



Single Channel LoRa IoT Kit v2 User Manual

Document Version: 1.0.1

Version	Description	Date
1.0.0	Release	2019-Jan-28
1.0.1	Modify limitation words	2019-May-10
1.0.2	Add description of MQTT publish format	2019-Jun-15
1.0.3	Add ThingSpeak downlink example, and Video Link	2019-Jun-19



Inde	ex:						
1	Ove	rview		4			
	1.1	What	t is Dragino Single Channel LoRa IoT Kit v2?	4			
	1.2	What	t can you learn from the kit?	4			
	1.3	What	t parts Dragino LoRa IoT v2 includes?	5			
2	Preparing						
	2.1	Softw	vare for End Node	6			
		2.1.1	Install Arduino IDE and CH340 driver	6			
		2.1.2	Install LoRa Library for Arduino	7			
	2.2	Prepa	are for LG01-N Gateway	8			
		2.2.1	Configure LG01-N for internet connection.	8			
		2.2.2	Download putty tool to access LG01-N via SSH	11			
3	Exa	mple 1:	Test a LoRaWAN network	12			
	3.1	Туро	logy and Data Flow	13			
	3.2	Creat	te a gateway in TTN Server	14			
	3.3	Confi	igure LG01-N Gateway	16			
		3.3.1	Configure to connect to LoRaWAN server	16			
		3.3.2	Configure LG01-N's LoRa Radio frequency	17			
	3.4	Creat	te LoRa Shield End Node	18			
		3.4.1	Hardware Connection	18			
		3.4.2	Set up OTAA device in TTN and upload sketch to UNO	18			
		3.4.3	Configure to connect to Cayenne Application Server	22			
		3.4.4	Use downlink message to control relay	25			
		3.4.5	Test with Interrupt	27			
	3.5	Creat	te LoRa/GPS Shield End Node	28			
		3.5.1	Hardware connection	28			
		3.5.2	Set up ABP device in TTN and upload software to UNO	28			
	3.6	Conc	lusion and limitation	32			
		3.6.1	Overview for the example	32			
		3.6.2	Limitations	34			
4	Exa	mple 2:	Test with a MQTT IoT Server	36			
	4.1	Туро	logy and Data Flow	36			
	4.2	Set u	p sensor channels in ThingSpeak	37			
	4.3	Simu	late MQTT uplink via PC's MQTT tool	39			
	4.4	Try N	IQTT Publish with LG01-N Linux command	40			
	4.5	Confi	igure LG01-N Gateway	42			
		4.5.1	Publish Logic	42			
		4.5.2	Configure LG01-N's Radio frequency	43			
	4.6	Creat	te LoRa Shield End Node	45			
		4.6.1	Hardware Connection	45			
		4.6.2	Test with uplink	46			
		4.6.3	Test with interrupt by flame detect	47			



	4.6.4 Test with downlink	49
	4.7 Conclusion and limitation	51
	4.7.1 Overview for the example	51
5	Order Info	52
6	FAQ & Trouble Shooting	53
	6.1 I can't upload sketch to LoRa Shield in MAC OS, shows "	
	dev/cu.usbmodem1421 is not available "	53
	6.2 My IoT Kit has the model LG01-P instead of LG01-N, Can I still use this	
	manual	53
7	Technical Support	54
8	Reference	55



1 Overview

1.1 What is Dragino Single Channel LoRa IoT Kit v2?

Dragino Single Channel LoRa IoT Kit v2 is designed to facilitate beginners and developers to quickly learn LoRa and IoT technology. It helps users to turn the idea into a practical application and make the Internet of Things a reality. It is easy to program, create and connect your things everywhere. A number of telecom operators are currently rolling out networks, but because LoRa operates in the open spectrum you can also set up your own network.

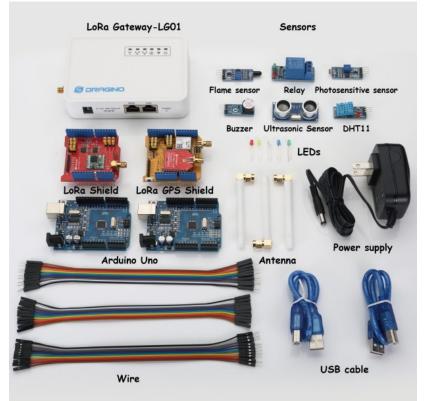
The LoRa IoT kit v2 shows how to build a LoRa network, and how to use the network to send data from a LoRa sensor node to the cloud server. Depends on the actually use environment, the LoRa gateway will connect your other LoRa nodes up to 500 ~ 5,000 meters.

1.2 What can you learn from the kit?

The goals through this LoRa IoT kit v2:

- ✓ Understand the structure of an Internet of Things network, and how does an IoT network works
- ✓ Learn coding method for Arduino micro controller
- ✓ Learn some common sensors.
- ✓ Learn some basic commands for Linux and
- ✓ Learn about LoRa and how to set up a LoRa network.
- ✓ Learn different way to connect LoRa network to IoT Server and compare their advantages / disadvantages.





1.3 What parts Dragino LoRa IoT v2 includes?

Single Channel LoRa IoT Kit Packing List.

- ✓ 1 x LG01-N single channel LoRa Gateway
- ✓ 1 x LoRa end node (LoRa Shield + Arduino UNO)
- ✓ 1 x LoRa end node (LoRa/GPS Shield + Arduino UNO)
- ✓ 1 x flame Sensor
- ✓ 1 x relay
- ✓ 1 x photosensitive sensor
- ✓ 1 x buzzer
- ✓ 1 x ultrasonic sensor
- ✓ 1 x DHT11 temperature and humidity sensor
- ✓ 20 x dupont cable (male to male)
- ✓ 20 x dupont cable (female to female)
- ✓ 20 x dupont cable (male to female)



2 Preparing

In the kit, there are two LoRa End Node, they are LoRa Shield + UNO and LoRa/GPS Shield + UNO. Both of them use Arduino UNO as MCU to control the LoRa transceiver.

We need to program the Arduino UNO during our testing to support the required functions for end nodes. To finish this, we need to install some software and library first.

2.1 Software for End Node

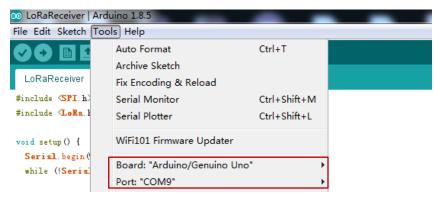
2.1.1 Install Arduino IDE and CH340 driver

First download and install Arduino IDE. This is the tool to program the Arduino UNO.

The Arduino UNO in the kit is clone version and is equipped with CH340 USB to UART chip. We need to install CH340 driver in the PC to let the Arduino IDE program it via USB. If we successful install the driver, a com port will show in the system device manager:



After install the driver, start Arduino and we will be able to use the board Arduino UNO and corresponding COM port to program UNO now.





We can enable compilation and upload in Arduino \rightarrow File \rightarrow Preference. This will help us in debug.

👓 LoRaReceiv	er Arduino 1.8.5						
File Edit Ske	Preferences	Preferences					
00 🗈	Settings Network						
LoRaReceiv	Sketchbook location:						
#include <sp:< th=""><th>C:\Users\edwin\Documents\Ar</th><th>duino</th></sp:<>	C:\Users\edwin\Documents\Ar	duino					
#include (Lol	Editor language:	English (English)					
void setup()	Editor font size:	12					
Serial.beg	Interface scale:	Automatic 100 💭 🦝 (requires restart of Arduino)					
while (! <mark>Se</mark> :	Show verbose output during:	🔽 compilation 👿 upload					

2.1.2 Install LoRa Library for Arduino

In our examples, we will use two different LoRa libraries for End Node to build different type of LoRa network. They are:

- Arduino-LMIC : LoRaWAN library to configure the End node as a standard LoRaWAN end node.
- LoRa-raw: This is a simple library for LoRa transmit & receive, all data transfer without ID control, encryption. If user wants to develop a LoRa network with private LoRa protocol, he can modify base on this Library.

We also need to install some libraries to connect to different sensors:

- > <u>DHTlib</u>: This is the library to use DHT11 temperature & humidity sensor.
- > <u>TinyGPS</u>: Library for LoRa GPS Shield to get the GPS data.

Download all above libraries and put them in the Arduino \rightarrow Libraries directory

rare > arduino-1.8.5-windows > arduino-1.8.5 > libraries > v 4 提案 libraries							
	*	名称	修改日期 类	型大小			
ire		퉬 Adafruit_Circuit_Playground	2017/7/18 23:21 文	件夹			
		퉬 arduino-Imic	2018/12/28 20:16 文	件夹			
		퉬 arduino-LoRa-master	2018/11/12 20:58 文	件夹			
		퉬 Bridge	2016/8/25 17:20 文	件夹			
件	Ξ	퉬 Esplora	2015/5/20 17:10 文	件夹			
ne		퉬 Ethernet	2017/10/2 15:37 文	件夹			
		🌗 Firmata	2017/3/19 0:47 文	件夹			



2.2 Prepare for LG01-N Gateway

In LoRa IoT Kit v2, we use LG01-N as LoRa Gateway. Unlike LG01-P in v1 kit, the LG01-N has its own LoRa utility and not need to program it via Arduino. Since we need to connect to Internet IoT Server, we need to configure the LG01-N to have internet access.

2.2.1 Configure LG01-N for internet connection.

Below steps show how to set up LG01-N to use WiFi for internet access.

Step1:

Connect PC to LG01-N's LAN port via RJ45 cable and set up PC Ethernet port to DHCP. PC will then get IP from LG01-N. The ip range is 10.130.1.xx Use browser to access the LG01-N via IP 10.130.1.1. (Recommend use Chrome here)

Step2:

Open a browser in the laptop and type http://10.130.1.1/cgi-bin/luci/admin User will see the login interface of LG01-N. The account for Web Login is: User Name: root Password: dragino 🖉 dragino-168cb0 - LuCI 🗙 ← → C 🗋 10.130.1.1/cgi-bin/luci/admin dragino-168cb0 Authorization Required Please enter your username and password. root Username root Password dragino 🖸 Login 🛛 🙆 Reset DRAGINO TECHNOLOGY CO., LIMITED

Step3:

In network -> Wireless, select radio0 interface and scan.

dragino-1b828	8 Status - System - Network - Service - Logout		AUTO REFRESH ON				
radio0: Master "dragino-1b8288"							
Wireless Ov	Wireless Overview						
👳 radio0	Generic MAC80211 802.11bgn Channel: 11 (2.462 GHz) Bitrate: ? Mbit/s	Restart	Scan Add				
%0 (آ <u>أ</u>	SSID: dragino-1b8288 Mode: Master BSSID: A8:40:41:1B:82:88 Encryption: None	Disable	Edit Remove				

Dragino Single Channel LoRa IoT Kit v2 User Manual --- 2019-Jun-19



Step4:

Select the wireless AP and join the wifi network:

dragino-1b	8288 Sta	atus - System - N	etwork 👻 S	ervice 👻 Logout		AUTO REFRESH ON
Join Network: Wireless Scan						
Signal	SSID	Channel	Mode	BSSID	Encryption	
100%	dragino-office	e 8	Master	50:64:2B:1A:B8:4D	mixed WPA/WPA2 - PSK	Join Network
a 84%	ChinaNet-gL	nb 2	Master	A4:29:40:66:F4:E7	mixed WPA/WPA2 - PSK	Join Network
dragino-1b8	3288 Stat					
Joining Network: "dragino-office" Replace wireless configuration Check this option to delete the existing networks from this radio. WPA passphrase						
		③ Specify the secret	encryption ke	ey here.		
warne of the	new network	wwan The allowed chara	cters are: A-	Z, a-z, 0-9 and _		
Create / Assign firewall-zone () Create / Assign firewall-zone () Create / Assign firewall-zone () Choose the firewall zone you want to assign to this interface. Select unspecified to remove the interface from the associated zone or fill out the create field to define a new zone and attach the interface to it.						
Back to scar	n results					Submit

Step5:

In network->wireless page, disable WiFi AP network. Notice: After doing that, you will lose connection if your computer connects to the LG01-N via its WiFi network.

