



**RAK**

**RAKwireless**  
Technology Co., Ltd.

# **WisNodeV1.3-LoRa-Arduino Library Use Guide V1.1**

Shenzhen Rakwireless Technology Co., Ltd.

[www.rakwireless.com](http://www.rakwireless.com)

[info@rakwireless.com](mailto:info@rakwireless.com)

© RAK copyright. All rights reserved.

Companies and product names referred in the instruction belong to trademarks of their respective owners.

Any part of this document may not be reproduced, and may not be stored in any retrieval system, or delivered without RAK's written permission.

The document will be updated without prior notice.



**RAK**

**RAKwireless**  
Technology Co., Ltd.

## CONTENTS

<b>1. RAK811 Arduino Library Use Guide.....</b>	<b>3</b>
<b>(1) Download.....</b>	<b>3</b>
<b>(2) Add to Arduino IDE.....</b>	<b>3</b>
<b>(3)Code introduction.....</b>	<b>4</b>
<b>2. Hardware connect.....</b>	<b>7</b>
<b>3. Flash firmware.....</b>	<b>8</b>
<b>4.Test LoRa Node with LoRaWAN.....</b>	<b>9</b>
<b>5.Quickly connection boards.....</b>	<b>错误！未定义书签。</b>



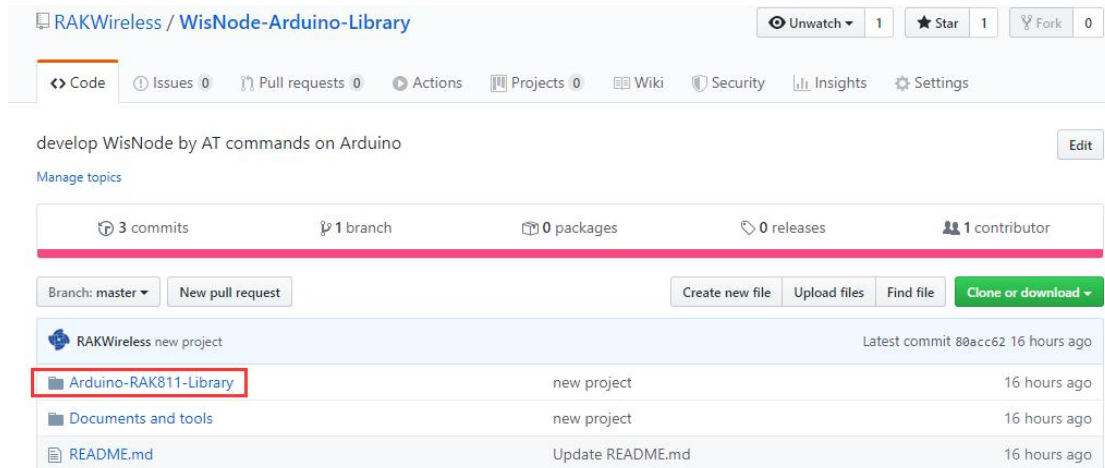
# RAK

RAKwireless  
Technology Co., Ltd.

