



**AgroSense Soil Moisture Sensor  
LoRaWAN® Manual  
V1.0**

# Contents

<b>1 Product Description .....</b>	<b>1</b>
1.1 Introduction .....	1
1.2 Feature .....	1
1.3 Parameter .....	2
<b>2 Technical route .....</b>	<b>3</b>
2.1 System Framework .....	3
2.2 Regional frequency band .....	4
<b>3 Usage .....</b>	<b>5</b>
3.1 Network Server configuration .....	5
3.2 Decoder .....	8
3.3 Application Server configuration .....	10
3.4 Connect the Network Server and Application Server .....	11

# 1 Product Description

## 1.1 Introduction

AgroSense LoRaWAN® Soil Moisture Sensor measures the Soil Humidity at the range of 12-bit ADC, -40°C to 85°C with capacitive method, also with highly waterproof performance tested to IP68, making it widely applicable in **agricultural environmental sensing scenarios to support the smart agricultural production.**

The sensor benefits from LoRaWAN, which ensures stability and reliability. It is capable of covering a long transmission range while maintaining low power consumption. Unlike wireline devices, it is battery-powered, reducing the workload and complexity of deployment, design and development for end-users that can work via powering it, and setting the configuration in the cloud server, for LoRaWAN® remote monitoring. It monitors the Soil Moisture and report every 1 hour.



## 1.2 Feature

- Includes a **high precision** sensor.
- Compatible with Worldwide **LoRaWAN® Networks**: Support the universal frequency bands EU868/ US915.
- **Long Range**: Up to 2 kilometers in the city, up to 10 kilometers in the wilderness, receive sensitivity -137dBm , transmit power up to 21dBm.
- **Ultra-low power** consumption design, traditional AAA alkaline dry battery can be used for one year.

- **Data encryption:** Provide end-to-end secure communication, including device authentication and network data encryption, to ensure the security of data transmission and prevent data theft and malicious attacks.
- **High stability and reliability:** good stability in noisy environments, able to penetrate buildings and obstacles, so it can maintain good communication quality in urban and suburban environments.
- Suitable for **Harsh Environments:** Can work normally under the temperature of -40°C ~ 85°C, IP68 waterproof, suitable for outdoor use in harsh conditions, high UV, dusty, heavy rain and other bad weather.
- Monitor data and upload **real-time** data regularly.
- Modify the product parameters through **AT commands**.

## 1.3 Parameter

### 1. General Parameters

Product Model	AGLWSM01
Measurement Range	12-bit ADC
Measurement Accuracy	1/4096

### 2. Wireless Parameters

Communication Protocol	Standard LoRaWAN® protocol
Network Access/Operating Mode	OTAA Class A
MAX Transmit Power	21dBm
Receiver Sensitivity	-137dBm @300bps
Frequency Band	EU868/US915

### 3. Physical Parameters

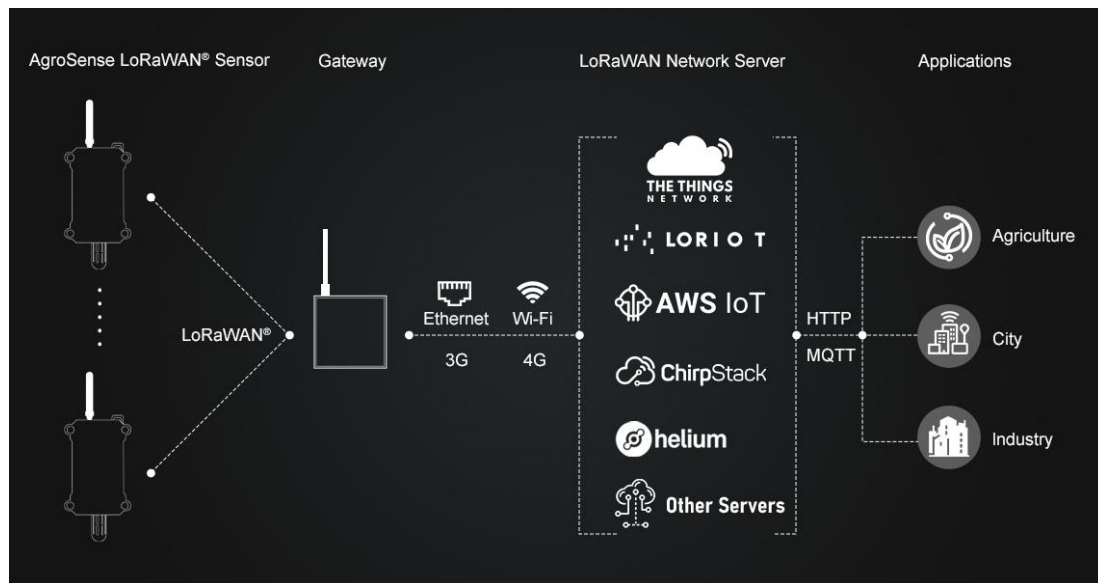
Lead Length	1.5 meter (custom length available)
Power Supply	2 x AAA 1.5V batteries
Operating Temperature	-40°C ~85°C
Protection Class	IP68
Dimensions	131 × 62.7 × 27.5 mm
Mounting	Wall Mounting

