
LMDS200 – LoRaWAN-Mikrowellenradar- Entfernungssensor Benutzerhandbuch

Zuletzt geändert von Mengting Qiu

am 07.03.2024 um 08:41 Uhr

Inhaltsverzeichnis

1. Einführung	3
1.1 Was ist der LMDS200 Mikrowellen-Radar-Entfernungssensor?	3
1.2 Funktionen	4
1.3 Spezifikationen der Radarsonde	4
1.4 Lagerungs- und Betriebstemperatur	5
1.5 Anwendungen	5
1.6 Installation	5
1.7 Pin-Zuordnung und Einschalten	6
2. Betriebsmodus	6
2.1 So funktioniert es	6
2.2 Beispiel für die Verwendung im LoRaWAN-Netzwerk	7
2.3 Uplink-Nutzlast	12
2.3.1 Gerätestatus, FPORT=5	12
2.3.2 Sensorkonfiguration, FPORT=4	13
2.3.3 Entfernung, Uplink FPORT=2	13
2.3.4 Decoder in TTN V3	15
2.4 Daten auf Datacake anzeigen	15
3. Konfigurieren Sie LMDS200 über AT-Befehl oder LoRaWAN-Downlink	21
3.1 Sendeintervallzeit einstellen (0x01)	22
3.2 Alarmsendeintervallzeit einstellen (0x0D)	22
3.3 Alarmreichweite einstellen (0xA2)	23
3.4 Alarmmodus aktivieren/deaktivieren (0xA3)	23
3.5 Unterbrechungsmodus einstellen (0x06)	23
4. Batterie und Stromverbrauch	24
5. Häufig gestellte Fragen	24
5.1 Verwendung des AT-Befehls zur Konfiguration des LMDS200	24
5.2 Wie aktualisiert man die Firmware?	27
5.3 So ändern Sie die LoRa-Frequenzbänder/Region	27
6. Fehlerbehebung	27
6.1 AT-Befehlseingabe funktioniert nicht	27
7. Bestellinformationen	28
8. Verpackungsinformationen	28
9. Support	28



Inhaltsverzeichnis:

- [1. Einleitung](#)
 - [1.1 Was ist der LMDS200 Mikrowellen-Radar-Entfernungssensor?](#)
 - [1.2 Merkmale](#)
 - [1.3 Spezifikationen der Radarsonde](#)
 - [1.4 Lagerungs- und Betriebstemperatur](#)
 - [1.5 Anwendungen](#)
 - [1.6 Installation](#)
 - [1.7 Pinbelegung und Einschalten](#)
- [2. Betriebsmodus](#)
 - [2.1 Funktionsweise](#)
 - [2.2 Anwendungsbeispiel für LoRaWAN-Netzwerk](#)
 - [2.3 Uplink-Nutzlast](#)
 - [2.3.1 Gerätestatus, FPORT=5](#)
 - [2.3.2 Sensorkonfiguration, FPORT=4](#)
 - [2.3.3 Entfernung, Uplink FPORT=2](#)
 - [2.3.4 Decoder in TTN V3](#)
 - [2.4 Daten auf Datacake anzeigen](#)
- [3. LMDS200 über AT-Befehl oder LoRaWAN-Downlink konfigurieren](#)
 - [3.1 Sendeintervallzeit einstellen \(0x01\)](#)
 - [3.2 Alarm-Sendeintervallzeit einstellen \(0x0D\)](#)
 - [3.3 Alarmreichweite einstellen \(0xA2\)](#)
 - [3.4 Alarmmodus aktivieren/deaktivieren \(0xA3\)](#)
 - [3.5 Unterbrechungsmodus einstellen \(0x06\)](#)
- [4. Batterie und Stromverbrauch](#)
- [5. Häufig gestellte Fragen](#)
 - [5.1 Verwendung des AT-Befehls zur Konfiguration des LMDS200](#)
 - [5.2 Wie aktualisiert man die Firmware?](#)
 - [5.3 Ändern der LoRa-Frequenzbänder/Region](#)
- [6. Fehlerbehebung](#)
 - [6.1 Die Eingabe von AT-Befehlen funktioniert nicht](#)
- [7. Bestellinformationen](#)
- [8. Verpackungsinformationen](#)
- [9. Support](#)

1. Einführung

1.1 Was ist der LMDS200 Mikrowellen-Radar-Entfernungssensor?

Der Dragino LMDS200 ist ein **LoRaWAN-Mikrowellenradar-Entfernungssensor**. Er nutzt **24-GHz-Mikrowellen**, um die Entfernung zwischen dem Sensor und verschiedenen Objekten zu ermitteln. Im Vergleich zu Ultraschall- oder Lidar-Messverfahren

ist das Mikrowellenradar **in Umgebungen mit Kondensation/Staub zuverlässiger**. Es kann die richtige Entfernung auch dann erfassen, wenn sich Wasser oder dichter Staub auf dem Sensor befindet.

Der LMDS200 kann in Szenarien wie horizontaler Entfernungsmessung, Parkplatzmanagementsystemen, Objektnäherungs- und Anwesenheitserkennung, intelligenten Abfallbehältermanagementsystemen, Roboter-Hindernisvermeidung, automatischer Steuerung, Kanalisation usw. eingesetzt werden.

LMDS200 kann **zwei Entfernungen messen**: das nächstgelegene Objekt und das nächste Objekt hinter dem nächstgelegenen.

LMDS200 unterstützt **eine Alarmfunktion**. Der Benutzer kann LMDS200 so einstellen, dass Daten in kurzen Intervallen hochgeladen werden, wenn die Entfernung außerhalb des konfigurierten Bereichs liegt.

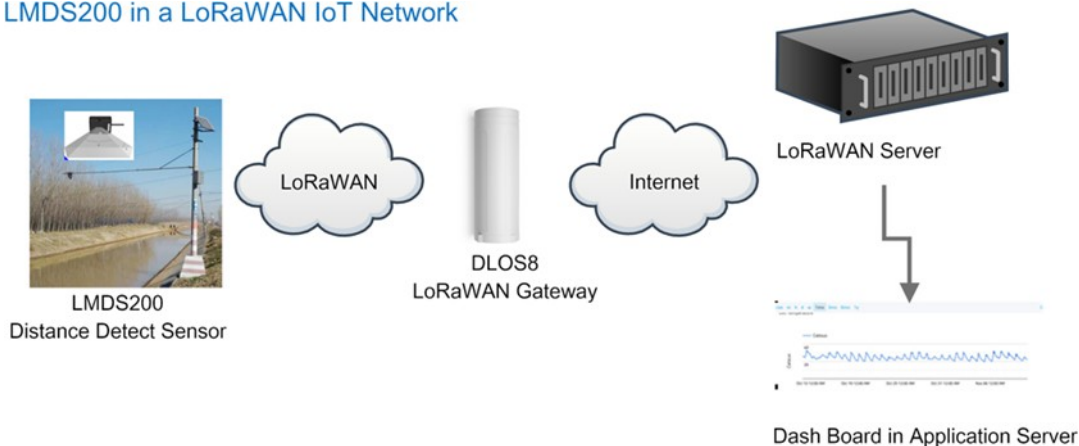
Die im LMDS200 verwendete LoRa-Funktechnologie ermöglicht es dem Gerät, Daten zu senden und bei niedrigen Datenraten extrem große Reichweiten zu erzielen. Sie bietet eine extrem weitreichende Spread-Spectrum-Kommunikation und eine hohe Störfestigkeit bei minimalem Stromverbrauch.

Der LMDS200 wird mit einem 8500-mAh-Li-SOCI2-Akku betrieben und ist für eine langfristige Nutzung von bis zu 5 Jahren ausgelegt.

Jedes LMDS200 ist mit einem Satz eindeutiger Schlüssel für LoRaWAN-Registrierungen vorinstalliert. Registrieren Sie diese Schlüssel auf dem lokalen LoRaWAN-Server, damit nach dem Einschalten automatisch eine Verbindung hergestellt wird.

*Die Batterielebensdauer hängt davon ab, wie oft Daten gesendet werden. Bitte beachten Sie [den Batterieanalysator](#).

LMDS200 in a LoRaWAN IoT Network



1.2 Funktionen

- LoRaWAN 1.0.3 Klasse A
- Extrem geringer Stromverbrauch
- Mikrowellenradar zur Entfernungsmessung
- Kurzes Uplink-Intervall für Entfernungsalarm
- Überwachung des Batteriestands
- Bänder: CN470/EU433/KR920/US915/EU868/AS923/AU915/IN865
- AT-Befehle zum Ändern von Parametern
- Uplink periodisch
- Downlink zum Ändern der Konfiguration
- 8500-mAh-Akku für langfristigen Einsatz
- Wandmontage möglich
- Verwendung im Außenbereich

1.3 Spezifikationen der Radarsonde

- Messmethode: FMCW
- Frequenz: 24,000 bis 24,500 GHz

- Messausgangsleistung: 6 dBm
- Messbereich: 0,5 bis 20 m
- Genauigkeit: $\pm 0,1$ m
- Auflösung: 0,01 m
- Horizontaler Winkel: 78°
- Vertikaler Winkel: 23°

1.4 Lager- und Betriebstemperatur

-20 °C bis +85 °C

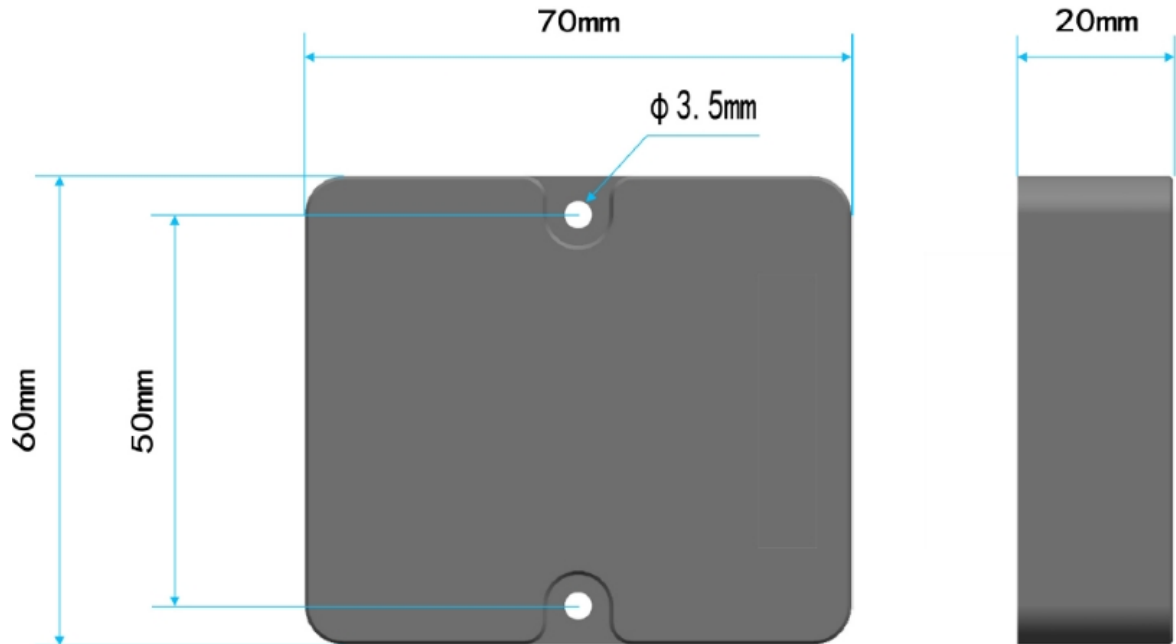
1.5 Anwendungen

- Horizontale Abstandsmessung
- Füllstandsmessung
- Parkplatzverwaltungssystem
- Objektnäherungs- und Anwesenheitserkennung
- Intelligentes Abfallbehälter-Managementsystem
- Hindernisvermeidung durch Roboter
- Automatische Steuerung
- Kanalisation
- Überwachung des Wasserstands am Boden

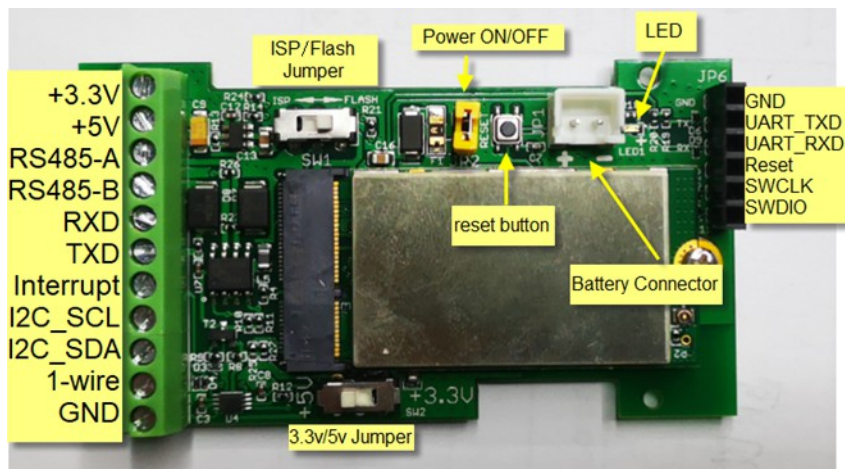
1.6 Installation

Die Messrichtung und der Messwinkel des Sensors sind wie folgt. Achten Sie bei der Installation des Sensors darauf, dass dieser auf das Objekt ausgerichtet ist.





1.7 Pin-Zuordnung und Einschalten



2. Betriebsmodus

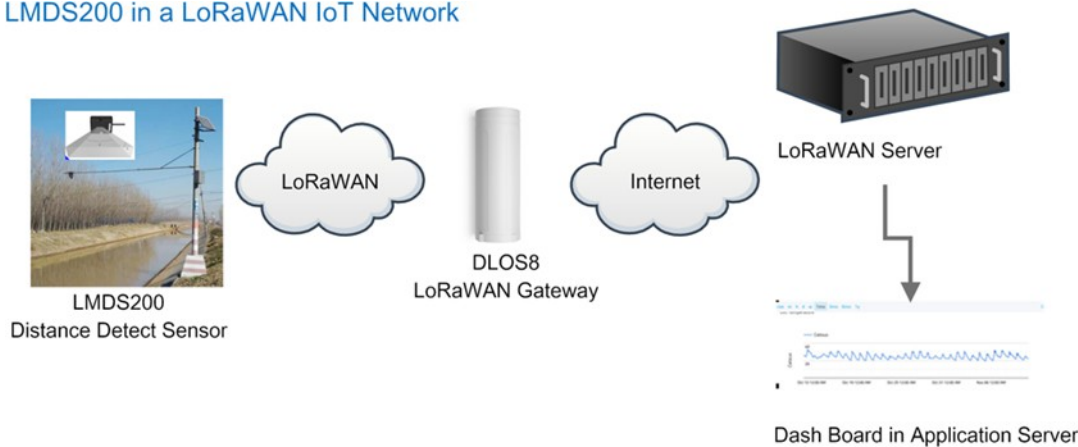
2.1 Funktionsweise

Jedes LMDS200 wird mit einem weltweit einzigartigen Satz von OTAA-Schlüsseln ausgeliefert. Um LMDS200 in einem LoRaWAN-Netzwerk zu verwenden, muss der Benutzer die OTAA-Schlüssel in den LoRaWAN-Netzwerkserver eingeben. So kann LMDS200 dem LoRaWAN-Netzwerk beitreten und mit der Übertragung von Sensordaten beginnen.

2.2 Beispiel für die Verwendung in einem LoRaWAN-Netzwerk

Im Folgenden finden Sie ein Beispiel für den Beitritt zum [TTN v3 LoRaWAN-Netzwerk](#). Nachfolgend sehen Sie die Netzwerkstruktur; in diesem Beispiel verwenden wir den [DLOS8](#) als LoRaWAN-Gateway.

LMDS200 in a LoRaWAN IoT Network

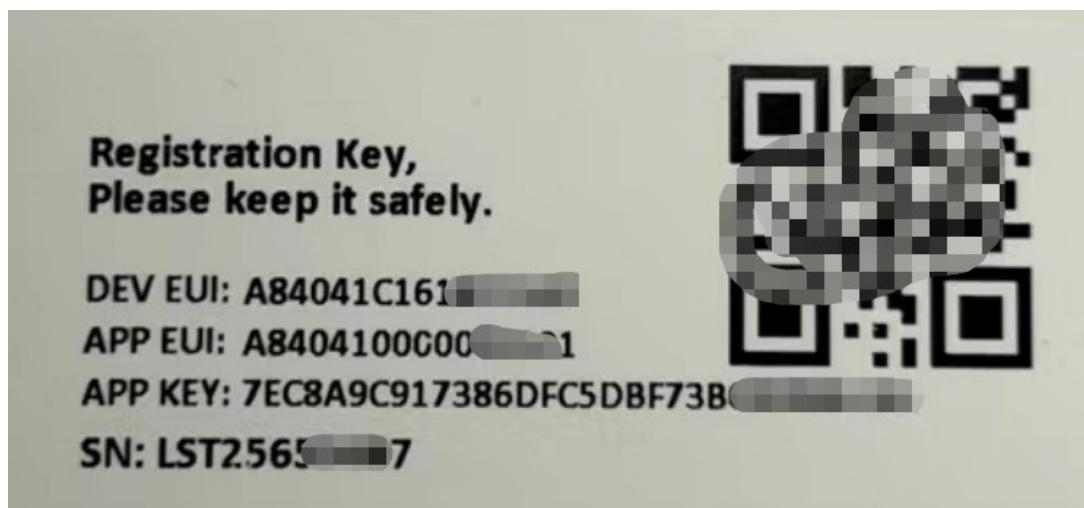


- In diesem Anwendungsfall wird das LMDS200 auf einem Fluss installiert, um den Wasserstand zu erfassen und die Pegelinformationen an den LoRaWAN-Server zu senden. Das LMDS200 sendet verschiedene Arten von Nachrichten an den LoRaWAN-Server. Weitere Informationen finden Sie unter „[Uplink-Nutzlast](#)“.

Angenommen, das LoRaWAN-Gateway DLOS8 ist bereits für die Verbindung mit dem [TTN V3-Netzwerk](#) konfiguriert. Wir müssen das LMDS200-Gerät in TTN V3 hinzufügen:

Schritt 1: Erstellen Sie ein Gerät in TTN mit den OTAA-Schlüsseln von LMDS200.

Jedes LMDS200 wird mit einem Aufkleber mit den Standard-Geräteschlüsseln geliefert, den der Benutzer in der Verpackung findet. Er sieht wie unten abgebildet aus.



Benutzer können diese Schlüssel im LoRaWAN-Server-Portal eingeben. Unten sehen Sie einen Screenshot von TTN V3:

Fügen Sie APP EUI in der Anwendung **hinzu**.

The screenshot shows the TTN V3 'Add application' form. The top navigation bar includes the TTN logo, 'THE THINGS STACK Community Edition', and menu items: Overview, Applications (highlighted with a red arrow), Gateways, and Orga. The form itself has the following fields:

- Owner***: A dropdown menu with 'davidhuang' selected.
- Application ID***: A text input field containing 'my-new-application'.
- Application name**: A text input field containing 'My new application'.
- Description**: A text area containing 'Description for my new application'.

Below the description field, there is a note: 'Optional application description; can also be used to save notes about the application'. At the bottom of the form is a blue button labeled 'Create application', which is pointed to by a red arrow.

**CCC**
ID: 123

4 End devices2 Collaborators2 API keysCreated 95 days ago

General information
Application ID123
Created atFeb 2, 2021 11:12:30
Last updated atApr 30, 2021 11:00:33

Live data
See all activity →
10:09:42 1231234234... Forward data message to Application Server
10:09:42 1231234234... Store upstream data message
10:09:42 1231234234... Forward uplink data message
10:09:42 1231234234... Receive uplink data message
10:09:42 1231234234... Successfully processed data message
10:09:42 1231234234... Drop data message

End devices (4)

Search by IDImport end devicesAdd end device

IDNameDevEUIJoinEUICreated

Register end device

[From The LoRaWAN Device Repository](#)[Manually](#)

1. Select the end device

Brand*
Dragino Technology Co.,...

Model*
Type to search...

Cannot find your exact end device?

LBT1
LDDS20
LDDS75
LDS01
LGT92
LHT65
LSE01
LSN50-V2

ice registration.

2. Enter registration data

Please choose an end device first to

Register end device

2. Enter registration data

Frequency plan ⓘ *

Select...

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 development), it can be filled with zeros.

Sie können das Gerät auch manuell erstellen.

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 ⓘ

APP KEY und DEV EUI hinzufügen

Register end device

From The LoRaWAN Device Repository Manually 1

Frequency plan 2

Select...

LoRaWAN version 3

MAC V1.0.3

Regional Parameters version

PHY V1.0.3 REV A

Show advanced activation, LoRaWAN class and cluster settings

DevEUI 4

... .. Generate 0/50 used

AppEUI 5

... .. Fill with zeros

AppKey 6

... .. Generate

End device ID 7

my-new-device

This value is automatically prefilled using the DevEUI

After registration

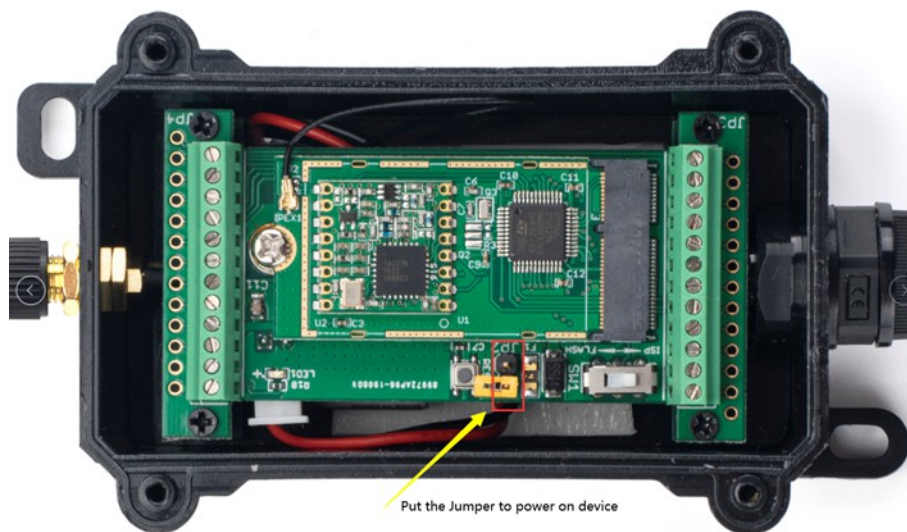
☒ View registered end device

☐ Register another end device of this type

8 Register end device

Schritt 2: LMDS200 einschalten

Setzen Sie einen Jumper auf JP2, um das Gerät einzuschalten. (Der Schalter muss sich in der Position FLASH befinden).



Setzen Sie den Jumper, um LMDS200 einzuschalten, und es wird automatisch mit dem TTN V3-Netzwerk verbunden. Nach erfolgreicher Verbindung beginnt es mit dem Hochladen der Sensordaten zu TTN V3, und der Benutzer kann diese im Panel sehen.

2.3 Uplink-Nutzlast

Es gibt zwei Arten von Uplink-Nutzlasten:

- Entfernungswert: Verwenden Sie FPORT=2
- Andere Steuerbefehle: Verwenden Sie andere FPORT-Felder.

Der Anwendungsserver sollte den korrekten Wert basierend auf den FPORT-Einstellungen analysieren.

2.3.1 Gerätestatus, FPORT=5

Gerätekonfigurationsstatus einbeziehen. Sobald LMDS200 dem Netzwerk beigetreten ist, sendet es diese Meldung an den Server. Benutzer können auch den Downlink-Befehl (0x26 01) verwenden, um LMDS200 aufzufordern, den Gerätestatus erneut zu senden.

Gerätestatus (FPORT=5)					
Größe (Byte)	1	2	1	1	2
Wert	Sensormodell	Firmware-Version	Frequenzband	Unterband	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
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	
15:32:20	Accept join-request	

- **Sensormodell:** Für LMDS200 ist dieser Wert 0x0C.
- **Firmware-Version:** 0x0100, bedeutet: Version v1.0.0
- **Frequenzband:**

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

- **Unterband:**
 - AU915 und US915: Wert 0x00 ~ 0x08
 - CN470: Wert 0x0B ~ 0x0C
 - Andere Bänder: Immer 0x00

- **Batterieinformationen:**

Überprüfen Sie die

Batteriespannung. Beispiel 1:

0x0B45 = 2885 mV Beispiel 2:

0x0B49 = 2889 mV

2.3.2 Sensor-Konfiguration, FPORT=4

LMDS200 sendet diesen Befehl nur, nachdem es den Downlink-Befehl (0x26 02) vom Server erhalten hat.

Sensor-Konfiguration FPORT=4				
Größe (Bytes)	3	1	4	1
Wert	TDC (Einheit: Sek.)	ATDC (Einheit: min)	Alarめinstellungen	Unterbrechungseinstellungen



2.3.3 Entfernung, Uplink FPORT=2

LMDS200 sendet diesen Uplink **nach** dem Gerätestatus, sobald die Verbindung zum LoRaWAN-Netzwerk erfolgreich hergestellt wurde. Außerdem wird LMDS200:

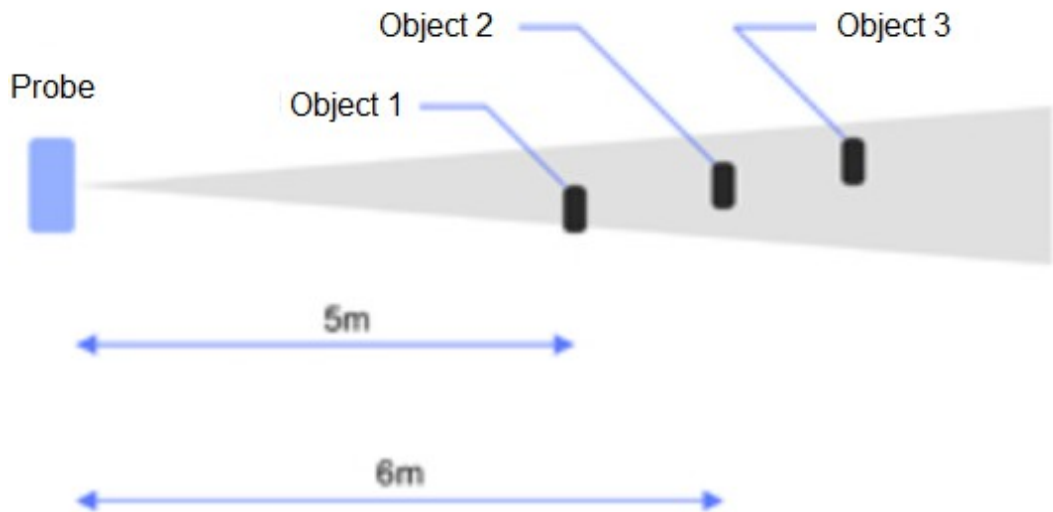
1. diesen Uplink regelmäßig alle 1 Stunde (TDC-Zeit) senden, wobei dieses Intervall [geändert werden kann](#).
2. Senden Sie diesen Uplink im Alarmmodus regelmäßig alle 1 Minute.
3. Senden Sie diesen Uplink, solange [ein Interrupt-](#)

[Ereignis](#) vorliegt. Die Uplink-Nutzlast beträgt

insgesamt 11 Byte.

