



E22/E32/E220系列LORA无线模块

EBYTE串口模组Arduino库驱动使用说明



**成都亿佰特电子科技有限公司**  
Chengdu Ebyte Electronic Technology Co.,Ltd.

## 目录

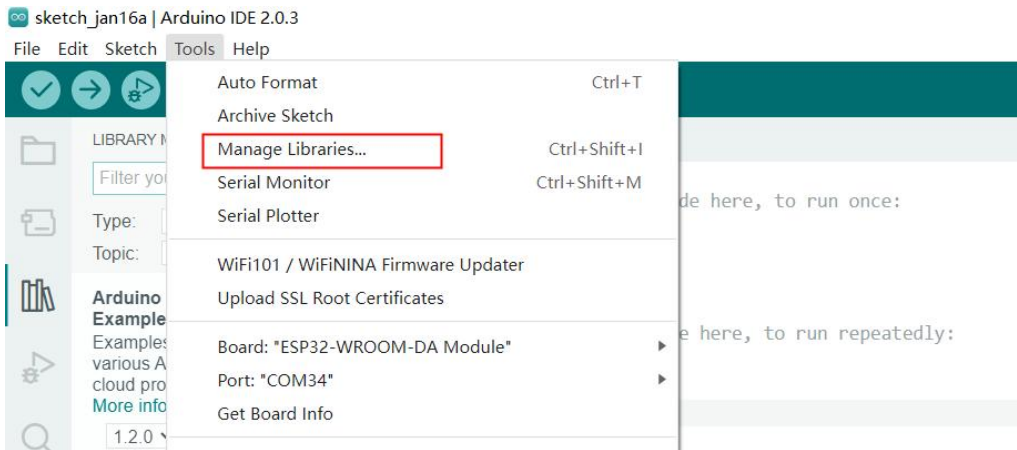
简介: .....	3
第一步 下载安装库文件 .....	3
第二步 E22 例程使用分析 .....	3
第三步 E220 例程使用 .....	11
第四步 E32 例程使用 .....	12
修订历史 .....	14
关于我们 .....	14

## 简介:

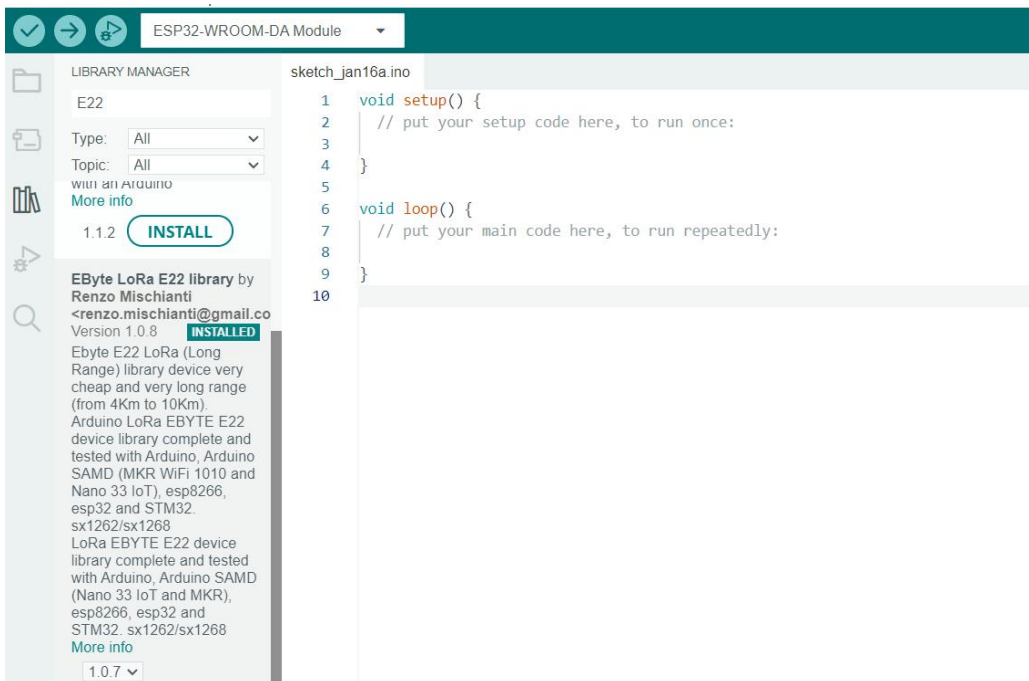
使用Arduino IDE进行编程时都希望越简单越好，一般驱动外设都是调用库文件。我们亿佰特的E22、E32、E220系列Lora串口模组也具有Arduino库文件，可以使用Arduino/esp32/esp8266 三种类型设备以Arduino库文件的方式来驱动这几类Lora串口模块。

