RAK10701-P Field Tester Pro for LoRaWAN Quick Start Guide

Prerequisites

What Do You Need?

Before going through each and every step in the installation guide of the RAK10701-P Field Tester Pro for LoRaWAN, make sure to prepare the necessary items listed below:

Hardware Tools

- 1. RAK10701-P Field Tester Pro for LoRaWAN
- 2. LoRa SubGhz Antenna with RP-SMA connector
- 3. USB Type-C Cable
- 4. Windows/Linux/macOS for PC or iOS/Android for mobile

Software Tools

WisToolBox C for configuration and firmware update.

NOTE:

It is mandatory that you are within the coverage of the LoRaWAN Gateway of the network you are trying to join. Without the coverage, the Field Tester will not be useable.

Product Configuration

RAK10701-P Field Tester Pro Physical Interface

The user interface of the RAK10701-P Field Tester Pro for LoRaWAN is via TFT Touchscreen LCD and one pushbutton at the side. There is also an external LoRA antenna port via RP-SMA connector and USB-C port for charging and configuration if connected to a PC.



Figure 1: Parts of RAK10701-P

NOTE:

You have to ensure that the LoRa antenna is attached before turning on the device.

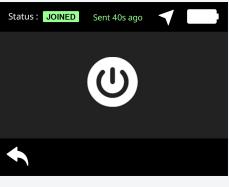
1. To turn on the device, you have to press and hold the button for at least five seconds.



Figure 2: RAK10701-P button to turn on

NOTE:

The same button can be used to power off. You have to hold it as well for at least five seconds.





2. When the device initializes, it will show the RAK logo on the screen. If there is any initialization error, it will be shown on the upper right section of the screen as well. A properly working device should not have any errors shown.



Figure 4: RAK10701-P power up successful



Figure 5: GPS error on boot up sequence

3. After the successful boot-up, the main home screen will be shown. Take note, that there will be no data at the first start of the device.

NOTE:

The field tester must be outside and has a clear view of the sky to get GPS coordinates. The GPS antenna is located on top of the device beside the RP-SMA connector of the LoRa Antenna.

If you are indoors, there will be no reception of the GPS signal. The latitude and longitude data will be empty.

Status : JOINING	G		
No. of Hotspot	Distance ► MAX : n MIN : n	RSSI ► n MAX : n MIN : SNR :	dBm dBm dB ►
Lat : Long :	•	\bigcirc	Ø

Figure 6: RAK10701-P Main Page waiting for valid data

- 4. Once fully powered on, the external button at the side can sleep or wake up the display on the LCD screen via a single press on it.
- 5. If the device is connected via USB-C to a computer, then the button is pressed, it will not remove the display but will lock the screen (touch screen behavior is disabled).

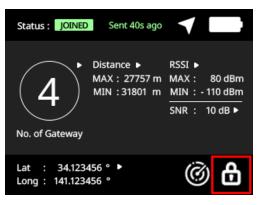


Figure 7: RAK10701-P locked screen

LoRaWAN Network Servers Guide for RAK10701-P Field Tester Pro

The field tester supports different network servers and can be used as well on others not listed in this guide as long as the uplink and downlink packets are configured correctly.

You can check each guide on how to use the RAK10701-P Field Tester Pro for LoRaWAN in the following network servers.

- Helium
- The Things Network
- Chirpstack (with Datacake)
- Loriot (with Datacake)

Additional information:

Packet Frame Format

NOTE:

This section will focus on the configuration of each network server. The procedure of Device Configuration of RAK10701-P via WisToolBox is the same for all network server and will be covered in a separate section of the guide.

RAK10701-P Field Tester Pro Guide for the Helium Network

RAK10701-P can be manually registered to Helium Console . This is a public LoRaWAN network server that you can use for your LoRaWAN end-devices powered by community-driven Helium Hotspots. This guide will show every detail of how to prepare the Helium Console for your RAK10701-P Field Tester Pro.

VOTE:

This guide is based on disk19 guide for the Field Tester ☐ configuration for the Helium Console.

1. You need to register an account and then purchase data credits (DC) to use the network. If you are a new user, there are free data credits (DC) included in your new account to get you started quickly.

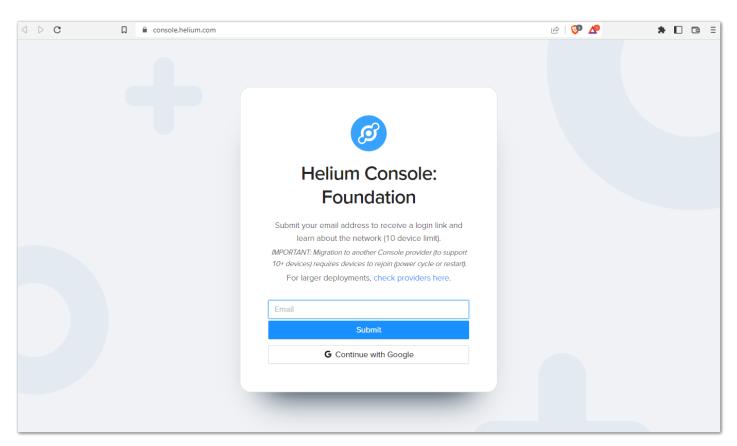


Figure 8: Heluim Console

ত Menu		Ø Q Search Console	?	@rakwireless.com RAKwireless v 250
	My Devices			
Flows	All Devices ((+)			
NODES Devices < Functions Integrations			Devices	
CONFIGS Alerts Profiles Packets		Devic	es can be added to the Helium network. Learn more about Devices	
ADMIN Coverage Organizations Data Credits Users				
				Ð

Figure 9: Console Home Page

2. Once you are logged in, you can start adding your device. You have two ways to add a device as shown in the image.

$\leftarrow \rightarrow G$	○ A == (https://console.helium.com/devices/home	☆	\boxtimes 7 \gg \equiv
🖻 Menu	Ø Q Search Console ?		@rakwireless.com RAKwireless ~ 250
	My Devices		
Flows	All Devices ((+)		
NODES Devices < Functions Integrations	Devices		
CONFIGS Alerts Profiles Packets	Devices can be added to the Helium network.		
ADMIN Coverage Organizations Data Credits Users			Add Device Add Label Add Function
			Add Function Add Integration

Figure 10: Adding Device

3. The newly added device parameters will be shown. You have to put a device name and click save.

NOTE:

The DEVEUI, APPEUI, and APPKEY are important in this step. These values must be configured on your RAK10701-P device using WisToolBox which will be covered later in this guide.

⊡ Menu	Ø Q Search Console	···· ?	Brakwireless.com RAKwireless v X250
	My Devices		
	All Devices ((+))		
Flows	Add New Device Important: Users can add up to 10 devices. The first time a device jo	ins the Network could take up to 20 mi	ns. Learn more about adding devices
NODES Devices <	ENTER DEVICE DETAILS	10 OF 10 DEVICES LEFT	£
Functions			Import Devices
Integrations	Name RAK1070x_Device	15/50	You can import your devices directly from the Things Network, or in bulk via .csv
CONFIGS Alerts	Dev EUI 6081F99CC58685CD	8 / 8 Bytes	upload. How do I format my .csv?
Profiles	App EUI 6081F90E8F3A4192	8 / 8 Bytes	Import from The Things Network
Packets	App Key O9B9D1EC5E52D7A607AF33DEE209E60E	16 / 16 Bytes	
ADMIN Coverage Organizations	Profile (Optional) Select a profile ∨		Drag .csv file here or click to choose file
Data Credits			
Users	Attach a Label (Optional) Search or Add Label		
		වී Save Device	Ð

Figure 11: Configuring Device Name

4. The device will be added to the blockchain and it will show pending beside its name.

i Menu		Ø Q Search Console	?		@rakwireless.com RAKwireless v 250
	My Devices				
Flows	All Devices ((·))				
NODES	All Devices			Edit Column	s Quick Action V
Functions	Device Name	EUI	Frame Down Packets Transferred	DC Used	d 💠 Last Connected ≑
Integrations	RAK1070x_Device Pending_	RCCS8685CD None	0	0 Nov 2, 2022	2:00 AM 💽 📵
Alerts					10 results V
Profiles Packets					
T doketo					
ADMIN					
Coverage					
Organizations					
Data Credits					
Users					
					Ð

Figure 12: Pending Device Status

5. While waiting for the device to be added to the blockchain, you can create a Label . This will allow you to group your device to have a common setting. This will be needed to attach the needed integrations to the backend server of dev.disk91.com . You have to click the folder with the + icon and add a Label name then click Save Label . The newly created label should now be shown in the Devices console.

⊡ Menu		Ø Q Search Console	?		@rakwireless.com RAKwireless ∨ X 250
	My Devices	abel			
Flows	All Devices ((+))				
NODES	All Devices			Edit Columns	Quick Action
Functions	Device Name	Device EUI	Frame Down ≑ Packets Transferred ≑	DC Used	Last Connected
Integrations	RAK1070x_Device Pending	6881F99CC58685CD None	0	0 Nov 2, 2022 2:00	АМ 💽 🖯
Alerts Profiles					10 results > < 1 >
Packets					
ADMIN Coverage Organizations Data Credits					
Users					•
					Ð

Figure 13: Add Label icon

画 Menu		Ø Q Search Console	?	@rakwireless.com RAKwireless v 250
	My Devices			
	1 Devices ((+)			
Flows				
NODES	ENTER LABEL DETAILS			
Functions Integrations	rak10701-devices			16/50
CONFIGS	Profile (Optional)			
Alerts Profiles	Select a profile V			
Packets				
ADMIN				Save Label
Coverage				
Organizations				
Data Credits				
Users				
				Ð

Figure 14: Add Label Name

i Menu		Q Search Console			@rakwireless.com RAKwireless >
	My Devices				
Flows	1 All Devices ((i))	k10701-devices kas			
NODES	All Devices			Edit Columns	Quick Action V
Functions	Device Name	UI	rame Down	DC Used	Last Connected \$\dot\$
Integrations	RAK1070x_Device Pending	CC58685CD None	0	0 Nov 2, 2022 2:00 /	
Alerts Profiles Packets					10 results V
ADMIN Coverage Organizations					
Data Credits Users					
					Đ

Figure 15: Label created successfully

6. Once the Label is created you have to associate it on the RAK10701 device. You can attach the Label on the device by clicking the **Add Label** button.

⊡ Menu	Ø Q Search Console		@rakwireless.com RAKwireless v X 250
	My Devices		
Flows NODES Devices <	RAK1070x_Device Pending	PACKETS TRANSFERRED DC USED	Delete Device
Functions Integrations CONRES Alerts Profiles Packets ADMIN Coverage Organizations Data Credits	Name RAK1070x_Device ID 2074542b-0a9e-47cd-abac-1fbb5527de78 Device EUI App EUI App Key Ø Activation Method OTA Profile None	All Time O Last 30 Days	Last 7 Days
Users	0 LABELS ATTACHED		+ Add Label

Figure 16: Attach a label to the device

7. A pop-up will be shown and you have to select the correct Label created for RAK10701 then click Add Label.

			×
Δ	dd Label 1	to 1 Device	
	i		
6	rak10701-device	s	
	rak10701-device	! 5	
	Cancel	Add Label	

Figure 17: Drop-down on label selection

8. After successful attachment of Label on the devices, it should show one (1) device is under that Label. The device is properly labeled which is needed for the next steps - Integrations and Flow.

⊡ Menu		Q Search Console		@rakwireless.com X 250 RAKwireless v
	My Devices			
		■ rak10701-devices 1 Devices		
	RAK1070x_Device			Delete Device
Flows	DEVICE DETAILS		PACKETS TRANSFERRED DC U	JSED
NODES Devices < Functions Integrations CONFIGS Alerts Profiles Packets ADMM Coverage		cd-abac-1fbb5527de78 685CD 0 2 A4192 0 2	All Time O Last 30 Days	Last 7 Days O Last 24 Hours O
Organizations Data Credits	1 LABEL ATTACHED			+ Add Label
Users	Label	Date /	Activated	
	rak10701-devices rak10701-devices	Nov 2	2, 2022 2:08 AM	8

Figure 18: Label added on the RAK10701 device

9. To connect the backend server, you have to create an Integration .

亘 Menu		🧭 Q Search Console	?	Grakwireless.com RAKwireless v X 250
	My Integrations Add New Integration			
Flows	All Integrations			
NODES	Name	Туре	Devices	
Devices Functions Integrations < conrises Alerts			No Data	
Profiles Packets				< 1 >
ADMIN Coverage Organizations Data Credits Users				

@rakwireless.com RAKwireless ∨ ★ 250			e ?	Q Search Consol	ø			르 Menu			
						ons	My Integrati				
						rations	All Integ O Infogrations				
						GRATION	ADD A CORE INTE	Flows			
						MOTT	П	NODES Devices Functions Integrations < CONFIGS Alerts			
						Y INTEGRATION	ADD A COMMUNIT	Profiles Packets			
	💷 🖻 🗉					my	Ø	ADMIN Coverage Organizations Data Credits Users			
	Akenza Datacake Google HTTP HTTP HTT	ntral Adafruit IO MOTT	Azure IoT Centra	Azure IoT Hub	AWS IoT Core	myDevices Cayenne HTTP	Helium Cargo HTTP				
Sheets Microshare	Akenza Datacake Google	ntral Adafruit IO			AWS IoT Core	Y INTEGRATION	ADD A COMMUNIT	NODES Devices Functions Integrations < CONRES Alerts Profiles Packets ADMN Coverage Organizations Data Credits			

Figure 20: HTTP integration

10. Then you must proceed to steps 2 and 3 sections of the Integration settings. You have to select POST then on the Endpoint URL, you must put https://dev.disk91.com/fieldtester/helium/v3. It is also needed to put the integration name before the clicking Add integration button.

프 Menu	Q Search Console	@rakwireless.com X 250 RAKwireless >
	This integration allows for sending data to an endpoint, as well as receiving data, over HTTP. Tell me more about setting up this Integration. Change	
	STEP 2 - ENDPOINT DETAILS	
Flows NODES Devices	POST GET PUT PATCH Endpoint URL (Required) https://dev.disk91.com/fieldtester/helium/v3	٥
Functions	HTTP Headers (Optional usage for payload interpolation) Key Value	
CONFIGS Alerts Profiles	+ Add Header	
Packets	URL Params (Optional usage for payload interpolation) + Add Param	
Coverage Organizations Data Credits Users	STEP 3 - NAME YOUR INTEGRATION (REQUIRED)	
USUS	RAK1070x_Integration 20/50	
	Add Integration	Đ

Figure 21: Details of HTTP Integration

11. After preparing the device and the integration, you can now proceed with creating the flow to connect them. You have to click Flows and then the + icon on NODES.