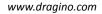
		vice - Logout		CHIJATL	D CHANGES. IS	AUTO REFRESH ON		
radio0: Master "dragino-1b8288"								
verview								
	-		Re	estart	Scan	Add		
			Dis	sable	Edit	Remove		
SSID: dragino-office Mode: Client Disable Edit BSSID: 50:64:2B:1A:B8:4D Encryption: - Disable Edit					Remove			
Associated Stations								
MAC-Address	Host	Signal / Noise	е	RX R	ate / TX Rate			
	Generic MAC80211 802 Channel: 11 (2.462 GHz) B SSID: dragino-1b8288 Mod BSSID: A8:40:41:1B:82:88 SSID: dragino-office Mode: BSSID: 50:64:2B:1A:B8:4D Stations	Channel: 11 (2.462 GHz) Bitrate: ? Mbit/s SSID: dragino-1b8288 Mode: Master BSSID: A8:40:41:1B:82:88 Encryption: None SSID: dragino-office Mode: Client BSSID: 50:64:2B:1A:B8:4D Encryption: - Stations	Channel: 11 (2.462 GHz) Bitrate: ? Mbit/s SSID: dragino-1b8288 Mode: Master BSSID: A8:40:41:1B:82:88 Encryption: None SSID: dragino-office Mode: Client BSSID: 50:64:2B:1A:B8:4D Encryption: - Stations	Generic MAC80211 802.11bgn Channel: 11 (2.462 GHz) Bitrate: ? Mbit/s Re SSID: dragino-1b8288 Mode: Master BSSID: A8:40:41:1B:82:88 Encryption: None Dis SSID: dragino-office Mode: Client BSSID: 50:64:2B:1A:B8:4D Encryption: - Dis Stations Stations	Generic MAC80211 802.11bgn Channel: 11 (2.462 GHz) Bitrate: ? Mbit/s Restart SSID: dragino-1b8288 Mode: Master BSSID: A8:40:41:1B:82:88 Encryption: None Disable SSID: dragino-office Mode: Client BSSID: 50:64:2B:1A:B8:4D Encryption: - Disable Stations Stations	Generic MAC80211 802.11bgn Channel: 11 (2 462 GHz) Bitrate: ? Mbit/s Restart Scan SSID: dragino-1b8288 Mode: Master BSSID: A8:40:41:1B:82:88 Encryption: None Disable Edit SSID: dragino-office Mode: Client BSSID: 50:64:2B:1A:B8:4D Encryption: - Disable Edit Stations		

(Note: make sure click the Save & Apply after configure)



After successful associate, the WiFi network interface can be seen in the same page and see LG01-N get the ip from the uplink router.

dragino-1b8288 Status -	System - Network - Service - Logou	t	AUTO REFRESH ON				
WAN WWAN LAN							
Interfaces							
LAN ⊕≇ (⊉ ⊛) br-lan	Protocol: Static address Uptime: 2h 0m 4s MAC: A8:40:41:1B:82:8B RX: 1.40 MB (13346 Pkts.) TX: 2.79 MB (10321 Pkts.) IPv4: 10.130.1.1/24	Restart Stop Edit	Delete				
WAN	Protocol: DHCP client MAC: A8:40:41:1B:82:8A RX: 4.30 MB (51840 Pkts.) TX: 55.77 KB (429 Pkts.)	Restart Stop Edit	Delete				
WWAN @ Client "dragino-office"	Protocol: DHCP client Uptime: 0h 6m 6s MAC: A8:40:41:1B:82:88 RX: 549.38 KB (5659 Pkts.) TX: 14.90 KB (94 Pkts.) IPv4: 10.130.2.169/24	Restart Stop Edit	Delete				
Add new interface							
		Save & Apply	Save Reset				





2.2.2 Download putty tool to access LG01-N via SSH

It will be helpful to see the LG01-N inside Linux system to understand the data flow and debug.

User can access to the Linux console via SSH protocol. Make sure your PC and the LG01-N is in the same network, then use a SSH tool (such as <u>putty</u>) to access it. Below are screenshots:

R PuTTY Configuration	x		
Session Logging Terminal Keyboard Bell Features Window Appearance Behaviour Input device's IP Port:22 Type: SSH Data Proxy Telnet Rlogin SSH Serial Close window on exit:	connect to Port 22 ● SSH ○ Serial	IP address: Port: User Name: Password:	IP address of LG01-N 22 root dragino (default)
About Open	Cancel		

After log in, you will be in the Linux console and can input commands here.

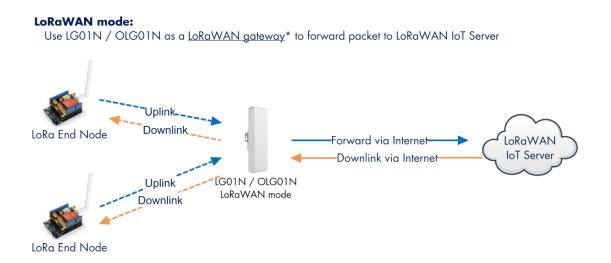


So we have prepare what we need and let's go for the examples!



3 Example 1: Test a LoRaWAN network

This example describes how to use LG01-N, LoRa Shield & LoRa GPS Shield to set up a LoRaWAN network and connect it to <u>TTN LoRaWAN Server</u>. It also shows how to use external application server to monitor / manage the LoRa Nodes.



Operate Principle:

- > LG01N/OLG01N running packet forward and will forward the uplink LoRa packet from end node to LoRaWAN server.
- > It will also forward downlink LoRa packet from LoRaWAN server to end node.
- > The end node can use OTAA or ABP mode in the LoRaWAN protocol.

Limitation:

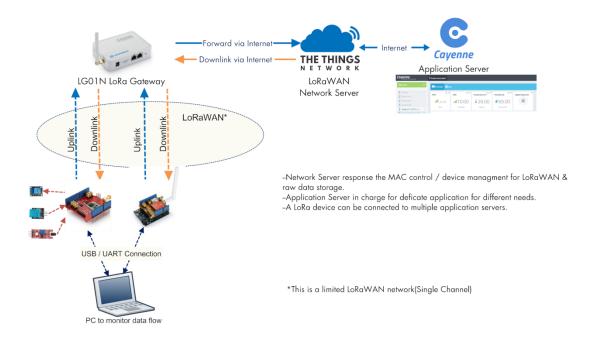
- > The LG01 only support one LoRaWAN frequency for uplink. So the end node should be set to fix frequency.
- > If end node use muliply frequencies to transfer, The LG01 will only be able to receive the same frequency set in LG01N.



3.1 Typology and Data Flow

The network topology and dataflow for the example is as below:

Topology for Thethingsnetwork Connection:



In next section we will start to configure for this example.



3.2 Create a gateway in TTN Server

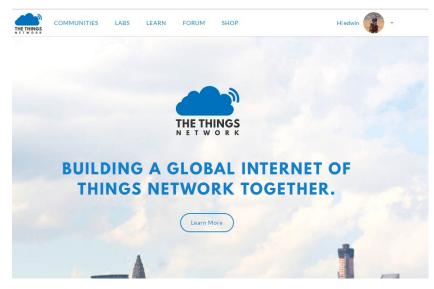
Step 1: Get a Unique gateway ID.

Every LG01-N has a unique gateway id. The id can be found at LoRaWAN page:

dragino-1b6fc4 Status							
LoRa Gateway Settings Configuration to communicate with LoRa devices and LoRaWAN server							
LoRaWAN Server Settin	ngs						
Service Provider	The Things Network	v					
Server Address	ttn-router-eu	v					
Server Port	1700						
Gateway ID	a840411b6fc44150						
Mail Address	dragino-1b6fc4@dragino.com						
Latitude	22.73						
Longtitude	114.23						
RadioMode	A for RX, B for TX	v					

The gateway id is: a840411b6fc44150

Step 2: Sign up a user account in TTN server





Step 3: Create a Gateway in TTN

HINGS	CONSOLE	Applications Gateways Support 👔 edwin
		Hi, edwin!
	This is where the magic happens. Here you can work with your da	Things Network Console. ta. Register applications, devices and gateways, manage your integrations, ators and settings.
	APPLICATIONS	GATEWAYS

又忤止) 编辑(L) 荁有(V) 历史(L) 书金(L) .		
The Things Network Console ×	and one bit while the	
← → ♂ ☆	https://console. thethingsnetwork.org /gateways/regist 器	III\ 🗭
JD 京东商城		
THE THINGS CONSOLE COMMUNITY EDITION		Applications Gatev
Ga	teways > Register	
	Gateway EUI The EUI of the gateway as read from the LoRa module	
Put the Gateway ID here	A8 40 41 1b 6f c4 41 50	Ø 8 bytes
Must use legacy packet forward	I'm using the legacy packet forwarder Select this if you are using the legacy <u>Semtech packet forwarder</u> .	
	Description A human-readable description of the gateway	
	LG02-Gateway-1	٥
	Frequency Plan The frequency plan this gateway will use	
Choose the right frequency	Europe 868MHz	\$
plan and router	Router The router this gateway will connect to. To reduce latency, pick a router that is in a region which is close to the location of the gateway.	
	ttn-router-eu	0

After create the gateway, we can see the gateway info, as below, the Status shows "not connected" because the LG01-N doesn't configure to send update status yet.

GATEWAY OVERVIEW		© settings
Gateway ID	eui-a840411b	
-	LG02-Gateway-1	
Owner	Contraction of the second seco	
Status	not connected	
Frequency Plan		
Router	ttn-router-eu	
Gateway Key	the set of the se	4 🖹



3.3 Configure LG01-N Gateway

3.3.1 Configure to connect to LoRaWAN server

We should configure the LG01-N now to let it connect to TTN network. Make sure your LG01-N has Internet Connection first.

Step1: Configure LG01-N to act as LoRaWAN forwarder mode

dragino-189	93c4 Stat	us - System -	Network -	Service -	Logout	
-		. oRa Gate h LoRa devices an	-	erver		
LoRaWAN S	Server Setti	ngs				
	IoT Service	LoRaWan/RAW	forwarder	•		
_	Debug Level	Little message o	utput	•		

Step2: Input server info and gateway id

Choose the correct the server address and gateway ID.

dragino	-1b8288 Sta				
	Gateway S	-	and LoRaWAN s	erver	
LoRaWA	N Server Sett	ings			
	Service Provider	The Things N	etwork	Ŧ	
	Server Address	ttn-router-eu		v	
	Server Port	1700			
	Gateway ID	a840411b			
	Mail Address	edwin@dragir	10.com		
	Latitude	22.73			
	Longtitude	114.23			

Check Result

After above settings, the LG01-N will be able to connect to TTN, as shown in below:





3.3.2 Configure LG01-N's LoRa Radio frequency

Now we should configure LG01-N's radio parameter to receive the LoRaWAN packets. We are using 868.1Mhz and other parameters as below:

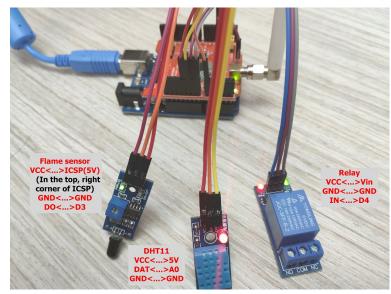
Radio Settings Radio settings for Channel	
Frequency (Unit:Hz)	868100000
Spreading Factor	SF7 •
Coding Rate	4/5 *
Signal Bandwidth	125 kHz 🔻
Preamble Length	8
LoRa Sync Word	 Length range: 6 ~ 65536 52 Value 52(0x34) for LoRaWAN
Encryption Key	Encryption Key

This parameters set is for uplink (receive data for LoRa End Node). According to LoRaWAN spec, the downlink radio parameters frequency is defined by network server (TTN). LG01-N will adjust downlink parameters according to info from TTN.