## 第一步 下载安装库文件

1、打开Arduino IDE, 选择Tools-->Manage Libraries



2、在库管理中分别搜索E22、E220、E32，选择名字为EBYTE LoRa E22/E32/E220 library by Renzo Mischianti的库安装



## 第二步 E22例程使用分析

### 1、连接方式

E22与Arduino、ESP32、ESP8266连接方式如下:

①按下表顺序连接E22模块和Arduino

E22	Arduino
MO	7
M1	6
RX	4
TX	5
AUX	3
VCC	3.3V/5V
GND	GND

②按下表顺序连接E22模块和esp32

E22	ESP32
MO	19
M1	21
RX	TX2 (17)
TX	RX2 (16)
AUX	18
VCC	3.3V/5V
GND	GND

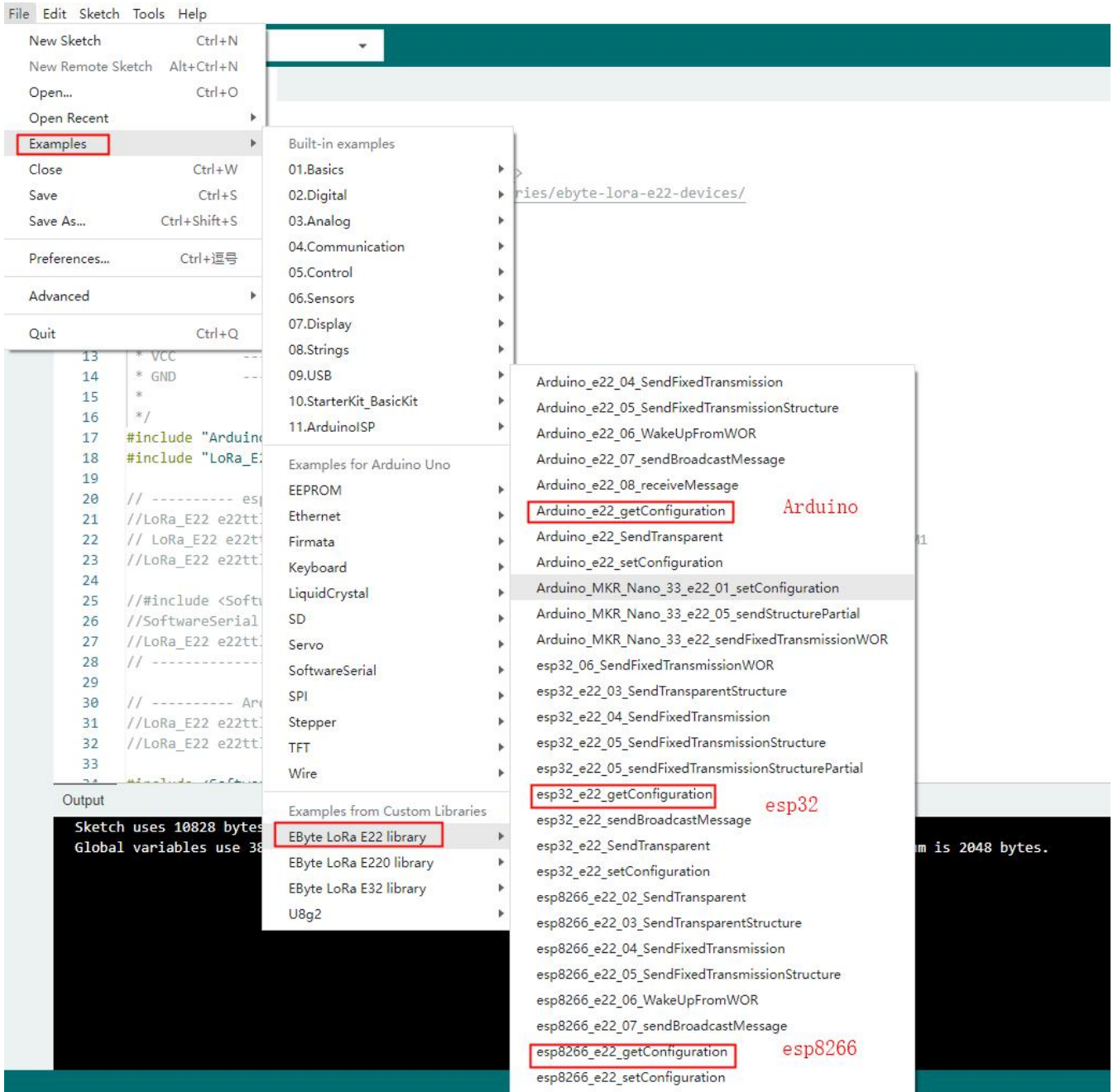
③按下表顺序连接E22模块和esp8266

E22	ESP8266
MO	D7
M1	D6
RX	D3
TX	D4
AUX	D5
VCC	3.3V/5V
GND	GND

## 2、读取模块参数

①打开Arduino IDE,

arduino选择file->Examples->Ebyte LoRa E22 library->Arduino\_e22\_getConfiguration,  
 esp32选择file->Examples->Ebyte LoRa E22 library->esp32\_e22\_getConfiguration,  
 esp8266选择file->Examples->Ebyte LoRa E22 library->esp32\_e22\_getConfiguration。