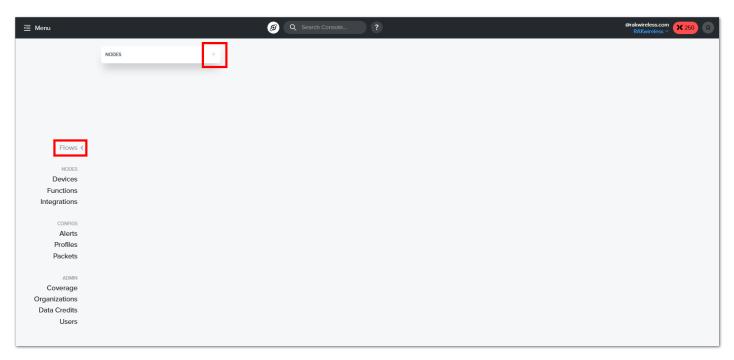


Figure 22: Setting up the Flows

12. You must select Labels and Integrations and then drag the correct blocks on the flows canvas.

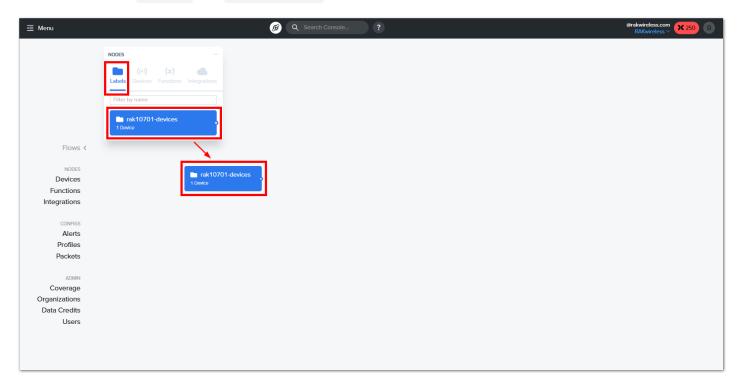


Figure 23: Drag the rak10701-devices label

⊡ Menu	Ø	Q Search Console ?		Brakwireless.com RAKwireless v X 250
	NODES			
Flows <				
NODES				
Devices	rak10701-devices 1 Device	• RAK1070x_Integration	16	
Functions				
Integrations				
CONFIGS				
Alerts				
Profiles				
Packets				
ADMIN				
Coverage				
Organizations				
Data Credits				
Users				

Figure 24: Drag the RAK1070x_Integration

13. You then have to connect the Label block to the Integration block via the tiny connector indicated by the red arrows by using your mouse cursor and dragging the line connector.

NOTE:

There is no need to save the changes created on the flows canvas since it is automatically saved as you do changes.

⊡ Menu	ß	Q Search Console ?	Brakwireless.com RAKwireless v X 250
			KARWIEIESS *
	NODES +		
Flows <	,		
NODES			
Devices	nak10701-devices	RAK1070x_Integration	
Functions			
Integrations		/	
CONFIGS			
Alerts			
Profiles Packets			
Packets			
ADMIN			
Coverage			
Organizations Data Credits			
Users			

Figure 25: Connecting Labels to Integrations

14. The final step in the setting up of the Helium Console for RAK10701 is the setting up of packets. You have to click on Packets and then the Add New Packet Config icon.

프 Menu		🧭 🔍 Search C	onsole ?		Ørakwireless.com RAKwireless ∽ → 250
	Packet Confi	igurations New Packet Config			
	All Packet Configs	ā			
Flows	All Packet Cor	figurations			
NODES	Name	Preferred Hotspot(s)		Multiple Packets	
Devices Functions Integrations					
CONFIGS Alerts			No Data		
Profiles Packets <					
ADMIN Coverage					
Organizations					
Data Credits Users					

Figure 26: Creating Packets

15. You must also input a name on the Packet Config Name, select Multiple Packets and drag the slider so it will show All Available Packets. Once done, you can now click on + Create Packet Config.

亘 Menu	ø Q Search Console	?	@rakwireless.com RAKwireless v 250
	Packet Configurations		
	All Packet Configs		
Flows NODES Devices Functions Integrations CONFIRS Alerts Profiles Packets < ADMIN Coverage Organizations Data Credits Users	For Configurations Use to choose which Hotspots to receive packets from or how may packets to purchase from Hotspots (if available). A Packet configuration requires either a Preferred Hotspot or a Multiple acket setting. Learn more about Packet Configurations	Packet Confin Name ALL Packets If this setting is chosen, packets will ONLY be purchased from the selected Preferred Hotspots. Vour Organization must have Preferred Hotspots to enable this setting. To prefer a hotspot, please visit the <u>Coverage Page</u> . Image: Coverage Page Multiple Packets How many packets do you want to purchase (if available)? Image: Coverage Page Image: Coverage Imag	11/25

Figure 27: Multiple packets configuration

画 Menu		ø Q Search Console	?	@rakwireless.com RAKwireless v X 250
	Packet Config	jurations		
	All Packet Configs	ALL Packets Packet Config		
Flows	All Packet Confi	gurations		
NODES	Name	Preferred Hotspot(s)	Multiple Packets	
Devices Functions	ALL Packets		Up to 21 packets	0
Integrations CONFIGS Alerts Profiles Packets <				
ADMIN Coverage Organizations Data Credits Users				
03613				

Figure 28: Successful packets creation

16. You must associate the multiple packet setup with the rak10701-devices label. To do this, you have to go back on Flows, double-click on the rak10701-devices label, choose the Pakets tab under rak10701-devices settings then enable ALL Packets. After this, you can now proceed with the configuration of RAK10701 using WisToolBox.

프 Menu		erakwireless.com RAKwireless v 🗙 250
E Menu Flows < NODES Devices Functions Integrations ONFIGS Alerts Profiles Packets Packets ADMN Coverage Organizations Data Credits Users	NODE	Image: Description of the second state of the second st

Figure 29: Attaching multiple packets to the rak10701-devices label

17. You can now proceed on device configuration so that the proper EUIs and KEY will match the one in the Helium network.

RAK10701-P Field Tester Pro Guide for The Things Network

This section shows how to use the RAK10701-P Field Tester Pro for LoRaWAN to The Things Stack.

1. Log in to TTNv3. To do so, head to the TTNv3 site ☑ and select your cluster. If you already have a TTN account, you can use your The Things ID credentials to log in.

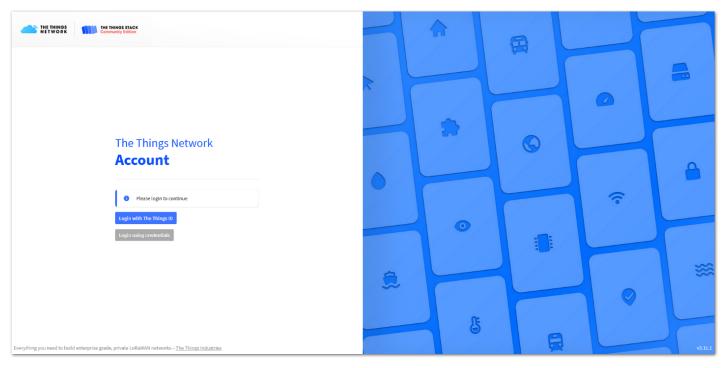


Figure 30: The Things Stack home page

THE THINGS THE THINGS STACK	👫 Overview 🗖 Applications 🗳 Gateways 🚉 Organizations		🔀 eu1 Community rakwirelesseu-67og 🔹
	Get started right away by creating a	n application or registering a gateway. Documentation ^{III} or <u>Get Support</u> I ^{III} .	
		°	
	Create an application	Register a gateway	

Figure 31: Console page after a successful login

VOTE:

To connect RAK10701-P Field Tester Pro to TTNv3, you should already have connected a gateway in range to TTNv3. Or, you have to be sure that you are in the range of a public gateway.

2. Now that you are logged in to the platform, the next step is to create an application. In your console, click **Create an application**.

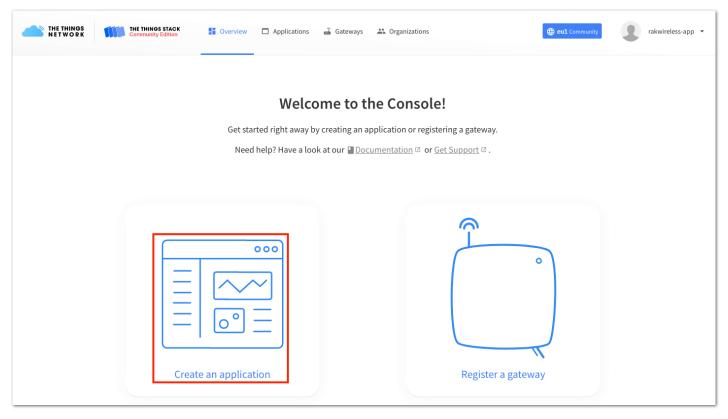


Figure 32: Create an application

3. To have an application registered, you need to input first the specific details and necessary information about your application then click **Create application**.

THE THINGS NET WORK	THE CO	E THINGS STACK mmunity Edition	Overview	Applications	📸 Gateways	A Organizations	EU1 Con		a ra	k-discovery 👻
		Add ap	plication							
Owner* rak-discovery										
		Application II lorawan-dev Application n	ices-v1							
		LoRaWAN De	vices Application							
The Application involves LoRaWAN Applications Optional application description; can also be used to save notes about the application										
		Create ap	plication							
© 2021 The Th	nings Stack by T	he Things Network	and The Things Inc	dustries			🌐 en	v3.16.1	Documentation	? Get support

Figure 33: Creating an Application

4. If you had no error during the previous step, you should now be on the application console page. The next step is to **add end-devices to your TTN application**.

THE THINGS STACK Community Edition	Serview Applications	🗳 Gateways 🛛 🚢 Organizations			EU1 Community support plan ⑦	rak-discovery •			
LoRaWan Devices Application	Applications > LoRaWan De	vices Application							
Overview	LoRaWan Devices Application ID: Iorawan-devices-v1								
1 End devices	• No recent activity ③			🙏 1 End	d device 🛛 👫 2 Collab	orators 🛛 🗣 0 API keys			
1. Live data	General information			Live data		See all activity →			
<> Payload formatters ~	Application ID	lorawan-devices-v1	•			out an outring			
↑ Integrations ~	Created at	Dec 14, 2021 12:19:37							
2 Collaborators	Last updated at	Dec 14, 2021 12:19:37		Waiting for events from	n lorawan-devices-v1.				
Or API keys									
General settings									
	End devices (1)			Q Search by ID	+ Import end devices	+ Add end device			
< Hide sidebar	ID \$	Name Φ	DevEUI	JoinEUI		Last activity			

Figure 34: Add end-devices to your TTN application

5. To register the RAK10701-P Field Tester Pro, you need to click **Manually** first.

THE THINOS NET WORK	THE THINGS STACK Community Edition	Overview	Applications	Gateways Greanizations	EU1 Con ir use policy a			ak-discovery 🔻		
111 LoR	aWAN Apps		Applications	> LoRaWAN Apps > End devices > Register from The LoRaWAN Device Repository						
Uver	nview		Registe	r end device						
📩 End			From The L	RaWAN Device Repository Manually						
 Live Payl Integ Colla API I 	oad formatters v grations v aborators		1. Select the end device Brand ©* Type to search \ Cannot find your exact end device? <u>Get help here</u> and <u>try manual device registration</u> .							
🔅 Gen	eral settings			zistration data						
			Please choose Register e	an end device first to proceed with entering registration data						
< Hide sid	debar									
© 2021 The T	© 2021 The Things Stack by The Things Network and The Things Industries V3.16.1 Documentation									

Figure 35: Adding end devices manually

6. Choose the following configurations in adding the end devices. You must choose the correct Frequency Plan and the LoRaWAN version must be 1.0.3.

THE THINGS STACK	Overview Applications Gateways Corganizations	rak-discovery *						
LoRaWAN Apps	Applications > LoRaWAN Apps > End devices > Register manually							
Overview	Register end device							
End devices	From The LoRaWAN Device Repository Manually							
u Live data	Frequency plan (2)*							
Payload formatters ~	Europe 863-870 MHz (SF9 for RV2 - recommended)							
↑ Integrations ~	LoRaWAN version 🔊 *							
Collaborators	LoRaWAN Specification 1.0.3							
Or API keys	Regional Parameters version ③ * RP001 Regional Parameters 1.0.3 revision A							
General settings	Show advanced activation, LoRaWAN class and cluster settings ~							
	DevEUI ③ *							
< Hide sidebar	AppEUI ③* Fill with zeros							

Figure 36: Configurations for adding end devices

7. Click **Show advanced activation, LoRaWAN class, and cluster settings**, then select **Over the air action** (OTAA).

LoRaWAN Apps	Show advanced activation. LoRaWAN class and cluster settings ^
Overview	Activation mode [®] * • Over the air activation (OTAA)
🙏 End devices	Activation by personalization (ABP) Define multicast group (ABP & Multicast)
Live data Average Ave	Additional LoRaWAN class capabilities None (class A only)
 Integrations Collaborators API keys 	 Network defaults ① Use network's default MAC settings Cluster settings ① Use external LoRaWAN backend servers
General settings	DevEUI ③* · · · · · · · · · · · · · · · · · · ·
< Hide sidebar	End device ID ③* my-new-device This value is automatically prefilled using the DevEUI

Figure 37: OTAA settings

8. Then input the LoRaWAN OTAA parameters. For **AppEUI**, you may click **Fill with Zeros**. For **AppKey** and **DevEUI**, you can click **Generate**. Then the parameters will be automatically filled by the TTS platform. Finally, click **Register End Device**.

LoRaWAN Apps	None (class A only)				
	Network defaults 🗇				
Overview	Use network's default MAC settings				
🙏 End devices	Cluster settings [®] Use external LoRaWAN backend servers				
1. Live data	DevEUI ⊚*				
<> Payload formatters ~					
↑ Integrations ~	AppEUI ⊙*				
Collaborators	Fill with zeros AppKey ☉ *				
O+ API keys	¢ Generate				
Ceneral settings	End device ID ⑦*				
Generalisettings	eui-acif09fffe03ef47				
	This value is automatically prefilled using the DevEUI				
	After registration				
	View registered end device				
	Register another end device				
	 подним иничны чим меттее от ила туре 				
< Hide sidebar	Register end device				
© 2021 The Things Stack by The Things Network a	nd The Thines Industries , , , , , , , , , , , , , , , , , , ,	∰ FN \	v3.16.1	Documentation	Get support

Figure 38: Registering the end device

9. You should now be able to see the device on the TTN console after you fully register your device. Take note of these OTAA parameters, such as the AppEUI, DevEUI, and the AppKey, as they are needed in the configuration of the RAK10701-P Field Tester Pro hardware later on in this guide.

	HE THINGS STACK	Overview	Applications	ᡖ Gatewa	ys 🎎 Organizations	EU1 Community No SLA applicable	rak-discovery 🔹					
LoRaWan Devices	Application		Applications 3									
LoRaWan Devices	Application		-	eui-ac1f09fffe0		ing General settings						
Payload formatters												
↑ Uplink			General inform End device ID	nation	eui-ac1f09fffe03ef47	Live data	See	all activity →				
↓ Downlink			Description		This end device has no description							
	~		Created at		Dec 14, 2021 12:50:14	Waiting for events from eui-aclf09fffe03ef47	ling for events from eui-ac1f09fffe03ef47					
2 Collaborators			Activation info									
Or API keys			AppEUI		0	6						
🔹 General settings			DevEUI		0	Location	Change locatio	n settings →				
			Root key ID		n/a	_						
			АррКеу		•••••	0						
			NwkKey		n/a							
< Hide sidebar			Session inform	nation			No location information available					

Figure 39: OTAA device successfully registered to TTN

10. After adding the device to the LoRaWAN application, link it to the backend server. The first step is to create an **API key**.