3.4 Create LoRa Shield End Node

3.4.1 Hardware Connection



There are three sensors connect to the LoRa Shield + UNO. These sensors are flame sensors, DHT11 (Temperature & Humidity sensor) and Relay. Please use the connection as we show in the photo.

Note: There is a trick in above connection, the relay connects to VIN. In this case, The UNO can only be power via USB port. If user need to power via DC power adapter, please use another 5v pin to power the relay.

3.4.2 Set up OTAA device in TTN and upload sketch to UNO

Here we set up the LoRa Shield + UNO as an OTAA device in TTN. We will tell the difference of OTAA and ABP mode later.

CONSOLE COMMUNITY EDITION				Applications	Gateways	Sup
Applications > 🤤 dragino_test_application1						
APPLICATION EUIS					to manage	euis
↔ 二 70 B3 D5 7E F0 00 46 18						
DEVICES			G	register device	🌣 manage devi	ices
	5 registe	ered devices				

<u>Step 1</u>: Create an OTAA device in TTN server -- > Application page.



Device EUI The serial number of your radio module, similar to a MAC address	
× A8 40 41 12 34 56 78 90	🔗 8 bytes
Application EUI	
70 B3 D5 7E F0 00 46 18	٢
Activation Method	

For this device, set up to use Cayenne payload, so TTN can parse the sensor data properly.

	Overview	Devices	Payload Formats	Integrations	Data	Setting
AYLOAD FORMATS						
Payload Format The payload format sent by your devices						

<u>Step 2</u>: Modify the LMIC library

To use LoRaWAN with LG01-N, we need to modify the LMIC library to support single channel mode.

Find the <u>Arduino LMIC</u> install path in Arduino library. Before compiling the code, user needs to change the Frequency Band to use with LG01-N. The change is in the file

arduino\libraries\arduino-	lmic \src	\lmic\conf	fig.h. C	Changes are as	below:
----------------------------	-----------	------------	----------	----------------	--------

#define CFG eu868 1		
//#define CFG_us915 1 //#define CFG_au921 1 //#define CFG_as923 1	Choose the Frequency Band, same as in LoRaWAN server	
//#define CFG_in866 1		
#define LG02_LG01 1	uncomment this for LG01 / LG02	
<pre>// DR_SF12CR=8, DR_S =#if defined(CFG_u9915) && // CFG_u9915 CFG_a8923 #define LG02_UPFREQ 9023 #define LG02_DNWFREQ 9233 #define LG02_RXSF 3 #define LG02_RXSF 8 #elif defined(CFG_eu868) & // CFG_eu868</pre>	320000 LG02_RXSF: End Device Uplink (transmit) SF 300000 LG02_TXSF: End Device Downlink (receive) SF // DR_SF7 The TXSF is now set to default value: // DR_SF12CR The TXSF is now set to default value: ss defined (LG02_LG01) US915/AS923 : 923300000 , SF12BW500 EU868: 869525000, SF12BW125 EU868: 869525000, SF12BW125 11=1, DR_SF10=2, DR_SF9=3, DR_SF6=4, DR_SF7=5, DR_SF7B=1, DR_FS DR_S525000 // DR_SF7 // DR_SF7	K, DR_NONE



WiFi101 Firmware Updater Board: "Arduino/Genuino Uno'

Programmer: "AVRISP mkII" Burn Bootloader

Port: "COM3" Get Board Info

<u>Step 3</u>: Input keys in Arduino Sketch and upload to device.

The sketch for this example is <u>lora_shield_cayenne_and_ttn-otaaClient.ino</u>. Download and open it, we need to modify the keys to match the keys in TTN. Get Device EUI/Application EUI & APP Key from TTN and put them in the sketch, make sure the Device EUI and Application Key are lsb and the APP key is msb.

ns > 🥪 dragino_test_	application1	. > Devices > 🚦	📄 otaa-devio	e-1						
Application ID Device ID		est_application1			While library	put th Mak	Key, APF ese keys e sure th	e EUI, Applica Ney. in Arduino-L e Device EU y are in Isb		
Activation Method	OTAA			1						
Device EUI	<> 1	Isb (0x90, 0x7	8, 0x56, 0x34	4, 0x12, 0x41	, 0x40,	0xA8]	8			
Application EUI	⇔	ISD [0x18, 0x4	6, 0x00, 0xF0	0, 0x7E, 0xD5	, 0xВ3,	0x70]) E			
Арр Кеу	4	ø	, 0x95, 0x15,	0x93, 0xAD, III	0x55, 0	x1A, 0	x83, 0x2	F, 0x31, 0x25	6, 0xB6, 0x7	'A, ▶ Ê
Device Address	• ±	26 01 2D 5E	Ê							
letwork Session Key	↔ ≒	•				Ē				
App Session Key	↔ \$	۰۰۰۰۰				Ē				
// This EVI must be in							Inp	out Keys in A	rduino Ske	etch
// first. When copying // the bytes. For TIN										
// 0x70. static const u1_t PROG	MEM APPEUI	[8]={ 0x18, 0x46,	0x00, 0xF0, 0x7	7E, 0xD5, 0xB3	3, 0x70 };					
void os_getArtEui (u1_	t* buf) {	nemcpy_P(buf, APPE	UI, 8);}			_				
// This should also be static const u1_t PROG				12 0-41 0-40	0					
void os_getDevEui (u1_				12, 0841, 0840	, OXAO J.					
// This key should be										
/ number but a block / practice, a key tak				y). In						
/ The key shown h <u>ere</u>			45 15.							
static const u1_t PROG				0xAD, 0x55, 0	x1A, 0x83	3, 0x2F	, 0x31, 0:	x25, 0xB6, 0x7	A, 0xF5, 0x7	4, 0x1D };
void os_getDevKey (u1_	_t* buf) {	memcpy_P(buf, APP	KEY, 16);}							
Ipload the cod	le to Ul	NO:								
Auto Format		Cui+i								
Archive Sketch										
Fix Encoding & Rel Serial Monitor	load	Ctrl+Shift+M								
Serial Plotter		Ctrl+Shift+L								



<u>Step 4</u>: Analyze output result

From output of LoRa Node Serial Monitor, we can see it send Joining after start(TX), then get join ACK (RX), then upload the data to TTN (TX).

🗢 ttn-abp Arduino 1.8.5	COM12 (Arduino/Genuino Uno)
文件 編書 项目 工具 帮助	发送
The abp	Connect to TTN and Send data to mydevice cayenne(Use DHT11 Sensor):
1/ xalalalalalalalalalalalalalalalalalalal	RXMODE_RSSI
2 * Copyright (c) 2019	############ COUNT=1 ####################################
3 *	The temperature and humidity:
4 * Permission is here	[28. 00°C, 63. 00%]
5 * obtaining a copy o	6024: engineUpdate, opmode=0x8
6 * to do whatever the	Packet queued
7 * including, but not	9013: EV_JOINING
8 * NO WARRANTY OF ANY	10182: engineUpdate, opmode=0xc
9 *	196226: engineUpdate, opmode=0xc
10 * This example sends	196532: TXMODE, freq=868100000, len=23, SF=7, BW=125, CR=4/5, IH=0
11 * world!", using fre	503011: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5, IH=0, rxsyms=255
12 * the The Things Net	514692: JaccRX1, dataLen=33
	515265: EV_JOINED
Using proxy DIRECT	515290: engineUpdate, opmode=0x808
	516157: TXMODE, freq=868100000, len=20, SF=7, BW=125, CR=4/5, IH=0
Using proxy DIRECT	570725: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5, IH=0, rxsyms=255

From gateway logread, we can see the data send from end node (txpk), dats get from server(rxpk).

dragino-1b6fb0 Status - System - Network - Servic	ce - Logout
Logread FreqINFO Report RxTxJson ErrorMSG	Gateway Log shows TX / RX LoRa Packet
Reserve(HEX)/200575628b/650a47b13497b2d5344f-c4a2s3d2b3f784d6 (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-49-50.6661622", "time": 366 Reserve(HEX)/201NFO (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-49-51.3108372", "time": 366 Reserve(HEX)/201NFO (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-51.512.8681.1; "time Reserve(HEX)/2018/001/2013716bb/3036494009aebe7d/ (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-51:12.7687142", "time: 375 (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-51:16.7687142", "time: 375 Reserve(HEX)/202b875111253b96b0603017316bb/3036494009aebe7d/ (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-51:16.7687142", "time: 375 Reserve(HEX)/202b875111253b96b0603017316bb/3036494009aebe7d/ (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-51:16.7687142", "time: 375 Reserve(HEX)/202b875111253b96b0603017316bb/30364494009aebe7d/ (RXPK): [up] ["rxpk": ["time": 2018-10-19T15-51:17.5290662", "time: 377 Reserve(HEX)/40b32012680010001462175b760716ad6224564b4062", "time: 378 Reserve(HEX)/40b320126800200103092d2456716aab627", "time: 338 Reserve(HEX)/40b32012680020013092d2456716aab62746a494597" Reserve(HEX)/40b320126800200013092d2456716a4624546716a496454916191 Reserve(HEX)/40b3201268003001638022666a2806.676d5491616191 Reserve(HEX)/40b320126800300016380226666a2806.676d5491616191 </th <th>56685421, "chan".0, "rfch":1, "freq".868.100000, "stat".1, "modu"."LORA", "datr"."SF7BW125", 57330098, "chan".0, "rfch".1, "freq".868.100000, "stat".1, "modu"."LORA", "datr"."SF7BW125", 5 5 5 5 5 5 5 5 5 5 5 5 5</th>	56685421, "chan".0, "rfch":1, "freq".868.100000, "stat".1, "modu"."LORA", "datr"."SF7BW125", 57330098, "chan".0, "rfch".1, "freq".868.100000, "stat".1, "modu"."LORA", "datr"."SF7BW125", 5 5 5 5 5 5 5 5 5 5 5 5 5

In TTN-Gateway page, we can also see the traffic.

	Page shows e status	the				Applications Gateway
ateways > 🏷 eui	-a840411b6ft	04150	> Tr	affic ^{beta}		
						TTN Send a Join reply. LoRa
▲ 23:56:34	868.1	lora	4/5	SF 7 BW 125	61.7	End node must get this packet ize: 26 bytes
▲ 23:55:30	868.1	lora	4/5 <mark>m</mark>	mmeditely send a essage after join	a Uplink success	to finish Join. The frquency shows use 868.1Mhz frequency, must be the same
▲ 23:54:27	868.1	lora	4/5	SF 7 BW 125	61.7	as the "LG02_DNWFREQ" in ize: 26 bytes
▲ 23:53:24	868.1	lora	4/5	SF 7 BW 125	61.7	Lmic config.c file
▲ 23:52:20	868.1	lora	4/5	SF 7 BW 125	61.7	1 dev addr: 6 01 2F B3 payload size: 26 bytes
 23:51:17 	868.1	lora	4/5	SF 7 BW 125	61.7	0 dev andr: 26 01 2F B3 paymed size: 26 bytes
<i>4</i> 23:51:16	868.1		4/5	SF 7 BW 125	71.9	

Dragino Single Channel LoRa IoT Kit v2 User Manual ---2019-Jun-19



Note: The LG02_DNWFREQ value in Arduno_LMIC/src/Imic/config.h should match downlink frequency from TTN. TTN shows 868.1 here, So LG02_DNWFREQ should be 868100000

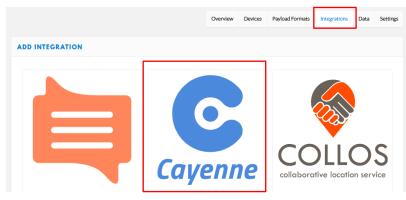
After success Joined, we can see the data in the device page:

APPLIC	ATION	DATA							II pa	ause 🛍	1
Filters	uplink	downlink	activation	ack	error						
	time	counter	port								
▲ 11	1:57:36	8	1		payload: 0	1 67 01 0E 02 68 7C 03 01 01	digital_out_3: 1	relative_humidity_2: 62	temperature_1: 27		
^ 11	1:56:33	7	1		payload: C	1 67 01 0E 02 68 7A 03 01 01	digital_out_3: 1	relative_humidity_2: 61	temperature_1: 27		
▲ 11	1:55:30	6	1		payload: C	1 67 01 18 02 68 78 03 01 01	digital_out_3: 1	relative_humidity_2: 60	temperature_1: 28		
▲ 11	1:54:28	5	1		payload: 0	1 67 01 18 02 68 76 03 01 01	digital_out_3: 1	relative_humidity_2: 59	temperature_1: 28	91%).