Entfernungswert, FPORT=2				
Größe (Bytes)	2	2	2	1
Wert	BAT	Objekt1 Entfernung	Objekt2 Entfernung	Status & Alarm

Status- und Alarmfeld			
Größe (Bit)	6	1	1
Wert	DALARM-Zähler	Entfernungsalarm 0: Normalwert 1: Distanzalarm	Interrupt-Alarm 0: Kein Alarm 1: Externer Interrupt-Alarm



Objekt1 Entfernung:

Entfernung zwischen Sensorsonde und dem ersten Objekt. (Einheit: cm)

Wenn die Daten, die Sie aus dem Register erhalten, beispielsweise 0x00 0x73 lauten, beträgt der Abstand zwischen dem Sensor und dem gemessenen Objekt **0073(H) = 115 (D) = 115 cm**.

Hinweis: Es gibt zwei Sonderwerte für die Entfernung von Objekt 1:

- **0x0001:** Sonde nicht erkannt
- **0x0002:** Ungültiger Messwert (überschreitet den gültigen Bereich der Sonde)

Entfernung zu Objekt 2:

Abstand zwischen der Sensorsonde und dem zweiten Objekt. (Einheit: cm)

DALARM-Zähler: Alarmzähler.

↑ 14:30:18	Forward uplink data message	Payload: { Bat: 3.328, DALARM_count: 19, Distance_alarm: 1, Interrupt_alarm: 0, dis1: 115, dis2: 164 }	80 00 00 73 00 A4 4E	FP
ⓘ 14:29:18	Fail to send webhook	Error:undefined:undefined		
ⓘ 14:29:18	Fail to send webhook	Error:undefined:undefined		
ⓘ 14:29:18	Fail to send webhook	Error:undefined:undefined		
ⓘ 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 }	80 00 00 73 00 A4 4A	FP
ⓘ 14:28:18	Fail to send webhook	Error:undefined:undefined		
ⓘ 14:28:18	Fail to send webhook	Error:undefined:undefined		
ⓘ 14:28:18	Fail to send webhook	Error:undefined:undefined		
ⓘ 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 }	80 02 00 73 00 A4 46	FP

2.3.4 Decoder in TTN V3

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

Uplink Downlink

These payload formatters are executed on uplink messages from this end device and take precedence over application level payload formatters.

Formatter type

☐ Use application payload formatter ☐ None ☒ Javascript ☐ 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

Bitte überprüfen Sie den Decoder unter diesem Link: <https://github.com/dragino/dragino-end-node-decoder>

2.4 Daten auf Datacake anzeigen

Die Datacake-IoT-Plattform bietet eine benutzerfreundliche Oberfläche zur Anzeige der Sensordaten. Sobald wir Sensordaten in TTN V3 haben, können wir Datacake verwenden, um eine Verbindung zu TTN V3 herzustellen und die Daten in Datacake anzuzeigen. Nachfolgend sind die Schritte aufgeführt:

Schritt 1: TTNv3 mit [DATACAKE](#) verbinden

Schritt 2: LMDS200 in Datacake konfigurieren

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

LDS03A

Next

Network Server

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

1

☒		The Things Stack V3 TTN V3 / Things Industries	Uplinks	Downlinks
☐		The Things Network V2 The old Things Network	Uplinks	Downlinks
☐		Helium	Uplinks	Downlinks
☐		LORIoT	Uplinks	Downlinks
☐		Kerlink Wanasy	Uplinks	

Showing 1 to 5 of 8 results

Previous Next

2

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	

Back

Next

1 2 3

 DATA CAKE

Fleet > LDS03A

LDS03A

Serial Number
9955663322441140

Last update
Never

Dashboard

History

Downlinks

Configuration

Debug

Rules

Permissions

General Configuration

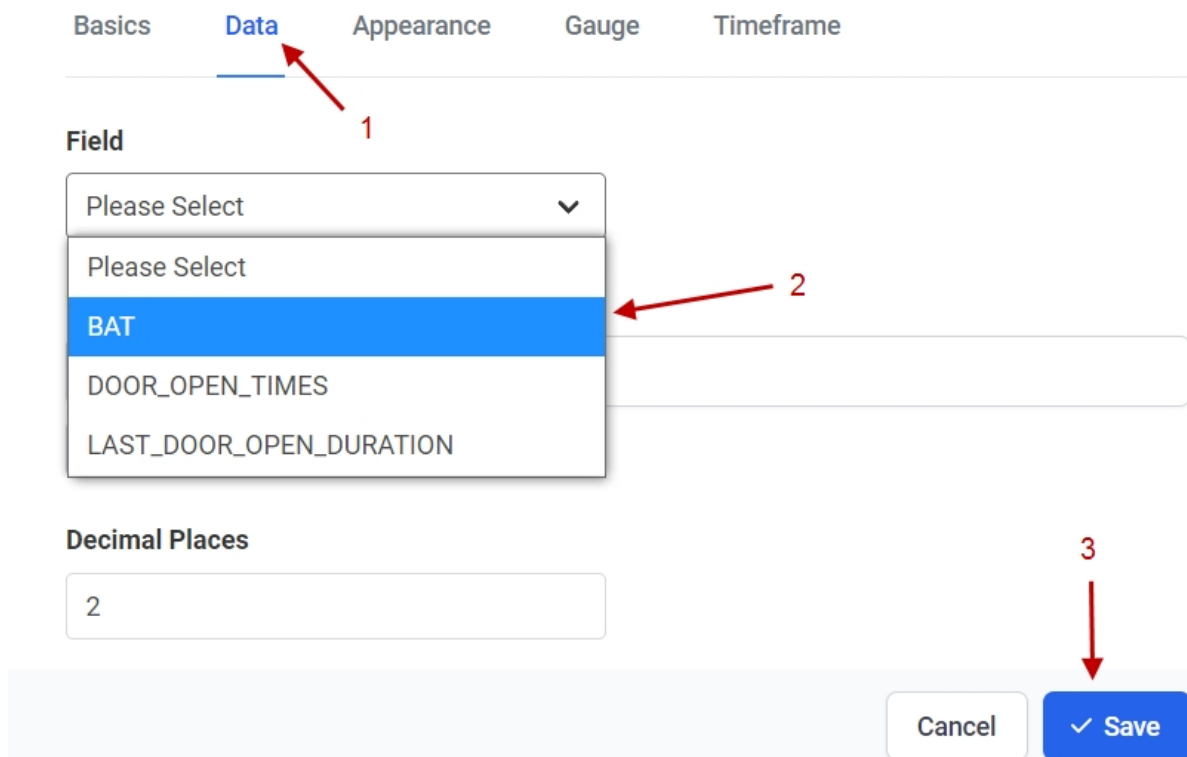
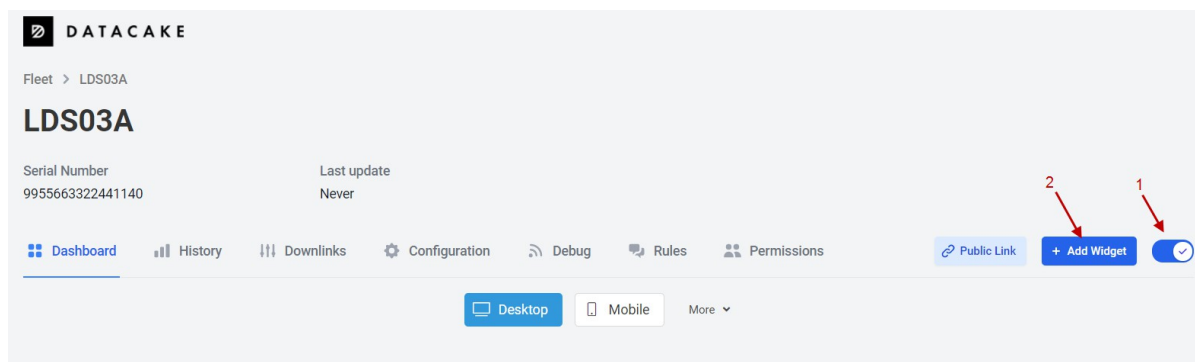
Device Name

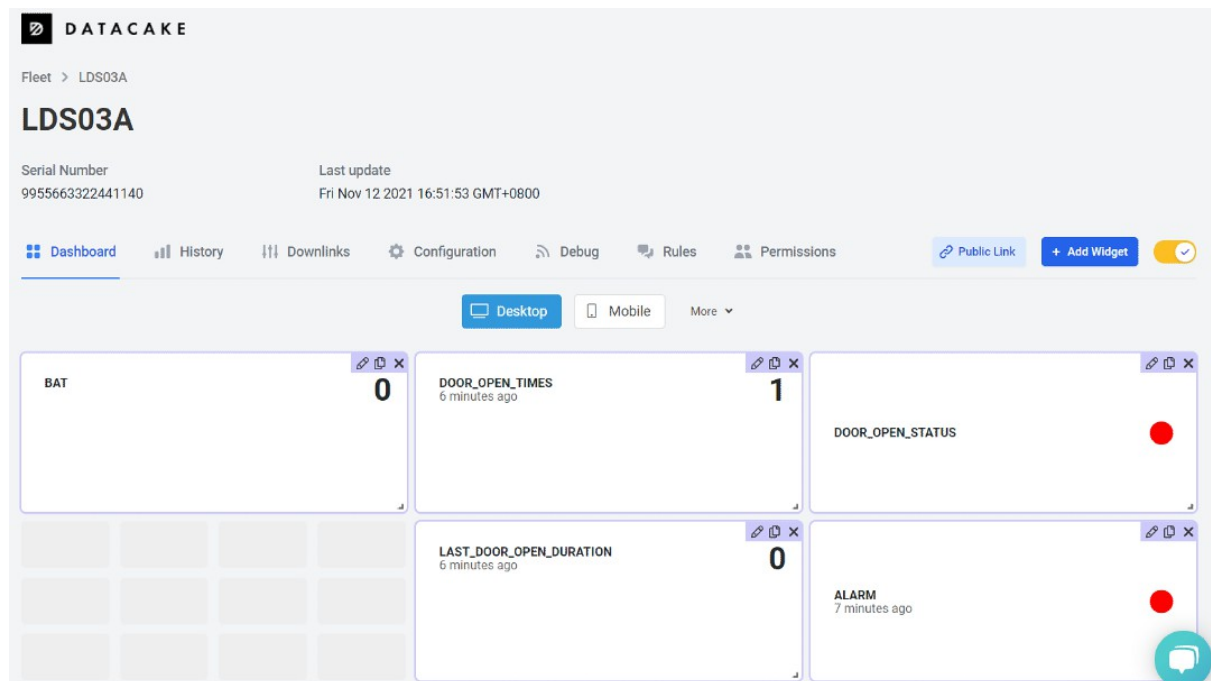
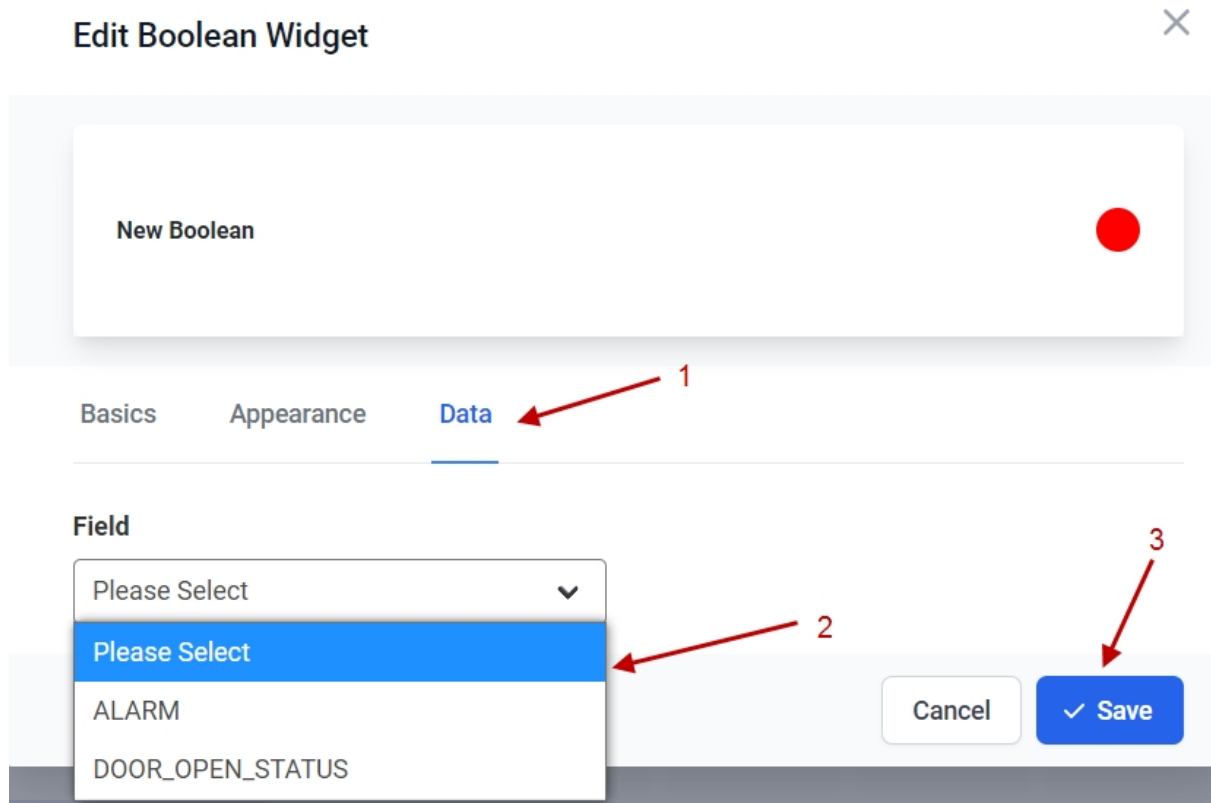
LDS03A