THE THINGS STACK Community Edition	Serview Applications	🛋 Gateways 🛛 🗮 Organizations	EU1 Community No support plan (?)	rak-discovery •					
LoRaWan Devices Application	Applications > LoRaWan Devices Application								
Cverview	LoRaWan Devices Application ID: lorawan-devices-v1								
🙏 End devices	• No recent activity ⑦		🚴 1 End device 🛛 🚢 2 Collaborators 💁 0 API keys						
1. Live data	General information		• Live data	See all activity →					
A Payload formatters	Application ID	lorawan-devices-v1	Live data	See all activity ->					
↑ Integrations ~	Created at	Dec 14, 2021 12:19:37							
2 Collaborators	Last updated at	Dec 14, 2021 12:19:37	Waiting for events from lorawan-devices-	v1					
Or API keys									
General settings									

Figure 40: Creating API key

	THE THINGS STACK Community Edition	Overview	Applications	🗳 Gateways	😃 Organizations		rak-discovery •			
LoRaV	Nan Devices Application	Applications > LoRaWan Devices Application > API keys								
	Lonawan Devices Application	API keys (0)					+ Add API key			
Overvi	ew	Key ID 💠		Name ¢	Granted Rights		Created at 🔺			
📩 End de	evices				No items found					
Live da	ata									
<> Payloa	d formatters 🗸 🗸									
1 Integra	ations 🗸									
🚢 Collab	orators									
OT API key	ys									
🔅 Genera	al settings									

Figure 41: Creating API key

11. Configure the API key parameters. You can put any names that will easily track your API. You have to set the expiration date as well. Then you must check Write downlink application traffic . After the configuration, you can now click Create API key .

<u>u</u>	LoRaWan Devices Application	Name field-mapper					
	Overview	Expiry date 01/01/2030					
*	End devices	Rights*					
11	Live data	Grant all current and future rights Grant individual rights					
$\langle \rangle$	Payload formatters 🗸 🗸	Select all					
t	Integrations 🗸	Delete application					
	Collaborators	View devices in application					
		View device keys in application					
••	API keys	Create devices in application					
\$	General settings	Edit device keys in application					
		View application information					
		Link as Application to a Network Server for traffic exchange, i.e. read uplink and write downlink					
		This implicitly includes the rights to view application information, read application traffic and write downlinks					
		View and edit application API keys					
		Edit basic application settings					
		View and edit application collaborators					
		View and edit application packages and associations					
		Vite downlink application traffic					
		Read application traffic (uplink and downlink)					
		Write uplink application traffic					
< Hid	le sidebar	Create API key					

Figure 42: API key parameters

12. This step is critical. You need to copy the API key because this will be used on Webhook integration.

Granted rights	Your API key has been created successfully. Note: After closing this window, the value of the key secret will not be accessible anymore. Make sure to copy and store it in a safe place now.
	API key
	•••••• 🚡 📀

Figure 43: Copy API key

13. With the API key created, you can proceed with creating the Webhook integration.

THE THINGS NETWORK	THE THINGS STACK Community Edition	Overview	Applications	🚔 Gateways 🛛 🗮 Organizations		EU1 Community No support plan (2)	rak-discovery •
11 LoR	aWan Devices Application		Applications > Lot	aWan Devices Application > Webhooks			
	awan bevices application		Webhooks (0)			+	Add webhook
Over	rview		ID \$	Base URL 🗢	Template ID 💠	Status	Created at 🔺
📩 End	devices			No items found			
il. Live	data						
<> Payl	oad formatters 🗸 🗸						
大 Integ	grations ^						
🗯 м	QTT						
* w	ebhooks						
🛸 St	torage Integration						
🗯 A)	WS IoT						
🌲 Az	zure IoT						
🌲 La	oRa Cloud						
🚜 Colla	aborators						



14. Select Custom Webhook.

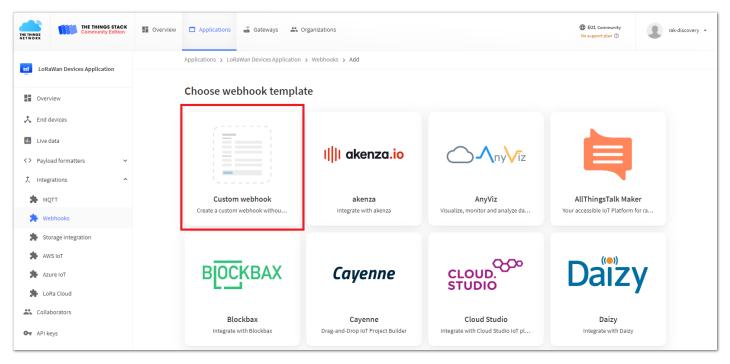


Figure 45: Custom Webhook

15. Configure the necessary parameters on the Webhook. You can select any name for the webhook. You then need to set the base URL going to disk19 server https://dev.disk91.com/fieldtester/ttn/v3, add the API key from the previous step and lastly put a check on the Uplink message under **Enabled event types**.

LoRaWan Devices Application	General settings
Overview	Webbook ID* field-tester-Integration
Lend devices	Webhook format*
Live data	JSON V
<> Payload formatters ~	https://dev.disk91.com/fieldtester/ttn/v3
	Downlink API key
MQTT	The API key will be provided to the endpoint using the "X-Downlink-Apikey" header
Storage Integration	Request authentication [®]
AWS IOT	Additional headers + Add header entry
Azure IoT	Filter event data 🗇
 LoRa Cloud Collaborators 	+ Add filter path
G API keys	Enabled event types
General settings	For each enabled event type an optional path can be defined which will be appended to the base URL Uplink message /path/to/webhook
	An uplink message is received by the application
< Hide sidebar	Join accept An end device successfully joins the network and starts a session

Figure 46: Webhook parameters

16. After setting all the configurations, you can now add the webhook.

		For each enabled event type an optional path can be defined which will be appended to the base URL
ul.	LoRaWan Devices Application	Uplink message /path/to/webhook
		An uplink message is received by the application
	Overview	Join accept
		An end device successfully joins the network and starts a session
~	End devices	
ıl.	Live data	Downlink ack
_		A confirmed downlink is acknowledged by an end device
<>	Payload formatters 🗸	Downlink nack
t	Integrations ^	A sent confirmed downlink fails confirmation by the end device
	integrations	
1	MQTT	Downlink sent A downlink is sent to an end device or multicast group
	Webhooks	A downlink is sent to an end device of multicast group
	- Webnooks	Downlink failed
1	Storage Integration	A downlink cannot be sent
	AWS IOT	Downlink queued
		A downlink is added to the downlink queue
	Azure IoT	
	LoRa Cloud	Downlink queue invalidated \odot
	Edita ciolu	The downlink queue is reset due to frame counter mismatch
**	Collaborators	Location solved
		An integration succeeded locating the end device
От	API keys	
¢	General settings	Service data
		An integration emits an event
< Hid	le sidebar	Add webhook

Figure 47: Add Webhook

17. You should see now the newly created webhook.

THE THIN NET WO	THE THINGS STACK Community Edition	Overview	Applications	🛋 Gateways	👪 Organizati	ions		B EU1 Commu No support plan		rak-discovery 🔹	
nt	LoRaWan Devices Application		Applications > LoR	aWan Devices Applic	cation > Webł	hooks					
	containan bertes application		Webhooks (1)						+ Add webhook		
5	Overview		ID \$			Base URL 🗢	Template ID 💠	Status	Created at		
*	End devices		field-tester-integr	ation		https://dev.disk91.com/fieldtester/ttn/v3	None	Pending •	16 seconds ago		
ıh	Live data										
\diamond	Payload formatters ~										
, t	Integrations ^										
	мотт										
	Webhooks										
3	Storage Integration										
	AWS IOT										
	Azure IoT										
1	LoRa Cloud										
*	Collaborators										

Figure 48: Add Webhook

- 18. After adding the application, device, and webhook integration to the console, you have to configure the parameters in your device to match the parameters on the TTN console. You can use WisToolBox 🗗 via USB connection or wirelessly via BLE. You can now proceed on the RAK10701-P Configuration using WisToolBox. You also have the option to update device parameters directly via RUI3 AT Commands 🖄 (if you prefer AT commands instead of WisToolBox).
- Once you configured the RAK10701-P with the correct Frequency Band and EUIs/Key by following the guide on the RAK10701-P Configuration using WisToolBox, you should see the join request/accept, uplinks and downlinks to The Things Stack console. These uplinks contains the coordinates of the field tester and the downlinks contains the data calculated by the disk91 server. The uplink uses fport 1 and the downlink uses fport 2. To view the actual coordinates on the console, you need to add a payload decoder on your uplink data.

field-tester	↑ 40 🔸 79 🔹 Last activity 16 seconds ago 💿		
Theoreman and the second	Overview Live data Messaging Location Payload formatters Claiming	General settings	
Overview			
🙏 End devices	Uplink Downlink		
🔲 Live data	Setup	Test	
Payload formatters ~	Formatter type *	Byte payload	FPort
	Custom Javascript formatter		1 Test decoder
Collaborators	Formatter code *	Decoded test payload	
• API keys	1 function Decoder(Dytes, port) [] var decoded = []; // avoid sending Downlink ACK to integration (Cargo) if (port == 1)		
🕸 General settings	6 varlonsign = (bytes[0] >> 7) & 0x01 ? -1 : 1; 6 7 Varlonsign = (bytes[0] >> 6) & 0x01 ? -1 : 1;		
	var enclat = ((bytes[] & 0x3f) << 17) + (bytes[] << 0 + (bytes[2] << 1) + (bytes[3] >> 7);		
	<pre>var enclon = ((bytes[3] & 0x7f) << 16) + (bytes[4] << 8) + bytes[5];</pre>	Complete uplink data	
	<pre>var hdop = bytes[8] / 10; var sats = bytes[9];</pre>		
	9 8 const maxHdop = 2; 1 const minSats = 5;		
	<pre>if ((hdop < maxHdop) && (sats >= minSats)) { // Send only acceptable quality of position to mappers deceded.latticds = latSign = (enclat + 200 + 53) / 10000000;</pre>		
		Learn more about payload formatters [2]	
< Hide sidebar	Save changes		

Figure 49: Adding Uplink Payload Decoder

Uplink payload decoder script.

```
function Decoder(bytes, port) {
   var decoded = {};
   // avoid sending Downlink ACK to integration (Cargo)
   if (port === 1) {
        var lonSign = (bytes[0] >> 7) & 0x01 ? -1 : 1;
        var latSign = (bytes[0] >> 6) & 0x01 ? -1 : 1;
        var encLat = ((bytes[0] & 0x3f) << 17) +</pre>
            (bytes[1] << 9) +
            (bytes[2] << 1) +
            (bytes[3] >> 7);
        var encLon = ((bytes[3] & 0x7f) << 16) +</pre>
            (bytes[4] << 8) +
            bytes[5];
        var hdop = bytes[8] / 10;
        var sats = bytes[9];
        const maxHdop = 2;
        const minSats = 5;
        if ((hdop < maxHdop) && (sats >= minSats)) {
            decoded.latitude = latSign * (encLat * 108 + 53) / 10000000;
            decoded.longitude = lonSign * (encLon * 215 + 107) / 10000000;
            decoded.altitude = ((bytes[6] << 8) + bytes[7]) - 1000;</pre>
            decoded.accuracy = (hdop * 5 + 5) / 10
            decoded.hdop = hdop;
            decoded.sats = sats;
            decoded.error = "Need more GPS precision (hdop must be <" + maxHdop +</pre>
                " & sats must be >= " + minSats + ") current hdop: " + hdop + " & sats:" + sats;
        return decoded;
```

20. With the correct payload decoder, you should now see GPS coordinates data which you can also use to other integration.

111 field-tester		Applica	tions > field-tester- > Live data					
- Held-tester	Time Entity I	D	Туре	Data preview	Verbose stream 🔿 🗙	Export as JSC	N 🕨 Resume	🔋 Clear
Overview	01:26:11		Console: Stream paused	The event stream has been paused				
👗 End devices	↓ 01:25:58 eui-70	b3d57ed805ba3a	Receive downlink data message	Payload: 04 98 98 01 01 01 (>)				
	↑ 01:25:57 eui-78	b3d57ed005ba3a	Forward location solved message	Latitude: 14.6263157 Longitude: 121.1690657 Altitude: 170 Source: GPS				
💷 Live data	/ 01:25:57 eui-70	b3d57ed005ba3a	Update end device	["locations"]				
<> Payload formatters ~	↑ 01:25:57 eui-70	b3d57ed805ba3a	Forward uplink data message	DevAddr: 26 00 CB 16 😔 🖺 Payload: { accuracy: 0.9, altitude: 170, hdo	op: 0.8, latitude: 14.626315	7, longitude: 121	.1690657, sats	: 10 } OA E
大 Integrations ~	↓ 01:25:38 eui-70	b3d57ed805ba3a	Receive downlink data message	Payload: 03 99 99 01 01 01 ↔ 🚡 FPort: 2				
Collaborators	↑ 01:25:37 eui-70	b3d57ed805ba3a	Forward location solved message	wessage Latitude: 14.6263157 Longitude: 121.1690657 Altitude: 174 Source: GPS				
Or API keys	/ 01:25:37 eui-70	b3d57ed805ba3a	Update end device	["locations"]				
General settings	↑ 01:25:37 eui-70	b3d57ed805ba3a	Forward uplink data message	age DevAddr: 26 60 CB 16 🛛 🐞 Payload: { accuracy: 0.95, altitude: 174, hdop: 0.9, latitude: 14.6263157, longitude: 121.1690657, sats: 9				
	↓ 01:25:18 eui-70	b3d57ed805ba3a	Receive downlink data message	Payload: 02 91 91 01 01 01 ()				
	↑ 01:25:17 eui-70	b3d57ed005ba3a	Forward location solved message	Latitude: 14.6262725 Longitude: 121.1690657 Altitude: 181 Source: GPS				
	/ 01:25:17 eui-70	b3d57ed805ba3a	Update end device	["locations"]				
	↑ 01:25:17 eui-70	b3d57ed805ba3a	Forward uplink data message	DevAddr: 26 0D C8 16 ↔ 🖺 Payload: { accuracy: 0.9, altitude: 181, hdc	op: 0.8, latitude: 14.626272	5, longitude: 121	.1690657, sats	: 10 } OA 8
	↓ 01:23:19 eui-78	b3d57ed805ba3a	Receive downlink data message	Payload: 01 B0 80 00 00 01 🗘 🚡 FPort: 2				
	↑ 01:23:17 eui-70	b3d57ed805ba3a	Forward uplink data message	DevAddr: 26 0D CB 16 \leftrightarrow Payload: { error: "Need more GPS precision ((hdop must be <2 & sats must	be >= 5) current	hdop: 0 & sat	s:0" } 00(
< Hide sidebar	↑ 01:16:24 eui-70	b3d57ed805ba3a	Forward join-accept message	DevAddr: 26 00 CB 16 🗘 🖺				
X mue sidebai	⊕ 01:16:22 eui-70	b3d57ed805ba3a	Accept join-request	DevAddr: 26 00 CB 16 ↔ 🖺			_	
© 2023 The Things Stack by The Things Netwo	rk and The Things Industri	ies			EN v3.24.2 (271c29761)	Documentation	Status page 🤇	Get support

Figure 50: Decoded payload and TTS console

RAK10701-P Field Tester Pro Guide for Chirpstack How Does It Work?

There are two steps under the hood of the Field Tester. In step one, the Field Tester is sending out data packets over LoRaWAN. These packets are received by one or multiple gateways. These packets are forwarded from the LoRaWAN network server to another backend server. When the packets are forwarded, they include information about signal strength and the number of gateways that have received the packet.