## 1. RAK811 Arduino Library Use Guide

### (1) Download

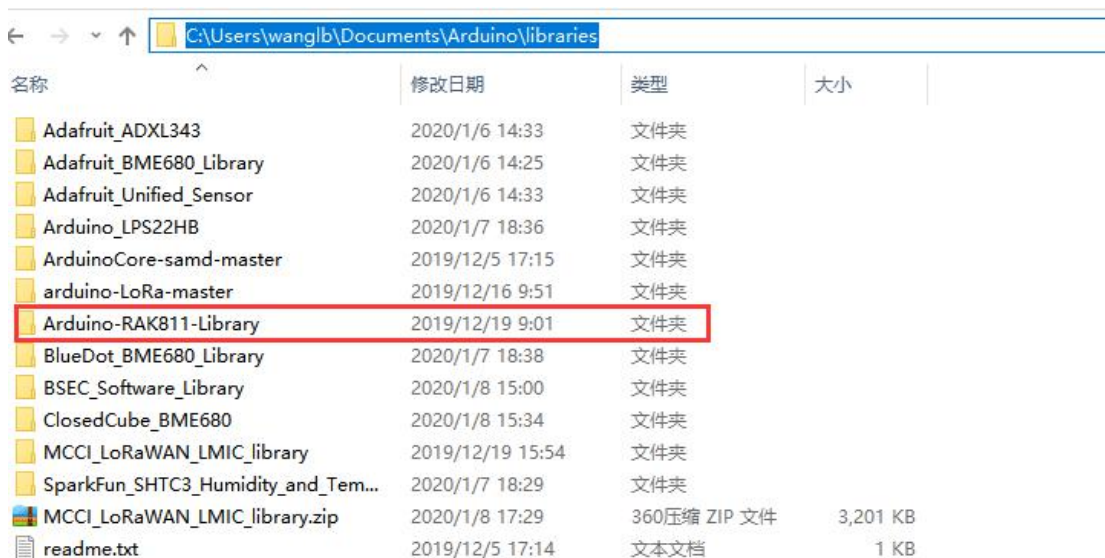
We upload the RAK811 Arduino library code to the official github. You can find this library at github: <https://github.com/RAKWireless/WisNode-Arduino-Library>



Download the library folder “[Arduino-RAK811-Library](#)”.

### (2) Add to Arduino IDE

① Copy the “[Arduino-RAK811-Library](#)” folder to the Arduino library folder.



② And then open the Arduino IDE, you can see the RAK811 sample code in the Arduino example.



# RAK

RAKwireless  
Technology Co., Ltd.

JoinNetworkOTAA | Arduino 1.8.10

File Edit Sketch Tools Help

The screenshot shows the Arduino IDE interface. The 'File' menu is open, and the 'Examples' submenu is displayed. The 'JoinNetworkOTAA' example is highlighted. The background shows a sketch for RAK811 LoRa module.

```
14 #define JOIN_NODE = RAK811
15 String NwkSKey = "69AF
16 String AppSKey = "8419
17 String DevAddr = "2601
18 #endif
19 #define TXpin 11 //
20 #define RXpin 10
21 #define ATSerial Serial
22 SoftwareSerial DebugSe
23 char buffer[] = "72616B
24
25 bool InitLoRaWAN(void)
26 RAK811 RAKLoRa(ATSerial
27
28
29 void setup() {
30   DebugSerial.begin(11
31   while(DebugSerial.re
32   while(!DebugSerial);
33   DebugSerial.println(
34
35   ATSerial.begin(11520
36   while(ATSerial.avail
37   {
38     ATSerial.read();
39   }
40
41   DebugSerial.println(
42   delay(200);
43
44   DebugSerial.println(
45   if(!RAKLoRa.rk_setSe
46   {
47     DebugSerial.printl
```

### (3)Code introduction

On the library contains the available functions, the user can refer to the RAK811.h file, which has a detailed note on the use of each function.