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= function decode(c_json){
2   var aa=bytes[0];
3   var bb=bytes[1];
4   var cc=bytes[2];
5   var dd=bytes[3];
6   var ee=bytes[4];
7   var ff=bytes[5];
8   var gg=bytes[6];
9   var hh=bytes[7];
10  var ii=bytes[8];
11  var jj=bytes[9];
12  var kk=bytes[10];
13  var ll=bytes[11];
14  var mm=bytes[12];
15  var nn=bytes[13];
16  var oo=bytes[14];
17  var pp=bytes[15];
18  var qq=bytes[16];
19  var rr=bytes[17];
20  var ss=bytes[18];
21  var tt=bytes[19];
22  var uu=bytes[20];
23  var vv=bytes[21];
24  var ww=bytes[22];
25  var xx=bytes[23];
26  var yy=bytes[24];
27  var zz=bytes[25];
28  var aa=bytes[26];
29  var bb=bytes[27];
30  var cc=bytes[28];
31  var dd=bytes[29];
32  var ee=bytes[30];
33  var ff=bytes[31];
34  var gg=bytes[32];
35  var hh=bytes[33];
36  var ii=bytes[34];
37  var jj=bytes[35];
38  var kk=bytes[36];
39  var ll=bytes[37];
40  var mm=bytes[38];
41  var nn=bytes[39];
42  var oo=bytes[40];
43  var pp=bytes[41];
44  var qq=bytes[42];
45  var rr=bytes[43];
46  var ss=bytes[44];
47  var tt=bytes[45];
48  var uu=bytes[46];
49  var vv=bytes[47];
50  var ww=bytes[48];
51  var xx=bytes[49];
52  var yy=bytes[50];
53  var zz=bytes[51];
54  var aa=bytes[52];
55  var bb=bytes[53];
56  var cc=bytes[54];
57  var dd=bytes[55];
58  var ee=bytes[56];
59  var ff=bytes[57];
60  var gg=bytes[58];
61  var hh=bytes[59];
62  var ii=bytes[60];
63  var jj=bytes[61];
64  var kk=bytes[62];
65  var ll=bytes[63];
66  var mm=bytes[64];
67  var nn=bytes[65];
68  var oo=bytes[66];
69  var pp=bytes[67];
70  var qq=bytes[68];
71  var rr=bytes[69];
72  var ss=bytes[70];
73  var tt=bytes[71];
74  var uu=bytes[72];
75  var vv=bytes[73];
76  var ww=bytes[74];
77  var xx=bytes[75];
78  var yy=bytes[76];
79  var zz=bytes[77];
80  var aa=bytes[78];
81  var bb=bytes[79];
82  var cc=bytes[80];
83  var dd=bytes[81];
84  var ee=bytes[82];
85  var ff=bytes[83];
86  var gg=bytes[84];
87  var hh=bytes[85];
88  var ii=bytes[86];
89  var jj=bytes[87];
90  var kk=bytes[88];
91  var ll=bytes[89];
92  var mm=bytes[90];
93  var nn=bytes[91];
94  var oo=bytes[92];
95  var pp=bytes[93];
96  var qq=bytes[94];
97  var rr=bytes[95];
98  var ss=bytes[96];
99  var tt=bytes[97];
100 var uu=bytes[98];
101 var vv=bytes[99];
102 var ww=bytes[100];
103 var xx=bytes[101];
104 var yy=bytes[102];
105 var zz=bytes[103];
106 var aa=bytes[104];
107 var bb=bytes[105];
108 var cc=bytes[106];
109 var dd=bytes[107];
110 var ee=bytes[108];
111 var ff=bytes[109];
112 var gg=bytes[110];
113 var hh=bytes[111];
114 var ii=bytes[112];
115 var jj=bytes[113];
116 var kk=bytes[114];
117 var ll=bytes[115];
118 var mm=bytes[116];
119 var nn=bytes[117];
120 var oo=bytes[118];
121 var pp=bytes[119];
122 var qq=bytes[120];
123 var rr=bytes[121];
124 var ss=bytes[122];
125 var tt=bytes[123];
126 var uu=bytes[124];
127 var vv=bytes[125];
128 var ww=bytes[126];
129 var xx=bytes[127];
130 var yy=bytes[128];
131 var zz=bytes[129];
132 var aa=bytes[130];
133 var bb=bytes[131];
134 var cc=bytes[132];
135 var dd=bytes[133];
136 var ee=bytes[134];
137 var ff=bytes[135];
138 var gg=bytes[136];
139 var hh=bytes[137];
140 var ii=bytes[138];
141 var jj=bytes[139];
142 var kk=bytes[140];
143 var ll=bytes[141];
144 var mm=bytes[142];
145 var nn=bytes[143];
146 var oo=bytes[144];
147 var pp=bytes[145];
148 var qq=bytes[146];
149 var rr=bytes[147];
150 var ss=bytes[148];
151 var tt=bytes[149];
152 var uu=bytes[150];
153 var vv=bytes[151];
154 var ww=bytes[152];
155 var xx=bytes[153];
156 var yy=bytes[154];
157 var zz=bytes[155];
158 var aa=bytes[156];
159 var bb=bytes[157];
160 var cc=bytes[158];
161 var dd=bytes[159];
162 var ee=bytes[160];
163 var ff=bytes[161];
164 var gg=bytes[162];
165 var hh=bytes[163];
166 var ii=bytes[164];
167 var jj=bytes[165];
168 var kk=bytes[166];
169 var ll=bytes[167];
170 var mm=bytes[168];
171 var nn=bytes[169];
172 var oo=bytes[170];
173 var pp=bytes[171];
174 var qq=bytes[172];
175 var rr=bytes[173];
176 var ss=bytes[174];
177 var tt=bytes[175];
178 var uu=bytes[176];
179 var vv=bytes[177];
180 var ww=bytes[178];
181 var xx=bytes[179];
182 var yy=bytes[180];
183 var zz=bytes[181];
184 var aa=bytes[182];
185 var bb=bytes[183];
186 var cc=bytes[184];
187 var dd=bytes[185];
188 var ee=bytes[186];
189 var ff=bytes[187];
190 var gg=bytes[188];
191 var hh=bytes[189];
192 var ii=bytes[190];
193 var jj=bytes[191];
194 var kk=bytes[192];
195 var ll=bytes[193];
196 var mm=bytes[194];
197 var nn=bytes[195];
198 var oo=bytes[196];
199 var pp=bytes[197];
200 var qq=bytes[198];
201 var rr=bytes[199];
202 var ss=bytes[200];
203 var tt=bytes[201];
204 var uu=bytes[202];
205 var vv=bytes[203];
206 var ww=bytes[204];
207 var xx=bytes[205];
208 var yy=bytes[206];
209 var zz=bytes[207];
210 var aa=bytes[208];
211 var bb=bytes[209];
212 var cc=bytes[210];
213 var dd=bytes[211];
214 var ee=bytes[212];
215 var ff=bytes[213];
216 var gg=bytes[214];
217 var hh=bytes[215];
218 var ii=bytes[216];
219 var jj=bytes[217];
220 var kk=bytes[218];
221 var ll=bytes[219];
222 var mm=bytes[220];
223 var nn=bytes[221];
224 var oo=bytes[222];
225 var pp=bytes[223];
226 var qq=bytes[224];
227 var rr=bytes[225];
228 var ss=bytes[226];
229 var tt=bytes[227];
230 var uu=bytes[228];
231 var vv=bytes[229];
232 var ww=bytes[230];
233 var xx=bytes[231];
234 var yy=bytes[232];
235 var zz=bytes[233];
236 var aa=bytes[234];
237 var bb=bytes[235];
238 var cc=bytes[236];
239 var dd=bytes[237];
240 var ee=bytes[238];
241 var ff=bytes[239];
242 var gg=bytes[240];
243 var hh=bytes[241];
244 var ii=bytes[242];
245 var jj=bytes[243];
246 var kk=bytes[244];
247 var ll=bytes[245];
248 var mm=bytes[246];
249 var nn=bytes[247];
250 var oo=bytes[248];
251 var pp=bytes[249];
252 var qq=bytes[250];
253 var rr=bytes[251];
254 var ss=bytes[252];
255 var tt=bytes[253];
256 var uu=bytes[254];
257 var vv=bytes[255];
258 var ww=bytes[256];
259 var xx=bytes[257];
260 var yy=bytes[258];
261 var zz=bytes[259];
262 var aa=bytes[260];
263 var bb=bytes[261];
264 var cc=bytes[262];
265 var dd=bytes[263];
266 var ee=bytes[264];
267 var ff=bytes[265];
268 var gg=bytes[266];
269 var hh=bytes[267];
270 var ii=bytes[268];
271 var jj=bytes[269];
272 var kk=bytes[270];
273 var ll=bytes[271];
274 var mm=bytes[272];
275 var nn=bytes[273];
276 var oo=bytes[274];
277 var pp=bytes[275];
278 var qq=bytes[276];
279 var rr=bytes[277];
280 var ss=bytes[278];
281 var tt=bytes[279];
282 var uu=bytes[280];
283 var vv=bytes[281];
284 var ww=bytes[282];
285 var xx=bytes[283];
286 var yy=bytes[284];
287 var zz=bytes[285];
288 var aa=bytes[286];
289 var bb=bytes[287];
290 var cc=bytes[288];
291 var dd=bytes[289];
292 var ee=bytes[290];
293 var ff=bytes[291];
294 var gg=bytes[292];
295 var hh=bytes[293];
296 var ii=bytes[294];
297 var jj=bytes[295];
298 var kk=bytes[296];
299 var ll=bytes[297];
300 var mm=bytes[298];
301 var nn=bytes[299];
302 var oo=bytes[300];
303 var pp=bytes[301];
304 var qq=bytes[302];
305 var rr=bytes[303];
306 var ss=bytes[304];
307 var tt=bytes[305];
308 var uu=bytes[306];
309 var vv=bytes[307];
310 var ww=bytes[308];
311 var xx=bytes[309];
312 var yy=bytes[310];
313 var zz=bytes[311];
314 var aa=bytes[312];
315 var bb=bytes[313];
316 var cc=bytes[314];
317 var dd=bytes[315];
318 var ee=bytes[316];
319 var ff=bytes[317];
320 var gg=bytes[318];
321 var hh=bytes[319];
322 var ii=bytes[320];
323 var jj=bytes[321];
324 var kk=bytes[322];
325 var ll=bytes[323];
326 var mm=bytes[324];
327 var nn=bytes[325];
328 var oo=bytes[326];
329 var pp=bytes[327];
330 var qq=bytes[328];
331 var rr=bytes[329];
332 var ss=bytes[330];
333 var tt=bytes[331];
334 var uu=bytes[332];
335 var vv=bytes[333];
336 var ww=bytes[334];
337 var xx=bytes[335];
338 var yy=bytes[336];
339 var zz=bytes[337];
340 var aa=bytes[338];
341 var bb=bytes[339];
342 var cc=bytes[340];
343 var dd=bytes[341];
344 var ee=bytes[342];
345 var ff=bytes[343];
346 var gg=bytes[344];
347 var hh=bytes[345];
348 var ii=bytes[346];
349 var jj=bytes[347];
350 var kk=bytes[348];
351 var ll=bytes[349];
352 var mm=bytes[350];
353 var nn=bytes[351];
354 var oo=bytes[352];
355 var pp=bytes[353];
356 var qq=bytes[354];
357 var rr=bytes[355];
358 var ss=bytes[356];
359 var tt=bytes[357];
360 var uu=bytes[358];
361 var vv=bytes[359];
362 var ww=bytes[360];
363 var xx=bytes[361];
364 var yy=bytes[362];
365 var zz=bytes[363];
366 var aa=bytes[364];
367 var bb=bytes[365];
368 var cc=bytes[366];
369 var dd=bytes[367];
370 var ee=bytes[368];
371 var ff=bytes[369];
372 var gg=bytes[370];
373 var hh=bytes[371];
374 var ii=bytes[372];
375 var jj=bytes[373];
376 var kk=bytes[374];
377 var ll=bytes[375];
378 var mm=bytes[376];
379 var nn=bytes[377];
380 var oo=bytes[378];
381 var pp=bytes[379];
382 var qq=bytes[380];
383 var rr=bytes[381];
384 var ss=bytes[382];
385 var tt=bytes[383];
386 var uu=bytes[384];
387 var vv=bytes[385];
388 var ww=bytes[386];
389 var xx=bytes[387];
390 var yy=bytes[388];
391 var zz=bytes[389];
392 var aa=bytes[390];
393 var bb=bytes[391];
394 var cc=bytes[392];
395 var dd=bytes[393];
396 var ee=bytes[394];
397 var ff=bytes[395];
398 var gg=bytes[396];
399 var hh=bytes[397];
400 var ii=bytes[398];
401 var jj=bytes[399];
402 var kk=bytes[400];
403 var ll=bytes[401];
404 var mm=bytes[402];
405 var nn=bytes[403];
406 var oo=bytes[404];
407 var pp=bytes[405];
408 var qq=bytes[406];
409 var rr=bytes[407];
410 var ss=bytes[408];
411 var tt=bytes[409];
412 var uu=bytes[410];
413 var vv=bytes[411];
414 var ww=bytes[412];
415 var xx=bytes[413];
416 var yy=bytes[414];
417 var zz=bytes[415];
418 var aa=bytes[416];
419 var bb=bytes[417];
420 var cc=bytes[418];
421 var dd=bytes[419];
422 var ee=bytes[420];
423 var ff=bytes[421];
424 var gg=bytes[422];
425 var hh=bytes[423];
426 var ii=bytes[424];
427 var jj=bytes[425];
428 var kk=bytes[426];
429 var ll=bytes[427];
430 var mm=bytes[428];
431 var nn=bytes[429];
432 var oo=bytes[430];
433 var pp=bytes[431];
434 var qq=bytes[432];
435 var rr=bytes[433];
436 var ss=bytes[434];
437 var tt=bytes[435];
438 var uu=bytes[436];
439 var vv=bytes[437];
440 var ww=bytes[438];
441 var xx=bytes[439];
442 var yy=bytes[440];
443 var zz=bytes[441];
444 var aa=bytes[442];
445 var bb=bytes[443];
446 var cc=bytes[444];
447 var dd=bytes[445];
448 var ee=bytes[446];
449 var ff=bytes[447];
450 var gg=bytes[448];
451 var hh=bytes[449];
452 var ii=bytes[450];
453 var jj=bytes[451];
454 var kk=bytes[452];
455 var ll=bytes[453];
456 var mm=bytes[454];
457 var nn=bytes[455];
458 var oo=bytes[456];
459 var pp=bytes[457];
460 var qq=bytes[458];
461 var rr=bytes[459];
462 var ss=bytes[460];
463 var tt=bytes[461];
464 var uu=bytes[462];
465 var vv=bytes[463];
466 var ww=bytes[464];
467 var xx=bytes[465];
468 var yy=bytes[466];
469 var zz=bytes[467];
470 var aa=bytes[468];
471 var bb=bytes[469];
472 var cc=bytes[470];
473 var dd=bytes[471];
474 var ee=bytes[472];
475 var ff=bytes[473];
476 var gg=bytes[474];
477 var hh=bytes[475];
478 var ii=bytes[476];
479 var jj=bytes[477];
480 var kk=bytes[478];
481 var ll=bytes[479];
482 var mm=bytes[480];
483 var nn=bytes[481];
484 var oo=bytes[482];
485 var pp=bytes[483];
486 var qq=bytes[484];
487 var rr=bytes[485];
488 var ss=bytes[486];
489 var tt=bytes[487];
490 var uu=bytes[488];
491 var vv=bytes[489];
492 var ww=bytes[490];
493 var xx=bytes[491];
494 var yy=bytes[492];
495 var zz=bytes[493];
496 var aa=bytes[494];
497 var bb=bytes[495];
498 var cc=bytes[496];
499 var dd=bytes[497];
500 var ee=bytes[498];
501 var ff=bytes[499];
502 var gg=bytes[500];
503 var hh=bytes[501];
504 var ii=bytes[502];
505 var jj=bytes[503];
506 var kk=bytes[504];
507 var ll=bytes[505];
508 var mm=bytes[506];
509 var nn=bytes[507];
510 var oo=bytes[508];
511 var pp=bytes[509];
512 var qq=bytes[510];
513 var rr=bytes[511];
514 var ss=bytes[512];
515 var tt=bytes[513];
516 var uu=bytes[514];
517 var vv=bytes[515];
518 var ww=bytes[516];
519 var xx=bytes[517];
520 var yy=bytes[518];
521 var zz=bytes[519];
522 var aa=bytes[520];
523 var bb=bytes[521];
524 var cc=bytes[522];
525 var dd=bytes[523];
526 var ee=bytes[524];
527 var ff=bytes[525];
528 var gg=bytes[526];
529 var hh=bytes[527];
530 var ii=bytes[528];
531 var jj=bytes[529];
532 var kk=bytes[530];
533 var ll=bytes[531];
534 var mm=bytes[532];
535 var nn=bytes[533];
536 var oo=bytes[534];
537 var pp=bytes[535];
538 var qq=bytes[536];
539 var rr=bytes[537];
540 var ss=bytes[538];
541 var tt=bytes[539];
542 var uu=bytes[540];
543 var vv=bytes[541];
544 var ww=bytes[542];
545 var xx=bytes[543];
546 var yy=bytes[544];
547 var zz=bytes[545];
548 var aa=bytes[546];
549 var bb=bytes[547];
550 var cc=bytes[548];
551 var dd=bytes[549];
552 var ee=bytes[550];
553 var ff=bytes[551];
554 var gg=bytes[552];
555 var hh=bytes[553];
556 var ii=bytes[554];
557 var jj=bytes[555];
558 var kk=bytes[556];
559 var ll=bytes[557];
560 var mm=bytes[558];
561 var nn=bytes[559];
562 var oo=bytes[560];
563 var pp=bytes[561];
564 var qq=bytes[562];
565 var rr=bytes[563];
566 var ss=bytes[564];
567 var tt=bytes[565];
568 var uu=bytes[566];
569 var vv=bytes[567];
570 var ww=bytes[568];
571 var xx=bytes[569];
572 var yy=bytes[570];
573 var zz=bytes[571];
574 var aa=bytes[572];
575 var bb=bytes[573];
576 var cc=bytes[574];
577 var dd=bytes[575];
578 var ee=bytes[576];
579 var ff=bytes[577];
580 var gg=bytes[578];
581 var hh=bytes[579];
582 var ii=bytes[580];
583 var jj=bytes[581];
584 var kk=bytes[582];
585 var ll=bytes[583];
586 var mm=bytes[584];
587 var nn=bytes[585];
588 var oo=bytes[586];
589 var pp=bytes[587];
590 var qq=bytes[588];
591 var rr=bytes[589];
592 var ss=bytes[590];
593 var tt=bytes[591];
594 var uu=bytes[592];
595 var vv=bytes[593];
596 var ww=bytes[594];
597 var xx=bytes[595];
598 var yy=bytes[596];
599 var zz=bytes[597];
600 var aa=bytes[598];
601 var bb=bytes[599];
602 var cc=bytes[600];
603 var dd=bytes[601];
604 var ee=bytes[602];
605 var ff=bytes[603];
606 var gg=bytes[604];
607 var hh=bytes[605];
608 var ii=bytes[606];
609 var jj=bytes[607];
610 var kk=bytes[608];
611 var ll=bytes[609];
612 var mm=bytes[610];
613 var nn=bytes[611];
614 var oo=bytes[612];
615 var pp=bytes[613];
616 var qq=bytes[614];
617 var rr=bytes[615];
618 var ss=bytes[616];
619 var tt=bytes[617];
620 var uu=bytes[618];
621 var vv=bytes[619];
622 var ww=bytes[620];
623 var xx=bytes[621];
624 var yy=bytes[622];
625 var zz=bytes[623];
626 var aa=bytes[624];
627 var bb=bytes[625];
628 var cc=bytes[626];
629 var dd=bytes[627];
630 var ee=bytes[628];
631 var ff=bytes[629];
632 var gg=bytes[630];
633 var hh=bytes[631];
634 var ii=bytes[632];
635 var jj=bytes[633];
636 var kk=bytes[634];
637 var ll=bytes[635];
638 var mm=bytes[636];
639 var nn=bytes[637];
640 var oo=bytes[638];
641 var pp=bytes[639];
642 var qq=bytes[640];
643 var rr=bytes[641];
644 var ss=bytes[642];
645 var tt=bytes[643];
646 var uu=bytes[644];
647 var vv=bytes[645];
648 var ww=bytes[646];
649 var xx=bytes[647];
650 var yy=bytes[648];
651 var zz=bytes[649];
652 var aa=bytes[650];
653 var bb=bytes[651];
654 var cc=bytes[652];
655 var dd=bytes[653];
656 var ee=bytes[654];
657 var ff=bytes[655];
658 var gg=bytes[656];
659 var hh=bytes[657];
660 var ii=bytes[658];
661 var jj=bytes[659];
662 var kk=bytes[660];
663 var ll=bytes[661];
664 var mm=bytes[662];
665 var nn=bytes[663];
666 var oo=bytes[664];
667 var pp=bytes[665];
668 var qq=bytes[666];
669 var rr=bytes[667];
670 var ss=bytes[668];
671 var tt=bytes[669];
672 var uu=bytes[670];
673 var vv=bytes[671];
674 var ww=bytes[672];
675 var xx=bytes[673];
676 var yy=bytes[674];
677 var zz=bytes[675];
678 var aa=bytes[676];
679 var bb=bytes[677];
680 var cc=bytes[678];
681 var dd=bytes[679];
682 var ee=bytes[680];
683 var ff=bytes[681];
684 var gg=bytes[682];
685 var hh=bytes[683];
686 var ii=bytes[684];
687 var jj=bytes[685];
688 var kk=bytes[686];
689 var ll=bytes[687];
690 var mm=bytes[688];
691 var nn=bytes[689];
692 var oo=bytes[690];
693 var pp=bytes[691];
694 var qq=bytes[692];
695 var rr=bytes[693];
696 var ss=bytes[694];
697 var tt=bytes[695];
698 var uu=bytes[696];
699 var vv=bytes[697];
700 var ww=bytes[698];
701 var xx=bytes[699];
702 var yy=bytes[700];
703 var zz=bytes[701];
704 var aa=bytes[702];
705 var bb=bytes[703];
706 var cc=bytes[704];
707 var dd=bytes[705];
708 var ee=bytes[706];
709 var ff=bytes[707];
710 var gg=bytes[708];
711 var hh=bytes[709];
712 var ii=bytes[710];
713 var jj=bytes[711];
714 var kk=bytes[712];
715 var ll=bytes[713];
716 var mm=bytes[714];
717 var nn=bytes[715];
718 var oo=bytes[716];
719 var pp=bytes[717];
720 var qq=bytes[718];
721 var rr=bytes[719];
722 var ss=bytes[720];
723 var tt=bytes[721];
724 var uu=bytes[722];
725 var vv=bytes[723];
726 var ww=bytes[724];
727 var xx=bytes[725];
728 var yy=bytes[726];
729 var zz=bytes[727];
730 var aa=bytes[728];
731 var bb=bytes[729];
732 var cc=bytes[730];
733 var dd=bytes[731];
734 var ee=bytes[732];
735 var ff=bytes[733];
736 var gg=bytes[734];
737 var hh=bytes[735];
738 var ii=bytes[736];
739 var jj=bytes[737];
740 var kk=bytes[738];
741 var ll=bytes[739];
742 var mm=bytes[740];
743 var nn=bytes[741];
744 var oo=bytes[742];
745 var pp=bytes[743];
746 var qq=bytes[744];
747 var rr=bytes[745];
748 var ss=bytes[746];
749 var tt=bytes[747];
750 var uu=bytes[748];
751 var vv=bytes[749];
752 var ww=bytes[750];
753 var xx=bytes[751];
754 var yy=bytes[752];
755 var zz=bytes[753];
756 var aa=bytes[754];
757 var bb=bytes[755];
758 var cc=bytes[756];
759 var dd=bytes[757];
760 var ee=bytes[758];
761 var ff=bytes[759];
762 var gg=bytes[760];
763 var hh=bytes[761];
764 var ii=bytes[762];
765 var jj=bytes[763];
766 var kk=bytes[764];
767 var ll=bytes[765];
768 var mm=bytes[766];
769 var nn=bytes[767];
770 var oo=bytes[768];
771 var pp=bytes[769];
772 var qq=bytes[770];
773 var rr=bytes[771];
774 var ss=bytes[772];
775 var tt=bytes[773];
776 var uu=bytes[774];
777 var vv=bytes[775];
778 var ww=bytes[776];
779 var xx=bytes[777];
780 var yy=bytes[778];
781 var zz=bytes[779];
782 var aa=bytes[780];
783 var bb=bytes[781];
784 var cc=bytes[782];
785 var dd=bytes[783];
786 var ee=bytes[784];
787 var ff=bytes[785];
788 var gg=bytes[786];
789 var hh=bytes[787];
790 var ii=bytes[788];
791 var jj=bytes[789];
792 var kk=bytes[790];
793 var ll=bytes[791];
794 var mm=bytes[792];
795 var nn=bytes[793];
796 var oo=bytes[794];
797 var pp=bytes[795];
798 var qq=bytes[796];
799 var rr=bytes[797];
800 var ss=bytes[798];
801 var tt=bytes[799];
802 var uu=bytes[800];
803 var vv=bytes[801];
804 var ww=bytes[802];
805 var xx=bytes[803];
806 var yy=bytes[804];
807 var zz=bytes[805];
808 var aa=bytes[806];
809 var bb=bytes[807];
810 var cc=bytes[808];
811 var dd=bytes[809];
812 var ee=bytes[810];
813 var ff=bytes[811];
814 var gg=bytes[812];
815 var hh=bytes[813];
816 var ii=bytes[814];
817 var jj=bytes[815];
818 var kk=bytes[816];
819 var ll=bytes[817];
820 var mm=bytes[818];
821 var nn=bytes[819];
822 var oo=bytes[820];
823 var pp=bytes[821];
824 var qq=bytes[822];
825 var rr=bytes[823];
826 var ss=bytes[824];
827 var tt=bytes[825];
828 var uu=bytes[826];
829 var vv=bytes[827];
830 var ww=bytes[828];
831 var xx=bytes[829];
832 var yy=bytes[830];
833 var zz=bytes[831];
834 var aa=bytes[832];
835 var bb=bytes[833];
836 var cc=bytes[834];
837 var dd=bytes[835];
838 var ee=bytes[836];
839 var ff=bytes[837];
840 var gg=bytes[838];
841 var hh=bytes[839];
842 var ii=bytes[840];
843 var jj=bytes[841];
844 var kk=bytes[842];
845 var ll=bytes[843];
846 var mm=bytes[844];
847 var nn=bytes[845];
848 var oo=bytes[846];
849 var pp=bytes[847];
850 var qq=bytes[848];
851 var rr=bytes[849];
852 var ss=bytes[850];
853 var tt=bytes[851];
854 var uu=bytes[852];
855 var vv=bytes[853];
856 var ww=bytes[854];
857 var xx=bytes[855];
858 var yy=bytes[856];
859 var zz=bytes[857];
860 var aa=bytes[858];
861 var bb=bytes[859];
862 var cc=bytes[860];
863 var dd=bytes[861];
864 var ee=bytes[862];
865 var ff=bytes[863];
866 var gg=bytes[864];
867 var hh=bytes[865];
868 var ii=bytes[866];
869 var jj=bytes[867];
870 var kk=bytes[868];
871 var ll=bytes[869];
872 var mm=bytes[870];
873 var nn=bytes[871];
874 var oo=bytes[872];
875 var pp=bytes[873];
876 var qq=bytes[874];
877 var rr=bytes[875];
878 var ss=bytes[876];
879 var tt=bytes[877];
880 var uu=bytes[878];
881 var vv=bytes[879];
882 var ww=bytes[880];
883 var xx=bytes[881];
884 var yy=bytes[882];
885 var zz=bytes[883];
886 var aa=bytes[884];
887 var bb=bytes[885];
888 var cc=bytes[886];
889 var dd=bytes[887];
890 var ee=bytes[888];
891 var ff=bytes[889];
892 var gg=bytes[890];
893 var hh=bytes[891];
894 var ii=bytes[892];
895 var jj=bytes[893];
896 var kk=bytes[894];
897 var ll=bytes[895];
898 var mm=bytes[896];
899 var nn=bytes[897];
900 var oo=bytes[898];
901 var pp=bytes[899];
902 var qq=bytes[900];
903 var rr=bytes[901];
904 var ss=bytes[902];
905 var tt=bytes[903];
906 var uu=bytes[904];
907 var vv=bytes[905];
908 var ww=bytes[906];
909 var xx=bytes[907];
910 var yy=bytes[908];
911 var zz=bytes[909];
912 var aa=bytes[910];
913 var bb=bytes[911];
914 var cc=bytes[912];
915 var dd=bytes[913];
916 var ee=bytes[914];
917 var ff=bytes[915];
918 var gg=bytes[916];
919 var hh=bytes[917];
920 var ii=bytes[918];
921 var jj=bytes[919];
922 var kk=bytes[920];
923 var ll=bytes[921];
924 var mm=bytes[922];
925 var nn=bytes[923];
926 var oo=bytes[924];
927 var pp=bytes[925];
928 var qq=bytes[926];
929 var rr=bytes[927];
930 var ss=bytes[928];
931 var tt=bytes[929];
932 var uu=bytes[930];
933 var vv=bytes[931];
934 var ww=bytes[932];
935 var xx=bytes[933];
936 var yy=bytes[934];
937 var zz=bytes[935];
938 var aa=bytes[936];
939 var bb=bytes[937];
940 var cc=bytes[938];
941 var dd=bytes[939];
942 var ee=bytes[940];
943 var ff=bytes[941];
944 var gg=bytes[942];
945 var hh=bytes[943];
946 var ii=bytes[944];
947 var jj=bytes[945];
948 var kk=bytes[946];
949 var ll=bytes[947];
950 var mm=bytes[
```





3. Konfigurieren Sie LMDS200 über AT-Befehl oder LoRaWAN-Downlink

Sie können LMDS200 über AT-Befehle oder LoRaWAN Downlink konfigurieren.

- AT-Befehl-Verbindung: Siehe [FAQ](#).
- LoRaWAN-Downlink-Anweisungen für verschiedene Plattformen: [IoT LoRaWAN Server](#)

Es gibt zwei Arten von Befehlen zur Konfiguration des LMDS200:

- **Allgemeine Befehle.** Diese

Befehle dienen zur Konfiguration von:

- Allgemeine Systemeinstellungen wie: Uplink-Intervall.
- Befehle im Zusammenhang mit dem LoRaWAN-Protokoll und der Funkverbindung.

Sie sind für alle Dragino-Geräte, die DLWS-005 LoRaWAN Stack unterstützen (**Hinweis****), identisch. Diese Befehle finden Sie im Wiki: [Endgeräte-AT-Befehle und Downlink-Befehle](#)

- **Speziell für LMDS200 entwickelte Befehle**

Diese Befehle gelten nur für LMDS200, wie unten aufgeführt:

3.1 Sendeintervallzeit einstellen (0x01)

Funktion: Ändern des LoRaWAN-Endknoten-Sendeintervalls.

AT-Befehl: AT+TDC

Befehlsbeispiel	Funktion	Antwort
AT+TDC=?	Aktuelles Sendeintervall anzeigen	30000 OK Das Intervall beträgt 30000 ms = 30 s
AT+TDC=60000	Sendeintervall einstellen	OK Sendeintervall auf 60000 ms = 60 Sekunden einstellen

Downlink-Befehl: 0x01

Format: Befehlscode (0x01) gefolgt von einem Zeitwert mit 3 Bytes.

Wenn die Downlink-Nutzlast = 0100003C ist, bedeutet dies, dass das Sendeintervall des END-Knotens auf 0x00003C = 60 (S) gesetzt wird, während der Typcode 01 ist.

- Beispiel 1: Downlink-Nutzlast: 0100001E // Sendeintervall (TDC) = 30 Sekunden einstellen
- Beispiel 2: Downlink-Nutzlast: 0100003C // Sendeintervall (TDC) = 60 Sekunden einstellen

3.2 Alarm-Sendeintervallzeit (0x0D) einstellen

Funktion: Ändern Sie das Alarmübertragungsintervall des LoRaWAN-Endknotens. Standardwert: 1 Minute

AT-Befehl: AT+ATDC

Befehlsbeispiel	Funktion	Antwort
AT+ATDC=?	Aktuelles Alarmübertragungsintervall	an zei ge n 1 OK Das Intervall beträgt 1 Minute
AT+ATDC=1	Alarmübertragungsintervall einstellen	OK Alarmübertragungsintervall auf 1 Minute einstellen

Downlink-Befehl: 0x0D

Format: Befehlscode (0x0D) gefolgt von 1 Byte für den Zeitwert.

Wenn die Downlink-Nutzlast = 0D02 ist, bedeutet dies, dass das Alarmübertragungsintervall des END-Knotens auf 2 Minuten eingestellt wird, während der Typcode 0D ist.

3.3 Alarmabstand einstellen (0xA2)

LMDS200 unterstützt eine Alarmfunktion. Wenn LMDS200 feststellt, dass die Entfernung die Alarmeinstellungen überschreitet, wechselt LMDS200 in den Alarmmodus und verwendet das [ATDC-Intervall](#) (Standardwert ist 1 Minute), um den Entfernungswert hochzuladen. Der Alarmmodus dauert 60 Uplinks (Standardwert 1 Stunde) und kann mit dem Downlink-Befehl 0xA300 beendet werden.

Hinweis: Der Alarmmodus gilt nur für die Entfernung 1 (das nächstgelegene Objekt).

Der Benutzer kann die Alarmdistanz einstellen, um den Alarmmodus zu aktivieren/deaktivieren.

AT+ALARMC=AABBCCDD

- AABB: Hexadezimalwert für den unteren Alarmschwellenwert, CCDD: Hexadezimalwert für den oberen Alarmschwellenwert
- Wenn 0xAABB=0 und 0xCCDD≠0, wird bei Überschreiten des Maximalwerts ein Alarm ausgelöst.
- Wenn 0xAABB≠0 und 0xCCDD=0xFFFF, wird der Alarm ausgelöst, wenn der Wert unter dem Minimalwert liegt.
- Wenn 0xAABB≠0 und 0xCCDD≠0, wird der Alarm ausgelöst, wenn der Wert über dem Maximalwert oder unter dem Minimalwert liegt.

Beispiel:

AT+ALARMC=006400C8 // Alarm bei < 100 oder höher als 200.

Downlink-Nutzlast:

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

3.4 Alarmmodus aktivieren/deaktivieren (0xA3)

Funktion: Alarmmodus aktivieren/deaktivieren:

AT-Befehl: AT+ALARM=0 (Alarmmodus verlassen) oder AT+ALARM=1 (Alarmmodus aktivieren)

Downlink-Befehl:

0xA3 00 // Alarmmodus beenden.

0xA3 01 // Alarmmodus aktivieren. Bei Aktivierung des Alarmmodus wird das ATDC-Intervall in den nächsten 59 Uplinks verwendet.

3.5 Interrupt-Modus einstellen (0x06)

Funktion, Interrupt-Modus für GPIO_EXIT einstellen.

AT-Befehl: AT+INTMOD

Befehlsbeispiel	Funktion	Antwort
AT+INTMOD=?	Aktuellen Interrupt-Modus anzeigen	0 OK Der Modus ist 0 = Keine Unterbrechung
AT+INTMOD=2	Interrupt-Auslösemodus einstellen 0. (Interrupt deaktivieren), 1. (Auslösung durch steigende und fallende Flanke) 2. (Auslösung durch fallende Flanke) 3. (Auslösung durch steigende Flanke)	OK

Downlink-Befehl: 0x06

Format: Befehlscode (0x06) gefolgt von 3 Bytes.

Das bedeutet, dass der Interrupt-Modus des Endknotens auf 0x000003=3 (steigende Flanke) gesetzt ist und der Typcode 06 lautet.

- Beispiel 1: Downlink-Nutzlast: 06000000 // Interrupt-Modus ausschalten
- Beispiel 2: Downlink-Nutzlast: 06000003 // Interrupt-Modus auf steigende Flanke setzen

4. Batterie & Stromverbrauch

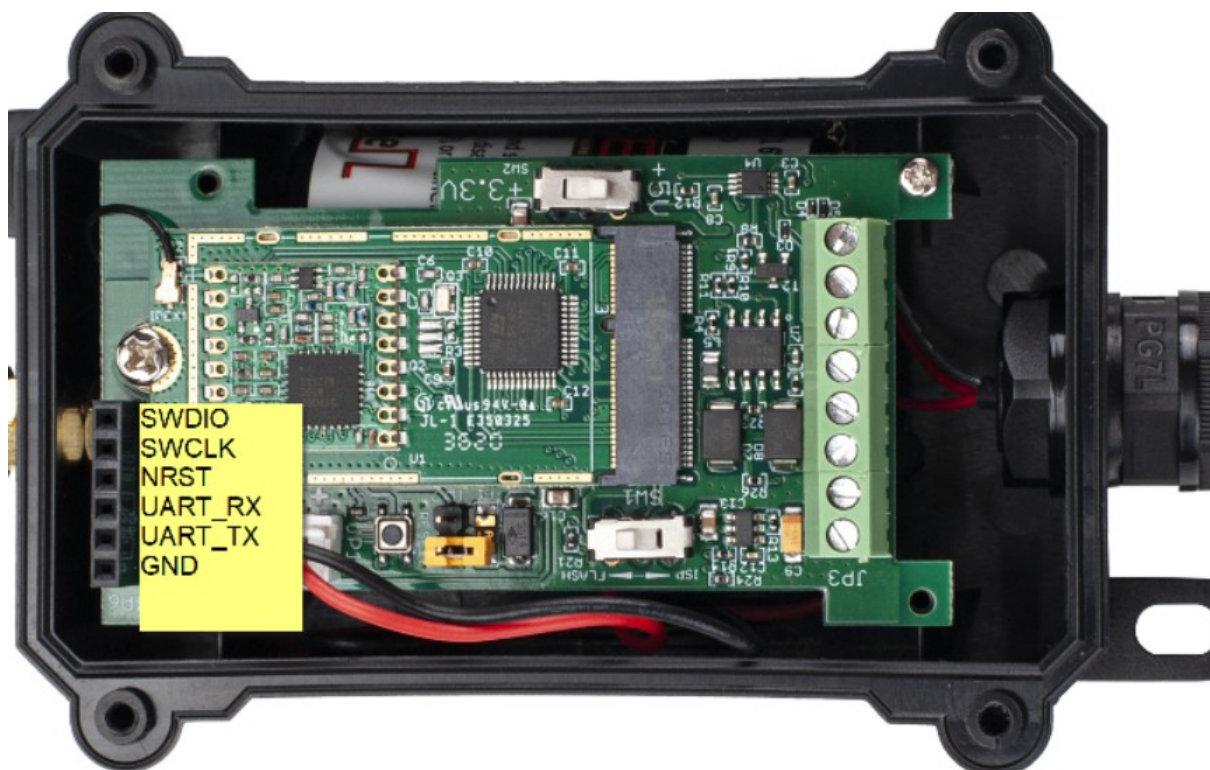
LMDS200 verwendet einen ER26500 + SPC1520-Akku. Unter dem folgenden Link finden Sie detaillierte Informationen zum Akku und zum Austausch.

[Batterieinformationen und Analyse des Stromverbrauchs.](#)

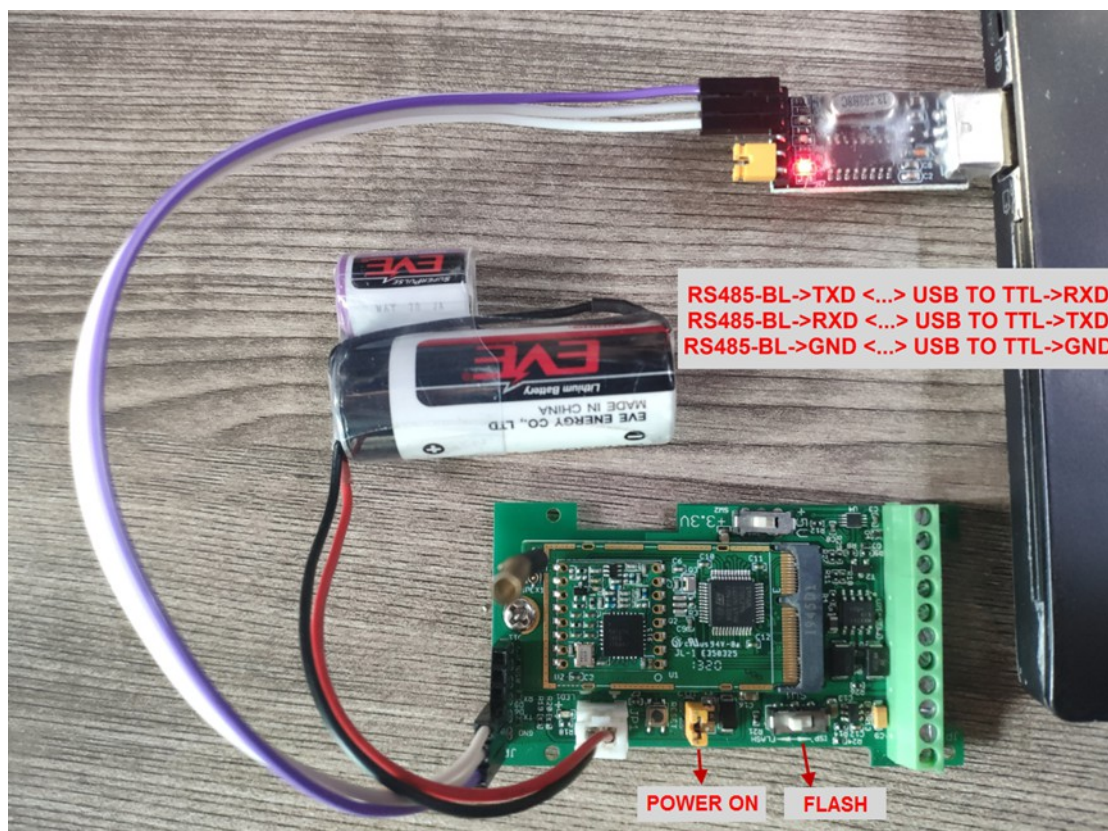
5. FAQ

5.1 Verwendung von AT-Befehlen zur Konfiguration des LMDS200

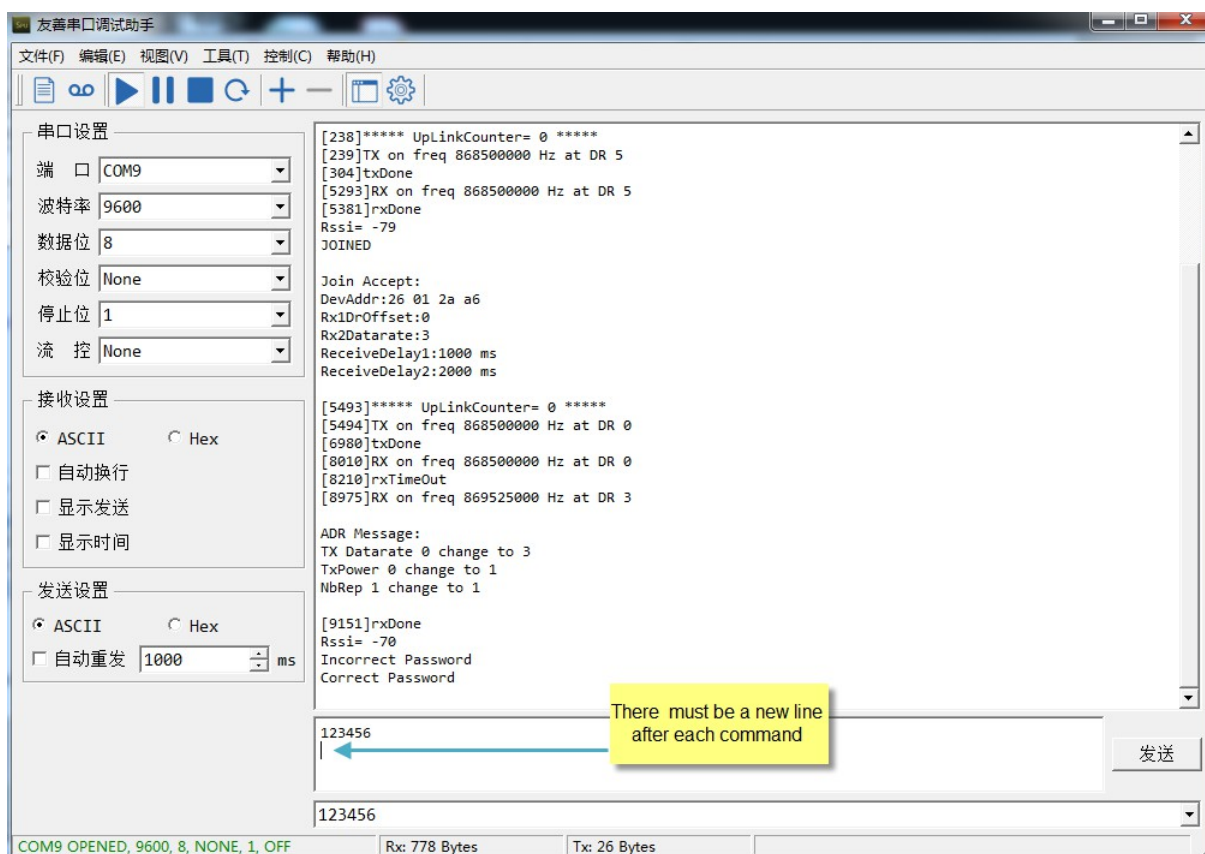
Foto der UART-Verbindung des LMDS200



LMDS200 unterstützt den AT-Befehlssatz. Benutzer können einen USB-zu-TTL-Adapter und das 3,5-mm-Programmierskabel verwenden, um eine Verbindung zum LMDS200 herzustellen und AT-Befehle zu verwenden, wie unten gezeigt.



Auf dem PC müssen Sie die serielle Baudrate auf **9600** einstellen, um auf die serielle Konsole für LMDS200 zugreifen zu können. LMDS200 gibt nach dem Einschalten die folgenden Systeminformationen aus:



5.2 Wie kann die Firmware aktualisiert werden?

Eine neue Firmware ist möglicherweise verfügbar für:

- Unterstützung neuer Funktionen
- Zur Behebung von Fehlern
- Änderung der LoRaWAN-Bänder.

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

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

5.3 So ändern Sie die LoRa-Frequenzbänder/Region

Befolgen Sie die Anweisungen [zum Aktualisieren des Images](#).

Wählen Sie beim Herunterladen der Bilder die gewünschte Bilddatei zum Herunterladen aus.

6. Fehlerbehebung

6.1 AT-Befehlseingabe funktioniert nicht

Wenn der Benutzer die Konsolenausgabe sehen kann, aber keine Eingaben in das Gerät vornehmen kann, überprüfen Sie bitte, ob Sie beim Senden des Befehls bereits die **Eingabetaste** gedrückt haben. Einige serielle Tools senden beim Drücken der Senden-Taste keine **Eingabetaste**, sodass der Benutzer die Eingabetaste in seiner Zeichenfolge hinzufügen muss.

7. Bestellinformationen

Teilenummer: **LMDS200-XX XX**:

Das Standardfrequenzband

- **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. Verpackungsinformationen

Lieferumfang

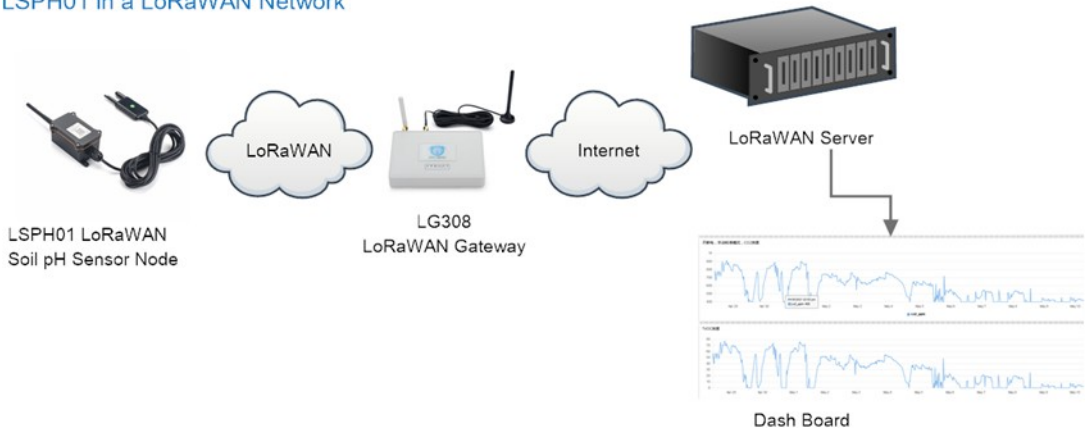
LMDS200 LoRaWAN-Mikrowellenradar-Entfernungssensor x 1

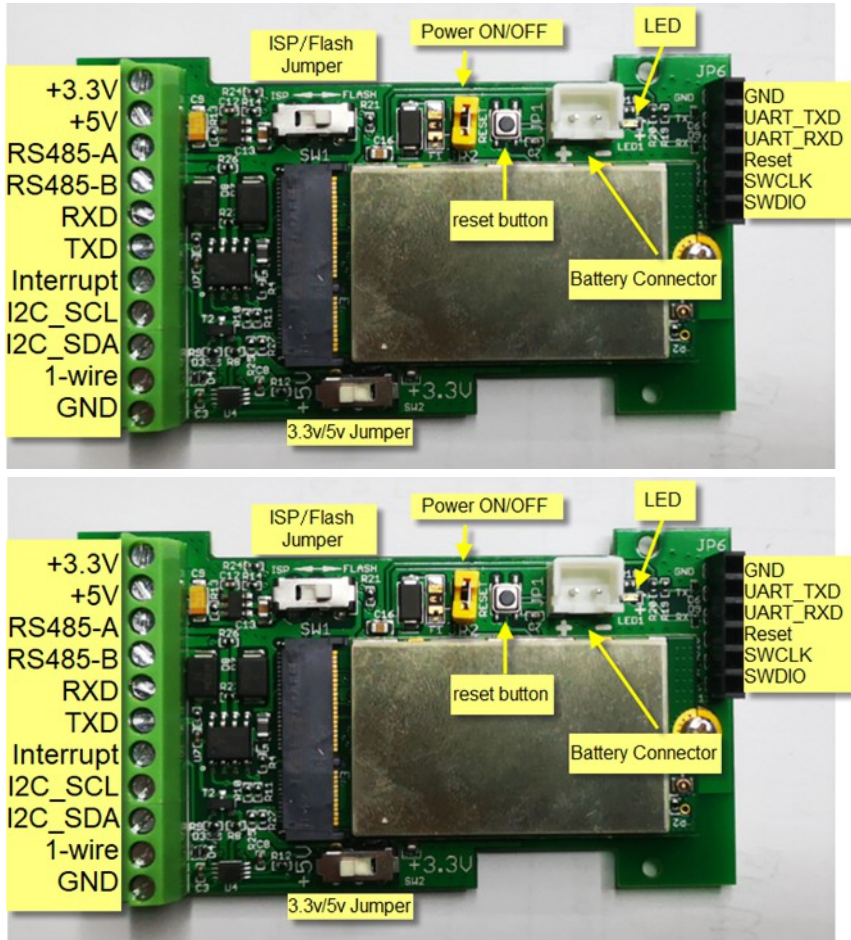
9. Unterstützung

- Der Support ist montags bis freitags von 09:00 bis 18:00 Uhr GMT+8 verfügbar. Aufgrund unterschiedlicher Zeitzonen können wir keinen Live-Support anbieten. Ihre Fragen werden jedoch so schnell wie möglich innerhalb der oben genannten Zeiten beantwortet.
- Geben Sie so viele Informationen wie möglich zu Ihrer Anfrage an (Produktmodelle, genaue Beschreibung Ihres Problems und Schritte zur Reproduktion usw.) und senden Sie eine E-Mail an support@dragino.com .



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

2

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 ⓘ *

.. .. . 00

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
- 2 Network layer settings**
Frequency plan, regional parameters, end device class and session keys.
- 3 **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 REV A

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.
- 3 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

Benutzerhandbuch für LoRaWAN/NB-IoT-Endknoten – LMS200
-- LoRaWAN-Mikrowellenradar-Entfernungssensor
Benutzerhandbuch

16:00:36 Link ADR request enqueued DevAddr: 26 00 2E E4

16:00:36 Successfully scheduled data downlink ... DevAddr: 26 00 2E E4

16:00:36 Schedule data downlink for transmissi... DevAddr: 26 00 2E E4 Rx1 Delay: 5

16:00:36 Store upstream data message DevAddr: 26 00 2E E4

16:00:36 Forward data message to Application S... DevAddr: 26 00 2E E4 MAC payload: 66 18 21 93 7E 92 00 94 0F A6 DC FPort: 2 SNR: 6.5 RSSI: -89 Bandwidth: 125000

16:00:36 Forward uplink data message DevAddr: 26 00 2E E4 Payload: { Batt: "3.365 V", Interupt_flag: 0, Message_Type: 1, PH1_SOIL: "0.62", TEMP_SOIL: "20.68", TempC_DS18B20: "0.00 °C" } 80 25 00 00 00 00 00 00 00 00 00 00 00 00 00 00

16:00:36 Receive uplink data message DevAddr: 26 00 2E E4

16:00:36 Successfully processed data message DevAddr: 26 00 2E E4 FPort: 2 MAC payload: 66 18 21 93 7E 92 00 94 0F A6 DC Bandwidth: 125000 SNR: 6.5 RSSI: -89 Raw payload: 40 E4 2E 00 26 00 00 00 02 66 18 21 93 7E 92 00 94 0F A

16:00:36 Receive data message DevAddr: 26 00 2E E4 FPort: 2 MAC payload: 66 18 21 93 7E 92 00 94 0F A6 DC Bandwidth: 125000 SNR: 6.5 RSSI: -89 Raw payload: 40 E4 2E 00 26 00 00 00 02 66 18 21 93 7E 92 00 94 0F A

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

Uplink Downlink

These payload formatters are executed on uplink messages from this end device and take precedence over application level payload formatters.

Formatter type

☐ Use application payload formatter ☐ None ☒ Javascript ☐ 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

THE THINGS NETWORK THE THINGS STACK Community Edition Overview Applications Gateways Organizations

Webhooks

Choose webhook template

Ubidots Integrate with Ubidots over Ubidots Functions

Datacake Send data to Datacake via TTN adapter

TagoIO Integrate with TagoIO

Akenza Integrate with Akenza Core

ThingSpeak

Qubitro

thethings.io

Applications > lgt92test > Webhooks > Add > Datacake

Add custom webhook

Template information



Datacake

Send data to Datacake via TTI adapter

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

Template settings

Webhook ID *

Token *

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
☐		The Things Network V2 The old Things Network	Uplinks	Downlinks
☐		Helium	Uplinks	Downlinks
☐		LORIoT	Uplinks	Downlinks
☐		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 66 8 bytes	 LSPH01

+ Add another device

Consistent with DEUI on TTN

Back

Next

LSPH01

Location	Serial Number	Last update	Product Slug
-	00000000000000066	-	lsph01

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

2/1/1 1:00

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.