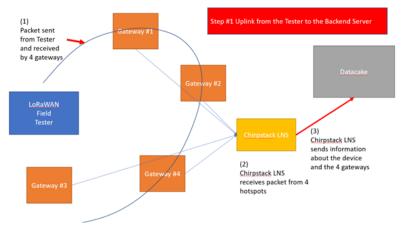


Figure 51: Step 1 - Field Tester Sending Uplink Payload

In the second step, the backend server is calculating the minimum and maximum distance between the Field Tester Pro and the gateways that received the data. Together with the minimum and maximum RSSI levels, this information is then sent back to the Field Tester Pro as a LoRaWAN downlink.

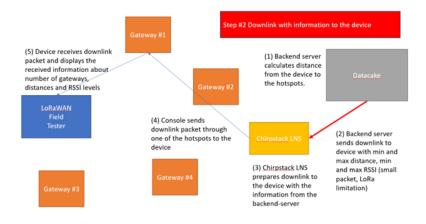


Figure 52: Step 2 - Backend Server Sending Useful Information as Downlink

To use Chirpstack for RAK10701-P, you must have a working installation of the Chirpstack LoRaWAN network server. It can be on a dedicated machine, Raspberry Pi, or in a cloud VPS instance. It should have a fixed IP address and port to where Datacake.co will connect to.

To start with Chirpstack, you must create a device profile for your RAK10701-P Field Tester Pro device. You
must select LoRaWAN MAC version 1.0.3 which is the LoRaWAN specification version that the RAK10701
Field Tester supports.

∉	ChirpStack	Q Search organization, application, gateway or device ?	@RAKwireless.com
ŧ	Dashboard	Device-profiles / LoRaWAN-Field-Tester-AS923-3	DELETE
8 1 8 1	Network-servers		
R	Gateway-profiles	GENERAL JOIN (OTAA / ABP) CLASS-B CLASS-C CODEC TAGS	
	Organizations	Device profile name *	
:	All users	LoRaWAN-Field-Tester-AS923-3	
٩	API keys	A name to identify the device-profile.	
Bee	Gee-Chirpstack 👻	1.0.3 The LoRaWAN MAC version supported by the device.	· · · · · · · · · · · · · · · · · · ·
A	Org. dashboard	LoRaWAN Regional Parameters revision * RP002-1.0.3	··· ·
*	Org. users	Revision of the Regional Parameters specification supported by the device.	
٩	Org. API keys	ADR algorithm * Default ADR algorithm (LoRa only)	.
•=	Service-profiles	The ADR algorithm that will be used for controlling the device data-rate.	

Figure 53: Creating Device Profile in Chirpstack

2. You must enable Device supports OTAA as the network join method as well.

€	ChirpStack		Q Search organizatio		?	Θ	@RAKwireless.com
↑	Dashboard Network-servers	Device-profiles / LoRaWAN-Field-Tester-AS923-3					DELETE
R	Gateway-profiles	GENERAL JOIN (OTAA / ABP) CLASS-B	CLASS-C	CODEC TAG	s		
	Organizations						
÷	All users	Device supports OTAA					
٩	API keys					UPDA	TE DEVICE-PROFILE
Bee	Gee-Chirpstack 👻						
f	Org. dashboard						
•	Org. users						
٩	Org. API keys						
* =	Service-profiles						
<u></u>	Device-profiles	•					

Figure 54: Enable support for OTAA

3. You can also include a custom javascript decoder under the **CODEC** tab. This will allow you to see the specific information transmitted by the device.

€	ChirpStack	Q Search organization, application, gateway or device	@RAKwireless.com
ŧ	Dashboard	Device-profiles / LoRaWAN-Field-Tester-AS923-3	DELETE
81 81	Network-servers	Device-profiles / Longwale-Field-Testel-A5925-3	
\bigcirc	Gateway-profiles	GENERAL JOIN (OTAA / ABP) CLASS-B CLASS-C CODEC TAGS	
	Organizations	Payload codec	
•	All users	Custom JavaScript codec functions By defining a payload codec, ChilpStack Application Server can encode and decode the binary device payload for you.	•
٩	API keys	1 // Decode decodes an array of bytes into an object.	*
Bee	Gee-Chirpstack 👻	<pre>2 // - fPort contains the LoRaWAN fPort number 3 // - bytes is an array of bytes, e.g. [225, 230, 255, 0] 4 // - variables contains the device variables e.g. {"calibration": "3.5"} (both the key / value are of type string)</pre>	- I
ħ	Org. dashboard	<pre>// The function Decode(fPort, bytes, variables) { // var decoded = {}; // var decoded =</pre>	
•	Org. users	<pre>8 // avoid sending Downlink ACK to integration (Cargo) 9 if ((fPort == 1) (fPort == 2)){</pre>	
٩	Org. API keys	var lonSign = (bytes[0] >> 7) & 0x01 ? -1 : 1; uar latSign = (bytes[0] >> 6) & 0x01 ? -1 : 1;	
. =	Service-profiles	2 3 var encLat + ((bytes[0] & 0x3f) << 17) + (bytes[1] << 9) +	

Figure 55: Custom Javascript Decoder for RAK10701 Field Mapper

Here's the complete decoder script:

```
function Decode(fPort, bytes, variables) {
   var decoded = {};
    // avoid sending Downlink ACK to integration (Cargo)
    if ((fPort === 1) || (fPort === 2)){
        var lonSign = (bytes[0] >> 7) & 0x01 ? -1 : 1;
        var latSign = (bytes[0] >> 6) & 0x01 ? -1 : 1;
        var encLat = ((bytes[0] & 0x3f) << 17) +</pre>
            (bytes[1] << 9) +
            (bytes[2] << 1) +
            (bytes[3] >> 7);
        var encLon = ((bytes[3] & 0x7f) << 16) +</pre>
            (bytes[4] << 8) +
            bytes[5];
        var hdop = bytes[8] / 10;
        var sats = bytes[9];
        var maxHdop = 2;
        var minSats = 5;
        if ((hdop < maxHdop) && (sats >= minSats)) {
            decoded.latitude = latSign * (encLat * 108 + 53) / 10000000;
            decoded.longitude = lonSign * (encLon * 215 + 107) / 10000000;
            decoded.altitude = ((bytes[6] << 8) + bytes[7]) - 1000;</pre>
            decoded.accuracy = (hdop * 5 + 5) / 10
            decoded.hdop = hdop;
            decoded.sats = sats;
            decoded.error = "Need more GPS precision (hdop must be <" + maxHdop +</pre>
                " & sats must be >= " + minSats + ") current hdop: " + hdop + " & sats:" + sats;
            decoded.latitude = latSign * (encLat * 108 + 53) / 10000000;
            decoded.longitude = lonSign * (encLon * 215 + 107) / 10000000;
            decoded.altitude = ((bytes[6] << 8) + bytes[7]) - 1000;</pre>
            decoded.accuracy = (hdop * 5 + 5) / 10
            decoded.hdop = hdop;
            decoded.sats = sats;
        return decoded;
```

VOTE:

This decoder script can be found on RAKwireless Standardize Payload repository 🗗 which also includes a custom decoder script for TTN and Helium.

4. After creating the device profile, you can now create an application and add the RAK10701 device. And then attached the Device-profile you created. You have to take note of the DEVEUI and APPKEY in this section. These parameters must match the ones in our RAK10701 Field Tester.

€	ChirpStack	Q Search organization, application, gateway or device ? e @RAKwireless.com
A	Dashboard Network-servers	Applications / Create
@ ==	Gateway-profiles Organizations	Application name * RAK_Las_Pinas_AS923_3_Test_App
• «	All users API keys	The name may only contain words, numbers and dishes. Application description * AS923-3 Test Application
Bee	See-Chirpstack 👻	Service-profile * as923_3_service-profile-build-in The service-profile to which this application will be attached. Note that you can't change this value after the application has been created.
↑	Org. dashboard Org. users	CREATE APPLICATION
∿ ∡≡	Org. API keys Service-profiles	
밵	Device-profiles	
@ 	Gateways Applications	

Figure 56: Create application in Chirpstack

æ	ChirpStack	Q Search organization, application, gateway or device 3	@RAKwireless.com
•	Dashboard Network-servers	Applications / RAK_Las_Pinas_AS923_3_Test_App / Devices / Create	
@ ==	Gateway-profiles Organizations	GENERAL VARIABLES TAGS	
•	All users API keys	LoRaWAN-Field-Tester-08EB0C The name may only contain words, numbers and dashes. Device description *	
Bee	Gee-Chirpstack 👻	LoRaWAN-Field-Tester-08EB0C	
A	Org. dashboard	d 86 bb 20 40 f4 c0 d4	msb C
•	Org. users	Device-profile * LoRaWAN-Field-Tester-AS923-3	
٩	Org. API keys		
* =	Service-profiles	Disable frame-counter validation Note that disabling the frame-counter validation will compromise security as it enables people to perform replay-attacks.	
	Device-profiles	Device is disabled	
R	Gateways	ChirpStack Network Server will ignore received uplink frames and join-requests from disabled devices.	
	Applications		CREATE DEVICE

Figure 57: Create device in Chirpstack.

€	ChirpStack	Q Search organization, application, gateway or device ? 🕒	@RAKwireless.com
ŧ	Dashboard	Applications / RAK_Las_Pinas_AS923_3_Test_App / Devices / LoRaWAN-Field-Tester-08EB0C	DELETE
81 81 81	Network-servers		
\bigcirc	Gateway-profiles	DETAILS CONFIGURATION KEYS (OTAA) ACTIVATION DEVICE DATA LORAWAN FRAMES	
	Organizations		
•	All users	Application key*	
٩	API keys	For LoRaWAN 1.0 devices. In case your device supports LoRaWAN 1.1, update the device-profile first.	Ø
Bee	Gee-Chirpstack 👻		SET DEVICE-KEYS
♠	Org. dashboard		
•	Org. users		
٩	Org. API keys		
* =	Service-profiles		
Ŧ	Device-profiles		
\bigcirc	Gateways		
	Applications		

Figure 58: Device APPKEY

5. You also need to secure that you have a Gateway registered in Chirpstack and with the correct Network Server profile.

⇐	ChirpStack			Q Search organ	ization, application, gatew	vay or device ? O @RAKwireless.com
ŧ	Dashboard	Gateways				+ CREATE
	Network-servers	cutomayo				
\bigcirc	Gateway-profiles	Laters	News	O stanov ID	Network server	
	Organizations	Last seen	Name	Gateway ID	Network server	Gateway activity (30d)
•	All users	3 months ago	Differe Octowy	C01f00fff0052058	as923_3_build_in_ns	
٩	API keys	19 days ago	RAX2240 20000	Loz7ul/ffteotobo5	eu868_build_in_ns	.6
BeeG	ee-Chirpstack 👻	15 days ago	PAM7050 A0020 3	():[-0446744092	as923_3_build_in_ns	. 1
♠	Org. dashboard	a minute ago	RAN72000 V2	601/00///002004	as923_3_build_in_ns	
•	Org. users	a few seconds ago	PAK7252 V1	confeefff-eed (C20	as923_3_build_in_ns	
٩,	Org. API keys	a minute ago	RAKT200V2 OUI	uc100776002168	us915_0_build_in_ns	
.≡	Service-profiles	5 months ago	PAX7220	cotfoofff.cottuct	as923_3_build_in_ns	
	Device-profiles					Rows per page: 10 - 1-7 of 7 < >
R	Gateways					nows per page. To 👻 T-7 01 7 🔍 🤉
	Applications					

Figure 59: Gateways registered in Chripstack

6. The next step after setting up the network server, devices, and gateway, is the integration of the Chirpstack application to Datacake. You must choose HTTP, then click Edit . Then you have to use this endpoint going to datacake https://api.datacake.co/integrations/lorawan/chirpstack/.

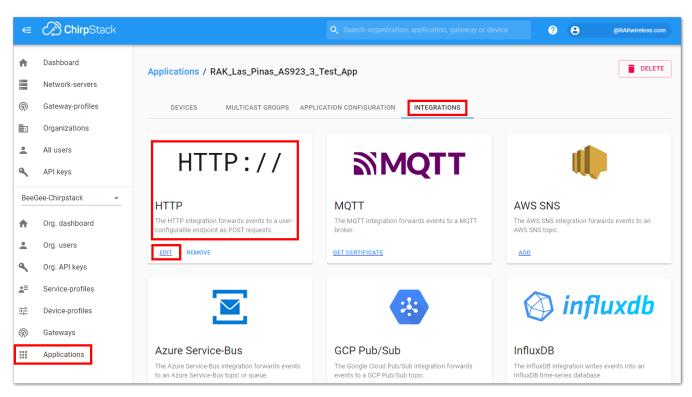


Figure 60: Creating integration

€	ChirpStack	Q Search organization, application, gateway or device ? O @RAKwireless.com
A	Dashboard	Applications / RAK_Las_Pinas_AS923_3_Test_App
	Network-servers	
R	Gateway-profiles	DEVICES MULTICAST GROUPS APPLICATION CONFIGURATION
	Organizations	
•	All users	Update HTTP integration
٩	API keys	Payload marshaler *
Bee	Gee-Chirpstack 👻	JSON This defines how the payload will be encoded.
A	Org. dashboard	Headers
*	Org. users	Header name Header value
٩	Org. API keys	ADD HEADER
≞ ≡	Service-profiles	Endpoints
코는	Device-profiles	Endpoint URL(s) for events https://api.datacake.co/integrations/lorawan/chirpstack/
R	Gateways	ChirpStack will make a POST request to this URL(s) with 'event' as query parameter. Multiple URLs can be defined as a comma separated list. Whitespace will be automatically removed.
	Applications	UPDATE INTEGRATION

Figure 61: Creating endpoint for Datacake.co

7. The last step on the Chirpstack side, is the creation of the API key. This is needed to allow Datacake in sending downlink packets to the RAK10701 Field Tester. Make sure the key is copied and saved somewhere, it is only retrievable during the key creation. Copy the Token and save it in a text editor.

∉	ChirpStack		Q Search organization, application, gateway or device	e @ra	Kwireless.com
A	Dashboard	Global API keys			+ CREATE
	Network-servers			-	
R	Gateway-profiles	ID	Name		
E	Organizations				
:	All users	Suddocho 0400 4000 u700 04017 0669 u5	Datacake		
٩	API keys		Rows per page: 10	0 ▼ 1-1 of 1	< >
Bee	Gee-Chirpstack 👻				
A	Org. dashboard				
+	Org. users				
٩	Org. API keys				
. =	Service-profiles				
	Device-profiles				
R	Gateways				
	Applications				

Figure 62: Creation of API Key

€	ChirpStack		Q Search organization, application, gateway or device	?	@RAKwireless.com
ŧ	Dashboard	Global API keys / Create			
	Network-servers	ologar Ar rikeys / oreate			
\bigcirc	Gateway-profiles				
₽	Organizations	API key name * A descriptive name for the API key			
•	All users	A descriptive name for the AFT key			
٩	API keys				CREATE API KEY
Bee	Gee-Chirpstack 👻				
♠	Org. dashboard				
•	Org. users				
٩	Org. API keys				
	Service-profiles				
프는	Device-profiles				
R	Gateways				
	Applications				

Figure 63: Creation of API Key

8. You can now proceed on Datacake and add a LoRaWAN device that will be linked to your RAK10701 created in ChirpStack. You have to create an account if you do not have one yet.