## 2 Technical route

### 2.1 System Framework

AgroSense Soil Moisture Sensor uses LoRAWAN technology, and its network architecture includes four parts: End Nodes, Concentrator/Gateway, Network Server and Application Server.

End Nodes	It is responsible for collecting sensing data and then transmitting it to Gateway via the LoRaMAC protocol.
Concentrator/Gateway	It is mainly responsible for transmitting node data to the server.
Network Server	Organize the data into JSON packets and decode them.
Application Server	Display the data.



**The steps to achieve the detection of Soil Moisture is:**

1. Collect the Soil Moisture data by sensor, and send the data from End Node to Gateway.
2. The Gateway packages node data and transmits it to the Network Server.
3. The Network Server decodes the data and sends it to the Applications.
4. Finally, user can monitor the Soil Moisture in the APP.

## 2.2 Regional frequency band

At the present moment, our product solely accommodates compatibility with the US915 and EU868.

area	frequency band	center frequency
China	470-510MHz	CN486MHz
America	902-928MHz	US915MHz
Europe	863-870MHz	EU868MHz
Korea	920-923MHz	KR922MHz
Australia	915-928MHz	AU923MHz
New Zealand	921-928MHz	NZ922MHz
Asia	920-923MHz	AS923MHz

## 3 Usage

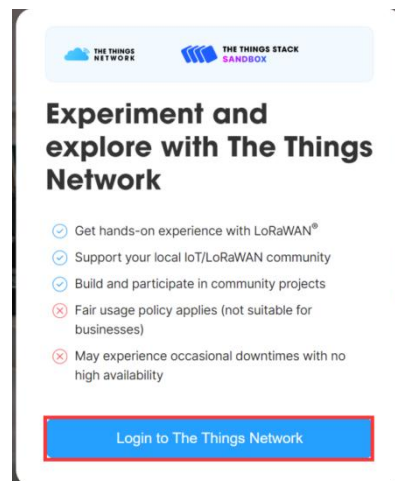
We use The Things Network as our Network Server, we need to configuration the country/ area frequency, inputting DEV EUI/ APP EUI/ APP Key, decodes, and connect to ThingSpeak.

DEV EUI	Unique identification of device, authorized by IEEE
APP EUI	Unique identification of application
APP Key	One of the join network parameters on OTAA mode, calculated by DE EUI

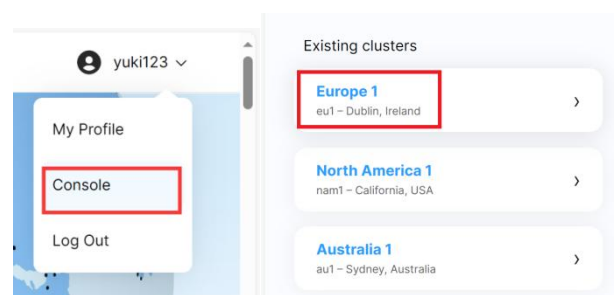
- End Nodes and Gateway: AgroSense Soil Moisture Sensor.(The AgroSense series is applicable)
- Network Server: The Things Network. ( Loriot, AWS IoT, ChirpStack, ect)
- Application Server: ThingSpeak.(Datacake, Blockbax, akenza, ect)

