate	Downlink Command	Read Cal Value
4.00	AT+PHCAL=4	0x13 04 Reply with Calibrate payload	AT+PHCAL=? Example 41,61,91
6.86	AT+PHCAL=6	0x13 06 Reply with Calibrate payload	AT+PHCAL=?
9.18	AT+PHCAL=9	0x13 09 Reply with Calibrate payload	AT+PHCAL=?
Factory Default	AT+PHCAL=15	0x13 15 Reply with Calibrate payload	AT+PHCAL=? Example 151

Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
Get Calibration Version Info	Any	14	2

Command Example	Function	Response
AT+TDC=?	Show current transmit Interval	30000 OK the interval is 30000ms = 30s
AT+TDC=60000	Set Transmit Interval	OK Set transmit interval to 60000ms = 60 seconds

Command Example	Function	Response
AT+INTMOD=?	Show current interrupt mode	0 OK the mode is 0 = No interruption
AT+INTMOD=2	Set Transmit Interval 1. (Disable Interrupt), 2. (Trigger by rising and falling edge), 3. (Trigger by falling edge) 4. (Trigger by rising edge)	OK

Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
Get Firmware Version Info	Any	26	2





pH buffer solution	AT Command to calibrate	Downlink Command	Read Cal Value
4.00	AT+PHCAL=4	0x13 04 Reply with Calibrate payload	AT+PHCAL=? Example 41,61,91
6.86	AT+PHCAL=6	0x13 06 Reply with Calibrate payload	AT+PHCAL=?
9.18	AT+PHCAL=9	0x13 09 Reply with Calibrate payload	AT+PHCAL=?
Factory Default	AT+PHCAL=15	0x13 15 Reply with Calibrate payload	AT+PHCAL=? Example 151

pH buffer solution	AT Command to calibrate	Downlink Command	Read Cal Value
4.00	AT+PHCAL=4	0x13 04 Reply with Calibrate payload	AT+PHCAL=? Example 41,61,91
6.86	AT+PHCAL=6	0x13 06 Reply with Calibrate payload	AT+PHCAL=?
9.18	AT+PHCAL=9	0x13 09 Reply with Calibrate payload	AT+PHCAL=?
Factory Default	AT+PHCAL=15	0x13 15 Reply with Calibrate payload	AT+PHCAL=? Example 151

Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
Get Calibration Version Info	Any	14	2

Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
Get Calibration Version Info	Any	14	2

Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
Get Calibration Version Info	Any	14	2

Command Example	Function	Response
AT+TDC=?	Show current transmit Interval	30000 OK the interval is 30000ms = 30s
AT+TDC=60000	Set Transmit Interval	OK Set transmit interval to 60000ms = 60 seconds

Command Example	Function	Response
AT+INTMOD=?	Show current interrupt mode	0 OK the mode is 0 = No interruption
AT+INTMOD=2	Set Transmit Interval 1. (Disable Interrupt), 2. (Trigger by rising and falling edge), 3. (Trigger by falling edge) 4. (Trigger by rising edge)	OK

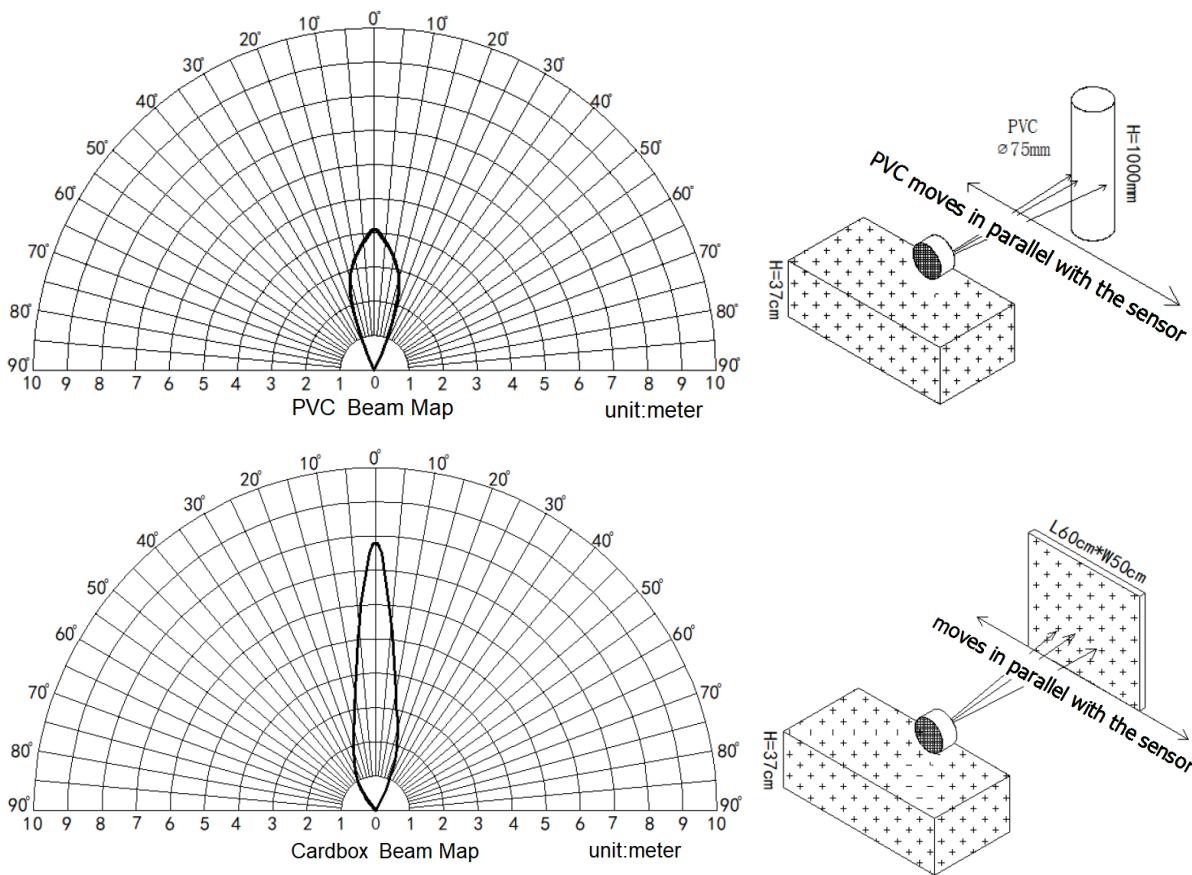
Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
Get Firmware Version Info	Any	26	2





Command Example	Function	Response
AT+INTMOD=?	Show current interrupt mode	0 OK the mode is 0 = No interruption
AT+INTMOD=2	Set Transmit Interval 0- (Disable Interrupt), 1- (Trigger by rising and falling edge), 2- (Trigger by falling edge), 3- (Trigger by rising edge).	OK

Item	Minimum value	Typical value	Maximum value	Unit	Remarks
Storage temperature	-25	25	80	°C	
Storage humidity		65%	90%	RH	(1)
Operating temperature	-15	25	60	°C	
Working humidity		65%	80%	RH	(1)



The screenshot shows the 'THE THINGS STACK Community Edition' interface. The top navigation bar includes links for Overview, Applications (which is highlighted in blue), Gateways, and Orga. Below the navigation, a large heading says 'Add application'. The form contains fields for 'Owner*' (davidhuang), 'Application ID*' (my-new-application), 'Application name' (My new application), and a 'Description' text area (Description for my new application). A note below the description field states: 'Optional application description; can also be used to save notes about the application'. At the bottom of the form is a prominent blue 'Create application' button.