```
3 // (array) or bytes to an object or value.
4 var value=(bytes[0]<<8 | bytes[1]) & 0xFFFF;
5 var batValue=(value<<8 | bytes[2]) & 0xFFFF;
6
7 value=bytes[2]<<8 | bytes[3];
8 if(bytes[2] & 0x00)
9   (value |= 0xFFFF0000);
10 var temp_0518020=(value/10).toFixed(2); //0518020,temperature
11
12 value=bytes[4]<<8 | bytes[5];
13 var ph=(value/100).toFixed(2);
14
15 value=bytes[6]<<8 | bytes[7];
16 var temp0;
17 if((value & 0x0000)>>15 == 0)
18   temp0=(value/10).toFixed(2); //temp_501,temperature
19 else if((value & 0x0000)>>15 == 1)
20   temp0=(value-0xFFFF)/10).toFixed(2);
21
22 var i_flag = bytes[8];
23 var msg_type = bytes[10];
24 return {
25   {
26     field: "BAT",
27     value: batV
28   },
29   {
30     field: "PH",
31     value: ph
32   },
33   {
34     field: "TEMP",
35     value: temp
36   }
37 };
```

Payload

Port

Try Decoder

Fields

Add Field

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

LSPH01

Location

Serial Number

Last update

Product Slug

0000000000000066

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

lspH01

Dashboard

History

Downlinks

Configuration

Debug

Rules

Permissions

BAT

a few seconds ago

BAT

TEMP

a few seconds ago

temp

PH

a few seconds ago

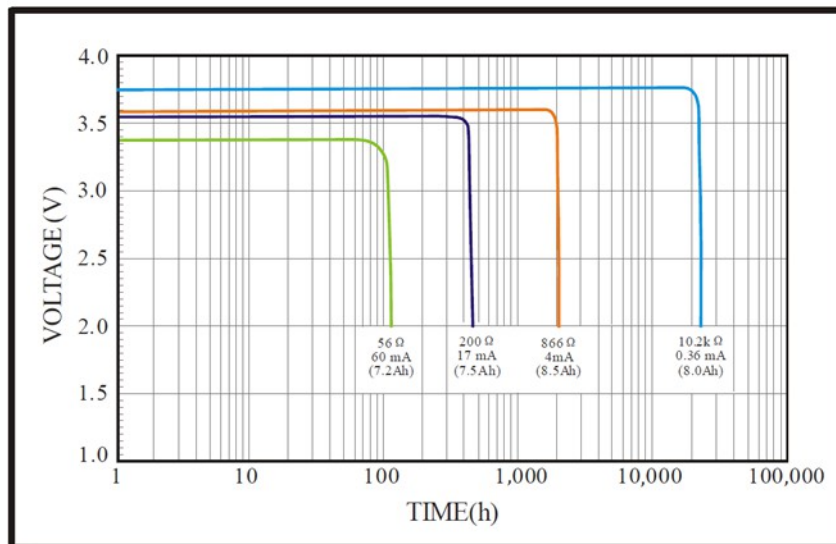
PH

Seite38 /105 – zuletzt geändert von Mengting Qiu am 07.03.2024 um 08:41
Uhr

Übersetzt mit DeepL
iot-shop



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



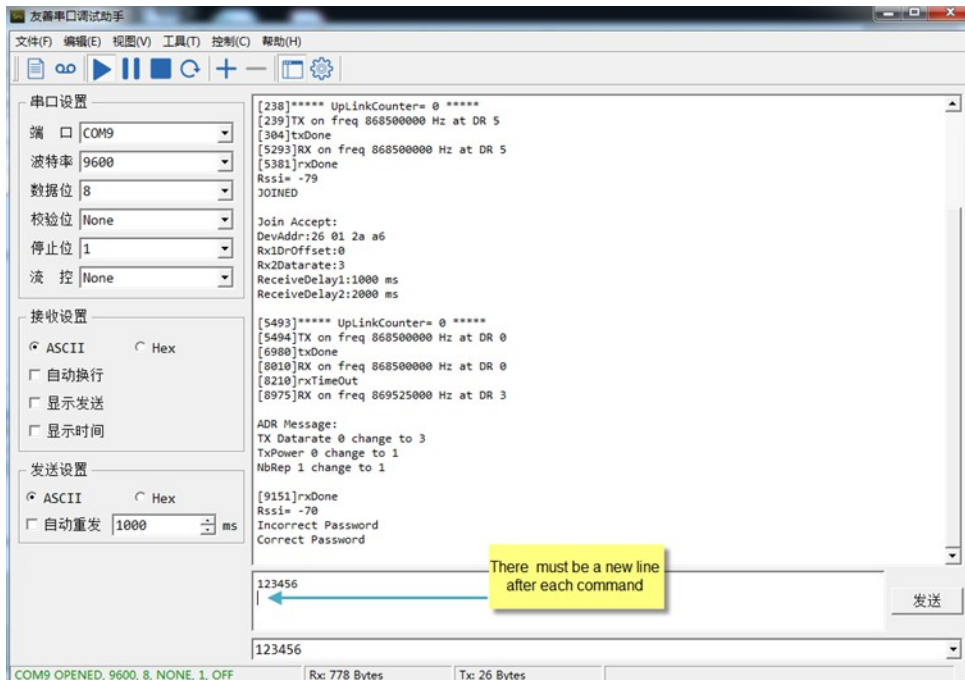
How to use:

- Please do not modify the formula in the table
- 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→Watching TX→Transmit RX→Receive

Battery Life Calculator

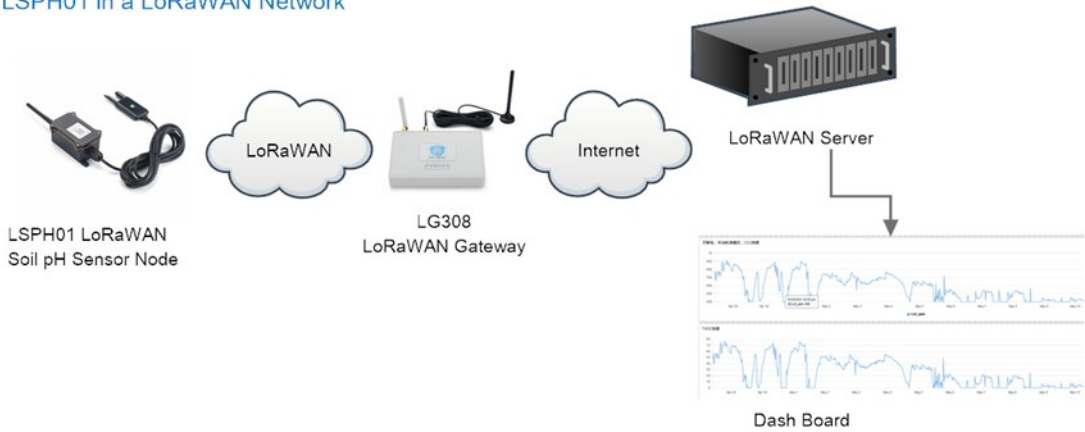
Product	battery capacity(mAh)	
LD001_LoRaWAN_Door_Sensor	240	
UNIT	TDC (Link Interval)	Work Mode
min	20	MOD-1

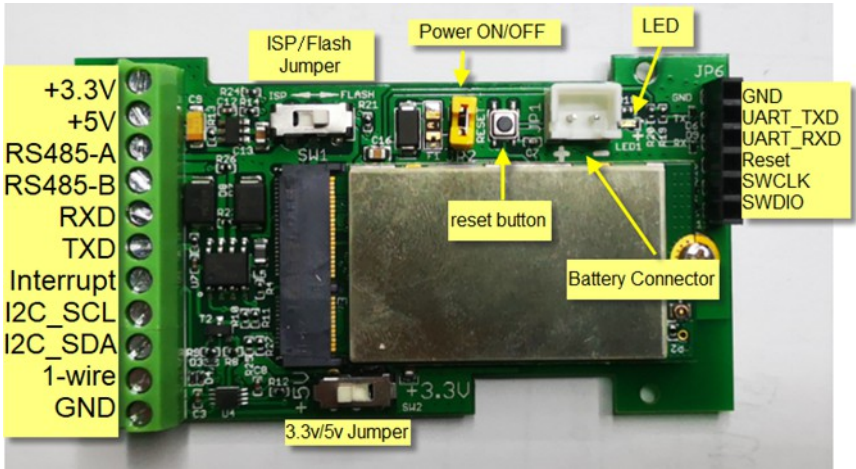
	Sleep power (mA/ms)	Sampling power (mA/ms)	TX power (mA/ms)	RX1 power (mA/ms)	RX2 power (mA/ms)	Watching power (mA/ms)	Average power (mA)	Detect power (mA/s)	Life expectancy (yr)
11000	DR1_SF7_125K_16dB	8400	427.10444	7367.8144	880.58488	4097.083	757.1706667	0.018706667	1.5
	DR4_SF5_125K_16dB	8400	427.10444	11210.2528	950.0943	4097.083	757.1706667	0.021192528	1.2
	DR1_SF9_125K_16dB	8400	427.10444	7367.8144	1068.0336	4097.083	757.1706667	0.011986736	0.8
	DR2_SF10_125K_16dB	8400	427.10444	42244.125	1461.4676	4097.083	757.1706667	0.047792297	0.6
	DR3_SF11_125K_16dB	8400	427.10444	94011.6	2230.4828	4097.083	757.1706667	0.091509095	0.3
	DR0_SF12_125K_16dB	8400	427.10444	168081	4097.083	4097.083	757.1706667	0.154625330	0.2
35005	DR1_SF7_125K_20dB	8400	427.10444	8461.476	683.13991	1587.135	757.1706667	0.010008139	1.6
	DR2_SF8_125K_20dB	8400	427.10444	15170.785	921.6491	1587.135	757.1706667	0.027707198	1.2
	DR3_SF9_125K_20dB	8400	427.10444	27754.383	941.388	1587.135	757.1706667	0.017794722	0.8
	DR0_SF10_125K_20dB	8400	427.10444	40740.42	995.2391	1587.135	757.1706667	0.060701365	0.5



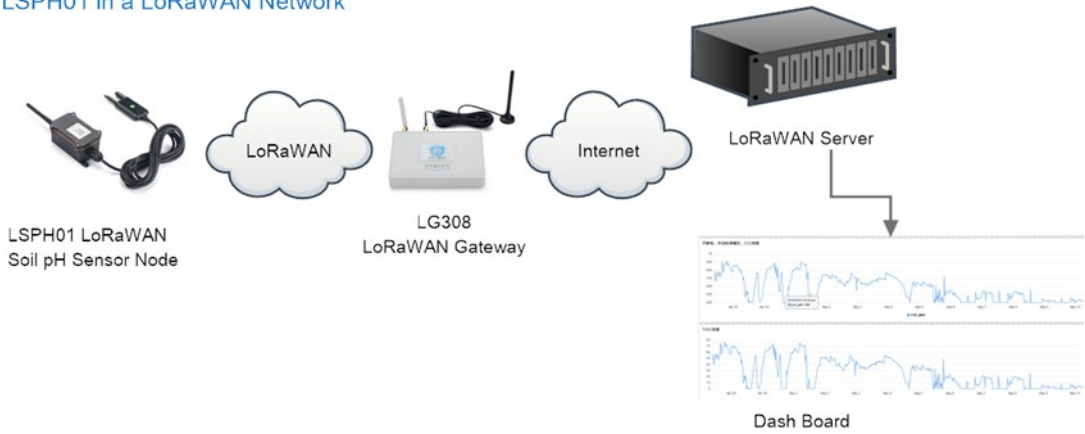


LSPH01 in a LoRaWAN Network





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

1

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

2

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 ⓘ *

.. .. . 00

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
- 2 Network layer settings**
Frequency plan, regional parameters, end device class and session keys.
- 3 **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 REV A

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.
- 3 **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

Benutzerhandbuch für LoRaWAN/NB-IoT-Endknoten – LMS200

-- LoRaWAN-Mikrowellenradar-Entfernungssensor

Benutzerhandbuch

↔ 16:00:36 Link ADR request enqueued DevAddr: 26 00 2E E4

↓ 16:00:36 Successfully scheduled data downlink ... DevAddr: 26 00 2E E4

↓ 16:00:36 Schedule data downlink for transmissi... DevAddr: 26 00 2E E4 Rx1 Delay: 8

⬆ 16:00:36 Store upstream data message DevAddr: 26 00 2E E4

↑ 16:00:36 Forward data message to Application S... DevAddr: 26 00 2E E4 MAC payload: 66 18 21 93 7E 92 00 94 0F A6 DC FPort: 2 SNR: 6.5 RSSI: -89 Bandwidth: 125000

↑ 16:00:36 Forward uplink data message DevAddr: 26 00 2E E4 Payload: { Batt: "3.365 V", Interupt_flag: 0, Message_Type: 1, PH1_SOIL: "0.62", TEMP_SOIL: "20.68", TempC_DS10B20: "0.68 °C" } 80 25 00 00 00 00 00 00 00 00

↑ 16:00:36 Receive uplink data message DevAddr: 26 00 2E E4

↑ 16:00:36 Successfully processed data message DevAddr: 26 00 2E E4 FPort: 2 MAC payload: 66 18 21 93 7E 92 00 94 0F A6 DC Bandwidth: 125000 SNR: 6.5 RSSI: -89 Raw payload: 40 E4 2E 00 26 00 00 00 02 66 18 21 93 7E 92 00 94 0F A

↑ 16:00:36 Receive data message DevAddr: 26 00 2E E4 FPort: 2 MAC payload: 66 18 21 93 7E 92 00 94 0F A6 DC Bandwidth: 125000 SNR: 6.5 RSSI: -89 Raw payload: 40 E4 2E 00 26 00 00 00 02 66 18 21 93 7E 92 00 94 0F A

Overview

End devices

Live data

Payload formatters

Integrations

Collaborators

API keys

General settings

Overview

Live data

Messaging

Location

Payload formatters

Claiming

General settings

Uplink

Downlink

These payload formatters are executed on uplink messages from this end device and take precedence over application level payload formatters.

Formatter type

Use application payload formatter

None

JavaScript

GRPC service

CayennePP

Repository

Formatter parameter

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

Save changes

THE THINGS NETWORK

THE THINGS STACK Community Edition

Overview

Applications

Gateways

Organizations

Webhooks

API keys

Overview

End devices

Live data

Integrations

Webhooks

Pub/Subs

Storage integration

AWS IoT

LoRa Cloud

Collaborators

API keys

Choose webhook template

Ubidots

Datascade

TagoIO

Akenza

ThingSpeak

Qubitro

thethings.io

Applications > lgt92test > Webhooks > Add > Datacake

Add custom webhook

Template information



Datacake

Send data to Datacake via TTI adapter

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

Template settings

Webhook ID *

Token *

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
☐		The Things Network V2 The old Things Network	Uplinks	Downlinks
☐		Helium	Uplinks	Downlinks
☐		LORIoT	Uplinks	Downlinks
☐		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 66 8 bytes	 LSPH01

+ Add another device

Consistent with DEUI on TTN

Back

Next

LSPH01

Location

-

Serial Number

00000000000000066

Last update

-

Product Slug

lsph01

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

2/1/1 1:00

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.

