WisNode Arduino Library API Manual V1.0

Shenzhen Rakwireless Technology Co., Ltd.
www.rakwireless.com
info@rakwireless.com

1.bool rk_getVersion(void);

@Brief gets the firmware version number of the module.

@Return true/false:true->success,false->error

@param none

2. void rk_sleep(int mode);

@Brief Let the module enter the ultra low power sleep mode, When the module is in sleep mode, the host can send any character to wake it up.

@Return none

@param int mode：0->wakeup, 1-> sleep

3. void rk_reset(void);

@Brief Reset the module.

@Return none

@param none

4. bool rk_setSendinterval(int mode,int value);

@Brief

This command is used to set the interval time of sending data.

This feature can't use on RAK811 module,so must

manually close the feature before user APP.

@Return true/false:true->success,false->error

@param int mode:0->close auto send data,1->auto send with sleep,2->auto send with no sleep.
int value:auto send interval value

5.bool rk_setRate(int rate);

@Brief Use to change the next send data rate temporary when a dr function is off.

@Return true/false:true->success,false->error

@param int rate:DR value,the range depends on the protocol stack.

6.bool rk_setClass(int classMode);

@Brief Use to change the LoRaWAN class.

@Return true/false:true->success,false->error

@param int classMode:0->ClassA,
1->ClassB,
2->ClassC

7.bool rk_setRegion(int region);

@Brief Use to change the LoRaWAN region.

@Return true/false:true->success,false->error

@param int region:0->AS923,

1->AU915, 2->CN470,

3->CN779, 4->EU433,

5->EU868, 6->KR920,

7->IN865, 8->US915,

9->US915_Hybrid

8. bool rk_setWorkingMode(int mode);

@Brief Set the module work mode, the module defaults to LoRaWAN mode.This command could cause modual reset,so this function must only be executed once.

@Return true/false:true->success,false->error

@param int mode: 0-> Set the module to LoRaWAN mode.

1: Set the module to LoRaP2P mode.

9. bool rk_initOTAA(String devEUI, String appEUI, String appKEY);

@Brief Initialize the module parameter, which is the parameter that the module must use when adding the OTAA to the network.

@Return true/false:true->success,false->error

@param String devEUI : Device EUI as a HEX string(16bytes). Example "60C5A8FFFE000001",

String appEUI : Application EUI as a HEX string(16bytes). Example "70B3D57EF00047C0",

String appKEY : Application key as a HEX string(32bytes). Example "5D833B4696D5E01E2F8DC880E30BA5FE"

10.bool rk_initABP(String devADDR, String nwksKEY, String appsKEY);

@Brief Initialize the module parameter, which is the parameter that the module must use when adding the ABP to the network.

@Return true/false:true->success,false->error

@param String devADDR : The device address as a HEX string(8 bytes). Example "00112233",

String nwksKEY : Network Session Key as a HEX string(32

bytes). Example "3432567afde4525e7890cfea234a5821",

String appsKEY : Application Session Key as a HEX string(32bytes). Example "a48adfc393a0de458319236537a11d90"

11. bool rk_setJoinMode(int mode);

@Brief Set the activation mode to join the network.

@Return true/false:true->success,false->error

@param int mode: 0-> join a network using over the air activation(OTAA)

1->join a network using personalization (ABP).

12. bool rk_joinLoRaNetwork(int timeout);

@Brief Join the LoRaWAN network.Before using this command, you must call one of the rk_setJoinMode functions.

@Return true/false:true->success,false->error

@param int timeout: timeout value (unit:s)

13. bool rk_sendData(int port, char* datahex);

@Brief After joining the network, send the packet at the specified application port.This function can only be used in module work in LoRaWAN mode.

@Return true/false:true->success,false->error

@param int port : The port number.(1-223)

int datahex : hex value(no space). max 220 bytes.

14.bool rk_isConfirm(int type);

@Brief Set LoRa data send package type.This function can only be used in module work in LoRaWAN mode.

@Return true/false:true->success,false->error

@param int type : 0->unconfirm, 1->confirm

15.String rk_recvData(void);

@Brief Used to receive information by the module.This function is used to LoRaWAN mode.

@Return Received the data or event information,string type

@param none

16.bool rk_initP2P(String FREQ, int SF, int BW, int CR, int PRlen, int PWR);

@Brief Initialize the required parameters in LoRaP2P mode.

User must first switch the module operating mode to LoRaP2P mode.

@Return true/false:true->success,false->error

@param String FREQ : frequency, default 869525000Hz with HF module,505300000Hz with LF module.

int SF : spread factor, default 12 (6-12) more low more fast datarate

int BW : Band-with, default 0 (0:125KHz, 1:250KHz, 2:500KHz)

int CR : coding Rate, default 1 (1:4/5, 2:4/6, 3:4/7, 4:4/8)

int PRlen : Preamlen, default 8 (8-65535)

int PWR : Tx power, default 20 (5-20)

17. bool rk_sendP2PData(char* datahex);

@Brief Send the packet .This function can only be used in module work in LoRaP2P mode.

@Return true/false:true->success,false->error

@param int datahex : hex value(no space). max 220 bytes.

18.String rk_recvP2PData(void);

@Brief Used to receive information by the module.This function is used to LoRaP2P mode.

@Return Received the data or event information,string type

@param none

19.bool sendRawCommand(String command);

@Brief Used to send a raw command to the RAK811 module.

@Return true/false:true->success,false->error

@param String command:The raw AT command string,refer to https://downloads.rakwireless.com/en/LoRa/RAK811/Application_Notes/Get_Start_with_RAK811_WisNode-LoRa.pdf

20.bool rk_setWorkingMode(int mode);

@Brief Set the module work mode, the module defaults to LoRaWAN mode.This command could cause modual reset,so this function must only be executed once.

@Return true/false:true->success,false->error

@param int mode:0->Set the module to LoRaWAN mode.

1-> Set the module to LoRaP2P mode.

Notice：

1.Before connecting Arduino, be sure to confirm the AT serial port baud rate of WisNode Board.

Method of inquiring AT serial port baud rate: connect WisNode Board directly with Micro USB, the default is 115200bps. After restarting the Board, log output is specified, indicating that the baud rate is the current baud rate. Otherwise, try to select and modify the baud rate from the following list until there is normal log output.

BaudRate List: 1200,2400,4800,9600,19200,38400,57600,115200.

After the current baud rate is found, you can use "at+set_config=device:uart:1:X" X is the baud rate to be set currently.

2.When updating an Arduino application through serial port of Arduino board.First disconnect the WisNode board from the

Arduino board, then connect it after updating.