User Manual for LoRaWAN /NB-IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

The screenshot shows a web-based LoRaWAN management interface. At the top, there's a header with a user icon, 'ccc', and 'ID: 123'. Below the header, it says '4 End devices', '2 Collaborators', '2 API keys', and 'Created 95 days ago'. The main area has two tabs: 'General information' and 'Live data'. Under 'General information', there are fields for 'Application ID' (123), 'Created at' (Feb 2, 2021 11:12:30), and 'Last updated at' (Apr 30, 2021 11:00:33). The 'Live data' tab shows a list of recent events with timestamps and descriptions. Below this, there's a section for 'End devices (4)' with columns for 'ID', 'Name', 'DevEUI', 'JoinEUI', and 'Created'. A red arrow points to the '+ Add end device' button. The next section is titled 'Register end device' with two options: 'From The LoRaWAN Device Repository' (selected) and 'Manually'. A red arrow points to the 'From The LoRaWAN Device Repository' link. The '1. Select the end device' step shows dropdowns for 'Brand*' (Dragino Technology Co,...) and 'Model*' (Type to search...). A red arrow points to the 'Brand' dropdown. Below these is a link 'Cannot find your exact end device?'. The '2. Enter registration data' step shows a message 'Please choose an end device first to' and a 'Register end device' button. To the right of this is a dropdown menu listing various device models: LBT1, LDDS20, LDDS75, LDS01, LGT92, LHT65, LSE01, and LSN50-V2. A red arrow points to the 'Model' dropdown.

2. Enter registration data

Frequency plan ⓘ *

The frequency plan used by the end device

AppEUI ⓘ *

The AppEUI uniquely identifies the owner of the end device. If no AppEUI is provided by the device manufacturer (usually for development), it can be filled with zeros.

Register end device

From The LoRaWAN Device Repository

[Manually](#)

Preparation

Activation mode *

- Over the air activation (OTAA)
- Activation by personalization (ABP)
- Multicast
- Do not configure activation

LoRaWAN version ⓘ *

Select...

Network Server address

eu1.cloud.thethings.network

Application Server address

eu1.cloud.thethings.network

External Join Server ⓘ

2. Enter registration data

Frequency plan *

Europe 863-870 MHz (SF12 for RX2) | ▾

The frequency plan used by the end device

AppEUI *

00

The AppEUI uniquely identifies the owner of the end device. If no AppEUI is provided by the device manufacturer (usually for dev

DevEUI  *

.....

The DevEUI is the unique identifier for this end device

AppKey  *

----- |

The root key to derive session keys to secure communication between the end device and the application