## ②主程序简要分析

```
// ----- Arduino pins -----
/*指定Arduino 的软件串口引脚、AUX、M0、M1的引脚号*/
LoRa_E22 e22ttl(4, 5, 3, 7, 6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1

// ----- esp32 pins -----
//LoRa_E22 e22ttl(&Serial2, 18, 21, 19); // RX AUX M0 M1

// ----- esp8266 pins -----
//LoRa_E22 e22ttl(RX, TX, AUX, M0, M1); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX
LoRa_E22 e22ttl(D3, D4, D5, D7, D6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1
//使用Arduino/esp32/esp8266的时候各自的引脚初始化不同
void setup() {
    Serial.begin(9600); //初始化打印信息的串口
    while(!Serial){};
    delay(500);

    Serial.println();
    // Startup all pins and UART
```

```
e22ttl.begin(); //初始化与模块连接的串口

ResponseStructContainer c;
c = e22ttl.getConfiguration(); //获取E22模块的配置参数, 该函数内部实现了工作模式切换
// It's important get configuration pointer before all other operation
Configuration configuration = *(Configuration*) c.data;
Serial.println(c.status.getResponseDescription()); //配置模块参数状态
Serial.println(c.status.code);

printParameters(configuration); //打印模块参数

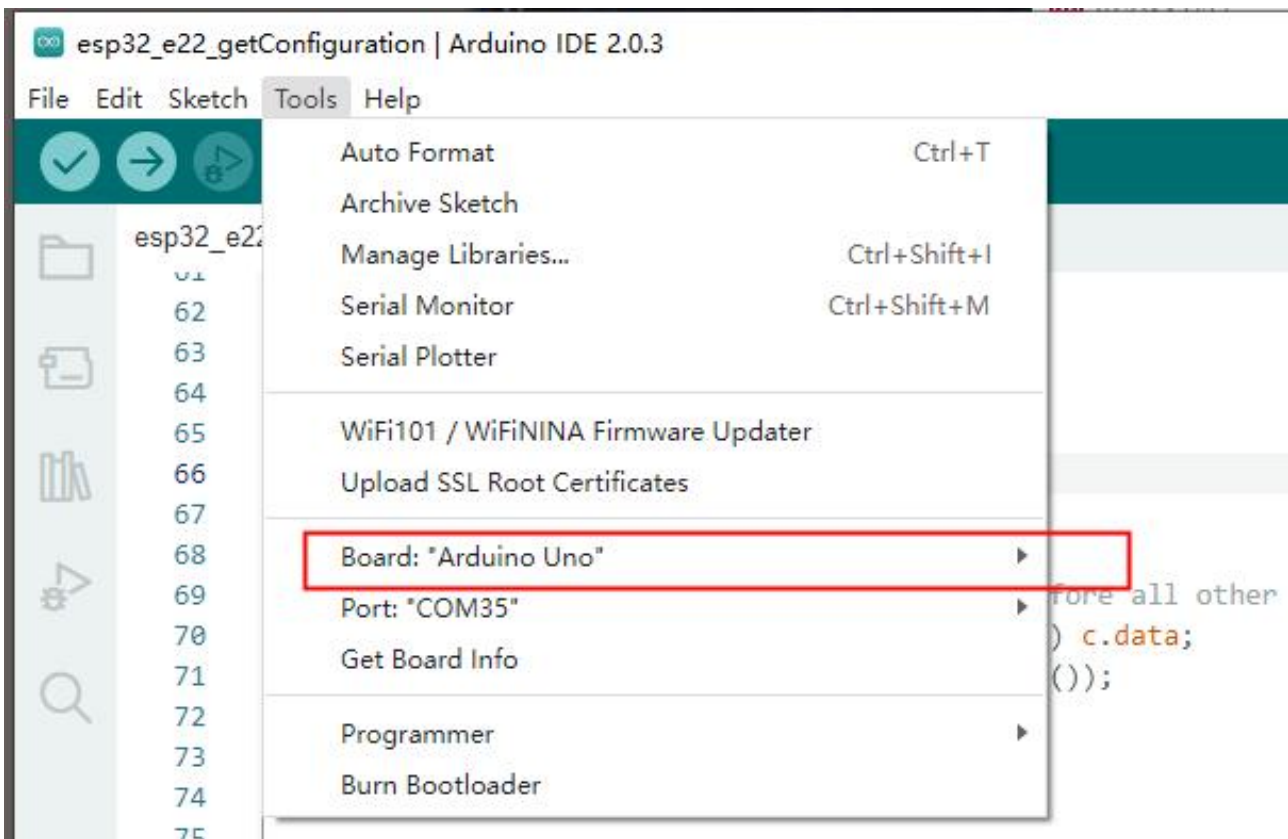
ResponseStructContainer cMi;
cMi = e22ttl.getModuleInformation();
// It's important get information pointer before all other operation
ModuleInformation mi = *(ModuleInformation*)cMi.data;