```
3 // (array) of bytes to an object or fields.
4 var value=(bytes[0]<<8 | bytes[1]) & 0xffff;
5 var batv=value/1000; //Battery, units:V
6
7 value=bytes[2]<<8 | bytes[3];
8 if(bytes[2] & 0x00)
9 (value |= 0xffff0000);
10 var temp_0518020=(value/10).toFixed(2); //0518020,temperature
11
12 value=bytes[4]<<8 | bytes[5];
13 var phv=value/100; toFixed(2);
14
15 value=bytes[6]<<8 | bytes[7];
16 var temp0;
17 if((value & 0x0000)>>15 == 0)
18 temp=(value/10).toFixed(2); //temp_501,temperature
19 else if((value & 0x0000)>>15 == 1)
20 temp=(value-0xffff)/10).toFixed(2);
21
22 var i_flag = bytes[8];
23 var msg_type = bytes[10];
24 return {
25   {
26     field: "BAT",
27     value: batv
28   },
29   {
30     field: "PH",
31     value: phv
32   },
33   {
34     field: "TEMP",
35     value: temp
36   }
37 };
38 }
```

Payload

Port

Try Decoder

Fields

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

LSPH01

Location

Serial Number

Last update

Product Slug

-

0000000000000066

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

lspH01

Dashboard

History

Downlinks

Configuration

Debug

Rules

Permissions

BAT

a few seconds ago

BAT

TEMP

a few seconds ago

temp

PH

a few seconds ago

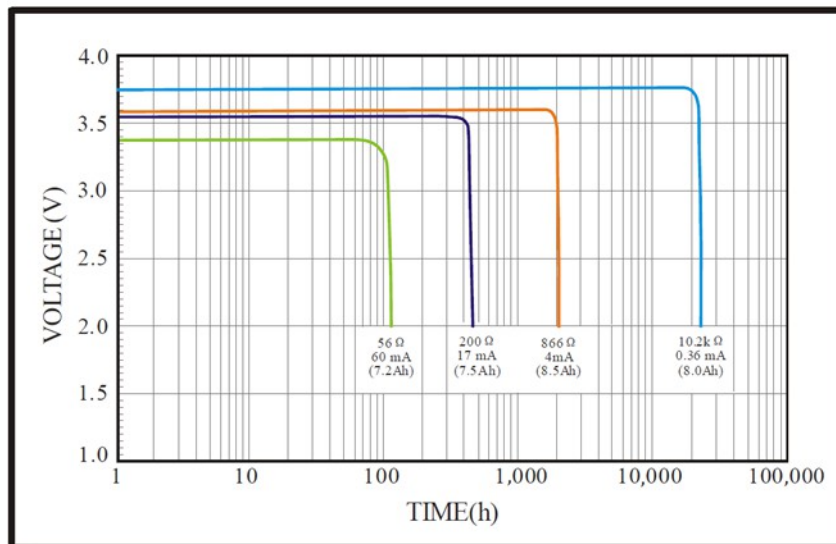
PH

Seite50 /105 – zuletzt geändert von Mengting Qiu am 07.03.2024 um 08:41
Uhr

Übersetzt mit DeepL
iot-shop



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

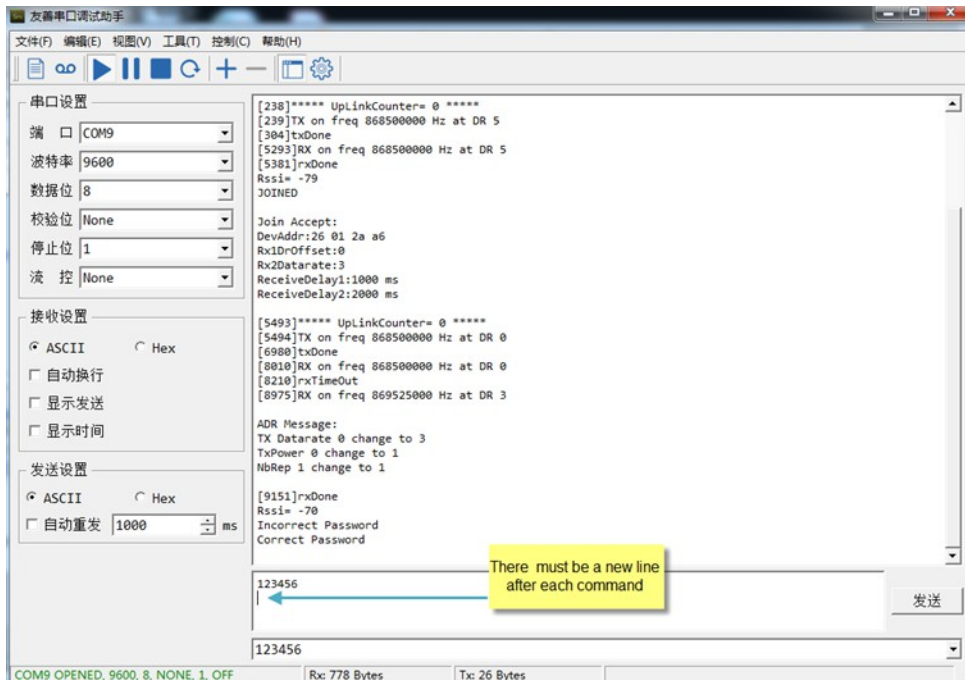


How to use:

- Please do not modify the formula in the table
- 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→Watching TX→Transmit RX→Receive

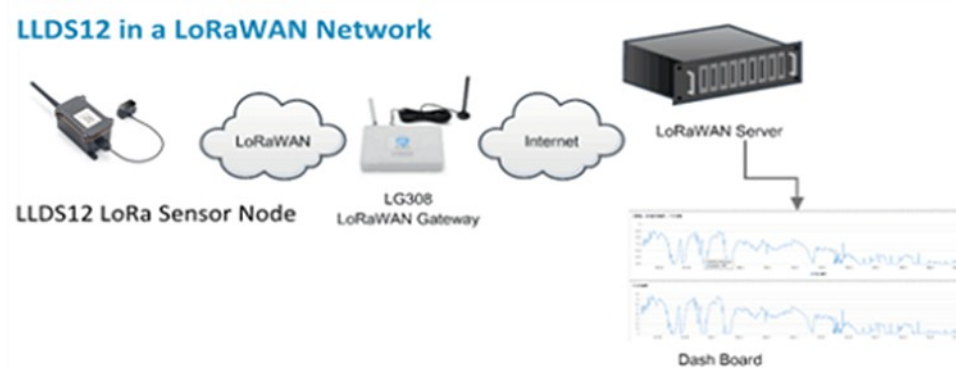
Battery Life Calculator

Product	Battery capacity (mAh)	UNIT	TDC (Link interval)	Work Mode	min	MOD-1	Sleep power (mA/mA)	Sampling power (mA/mA)	TX power (mA/mA)	RX1 power (mA/mA)	RX2 power (mA/mA)	Watching power (mA/mA)	Average power (mA)	Detect power (mA/mA)	Life expectancy (yr)
LD001_LoRaWAN_Door_Sensor	240		20				0.008	427.10444	7367.8144	880.5848	4097.083	757.1706667	0.018766667	0	1.5
							0.008	427.10444	11210.2528	950.0943	4097.083	757.1706667	0.021192528	0	1.2
							0.008	427.10444	7365.2608	1068.0136	4097.083	757.1706667	0.011986736	0	0.8
							0.008	427.10444	42244.125	1461.4676	4097.083	757.1706667	0.047792297	0	0.6
							0.008	427.10444	94011.6	2230.4828	4097.083	757.1706667	0.091509095	0	0.3
							0.008	427.10444	168081	4097.083	4097.083	757.1706667	0.154625135	0	0.2
LD005							0.008	427.10444	8461.476	683.13991	1587.135	757.1706667	0.010008134	0	1.6
							0.008	427.10444	15170.785	921.6491	1587.135	757.1706667	0.027707198	0	1.2
							0.008	427.10444	27754.183	941.388	1587.135	757.1706667	0.01779472	0	0.8
							0.008	427.10444	40740.42	995.2391	1587.135	757.1706667	0.060701365	0	0.3





LLDS12 in a LoRaWAN Network



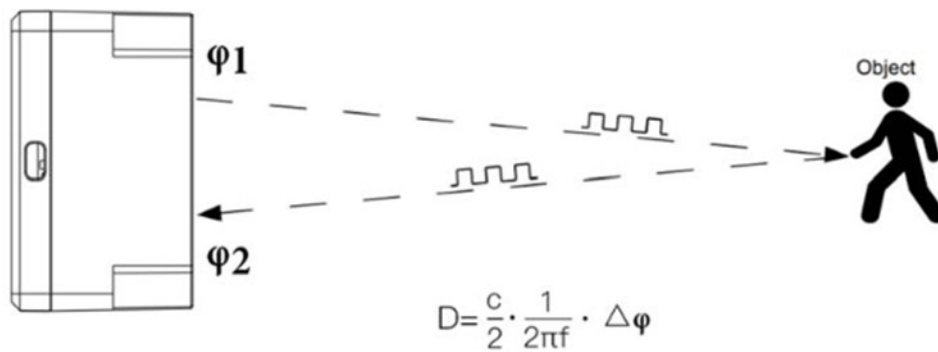
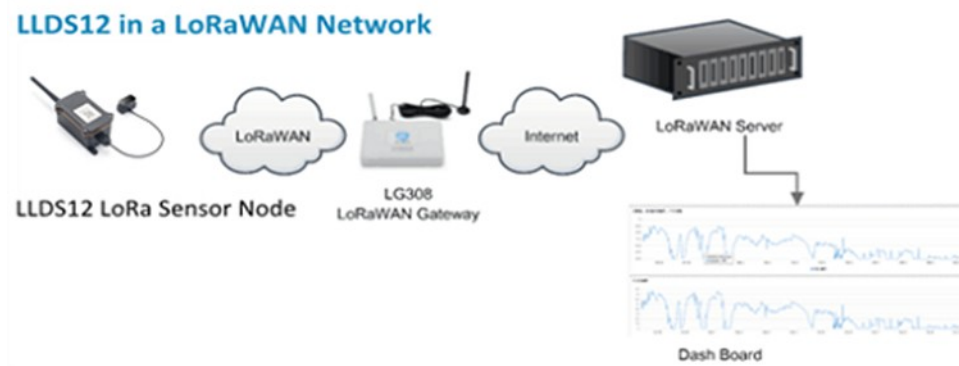
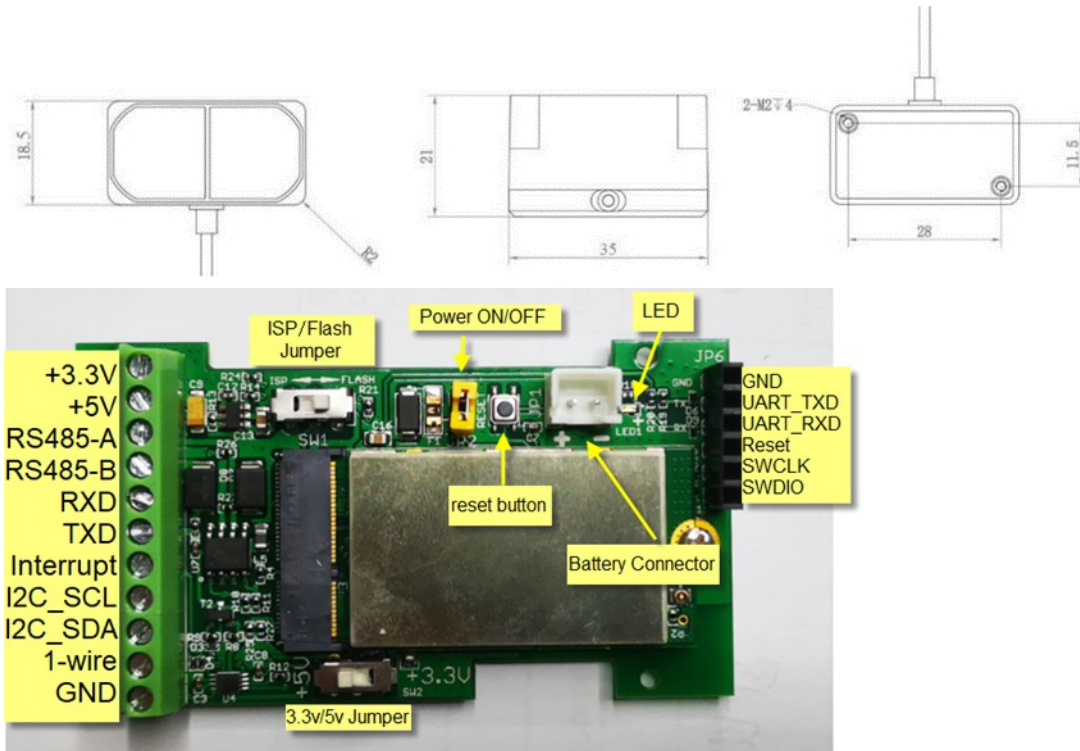


Figure 1 Schematics of TOF Principle

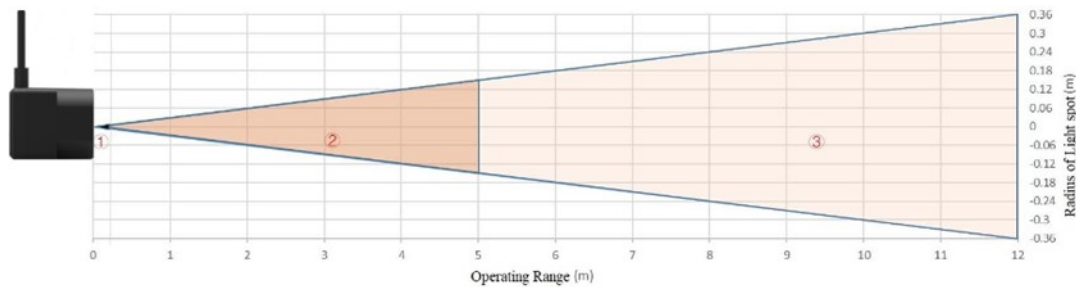


Figure 2 Schematic diagram of size of light spot

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

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

DEVICE

laq4

LAQ4-template

ltd75-test

lht65-one

lht65-test

lht65-two

lsn50

LSN50-template

LSPH01-template

LT22222

RS485-LN

ving 1 to 11 of 11 re

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

LLDS12

Next

Seite55 /105 – zuletzt geändert von Mengting Qiu am 07.03.2024 um 08:41 Uhr

Übersetzt mit DeepL
iot-shop

Add Device

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
 21 21 24 25 35 34 14 8 bytes	 LLDS12

+ Add another device

Back Next

Location: - Serial Number: A8404166A18219CF Last update: Mon May 31 2021 19:27:51 GMT+0800 Product Slug: dragino-ladd75-6

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

General Configuration

Name

ladd75-test

Location

Tags

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

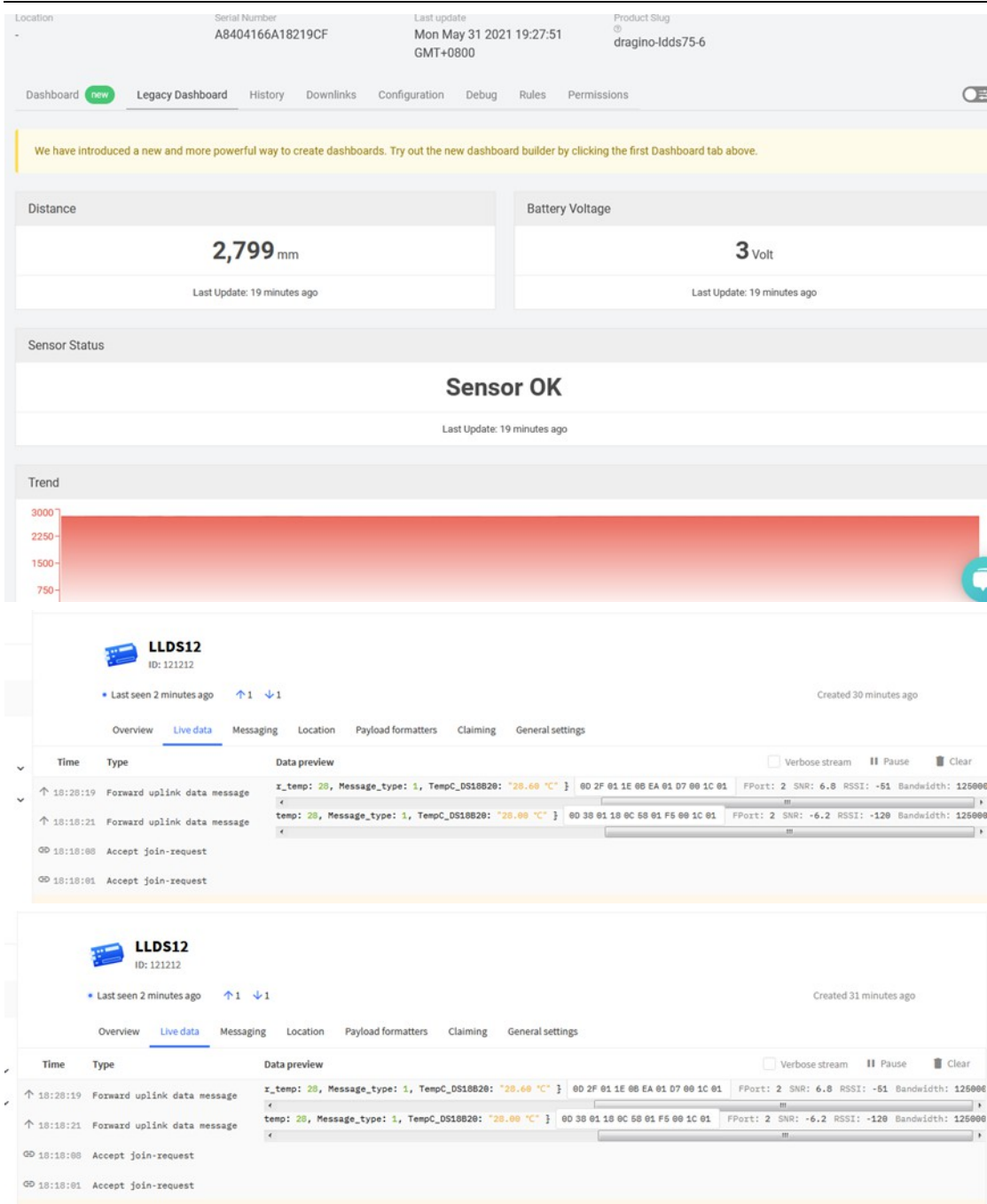
Add tag

Metadata

Metadata is displayed on the device overview and can be used in dashboards

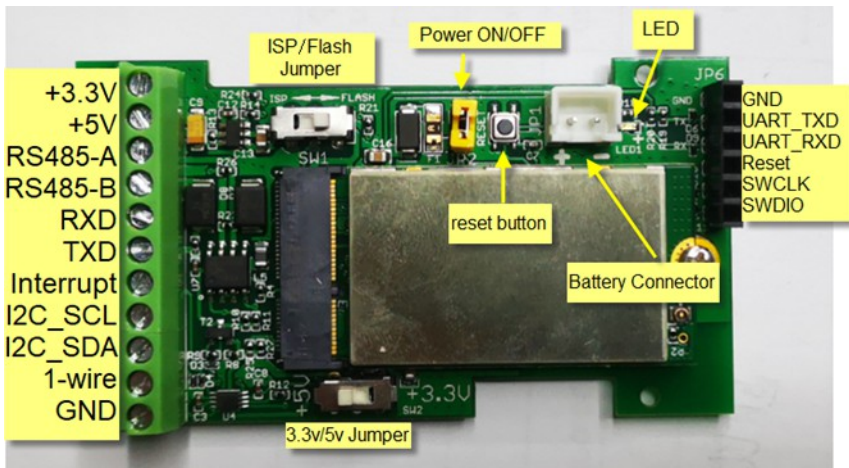


Benutzerhandbuch für LoRaWAN/NB-IoT-Endknoten – LMDS200
-- LoRaWAN-Mikrowellenradar-Entfernungssensor
Benutzerhandbuch





LDDS75 in a LoRaWAN Network



LDDS75 in a LoRaWAN Network



Log messages showing data transmission and processing:

- Forward data message to Application Server
- Forward uplink data message
- Receive uplink data message
- Successfully processed data message
- Drop data message (Uplink is a duplicate)
- Receive data message

Log details include DevAddr, MAC payload, FPort, SNR, RSSI, Bandwidth, and Payload (e.g., bat: 3.35, distance: 2671).

TTN Web Interface - Payload formatters configuration:

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

Uplink / Downlink

These payload formatters are executed on uplink messages from this end device and take precedence over application level payload formatters.

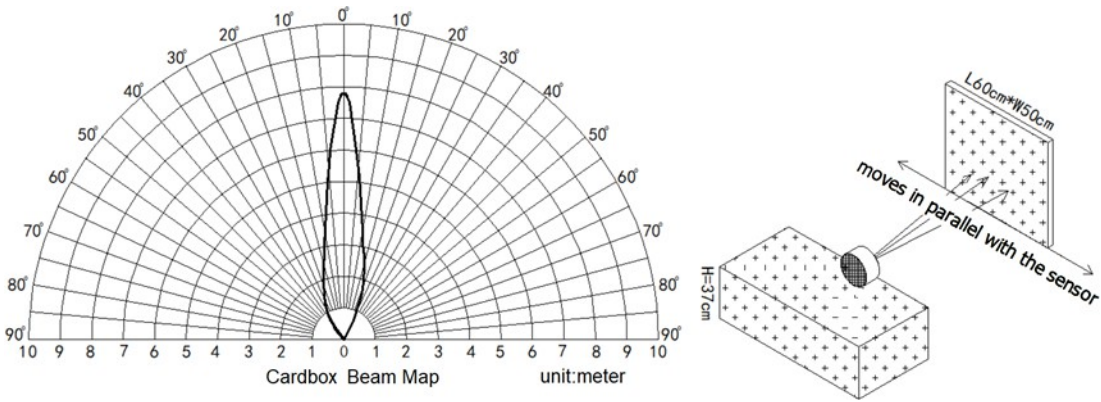
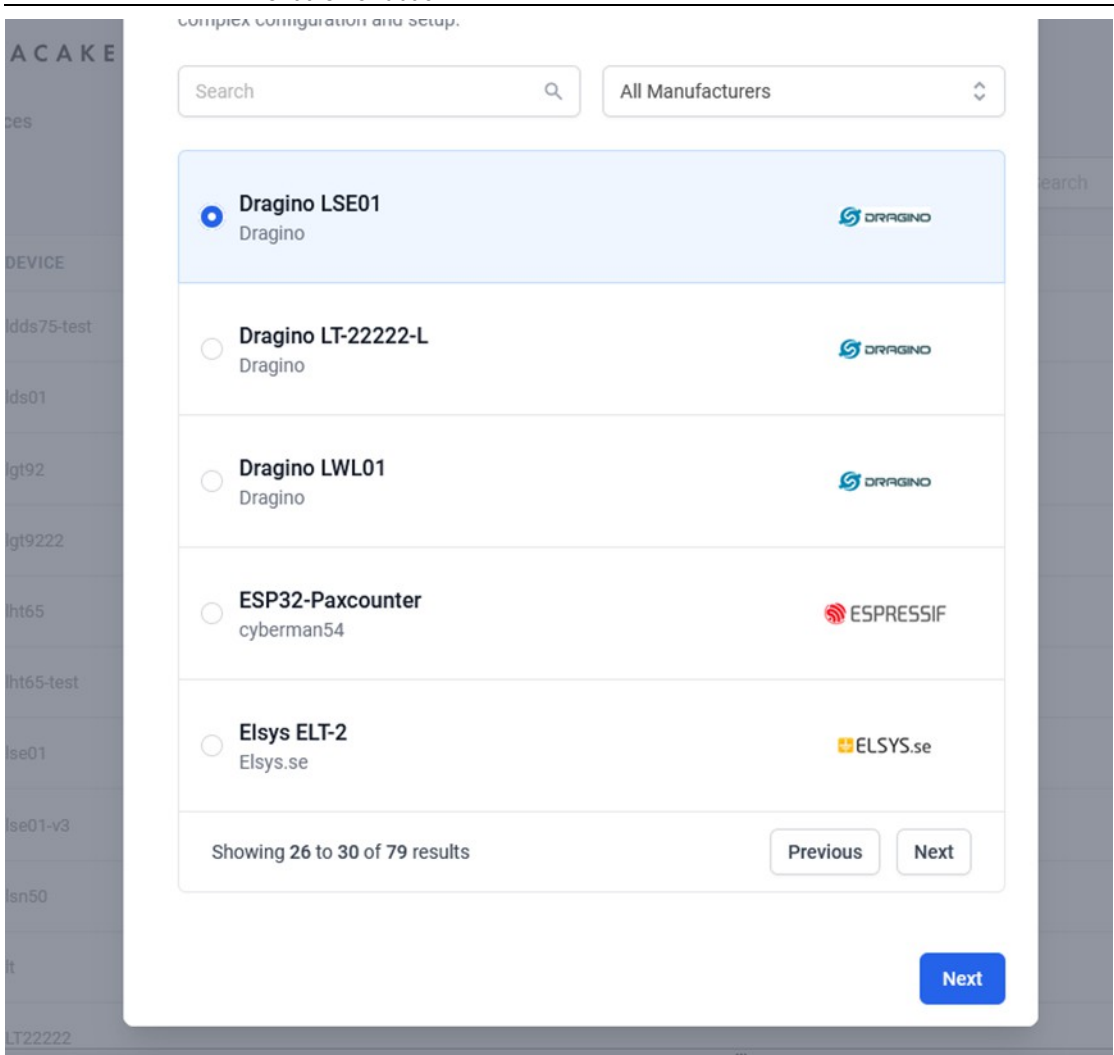
Formatter type:

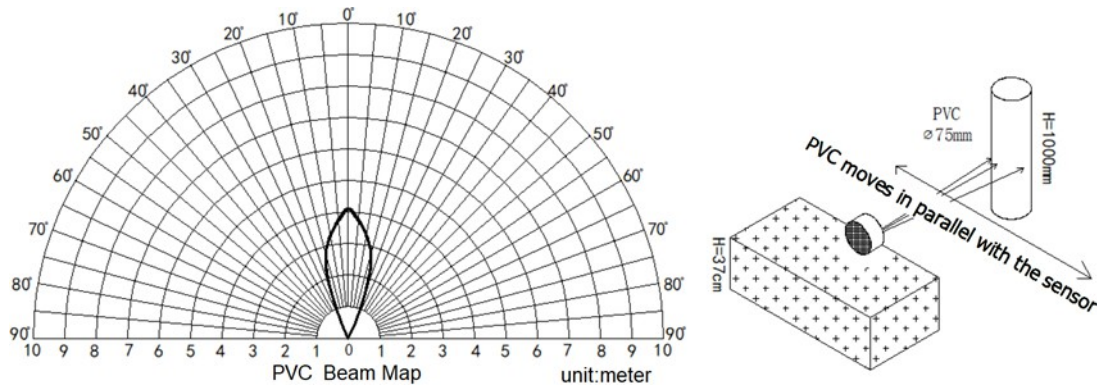
- Use application payload formatter
- None
- JavaScript**
- GRPC service
- CayenneLPP
- Repository

Formatter parameter:

```
function decodeUplink(input) {  
  return {  
    data: {  
      bytes: input.bytes  
    },  
    warnings: [],  
    errors: []  
  };  
}
```

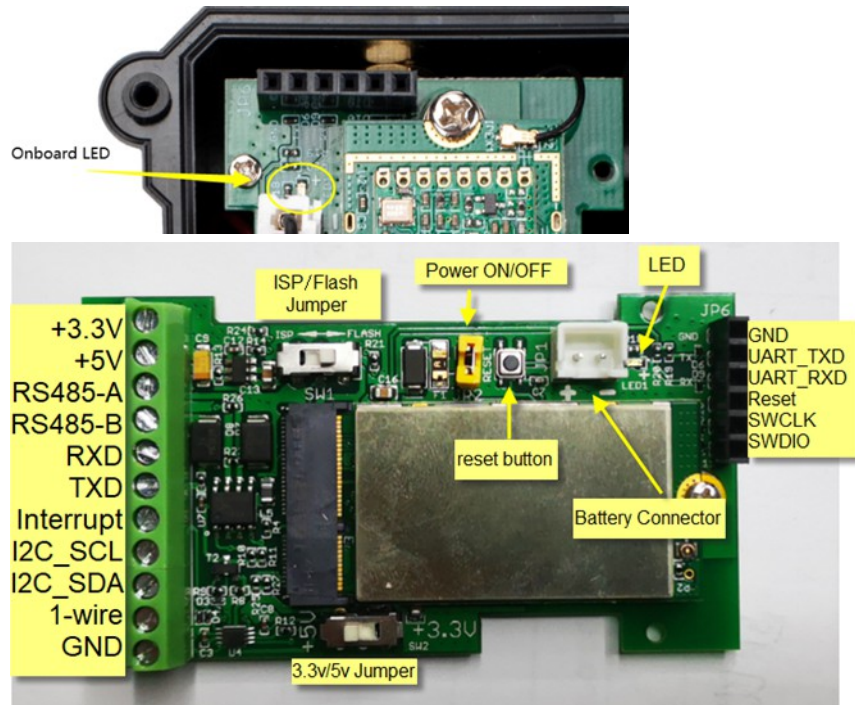
Save changes



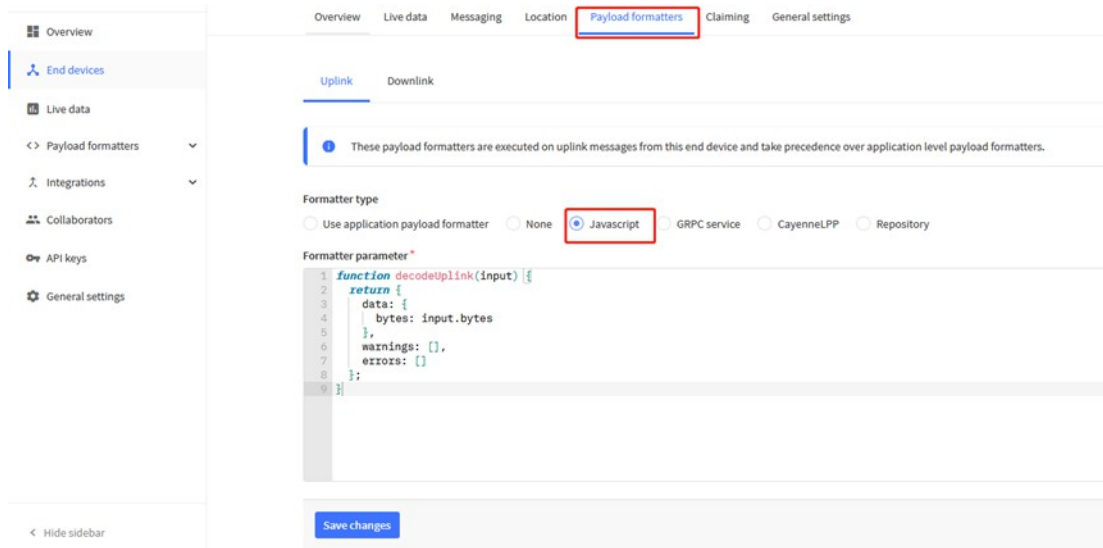


LDSS20 in a LoRaWAN Network





LMDS200 in a LoRaWAN Network

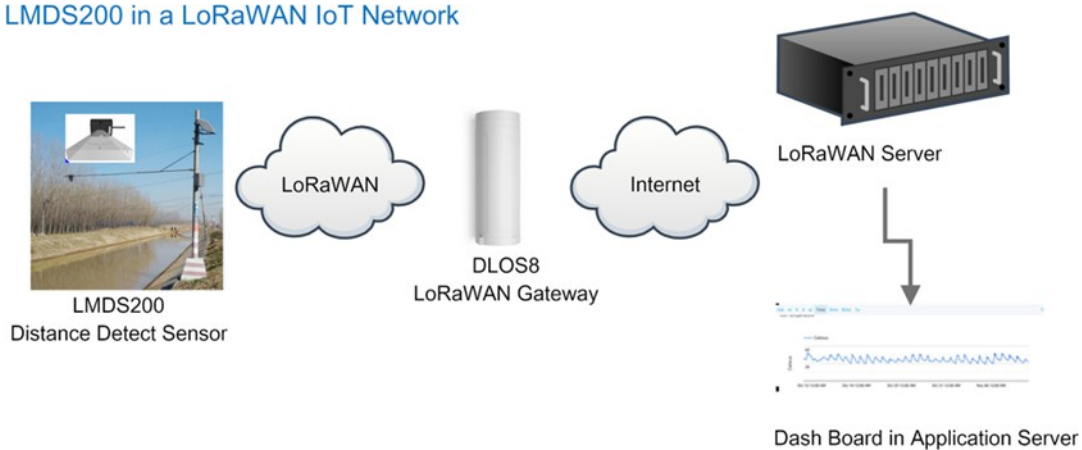




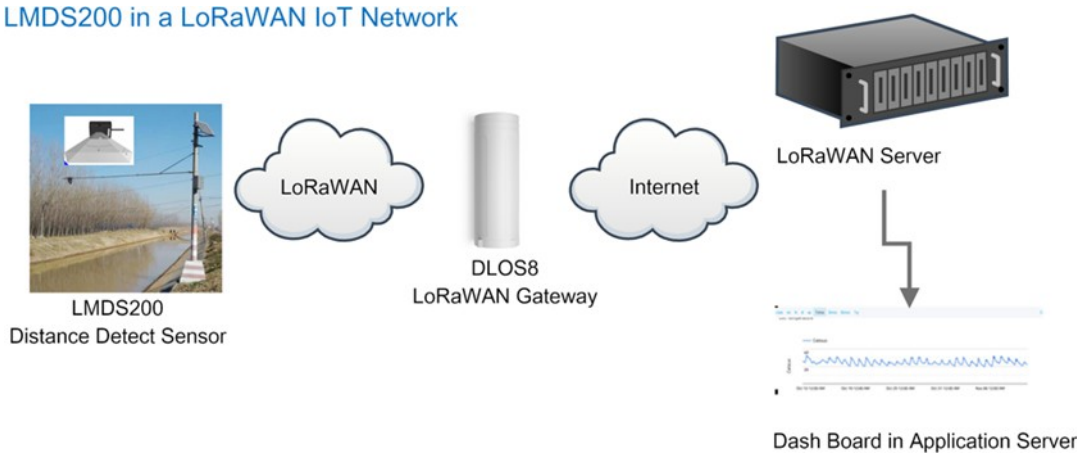




LMDS200 in a LoRaWAN IoT Network



LMDS200 in a LoRaWAN IoT Network



Register end device

From The LoRaWAN Device Repository [Manually](#) 1

Frequency plan [ⓘ] *
 2

LoRaWAN version [ⓘ] *

MAC V1.0.3

3

Regional Parameters version [ⓘ] *

[Show advanced activation, LoRaWAN class and cluster settings](#) ▾

DevEUI [ⓘ] *
 0/50 used 4

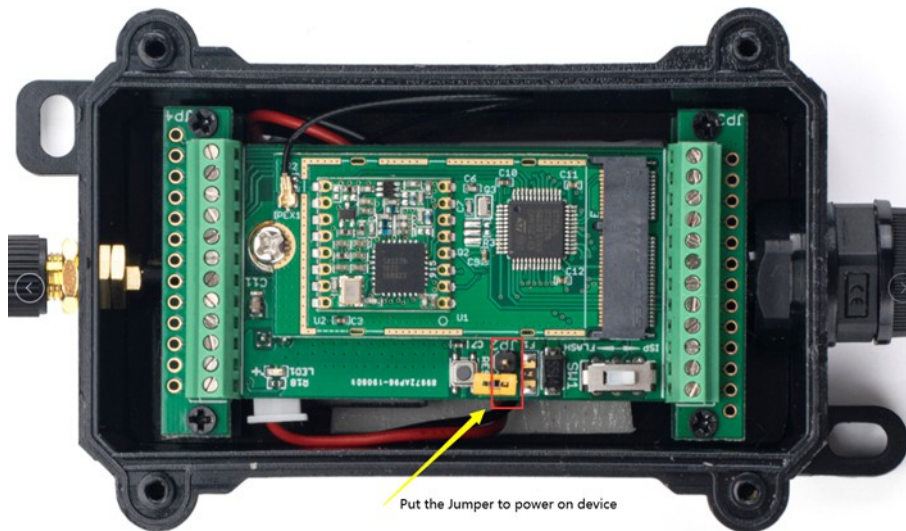
AppEUI [ⓘ] *
 5

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





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	

Back

Next

 **DATA CAKE**

Fleet > LDS03A

LDS03A

Serial Number
9955663322441140

Last update
Never

 Dashboard

 History

 Downlinks

 Configuration

 Debug

 Rules

 Permissions

General Configuration

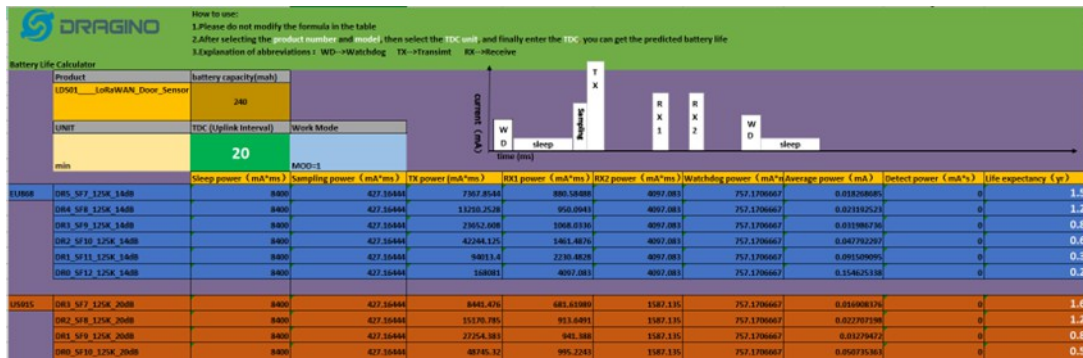
Device Name
LDS03A

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 serialize(payload) {
2   var arr=bytesToHex(payload.toString().replace(/ /g, ''));
3   var arr=arr.split('').map(function(v) { return v.charCodeAt(0).toString(16) }).join('');
4   return arr;
5 }
6 function deserialize(payload) {
7   var arr=hexToBytes(payload);
8   return arr;
9 }
10 function getPayload(payload) {
11   if(payload === null) {
12     return null;
13   }
14   if(payload === '') {
15     return null;
16   }
17   if(payload === '0') {
18     return null;
19   }
20   if(payload === '1') {
21     return null;
22   }
23   if(payload === '2') {
24     return null;
25   }
26   if(payload === '3') {
27     return null;
28   }
29   if(payload === '4') {
30     return null;
31   }
32   if(payload === '5') {
33     return null;
34   }
35   if(payload === '6') {
36     return null;
37   }
38   if(payload === '7') {
39     return null;
40   }
41   if(payload === '8') {
42     return null;
43   }
44   if(payload === '9') {
45     return null;
46   }
47   if(payload === 'a') {
48     return null;
49   }
50   if(payload === 'b') {
51     return null;
52   }
53   if(payload === 'c') {
54     return null;
55   }
56   if(payload === 'd') {
57     return null;
58   }
59   if(payload === 'e') {
60     return null;
61   }
62   if(payload === 'f') {
63     return null;
64   }
65   if(payload === '00') {
66     return null;
67   }
68   if(payload === '01') {
69     return null;
70   }
71   if(payload === '02') {
72     return null;
73   }
74   if(payload === '03') {
75     return null;
76   }
77   if(payload === '04') {
78     return null;
79   }
80   if(payload === '05') {
81     return null;
82   }
83   if(payload === '06') {
84     return null;
85   }
86   if(payload === '07') {
87     return null;
88   }
89   if(payload === '08') {
90     return null;
91   }
92   if(payload === '09') {
93     return null;
94   }
95   if(payload === '0a') {
96     return null;
97   }
98   if(payload === '0b') {
99     return null;
100  }
101  if(payload === '0c') {
102    return null;
103  }
104  if(payload === '0d') {
105    return null;
106  }
107  if(payload === '0e') {
108    return null;
109  }
110  if(payload === '0f') {
111    return null;
112  }
113  if(payload === '10') {
114    return null;
115  }
116  if(payload === '11') {
117    return null;
118  }
119  if(payload === '12') {
120    return null;
121  }
122  if(payload === '13') {
123    return null;
124  }
125  if(payload === '14') {
126    return null;
127  }
128  if(payload === '15') {
129    return null;
130  }
131  if(payload === '16') {
132    return null;
133  }
134  if(payload === '17') {
135    return null;
136  }
137  if(payload === '18') {
138    return null;
139  }
140  if(payload === '19') {
141    return null;
142  }
143  if(payload === '1a') {
144    return null;
145  }
146  if(payload === '1b') {
147    return null;
148  }
149  if(payload === '1c') {
150    return null;
151  }
152  if(payload === '1d') {
153    return null;
154  }
155  if(payload === '1e') {
156    return null;
157  }
158  if(payload === '1f') {
159    return null;
160  }
161  if(payload === '20') {
162    return null;
163  }
164  if(payload === '21') {
165    return null;
166  }
167  if(payload === '22') {
168    return null;
169  }
170  if(payload === '23') {
171    return null;
172  }
173  if(payload === '24') {
174    return null;
175  }
176  if(payload === '25') {
177    return null;
178  }
179  if(payload === '26') {
180    return null;
181  }
182  if(payload === '27') {
183    return null;
184  }
185  if(payload === '28') {
186    return null;
187  }
188  if(payload === '29') {
189    return null;
190  }
191  if(payload === '2a') {
192    return null;
193  }
194  if(payload === '2b') {
195    return null;
196  }
197  if(payload === '2c') {
198    return null;
199  }
200  if(payload === '2d') {
201    return null;
202  }
203  if(payload === '2e') {
204    return null;
205  }
206  if(payload === '2f') {
207    return null;
208  }
209  if(payload === '30') {
210    return null;
211  }
212  if(payload === '31') {
213    return null;
214  }
215  if(payload === '32') {
216    return null;
217  }
218  if(payload === '33') {
219    return null;
220  }
221  if(payload === '34') {
222    return null;
223  }
224  if(payload === '35') {
225    return null;
226  }
227  if(payload === '36') {
228    return null;
229  }
230  if(payload === '37') {
231    return null;
232  }
233  if(payload === '38') {
234    return null;
235  }
236  if(payload === '39') {
237    return null;
238  }
239  if(payload === '3a') {
240    return null;
241  }
242  if(payload === '3b') {
243    return null;
244  }
245  if(payload === '3c') {
246    return null;
247  }
248  if(payload === '3d') {
249    return null;
250  }
251  if(payload === '3e') {
252    return null;
253  }
254  if(payload === '3f') {
255    return null;
256  }
257  if(payload === '40') {
258    return null;
259  }
260  if(payload === '41') {
261    return null;
262  }
263  if(payload === '42') {
264    return null;
265  }
266  if(payload === '43') {
267    return null;
268  }
269  if(payload === '44') {
270    return null;
271  }
272  if(payload === '45') {
273    return null;
274  }
275  if(payload === '46') {
276    return null;
277  }
278  if(payload === '47') {
279    return null;
280  }
281  if(payload === '48') {
282    return null;
283  }
284  if(payload === '49') {
285    return null;
286  }
287  if(payload === '4a') {
288    return null;
289  }
290  if(payload === '4b') {
291    return null;
292  }
293  if(payload === '4c') {
294    return null;
295  }
296  if(payload === '4d') {
297    return null;
298  }
299  if(payload === '4e') {
300    return null;
301  }
302  if(payload === '4f') {
303    return null;
304  }
305  if(payload === '50') {
306    return null;
307  }
308  if(payload === '51') {
309    return null;
310  }
311  if(payload === '52') {
312    return null;
313  }
314  if(payload === '53') {
315    return null;
316  }
317  if(payload === '54') {
318    return null;
319  }
320  if(payload === '55') {
321    return null;
322  }
323  if(payload === '56') {
324    return null;
325  }
326  if(payload === '57') {
327    return null;
328  }
329  if(payload === '58') {
330    return null;
331  }
332  if(payload === '59') {
333    return null;
334  }
335  if(payload === '5a') {
336    return null;
337  }
338  if(payload === '5b') {
339    return null;
340  }
341  if(payload === '5c') {
342    return null;
343  }
344  if(payload === '5d') {
345    return null;
346  }
347  if(payload === '5e') {
348    return null;
349  }
350  if(payload === '5f') {
351    return null;
352  }
353  if(payload === '60') {
354    return null;
355  }
356  if(payload === '61') {
357    return null;
358  }
359  if(payload === '62') {
360    return null;
361  }
362  if(payload === '63') {
363    return null;
364  }
365  if(payload === '64') {
366    return null;
367  }
368  if(payload === '65') {
369    return null;
370  }
371  if(payload === '66') {
372    return null;
373  }
374  if(payload === '67') {
375    return null;
376  }
377  if(payload === '68') {
378    return null;
379  }
380  if(payload === '69') {
381    return null;
382  }
383  if(payload === '6a') {
384    return null;
385  }
386  if(payload === '6b') {
387    return null;
388  }
389  if(payload === '6c') {
390    return null;
391  }
392  if(payload === '6d') {
393    return null;
394  }
395  if(payload === '6e') {
396    return null;
397  }
398  if(payload === '6f') {
399    return null;
400  }
401  if(payload === '70') {
402    return null;
403  }
404  if(payload === '71') {
405    return null;
406  }
407  if(payload === '72') {
408    return null;
409  }
410  if(payload === '73') {
411    return null;
412  }
413  if(payload === '74') {
414    return null;
415  }
416  if(payload === '75') {
417    return null;
418  }
419  if(payload === '76') {
420    return null;
421  }
422  if(payload === '77') {
423    return null;
424  }
425  if(payload === '78') {
426    return null;
427  }
428  if(payload === '79') {
429    return null;
430  }
429  if(payload === '7a') {
431    return null;
432  }
433  if(payload === '7b') {
434    return null;
435  }
436  if(payload === '7c') {
437    return null;
438  }
439  if(payload === '7d') {
440    return null;
441  }
442  if(payload === '7e') {
443    return null;
444  }
445  if(payload === '7f') {
446    return null;
447  }
448  if(payload === '80') {
449    return null;
450  }
451  if(payload === '81') {
452    return null;
453  }
454  if(payload === '82') {
455    return null;
456  }
457  if(payload === '83') {
458    return null;
459  }
460  if(payload === '84') {
461    return null;
462  }
461  if(payload === '85') {
463    return null;
464  }
465  if(payload === '86') {
466    return null;
467  }
466  if(payload === '87') {
468    return null;
469  }
467  if(payload === '88') {
470    return null;
471  }
468  if(payload === '89') {
472    return null;
473  }
469  if(payload === '8a') {
474    return null;
475  }
470  if(payload === '8b') {
476    return null;
477  }
471  if(payload === '8c') {
478    return null;
479  }
472  if(payload === '8d') {
480    return null;
481  }
473  if(payload === '8e') {
482    return null;
483  }
474  if(payload === '8f') {
484    return null;
485  }
475  if(payload === '90') {
486    return null;
487  }
476  if(payload === '91') {
488    return null;
489  }
477  if(payload === '92') {
490    return null;
491  }
478  if(payload === '93') {
492    return null;
493  }
479  if(payload === '94') {
494    return null;
495  }
480  if(payload === '95') {
496    return null;
497  }
481  if(payload === '96') {
498    return null;
499  }
482  if(payload === '97') {
500    return null;
501  }
483  if(payload === '98') {
502    return null;
503  }
484  if(payload === '99') {
504    return null;
505  }
485  if(payload === '9a') {
506    return null;
507  }
486  if(payload === '9b') {
508    return null;
509  }
487  if(payload === '9c') {
510    return null;
511  }
488  if(payload === '9d') {
512    return null;
513  }
489  if(payload === '9e') {
514    return null;
515  }
490  if(payload === '9f') {
516    return null;
517  }
491  if(payload === 'a0') {
518    return null;
519  }
492  if(payload === 'a1') {
520    return null;
521  }
493  if(payload === 'a2') {
522    return null;
523  }
494  if(payload === 'a3') {
524    return null;
525  }
495  if(payload === 'a4') {
526    return null;
527  }
496  if(payload === 'a5') {
528    return null;
529  }
497  if(payload === 'a6') {
530    return null;
531  }
498  if(payload === 'a7') {
532    return null;
533  }
499  if(payload === 'a8') {
534    return null;
535  }
500  if(payload === 'a9') {
536    return null;
537  }
501  if(payload === 'aa') {
538    return null;
539  }
502  if(payload === 'ab') {
540    return null;
541  }
503  if(payload === 'ac') {
542    return null;
543  }
504  if(payload === 'ad') {
544    return null;
545  }
505  if(payload === 'ae') {
546    return null;
547  }
506  if(payload === 'af') {
548    return null;
549  }
507  if(payload === 'b0') {
550    return null;
551  }
508  if(payload === 'b1') {
552    return null;
553  }
509  if(payload === 'b2') {
554    return null;
555  }
510  if(payload === 'b3') {
556    return null;
557  }
511  if(payload === 'b4') {
558    return null;
559  }
512  if(payload === 'b5') {
560    return null;
561  }
513  if(payload === 'b6') {
562    return null;
563  }
514  if(payload === 'b7') {
564    return null;
565  }
515  if(payload === 'b8') {
566    return null;
567  }
516  if(payload === 'b9') {
568    return null;
569  }
517  if(payload === 'ba') {
570    return null;
571  }
518  if(payload === 'bb') {
572    return null;
573  }
519  if(payload === 'bc') {
574    return null;
575  }
520  if(payload === 'bd') {
576    return null;
577  }
521  if(payload === 'be') {
578    return null;
579  }
522  if(payload === 'bf') {
580    return null;
581  }
523  if(payload === 'c0') {
582    return null;
583  }
524  if(payload === 'c1') {
584    return null;
585  }
525  if(payload === 'c2') {
586    return null;
587  }
526  if(payload === 'c3') {
588    return null;
589  }
527  if(payload === 'c4') {
590    return null;
591  }
528  if(payload === 'c5') {
592    return null;
593  }
529  if(payload === 'c6') {
594    return null;
595  }
530  if(payload === 'c7') {
596    return null;
597  }
531  if(payload === 'c8') {
598    return null;
599  }
532  if(payload === 'c9') {
600    return null;
601  }
533  if(payload === 'ca') {
602    return null;
603  }
534  if(payload === 'cb') {
604    return null;
605  }
535  if(payload === 'cc') {
606    return null;
607  }
536  if(payload === 'cd') {
608    return null;
609  }
537  if(payload === 'ce') {
610    return null;
611  }
538  if(payload === 'cf') {
612    return null;
613  }
539  if(payload === 'd0') {
614    return null;
615  }
540  if(payload === 'd1') {
616    return null;
617  }
541  if(payload === 'd2') {
618    return null;
619  }
542  if(payload === 'd3') {
620    return null;
621  }
543  if(payload === 'd4') {
622    return null;
623  }
544  if(payload === 'd5') {
624    return null;
625  }
545  if(payload === 'd6') {
626    return null;
627  }
546  if(payload === 'd7') {
628    return null;
629  }
547  if(payload === 'd8') {
630    return null;
631  }
548  if(payload === 'd9') {
632    return null;
633  }
549  if(payload === 'da') {
634    return null;
635  }
550  if(payload === 'db') {
636    return null;
637  }
551  if(payload === 'dc') {
638    return null;
639  }
552  if(payload === 'dd') {
640    return null;
641  }
553  if(payload === 'de') {
642    return null;
643  }
554  if(payload === 'df') {
644    return null;
645  }
555  if(payload === 'e0') {
646    return null;
647  }
556  if(payload === 'e1') {
648    return null;
649  }
557  if(payload === 'e2') {
650    return null;
651  }
558  if(payload === 'e3') {
652    return null;
653  }
559  if(payload === 'e4') {
654    return null;
655  }
560  if(payload === 'e5') {
656    return null;
657  }
561  if(payload === 'e6') {
658    return null;
659  }
562  if(payload === 'e7') {
660    return null;
661  }
563  if(payload === 'e8') {
662    return null;
663  }
564  if(payload === 'e9') {
664    return null;
665  }
565  if(payload === 'ea') {
666    return null;
667  }
566  if(payload === 'eb') {
668    return null;
669  }
567  if(payload === 'ec') {
670    return null;
671  }
568  if(payload === 'ed') {
672    return null;
673  }
569  if(payload === 'ee') {
674    return null;
675  }
570  if(payload === 'ef') {
676    return null;
677  }
571  if(payload === 'f0') {
678    return null;
679  }
572  if(payload === 'f1') {
680    return null;
681  }
573  if(payload === 'f2') {
682    return null;
683  }
574  if(payload === 'f3') {
684    return null;
685  }
575  if(payload === 'f4') {
686    return null;
687  }
576  if(payload === 'f5') {
688    return null;
689  }
577  if(payload === 'f6') {
690    return null;
691  }
578  if(payload === 'f7') {
692    return null;
693  }
579  if(payload === 'f8') {
694    return null;
695  }
580  if(payload === 'f9') {
696    return null;
697  }
581  if(payload === 'fa') {
698    return null;
699  }
582  if(payload === 'fb') {
700    return null;
701  }
583  if(payload === 'fc') {
702    return null;
703  }
584  if(payload === 'fd') {
704    return null;
705  }
585  if(payload === 'fe') {
706    return null;
707  }
586  if(payload === 'ff') {
708    return null;
709  }
587  if(payload === '0000') {
710    return null;
711  }
588  if(payload === '0001') {
712    return null;
713  }
589  if(payload === '0002') {
714    return null;
715  }
590  if(payload === '0003') {
716    return null;
717  }
591  if(payload === '0004') {
718    return null;
719  }
592  if(payload === '0005') {
720    return null;
721  }
593  if(payload === '0006') {
722    return null;
723  }
594  if(payload === '0007') {
724    return null;
725  }
595  if(payload === '0008') {
726    return null;
727  }
596  if(payload === '0009') {
728    return null;
729  }
597  if(payload === '000a') {
730    return null;
731  }
598  if(payload === '000b') {
732    return null;
733  }
599  if(payload === '000c') {
734    return null;
735  }
600  if(payload === '000d') {
736    return null;
737  }
601  if(payload === '000e') {
738    return null;
739  }
602  if(payload === '000f') {
740    return null;
741  }
603  if(payload === '0010') {
742    return null;
743  }
604  if(payload === '0011') {
744    return null;
745  }
605  if(payload === '0012') {
746    return null;
747  }
606  if(payload === '0013') {
748    return null;
749  }
607  if(payload === '0014') {
750    return null;
751  }
608  if(payload === '0015') {
752    return null;
753  }
609  if(payload === '0016') {
754    return null;
755  }
610  if(payload === '0017') {
756    return null;
757  }
611  if(payload === '0018') {
758    return null;
759  }
612  if(payload === '0019') {
760    return null;
761  }
613  if(payload === '001a') {
762    return null;
763  }
614  if(payload === '001b') {
764    return null;
765  }
615  if(payload === '001c') {
766    return null;
767  }
616  if(payload === '001d') {
768    return null;
769  }
617  if(payload === '001e') {
770    return null;
771  }
618  if(payload === '001f') {
772    return null;
773  }
619  if(payload === '0020') {
774    return null;
775  }
620  if(payload === '0021') {
776    return null;
777  }
621  if(payload === '0022') {
778    return null;
779  }
622  if(payload === '0023') {
780    return null;
781  }
623  if(payload === '0024') {
782    return null;
783  }
624  if(payload === '0025') {
784    return null;
785  }
625  if(payload === '0026') {
786    return null;
787  }
626  if(payload === '0027') {
788    return null;
789  }
627  if(payload === '0028') {
790    return null;
791  }
628  if(payload === '0029') {
792    return null;
793  }
629  if(payload === '002a') {
794    return null;
795  }
630  if(payload === '002b') {
796    return null;
797  }
631  if(payload === '002c') {
798    return null;
799  }
632  if(payload === '002d') {
800    return null;
801  }
633  if(payload === '002e') {
802    return null;
803  }
634  if(payload === '002f') {
804    return null;
805  }
635  if(payload === '0030') {
806    return null;
807  }
636  if(payload === '0031') {
808    return null;
809  }
637  if(payload === '0032') {
810    return null;
811  }
638  if(payload === '0033') {
812    return null;
813  }
639  if(payload === '0034') {
814    return null;
815  }
640  if(payload === '0035') {
816    return null;
817  }
641  if(payload === '0036') {
818    return null;
819  }
642  if(payload === '0037') {
820    return null;
821  }
643  if(payload === '0038') {
822    return null;
823  }
644  if(payload === '0039') {
824    return null;
825  }
645  if(payload === '003a') {
826    return null;
827  }
646  if(payload === '003b') {
828    return null;
829  }
647  if(payload === '003c') {
830    return null;
831  }
648  if(payload === '003d') {
832    return null;
833  }
649  if(payload === '003e') {
834    return null;
835  }
650  if(payload === '003f') {
836    return null;
837  }
651  if(payload === '0040') {
838    return null;
839  }
652  if(payload === '0041') {
840    return null;
841  }
653  if(payload === '0042') {
842    return null;
843  }
654  if(payload === '0043') {
844    return null;
845  }
655  if(payload === '0044') {
846    return null;
847  }
656  if(payload === '0045') {
848    return null;
849  }
657  if(payload === '0046') {
850    return null;
851  }
658  if(payload === '0047') {
852    return null;
853  }
659  if(payload === '0048') {
854    return null;
855  }
660  if(payload === '0049') {
856    return null;
857  }
661  if(payload === '004a') {
858    return null;
859  }
662  if(payload === '004b') {
860    return null;
861  }
663  if(payload === '004c') {
862    return null;
863  }
664  if(payload === '004d') {
864    return null;
865  }
665  if(payload === '004e') {
866    return null;
867  }
666  if(payload === '004f') {
868    return null;
869  }
667  if(payload === '0050') {
870    return null;
871  }
668  if(payload === '0051') {
872    return null;
873  }
669  if(payload === '0052') {
874    return null;
875  }
670  if(payload === '0053') {
876    return null;
877  }
671  if(payload === '0054') {
878    return null;
879  }
672  if(payload === '0055') {
880    return null;
881  }
673  if(payload === '0056') {
882    return null;
883  }
674  if(payload === '0057') {
884    return null;
885  }
675  if(payload === '0058') {
886    return null;
887  }
676  if(payload === '0059') {
888    return null;
889  }
677  if(payload === '005a') {
890    return null;
891  }
678  if(payload === '005b') {
892    return null;
893  }
679  if(payload === '005c') {
894    return null;
895  }
680  if(payload === '005d') {
896    return null;
897  }
681  if(payload === '005e') {
898    return null;
899  }
682  if(payload === '005f') {
900    return null;
901  }
683  if(payload === '0060') {
902    return null;
903  }
684  if(payload === '0061') {
904    return null;
905  }
685  if(payload === '0062') {
906    return null;
907  }
686  if(payload === '0063') {
908    return null;
909  }
687  if(payload === '0064') {
910    return null;
911  }
688  if(payload === '0065') {
912    return null;
913  }
689  if(payload === '0066') {
914    return null;
915  }
690  if(payload === '0067') {
916    return null;
917  }
691  if(payload === '0068') {
918    return null;
919  }
692  if(payload === '0069') {
920    return null;
921  }
693  if(payload === '006a') {
922    return null;
923  }
694  if(payload === '006b') {
924    return null;
925  }
695  if(payload === '006c') {
926    return null;
927  }
696  if(payload === '006d') {
928    return null;
929  }
697  if(payload === '006e') {
930    return null;
931  }
698  if(payload === '006f') {
932    return null;
933  }
699  if(payload === '0070') {
934    return null;
935  }
700  if(payload === '0071') {
936    return null;
937  }
701  if(payload === '0072') {
938    return null;
939  }
702  if(payload === '0073') {
940    return null;
941  }
703  if(payload === '0074') {
942    return null;
943  }
704  if(payload === '0075') {
944    return null;
945  }
705  if(payload === '0076') {
946    return null;
947  }
706  if(payload === '0077') {
948    return null;
949  }
707  if(payload === '0078') {
950    return null;
951  }
708  if(payload === '0079') {
952    return null;
953  }
709  if(payload === '007a') {
954    return null;
955  }
710  if(payload === '007b') {
956    return null;
957  }
711  if(payload === '007c') {
958    return null;
959  }
712  if(payload === '007d') {
960    return null;
961  }
713  if(payload === '007e') {
962    return null;
963  }
714  if(payload === '007f') {
964    return null;
965  }
715  if(payload === '0080') {
966    return null;
967  }
716  if(payload === '0081') {
968    return null;
969  }
717  if(payload === '0082') {
970    return null;
971  }
718  if(payload === '0083') {
972    return null;
973  }
719  if(payload === '0084') {
974    return null;
975  }
720  if(payload === '0085') {
976    return null;
977  }
721  if(payload === '0086') {
978    return null;
979  }
722  if(payload === '0087') {
980    return null;
981  }
723  if(payload === '0088') {
982    return null;
983  }
724  if(payload === '0089') {
984    return null;
985  }
725  if(payload === '008a') {
986    return null;
987  }
726  if(payload === '008b') {
988    return null;
989  }
727  if(payload === '008c') {
990    return null;
991  }
728  if(payload === '008d') {
992    return null;
993  }
729  if(payload === '00
```