### 3.1 Network Server configuration

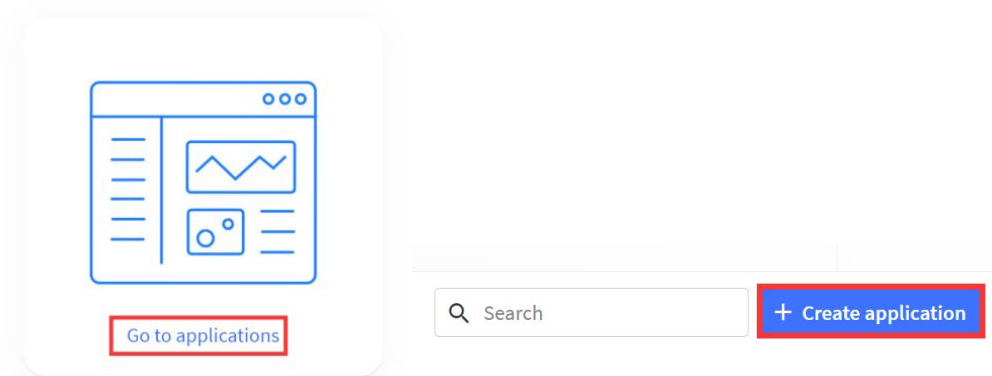
- Open The Things Network in your browser and login it. (Or register an account)



- Click “Console” and select clusters. (we take the European region for example.)



- Click “Go to applications” --> “+ Create application”.



- Write the Application ID and click “Create application”.

**Application ID \***

agrosense-sensor

**Application name**

My new application

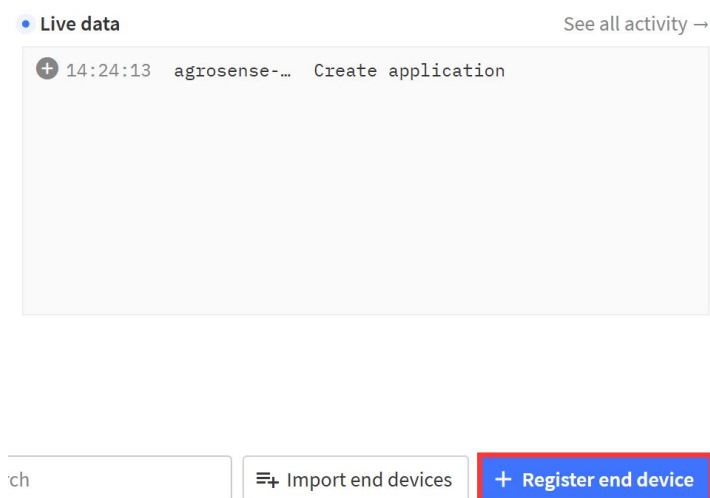
**Description**

Description for my new application

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

**Create application**

- Click “+ Register and device”.





- Following the steps, and input the DEV EUI/ APP EUI/ APP Key (notice: JoinEUI=APP EUI) and subsequently click on "Register end device" to complete the registration process.

Input method

Select the end device in the LoRaWAN Device Repository

Enter end device specifics manually

Frequency plan

Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version

LoRaWAN Specification 1.0.2

Regional Parameters version

RP001 Regional Parameters 1.0.2

Show advanced activation, LoRaWAN class and cluster settings

Provisioning information

JoinEUI

48 FF 00 00 00 01 65 Confirm

To continue, please enter the JoinEUI of the end device so we can determine



Provisioning information

JoinEUI

48 FF 00 00 00 01 65 Reset

This end device can be registered on the network

DevEUI

48 E6 63 FF FE 38 01 65 Generate 0/50 used

AppKey

4A 35 62 6B 95 A8 5B 4D 3F 38 DE 12 71 B1 6F 2A Generate

End device ID

eui-48e663ffe300165

This value is automatically prefilled using the DevEUI

After registration

View registered end device

Register another end device of this type

Register end device

- Plug the battery and press RES button, you can see the device is connected successfully in the TTN.