Serial.println(cMi.status.getResponseDescription());
Serial.println(cMi.status.code);

printModuleInformation(mi); //打印模块产品信息
}
```

③下载程序查看结果, 下载时根据自己的开发板选择对应的型号

Note: 该驱动库显示打印配置信息时是以400频段为标准的, 如果是900频段的模块打印信息会不正确



```

RX: Success
1
-----
HEAD : C1 0 9

AddM : 0
AddL : 0
MetID : 0

Chan : 23 -> 433MHz

SpeedParityBit : 0 -> 8N1 (Default)
SpeedUARTDate : 11 -> 9600bps (default)
SpeedAirDataRate : 10 -> 2.4kbps (default)

OptionSubPacketSett: 0 -> 240bytes (default)
OptionTranPower : 0 -> 22dBm (Default)
OptionRSSIAmbientNo: 0 -> Disabled (default)

TransModeWORPeriod : 11 -> 2000ms (default)
TransModeTransContr: 0 -> WOR Receiver (default)
TransModeEnableLBT : 0 -> Disabled (default)
TransModeEnableRSSI: 0 -> Disabled (default)
TransModeEnabRepeat: 0 -> Disabled (default)
TransModeFixedTrans: 0 -> Transparent transmission (default)
-----
[2023-01-29 16:52:54.283]
RX: Success
1
-----
HEAD: C1 80 7
Model no.: 16
Version : A
Features : 0
-----
    
```

### 3、配置模块参数

#### ①打开Arduino IDE,

arduino选择file->Examples->Ebyte LoRa E22 library->Arduino\_e22\_setConfiguration,  
 esp32选择file->Examples->Ebyte LoRa E22 library->esp32\_e22\_setConfiguration,  
 esp8266选择file->Examples->Ebyte LoRa E22 library->esp32\_e22\_setConfiguration.