# RAK

RAKwireless  
Technology Co., Ltd.

```
1  /*  
2  * A library for controlling RAK811 LoRa radio.  
3  *  
4  * @Author Leopold.wang  
5  * @Date 14/01/2020  
6  */  
7  
8  
9  
10 #ifndef RAK811_h  
11 #define RAK811_h  
12 #define LoRaWAN 0  
13 #define LoRaP2P 1  
14 #define OTAA 0  
15 #define ABP 1  
16  
17 #include "Arduino.h"  
18  
19 // #define DEBUG_MODE  
20  
21 class RAK811  
22 {  
23 public:  
24  
25 /*  
26 * A simplified constructor taking only a Stream ((Software/Hardware)Serial) object.  
27 * The serial port should already be initialised when initialising this library.  
28 */  
29 RAK811(Stream& serial,Stream& serial1);  
30  
31 /*  
32 * Gets the firmware version number of the module.  
33 * Only applies to the firmware that the module programmed for the RAK811 AT command.  
34 * AT commands refer to: https://downloads.rakwireless.com/en/LoRa/RAK811/Application\_Notes/Get\_Start\_with\_RAK811\_WisNode-LoRa.pdf  
35 */  
36 String rk_getVersion(void);  
37  
38 /*  
39 * Get the frequency band of the module.  
40 * This feature request to receive at least 800 bytes buffer size.  
41 */  
42 String rk_getLoRaStatus(void);  
43  
44 /*  
45 * Let the module enter the ultra low power sleep mode.  
46 * When the module is in sleep mode, the host can send any character to wake it up.  
47 * mode = 0->wakeup, 1-> sleep  
48 * When the module is awakened, the event response will automatically return through the serial information.  
49 */  
50 void rk_sleep(int mode);  
51  
52 /*  
53 * Reset the module or reset the LoRaWAN or LoRaP2P protocol stack.  
54 * mode = 0: Reset and restart module.  
55 * mode = 1: Reset LoRaWAN or LoRaP2P stack and Module will reload LoRa configuration from EEPROM.  
56 */  
57 void rk_reset(int mode);  
58
```

**Note:** Before compile by Arduino IDE, user should better configure Serial RX and TX buffer size. This must be configured manually in Arduino installation directory. The following are the minimum recommended sizes.

名称	修改日期	类型	大小
abi.cpp	2017/12/11 16:14	C++ Source File	2 KB
Arduino.h	2019/5/16 20:52	H 文件	8 KB
binary.h	2019/5/16 20:52	H 文件	11 KB
CDC.cpp	2019/5/16 20:52	C++ Source File	9 KB
Client.h	2019/5/16 20:52	H 文件	2 KB
HardwareSerial.cpp	2019/5/16 20:52	C++ Source File	9 KB
HardwareSerial.h	2020/1/16 14:48	H 文件	6 KB
HardwareSerial_private.h	2019/5/16 20:52	H 文件	5 KB
HardwareSerial0.cpp	2019/5/16 20:52	C++ Source File	3 KB
HardwareSerial1.cpp	2019/5/16 20:52	C++ Source File	3 KB
HardwareSerial2.cpp	2019/5/16 20:52	C++ Source File	2 KB
HardwareSerial3.cpp	2019/5/16 20:52	C++ Source File	2 KB
hooks.c	2017/11/27 19:21	C 文件	2 KB
IPAddress.cpp	2019/5/16 20:52	C++ Source File	3 KB
IPAddress.h	2019/5/16 20:52	H 文件	3 KB
main.cpp	2019/5/16 20:52	C++ Source File	2 KB
new.cpp	2017/12/11 16:14	C++ Source File	2 KB
new.h	2017/12/11 16:14	H 文件	1 KB
PluggableUSB.cpp	2019/5/16 20:52	C++ Source File	3 KB
PluggableUSB.h	2019/5/16 20:52	H 文件	3 KB
Print.cpp	2019/5/16 20:52	C++ Source File	6 KB
Print.h	2019/5/16 20:52	H 文件	3 KB
Printable.h	2019/5/16 20:52	H 文件	2 KB
Server.h	2019/5/16 20:52	H 文件	1 KB
Stream.cpp	2019/5/16 20:52	C++ Source File	9 KB
Stream.h	2019/5/16 20:52	H 文件	6 KB
Tone.cpp	2019/5/16 20:52	C++ Source File	15 KB
Udp.h	2019/5/16 20:52	H 文件	5 KB
USBAPI.h	2019/5/16 20:52	H 文件	7 KB
USBCore.cpp	2019/5/16 20:52	C++ Source File	20 KB
USBCore.h	2019/5/16 20:52	H 文件	9 KB
USBDesch	2019/5/16 20:52	H 文件	2 KB
WCharacter.h	2019/5/16 20:52	H 文件	5 KB
WInterrupts.c	2019/5/16 20:52	C 文件	10 KB
wiring.c	2018/10/29 15:58	C 文件	12 KB
wiring_analog.c	2017/12/18 15:53	C 文件	8 KB
wiring_digital.c	2019/5/16 20:52	C 文件	5 KB





