

LoRaWAN User Guide

Amp'ed RF Technology, Inc.

1. Configuration Table

Name	Description	Default	Remarks
VAR_ID_Default_Freq	"DefaultFreq"	"867000000"	867MHz
VAR_ID_Preamble_Len	"PreambleLen"	"8"	
VAR_ID_DevNonce	"DevNonce"	"0"	OTAA activation mode
VAR_ID_DevEui	"DevEui"	"70B3D57ED006D16F"	generated by TTN
VAR_ID_JoinEui	"JoinEui"	"0000000000000000"	
VAR_ID_AppKey	"AppKey"	"0C808B7134A06DF9C0650EE95F6CA935"	generated by TTN
VAR_ID_DevAddr	"DevAddr_ABP"	"260BE2DC"	
VAR_ID_DevEUI_ABP	"DevEUI_ABP"	"70B3D57ED006DA46"	generated by TTN
VAR_ID_NwkSKey	"NwkSKey_ABP"	"DDE7F019A8BC3FA51E0BB428D3DEF6E1"	generated by TTN
VAR_ID_AppSKey	"AppSKey_ABP"	"1B2EFEC0ADED645484423FE887DFA0E"	generated by TTN
VAR_ID_JOINMODE	"JoinMode"	"1"	1: OTAA mode 0: ABP mode
VAR_ID_AppPort	"fPort"	"101"	target port

- TTN will generate fixed values for DevEUI, AppKey, DevEUI_ABP, NwkSKey_ABP, and AppSKey_ABP
- Copy these values into the config settings

1.1. Port Range

Port range	Usage Category	Details
0	Reserved	
1-63	Standard application	Predefined application services (see the following example)
64-223	User-defined	developer freely allocated application channel
224-255	Experimental uses	interim testing/proprietary protocol development

2. AT Commands

2.1. LRJoin

After configuring the LoRAWAN end device, the following AT command joins the TTN network using the JoinMode configuration.

Syntax

```
at+ab LRJoin
```

2.2. LRSend

Send data over the LoRAWAN TTN network to the destination port in fPort configuration

Syntax

```
at+ab LRSend [Data]
```

[Data] is the hex data, ASCII hex, or ASCII data if preceded by a double-quote (“)

Example

```
at+ab SendData 31323334 (needs a CR LF to terminate)
at+ab SendData "12345" (needs a CR LF to terminate)
```

3. LoRAWAN TTN Network Setup

Configuring LoRAWAN Gateways and Nodes on The Things Network (TTN)

Prerequisites

3.1. Create a TTN Account

Visit the [TTN Console](#) and click Login to sign up using GitHub, Google, or other account.

3.2. Hardware Requirements

Gateway: A LoRaWAN-compatible gateway (WF288)

The gateway should be able to connect to the internet and transmit data to the LoRaWAN network.

Node Device: A LoRaWAN-enabled end device (LR71)

The node device should be able to communicate with the gateway and the LoRaWAN network.

3.3. Network Requirements

Connect the Gateway to the internet.

Ensure that the Node and Gateway are on the same frequency (e.g., 867Mhz)

4. Configuring a LoRaWAN Gateway

4.1. Add the Gateway to TTN

- Log in to the TTN Console, navigate to the Gateways page, and click + Add gateway.
- Fill in the following details:

Gateway ID: A unique custom identifier (e.g., my-lora-gateway)

Gateway EUI: The gateway's EUI (found on the device label or management interface).

Frequency Plan: Select the desired frequency band.

Antenna Placement: Choose Indoor or Outdoor.

- Click Create gateway.
- Document reference: [adding-gateways](#)

4.2. Configure Gateway Parameters

4.2.1. On the gateway management page, copy the Gateway Server Address (e.g., eu1.cloud.thethings.network).

4.2.2. Access the gateway's local management interface (via its IP address in a browser) and configure:

- Server Address: Paste the Gateway Server Address
- Port: Typically 1700 (UDP).
- Frequency Plan: Ensure it matches the TTN configuration.

4.2.3. Save the settings and restart the gateway.

4.3. Verify Gateway Connection

- Return to the TTN Console and check if the gateway status is **Connected**.
- Under the Live Data tab, confirm that the gateway is receiving data.