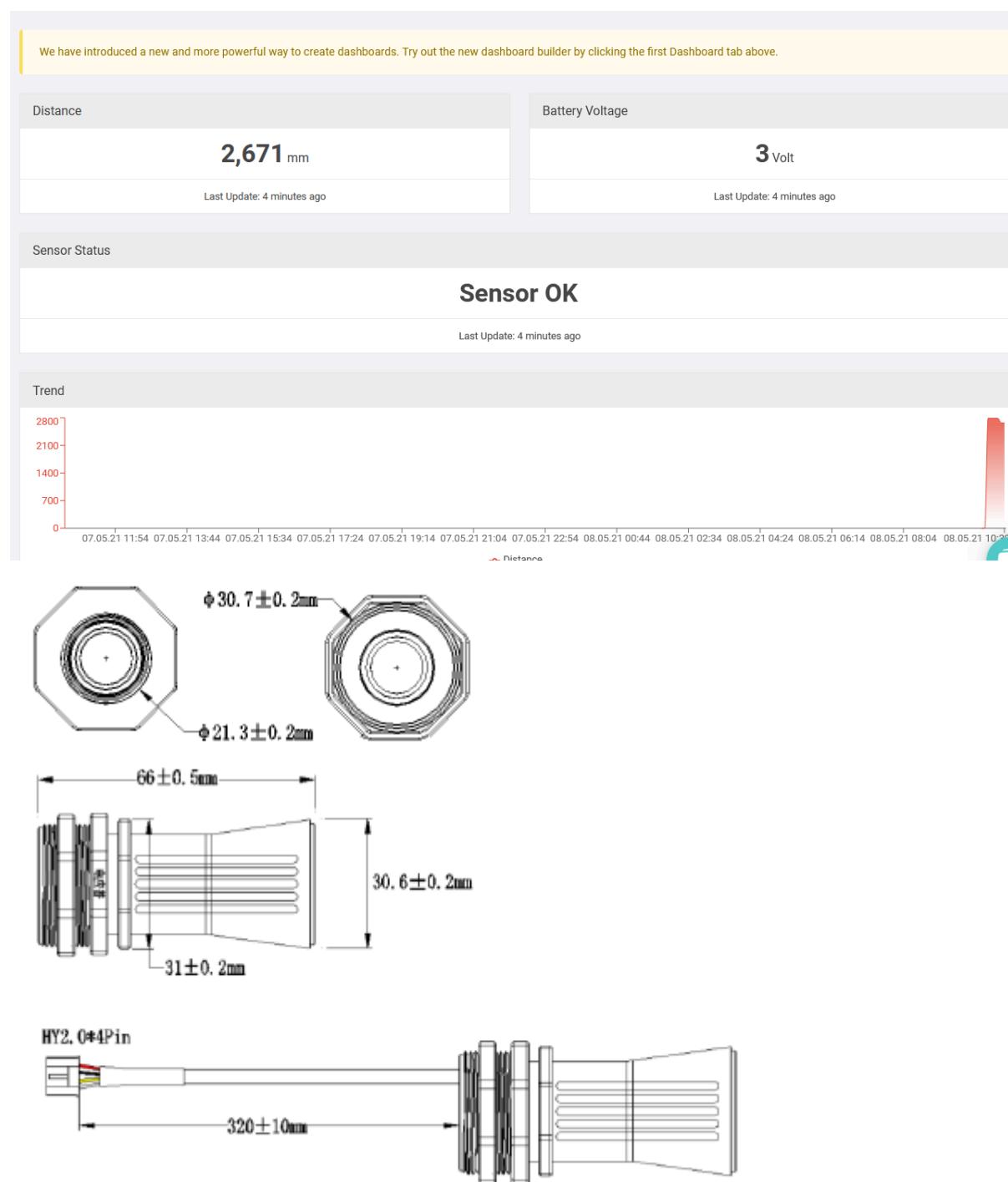
End device ID*

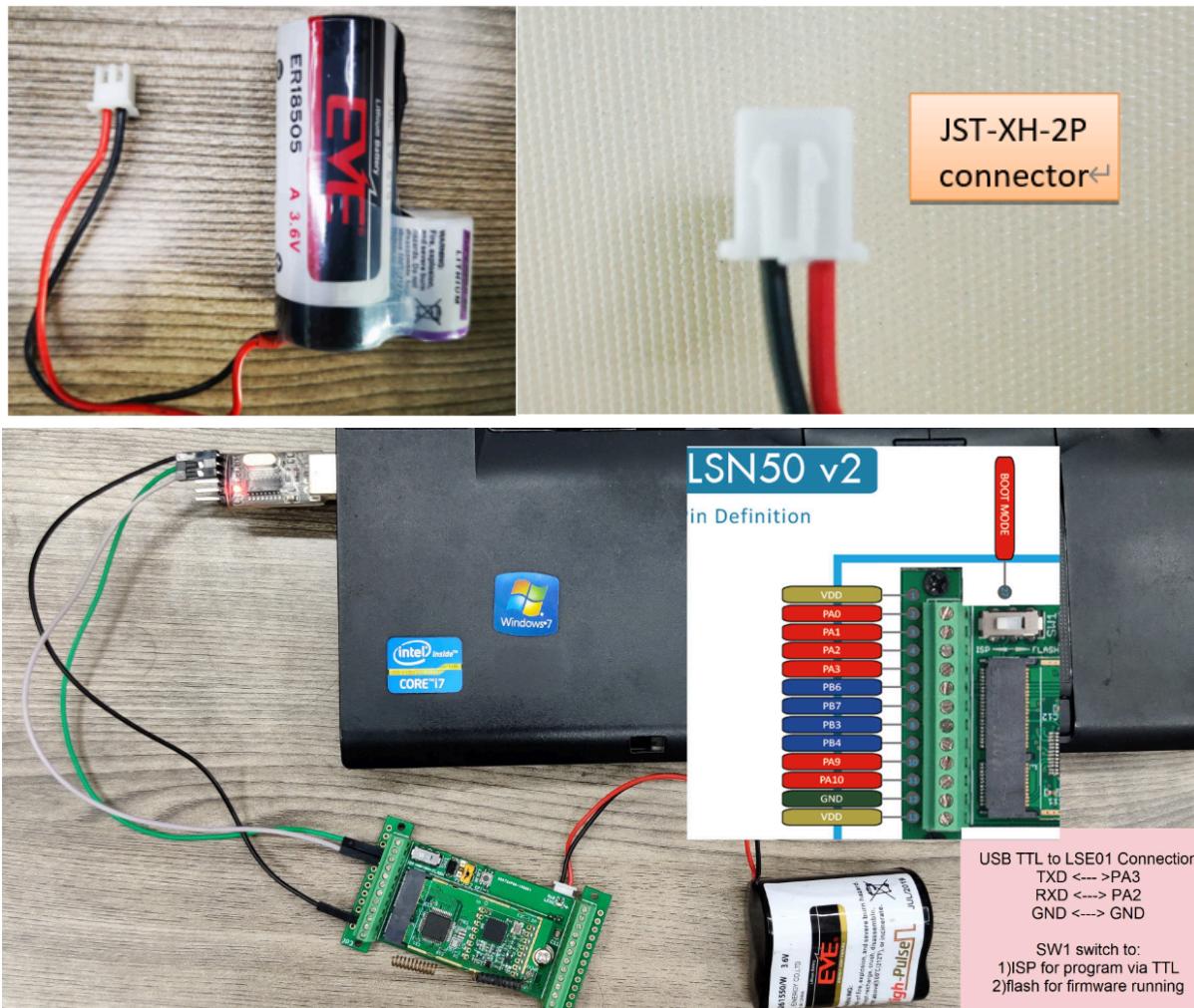
After registration

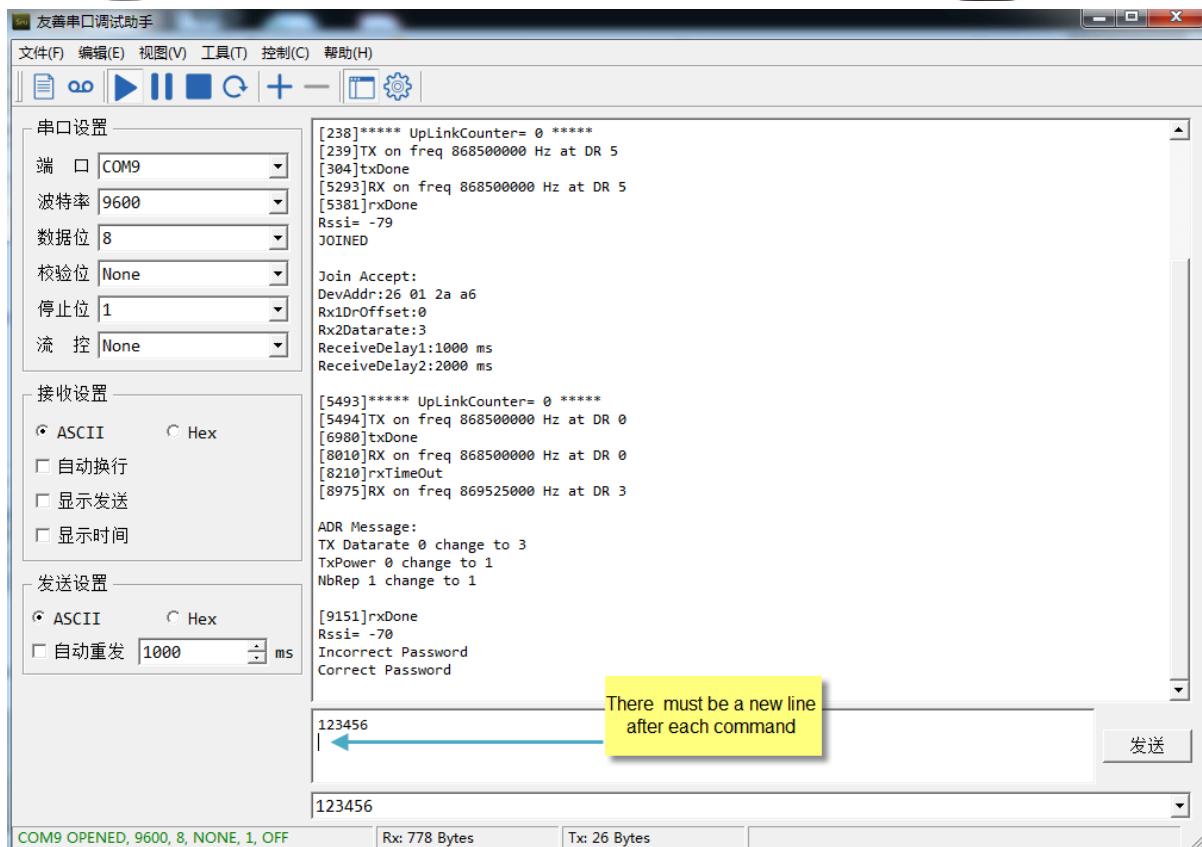
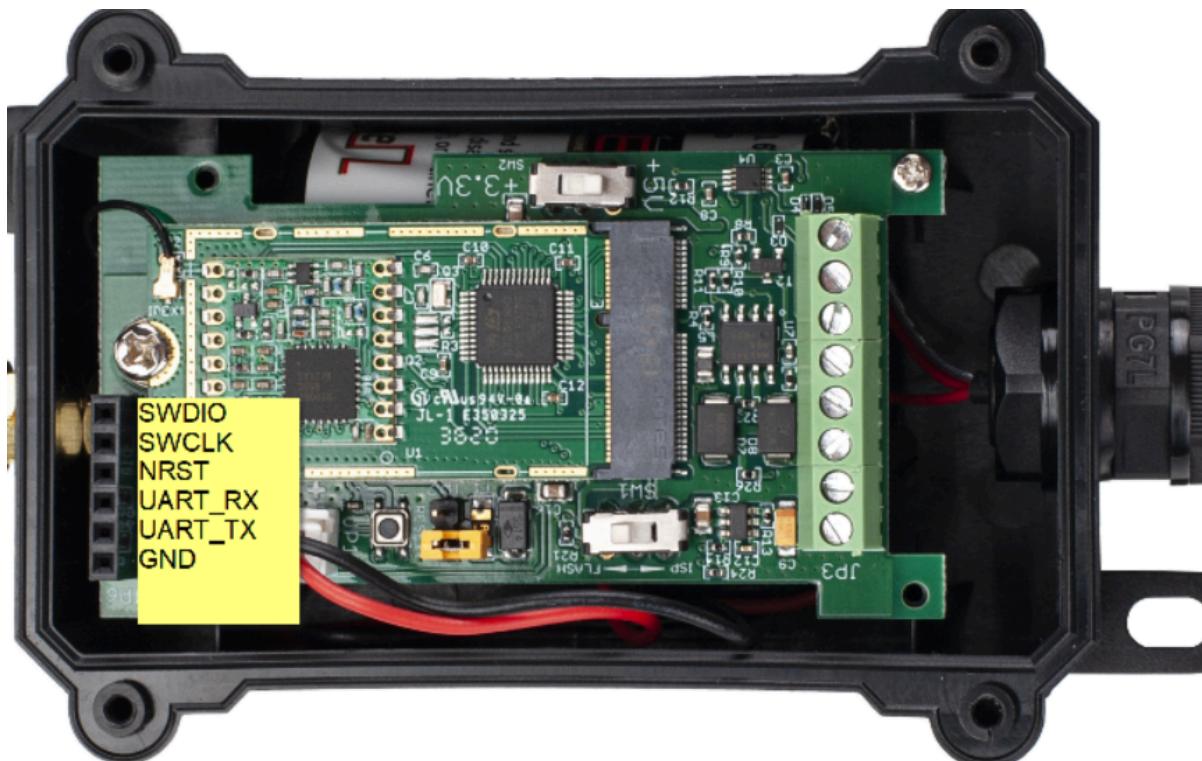