# RAK

RAKwireless  
Technology Co., Ltd.

```
C:\Program Files (x86)\Arduino\hardware\arduino\avr\cores\arduino\HardwareSerial.h - Notepad++ [Administrator]
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?

HardwareSerial.h
28 #include "Stream.h"
29
30 // Define constants and variables for buffering incoming serial data. We're
31 // using a ring buffer (I think), in which head is the index of the location
32 // to which to write the next incoming character and tail is the index of the
33 // location from which to read.
34 // NOTE: a "power of 2" buffer size is recommended to dramatically
35 // optimize all the modulo operations for ring buffers.
36 // WARNING: When buffer sizes are increased to > 256, the buffer index
37 // variables are automatically increased in size, but the extra
38 // atomicity guards needed for that are not implemented. This will
39 // often work, but occasionally a race condition can occur that makes
40 // Serial behave erratically. See https://github.com/arduino/Arduino/issues/2405
41
42 #if !defined(SERIAL_TX_BUFFER_SIZE)
43 #if ((RAMEND - RAMSTART) < 1023)
44 #define SERIAL_TX_BUFFER_SIZE 16
45 #else
46 #define SERIAL_TX_BUFFER_SIZE 70
47 #endif
48 #endif
49 #if !defined(SERIAL_RX_BUFFER_SIZE)
50 #if ((RAMEND - RAMSTART) < 1023)
51 #define SERIAL_RX_BUFFER_SIZE 16
52 #else
53 #define SERIAL_RX_BUFFER_SIZE 110
54 #endif
55 #endif
56 #if (SERIAL_TX_BUFFER_SIZE > 256)
57 typedef uint16_t tx_buffer_index_t;
58 #else
59 typedef uint8_t tx_buffer_index_t;
60 #endif
61 #if (SERIAL_RX_BUFFER_SIZE > 256)
62 typedef uint16_t rx_buffer_index_t;
63 #else
64 typedef uint8_t rx_buffer_index_t;
65 #endif
```

If the RAM size of Arduino board is enough big, these two buffer size need config better bigger.