3.4.3 Configure to connect to Cayenne Application Server

In TTN, we can see the raw data, now we try to connect it to the application server.

Step 1: Add Cayenne in Application page

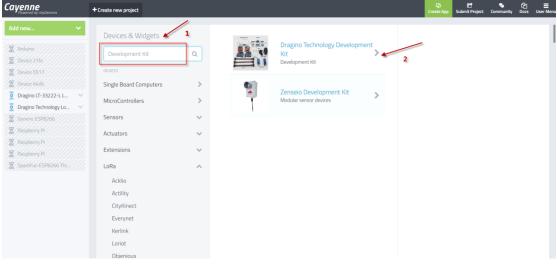


Process ID yourid	
Status • Running	
Platform Cayenne (v2.6.0) documentation	
Author myDevices	
Description Quickly design, prototype and commercialize IoT	solutions with myDevices Cayenne
SETTINGS	
Access Key The access key used for downlink	
default key devices messages	\$

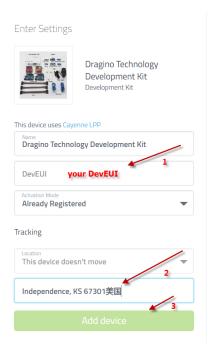


Step2: Log in Cayenne account and add devices.

evice/Widget	LoRa					
<i>i</i> ent				8		
gger	🛞 acklio	actility		J	ker link	ORIO T
bject			citykinect	everynet		
Dragino LT-33222-L L	Acklio	Actility	CityKinect	Everynet	Kerlink	Loriot
Dragino Technology Lo 💙						
Dragino Technology Lo 🗸	Objectivi				SEMTECH	
Generic ESP8266	Objenious	‱Orbi Wise	pixelnetworks	Sagemcom	SEMTECH	
Raspberry Pi			small smart sensitive.			
Raspberry Pi	Objenious	OrbiWise	Pixel Networks	Sagemcom	Semtech	Senet
Raspberry Pi						4
SparkFun ESP8266 Thi	SenRa	Sparker	. G	🔦 swisscom		X-TELIA
	M. Sellka	The Spark	iot ^x	SWISSCOTT		A-I ELIA
					The Things	
	SenRa	Spark	Stream	Swisscom	Network	X-Telia



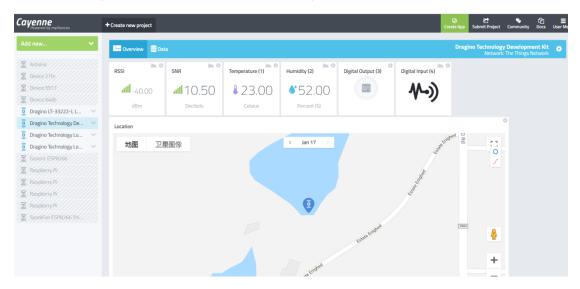
Add DevEUI of the End node



Dragino Single Channel LoRa IoT Kit v2 User Manual --- 2019-Jun-19



After above steps, we can now the sensor data in Cayenne:





3.4.4 Use downlink message to control relay

We can use either TTN or Cayenne to control the relay.

Control relay via TTN:

The string for ON is: 03 00 64 FF

The string for OFF is: 03 00 00 FF

In put above value in the TTN Downlink payload, we can see the relay can switch between

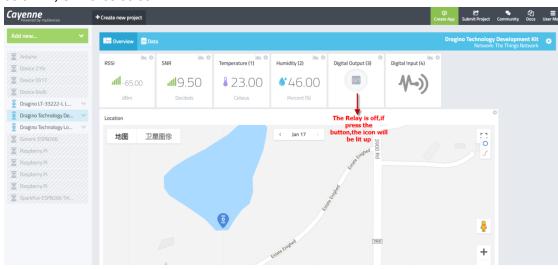
different states.	since we are in	Class A. the	downlink will c	only happen a	after each uplink.

	THE THINGS	COMM	ISOLE	TION						Applications	s Gatew	/ays	Support	\Lambda linsongx
,	Applications >		ora-shield	> Device	es > 🛿	😑 shiel	d-otta							
	DOWNLI	NK												
	Scheduli	ng							FPort					
	replace	firs	t la	st		1			1				S	Confirmed
	Payload				-*									
	bytes	field	ls C	03 00 64 F	F									Ø 4 bytes
														2
														Send
			DLE	N							Applicati	ons	Gateways	Support
	Appl	ications	> 😣 la	ora-shield >	Device	s > 🐖	shield-otta		Data					
					Derree		,							
												Ove	erview Dat	ta Settings
	A	PPLIC	ATION	DATA									П	pause 📋 <u>clear</u>
			uplink	downlink	activatio	on ack	error							
	1	Filters	time	counter	port						Relay	/ is on		
	downlink	1 1	:40:58	39	1		payload: 0	1 67 0	0 E6 02 68 58 03 01 01 04 00 01	digital_in_4: 1	digital_out_		elative_humidit	y_2: 44 1
succes	s and reply	4	.40.25	29	0		naudaadu fu	at an ai	dedl		1			•
			:40:35	38	0	confirmed	payload: [n app id: lor :							
			:41:43			ack confirmed	payload: 0							
			:40:34	37	1				0 E6 02 68 58 03 01 00 04 00 01	digital_in_4: 1	digital_out_		f relative_humidit	y_2: 44 1
		4								2				•
		▼ 11	:41:29		1	scheduled confirmed	payload: 0	3 00 64	4 FF					



Control relay via Cayenne

In Cayenne, just click the digital output button, it will auto send out the command strings: ON: 03 00 64 FF , OFF is: 03 00 00 FF



Cayenne will pass the string to TTN and TTN will show as above. In the serial monitor of End Node, we can see below output if downlink string arrives:

co lora_shield_cayenne_and_ttn-abpClient Arduino 1.8.5	COM12 (Arduino/Genuino Uno)
文件 编辑 项目 王具 帮助	发送
lora_shield_cayenne_and_thr-abpClient	[25.00°C, 66.00%]
49 static const u1_t PROGMEM APPSKE	3955087: engineUpdate, opmode=0x908
50	3957856: TXMODE, freq=868100000, len=23, SF=7, BW=125, CR=4/
51 // LoRaWAN end-device address (De	Packet queued
52 static const u4_t DEVADDR = 0x260	4012411: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5
53	4022496: Received downlink, window=RX1, port=1, ack=0
54 / / These callbacks are only used	4022541: EV_TXCOMPLETE (includes waiting for RX windows)
55// left empty here (we cannot le	Received :
56// DISABLE_JOIN is set in config	3 0 64 FF
57 void os_getArtEui (u1_t* buf) {	Set pin to HIGH.
58 void os_getDevEui (u1_t* buf) {	4028292: engineUpdate, opmode=0x810
59 void os_getDevKey (u1_t* buf) {]	4030903: TXMODE, freq=868100000, len=12, SF=7, BW=125, CR=4/
60	4085487: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/ ξ _
< [4124275: RXMODE_SINGLE, freq=869525000, SF=9, BW=125, CR=4/5
Léguis	4188702: EV_TXCOMPLETE (includes waiting for RX windows)
avrdude done. Thank you.	4188747: engineUpdate, opmode=0x900
	-
	▼ 自动滚屏 回车 ▼ 9600 波特率 ▼ Clear output



3.4.5 Test with Interrupt

The temperature & humidity in this example are updated periodically (once several minutes/hours), in some case, we need to update the data once an action is happen. So we need to use interrupt.

The DO pin of Flame sensor is high in normal case. While it detects a flame, this pin will become low and act as an external interrupt for Arduino. The Arduino UNO will then immediately upload the temperature and humidity to TTN

	COMMU	SOLE	N				Applications	Gateways	Support	A linsc
Applications	> 🥪 lo	ra-shield	Devices	> 📰 s	shield-otta	> Data				
								Ove	rview Data	Settin
APPLIC	ATION	DATA							II pa	use 🛍 cle
Filters	uplink	downlink	activation	ack	error	The flame s interrup				
	time	counter	port							<u>^</u>
▲ 11 <	:29:51	6	1	ţ	payload: 0	1 67 00 E6 02 68 5A 03 01 00 04 00 01	digital_in_4: 1 di	gital_out_3: 0 re	elative_humidity_	2:45 t
▲ 11	:29:28	5	1	ţ	payload: 0	1 67 00 E6 02 68 5A 03 01 00 04 00 00	digital_in_4: 0 di	gital_out_3: 0 re	elative_humidity_	2: 45 t
▲ 11	:29:21	4	1	ţ	payload: 0	1 67 00 E6 02 68 5A 03 01 00 04 00 01	digital_in_4: 1 di	gital_out_3: 0 re	elative_humidity_	2: 45 t
 11 	:28:58	3	1	ţ	payload: O	1 67 00 E6 02 68 5A 03 01 00 04 00 00	digital_in_4: 0 di	gital_out_3: 0 re	elative_humidity_	, 2:45 t

Then we can see on the cayenne:

Cayenne	+ Create new project	Create App Submit Project Community Docs User M
Add new 🗸	III Derview Elata	Dragino Technology Development Kit Network: The Things Network
Arduino Device 2.16 Device 5517 Device 64db Oragino UT-33222-L L	RSSI Max SNR Max Temperature (1) Max Humidity (2) Max Digital Output (3) Digital Input (4) atll 65:00 atll 9:00 \$ 23:00 \$ 46:00 \$ 100 \$ 100 dBm Decisies Cebias Percent (3) \$ 100 \$ 100 \$ 100 \$ 100	Have fire,the DO pin of flame sensor is low
Dragno Technology De Cagno Technology Le Cagno Technology Le Cagno Technology Le Cagnos ES78066 Rospharry B Rospharry B Rospharry B Sparking S5980565 TM.	Location 地理 卫重图作 《 Jan 17 》 	
	3 mentioned me	*



www.dragino.com

-	eld_cayenne_and_ttn-otaaClient Arduino 1.8.5	COM12 (Arduino/Genuino Uno)
文件 编辑 → ●	项目 工具 帮助	
	_cayenne_and_th-ctaaClient	3002238: engineUpdate, opmode=0x908
230 v	oid setup() {	Have fire, the tem###################################
231	<pre>Serial.begin(9600);</pre>	The temperature and humidity:
232	<pre>while(!Serial);</pre>	[23.00°C, 45.00%]
233	Serial.println("Connec	3003454: engineUpdate, opmode=0x108
234		Haveerature is send
235	<pre>pinMode(ctl_pin, OUTPUT)</pre>	############ COUNT=7 ####################################
236	pinMode(flame_pin, INPUT	The temperature and humidity:
237	attachInterrupt(1,fire,	[23.00°C, 45.00%]
238		3010302: engineUpdate, opmode=0x108
239	<pre>#ifdef VCC_ENABLE</pre>	Have fire, the temperature is send
240	// For Pinoccio Scout b	3109920: engineUpdate, opmode=0x108
241	pinMode(VCC_ENABLE, OUT	3110383: TXMODE, freq=868100000, len=26, SF=7, BW=125, CR=4/5, IH=0
< -0.sk.th.	III	3166736: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5, IH=0, rxsyms=255
	ide done. Thank you.	3205524: RXMODE_SINGLE, freq=869525000, SF=9, BW=125, CR=4/5, IH=0, rxsyms=255
		3269940: EV_TXCOMPLETE (includes waiting for RX windows)
		3269986: engineUpdate, opmode=0x900

3.5 Create LoRa/GPS Shield End Node

3.5.1 Hardware connection

The method to use LoRa/GPS Shield is similar with LoRa Shield. Below is the hardware connection of LoRa GPS Shield.