Benutzerhandbuch für LoRaWAN/NB-IoT-Endknoten – LMS200
 -- LoRaWAN-Mikrowellenradar-Entfernungssensor
 Benutzerhandbuch





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

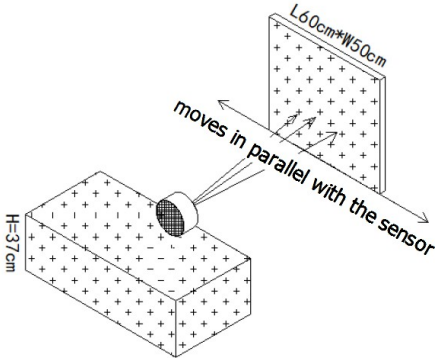
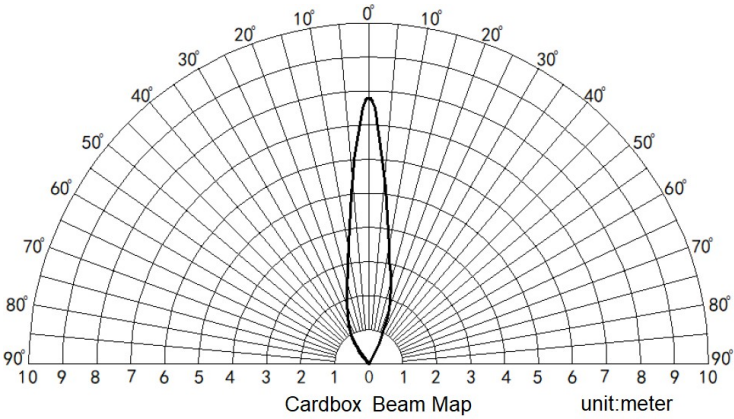
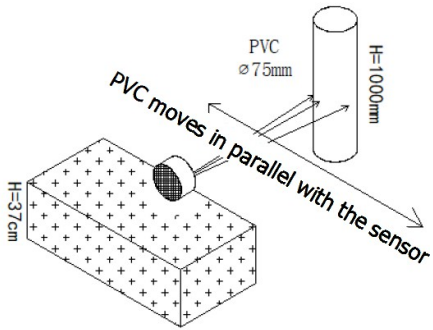
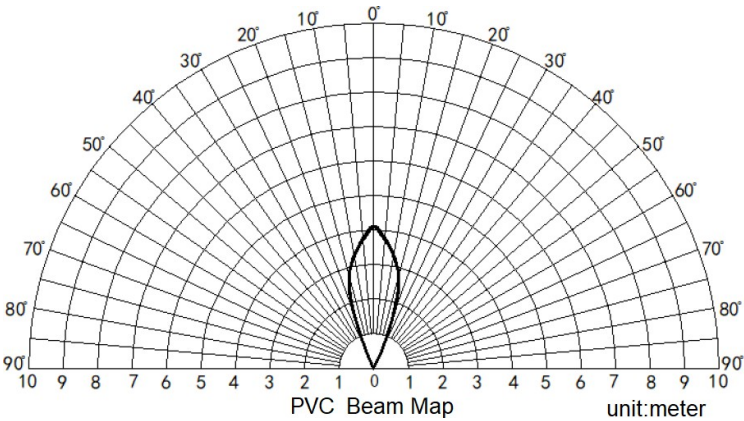
Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
Get Firmware Version Info	Any	26	2





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)



S

K

 **THE THINGS STACK**
Community Edition

 Overview

 Applications

 Gateways

 Orga

Add application

Owner*

davidhuang

▼

Application ID*

my-new-application

Application name

My new application

Description

Description for my new application

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

Create application

The screenshot displays the 'Register end device' page in a LoRaWAN management system. The page is divided into two main sections: '1. Select the end device' and '2. Enter registration data'. In the first section, the 'Brand' dropdown is set to 'Dragino Technology Co.,...' and the 'Model' dropdown is open, showing a list of models including LBT1, LDDS20, LDDS75, LDS01, LGT92, LHT65, LSE01, and LSN50-V2. A red arrow points to the 'Add end device' button in the top right corner. Another red arrow points to the 'From The LoRaWAN Device Repository' tab. A third red arrow points to the 'Brand' dropdown. A fourth red arrow points to the 'Model' dropdown menu. The '2. Enter registration data' section is currently empty, with a 'Register end device' button at the bottom.

2. Enter registration data

Frequency plan ⓘ *

Select...

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 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 ⓘ *

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 dev

DevEUI ⓘ *

The DevEUI is the unique identifier for this end device

AppKey ⓘ *

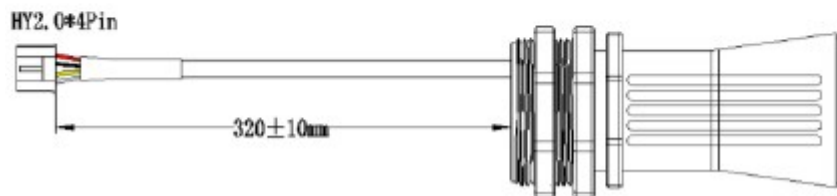
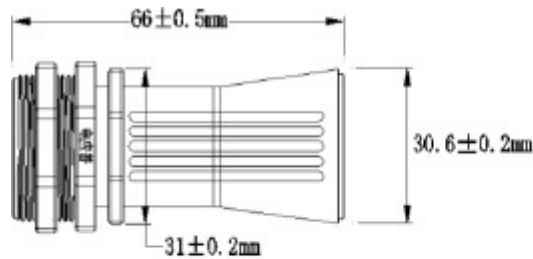
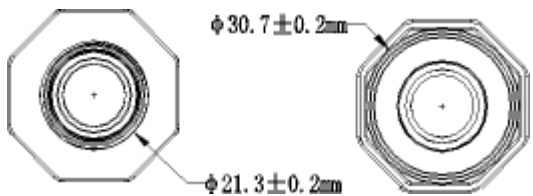
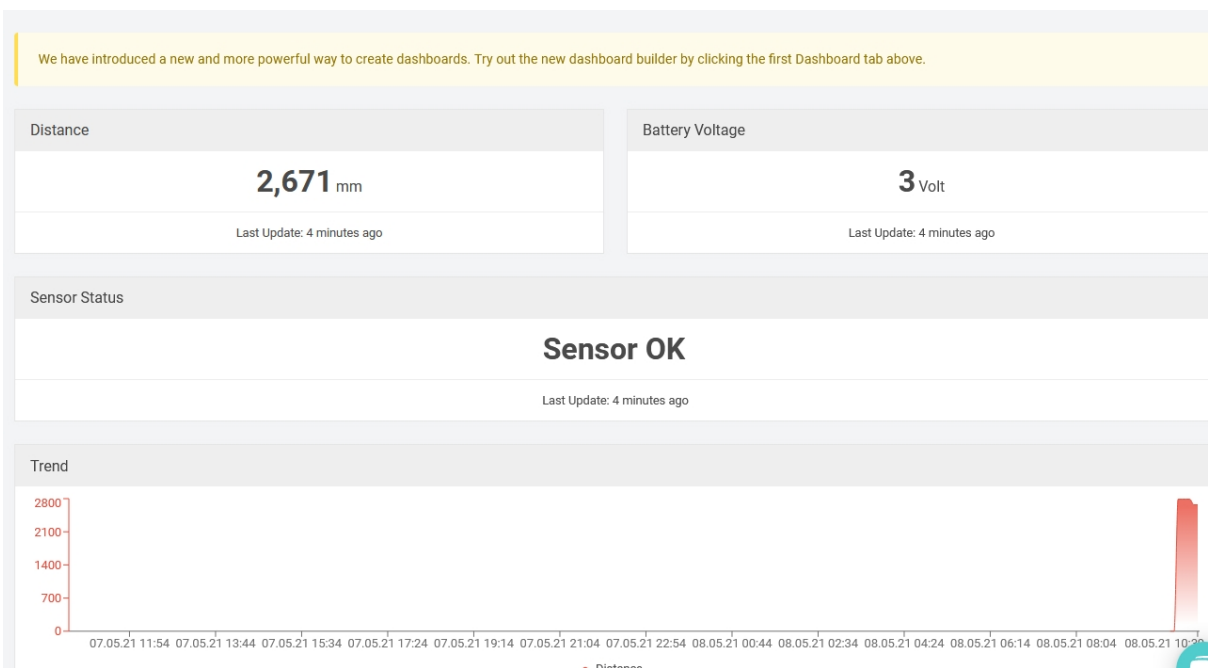
 

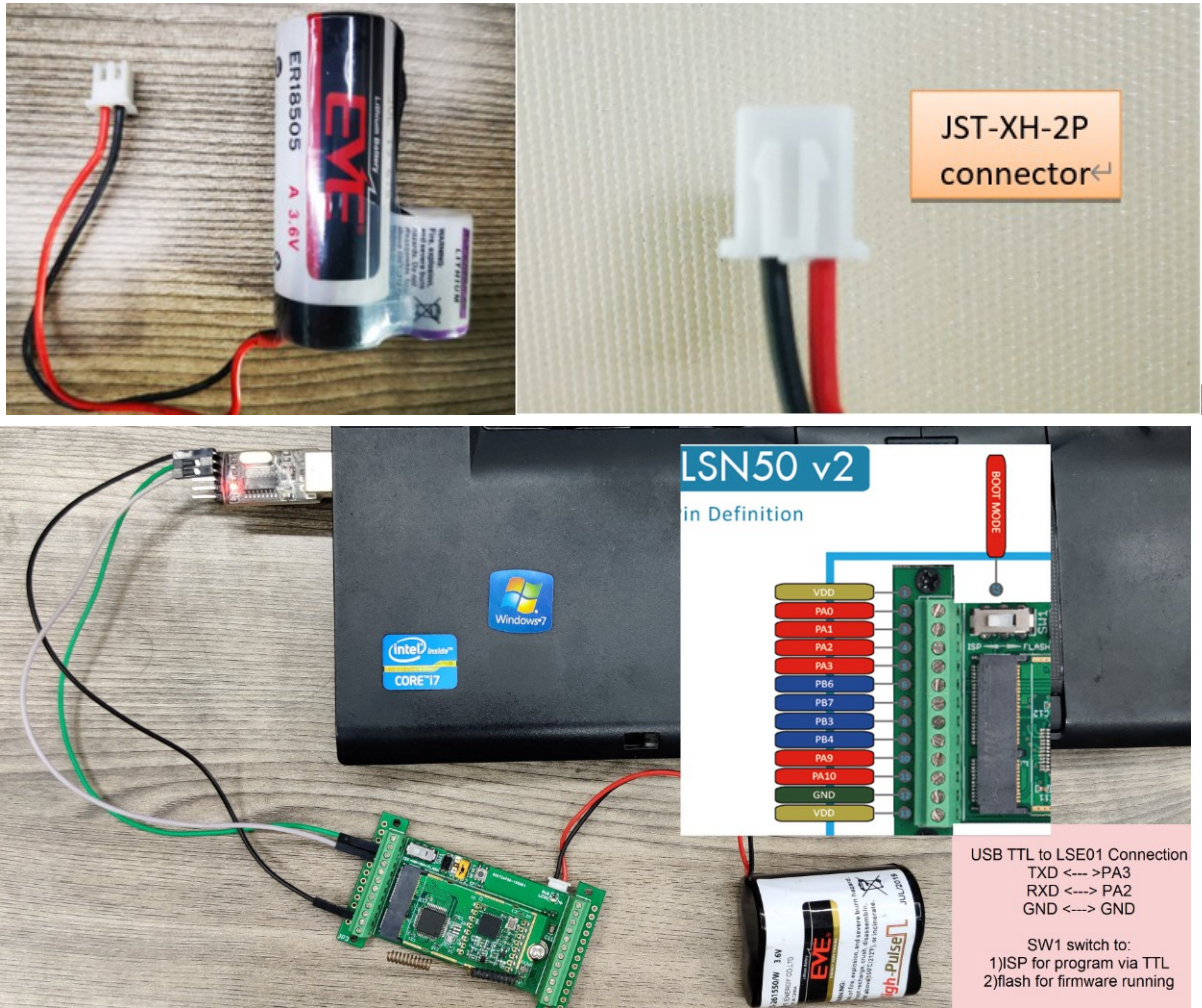
The root key to derive session keys to secure communication between the end device and the application

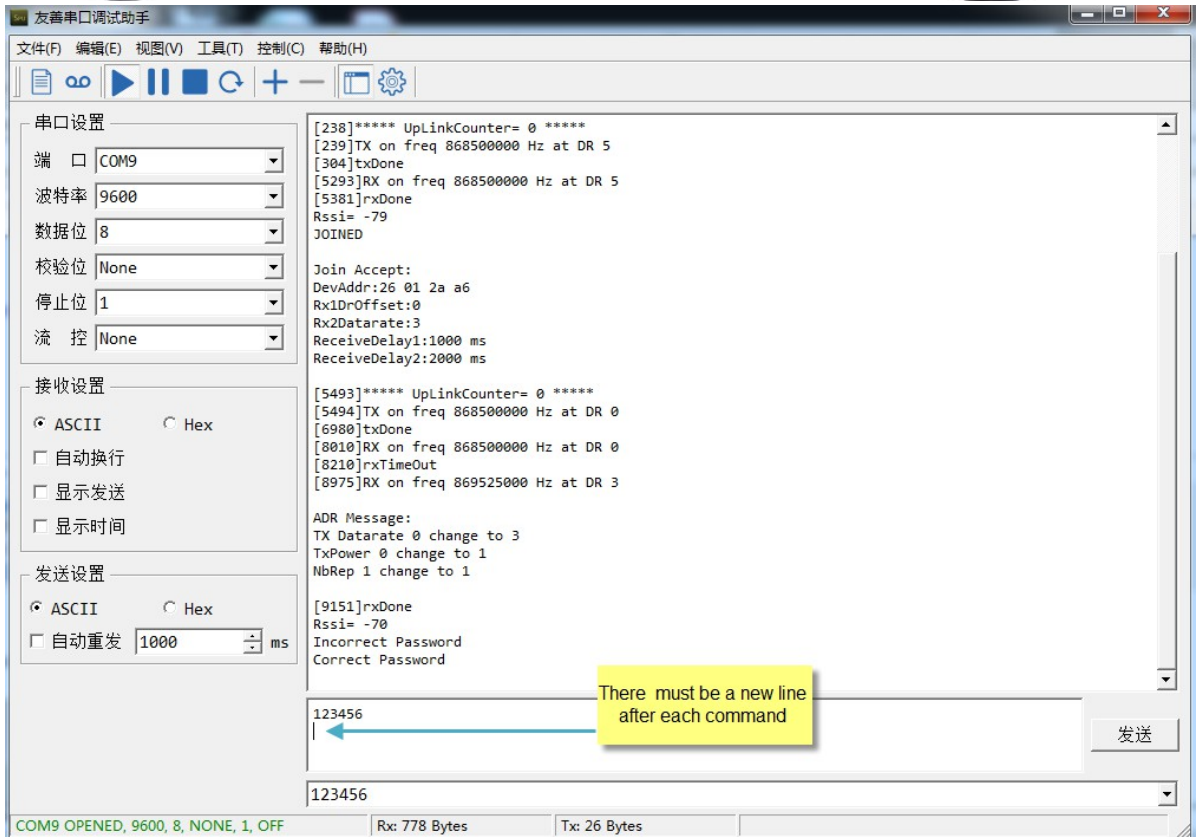
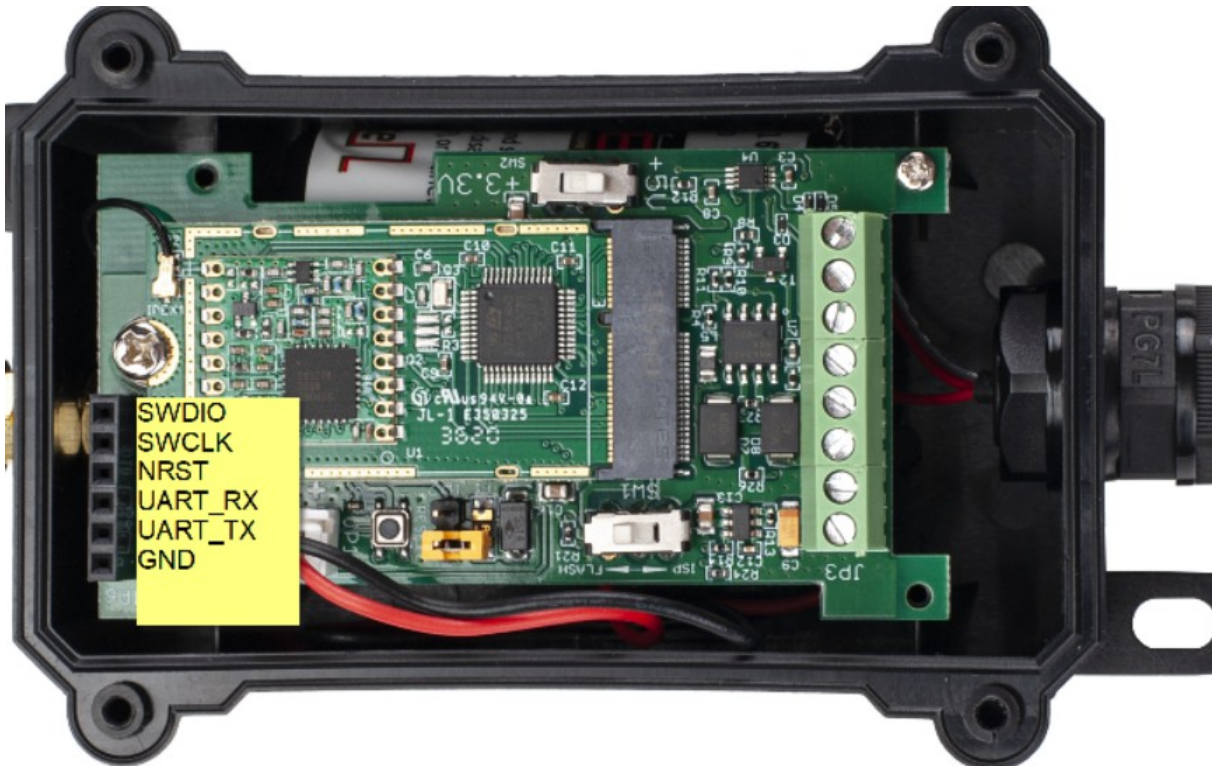
End device ID *