User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual



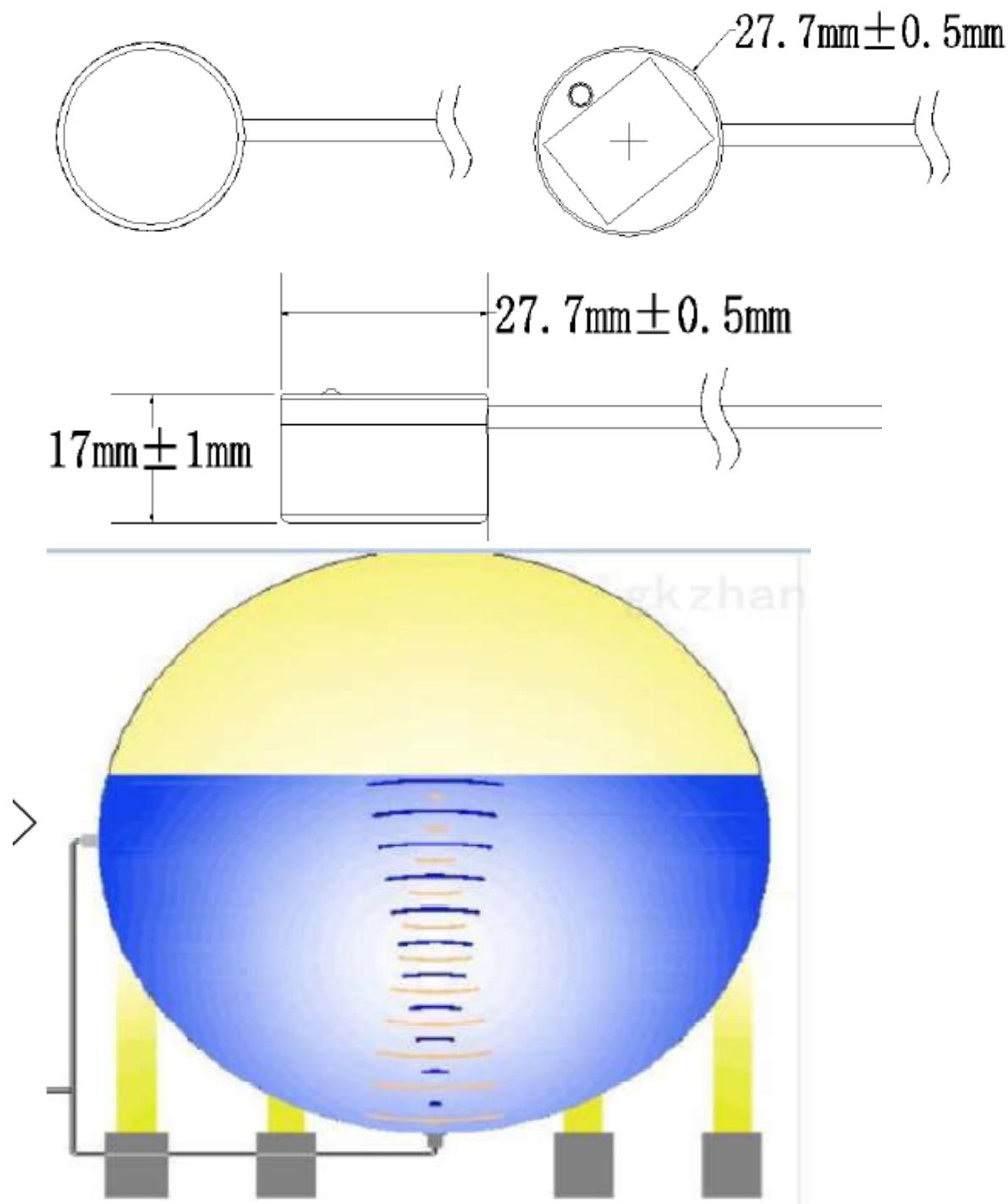


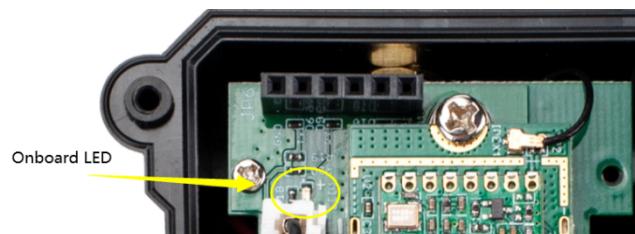
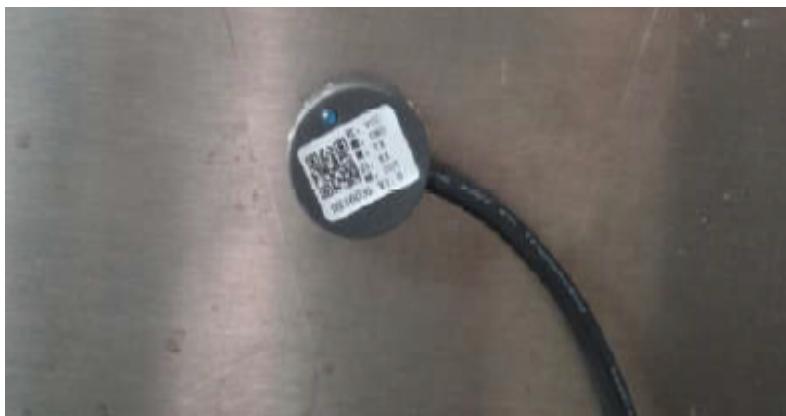


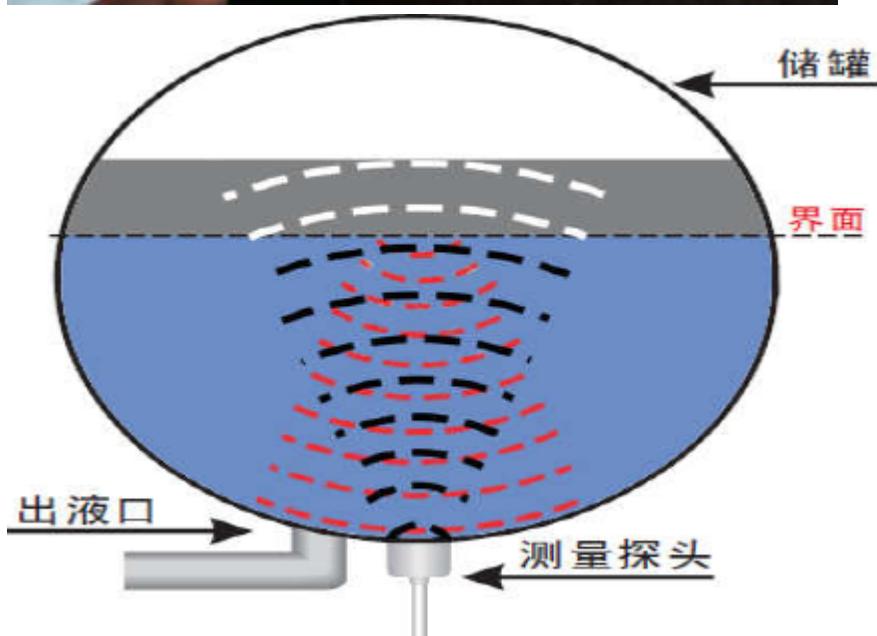
Command Example	Function	Response
AT+TDC=?	Show current transmit Interval	30000 OK the interval is 30000ms = 30s
AT+TDC=60000	Set Transmit Interval	OK Set transmit interval to 60000ms = 60 seconds