Live data

See all activity

09:43:10 Schedule data downlink for transmission on Gateway Server DevAddr: 26

09:43:10 Update end device [ "activated\_at" ]

09:43:10 Fail to send webhook Request

09:43:10 Forward uplink data message DevAddr: 26 0C 28 1F 00 01 06 31 1

09:43:10 Successfully processed data message DevAddr: 26 0C 28 1F

09:43:04 Forward join-accept message DevAddr: 26 0C 28 1F JoinEUI: 48

## 3.2 Decoder

- Now, we need to decode the data.

The screenshot shows the 'Live data' tab of a LoRaWAN decoder. A message at 09:52:55, 'Forward uplink data message', is selected. The payload is displayed as '00 01 06 31 1C 00 00 00 00 ...'. The interface includes tabs for Overview, Live data, Messaging, Location, Payload formatters, and General settings. It also has controls for Verbose stream, Export as JSON, Pause, and Clear.

Data length	Data description	Value range	Explanation
byte 0	Data packet sequence number high 8 bits	0-0xFFFF	Counting starts from 0 and increments, resetting back to 0 after reaching 65535
byte 1	Data packet sequence number low 8 bits		
byte 2	Soil Moisture sensor bits 8 to 15		Soil relative humidity 12-bit ADC value
byte 3	Soil Moisture sensor bits 0 to 7		
byte 4	Battery voltage		The value is obtained by amplifying the data by 10 times, and the actual value needs to be divided by 10 to convert to the actual battery voltage. The purpose of multiplying by 10 is to retain one decimal place of the voltage value. For example, if the value is 0x21 = 33, then the battery voltage is 3.3V.

Example: 0x00, 0x08, 0x05, 0xC7, 0x1E

Data parsing:

Soil ADC value is 1479.

Battery voltage is 3.0V.

- Know how to decode it after, we need to write it in code. (you can check it out on [Github](#))

```
function decodeUplink(input) {

    var num = input.bytes[0] * 256 + input.bytes[1]

    var bat = input.bytes[4] / 10.0

    var adc = input.bytes[2] * 256 + input.bytes[3]

    return {

        data: {

            field1: num,

            field2: bat,

            field3: adc,

        },

    };

}
```

- Select “Payload formatters” and follow the steps.

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

Uplink Downlink

**Setup**

Formatter type \*

Custom Javascript formatter

Formatter code \*

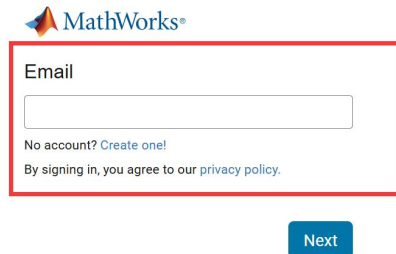
```
1 function decodeUplink(input) {
2     var num = input.bytes[0] * 256 + input.bytes[1]
3     var bat = input.bytes[4] / 10.0
4     var adc = input.bytes[2] * 256 + input.bytes[3]
5     return {
6         data: {
7             field1: num,
8             field2: bat,
9             field3: adc,
10        },
11    };
12 }
```

Save changes

### 3.3 Application Server configuration

In the Application Server configuration, we need to create ThingSpeak channel and get Channel ID and API Key, this is the key to our connection to TTN.

- Login to the ThingSpeak. (Or register an account)



MathWorks®

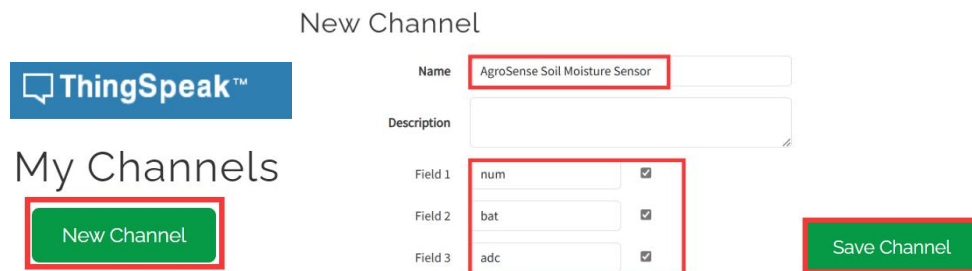
Email

No account? [Create one!](#)

By signing in, you agree to our [privacy policy](#).