RAK @rakwirel	DATACAKE			
	Fleet > Devices			
+ Add Dashboard 🚳	Devices		Q Search	Columns 🗸 + Add Device
Devices Reports	E List			
Augusto Members	DEVICE F	PRIMARY SECONDARY	DEVICE SIGNAL	DEVICE BATTERY
	• fall total and the •	N/A N/A	N/A	N/A 💿 :
Workspace Integrations 	Showing 1 to 1 of 1 results			50 per page v Previous Next
 White Label Billing 				
- Shing				

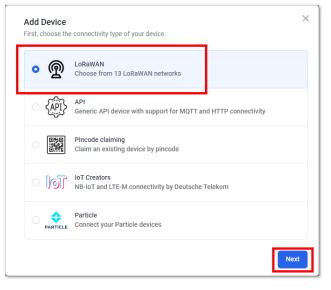


Figure 65: Selecting LoRaWAN

STEP 1 Product	STEP 2 Network Server	STEP 3 Devices	STEP 4 Plan
	to an existing product on . Products allow you to sl		ew empty product or start wit ıration (fields, dashboard and
New Product fro template Create new prod from a template	m Existing Add dev	Product ices to an product	New Product Create new empty product
			mpty device. You will have to oad decoder in the device's
rak1070x			

Figure 66: Add new product

STEP 1 Product	STEP 2 Network Server	STEP 3 Devices	STEP 4 Plan
Vetwork Service the service of the s	ver ne LoRaWAN Network Server 1	that your devices are	connected to.
THE THINGS	The Things Stack V3 TTN V3 / Things Industrie	:5	Uplinks Downlinks
🔵 🧭 heliur	n Helium		Uplinks Downlinks
	LORIOT		Uplinks Downlinks
 ChipStack 	ChirpStack		Uplinks Downlinks
🔿 🔏 Actility	y Actility		Uplinks Downlinks
Showing 1 to	5 of 13 results		Previous Next

Figure 67: Selecting chirpstack

Product	STEP 2 Network Server	STEP 3 Devices	STEP 4 Plan
Add Devices			
Enter one or mor	e LoRaWAN Device EUIs and t	he names they will ha	ve on Datacake.
New: You can no	w upload a CSV file with either	one column (just the	device's DevFUI) or two
	I and Name), which will popula		
(
	Drag and drop a .csv f	file here or click to cho	oose one
·			
DEVEUI		NAME	
	09 FF FF 08 FB 0C 8 bytes		d
	09 FF FE 08 EB 0C 8 bytes	NAME	4

Figure 68: Adding device name

9. Next step is to enable the downlink. This is the step where you'll be needing the previously created API key in step 7 of this guide. Take note that the ChirpStack URL should be based on your deployed Chirpstack network server. After doing all configurations, click update and save.

RAK @rakwirel	Ø DATACAKE	
	Fleet > field-tester-unit-white	
+ Add Dashboard 🛞	field toster unit white	
📧 Devices	Serial Number Last update Tags	
Reports	59513A12348528AE Mon Oct 24 2022 07:59:54 GMT+0800	
Se Members	🖽 Dashboard 📊 History 🚎 Downlinks 🙀 Configuration 🔊 Debug 🦷 Rules 🐇 Permissions	
€ ⁴ Rules		
Workspace	General Configuration	
+ Integrations	Device Name	
🗞 White Label		
🖄 Billing	Icon	
	No icon selected 🐱	
	You can override the default product icon for this device on the map view.	
	Location Description Opt	tional

Figure 69: Configuration settings

RAK @rakwirel		Save
+ Add Dashboard 🛞	LoRaWAN	Show setup instructions
	DevEUI	
Reports	(m) 100 11 101 100 100 100 100	Change
යි Members		
€ ⁴ Rules	Network Server	
Workspace	ChirpStack	Change
 ↔ Integrations ◇ White Label 		
🖄 Billing	Authenticate Uplink Webhooks Learn more about securing the webhook	Product-wide setting
		Save
	Payload Decoder	Product-wide setting

Figure 70: Chirpstack downlink configuration

● @cnupStack ChirpStack	
Actility	
Showing 1 to 5 of 13 results	Previous Next
Jplink URL	
https://api.datacake.co/integrations/lorawan/ch	nirpstack/
Configure your LoRaWAN Network Server to send data to t	his URL
ChirpStack URL	Product-wide setti
http://ciwi.tk:8082	
ne URL of your Chirpstack server without a trailing slash,	e.g. https://mychirpstack.com

Figure 71: Chirpstack URL and API key

10. Just below the LoRawAN section in datacake.co, you'll see the Payload Decoder. This is a very critical step to ensure that all important data will be covered.

```
function distance(lat1, lon1, lat2, lon2) {
    if ((lat1 == lat2) && (lon1 == lon2)) {
       return 0;
        var radlat1 = Math.PI * lat1 / 180;
       var radlat2 = Math.PI * lat2 / 180;
       var theta = lon1 - lon2;
       var radtheta = Math.PI * theta / 180;
        var dist = Math.sin(radlat1) * Math.sin(radlat2) + Math.cos(radlat1) * Math.cos(radlat2)
        if (dist > 1) {
            dist = 1;
        dist = Math.acos(dist);
        dist = dist * 180 / Math.PI;
        dist = dist * 60 * 1.1515;
       dist = dist * 1.609344;
        return dist;
function Decoder(bytes, fPort) {
   var decoded = {};
   // avoid sending Downlink ACK to integration (Cargo)
   if (fPort === 1) {
        var lonSign = (bytes[0] >> 7) & 0x01 ? -1 : 1;
        var latSign = (bytes[0] >> 6) & 0x01 ? -1 : 1;
        var encLat = ((bytes[0] & 0x3f) << 17) +</pre>
            (bytes[1] << 9) +
            (bytes[2] << 1) +
            (bytes[3] >> 7);
        var encLon = ((bytes[3] & 0x7f) << 16) +</pre>
            (bytes[4] << 8) +
            bytes[5];
        var hdop = bytes[8] / 10;
        var sats = bytes[9];
        var maxHdop = 2;
        var minSats = 5;
        if ((hdop < maxHdop) && (sats >= minSats)) {
            decoded.latitude = latSign * (encLat * 108 + 53) / 10000000;
            decoded.longitude = lonSign * (encLon * 215 + 107) / 10000000;
            decoded.altitude = ((bytes[6] << 8) + bytes[7]) - 1000;</pre>
            decoded.accuracy = (hdop * 5 + 5) / 10
            decoded.hdop = hdop;
            decoded.sats = sats;
            decoded.location = "(" + decoded.latitude + "," + decoded.longitude + ")";
            decoded.error = "Need more GPS precision (hdop must be <" + maxHdop +</pre>
                " & sats must be >= " + minSats + ") current hdop: " + hdop + " & sats:" + sats;
            decoded.latitude = latSign * (encLat * 108 + 53) / 10000000;
            decoded.longitude = lonSign * (encLon * 215 + 107) / 10000000;
            decoded.altitude = ((bytes[6] << 8) + bytes[7]) - 1000;</pre>
            decoded.accuracy = (hdop * 5 + 5) / 10
            decoded.hdop = hdop;
            decoded.sats = sats;
            decoded.location = "(" + decoded.latitude + "," + decoded.longitude + ")";
```

```
decoded.raw = rawPayload.uplink_message.rx_metadata[0].location;
decoded.num_gw = normalizedPayload.gateways.length;
decoded.minRSSI = 0;
decoded.maxRSSI = 0;
decoded.minSNR = 0;
decoded.maxSNR = 0;
decoded.minDistance = 0;
decoded.maxDistance = 0;
var server_type = 0;
if (typeof (rawPayload.uplink_message) != "undefined") {
    console.log("Found TTN format");
    server_type = 1;
else if (typeof (rawPayload.hotspots) != "undefined") {
    console.log("Found Helium format");
    server_type = 2;
else if (typeof (rawPayload.rxInfo) != "undefined") {
    console.log("Found Chirpstack format");
    server_type = 3;
    decoded.is_chirpstack = 1;
    console.log("Unknown raw format");
var gw_lat = {};
var gw_long = {};
decoded.num_gw = 0;
for (idx_tst = 0; idx_tst < 10; idx_tst++)</pre>
    if (typeof (normalizedPayload.gateways[idx_tst]) != "undefined")
        console.log("Found gateway with IDX " + idx_tst);
        decoded.num_gw += 1;
for (idx = 0; idx < decoded.num_gw; idx++) {</pre>
    var new_rssi = (!!normalizedPayload.gateways && !!normalizedPayload.gateways[idx] &&
    var new_snr = (!!normalizedPayload.gateways && !!normalizedPayload.gateways[idx] &&
    if ((new_rssi < decoded.minRSSI) || (decoded.minRSSI == 0)) {</pre>
        decoded.minRSSI = new_rssi;
    if ((new_rssi > decoded.maxRSSI) || (decoded.maxRSSI == 0)) {
        decoded.maxRSSI = new_rssi;
    }
    if ((new_snr < decoded.minSNR) || (decoded.minSNR == 0)) {</pre>
        decoded.minSNR = new_snr;
    if ((new_snr > decoded.maxSNR) || (decoded.maxSNR == 0)) {
        decoded.maxSNR = new_snr;
    // var gw_lat = 0.0;
    // var gw_long = 0.0;
    switch (server_type) {
```

```
gw_lat[idx] = rawPayload.uplink_message.rx_metadata[idx].location.latitude;
            gw_long[idx] = rawPayload.uplink_message.rx_metadata[idx].location.longitude
            break;
        case 2:
            gw_lat[idx] = rawPayload.hotspots[idx].lat;
            gw_long[idx] = rawPayload.hotspots[idx].long;
            break:
        // Chirpstack
        case 3:
            gw_lat[idx] = rawPayload.rxInfo[idx].location.latitude;
            gw_long[idx] = rawPayload.rxInfo[idx].location.longitude;
            break:
        default:
            console.log("Unknown LNS");
   console.log("IDX " + idx + " lat " + gw_lat[idx] + " long " + gw_long[idx]);
    // decoded.gw_lat[idx] = gw_lat;
   // decoded.gw_long[idx] = gw_long;
   var new_distance = distance(gw_lat[idx], gw_long[idx], decoded.latitude, decoded.long
    if ((new_distance < decoded.minDistance) || (decoded.minDistance == 0)) {
        decoded.minDistance = new_distance * 1000;
    }
    if ((new_distance > decoded.maxDistance) || (decoded.maxDistance == 0)) {
        decoded.maxDistance = new_distance * 1000;
switch (decoded.num_gw) {
   case 20:
        decoded.hotspot_10 = "(" + gw_lat[19] + "," + gw_long[19] + ")";
   case 19:
        decoded.hotspot_09 = "(" + gw_lat[18] + "," + gw_long[18] + ")";
   case 18:
        decoded.hotspot_08 = "(" + gw_lat[17] + "," + gw_long[17] + ")";
        decoded.hotspot_07 = "(" + gw_lat[16] + "," + gw_long[16] + ")";
   case 16:
        decoded.hotspot_06 = "(" + gw_lat[15] + "," + gw_long[15] + ")";
        decoded.hotspot_05 = "(" + gw_lat[14] + "," + gw_long[14] + ")";
   case 14:
        decoded.hotspot_04 = "(" + gw_lat[13] + "," + gw_long[13] + ")";
        decoded.hotspot_03 = "(" + gw_lat[12] + "," + gw_long[12] + ")";
   case 12:
        decoded.hotspot_02 = "(" + gw_lat[11] + "," + gw_long[11] + ")";
        decoded.hotspot_01 = "(" + gw_lat[10] + "," + gw_long[10] + ")";
   case 10:
        decoded.hotspot_10 = "(" + gw_lat[9] + "," + gw_long[9] + ")";
   case 9:
        decoded.hotspot_09 = "(" + gw_lat[8] + "," + gw_long[8] + ")";
        decoded.hotspot_08 = "(" + gw_lat[7] + "," + gw_long[7] + ")";
```

decoded.hotspot_07 = "(" + gw_lat[6] + "," + gw_long[6] + ")";

```
case 6:
        decoded.hotspot_06 = "(" + gw_lat[5] + "," + gw_long[5] + ")";
    case 5:
        decoded.hotspot_05 = "(" + gw_lat[4] + "," + gw_long[4] + ")";
    case 4:
        decoded.hotspot_04 = "(" + gw_lat[3] + "," + gw_long[3] + ")";
    case 3:
        decoded.hotspot_03 = "(" + gw_lat[2] + "," + gw_long[2] + ")";
        decoded.hotspot_02 = "(" + gw_lat[1] + "," + gw_long[1] + ")";
    case 1:
        decoded.hotspot_01 = "(" + gw_lat[0] + "," + gw_long[0] + ")";
decoded.maxMod = parseInt((decoded.maxDistance / 250), 10);
decoded.minMod = parseInt((decoded.minDistance / 250), 10);
decoded.maxDistance = parseInt((decoded.maxMod * 250), 10);
decoded.minDistance = parseInt((decoded.minMod * 250), 10);
if (decoded.maxDistance <= 1) {</pre>
    decoded.maxDistance = parseInt(250, 10);
}
if (decoded.minDistance <= 1) {</pre>
    decoded.minDistance = parseInt(250, 10);
return decoded;
```

This decoder is not only decoding data from the LoRaWAN packet but is as well reading gateway information from the additional data that the LoRaWAN server added to the data it forwarded to Datacake.

Each LoRaWAN server uses a different format for this additional information, so there is a code section that tries to detect whether the data came from a Chirpstack LSN, from TTN, or from a Helium Console:

```
js
var server_type = 0;
// Check if payload comes from TTN
if (typeof (rawPayload.uplink_message) != "undefined") {
    console.log("Found TTN format");
    server_type = 1;
}
// Check if payload comes from Helium
else if (typeof (rawPayload.hotspots) != "undefined") {
    console.log("Found Helium format");
    server_type = 2;
}
// Check if payload comes from Chirpstack
else if (typeof (rawPayload.rxInfo) != "undefined") {
    console.log("Found Chirpstack
else if (type = 3;
    decoded.is_chirpstack = 1;
}
else {
    console.log("Unknown raw format");
}
```

Once the data is extracted, it calculates the distance between the RAK10701 Field Tester location and the different gateways that received the LoRaWAN packet. This version of the decoder can handle up to 10 gateways, but it can be extended.



In the next step, it analyzes the different distances and RSSI levels to find the closest and farthest gateway and the lowest and highest RSSI and SNR levels.

The result of the decoding is then put into different data fields that are used by Chirpstack for the visualization and by the rule, we will define to create the downlink to the RAK10701 Field Tester.

11. The next step is to create the different data fields that are filled by the data decoder. This is done in the Fields section of the device configuration, just below the data encoder section.