Command Example	Function	Response
AT+INTMOD=?	Show current interrupt mode	0 OK the mode is 0 = No interruption
AT+INTMOD=2	Set Transmit Interval 0- (Disable Interrupt), 1- (Trigger by rising and falling edge), 2- (Trigger by falling edge), 3- (Trigger by rising edge).	OK

Command Example	Function	Response
AT+INTMOD=?	Show current interrupt mode	0 OK the mode is 0 = No interruption
AT+INTMOD=2	Set Transmit Interval 0- (Disable Interrupt), 1- (Trigger by rising and falling edge), 2- (Trigger by falling edge), 3- (Trigger by rising edge).	OK







Metal Surface with paint needs polish



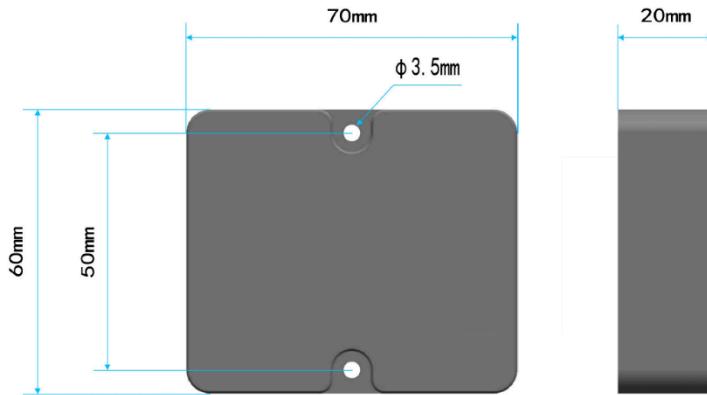


Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
TDC (Transmit Time Interval)	Any	01	4
RESET	Any	04	2
AT+CFM	Any	05	4
INTMOD	Any	06	4





User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
-- LoRaWAN Microwave Radar Distance Sensor User Manual



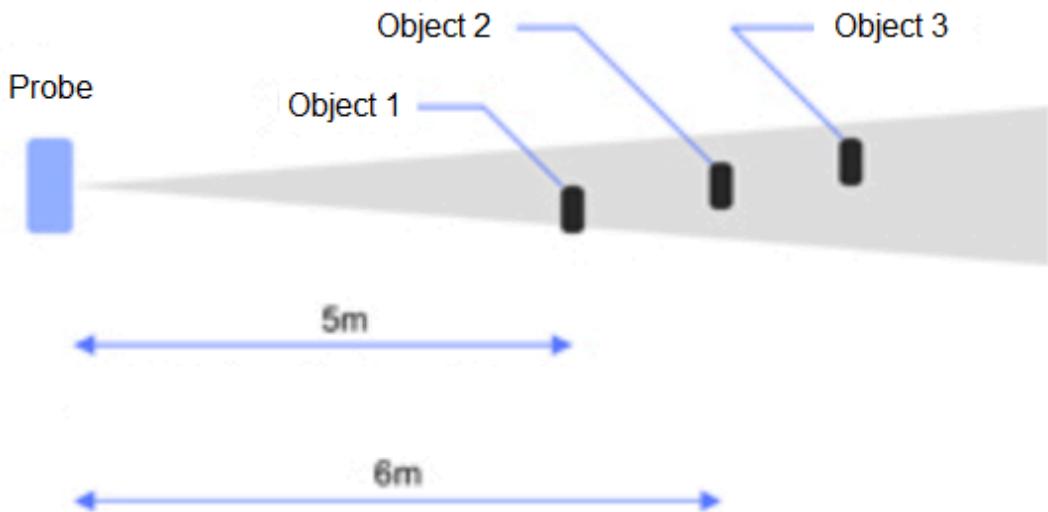
lmds200
ID: lmds200

↑ 1 ↓ n/a • Last activity 6 seconds ago ⓘ

Overview **Live data** Messaging Location Payload formatters Claiming General settings

Time	Type	Data preview
15:32:27	Fail to send webhook	Error:undefined:undefined
↑ 15:32:27	Forward uplink data message	Payload: { Bat: 3.304, Fre_band: "EU868", Sensor_model: "LMDS200", Sub_band: 0, Ver: 100 } 0C 01 00 01 00 0C E8 FPort: 5
↑ 15:32:22	Forward join-accept message	
GD 15:32:28	Accept join-request	

Device Status (FPORT=5)					
Size (bytes)	1	2	1	1	2
Value	Sensor Model	Firmware Version	Frequency Band	Sub-band	BAT



```

↑ 14:28:18 Forward uplink data message          Payload: { Bat: 3.328, DALARM_count: 19, Distance_alarm: 1, Interrupt_alarm: 0, dis1: 115, dis2: 164 } 0D 00 00 73 00 A4 4E FP
① 14:29:18 Fail to send webhook               Error:undefined:undefined
↑ 14:29:18 Forward uplink data message          Payload: { Bat: 3.328, DALARM_count: 18, Distance_alarm: 1, Interrupt_alarm: 0, dis1: 115, dis2: 164 } 0D 00 00 73 00 A4 4A FP
① 14:28:18 Fail to send webhook               Error:undefined:undefined
↑ 14:28:18 Forward uplink data message          Payload: { Bat: 3.33, DALARM_count: 17, Distance_alarm: 1, Interrupt_alarm: 0, dis1: 115, dis2: 164 } 0D 02 00 73 00 A4 46 FPPo

```

Status & Alarm field			
Size (bit)	6	1	1
Value	DALARM Counter	Distance Alarm 0: Normal Value 1: Distance Alarm	Interrupt Alarm 0: No Alarm 1: external Interrupt Alarm

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

Payload Decoder

When your devices sends data, the payload will be passed to the payload decoder, alongside the event's name. The payload decoder then transforms it to measurements.

```

1: function decode(t,r){var sa=(bytes[4]<>bytes[5])>>>"INT";"FALSE"
2: var bbs=bytes[6]<>bytes[7]>>>"GATE","CLOSING"
3: var ddb=bytes[8]<>bytes[9]>>>bytes[4]>>>bytes[4].toString(16);
4: var eea=bytes[10]<>bytes[11]<>bytes[12]<>bytes[13]>>>bytes[4]>>>bytes[4].toString(16);
5: var eea_gw=bytes[bytes[14]<>bytes[15]<>bytes[16]<>bytes[17]<>bytes[18]<>bytes[19]]>>>bytes[4]>>>bytes[4].toString(16));
6: var strg=[bytes[20]<>bytes[21]<>bytes[22]<>bytes[23]>>>bytes[24]>>>bytes[24].toString(16));
7: var strg2=[bytes[25]<>bytes[26]<>bytes[27]<>bytes[28]>>>bytes[29]>>>bytes[29].toString(16)];
8: return strg;
9: }
10: function getc(c_num){
11: if(parseFloat(c_num)<0) c_num = "0"+c_num;
12: return c_num;
13: }
14: function getmDate(t,r){
15: var c_date;
16: if(r> 9999999999) c_date = new Date(parseFloat(r));
17: else
18: c_date = new Date(parseFloat(r)*1000);
19: var c_year = c_date.getFullYear();
20: c Day = c_date.getDate();
21: c Month = c_date.getMonth();
22: c Hour = c_date.getHours();
23: c Min = c_date.getMinutes();
24: c Sec = c_date.getSeconds();
25: var c_time = c_year + "-" + getif(c_Month) + "-" + getif(c_Day) + " " + getif(c_Hour) + ":" + getif(c_Min) + ":" + getif(c_Sec);
26: return c_time;
27: }
28: var cYear = c_date.getFullYear();
29: cDay = c_date.getDate();
30: cMonth = c_date.getMonth();
31: cHour = c_date.getHours();
32: cMin = c_date.getMinutes();
33: cSec = c_date.getSeconds();
34: }
35: return c_time;
36: }
37: 
```

Payload

Output

console.log Output

Recognized measurements

Port: 1 Try Decoder

Fields

Fields describe the data the device will store.