#### ②主程序简要分析

```

// ----- Arduino pins -----
/*指定Arduino 的软件串口引脚、AUX、M0、M1的引脚号*/
LoRa_E22 e22ttl(4, 5, 3, 7, 6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1

// ----- esp32 pins -----
//LoRa_E22 e22ttl(&Serial2, 18, 21, 19); // RX AUX M0 M1

// ----- esp8266 pins -----
//LoRa_E22 e22ttl(RX, TX, AUX, M0, M1); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX
LoRa_E22 e22ttl(D3, D4, D5, D7, D6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1
//使用Arduino/esp32/esp8266的时候各自的引脚初始化不同

void setup() {
    Serial.begin(9600);
    while(!Serial){};
    delay(500);

    Serial.println();

    // Startup all pins and UART
    e22ttl.begin();

    ResponseStructContainer c;
    c = e22ttl.getConfiguration(); //获取配置参数
    // It's important get configuration pointer before all other operation
    
```



```

Configuration configuration = *(Configuration*) c.data;
Serial.println(c.status.getResponseDescription());
Serial.println(c.status.code);

printParameters(configuration);//打印配置参数

configuration.ADDL = 0x03; //设置模块地址低位
configuration.ADDH = 0x00; //设置模块地址高位
configuration.NETID = 0x00; //设置模块网络号

configuration.CHAN = 23; //设置模块信道

// ----- DEFAULT TRANSPARENT -----
configuration.ADDL = 0x03;
configuration.ADDH = 0x00;
configuration.NETID = 0x00;

configuration.CHAN = 23;

configuration.SPED.uartBaudRate = UART_BPS_9600; //设置模块波特率
configuration.SPED.airDataRate = AIR_DATA_RATE_010_24; //设置空速
configuration.SPED.uartParity = MODE_00_8N1; //设置模块波特率极性

configuration.OPTION.subPacketSetting = SPS_240_00; //设置模块分包长度
configuration.OPTION.RSSIambientNoise = RSSI_AMBIENT_NOISE_DISABLED; //设置环境rssi是否开启
configuration.OPTION.transmissionPower = POWER_22; //设置模块发送功率

configuration.TRANSMISSION_MODE.enableRSSI = RSSI_DISABLED; //设置数据rssi 是否开启
configuration.TRANSMISSION_MODE.fixedTransmission = FT_TRANSPARENT_TRANSMISSION; //设置是否为定点模式
configuration.TRANSMISSION_MODE.enableRepeater = REPEATER_DISABLED; //设置是否为中继模式
configuration.TRANSMISSION_MODE.enableLBT = LBT_DISABLED; //设置是否开启LBT功能
configuration.TRANSMISSION_MODE.WORTransceiverControl = WOR_RECEIVER; //设置WOR模式
configuration.TRANSMISSION_MODE.WORPeriod = WOR_2000_011; //设置WOR周期
// Set configuration changed and set to not hold the configuration
ResponseStatus rs = e22ttl.setConfiguration(configuration, WRITE_CFG_PWR_DWN_SAVE);//将配置完成的参数写入模块
Serial.println(rs.getResponseDescription()); //打印设置模块参数的结果
Serial.println(rs.code);

//将配置后的参数打印出来
c = e22ttl.getConfiguration();
// It's important get configuration pointer before all other operation
configuration = *(Configuration*) c.data;
Serial.println(c.status.getResponseDescription());
Serial.println(c.status.code);

printParameters(configuration);
}

```

③在Arduino IDE 中的 Tools--> Board根据自己的开发板选择对应型号，然后下载验证程序

#### 4、透传发送

①打开Arduino IDE,

arduino选择file->Examples->Ebyte LoRa E22 library->Arduino\_e22\_SendTransparent,  
 esp32选择file->Examples->Ebyte LoRa E22 library->esp32\_e22\_SendTransparent,  
 esp8266选择file->Examples->Ebyte LoRa E22 library->esp32\_e22\_SendTransparent.