3.5.2 Set up ABP device in TTN and upload software to UNO

In LoRa Shield, we set up OTAA for connection. In this example, we will try ABP mode.

<u>Step 1</u>: Create an ABP device in TTN server -- > Application page. And change it to ABP mode.



www.dragino.com

				_
CONSOLE			Applications	Gateways Su
Applications > 🥪 dragino_test_applications	ation1			
APPLICATION EUIS				o manage euis
↔	3 (1)			
DEVICES			register device	manage devices
	5 registered devices			
Applications > 🥪 dragino_test_application1 >	Devices > 📰 edwintest1 > Settings			
		Overview	Data Settings	
DEVICE SETTINGS	SETTINGS			
General	Description A human-readable description of the device			
			0	
	Device EUI The serial number of your radio module, similar to a MAC address			
	∞ 00 BA DE A0 36 70 68 72		🖉 8 bytes	
	Application EUI			
	70 B3 D5 7E F0 00 46 18		\$	
	Activation Method			

<u>Step 2</u>: Input keys into Arduino Sketch.

The sketch for the LoRa /GPS Shield is LoRa GPS Sketch c	<u>ode</u>
Applications \rangle \otimes dragino_test_application1 \rangle Devices \rangle $=$ edwintest1	TTN LoRaWAN End Device page
Application ID dragino_test_application1	The conditional child Device page
Device ID edwintest1	
Activation Method ABP	Make sure the Network Session Key and App Session
Device EUI <> 또 00 BA DE A0 36 70 68 72 밝	Key are in MSB order
Application EUI <> ニ 70 B3 D5 7E F0 00 46 18 皆	
Device Address <> == 26 01 1C 22	
Network Session Key <> = 💉 📷 🖞 @x9A, @xEA, @xDØ, @x93, @x06	. 0xE3, 0x2B, 0x73, 0xDD, 0x54, 0x7B, 0x8B, 0xFF, " ■
App Session Key <>	. 0xCE, 0x40, 0xA2, 0xA3, 0xEE, 0x7B, 0xDF, 0xDC,

S DRAGINO			ww	w.dragino.com
ttn-abp #include (Imic.h> #include (hal/hal h>	Arduino Sketc	h ttn-abp		
#include < SPI .h> // LoRaWAN NwkSKey, network session key		Input the ke	ys from TTN]
<pre>// This is the default Semtech key, which is // network. static const PROGMEM u1_t INWESKEY[16] = { 0x4</pre>		0x2B, 0x73, 0xDD, 0x54,	Ox7B, Ox8B, OxFF, OxDC,	0x20, 0xF9 };
<pre>// LoRaWAN AppSKey, application session key // This is the default Semtech key, which is // network. static const u1_t PROGMEN APPSKEY[16] = { 0x1</pre>		0x40, 0xA2, 0xA3, 0xEE,	0x7B, 0xDF, 0xDC, 0x23,	0x0E, 0x2B };
// LoRaWAN end-device address (DevAddr) static const u4_t DEVADDR = 0x26011C22 ; //				

Choose Arduino UNO to upload the sketch to LoRa GPS Shield and UNO

	Auto Format	Curi	- 1	
	Archive Sketch			
	Fix Encoding & Reload			
	Serial Monitor	Ctrl-	+Shift+M	
	Serial Plotter	Ctrl+	+Shift+L	
	WiFi101 Firmware Updater			
			10 C	
ſ	Board: "Arduino/Genuino Uno"			>
ſ	Board: "Arduino/Genuino Uno" Port: "COM3"			> >
				> >
Ĺ	Port: "COM3"		ļ	> > >
Ĺ	Port: "COM3" Get Board Info			> >

All other steps are similar with how we use with LoRa Shield. Below are the outputs for reference:

Output from LoRa GPS Shield:

文件 編輯 项目 工具 帮助	© COM12 (Arduino/Genuino Uno)	_ 🗆 🗙
		发送
62// The key shown he	803657: engineUpdate, opmode=0x808	^
63 static const u1 t P	900002, TVMODE free=968100000 1 = 94 SE=7 DW=12E CD=4/E III=0	
64 void os getDevKey (867344: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5, IH=0, rxsyms=255	
65	906132: RXMODE_SINGLE, freq=869525000, SF=9, BW=125, CR=4/5, IH=0, rxsyms=255	
	970667: EV_TXCOMPLETE (includes waiting for RX windows)	
66 static osjob_t send	971317: engineUpdate, opmode=0x900	
67	########## NO. 1 ###########	
68// Schedule TX ever	The longtitude and latitude and altitude are:	
69 // cycle limitation	[114. 21, 22. 72, 120. 30]	
70 const unsigned TX_I	4786638: engineUpdate, opmode=0x908	
71	4791947: TXMODE, freq=868100000, len=24, SF=7, BW=125, CR=4/5, IH=0	
72// Pin mapping	Packet queued	
73 const lmic_pinmap 1	4850199: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5, IH=0, rxsyms=255	
	4888923: RXMODE_SINGLE, freq=869525000, SF=9, BW=125, CR=4/5, IH=0, rxsyms=255	
上传成功。	4953334: EV_TXCOMPLETE (includes waiting for RX windows)	
avrdude done. Thank you.	4953379: engineUpdate, opmode=0x900	
	□ 同生 • 4400 波线型 • 6	lear output

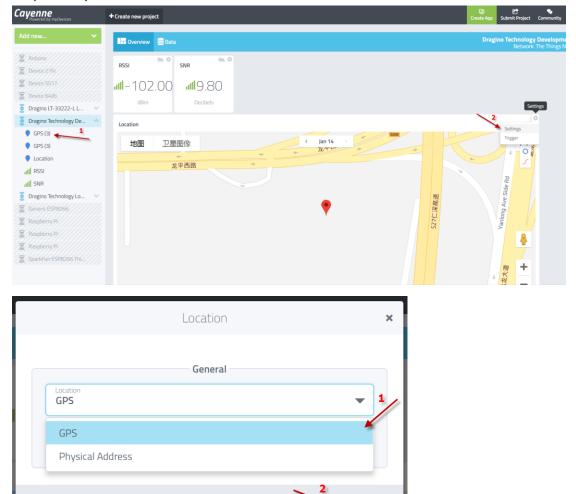


Upload GPS data to TTN:

NETWOR	K COMM	SOLE	N					Application	ns Gatew	ays Su	pport ဝ
Applications	> 🥪 k	ora-shield 🔿	Devices	gps	-shield	l-otta > Data					
APPLIC	ATION	DATA									II pause
Filters	uplink	downlink	activation	ack	error						
	time	counter	port								
1 7	:54:23	5	1	pay	load: 0	03 88 03 77 A5 11 6D4C 0	0 2E FE	gps_3.altitude: 120.3	gps_3.latitud	e: 22.7237	gps_3.longitud
17	:53:19	4	1	pay	load: 0	03 88 03 77 A5 11 6D4C (0 2E FE	gps_3.altitude: 120.3	gps_3.latitud	e: 22.7237	gps_3.longitud
1 7	:52:15	3	1	pay	load: 0	03 88 03 77 A5 11 6D 4C (0 2E FE	gps_3.altitude: 120.3	gps_3.latitud	e: 22.7237	gps_3.longitud
▲ 17	:51:12	2	1	pay	load: 0	03 88 03 77 A5 11 6D 4C (0 2E FE	gps_3.altitude: 120.3	gps_3.latitud	e: 22.7237	gps_3.longitud
7:50:08		1 1				3 77 A5 11 6D 4C 00 2E FI					

Output in Cayenne:

Remove





www.dragino.com

Cayenne Powered by myDevices	+ Create new project	Create App	C Submit Project	😒 Community	ද්ථ Docs	User
Add new 🗸	🔛 Overview 🛢 Data	Draj	gino Technology Network:	Developme The Things No	nt Kit stwork	•
Arduino Desuce 21fe Desuce 5517 Desuce 5517 Desuce 564b Desuce 664b Desuce 564b Desuce 564b Desuce 564b Desuce 564b Desuce 564 Desuce 564	Int Overview (Edita RSSI ha 0 sNR ha 0 sN	535日 339日 339日 339日 339日 339日 339日 339日	Network:	The Things N	etwork	
Image: Second System Image: Second System Image: Second System Ima		527	龙大道 十			

3.6 Conclusion and limitation

3.6.1 Overview for the example

This example shows how to set up a simple LoRaWAN network with public server. The LoRaWAN specification is for easy deploy the IoT network base on LoRa wireless. It contains the encryption, MAC control, device management etc. More info about LoRaWAN, please see <u>this link</u>.

There are some frequently ask points for the example:

1/ Difference between OTAA & ABP mode:

We have tested OTAA and ABP mode for LoRaWAN. They are two different modes. In OTAA mode, we can see the device will send a join request, the IoT server will send back a Join confirm with dynamic device address, network session key and app session key. Then the device will use these key to communicate with the LoRaWAN server. This make sure the device will only communicate with one server.

In ABP mode, it will use the FIX device address, network session key and app session key. It doesn't have join process. So in theory, any server with match keys is possible to decrypt the data from this end device.

We can see OTAA has better security than ABP mode.

2/ AES 128 encryption:

The data between end device and server are AES128 encryption. So the gateway can't parse the packets, it just forward them.

3/ LoRaWAN Network Server:



A LoRaWAN network server is necessary in a LoRaWAN network for device control/management/data management. If user wants to build the NS, there are some open sources LoRaWAN NS such as <u>LoRaServer</u> can be used. And some gateways already include LoRaWAN NS (this is also a plan for LG01-N).

4/ Downlink message

In this example, we use LoRaWAN Class A. The end node will open two short downlink windows after each uplink. More info about LoRaWAN class A, please refer<u>LoRaWAN specification</u>.



3.6.2 Limitations

The LG01-N is a single channel gateway (Same for LG02). And there are limitations: <u>1/ It works only on one frequency at a time</u>. It can support multiply end nodes, but all end nodes must transmit data at the same frequency so the LG01-N can receive it. For example: if the End node transmits at 868.1Mhz, The LG01-N's RX setting must be 868.1Mhz so to receive this packet.

<u>2/ It works only for one DR at a time.</u> DR specifies the Spreading Factor and Bandwidth. In LG01-N, even the rx frequency match , if DR doesn't match, it still can't get the sensor data.

3/ LoRaWAN compatible issue

In LoRaWAN protocol, the LoRaWAN end nodes send data in a hopping frequency. Since LG01-N only supports one single frequency, it will only be able to receive the packets sent from the same radio parameters (frequency & DR) in LG01-N.

For example, in EU868, a standard LoRaWAN device may send the data in eight frequencies with different Frequency & SF, such as:

LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND	D_CENTI); // g-band
LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND	_CENTI); // g-band
LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND	D_CENTI); // g-band
LMIC_setupChannel(3, 867100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND	D_CENTI); // g-band
LMIC_setupChannel(4, 867300000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND	D_CENTI); // g-band
LMIC_setupChannel(5, 867500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND	D_CENTI); // g-band
LMIC_setupChannel(6, 867700000, DR_RANGE_MAP(DR_SF12, DR_SF7), BANE	D_CENTI); // g-band
LMIC_setupChannel(7, 867900000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND	D_CENTI); // g-band
LMIC_setupChannel(8, 868800000, DR_RANGE_MAP(DR_FSK, DR_FSK), BAN	D_MILLI); // g2-band

So the LG01-N will only able to receive the 868100000, SF7 packet and will not receive others. Means only one packet will arrive the TTN server in every 8 packet sent from the LoRaWAN end node.

If user wants to receive all packets from LoRaWAN end node, user needs to set up the LoRaWAN node to send packets in a single frequency.

4/ Downlink & OTAA issue

According to the LoRaWAN class A spec, the end node will open two receive windows to get the message from LoRaWAN server for OTAA or downlink function. These two receive windows are quite short (milliseconds), if LoRa packet from the gateway can't reach End Node in the receive window time, the end node won't get the rx message and Downlink / OTAA won't work.