After registration





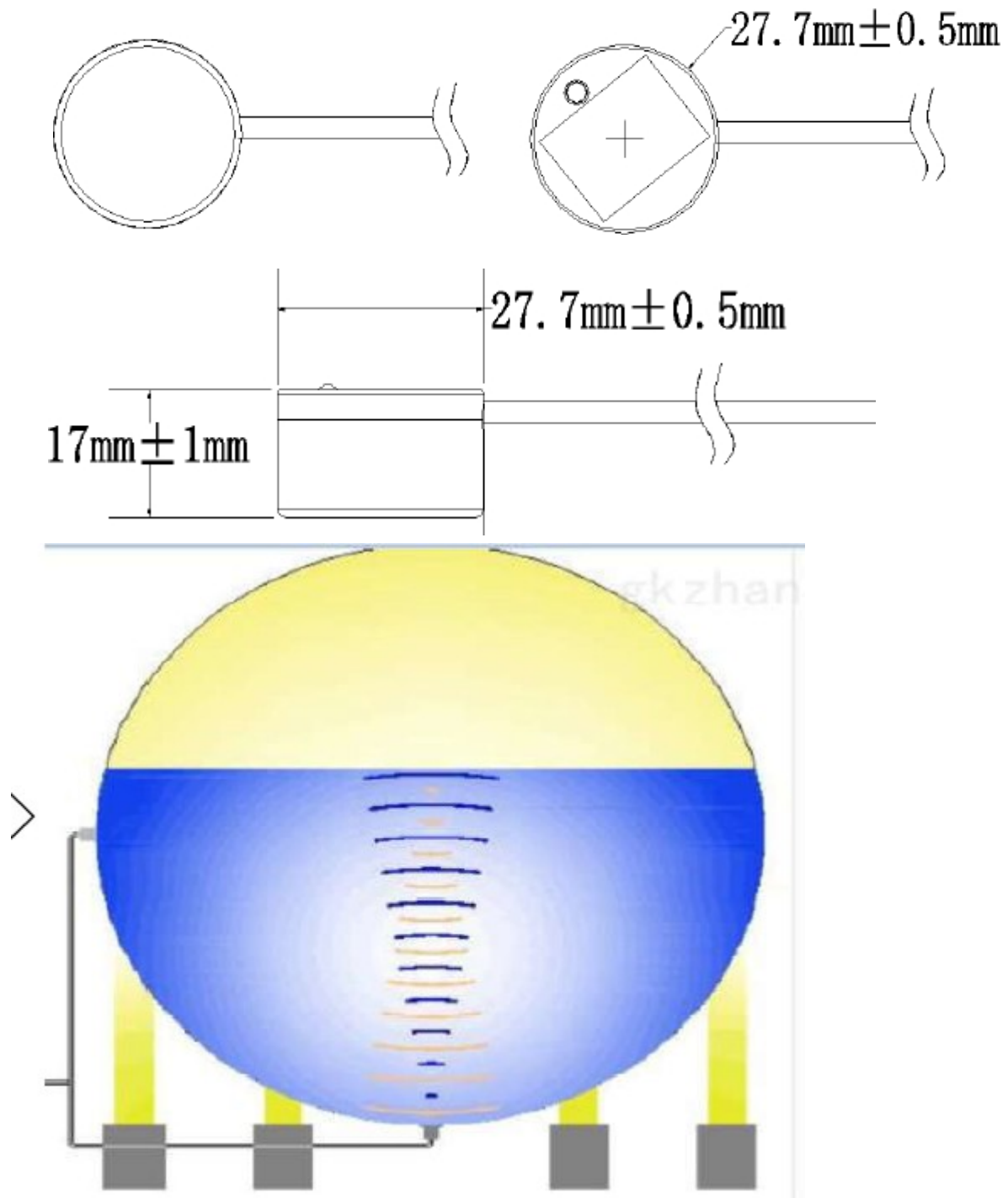


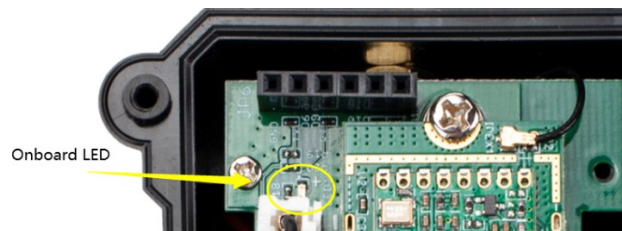


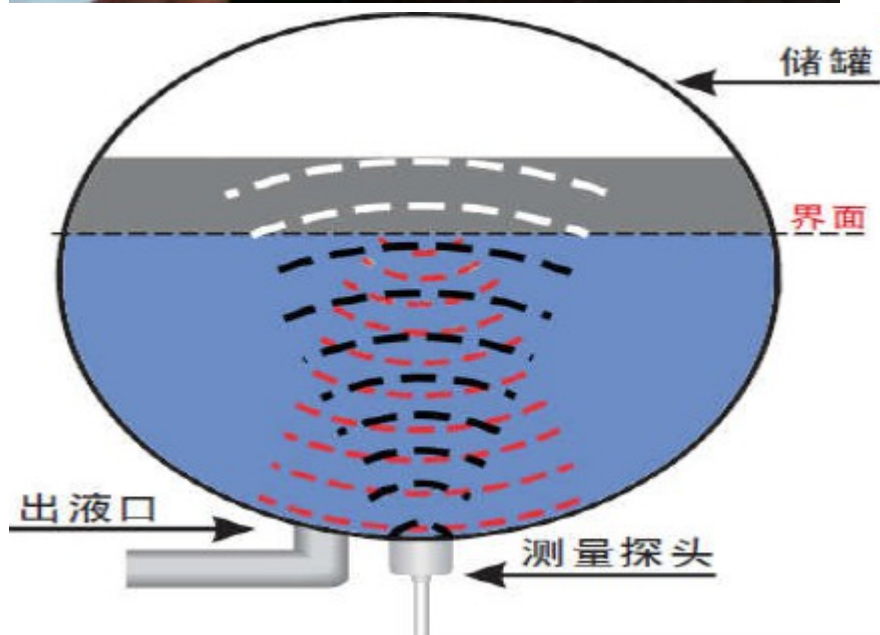
Command Example	Function	Response
AT+TDC=?	Show current transmit Interval	30000 OK the interval is 30000ms = 30s
AT+TDC=60000	Set Transmit Interval	OK Set transmit interval to 60000ms = 60 seconds

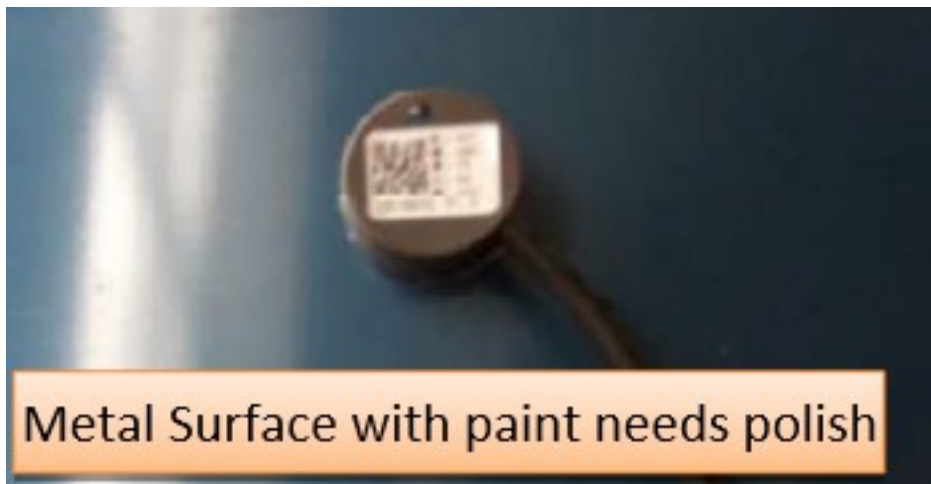
Command Example	Function	Response
AT+INTMOD=?	Show current interrupt mode	0 OK the mode is 0 = No interruption
AT+INTMOD=2	Set Transmit Interval 0- (Disable Interrupt), 1- (Trigger by rising and falling edge), 2- (Trigger by falling edge), 3- (Trigger by rising edge).	OK

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







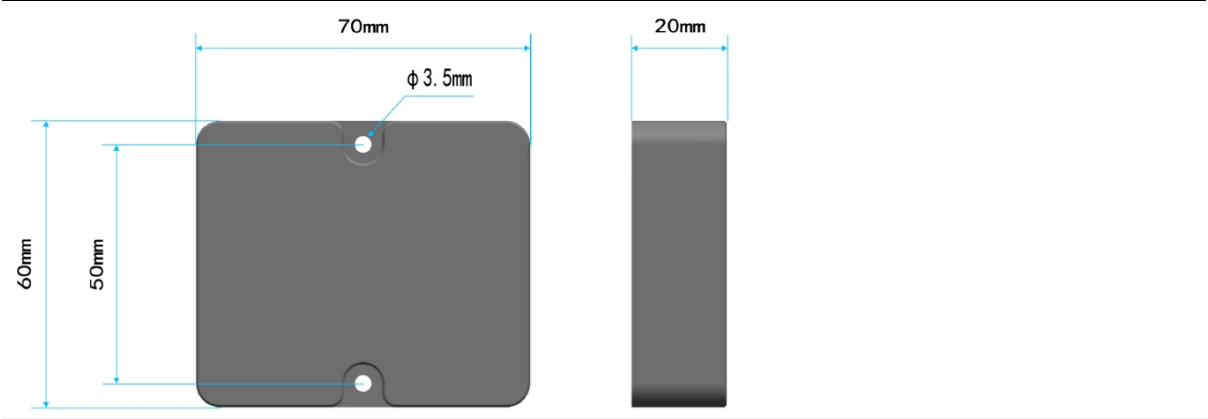




Downlink Control Type	FPort	Type Code	Downlink payload size(bytes)
TDC (Transmit Time Interval)	Any	01	4
RESET	Any	04	2
AT+CFM	Any	05	4
INTMOD	Any	06	4









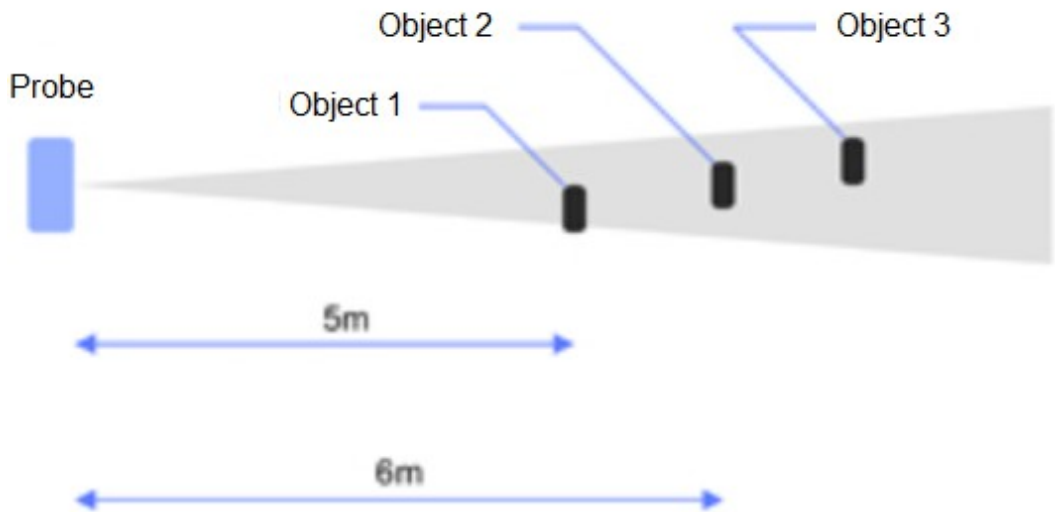
lmds200
 ID: lmds200

↑ 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: { Sat: 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	
Ⓜ 15:32:20	Accept join-request	

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



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

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

Benutzerhandbuch für LoRaWAN/NB-IoT-Endknoten – LMDS200

-- LoRaWAN-Mikrowellenradar-Entfernungssensor

Benutzerhandbuch

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(payload){
2   var aa=bytesToHex(payload)
3   var bb=bytesToHex(payload)
4   var cc=bytesToHex(payload)
5   var dd=bytesToHex(payload)
6   var ee=bytesToHex(payload)
7   var ff=bytesToHex(payload)
8   var string="aa"+"bb"+"cc"+"dd"+"ee"+"ff"
9   return string;
10 }
11
12 function getOff(c_num){
13   if(isNaN(c_num) < 0)
14     c_num = 0
15   return c_num;
16 }
17
18 function getOffDate(str){
19   var c_data
20   if(str < 0)
21     c_data = new Date(str)
22   else
23     c_data = new Date(str)
24   var c_year = c_data.getFullYear()
25   var c_month = c_data.getMonth()
26   var c_day = c_data.getDate()
27   var c_hour = c_data.getHours()
28   var c_min = c_data.getMinutes()
29   var c_sec = c_data.getSeconds()
30   var c_time = c_year+"-"+getOff(c_month)+"-"+getOff(c_day)+" "+getOff(c_hour)+"-"+getOff(c_min)+"-"+getOff(c_sec)
31   return c_time;
32 }
```

Payload

Payload

Port

1

Try Decoder

Output

console.log Output

Recognized measurements

Save

Fields

Fields describe the data the device will store.

NAME	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	...

DATA CAKE

Fleet > LDS03A

LDS03A

Serial Number

995566322441140

Last update

Never

Dashboard

History

Downlinks

Configuration

Debug

Rules

Permissions

Public Link

+ Add Widget

✓

Desktop

Mobile

More

Basics **Data** Appearance Gauge Timeframe

Field

Please Select

Please Select

BAT

DOOR_OPEN_TIMES

LAST_DOOR_OPEN_DURATION

Decimal Places

2

Cancel **✓ Save**

Edit Boolean Widget

New Boolean

Basics Appearance **Data**

Field

Please Select

Please Select

ALARM

DOOR_OPEN_STATUS

Cancel **✓ Save**

DATA CAKE

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 (6 minutes ago)

DOOR_OPEN_STATUS: [Red Dot]

LAST_DOOR_OPEN_DURATION: 0 (6 minutes ago)

ALARM: [Red Dot] (7 minutes ago)

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

LDS03A

Next

Network Server

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

1

☒		The Things Stack V3 TTN V3 / Things Industries	Uplinks	Downlinks
☐		The Things Network V2 The old Things Network	Uplinks	Downlinks
☐		Helium	Uplinks	Downlinks
☐		LORIoT	Uplinks	Downlinks
☐		Kerlink Wanesy	Uplinks	

Showing 1 to 5 of 8 results

Previous Next

2

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

DATA CAKE

Fleet > LDS03A

LDS03A

Serial Number
9955663322441140

Last update
Never

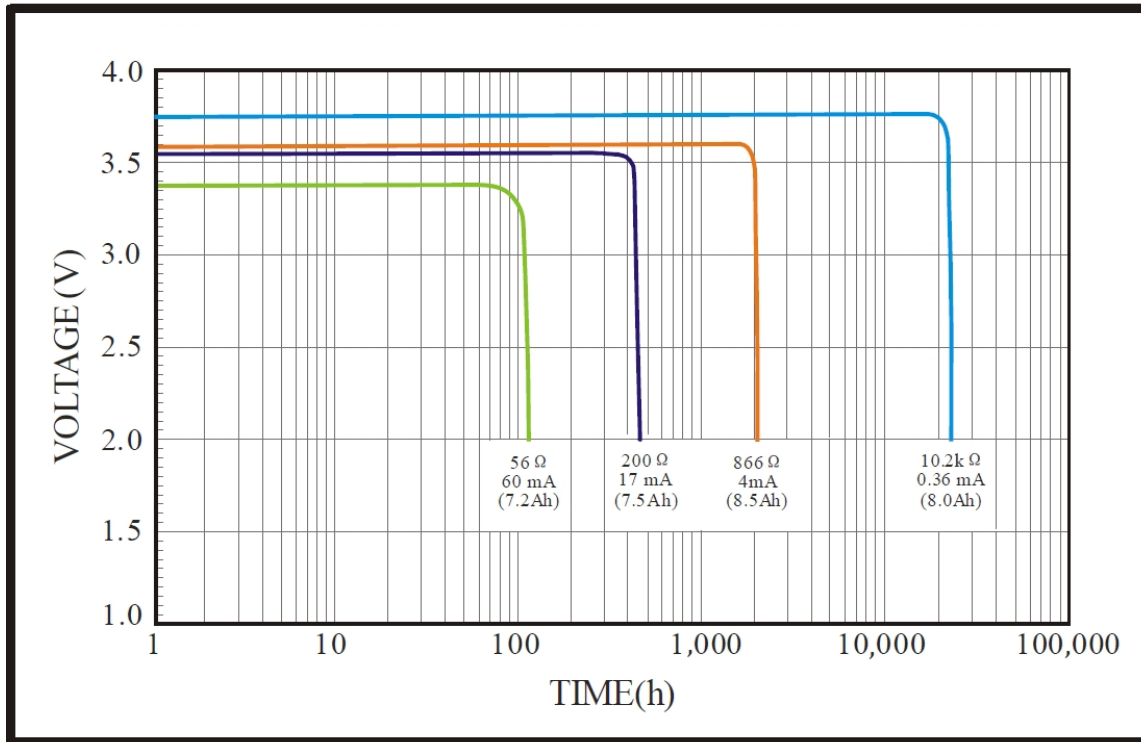
Dashboard History Downlinks **Configuration** Debug Rules Permissions

General Configuration

Device Name

LDS03A

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



How to use:

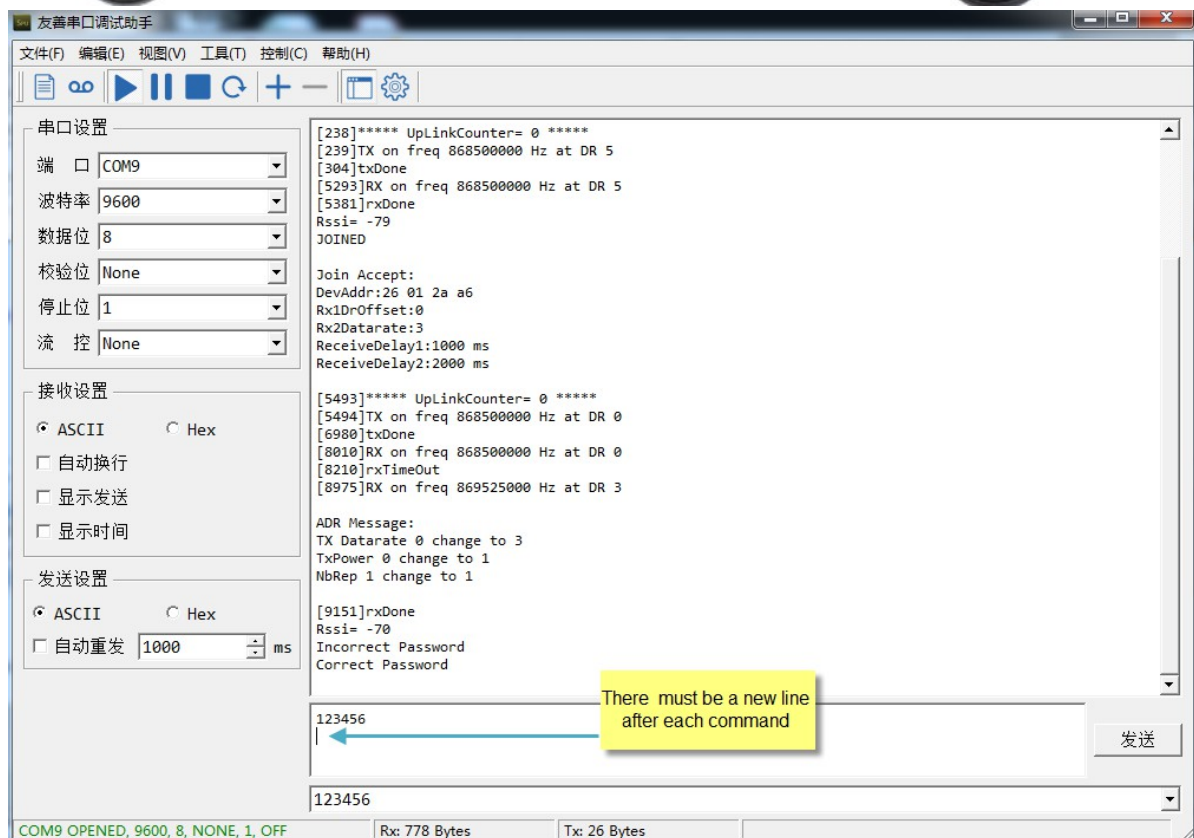
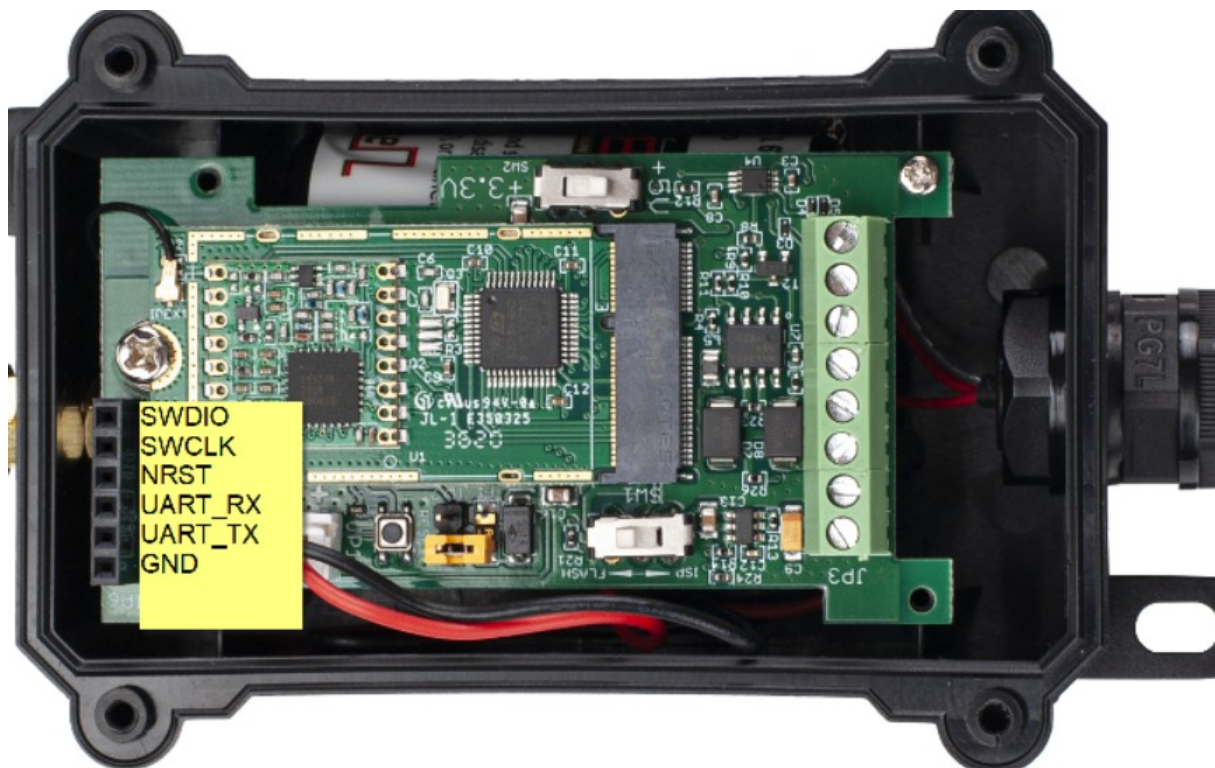
- Please do not modify the formula in the table
- After selecting the product number and model, then select the TDC unit, and finally enter the TDC, you can get the predicted battery life
- Explanation of abbreviations: WD→Watchdog TX→Transmit RX→Receive

Battery Life Calculator

Product	battery capacity(mah)	
LD501_LoRaWAN_Door_Sensor	240	
UNIT	TDC (Uplink Interval)	Work Mode
min	20	MOD=1

	Sleep power (mA*ms)	Sampling power (mA*ms)	TX power (mA*ms)	RX1 power (mA*ms)	RX2 power (mA*ms)	Watchdog power (mA*ms)	Average power (mA)	Detect power (mA*s)	life expectancy (yr)
FI968									
DR5_SF7_125K_14dB	8400	427.16444	7367.8544	880.58388	4097.083	757.1706667	0.01826885	0	1.5
DR4_SF8_125K_14dB	8400	427.16444	13210.2529	950.0943	4097.083	757.1706667	0.022192523	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	0.154625338	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



**Registration Key,
Please keep it safely.**

DEV EUI: A84041C161

APP EUI: A8404100C00

APP KEY: 7EC8A9C917386DFC5DBF73B

SN: LST2565



```
[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
```

Time	Type	Data preview	Verbose stream	Export as JSON	Pause	Clear
11:20:16	Schedule data downlink for transmi...	DevAddr: 26 08 95 42 Rx1 Delay: 5				
11:20:16	Forward uplink data message	DevAddr: 26 08 95 42 Payload: { Bat: 3.283, DALARM_count: 0, Distance_alarm: 0, Interrupt_alarm: 0, dis1: 1, dis2: 1 }	0C D3 00 01 00 01 00			FFC
11:20:16	Successfully processed data message	DevAddr: 26 08 95 42				
11:19:29	Schedule data downlink for transmi...	DevAddr: 26 08 95 42 Rx1 Delay: 5				
11:19:29	Forward uplink data message	DevAddr: 26 08 95 42 Payload: { Bat: 3.283, DALARM_count: 0, Distance_alarm: 0, Interrupt_alarm: 0, dis1: 1, dis2: 1 }	0C D3 00 01 00 01 00			FFC
11:19:29	Successfully processed data message	DevAddr: 26 08 95 42				
11:18:35	Forward uplink data message	DevAddr: 26 08 95 42 Payload: { DATALOG: "[3.283,260,383,NO,FALSE,0,2023-07-31 14:29:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:30:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:31:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:32:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:33:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:34:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:35:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:36:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:37:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:38:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:39:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:40:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:41:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:42:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:43:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:44:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:45:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:46:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:47:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:48:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:49:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:50:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:51:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:52:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:53:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:54:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:55:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:56:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:57:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:58:40],[3.283,260,383,NO,FALSE,0,2023-07-31 14:59:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:00:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:01:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:02:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:03:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:04:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:05:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:06:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:07:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:08:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:09:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:10:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:11:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:12:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:13:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:14:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:15:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:16:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:17:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:18:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:19:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:20:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:21:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:22:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:23:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:24:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:25:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:26:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:27:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:28:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:29:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:30:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:31:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:32:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:33:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:34:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:35:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:36:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:37:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:38:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:39:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:40:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:41:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:42:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:43:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:44:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:45:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:46:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:47:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:48:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:49:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:50:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:51:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:52:40],[3.283,260,383,NO,FALSE,0,2023-07-31 15:53:40],[3.283,260,383,NO,FALSE,0,2023-07-31 1				