NAME	2	IDENTIFIER	TYPE	CURRENT VALUE	LAST UPDATE	⋮
BAT		BAT	Float	0	5 minutes ago	⋮
DOOR_OPEN_TIMES		DOOR_OPEN_TIMES	Float	0	3 minutes ago	⋮
LAST_DOOR_OPEN_DURATION		LAST_DOOR_OPEN_DURATION	Float	0	3 minutes ago	⋮
ALARM		ALARM	Boolean	False	a few seconds ago	⋮
DOOR_OPEN_STATUS		DOOR_OPEN_STATUS	Boolean	False	a few seconds ago	⋮

DATACAKE

Fleet > LDS03A

LDS03A

Serial Number 9955663322441140	Last update Never
-----------------------------------	----------------------

Dashboard History Downlinks Configuration Debug Rules Permissions

Public Link + Add Widget

Desktop Mobile More ▾

Basics **Data** Appearance Gauge Timeframe

Field 1

Please Select

Please Select

BAT 2

DOOR_OPEN_TIMES

LAST_DOOR_OPEN_DURATION

Decimal Places

2

Cancel ✓ Save 3

This screenshot shows the 'Data' tab of a configuration interface. At the top, there are tabs for Basics, Data, Appearance, Gauge, and Timeframe. The 'Data' tab is selected and highlighted in blue. Below the tabs, there is a section labeled 'Field' with the value '1'. A dropdown menu is open, showing several options: 'Please Select', 'Please Select', 'BAT' (which is highlighted in blue), 'DOOR_OPEN_TIMES', and 'LAST_DOOR_OPEN_DURATION'. To the right of the dropdown is a large input field. Below the dropdown, there is a section for 'Decimal Places' with the value '2'. At the bottom right are two buttons: 'Cancel' and a blue 'Save' button with a checkmark icon.

Edit Boolean Widget

New Boolean

Basics Appearance **Data** 1

Field

Please Select

Please Select 2

ALARM

DOOR_OPEN_STATUS

Cancel ✓ Save 3

This screenshot shows the 'Edit Boolean Widget' configuration screen. At the top, it says 'Edit Boolean Widget' and 'New Boolean'. Below that is a red circular icon. There are tabs for Basics, Appearance, and Data, with 'Data' selected and highlighted in blue. A red arrow points to the 'Data' tab. Below the tabs is a section labeled 'Field' with the value '1'. A dropdown menu is open, showing 'Please Select', 'Please Select' (highlighted in blue), 'ALARM', and 'DOOR_OPEN_STATUS'. To the right is a large input field. At the bottom right are 'Cancel' and 'Save' buttons. A red arrow points to the 'Save' button.

User Manual for LoRaWAN /NB-IoT End Nodes - LMDS200

-- LoRaWAN Microwave Radar Distance Sensor User Manual

LDS03A

Serial Number: 9955663322441140 Last update: Fri Nov 12 2021 16:51:53 GMT+0800

Dashboard **History** **Downlinks** **Configuration** **Debug** **Rules** **Permissions** **Public Link** **+ Add Widget** **Switch**

Widgets:

- BAT: 0
- DOOR_OPEN_TIMES: 1 (6 minutes ago)
- DOOR_OPEN_STATUS: Red circle (7 minutes ago)
- LAST_DOOR_OPEN_DURATION: 0 (6 minutes ago)

Add Device

Product Templates:

- LoRaWAN
- PARTICLE
- API
- D Zero
- D Zero LTE
- PINCODE

Step Progress:

- STEP 1 Product
- STEP 2 Network Server
- STEP 3 Devices
- STEP 4 Plan

Datacake Product

You can add devices to an existing product on Datacake, create a new empty product or start with one of the templates. Products allow you to share the same configuration (fields, dashboard and more) between devices.

New Product from template
Create new product from a template

Existing Product
Add devices to an existing product

New Product
Create new empty product

New Product
If your device is not available as a template, you can start with an empty device. You will have to create the device definition (fields, dashboard) and provide the payload decoder in the device's configuration.

Product Name: LDS03A

Next

Network Server

Please choose the LoRaWAN Network Server that your devices are connected to.

	The Things Stack V3 TTN V3 / Things Industries	Uplinks	Downlinks
	The Things Network V2 The old Things Network	Uplinks	Downlinks
	helium Helium	Uplinks	Downlinks
	LORIOT	Uplinks	Downlinks
	Kerlink Wanesy	Uplinks	

Showing 1 to 5 of 8 results

Previous Next

Back

Next

Add Devices

Enter one or more LoRaWAN Device EUIs and the names they will have on Datacake.

New: You can now upload a CSV file with either one column (just the device's DevEUI) or two columns (DevEUI and Name), which will populate the form below.

Drag and drop a .csv file here or click to choose one

DEVEUI	NAME
99 55 66 33 22 44 11 44 8 bytes	LDS03A

+ Add another device

1 2 3

Back Next

DATACAKE

Fleet > LDS03A

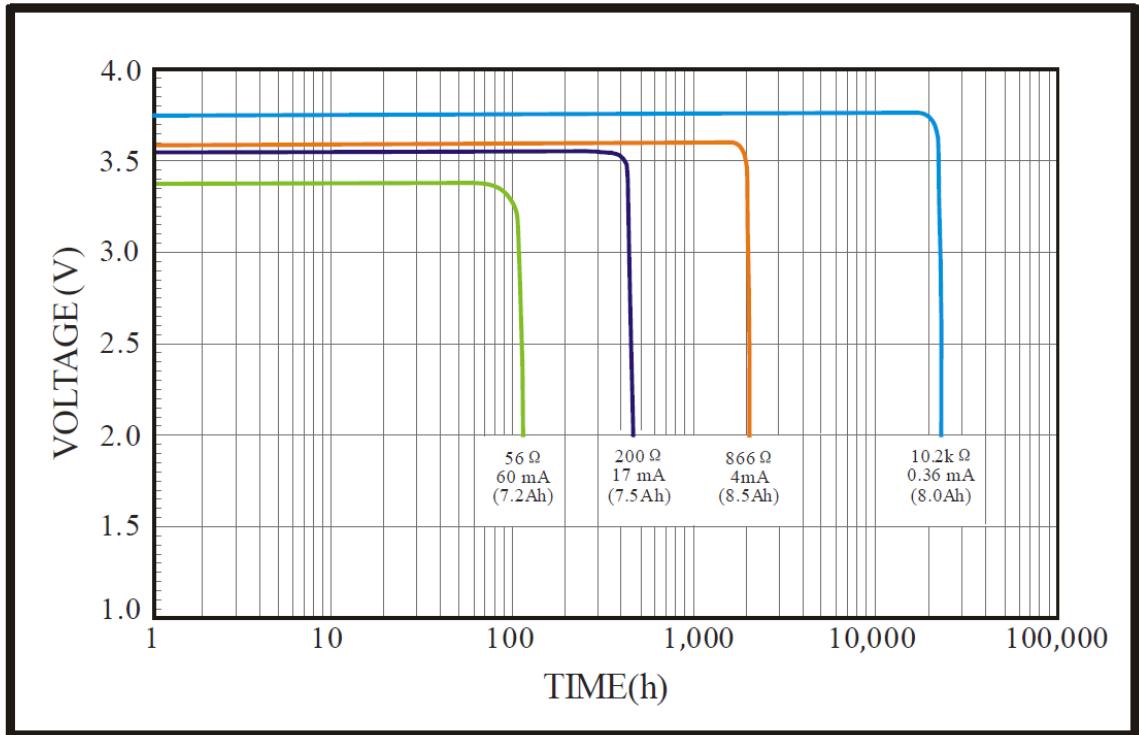
LDS03A

Serial Number 9955663322441140	Last update Never
Dashboard History Downlinks Configuration Debug Rules Permissions	Configuration