In our example, the Arduino LMIC library is modified to enlarge the RX window to let OTAA & downlink works.



4 Example 2: Test with a MQTT IoT Server

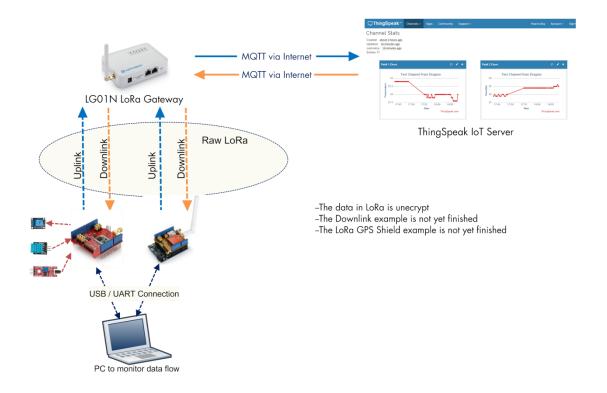
This example describes how to use LG01-N, LoRa Shield & LoRa GPS Shield to set up a LoRa network and connect it to <u>ThingSpeak IoT Server</u>.

A Video Instruction of this example can be found at this url: https://youtu.be/asoNyFYZamO

4.1 Typology and Data Flow

The network topology and dataflow for the example is as below:

Topology for ThingSpeak Connection:



In next section we will start to configure for this example.



4.2 Set up sensor channels in ThingSpeak

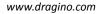
Step 1: Log in ThingSpeak and set up channels

🖵 ThingSpeak	™ Channels -	Apps - Communi	ty Support -		Commercial Use	How to Buy	Account -	Sign Out
My Channel	My Channels	nnels		٩	Help Collect dat from anot Click New channel. Click New channels Learn mor Examp Arduit Arduit Sespez Raspt Netdu Upgra	a in a ThingSpeak ter channel, or fro Channel to create e column headers hat column or clic ith that tag. eate channels, e e about ThingSpe DLES no no MKR1000 (66 herry Pi ino Plus CLE no McR1000 (66 herry Pi ino Plus	channel from a 4 m the web. a new ThingSper of the table to so K on a tag to sho xplore and trans aak Channels.	device, ak ort by the w form data.
up two channels								
up two channels Id 1: Temperatu Id 2: Humidity	re							
ıp two channels I d 1: Temperatu	re	Apps - Communi	ty Support -		Commercial Use	How to Buy	Account -	Sign Out
up two channels Id 1: Temperatu Id 2: Humidity	TY Channels ~	Apps - Communi	ty Support -	Help	Commercial Use	How to Buy	Account -	Sign Out
ip two channels d 1: Temperatu d 2: Humidity C ThingSpeak	re channels -	Apps - Communi T11 and Flame sensor	ty Support -		ata that a ThingSpeak a Id any type of data, plus	pplication collect	s. Each channel i cation data and c	ncludes
Ip two channels Id 1: Temperatu Id 2: Humidity C ThingSpeak New Char Name Description	re Channels - Innel Test from the DH MQTT example	T11 and Flame sensor	ty Support -	Help Channels store all the d eight fields that can hol status data. Once you co	ata that a ThingSpeak d any type of data, plus ollect data in a channel	pplication collect	s. Each channel i cation data and c	ncludes
up two channels Id 1: Temperatu Id 2: Humidity C ThingSpeak New Char	re Channels - nnel Test from the DH		ty Support -	Help Channels store all the d eight fields that can hol status data. Once you ca visualize it. Channel Settin - Channel Name: E	ata that a ThingSpeak a d any type of data, plus ollect data in a channel NGS inter a unique name for	pplication collect three fields for lo you can use Thin the ThingSpeak c	s. Each channel i cation data and gSpeak apps to a hannel.	ncludes
Ip two channels Id 1: Temperatu Id 2: Humidity C ThingSpeak New Char Name Description	re Channels - Innel Test from the DH MQTT example	T11 and Flame sensor	ty Support -	Help Channels store all the d eight field sthat can hol status data. Once you ca visualize it. Channel Settin - Channel Name: E - Description: Ento	ata that a ThingSpeak (d any type of data, plus ollect data in a channel ngS Enter a unique name for er a description of the T	pplication collect three fields for lo you can use Thin the ThingSpeak c hingSpeak channe	s. Each channel i cation data and c gSpeak apps to a hannel. el.	ncludes one for nalyze and
Ip two channels d 1: Temperatu d 2: Humidity ThingSpeak New Char Name Description Field 1	** Channels - ** Channels - ** Test from the DH MQTT example Temperature	T11 and Flame sensor	ty Support -	Help Channels store all the d eight field sthat can hol status data. Once you ca visualize it. Channel Settin - Channel Name: E - Description: Ento	ata that a ThingSpeak d any type of data, plus ollect data in a channel ngS arter a unique name for er a description of the T e box to enable the field	pplication collect three fields for lo you can use Thin the ThingSpeak c hingSpeak channe	s. Each channel i cation data and c gSpeak apps to a hannel. el.	ncludes one for nalyze and
ip two channels d 1: Temperatu d 2: Humidity ThingSpeak New Char Name Description Field 1 Field 2 Field 3	** Channels - ** Channels - ** Test from the DH MQTT example Temperature	T11 and Flame sensor	ty Support -	Help Channels store all the d eight fields that can holi status data. Once you co visualize it. Channel Settli - Channel Settli - Description: Ente - Field#: Check the channel can haw - Metadata: Enter if	ata that a ThingSpeak a d any type of data, plus ollect data in a channel ngS rater a unique name for rr a description of the T e box to enable the field up to 8 fields.	upplication collect three fields for lo- you can use Thin; the ThingSpeak channe , and enter a field nel data, includin;	s. Each channel i cation data and c gSpeak apps to a hannel. el. name. Each Thin g JSON, XML, or G	ncludes one for nalyze and gSpeak 2SV data.
up two channels Id 1: Temperatu Id 2: Humidity Q ThingSpeak New Char Name Description Field 1 Field 2	** Channels - ** Channels - ** Test from the DH MQTT example Temperature	T11 and Flame sensor	ty Support -	Help Channels store all the d eight fields that can hol status data. Once you ca visualize it. Channel Settii • Channel Settii • Description: Entik • Channel can haw • Metadata: Enter i • Tags: Enter keyw	ata that a ThingSpeak a d any type of data, plus ollect data in a channel ngS er a description of the T e box to enable the fields. information about char ords that identify the cl	pplication collect three fields for lo you can use Thin the ThingSpeak c hingSpeak channe , and enter a field nel data, includin nannel. Separate t	s. Each channel i cation data and c SSpeak apps to a hannel. 2l. name. Each Thin g JSON, XML, or c ags with comma:	ncludes one for nalyze and gSpeak CSV data. 5.
ip two channels d 1: Temperatu d 2: Humidity ThingSpeak New Char Name Description Field 1 Field 2 Field 3	** Channels - ** Channels - ** Test from the DH MQTT example Temperature	T11 and Flame sensor	ty Support -	Help Channels store all the d eight fields that can hol status data. Once you cu visualize it. Channel Settiil • Channel Settiil • Channel Name: E • Description: Entre • Fieldit: Check the channel can haw • Metadata: Enter • Tags: Enter keyw • Link to External S	ata that a ThingSpeak a d any type of data, plus ollect data in a channel ngS rater a unique name for rr a description of the T e box to enable the field up to 8 fields.	pplication collect three fields for lo you can use Thin the ThingSpeak c hingSpeak channe , and enter a field nel data, includin nannel. Separate t	s. Each channel i cation data and c SSpeak apps to a hannel. 2l. name. Each Thin g JSON, XML, or c ags with comma:	ncludes one for nalyze and gSpeak CSV data. 5.
up two channels Id 1: Temperatu Id 2: Humidity Description Field 1 Field 3 Field 3 Field 4 Field 5	** Channels - ** Channels - ** Test from the DH MQTT example Temperature	T11 and Flame sensor	ty Support -	Help Channels store all the d eight fields that can hol status data. Once you cu visualize it. Channel Settiil • Channel Settiil • Channel Name: E • Description: Entre • Fieldit: Check the channel can haw • Metadata: Enter • Tags: Enter keyw • Link to External S	ata that a ThingSpeak k d any type of data, plus ollect data in a channel ngS inter a unique name for er a description of the T e box to enable the field e up to 8 fields. information about char ords that identify the cl site : If you have a websi nnel, specify the URL.	pplication collect three fields for lo you can use Thin the ThingSpeak c hingSpeak channe , and enter a field nel data, includin nannel. Separate t	s. Each channel i cation data and c SSpeak apps to a hannel. 2l. name. Each Thin g JSON, XML, or c ags with comma:	ncludes one for nalyze and gSpeak CSV data. 5.
New Chat Name Description Field 1 Field 2 Field 3 Field 4	** Channels - ** Channels - ** Test from the DH MQTT example Temperature	T11 and Flame sensor	ty Support -	Help Channels store all the d iejpt fields that can hol status data. Once you co visualize it. Channel Settlin e. Channel Settlin e. Channel Name: Fi Bescription: Entit fichannel can haw Metadata: Enter li Tags: Enter keyw e. Link to External 3 Thinks peak char show Channel Lu o Latitude: 1	ata that a ThingSpeak k d any type of data, plus ollect data in a channel ngS inter a unique name for er a description of the T e box to enable the field e up to 8 fields. information about char ords that identify the cl site : If you have a websi nnel, specify the URL.	pplication collect three fields for lo you can use Thin the ThingSpeak c hingSpeak channe , and enter a field nel data, includin nannel. Separate t te that contains in ition in decimal de	s. Each channel cation data and o gSpeak apps to a hannel. N. name. Each Thin g JSON, XML, or o ags with commas	ncludes one for nalyze and gSpeak CSV data. 3. your

- Longitude: Specify the longitude position in decimal degrees. For example, the longitude of the city of London is -0.1275.
- Elevation: Specify the elevation position meters. For example, the elevation of the city of London is 35.052.
- Video URL: If you have a YouTube[™] or Vimeo[®] video that displays your channel

Field 8

Metadata





Step 2: Get MQTT keys for these channels.

Go to Account \rightarrow My profile and get the MQTT API Key

□ ThingSpeak [™]	Channels - Apps - Commun	nity Support -	Commercial Use How to Buy Account - Sign O
Edit MathWorks Account set	tings		Imme zone is used when displaying data in you your ThingSpeak apps. Wy Account Wer API key is required to create and manage My Profile API. API.
Edit MathWorks community	information		API Requests
			Get Channel List
ThingSpeak settings	5		GET https://api.thingspeak.com/channels.json?api_key=0QUBLUU0JQKX26NR
			Create a Channel
Time Zone	UTC Change Time Zone		POST https://api.thingspeak.com/channels.json api_key=@QUBLUU0JQKX26NR name=My New Channel
User API Key	OQU8LUUOJQKX26NR		Clear a Channel Feed
	Generate New API Key		DELETE https://api.thingspeak.com/channels/CHANNEL_ID/feeds.json api_key=0QUBLUU0JQKX26NR
MQTT API Key	BYR3I5ECL787PHG9	word of MQTT Server	Delete a Channel
	Generate New MOTT API Ke	ey 🛛	DELETE https://api.thingspeak.com/channels/CHANNEL_TD api_key=0QUBLUUD3QCK25NR
	2		Update Channel Metadata
			PUT https://api.thingspeak.com/channels.json api_key=00UBLUU03000260N name=Changed Channel Name

Go to channel page: get the sensor channel:

Channel ID: This is the remote Channel ID in ThingSpeak

Author: User Name for MQTT connection

Write API Key: API key for each channel

🖵 ThingSpe	ak™ Channels -	Apps -	Community	Support -		Commercial Use	How to Buy	Account -	Sign Out
Channel ID: 682338 Author: engineerlin Access: Private	Remote Channel User Name of MQT	- N	d Flam 1QTT example	ie ser	isor				
Private View Pu	blic View Channel S	ettings S	Sharing AP	l Keys D	ata Import / Export				
Write API Key	Key EVDKI16NV993M Generate New Write				keys are auto-gener API Keys Se • Write API Key been compro • Read API Key	r: Use this key to write da omised, click Generate N rs: Use this key to allow o	new channel. Ita to a channel. If ew Write API Key. ther people to vie	you feel your ke	y has hannel
Read API I _{Key}	Keys RUGYWWIVTLU44	IX8M			read key for t • Note: Use thi	s field to enter informati keep track of users with ts	on about channel	read keys. For e	



4.3 Simulate MQTT uplink via PC's MQTT tool

This step is not necessary, it just to help user to understand the MQTT protocol and simulate the MQTT connection to ThingSpeak. And check if the account info is valid and correct.