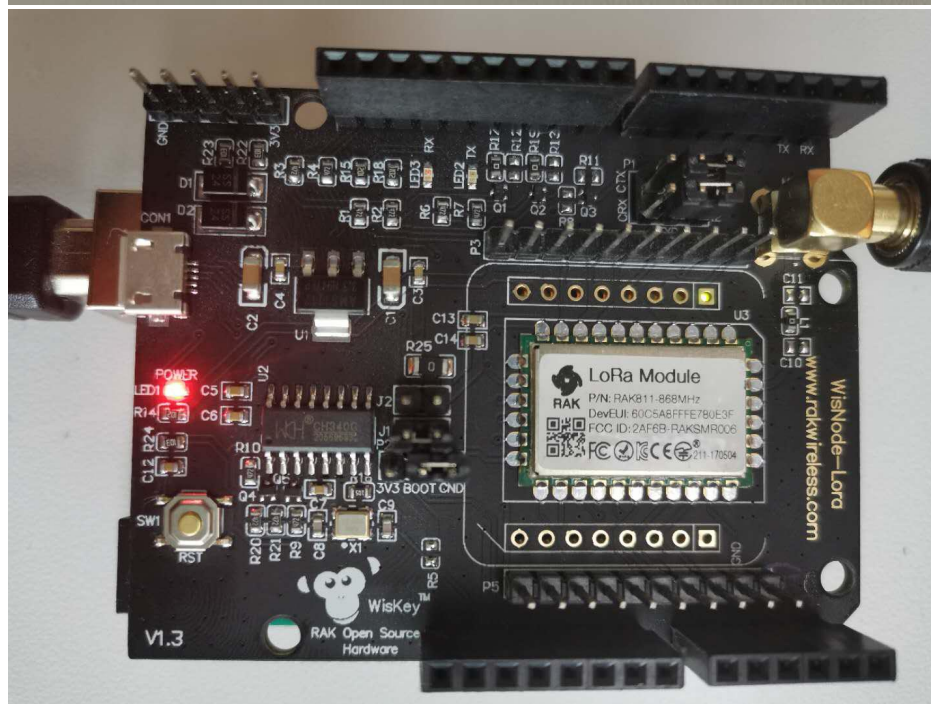
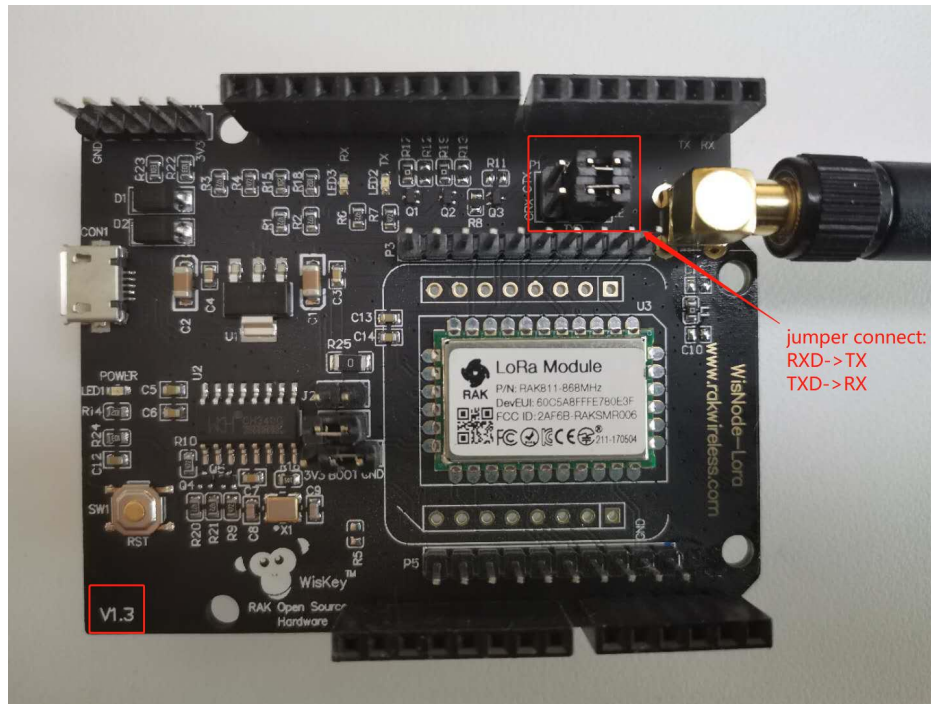


RAK

RAKwireless  
Technology Co., Ltd.

## 2. Hardware connect

This document will use **Arduino Uno + WisNode-LoRa** as an example.





# RAK

RAKwireless  
Technology Co., Ltd.

## 3. Flash firmware

① Flash firmware for RAK811 refer to :[Get Start with RAK811 WisNode-LoRa.pdf](#)

RAK811-Firmware:<https://github.com/RAKWireless/WisNode-Arduino-Library/tree/master/Documents%20and%20tools/RAK811-Firmware>

② Compile and Flash app demo [JoinNetworkOTAA.ino](#) through Arduino IDE.

The screenshot shows the Arduino IDE interface with the 'JoinNetworkOTAA.ino' sketch loaded. The 'Tools' menu is open, and the 'Board' dropdown is set to 'Arduino/Genuino Mega or Mega 2560'. The 'Port' dropdown is set to 'COM31 (Arduino/Genuino Mega or Mega 2560)'. The 'Burn Bootloader' option is selected. The 'Sketch' menu is open, and the 'Upload' button is highlighted. A red box highlights the 'Upload' button and the 'Burn Bootloader' option.