Carl RAK carl.rowan@rakwirel	Fields Fields describe the data th	he device will store.					+ Add Field
+ Add Dashboard 🛞	NAME	IDENTIFIER	TYPE	ROLE	CURRENT VALUE	LAST UPDATE	
 Devices 	Field tester	LOCATION	Location	N/A			:
Reports	MinDistance	MINDISTANCE	Integer	N/A	•••		
€ ⁴ Rules	MaxDistance	MAXDISTANCE	Integer	N/A		•••	(1)
Workspace Integrations	MinRSSI	MINRSSI	Float	N/A	***		(1)
🟷 White Label	MaxRSSI	MAXRSSI	Float	N/A	***		
🖄 Billing	MinSNR	MINSNR	Float	N/A	***		
	MaxSNR	MAXSNR	Float	N/A	***		
	hotspot_01	HOTSPOT_01	Location	N/A			
	hotspot_02	HOTSPOT_02	Location	N/A			:
	hotspot_03	HOTSPOT_03	Location	N/A			:

Figure 72: Creation of data field

The following fields are required:

Field Name	Identifier (returned value from the decoder)	Field Type
Field Tester	LOCATION	Location
MinDistance	MINDISTANCE	Integer
MaxDistance	MAXDISTANCE	Integer
MinRSSI	MINRSSI	Float
MaxRSSI	MAXRSSI	Float
MinSNR	MINSNR	Float
MaxSNR	MAXSNR	Float
minMod	MINMOD	Integer
maxMod	MAXMAD	Integer
hotspot_01	HOTSPOT_01	Location
hotspot_02	HOTSPOT_01	Location
hotspot_0	HOTSPOT	Location
hotspot_09	HOTSPOT_09	Location
hotspot_10	HOTSPOT_10	Location
is_chirpstack	IS_CHIRPSTACK	Boolean

There are more variables created by the decoder, but this is the minimum set required for the dashboard and to configure the download.

12. This is the most critical step so that the RAK10701 will be able to display the necessary details helpful in Field Testing the LoRaWAN network. This section is responsible for the "backend-server functions". In this step, we create the automatic downlink to the device that is executed every time a data packet from the RAK10701 Field Tester arrives. Take note that port number 2 is used by RAK10701 for downlinks and Trigger on measurements should be checked.

Carl RAK carl.rowan@rakwirel	Ø DATACAKE						
Can.rowan@rakwirei	Fleet > field-tester-unit-white						
+ Add Dashboard 🛞	field-tester-unit-white						
	Serial Number Last update Tags						
Reports	59513A12348528AE Mon Oct 24 2022 07:59:54 GMT+0800						
음일 Members	🖽 Dashboard 📊 History 🌐 Downlinks 🛞 Configuration 🔊 Debug 🣭 Rules 🐇 Permissions						
Workspace	Downlinks + Add Downlink						
- Integrationa							

Figure 73: Chirpstack downlink configuration

Configure Downlink	8
Name	1
field-tester-carl-downlink	
Description	
Field Tester Application Downlink	
If sour encoder function takes input from the device's fields, you can specify them here. They will be used to create the form for the downlink generator IS_CHIRPSTACK Trigger on measurements If activated, each time the device records a measurement in one of the fields used, the downlink will be sent automatically. Port 2	
<pre>Payload Encoder 1 + function Encoder(measurements, port) { 2 var buf = []; 3 buf[0] = 1; 4 buf[1] = measurements.MINNSSI.value + 200; 5 buf[2] = measurements.MINNOU.value + 200; 6 // var temp = parseInt(measurements.MINNOU.value, 0); 7 if (measurements.MINNOU.value = 0) { 8</pre>	

Figure 74: Chirpstack downlink configuration

In this guide, the downlink is only created when the Field Tester is connected through a Chirpstack server. When TTN or Helium is used, the downlink would be created by the original backend server.

This is controlled by the "Fields used" where you can see the IS_CHIRPSTACK.

The Payload Encoder is preparing a downlink packet that will be sent back to the RAK10701 Field Tester. The downlink packet is only 6 bytes large to avoid problems in LoRaWAN regions with limited downlink packet sizes.

This is the complete decoder script. The first byte is usually a counter, but it works well if the counter is ignored and always set to 1.

```
function Encoder(measurements, port) {
    var buf = [];
    buf[0] = 1;
    buf[1] = measurements.MINRSSI.value + 200;
    buf[2] = measurements.MAXRSSI.value + 200;
    // var temp = parseInt(measurements.MINMOD.value,10);
    if (measurements.MINMOD.value == 0) {
        measurements.MINMOD.value == 0) {
            measurements.MINMOD.value = 1;
        }
        console.log(measurements.MINMOD.value;
        if (measurements.MAXMOD.value == 0) {
            measurements.MINMOD.value;
        if (measurements.MAXMOD.value;
        if (measurements.MAXMOD.value;
        if (measurements.MAXMOD.value;
        if (measurements.MAXMOD.value;
        if (measurements.MAXMOD.value;
        if (measurements.NAXMOD.value;
        if (measurements.NAXMOD.value;
        if (measurements.NUM_GW.value;
        buf[4] = measurements.NUM_GW.value;
        return buf;
    }
}
```

13. You can now proceed on device configuration so that the proper EUIs and KEY will match the one in the network server.

RAK10701-P Field Tester Pro Guide for LORIOT and Datacake

In this document, you will find a step-by-step guide for performing a field mapping test using LORIOT network management system and Datacake's platform to visualize your results. This solution will help you in your network planning ventures and ensure your decisions are data-driven and adequate to your surroundings.

Prerequisites

- RAK10701 WisNode Field Tester for LoRaWAN
- Datacake account I III
- Gateway

Setting LORIOT as the LNS

1. Forward a gateway to LORIOT, which will be the LNS (LoRa Network Server) for this use case. For registration of the gateway to LORIOT, you will need the gateway's MAC and EUI, which can be found on the Overview page of WisGateOS 2.

WsGate	RAK7289V2_2359			
	Overview LoRa Statistics			
.≟ ≁	RAK7289V2	918100QG6021 SERIAL NUMBER	Packet capture Record data packets tran network for analysis.	\ominus
ہ *	 AC1F09FFFE EUI AC:1F:09: 	42.885937, 25.311689 REPORTED COORDINATES 868 MHz	Performance	
	MAC ADDRESS 8-CHANNEL NUMBER OF CHANNELS Wed Oct 4 13:57:19 2023 LOCAL TIME	PREQUENCY BAND 5d 23h 4m 1s UPTIME	сии 💽 10 /1005 Ш.Ш.Ш.Ш.Ш.Ш.Ш. Исмору 🕐	
	WisGateOS_2.1.7_RAK	Firmware details	53 /126MB	
٢	WAN Interfaces	Ð)	

Figure 75: WisGate Edges web UI

 Go to your LORIOT profile. From the menu on the left, navigate through Networks>{your_network}>+Add Gateway.

LORIO T	≡ Q Search								<u>د</u>	🗕 i 🔶
 Back to Networks 										
X SAMPLE NETWORK A0 60 SF-30	Sample network Creation Date 3/23/21, 3:40 PM Network ID A000173D + Add Gateway Gateways	K Configure •	Organization User Access Disabled O Network Organization UUD 27aadde8-a802-4518-9772-e342319	-						
	Gateways									
	Name	Q.↓† GWEUI	Q.IT MAC	Q. 11 Model	11 Version	Q.↓† Status	↓† Last Data			11
			0 Ga	ateways						
	Copyright © 2015 - 2023 LORIOT A	G. All rights reserved.						LORIOT I	nternet of Things at Long R	ange Impressum

Figure 76: LORIOT console

NOTE

The LORIOT platform provides you with a **Sample Network** at the point of your profile creation. You can use it for free. If you wish to create a new one, or delete the provided one, you will need a paid plan to continue.

3. For the base platform select **Basics Station Semtech**. You will be asked to provide eth0 MAC address and EUI, which you obtained in step 1. After filling in these values, press the **Register Basics Station Semtech gateway** at the bottom of the page.

HIL LORIOT	≡ Q Search		±	• 0 🔶
Back to Networks Sample Network AD-00-1F-3D	The LOROT gateway binary is custom built for all integrated gateways. Select your gateway from the list below and fill in the required fields.			
🕅 Мар	What is your base platform?			
	Basics Station Semtech	Radio Front-end SI(30) ♥ Bus SPI♥ Basics Station is supported		
				_
	MAC address of eth0 interface			
	The MAC Address of the Ethernet port can be queried by running [ifcortig ethe] gree Haador command from your device's console. A sample output will be similar to ethe Like despit Ethernet: Haador ABCD0F11230456 Copy and paste the highlighted part (six octets separated by colons) from the	output of your device console to the input field below.		
	teh0 MAC address AC:17:09: Upon successful registration, we will provide you with a setup guide for your The keys are field to the MAC address of the device, and cannot be moved to a	Custom EUI ACLPOPETEL gateway and a gateway binary with cryptographic keys field to this MAC address. nother device.		
	Gateway Location			

Figure 77: Registering The Gateway To LORIOT

4. The last thing you need to do to connect your gateway with LORIOT LNS is to provide the Basics Station configuration to the gateway. This can be done by going to the gateway's web UI>LoRa>Configuration and doing a Basics station server setup.

AK Documentation Center

₩s Gate	Configuration	
E Dashboard	Work mode Packet forwarder Basics station Built-in network server	
 √ Diagnostics ✿ Settings 	Log Level DEBUG	
Extensions	Basics station server setup. Configure Basics Station server setup. Basics Station Server Type CUPS Server CUPS Server Server DIR. Server DIR. Authentication Mode TLS Server Authentication TLS Server Authentication	
(root *	Save changes	-

Figure 78: Gateway Configuration Page

You can find the Trust (CA Certificate), the Server URL, and the Server port in LORIOT by navigating to the newly registered **Gateway**>**Certificate**. Use the configuration provided by LORIOT as it may differ from the guide depending on your region.

CORIO T	≡ Q Search		⊥ —
Back to Sample network			
AC-1F-09-FF-FE-	Network Server Details]	
Bevices Activity	Network Server Address wss://eul.loriot.io		
Decation	TLS CA CertificateBEGIN CERTIFICATE	Common Name Dig/Cert Global Root G2	
Latt Traffic	HIDD1JCC+nagAuEBag1QctrxSqc7AgAt7X6gc4qhf65T4UBgicqhic65WBB4Q5FADB HQscCQYDVQQGeHJVU2EVHBKALUECHHHKGInAULCngGs15HKki+HDVQQEEXB3 dlccu2GlnwHLcqu725H5KaHqrQvQQChxdEauBq2yvdCHbbG2HwqBHVdCH v		
Radio	MjAeFw8xHzA4HDExHjAwHD8aFw8zORAx0HUxMjAwHD8aHGExCzA3BgWVBAYTA1VT MRUwEwYDVQQKEwxEawGPQ2VydCB3DmMxGTAXBgWVBA3TEHd3dySkaWdpY2vydC5j		
Spectral Scan	b2exiDAe8gVVB4HTF0Rp22lD2XJ0E6sb23hbc8sb296IEc		
al GPS	TLS Configuration		Generate TLS Certificate
🛓 Software			
Certificate			
HH Log			
🔔 Alerts			
https://eu1.loriot.io/dashboard	Copyright © 2015 - 2023 LORIOT AG. All rights reserved.	000	LORIOT Internet of Things at Long Range Impressum

Figure 79: Certificate

5. If the steps are followed correctly, the gateway should show a **Connected** status.

Adding the Device and LORIOT to Datacake Integration

1. Add the device to LORIOT. In the LORIOT platform, navigate to **Applications**>**{your_appliaction}** and use the **Enroll Device** utility from the menu on the left. Fill out your Device EUI, Join (APP) EUI, and Application Key.

LORIO T	≡ Q Search			
				z — • • • • •
 Back to Applications 	Enroll a new device			
SAMPLEAPP	LoRaWAN* Version	Enrollment Process		
BE-7A-25-15	LoRaWAN®1.0.x	OTAA	~	About Over-the-air activation (OTAA) for LoRaWAN® 1.0:
+ Enroll Device	Location DISABLED ENABLED			Over-the-air activation (OTAA, also known as join or over-the-air join) is a method of associating a device with an application and a network by sending specific association request (join request) over the air. This guarantees the highest possible level of security in LoRaWAM ⁹ .
I Bulk Import	You can define coordinates for static devices enabling this option.			Note: you only need to use the import feature if you have already deployed the device with a specific
- Devices	Details			APPKEY that you cannot change. For all other cases, please use the standard enrollment procedure.
Devices Map	Title	Device EUI	Join EUI	
Output	MAPPERS	DB72BE1633	DB728E1633	
S Output	Description	Application Key		
n- API Data Format		50394C7F2F46E59FD0FD40495D		
Websocket Applications		Device Template		
✓ Statistics				
√ Join Server		Create Another	Enroll Reset	
4 Join Server				
Access Tokens	2			
₩ Log				
📥 Downloads				
	Copyright © 2015 - 2023 LORIOT AG. All rights reserved.	6	0 0	LORIOT Internet of Things at Long Range Impressum

Figure 80: Adding The Device To The LORIOT Platform

📝 NOTE

The LORIOT platform provides you with a **Sample Application** at the point of your profile creation. You can use it for free. If you wish to create a new one or delete the provided one, you will need a paid plan to continue.

2. Use the Output utility to set up the Datacake integration. For now, just give it a name. The Authorization requires additional settings that will not be covered by this guide. For more information regarding this process, refer to Datacake's guide 🗅 .

IN LORIO T	■ Q Search					<u> </u>	• 0	⇔
 Back to Applications 	Application Output / BE7A2	515						
BE-7A-25-15	Choose output type							
+ Enroll Device	Where should we feed your IoT dat	a? See the Application Output c	atalog for details					1
🗮 Bulk Import	1	Ö	HTTP://	Datacake				. 1
- Devices	Amazon AWS IoT	IBM Cloud	HTTP Push	Mechanism Delivery through 3rd party HTTP	PS cloud service			. 1
Devices Map	-	~		Setup guide				
Output	IronLio IronMa ⁰¹	IronMo ^{1/-}	pn	1. Create an account at Datacake				
n- API Data Format	Iron.io IronMQ v1	Iron.io IronMQ v3	PubNub 3rd party	2. Follow the steps described at LORIOT Public Documentation				
Websocket Applications Statistics	TLS Socket	WebSocket Longration	Acure IoT Hub Acure IoT Hub Ind party	Setup parameters output Name Custom "Authorization" header value	Datacake Custom "Authorization" header value			
Access Tokens Ku Log Downloads	MQIT	Cayenne myDevices Cayenne Endimety	Google IoT Core	Add Output	L			L.
	Althingstub bhingstub bhingstub	CoAP CoAP Push Coarts Datacake	Constant of Constant of Constant of Constant of					

Figure 81: Add Output

3. Now, you need to add the device in Datacake. To register a new device, navigate to the **Devices** tab in your Datacake account. Click the **+Add Device** button.

N :	Ø DATACAKE
· ·	Fleet > Devices
l≃ tgutgf	Devices Q, Search Settings 🗸 + Add Device
+ Add Dashboard 🛞	E List 🕕 Orid 🔟 Map
Devices	
Reports	Showing 0 to 0 of 0 results S0 per page V Previous Next
🐣 Members	
€ ⁴ Rules	
Workspace	
Integrations	
S White Label	
Billing	
89 Add-Ons	
Documentation Changelog	

Figure 82: Datacake Platform

4. Choose **New Product** under **Datacake Product**. Enter the device name in the **Product name** input box, and proceed by clicking **Next**.