General Configuration

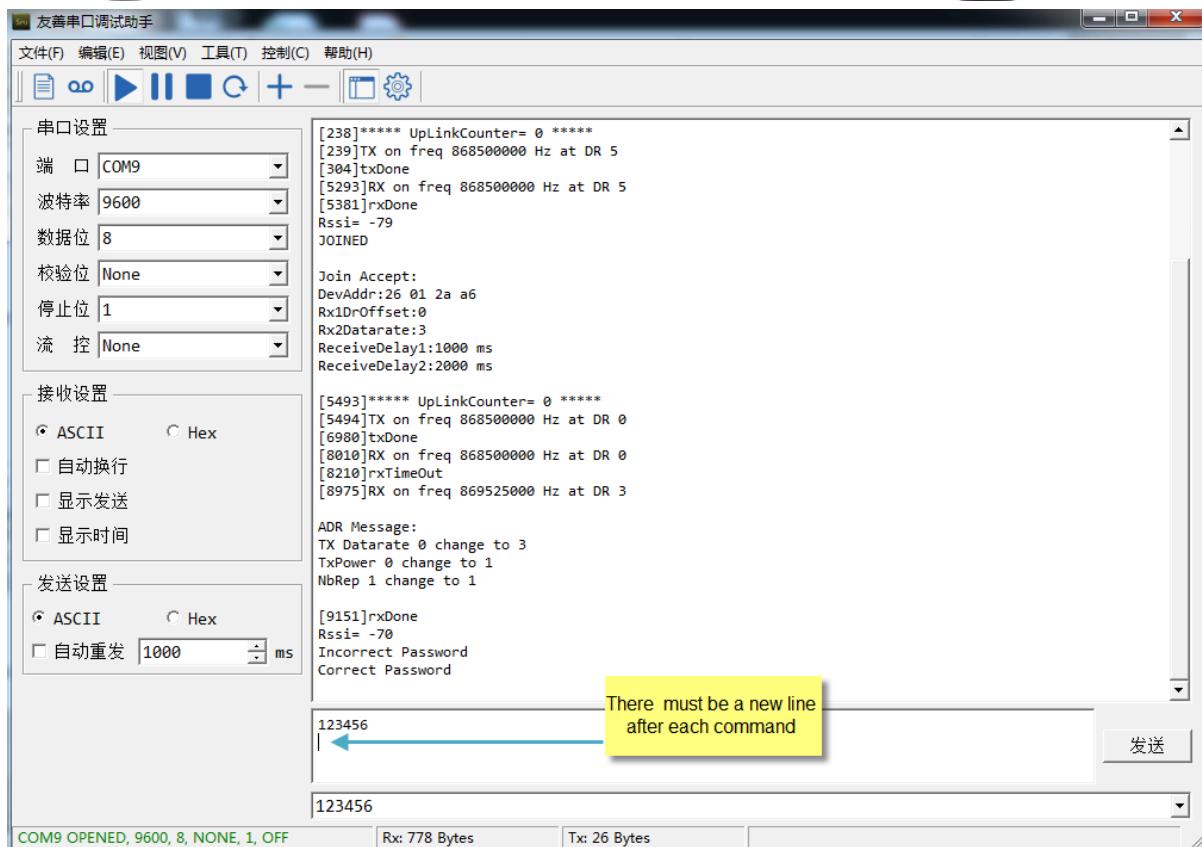
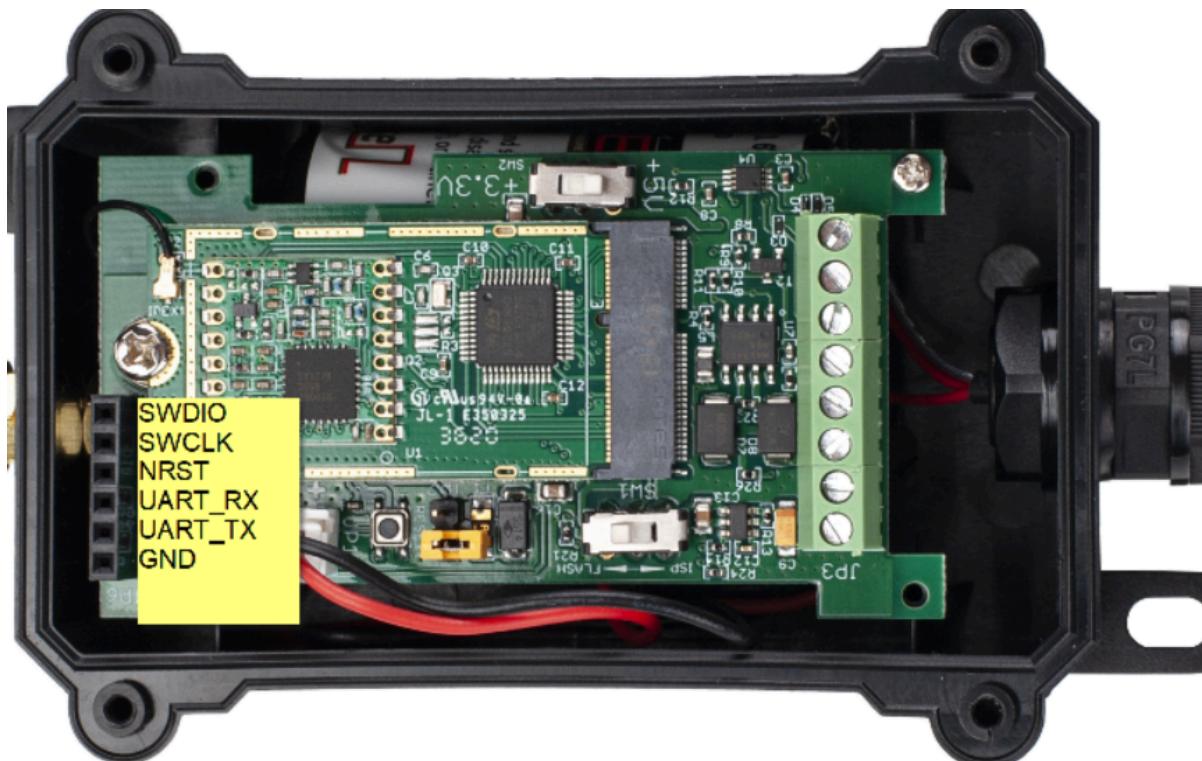
Device Name
LDS03A