The code in the sketch is as follows:

```
13 String AppKey = "000F1023555F9FF74D1A55202EDF2B1";
14 #else JOIN_MODE == ABP
15 String DevAddr = "49AF20A2A3C01B243940A30C9175D43";
16 String AppKey = "04186913ACD008C28E2475D70F3220";
17 String DevAddr = "260125D7";
18 #endif
19 #define TXpin 11 // Set the virtual serial port pins
20 #define RXpin 10
21 #define ATSerial Serial
22 SoftwareSerial DebugSerial(RXpin, TXpin); // Declare a virtual serial port
23 char buffer[] = "T2614B76972656C657373";
24
25 bool InitLoRaWAN(void);
26 RAK811 LoRaWAN(ATSerial, DebugSerial);
27
28
29 void setup() {
30   DebugSerial.begin(115200);
31   while(DebugSerial.read() >= 0) {}
32   while(!DebugSerial);
33   DebugSerial.println("StartUP");
34
35   ATSerial.begin(115200); // Note: Please manually set the baud rate of the WisNode device to 115200.
36   while(ATSerial.available())
37   {
38     ATSerial.read();
39   }
40
41   DebugSerial.println(RAKLoRa.ch_getVersion()); //get RAK811 firmware version
42   delay(2000);
43   DebugSerial.println("Start init RAK811 parameters...");
44   if(RAKLoRa.ch_getSendInterval(0,0) //close auto join and send to LoRaWAN
45   {
46     DebugSerial.println("Close auto join and send to LoRaWAN failed, please reset module.");
47     while(1);
48   }
49   if (!InitLoRaWAN()) //init LoRaWAN
50   {
51     //init LoRaWAN failed
52   }
53 }
```

The output window shows the following messages:

```
Done uploading.
Reading ( #####) : st500v2_ReceiveMessage(): timeout
##### : st500v2_ReceiveMessage(): timeout
##### : 1008 11.33s
avrduide: verifying ...
avrduide: 10238 bytes of flash verified
avrduide done. Thank you.
```





# RAK

RAKwireless  
Technology Co., Ltd.

## 4. Test LoRa Node with LoRaWAN

Wisnode V1.1 server log——Serial console log:

Applications / app / Devices / ceshi				Serial console log
DETAILS	CONFIGURATION	KEYS (OTAA)	ACTIVATION	
UPLINK	4:56:31 PM	UnconfirmedDataUp	01b2d425	[16:55:33.234]收←◆FW: RUI v3.0.0.13.H.T1 OK Start init RAKB11 parameters...
UPLINK	4:56:15 PM	UnconfirmedDataUp	01b2d425	[16:55:35.474]收←◆Current work region: EU868 [16:55:44.473]收←◆RAKB11 init OK! Start to join LoRaWAN...
UPLINK	4:55:58 PM	UnconfirmedDataUp	01b2d425	[16:55:54.450]收←◆[LoRa]:Join Success OK Join LoRaWAN success
DOWNLINK	4:55:46 PM	JoinAccept		[16:55:56.687]收←◆Start send data...
UPLINK	4:55:46 PM	JoinRequest	8680000000000001	[16:56:02.634]收←◆[LoRa]: RUI_MCPS_UNCONFIRMED send success OK Go to Sleep.
UPLINK	4:55:25 PM	UnconfirmedDataUp	009d2514	[16:56:13.295]收←◆Start send data...
				[16:56:19.447]收←◆[LoRa]: RUI_MCPS_UNCONFIRMED send success OK Go to Sleep.
				[16:56:29.908]收←◆Start send data...
				[16:56:36.056]收←◆[LoRa]: RUI_MCPS_UNCONFIRMED send success OK Go to Sleep.
				[16:56:46.517]收←◆Start send data...
				[16:56:52.664]收←◆[LoRa]: RUI_MCPS_UNCONFIRMED send success OK Go to Sleep.



**RAK**

**RAKwireless**  
Technology Co., Ltd.

If you have any questions, welcome to our forum to ask your question:

<http://support.rakwireless.com/>.

You can also send your question to this email: [ken.yu@rakwireless.com](mailto:ken.yu@rakwireless.com)