Next

- Click “New Channel”, fill in the Channel name and field names and click “Save Channel”.



ThingSpeak™

My Channels

New Channel

New Channel

Name: AgroSense Soil Moisture Sensor

Description:

Field 1: num ☒

Field 2: bat ☒

Field 3: adc ☒

Save Channel

- After successful creation, copy the Channel ID and API Key.

#### AgroSense Soil Moisture Sensor

Channel ID: 2599652

Author: mwa0000034232775

Access: Private

[Private View](#)

[Public View](#)

[Channel Settings](#)

[Sharing](#)

[API Keys](#)

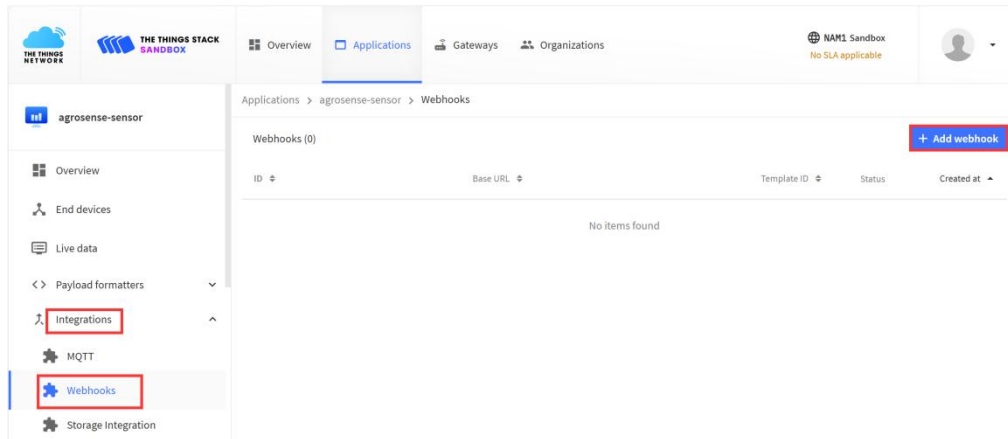
#### Write API Key

Key

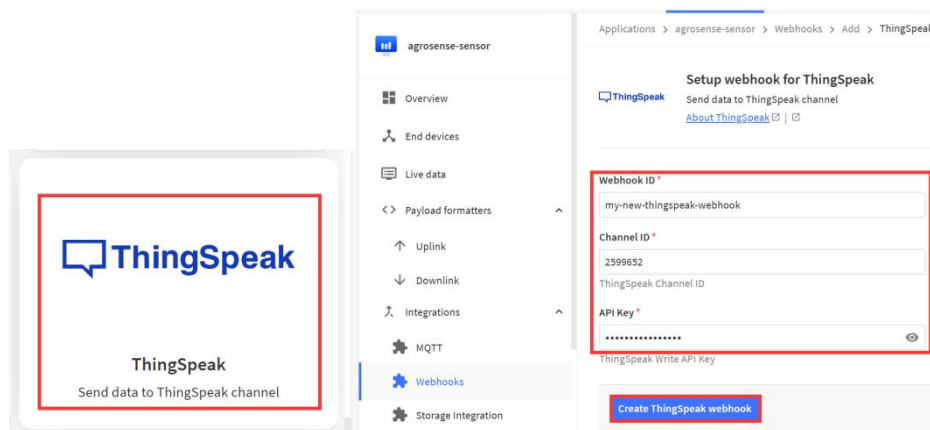
N9IBFTBI3J36T779

### 3.4 Connect the Network Server and Application Server

- In the TTN, click “integrations” --> “Webhooks” --> “+ Add webhook”.



- Select “ThingSpeak”, Fill in the Webhook ID and paste the Channel ID and API Key, click “Create ThingSpeak Webhook”.



- Press RST button, wait about a minute, you will successfully see the data in ThingSpeak.(You will receive the data every hour.)