5. Configuring a LoRaWAN Node Device

5.1. Create an Application

5.1.1. In the TTN Console, go to Applications and click + Add application.

5.1.2. Provide the following details:

- Application ID: A unique name (e.g., my-lora-app)
- Description: Optional description

5.1.3. Click Create application.

5.1.4. Document reference: [adding-devices](#)

5.2. Register an End Device

5.2.1. In the application details page, navigate to the End Devices tab and click + Add end device.

5.2.2. Choose the device setup method:

Manually: Enter parameters manually (for custom devices).

- Document reference: [manual](#)

5.2.3. OTAA for activating the device

- Document reference: [Over the Air Activation](#)

From the LoRaWAN Device Repository: Select a pre-configured device model

5.2.4. Enter critical parameters:

- **DevEUI:** The device's unique identifier (printed on the device or retrieved via AT commands)
- **AppEUI:** Application identifier (use TTN's default or customize)
- **AppKey:** Generate or enter a custom key (for OTAA activation)

5.2.5. Click Create end device.

6. Validate Communication

6.1. Send Uplink Test Data to the Application from the Node

Add data transmission logic to the node's code (e.g., send sensor data every second)

- configure the parameter of the node to send data first with "**at+ab config**" command.

Example for OTAA under TTN platform

- DevEUI
- AppEUI
- AppKey

use AT command "at+ab LrJoin" to join network first

use AT command "at+ab LrSend "1234567890" with serial port term tool

6.2. View Data in TTN

Go to the application's **Live Data** tab to check for uplink data.

Use Payload Formats to decode binary data.

View the data in the **Live Data** tab.

[illegible]

6.3. Send Downlink Data to the Device in Manual Mode

- Go to the application's End devices tab and find Messaging
- Fill payload you want to send to the device

The screenshot shows the 'my-new-device1' Messaging page in The Things Stack. The 'Schedule downlink' tab is active. The 'Insert Mode' is set to 'Replace downlink queue'. The 'FPort' is set to '101'. The 'Payload type' is set to 'Bytes'. The 'Payload' field contains the hexadecimal string '0A 0B 0C 0D 0E 0F 0A 0B 0C 0D 0E 0F 0A 0B 0C 0D 0E 0F'. The 'Schedule downlink' button is highlighted with a red box. Red arrows indicate the flow from the 'Messaging' tab to the 'Schedule downlink' form and the 'Schedule downlink' button.

- The message will be sent to the device triggered by the uplink message from the device received by the application on network server
- Check the message on the Live Data tab

The screenshot shows the 'my-new-device1' Live data tab in The Things Stack. The 'Live data' tab is active. The table shows a list of messages. The message at 15:27:17 is highlighted with a red box, showing a 'Receive downlink data message' with the payload '0A 0B 0C 0D 0E 0F 0A 0B 0C 0D 0E 0F 0A 0B 0C 0D 0E 0F' and 'FPort: 101'. Red arrows indicate the flow from the 'Live data' tab to the highlighted message.

- Also this can be seen on the serial port term tool.

```
[15:27:31.820] 接收 -- Receive a Valid downlink RX1 for stack_id = 0 valid_rx_packet = 1
Modem event callback
INFO: Event received: DOWNDATA
Data received on port 101
Received payload = (30 bytes):
0A 0B 0C 0D 0E 0F 0A 0B 0C 0D 0E 0F 0A 0B 0C 0D
0E 0F 0A 0B 0C 0D 0E 0F 0A 0B 0C 0D 0E 0F
Modem event callback
INFO: Event received: TXDONE
```

7. Troubleshooting

Gateway Not Connected:	Verify internet connectivity and server address settings.
Node Fails to Join:	Ensure DevEUI, AppEUI, and AppKey match the TTN Console
No Uplink Data:	Confirm the node and gateway use the same frequency plan.

8. Glossary

OTAA (Over-The-Air Activation), Dynamic session keys via join process

ABP **ABP (Activation By Personalization)**, where all keys are configured manually on both the device and the server (TTN in this case). Static session keys.

TTN The Things Network (**TTN**) is a **public, open-source LoRaWAN network infrastructure**