②主程序简要分析

//demo中默认定义RSSI宏定义，如果模块没有配置rssi则需要注释掉，不然会把正常数据的最后一字节作为rssi值



```
#define ENABLE_RSSI true

// ----- Arduino pins -----
/*指定Arduino 的软件串口引脚、AUX、M0、M1的引脚号*/
LoRa_E22 e22ttl(4, 5, 3, 7, 6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1

// ----- esp32 pins -----
//LoRa_E22 e22ttl(&Serial2, 18, 21, 19); // RX AUX M0 M1

// ----- esp8266 pins -----
//LoRa_E22 e22ttl(RX, TX, AUX, M0, M1); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX
LoRa_E22 e22ttl(D3, D4, D5, D7, D6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1
//使用Arduino/esp32/esp8266的时候各自的引脚初始化不同
void setup() {
    Serial.begin(9600);
    delay(500);

    // Startup all pins and UART
    e22ttl.begin();

    // If you have ever change configuration you must restore It
    // ResponseStructContainer c;
    // c = e22ttl.getConfiguration();
    // Configuration configuration = *(Configuration*) c.data;
    // Serial.println(c.status.getResponseDescription());
    // configuration.CHAN = 0x17;
    // configuration.OPTION.fixedTransmission = FT_TRANSPARENT_TRANSMISSION;
    // e22ttl.setConfiguration(configuration, WRITE_CFG_PWR_DWN_SAVE);

    Serial.println("Hi, I'm going to send message!");

    // Send message
    ResponseStatus rs = e22ttl.sendMessage("Hello, world?");
    // Check If there is some problem of successfully send
    Serial.println(rs.getResponseDescription());
}

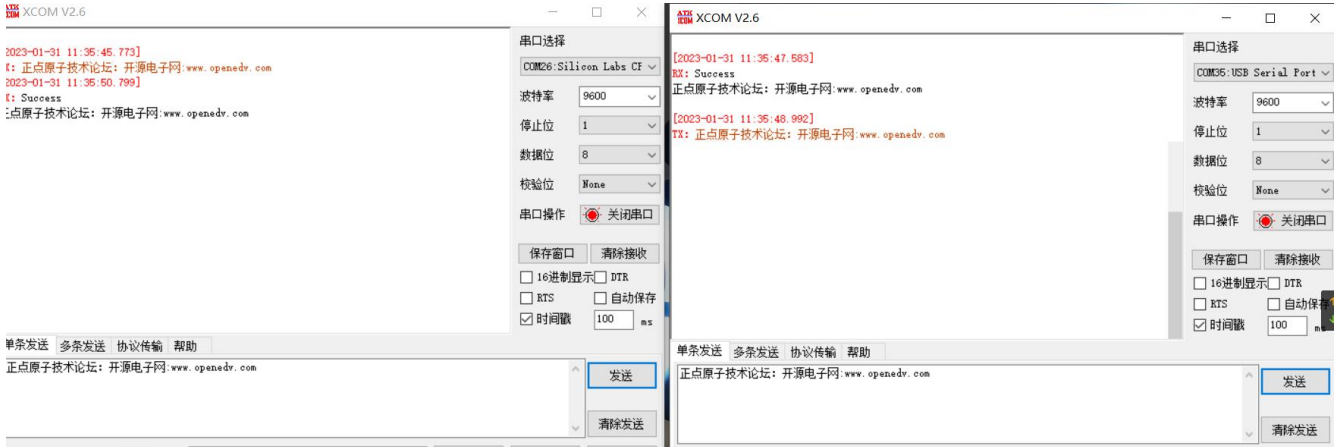
void loop() {
    // If something available
    if (e22ttl.available()>1) {
        // read the String message
        //如果定义RSSI,则把最后一个字节作为rssi强度值
#ifdef ENABLE_RSSI
        ResponseContainer rc = e22ttl.receiveMessageRSSI();
#else
        ResponseContainer rc = e22ttl.receiveMessage();
#endif
        // Is something goes wrong print error
        if (rc.status.code!=1){
            Serial.println(rc.status.getResponseDescription());
        }else{
            // Print the data received
            Serial.println(rc.status.getResponseDescription());
            Serial.println(rc.data);
#ifdef ENABLE_RSSI
            Serial.print("RSSI: "); Serial.println(rc.rssi, DEC);
#endif
        }
    }
    //将串口收到的数据发送出去
    if (Serial.available()) {
        String input = Serial.readString();
    }
}
```

```

    e22ttl.sendMessage(input);
}
}