1. Typical discharge profile at +20 °C(Typical value)



How to use: 1.Please do not modify the formula in the table 2.After selecting the product number and model, then select the TDC unit, and finally enter the TDC you can get the predicted battery life 3.Explanation of abbreviations : WWD->Watchdog TX->Transmit RX->Receive											
Battery Life Calculator											
Product	battery capacity(mah)		Current (mA)								
	LD501_LoRaWAN_Door_Sensor	240	W	D	sleep	T	X	R	R	W	D
UNIT	TDC (Uplink Interval)	Work Mode	time (ms)								sleep
min	20	MOD=1									
	Sleep power (mA*s)	Sampling power (mA*s)	TX power (mA*s)	RX1 power (mA*s)	I02 power (mA*s)	Watchdog power (mA*s)	Average power (mA)	Detec power (mA*s)	Life expectancy (yr)		
EU868	DR5_SF7_125K_14dB	8400	427.16444	7367.8544	880.59488	4097.083	757.1706667	0.018268685	0	1.5	
	DR4_SF8_125K_14dB	8400	427.16444	13210.2528	950.0943	4097.083	757.1706667	0.023192523	0	1.2	
	DR3_SF9_125K_14dB	8400	427.16444	23652.608	1068.0336	4097.083	757.1706667	0.031986736	0	0.8	
	DR2_SF10_125K_14dB	8400	427.16444	42244.125	1461.4876	4097.083	757.1706667	0.047792297	0	0.6	
	DR1_SF11_125K_14dB	8400	427.16444	94013.4	2230.4828	4097.083	757.1706667	0.091509095	0	0.3	
	DR0_SF12_125K_14dB	8400	427.16444	168081	4097.083	4097.083	757.1706667	0.154623338	0	0.2	
US915	DR3_SF7_125K_20dB	8400	427.16444	8441.476	681.61989	1587.135	757.1706667	0.016908376	0	1.6	
	DR2_SF8_125K_20dB	8400	427.16444	15170.785	913.6491	1587.135	757.1706667	0.022707198	0	1.2	
	DR1_SF9_125K_20dB	8400	427.16444	27254.383	941.388	1587.135	757.1706667	0.03279472	0	0.8	
	DR0_SF10_125K_20dB	8400	427.16444	48745.32	995.2243	1587.135	757.1706667	0.050735363	0	0.5	

Command Example	Function	Response
AT+ATDC?	Show current alarm transmit Interval	1 OK the interval is 1 minute
AT+ATDC=1	Set Alarm Transmit Interval	OK Set alarm transmit interval to 1 min





```
[10:51:38.308] 8034700 1970/1/3 18:27:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:38.368] 8034710 1970/1/3 18:28:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:38.438] 8034720 1970/1/3 18:29:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:38.508] 8034730 1970/1/3 18:30:47 88 0 0 0 distance1=3276 distance2=308
[10:51:38.568] 8034740 1970/1/3 18:31:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:38.638] 8034750 1970/1/3 18:32:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:38.708] 8034760 1970/1/3 18:33:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:38.768] 8034770 1970/1/3 18:34:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:38.868] 8034780 1970/1/3 18:35:47 3283 0 0 0 distance1=3276 distance2=309
[10:51:38.938] 8034790 1970/1/3 18:36:47 3283 0 0 0 distance1=3276 distance2=309
[10:51:39.008] 80347A0 1970/1/3 18:37:47 3283 0 0 0 distance1=3276 distance2=309
[10:51:39.068] 80347B0 1970/1/3 18:38:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:39.138] 80347C0 1970/1/3 18:39:47 3283 0 0 0 distance1=3276 distance2=310
[10:51:39.208] 80347D0 1970/1/3 18:40:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:39.268] 80347E0 1970/1/3 18:41:47 3283 0 0 0 distance1=3276 distance2=308
[10:51:39.338] 80347F0 1970/1/3 18:42:47 3283 0 0 0 distance1=3276 distance2=307
[10:51:39.438] 8034800 1970/1/3 18:43:47 3283 0 0 0 distance1=3276 distance2=308
```

User Manual for LoRaWAN /NB -IoT End Nodes - LMDS200
-- LoRaWAN Microwave Radar Distance Sensor User Manual

```
803BC50 2023/7/31 22:20:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.241] 803BC60 2023/7/31 22:21:40 3283 0 0 0 distance1=260 distance2=383
[10:53:51.303] 803BC70 2023/7/31 22:22:40 3283 0 0 0 distance1=260 distance2=383
[10:53:51.379] 803BC80 2023/7/31 22:23:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.441] 803BC90 2023/7/31 22:24:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.503] 803BCA0 2023/7/31 22:25:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.579] 803BCB0 2023/7/31 22:26:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.641] 803BCC0 2023/7/31 22:27:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.734]
803BCD0 2023/7/31 22:28:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.811] 803BCE0 2023/7/31 22:29:40 3283 0 0 0 distance1=260 distance2=383
[10:53:51.874] 803BCF0 2023/7/31 22:30:40 3283 0 0 0 distance1=260 distance2=382
[10:53:51.936] 803BD00 2023/7/31 22:31:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.014] 803BD10 2023/7/31 22:32:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.076] 803BD20 2023/7/31 22:33:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.138] 803BD30 2023/7/31 22:34:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.214] 803BD40 2023/7/31 22:35:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.303]
803BD50 2023/7/31 22:36:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.384] 803BD60 2023/7/31 22:37:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.447] 803BD70 2023/7/31 22:38:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.508] 803BD80 2023/7/31 22:39:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.572] 803BD90 2023/7/31 22:40:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.650] 803BDA0 2023/7/31 22:41:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.711] 803BDB0 2023/7/31 22:42:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.773] 803BDC0 2023/7/31 22:43:40 3283 0 0 0 distance1=260 distance2=382
[10:53:52.870]
```

Time	Type	Data preview	Verbose stream	Export as JSON	Pause	Clear
↓ 11:20:16	Schedule data downlink for transmi...	DevAddr: 26 00 95 42 <=> Rx1 Delay: 6				
↑ 11:20:16	Forward uplink data message	DevAddr: 26 00 95 42 <=> Payload: { Bat: 3.283, DALARM_count: 0, Distance_alarm: 0, Interrupt_alarm: 0, dis1: 1, dis2: 1 } 8C 03 00 01 00 01 00 <=>				FPC
↑ 11:20:16	Successfully processed data message	DevAddr: 26 00 95 42 <=>				
↓ 11:19:29	Schedule data downlink for transmi...	DevAddr: 26 00 95 42 <=> Rx1 Delay: 6				
↑ 11:19:29	Forward uplink data message	DevAddr: 26 00 95 42 <=> Payload: { Bat: 3.283, DALARM_count: 0, Distance_alarm: 0, Interrupt_alarm: 0, dis1: 1, dis2: 1 } 8C 03 00 01 00 01 00 <=>				FPC
↑ 11:19:29	Successfully processed data message	DevAddr: 26 00 95 42 <=>				
↑ 11:18:36	Forward uplink data message	DevAddr: 26 00 95 42 <=> Payload: { DATALOG: "[3.283,268,383,NO,FA", 0, 2023-07-31 14:29:48], [3.283,268,383,NO,FA", 0, 2023-07-31 14:38:48], [3.283,268,383,				
↑ 11:18:36	Successfully processed data message	DevAddr: 26 00 95 42 <=>				
↓ 11:18:29	Schedule data downlink for transmi...	DevAddr: 26 00 95 42 <=> FPort: 1 Confirmed downlink MAC payload: 0E DC 39 02 49 DE 99 3A <=> Rx1 Delay: 6				
↑ 11:18:29	Forward uplink data message	DevAddr: 26 00 95 42 <=> Payload: { Bat: 3.283, DALARM_count: 0, Distance_alarm: 0, Interrupt_alarm: 0, dis1: 1, dis2: 1 } 8C 03 00 01 00 01 00 <=>				FPC
↑ 11:18:29	Successfully processed data message	DevAddr: 26 00 95 42 <=>				