In the PC, download and install <u>MQTT.fx</u>. Open MQTT.fx and configure add a new MQTT client, as below:

Broker Address: mqtt.thingspeak.com Broker Port: 1883 Client ID: User Defined

Edit Connection Profiles	
M2M Eclipse ThingSpeak	Connection Profile
mydevices	Profile Name ThingSpeak
	Broker Address mqtt.thingspeak.com Broker Port 1883 Client ID Desktop_Client Generate
	General User Credentials SSL/TLS Proxy Last Will and Testament
	User Name dragino Password
Add a new MQTT Add a new MQTT Client Profile	Input User ID and MQTT API from Input User ID and MQTT API from ThingSpeak Account→My Profile
+ -	Revert OK Apply

After add the profile, connect it and publish to the corresponding Channel with correct API key. MQTT API see this document:

MQTT.fx - 1.3.1 File Extras Help	and the second s	
ThingSpeak	Connect Disconnect Replace Channel ID and API here	-
Publish Subscribe	rripts Broker Status Log « channels/200893/publish/8920R2 FY • Publish @050 QoS1 QoS2	Retained
	field1=36&field2=87&status=MQTTPUBLISH	
	Data to be sent, update both field1 & field2 of this channel	

If update successful, we can see the update in the channel:



	Channels -	Apps Community	Support -		How to		Sign Ou
Add Visualizations	Data Export				MATLAB Analysis	MATLAB Visua	alization
Channel Stats Created: 11 months ago Updated: Jess than a minute Last entry: Jess than a minute Entries: 1762							
Field 1 Chart		8 0	/ x	ield 2 Chart		₫ p / ×	
	est Channel Fron	n Dragino			hannel From Dragin	0	
75 50 25 25 0	est Channel Fron	n Dragino		100 Auguret	hannel From Dragin	0	

4.4 Try MQTT Publish with LG01-N Linux command

This step is also not necessary; it is to show the basic command used for MQTT connection and will help for further debug when connection fails.

First, we need to make sure the LG01-N has internet access. We can log in the SSH and ping an Internet address and see if there is reply. As below:



LG01-N has built-in Linux utility **mosquitto_pub**. We can use this command to publish the data to ThingSpeak.

The command to update a feed is as below:

mosquitto_pub -h mqtt.thingspeak.com -p 1883 -u dragino -P QZXTxxxxxO2J -i dragino_Client -t channels/200893/publish/B9Z0R25QNVEBKIFY -m "field1=34&field2=89&status=MQTTPUBLISH"

(Make sure the $^{\prime\prime\prime}$ is included, otherwise only one data will be uploaded)

Below is the output window:

🔚 172.31.255.254 - SecureCRT	
文件(F) 编辑(E) 查看(V) 选项(O) 传输(T) 脚本(S) 工具(L) 帮助(H)	
🖏 況 💭 🎣 🗶 I 🖻 🕵 👫 I 😼 😼 I 🖀 💥 I 🕐 I 🐼 📍 I 📀 I 🔤 💂	
172.31.255.254	
root@dragino-146d78:~# mosquitto_pub -h mqtt.thingspeak.com -p 1883 -u 2J -i dragino_Client -t channels/200893/publish/B9 field1=34&field2=89&status=MQTTPUBLISH" root@dragino-146d78:~#	i dragino -P Q

After running this command, we can see the data are updated to ThingSpeak, which has same result as what we did at mqtt.fx.

Dragino Single Channel LoRa IoT Kit v2 User Manual --- 2019-Jun-19



So we success to use LG01-N to uplink data to ThingSpeak, the **mosquitto_pub** command is executed in the Linux side, finally, we will have to call **mosquitto_pub** command while the LoRa sensor data arrive. We will explain how to do that in next step.

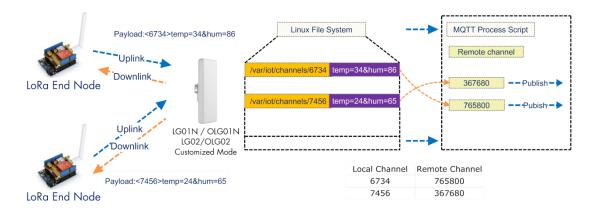


4.5 Configure LG01-N Gateway

4.5.1 Publish Logic

In LG01-N (firmware version > LG02_LG08--build-v5.1.1545908833-20181227-1908), there is a built-in script to process the MQTT data. The logic of this flow is as below:

How MQTT script works:



Operate Principle:

- > LoRa End Node sends the data to gateway in specify format: <node_ID>value
- > Gateway get the data and will put the data in corresponding files under /var/iot/channels.
- > MQTT Process Script will publish data to remote channel according to the pre-configure mapping

Step1: Configure LG01-N to act as MQTT mode

dragino-1b7060 sta	
LoRa Gateway S Configuration to communicate w	Settings ith LoRa devices and LoRaWAN server
LoRaWAN Server Sett	tings
IoT Service	LoRaRAW forward to MQTT sen 🔻
Debug Level	Little message output
Service Provider	The Things Network
Step2: Configure MQTT s	erver info
MQTT Server Settings Configuration to communicate with MQTT server	
Configure MQTT Server	
Select Server ThingSpeak M	NOLL *

User Name [-u]	dragino1
Password [-P]	32W6GMEXYTEQ7049
Client ID [-i]	dragino_Client

In step 2, we have below settings:

- ✓ UserName[-u option]: Input Author (user name for MQTT Connection)
- ✓ Password[-P option]: Input MQTT API key

Dragino Single Channel LoRa IoT Kit v2 User Manual ---2019-Jun-19



- ✓ Client_ID[-i]: dragino_Client (can put any string)
- Because we choose Thingspeak so we have below pre-set options but not show in web
 - Broker Address[-h]: mqtt.thingspeak.com
 - Broker Port[-p]: 1883
 - Topic Format[-t]: channels/CHANNEL/publish/WRITE_API.
 - Data String Format[-m]: DATA&status=MQTTPUBLISH

And we configure this channel:

- ✓ Local Channel ID: 10009
- ✓ Remote Channel ID: 396640
- ✓ Write_api_key: Write API key for this channel.

In the mqtt script, the upper **CHANNEL** will be replaced by the parameter (remote channel in IoT server). and the **WRITE_API** will be replaced by the settings in write api key. The **DATA** will be replaced by the value stored in the /var/iot/channels/LOCAL_CHANNEL file.

MQTT script will keep checking the files in /var/iot/channels/. If it finds a match Local channel, then the MQTT script will send out the data of this local channel to a remote channel according to the setting above.

User can also enable MQTT debug level and run logread in Linux console to see how the mqtt command is compose. Below is an example:

Tue Nov 27 15:08:03 2018 kern.notice syslog Tue Nov 27 15:08:03 2018 kern.notice syslog Tue Nov 27 15:08:09 2018 kern.notice syslog Tue Nov 27 15:08:09 2018 kern.notice syslog Tue Nov 27 15:08:09 2018 kern.notice syslog	IOT.NQTT: Check for sensor update IOT.NQTT: Check for sensor update IOT.NQTT: Found Local Channels: IOT.NQTT: Found Local Channels: IOT.NQTT: T: recomber Sion week: FAIL IOT.NQTT: Found Local Channels: IOT.NQTT: Server Found Index FAIL IOT.NQTT: Server Found Index (things/cLIENTID/data/CHANNEL IOT.NQTT: Data Format: DATA IOT.NQTT: Check for sensor update IOT.NQTT: Check for sensor update IOT.NQTT: Check for sensor update IOT.NQTT: Check for sensor update IOT.NQTT: Find Anche Entry for IOO
Tue Nov 27 15:08:09 2018 kern.notice syslog Tue Nov 27 15:08:09 2018 kern.notice syslog Tue Nov 27 15:08:09 2018 kern.notice syslog	: [IOT.MQTT]: Find Match Entry for 100 [IOT.MQTT]: [-t] v1/e74b78d0-3858-11e7-afce-8d5fd2a310a7/things/2b1fab30-3859-11e7-afce-8d5fd2a310a7/data/0
root@dragino-193a18:~#	

4.5.2 Configure LG01-N's Radio frequency

Now we should configure LG01-N's radio parameter to receive the LoRaWAN packets. We are using 868.0Mhz (868000000 Hz) as below:



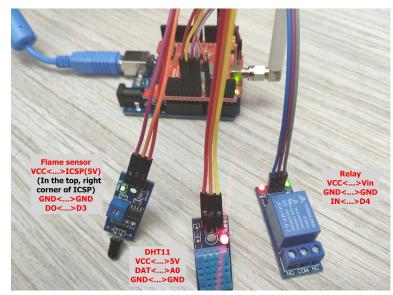
www.c	Iragino.	com

dragino-1893c4 sta			
Latitude	22.73		
Longtitude	114.23		
Radio Power (Unit:dBm)	range 5 ~ 20 dBm		
Radio Settings			
Radio settings for Channel			
Frequency (Unit:Hz)	868000000		
Spreading Factor	SF7 *	▲ 2	
Coding Rate	4/5 *		
Signal Bandwidth	125 kHz *		
Preamble Length	8		
	I Length range: 6 ~ 65536		
LoRa Sync Word	52		
	Value 52(0x34) for LoRaWAN		,
Encryption Key	Encryption Key		3
			Save & Apply Save Reset



4.6 Create LoRa Shield End Node

4.6.1 Hardware Connection



There are three sensors connect to the LoRa Shield + UNO. These sensors are flame sensors, DHT11 (Temperature & Humidity sensor) and Relay. Please use the connection as we show in the photo.

Note: There is a trick above, the relay is connected to VIN. In this case, The UNO can only be power via USB port. If need to power via DC power adapter, please use another 5v pin to power relay.

Upload <u>this sketch</u> to the UNO, this sketch will send temperature and humidity data to gateway at every 60 seconds. If there is a flame detect, it will then immediately send the value to gateway and then upload to the IoT Server.