STEP 1 Product	STEP 2 Network	Server	STEP 3 Devices	STEP 4 Plan	
Datacake Produc	t				
			a new empty product or st d more) between devices.	art with one of the templates. Product	S
	carrie consignation (
New Product from		Existing Prod		New Product	
Create new produc template	ct from a	Add devices to an existing product Create new empty product		Create new empty product	
New Product					
	ailable as a template	, you can start with a	an empty device. You will h	ave to create the device definition (field	ds,
dashboard) and provid	le the payload decode	er in the device's cor	figuration.		
Product Name					
Mapper					
	-				

Figure 83: Add Lorawan Device 1

5. Select LORIOT as the Network Server and click Next.

rep 1 roduct	STEP 2 Network Server	STEP 3 Devices	STEP 4 Plan	
etwork Serv lease choose th	er e LoRaWAN Network Server that your dev	vices are connected to.		
O THE THINGS	The Things Stack V3 TTN V3 / Things Industries		Uplinks	Downlinks
🔿 🧭 helium	Helium Use your own console		Uplinks	Downlinks
ି (୩୬ ୭	Managed Helium Powered by IoT Creators			Uplinks
O LORIOT	LORIOT		Uplinks	Downlinks
ChipStack	ChirpStack		Uplinks	Downlinks
Showing 1 to	5 of 16 results		Previous	Next

Figure 84: Add Lorawan Device 2

- 6. On the next page, you will have to enter the name and DEVEUI of the device. Select the plan for Datacake according to your needs and finish the device-adding procedure.
- 7. Now, navigate to **Configuration** in the newly created device at Datacake and scroll down to the **Network Server** configuration. Click **Change**.

		Save	• •
l≏* tgutgf	Product Configuration		
+ Add Dashboard 🛞	Icon No icon selected		
E: Devices	No icon selected ~ to icon selec		
Reports	Product Slug		
89 Members	mapper-1	🗎 Сору	
₿ ⁴ Rules	Used by the MQTT broker		
Workspace Workspace ■		Save	
Integrations			
🛇 White Label			
🖄 Billing	LoRaWAN	Show setup instructions	
8° Add-Ons	DevEll		
	06 72 86 16 33	Change	
	Network Server		
	LORIOT Lorio T Lorio T Lorio T Lorio Configured	Change	
	Authenticate Uplink Webhooks Learn more about securing the webhook	Product-wide setting	
		Save	
Documentation			
H Changelog			* *

8. You will need the LORIOT Access Token, which is generated from the LORIOT console. Navigate to **LORIOT>Access Tokens** and copy the token to put it in Datacake.

LORIO T	≡ Q Search	L 0 🔅
 Back to Applications 	Application Access Tokens	
E-7A-25-15	Security mechanism To connect to any of the application interfaces provided by LORIOT.io, you need to prove you are a legitimate user of the network application.	Authentication Tokens ・
tenroll Device Bulk Import	The public network is built for developers with ease of use in mind, so we don't require any complicated (but more secure) authentication mechanisms. The only mechanism used is a security taken (per-application). You will need to provide this taken before any other interactions with the interface. Note that anyone with knowledge of the token can access your data, so please keep the tokens as protected any of your passwords.	Generate another authentication token
- Devices	If you require a more secure authentication mechanism, please contact our 🕿 sales department.	
 Output API Data Format 	NEW! Updated token format With the latest update of our software, the format of the tokens has been changed. The token has been extended and now includes information about the app You can continue using any legacy tokens you have already in place, we provide full backward compatibility.	plication ID and the server origin, so that 3rd party platforms can use single value copy-paste to access our API.
Websocket Applications <u> <u> </u></u>		yaW90LmlvzXTUl6ESlOrvJST-
Join Server Access Tokens	gs	L_xQ==
₩ Log	basef4 wit	h special characters replaced
	$\label{eq:information:appld} \begin{array}{cc} \mbox{server UR}, \mbox{length} \\ \mbox{Length:} & 4 \left(0,4 \right) & 4 \left(4,8 \right) & \times \left(8,8 \right) \\ \mbox{Format: } \mbox{base64} <> \mbox{hex} <> \mbox{integer} & \mbox{Uint328E} \end{array}$	server URL token socret + "server URL length") 16 (8 + "server URL length")
	Token parsing code example static parsefncodedToken(encoded: string, appld: number null): LRT.JappToken (// Backward compatibility	

Figure 86: LORIOT Access Token

Network Server		Prod	uct-wide settin
THE THINGS	The Things Stack V3 TTN V3 / Things Industries		
🔿 🧭 helium	Helium Use your own console		
୍ତ 🕪 🥥	Managed Helium Powered by IoT Creators		
	LORIOT		
ChiepStack	ChirpStack		
Showing 1 to 5	of 16 results	Previous	Next
Uplink URL			
https://api.datao	ake.co/integrations/lorawan/loriot/?wo	rkspaceId=19101d4f-f2{	🗎 Сору
Configure your LoRaV	AN Network Server to send data to this URL		
LORIOT Access To	ken	Prodi	uct-wide settin
Please save to c	теск		
			Update

Figure 87: Access Token Field In Datacake

You can generate a new access token or use the existing one.

9. If everything is done correctly, you should see a trickle of RAW data in the Debug window of Datacake.

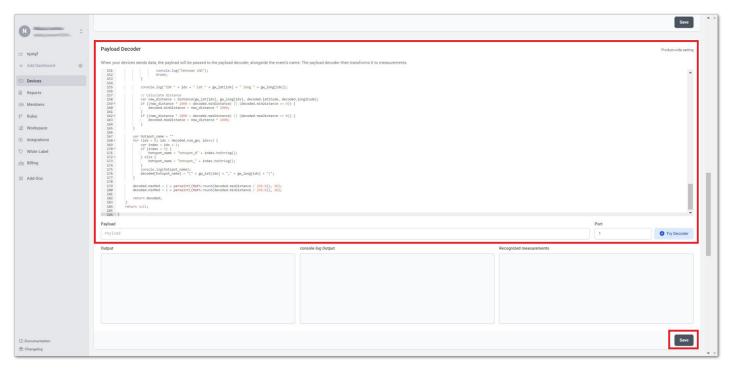
Setting up the Uplink Payload Decoder and the Downlink Encoder

 In the Datacake console, navigate to the Configuration tab of your RAK10701 device. At the bottom of the page, you will find the Payload Decoder field. Copy the decoder provided below and paste it in that field. This decoder will calculate the values displayed on the screen of the Field Tester.

```
function distance(lat1, lon1, lat2, lon2) {
    if ((lat1 == lat2) && (lon1 == lon2)) {
       return 0;
        var radlat1 = Math.PI * lat1 / 180;
       var radlat2 = Math.PI * lat2 / 180;
       var theta = lon1 - lon2;
       var radtheta = Math.PI * theta / 180;
        var dist = Math.sin(radlat1) * Math.sin(radlat2) + Math.cos(radlat1) * Math.cos(radlat2)
        if (dist > 1) {
            dist = 1;
        dist = Math.acos(dist);
        dist = dist * 180 / Math.PI;
        dist = dist * 60 * 1.1515;
       dist = dist * 1.609344;
        return dist;
function Decoder(bytes, fPort) {
   var decoded = {};
   // avoid sending Downlink ACK to integration (Cargo)
   if (fPort === 1) {
        var lonSign = (bytes[0] >> 7) & 0x01 ? -1 : 1;
        var latSign = (bytes[0] >> 6) & 0x01 ? -1 : 1;
        var encLat = ((bytes[0] & 0x3f) << 17) +</pre>
            (bytes[1] << 9) +
            (bytes[2] << 1) +
            (bytes[3] >> 7);
        var encLon = ((bytes[3] & 0x7f) << 16) +</pre>
            (bytes[4] << 8) +
            bytes[5];
        var hdop = bytes[8] / 10;
        var sats = bytes[9];
        var maxHdop = 2;
        var minSats = 5;
        if ((hdop < maxHdop) && (sats >= minSats)) {
            decoded.latitude = latSign * (encLat * 108 + 53) / 10000000;
            decoded.longitude = lonSign * (encLon * 215 + 107) / 10000000;
            decoded.altitude = ((bytes[6] << 8) + bytes[7]) - 1000;</pre>
            decoded.accuracy = (hdop * 5 + 5) / 10
            decoded.hdop = hdop;
            decoded.sats = sats;
            decoded.location = "(" + decoded.latitude + "," + decoded.longitude + ")";
            decoded.error = "Need more GPS precision (hdop must be <" + maxHdop +</pre>
                " & sats must be >= " + minSats + ") current hdop: " + hdop + " & sats:" + sats;
            decoded.latitude = latSign * (encLat * 108 + 53) / 10000000;
            decoded.longitude = lonSign * (encLon * 215 + 107) / 10000000;
            decoded.altitude = ((bytes[6] << 8) + bytes[7]) - 1000;</pre>
            decoded.accuracy = (hdop * 5 + 5) / 10
            decoded.hdop = hdop;
            decoded.sats = sats;
            decoded.location = "(" + decoded.latitude + "," + decoded.longitude + ")";
```

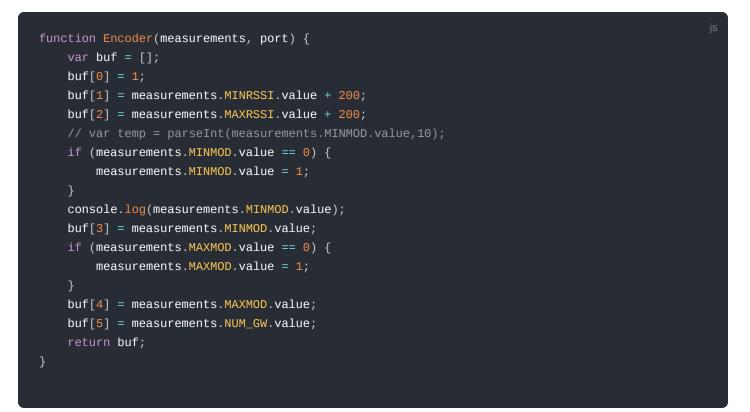
```
decoded.raw = rawPayload.uplink_message.rx_metadata[0].location;
decoded.num_gw = normalizedPayload.gateways.length;
decoded.minRSSI = 0;
decoded.maxRSSI = 0;
decoded.minSNR = 0;
decoded.maxSNR = 0;
decoded.minDistance = 0;
decoded.maxDistance = 0;
var server_type = 0;
if (typeof (rawPayload.uplink_message) != "undefined") {
    console.log("Found TTN format");
    server_type = 1;
   decoded.is_chirpstack = 1;
else if (typeof (rawPayload.hotspots) != "undefined") {
    console.log("Found Helium format");
    server_type = 2;
else if (typeof (rawPayload.rxInfo) != "undefined") {
    console.log("Found Chirpstack format");
    server_type = 3;
    decoded.is_chirpstack = 1;
else if (typeof (rawPayload.cmd) != "undefined") {
    console.log("Found LORIOT format");
    server_type = 4;
   decoded.is_chirpstack = 1;
    console.log("Unknown raw format");
var gw_lat = {};
var gw_long = {};
decoded.num_gw = 0;
for (idx_tst = 0; idx_tst < 10; idx_tst++)</pre>
    if (typeof (normalizedPayload.gateways[idx_tst]) != "undefined")
        console.log("Found gateway with IDX " + idx_tst);
        decoded.num_gw += 1;
for (idx = 0; idx < decoded.num_gw; idx++) {</pre>
    var new_rssi = (!!normalizedPayload.gateways && !!normalizedPayload.gateways[idx] &&
    var new_snr = (!!normalizedPayload.gateways && !!normalizedPayload.gateways[idx] && r
    if ((new_rssi < decoded.minRSSI) || (decoded.minRSSI == 0)) {</pre>
        decoded.minRSSI = new_rssi;
    if ((new_rssi > decoded.maxRSSI) || (decoded.maxRSSI == 0)) {
        decoded.maxRSSI = new_rssi;
    }
    if ((new_snr < decoded.minSNR) || (decoded.minSNR == 0)) {</pre>
        decoded.minSNR = new_snr;
    }
    if ((new_snr > decoded.maxSNR) || (decoded.maxSNR == 0)) {
```

```
decoded.maxSNR = new_snr;
    switch (server_type) {
        //TTN
            gw_lat[idx] = rawPayload.uplink_message.rx_metadata[idx].location.latitude;
            gw_long[idx] = rawPayload.uplink_message.rx_metadata[idx].location.longitude;
            break:
        case 2:
            gw_lat[idx] = rawPayload.hotspots[idx].lat;
            gw_long[idx] = rawPayload.hotspots[idx].long;
            gw_lat[idx] = rawPayload.rxInfo[idx].location.latitude;
            gw_long[idx] = rawPayload.rxInfo[idx].location.longitude;
            break;
        //LORIOT
            gw_lat[idx] = rawPayload.gws[0].lat;
            gw_long[idx] = rawPayload.gws[0].lon;
            break;
        default:
            console.log("Unknown LNS");
            break;
    console.log("IDX " + idx + " lat " + gw_lat[idx] + " long " + gw_long[idx]);
    var new_distance = distance(gw_lat[idx], gw_long[idx], decoded.latitude, decoded.long
    if ((new_distance * 1000 < decoded.minDistance) || (decoded.minDistance == 0)) {
        decoded.minDistance = new_distance * 1000;
    }
    if ((new_distance * 1000 > decoded.maxDistance) || (decoded.maxDistance == 0)) {
        decoded.maxDistance = new_distance * 1000;
var hotspot_name = ""
for (idx = 0; idx < decoded.num_gw; idx++) {</pre>
    var index = idx + 1;
    if (index < 9) {
        hotspot_name = "hotspot_0" + index.toString();
        hotspot_name = "hotspot_" + index.toString();
    console.log(hotspot_name);
    decoded[hotspot_name] = "(" + gw_lat[idx] + "," + gw_long[idx] + ")";
decoded.maxMod = 1 + parseInt((Math.round(decoded.maxDistance / 250.0)), 10);
decoded.minMod = 1 + parseInt((Math.round(decoded.minDistance / 250.0)), 10);
return decoded;
```





 After saving the payload decoder it is time to set the downlink payload encoder. Navigate to **Downlinks** and copy-paste the provided encoder in the field. Set the port to 2 and use the **IS_CHIRPSTACK** field to trigger the downlink.



Name	
LORIOT_DOWNLINK	
Description	
Downlink UUID	
a6c2e86f-75c7-4e65-9782-9efd8a	🗎 Сору
Fields used	
f your encoder function takes input from the device's fields, you can specify them here. They will	be used to create the form for the
downlink generator.	
IS_CHIRPSTACK ×	
IS_CHIRPSTACK ×	
Trigger on measurements	
Trigger on measurements	will be sent automatically.
Trigger on measurements If activated, each time the device records a measurement in one of the fields used, the downlink	will be sent automatically.
Trigger on measurements f activated, each time the device records a measurement in one of the fields used, the downlink	will be sent automatically.
Trigger on measurements f activated, each time the device records a measurement in one of the fields used, the downlink	will be sent automatically.
Trigger on measurements If activated, each time the device records a measurement in one of the fields used, the downlink Port 2	will be sent automatically.
Trigger on measurements f activated, each time the device records a measurement in one of the fields used, the downlink Port 2 Payload Encoder	will be sent automatically.
Trigger on measurements If activated, each time the device records a measurement in one of the fields used, the downlink Port 2 Payload Encoder 1 + function Encoder(measurements, port) { 2 y varia buf = []; }	will be sent automatically.
Trigger on measurements factivated, each time the device records a measurement in one of the fields used, the downlink Port 2 Payload Encoder 1 × function Encoder(measurements, port) { 2 var buf = []; 3 buf [0] = 1; }	will be sent automatically.
Trigger on measurements if activated, each time the device records a measurement in one of the fields used, the downlink: Port 2 Payload Encoder 2 var buf = []; 3 buf[0] = 1; 4 buf[1] = measurements, port) [(2 var buf = 2); 5 buf[2] = measurements.MINRSSI.value + 200;	will be sent automatically.
Trigger on measurements factivated, each time the device records a measurement in one of the fields used, the downlink of the fields used. The fields used the downlink of the fields used the downlink of the fields used. The fields used the downlink of the fields used the fields used the downlink of the fields used the fields use	will be sent automatically.
Trigger on measurements if activated, each time the device records a measurement in one of the fields used, the downlink: Port 2 Payload Encoder 2 var buf = []; 3 buf[0] = 1; 4 buf[1] = measurements, port) [(2 var buf = 2); 5 buf[2] = measurements.MINRSSI.value + 200;	will be sent automatically.
Trigger on measurements Factivated, each time the device records a measurement in one of the fields used, the downlink: Port 2 Payload Encoder 1 - function Encoder(measurements, port) { 2	will be sent automatically.
Trigger on measurements if activated, each time the device records a measurement in one of the fields used, the downlink of the fields used the fields used the downlink of the fields used the downlink of the fields used the downlink of the fields used the fields use	will be sent automatically.
Trigger on measurements Factivated, each time the device records a measurement in one of the fields used, the downlink: Port 2 Payload Encoder 1 - function Encoder(measurements, port) { 2	will be sent automatically.
<pre>Trigger on measurements if activated, each time the device records a measurement in one of the fields used, the downlink Port 2 Payload Encoder 1 = function Encoder(measurements, port) { ver buf = []; buf[] = nsignments.MINMESI, value + 200; buf[] = measurements.MINMESI, value + 200; buf[] = measurements.MINMEO, value = 0) { measurements.MINMEO, value; li = buf[] = measurements.MINMEO, value; li = console.log(measurements.MINMEO, value; li = console.log(measurements.MINMEO, value; li = buf[] = measurements.MINMEO, value; li = console.log(measurements.MINMEO, value;</pre>	will be sent automatically.
<pre>Trigger on measurements ff activated, each time the device records a measurement in one of the fields used, the downlink of Port 2 Payload Encoder 1 - function Encoder(measurements, port) { 2</pre>	will be sent automatically.
<pre>Trigger on measurements factivated, each time the device records a measurement in one of the fields used, the downlink of Port 2 Payload Encoder 1 - function Encoder(measurements, port) {</pre>	will be sent automatically.
<pre>Trigger on measurements factivated, each time the device records a measurement in one of the fields used, the downlink of Port 2 Payload Encoder 1 - function Encoder(measurements, port) {</pre>	will be sent automatically.

Figure 89: Downlink Configuration

3. You should now be able to see the data from the downlink on your Field Tester's screen.

Configuration of RAK10701-P Using WisToolBox

The Field Mapper should have the correct credentials to connect to the Helium Console. This can be done using WisToolBox and also with the help of the touchscreen LCD user interface.

- 1. Connect the RAK10701-P to the PC via USB Type-C cable and open the WisToolBox application. You can find more info on how to install and use the WisToolbox from its documentation 2 .
- 2. Click the **CONNECT DEVICE** button to launch WisToolBox Dashboard.

Wis ToolBox = Dashboard = Templates =		
Ermware	Connect a device to explore it's configuration and change it.	
🤊 🛆 🏼 🏟 RAK ID		

Figure 90: WisToolBox Desktop splash screen

3. Review the **Connection settings** parameters on the dashboard, then click on the **CONNECT** button.

			×
Dashboard	Connecti	on settings	
© Templates	Port	COM5	
Firmware	Baud Rate	115200 ~	
	Byte Size	8 ~	
	Parity	None 🗸	
	Stop Bits	1 2	
	Ċ	CONNECT	
	с	ANCEL	
?			

Figure 91: WisToolBox Desktop connection settings

4. On the WisToolBox Dashboard screen, select the RAK4630 module. This is the module inside the RAK10701-P Field Tester device.

Wis ToolBox =		0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00000000000000000000000000000000000000
Dashboard		WisBlock LPWAN Module for LoRaWAN	((•)) LORAWAN 5%
Templates		(MODEL RAK4630) (EUI TUUUUTECCCCCTAC)	PORT COM16
Firmware			
DEVICES			
☐ RAK4630 COM15 ▲			
	Ð		
? ⊘ 🍈 RAK ID			

Figure 92: Detected module by WisToolBox

5. You can now update the APPEUI, DEVEUI, and APPKEY. It must be the same as the parameters you have in the Setting up of Network Server. Then you can click **APPLY COMMANDS**.

Wis ToolBox =	BAK	Device Paramet	ters	Sync less then ?	_ 🗌 X
Dashboard	T Nusie Eary RAK4930 TO Constructions	SAVE AS TEMPL	ATE	APPLY A TEMPLAT	E
Templates					
Firmware	DEVICE INFO	 Global settings 			~
DEVICES	PARAMETERS	i LoRaWAN keys, ID, I	EUI		^
G RAK4630 COM16 ▲	FIRMWARE	Application EUI	00000000	0000001	16/16
		Application key	or 1001	0F70D001F00FD1F02010DD	32/32
		Device EUI	700005750		16/16
		Network ID	000013		
? 🐼 🚳 RAK ID	3 Commands in queue 🔨			APPLY CON	IMANDS

Figure 93: Changing device parameters

VOTE:

These are the only parameters that you need to change via WisToolBox. Other configurations like frequency plan, the interval of uplinks, TX power, and data rate can be done on the touchscreen of RAK10701.

For the frequency plan change, the device has to restart to activate this newly configured frequency band. There will be a notification on the UI touchscreen LCD. If you use WisToolBox to configure the band, you have to restart the device manually and there will be no notification from the UI of the LCD.

6. You will see the summary of commands that was applied successfully. If the update is unsuccessful, just resend the needed changes. After the successful update, click the **CLOSE** button to return to Dashboard.

Wis ToolBox =		
Dashboard	Commands applied to RAK4630 WisBlock LPWAN	
Templates	Module for LoRaWAN	
Firmware	O Device EUI	Successful 06:50
DEVICES	Application key	Successful 06:50
	Application EUI	Successful 06:50
🤊 🔕 🏟 RAK ID	CLOSE	

Figure 94: Successful update of parameters

7. You can now remove the USB Type-C cable and proceed to the screen of RAK10701. You can click the settings icon and then update the frequency plan, the interval of uplinks, TX power, and data rate as needed. You can use the arrows for navigation and click OK to save changes.

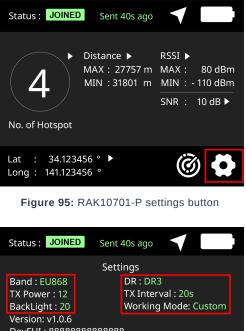




Figure 96: Configurable parameters

Miscellaneous

This part of the guide shows the Field Tester interface and how to update the firmware.

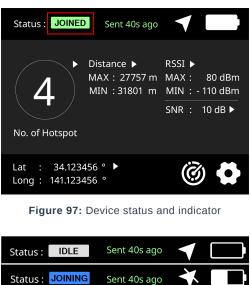
- User Interface
- Firmware Update

Field Tester Display Interface

This section discusses the interfaces on the LCD of the device as well as its pages.

Display Status and Indicator

The RAK10701-P WisNode Field Tester has status indicators that show the current state of the device.



Status.	Serie 105 ago		
Status : JOINING	Sent 40s ago	¥	
Status : JOINED	Sent 40s ago		
Status : FAILED	Sent 40s ago		
Status : SENDING	Sent 40s ago		

Figure 98: Different device status

Status:

- **IDLE**: RAK10701-P Field Tester Pro state is in between the previous uplink and the next uplink. The duration of IDLE depends on the interval configured on the device.
- **JOINING**: RAK10701-P Field Tester Pro is trying to join the network. This status is triggered when a Join request is sent.
- **JOINED**: RAK10701-P Field Tester Pro successfully received the Join accept the packet. This status will be displayed until refreshed when new data is sent.
- **FAILED**: RAK10701-P Field Tester Pro failed to join the network. Triggered by receive timeout. There might be no available gateway reachable by the Field Tester.
- **SENDING**: RAK10701-P Field Tester Pro's data such as GPS is being uploaded via an uplink. It will be displayed until the reception is completed or timed out.

Settings

The field tester has configurable parameters: Band, TX power, TX interval, backlight intensity, and DR. You can navigate the settings using the arrow widgets plus the back and ok buttons. The OTAA parameters APPEUI, DEVEUI, and APPKEY are also displayed but can't be changed on the touchscreen. WisToolBox or another Serial Port terminal tool is needed to send the AT commands to update the EUIs and key.

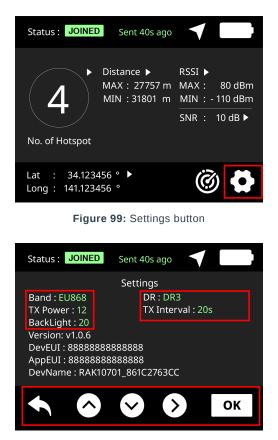


Figure 100: Settings page

Data Plots

There are four different data plots on the field tester: **number of gateways**, **RSSI**, **SNR**, and **approximate distance**. These graphs are accessible by touching the respective icons assigned to the parameter.

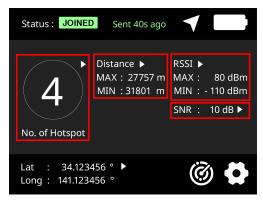


Figure 101: Accessing different data plots

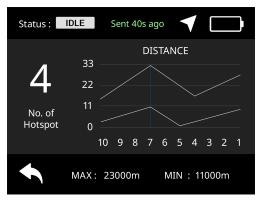


Figure 102: Distance plot

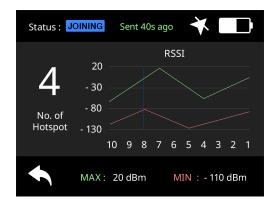


Figure 103: RSSI plot

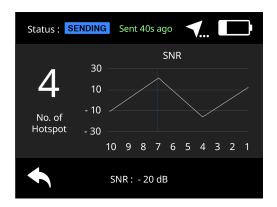


Figure 104: SNR plot

GPS Data

The main page shows the last GPS data captured by the device.

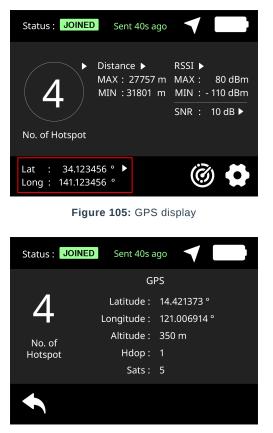


Figure 106: GPS data

Packet Frame Format

The Uplink packet format send on Fport 1:

Byte	Usage
0 - 5	GSP position see here for details. Decoding see below
6 - 7	Altitude in meters + 1000 m (1100 = 100 m)
8	HDOP * 10 (11 = 1.1)
9	Sats in view

When the GPS position is invalid of GPS is disable, the frame is fill with 0's.

The downlink response format send on Fport 2:

Byte	Usage
0	Sequence ID % 255
1	Min Rssi + 200 (160 = -40 dBm)
2	Max Rssi + 200 (160 = -40 dBm)
3	Min Distance step 250 m
4	Max Distance step 250 m
5	Seen hotspot

The distance is calculated from the GPS position and the gateways position returned by LoRaWAN server metadata. Under 250 m value is 250 m, over 32 km value is 32 km. 0 is considered as invalid response.

The following integration and payload transformation allows to decode the gps position and report is to mapper.

Dicovery uplink format send on Fport 3 (no ack):

Byte	Usage
0 - 5	GPS position 🗹

Discovery is sending 10 messages SF10 on every 40 seconds. All the other information comes from the metadata provided by the network server.

Upgrading the Firmware

It is recommended to update to the latest version of the firmware. To do this, download the latest RAK10701-P WisNode Field Tester firmware and use the WisToolBox to update the custom firmware.

1. Drag the downloaded firmware to the WisToolBox custom firmware section.

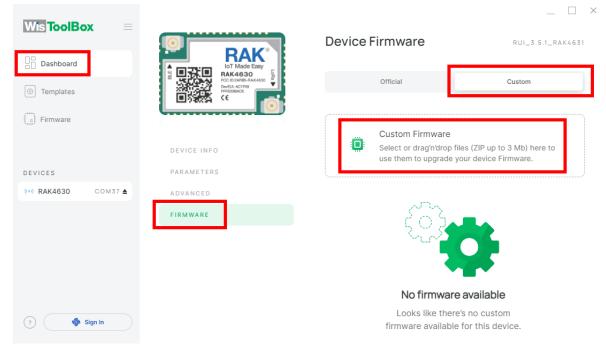


Figure 107: WisToolBox firmware

2. After the firmware file is uploaded to the application, you can now select **UPGRADE DEVICE**.

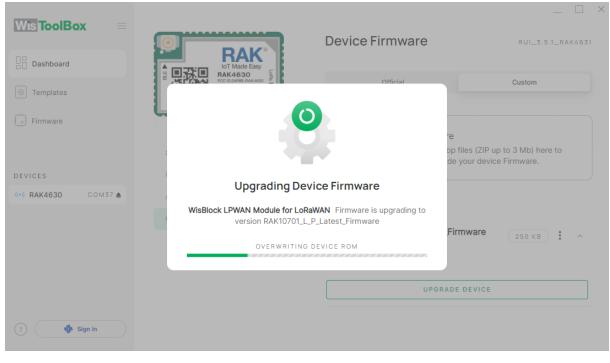
Wis ToolBox =	BAK	Device Firmware RUI_3.5.1_RAK4631	
Dashboard			
Templates		Official Custom	
Firmware			
	DEVICE INFO	Custom Firmware Select or drag'n'drop files (ZIP up to 3 Mb) here to use them to upgrade your device Firmware.	
DEVICES	PARAMETERS		
(••) RAK4630 COM37 ▲	ADVANCED	Select all	
	FIRMWARE	Select all	
		RAK10701_L_P_Latest_Firmware	
		Add description	
		UPGRADE DEVICE	

Figure 108: Upload the latest firmware

Wis ToolBox =	Device Firmware	AK4631
Dashboard		
Templates		Custom
Firmware	Upgrade Firmware	
	Are you sure you want to upgrade firmware for WisBlock LPWAN Module for LoRaWAN on COM37 to Official release RAK10701_LPLatest_Firmware ?	re op files (ZIP up to 3 Mb) here to de your device Firmware.
DEVICES		
(∞) RAK4630 COM37 ▲	When upgrading make sure (i) Device is connected to power supply	
	 You don't disconnect the device 	Firmware
	CANCEL	DE DEVICE
(?) 🔮 Sign In		

Figure 109: Confirm upgrading of firmware

3. If all proceed with no error, you should see Firmware update successful notification, and the RAK10701-P will restart automatically.





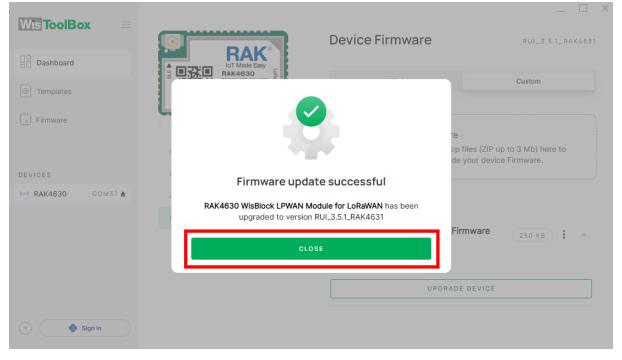


Figure 111: Successful upload of latest firmware

Last Updated: 2/20/2024, 6:23:14 AM