4.6.2 Test with uplink

After we upload the sketch to UNO, we can see below output from Arduino

MQTT_DHT11_and_Flame_sensor_Client_updata_to_ThingSpeak	COM12 (Arduino/Genuino Uno)
文件 编辑 项目 工具 帮助	
	The temperature and humidity:
MQTT_DHT11_and_Flame_sensor_Client_updata_to_ThingSpeak_	[24.00°C, 65.00%]
1 #include <dht.h></dht.h>	
2 #include <spi.h></spi.h>	The packet is send successful
3 #include <lora.h></lora.h>	############ COUNT=4 ####################################
4	The temperature and humidity:
5// Singleton instance of t	[24.00°C, 65.00%]
6	The packet is send successful
7 dht DHT;	############ COUNT=5 ####################################
8 #define DHT11_PIN A0	The temperature and humidity:
9 const int flame_pin=3; //	[24.00°C, 65.00%]
10 float temperature, humidity	The packet is send successful
11 char tem $1[8] = {" \setminus 0"}_{m}$ hum 1	########### COUNT=6 ####################################
上作成功。	The temperature and humidity:
avrdude done. Thank you.	[24.00°C, 65.00%]
	The packet is send successful

And we can see the logread of gateway as below, means the packet arrive gateway:

3 10.130.2.125 - SecureCRT	
文件(F) 編編(E) 查覆(V) 迭顷(O) 传输(T) 脚本(S) 工具(L) 帮助(H)	
3 W 口 2 N = 5 A - 5 G - 5 M + 6 A - 5 G - 1 M + 6 A - 5 G	
10.130.2.125	
17 - 18 - 18 - 18 - 18 - 18 - 18 - 18 -	

Finally, we can see on the ThingSpeak:

	nnel Settings Shar	ing API Keys D	lata import / Export		
Add Visualizations	Widgets Expo	rt recent data	I	MATLAB Analysis	MATLAB Visualization
Channel Stats reated: <u>about17 hours ago</u> ast entry: <u>2 minutes ago</u> ntries: 30					
Field 1 Chart		3 0 1 ×	Field 2 Chart		B O / X
	HT11 and Flame ser			DHT11 and Flame so	



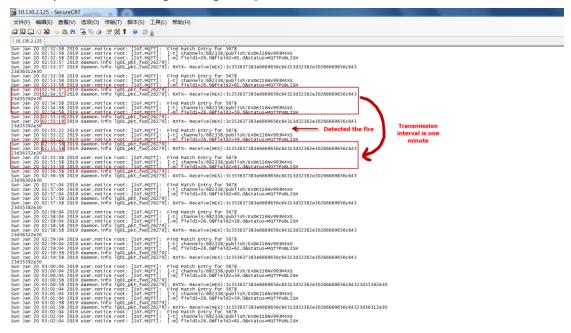
4.6.3 Test with interrupt by flame detect

The DO pin of Flame sensor is high in normal state. When a flame is detected, the DO pin of Flame sensor will become low, then, the UNO generates an external interrupt, and immediately uploads the temperature and humidity to the server.

The DO pin of Flame sensor is low when a flame is detected, and we can see on the Serial Monitor:

MQTT_DHT11_and_Flame_sensor_Client_updata_to_ThingSpeak_ Arduino 1	COM12 (Arduino/Genuino Uno)
文件編編项目工具構助	
MOTT_DHT11_and_Flame_sensor_Client_updata_to_ThingSpeak_ 73 void fire() // Interrupt	The temperature and humidity:
74 {	[26.00°C, 61.00%]
75 Serial.println("Have fire,	
76 dhtTem();	############## COUNT=3 ####################################
<pre>77 dhtWrite();</pre>	The temperature and humidity:
78 LoRa. beginPacket();	[26.00°C,61.00%] The packet is send successful
79 LoRa. print((char *)dataser	######################################
80 LoRa. endPacket ();	The temperature and humidity:
81 }	[26. 00°C, 59. 00%]
83 void SendData()	The packet is send successful
 III 	Have fire, the temperature is send
avrdude done. Thank you.	The temperature and humidity:
	[26.00°C, 59.00%]

Similarly, we can see the logread of gateway via SSH access:





Finally, we can see on the ThingSpeak:

, ThingSpeak™	Channels - Apps -	Community Supp	ort - Corr	mercial Use How to E	Buy Account - Sign Ou
Private View Public View	Channel Settings	Sharing API Keys	Data Import / Export		
Add Visualizations	Add Widgets	Export recent data		MATLAB Analysis	MATLAB Visualization
Channel Stats reated: <u>about 17 hours ago</u> ast entry: <u>2 minutes ago</u> ntries: 30 Field 1 Chart		୯୨≁×	Field 2 Chart		₫ Ç ℓ ¥
Test from	n the DHT11 and Flan	ne sensor	Test from th	e DHT11 and Flame s	sensor



4.6.4 Test with downlink

The http downlink feature is now support since firmware LG02_LG08--build--v5.2.1560931576--20190619-1607.

ThingSpeak downlink command can be found in TalkBack App.

The **Command String input box** is the command you want to send to LoRa device.

🖵 Apps - TalkBack - ThingSpea 🗙	api.thingspeak.c	om/talkbacks/30 🗙	÷									
←) → ♂ ☆	D 🔒 https://thi	ngspeak.com/apps/	talkbacks/3	06 🗱 🖸	80%	☆ ⊻	Q. 搜索		lil\	9 1		
京东商城 王 天猫618												
	🖵 ThingSp	eak™ Channels -	Apps -	Community	Support -		Commercial Use	How to Buy	Account +	Sign	Dut	
			All Apps			Add a TalkBack (Command					
	Name:	test	MATLAB A	nalysis		POST https://api.thingspeak.com/talkbacks/30660/commands.json						
	TalkBack ID:	TalkBack ID: 30660		sualizations		api_key=JZ3X4Y9HTCZN+9YO						
	API Key:	JZ3X4	Plugins			Get a TalkBack C						
			ThingTwee TimeConti			GET https://api.	thingspeak.com/talkbacks,	/30660/commands/C	OMPAND_ID.json?	api_key=J	z	
		Reg	React	01		Update a TalkBa				,		
	Created:	2019-01	TalkBack				thingspeak.com/talkbacks,	/30660/commands//	ONNUE TO SEA			
	Logged to Chann	el: dragino	ThingHTT	P			4Y9MTCZNH9YO	, 50000, 60000, 60				
	00					Execute the Next	: TalkBack Command				7	
							thingspeak.com/talkback	s/30660/commands/	execute.json			
	Commands					Update a Channel and Execute the Next TalkBack Command						
	Position	Command ID	Command	string		POST https://api field1=70	.thingspeak.com/update.j	son				
	1	16007288	DownlinkTe	est		api_key=P07	KVY59P5QEY6N6 Iy=3Z3X4Y9NTCZNH9Y0					
Γ	Position Comm	and string				Get the Last Exec	uted Command					
	DownlinkTest Save					GET https://api.	thingspeak.com/talkbacks,	/30660/commands/1	ast.json?api_ke	y=3Z3X4Y9	н	
L						•				Þ		

Execute The next Talkback Command is the API to get one command from the commands queue.

We can test in the web with this API. Format is:

https://api.thingspeak.com/talkbacks/XXXXX/commands/execute.json?api_key=XXXXXXXXX Result as below:

Apps - TalkBack - Apps - TalkBack - Thi	
$\leftarrow \rightarrow \bigcirc \bigcirc$	🛈 🔒 https://api.thingspeak.com/talkbacks/30660/commands 🗸 🎇 🚥 🏠 👱 🔍
🖨 京东商城 🔳 天雄	曲618
JSON 原始数据	头 头
保存复制全部折叠	全部展开
id: 1	16007322
command_string:	"DownlinkTest"
position: r	null
executed_at:	"2019-06-19T13:35:03Z"
created_at:	"2019-06-19T13:34:592"

We can also test this API in LG01-N Linux console:

By using:

☐ 172.31.255.254 - SecureCRT
文件(F) 编辑(E) 查看(V) 远项(O) 传输(T) 脚本(S) 工具(L) 帮助(H)
(1) 認 (二) (1) (1) (1) (1) (1) (1) (1) (1) (1) (1
172.31.255.254
root@dragino-1b9e2c:~# curl -k https://api.thingspeak.com/talkbacks/30660/commands/execute.json?api_key=JZ3X4Y9MTCZNH9Y0

root@dragino_lb922c:-# curl -k https://api.thingspeak.com/talbbacks/30660/commands/execute.json?api_key=lz3X4Y9WTCZNH9YO
{"id":16007340,"command_string":"DownlinkTest","position":null,"executed_at":"2019-06-19T13:40:02z","created_at":"2019-06-19T13:39:58z"}



To get this result automatically in LG01-P and send out via LoRa, we can configure as below:

,. ..., ..., ...,,, ,, ,,

查看网站信息)-1b9e2c Stat	tus ▼ System ▼ Network ▼ Service ▼ Logout
HTTP / HTTPS	er
General Settings	
Enable SSL Connection	
Enable HTTP Downlink	✓ Ø Forward downlink data via LoRa
Downlink URL	https://api.thingspeak.com/talkback
Downlink Datatype	Json - One Level
Downlink Parameter	command_string
Downlink Poll Interval	5
	@ unit:seconds.
	Save & Apply

- > Because URL is https, So need to Enable SSL Connection
- > Downlink URL use the URL we use in Web and Curl
- > Downlink datatype for ThingSpeak is Json.
- Downlink Parameter is command_string. We will fetch the value of command_string from the downlink data string.
- LG01-N will poll the URL every 5 seconds. When there is valid command_string found, it will send out via LoRa (Radio parameter is defined in LoRaWAN gateway Radio settings or Radio2 settings for LG02)

Result in the LoRa Shield:

The temperature and humidity:	
[30.00°C, 86.00%]	
Packet Sent	
############## COUNT=26 ####################################	
The temperature and humidity:	=
[30. 00°C, 87. 00%]	=
Packet Sent	
Received packet : DownlinkIest	
	-
☑ Autoscroll ☑ Autoscroll No line ending ▼ 9600 baud ▼ Clear output	



4.7 Conclusion and limitation

4.7.1 Overview for the example

This example shows how to set up a simple LoRa network with ThingSpeak IoT server. In this example, we use the raw LoRa protocol (private protocol) for transmission. It is simpler compare via LoRaWAN protocol

There are some frequently ask points for the example:

1/ Difference between LoRaWAN & Private LoRa protocol:

- The private LoRa protocol here doesn't have MAC control/management, (of course developer can develop this). In LoRaWAN protocol, this feature is supported already.
- The transmission is unencrypted in this example, user can see the data in gateway. In LoRaWAN, the transmission is designed in AES encryption.
- Private LoRa protocol means the gateway only works with specify LoRa End node which runs the same protocol, the gateway can't work with a standard LoRaWAN devices.
- Private LoRa protocol doesn't need the LoRaWAN IoT Server. Gateway can send data to user defined IoT server, in terms the gateway and the server can communicate with each other.
- User can more features in the private protocol such as MAC control, encryption, that is how LoRaWAN protocol comes, the advantage of LoRaWAN protocol is that it is designed for carrier level use, and developer can use it directly with many features and compatible with the LoRaWAN node from different manufacturers.



5 Order Info

LoRa_IoT_Kit-v2-XXX-YYY

XXX: Frequency Band

433: For Bands: EU433, CN470 868: For Bands: EU868,IN865 915: For Bands: US915,AU915,AS923,KR920

YYY: 4G Cellular Option

EC25-E: EMEA, Korea, Thailand, India.

EC25-A: North America/ Rogers/AT&T/T-Mobile.

EC25-AU: Latin America, New Zeland, Taiwan

EC25-J: Japan, DOCOMO/SoftBank/ KDDI

More info about valid bands, please see EC25-E product page

(https://www.quectel.com/product/ec25.htm)



6 FAQ & Trouble Shooting

6.1 I can't upload sketch to LoRa Shield in MAC OS, shows " dev/cu.usbmodem1421 is not available "

Error Info as below:

Arduino: 1.8.3 (Mac OS X), Board: "Arduino/Genuino Uno" Archiving built core (caching) in: /var/folders/jq/8fnvlfj90tgbnbcyd16_bbw00000gn/T/arduino_cache_833512/core/core_arduino _avr_uno_fc9a32205aafa27e4eda988d5ed9b7ac.a Sketch uses 20142 bytes (62%) of program storage space. Maximum is 32256 bytes. Global variables use 1189 bytes (58%) of dynamic memory, leaving 859 bytes for local variables. Maximum is 2048 bytes. Board at /dev/cu.usbmodem1421 is not available

The Arduino UNOs in the Kit are clone version and use CH340 USB to serial chip. User has to install the CH340 driver in PC to make it work. Above issue means the MAC OS doesn't has CH340 driver.

6.2 My IoT Kit has the model LG01-P instead of LG01-N, Can I still use this manual.

The gateway part of this manual is for LG01-N, if user has the LG01-P version, please check the LG01-P gateway manual.



7 Technical Support

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

support@dragino.com



8 Reference

- 1) LORaWAN official website. And Technicel document for LoRaWAN.
- 2) LG01-N LoRa Gateway User Manual
- 3) LoRa Low Energy design guide and Calculator Tool.
- 4) About Distance: LoRa Modem Design Guide
- 5) <u>SX1276 download resource</u>.
- 6) User Manual: LG01-N, LoRa Shield, LoRa/GPS Shield