```

③在Arduino IDE 中的 Tools--> Board根据自己的开发板选择对应型号，然后下载验证程序。效果如下图，两个Arduino透传通信。



## 5、定点发送

①打开Arduino IDE,

arduino选择file->Examples->Ebyte LoRa E22 library->Arduino\_e22\_04\_SendFixedTransmission,  
 esp32选择file->Examples->Ebyte LoRa E22 library->esp32\_e22\_04\_SendFixedTransmission,  
 esp8266选择file->Examples->Ebyte LoRa E22 library->esp8266\_e22\_04\_SendFixedTransmission.

②主程序简要分析

```

// With FIXED RECEIVER configuration
#define DESTINATION_ADDL 2

// ----- Arduino pins -----
/*指定Arduinio 的软件串口引脚、AUX、M0、M1的引脚号*/
LoRa_E22 e22ttl(4, 5, 3, 7, 6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1

// ----- esp32 pins -----
//LoRa_E22 e22ttl(&Serial2, 18, 21, 19); // RX AUX M0 M1

// ----- esp8266 pins -----
//LoRa_E22 e22ttl(RX, TX, AUX, M0, M1); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX
LoRa_E22 e22ttl(D3, D4, D5, D7, D6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1
//使用Arduino/esp32/esp8266的时候各自的引脚初始化不同

void setup() {
    Serial.begin(9600);
    delay(500);

    // Startup all pins and UART
    e22ttl.begin();

    /*获取模块参数配置并打印出来*/
    ResponseStructContainer c;
    c = e22ttl.getConfiguration();
    // It's important get configuration pointer before all other operation
    Configuration configuration = *(Configuration*) c.data;
    Serial.println(c.status.getResponseDescription());
    Serial.println(c.status.code);

    printParameters(configuration);
    c.close();
}

```

```

Serial.println("Hi, I'm going to send message!");
//以定点方式发送数据, 指定地址和信道 (模块需要先配置为定点模式)
// Send message
ResponseStatus rs = e22ttl.sendFixedMessage(0, DESTINATION_ADDL, 23, "Hello, world?");
// Check If there is some problem of succesfully send
Serial.println(rs.getResponseDescription());
}

/*循环读取接收数据buffer, 把接收到的数据打印出来*/
void loop() {
    // If something available
    if (e22ttl.available()>1) {
        // read the String message
#ifdef ENABLE_RSSI
        ResponseContainer rc = e22ttl.receiveMessageRSSI();
#else
        ResponseContainer rc = e22ttl.receiveMessage();
#endif
        // Is something goes wrong print error
        if (rc.status.code!=1){
            Serial.println(rc.status.getResponseDescription());
        }else{
            // Print the data received
            Serial.println(rc.status.getResponseDescription());
            Serial.println(rc.data);
#ifdef ENABLE_RSSI
            Serial.print("RSSI: "); Serial.println(rc.rssi, DEC);
#endif
        }
    }
    if (Serial.available()) {
        String input = Serial.readString();
        e22ttl.sendFixedMessage(0, DESTINATION_ADDL, 23, input);
    }
}

```

③在Arduino IDE 中的 Tools-->Board根据自己的开发板选择对应型号, 然后下载验证程序

6、剩余示例按照同样的方式使用。

## 第三步 E220例程使用

### 1、连接方式

E220	ESP32	Raspberrt Pi Pico	Stm32	Arduino UNO
M0	19	10	PB0	7
M1	21	11	PB10	6
RX	TX2	8	PA2 TX2	4
TX	RX2	9	PA3 RX2	5
AUX	18	2	2	3
VCC	3.3/5V	3.3/5V	3.3/5V	3.3/5V
GND	GND	GND	GND	GND

### 2、读取模块参数

①打开Arduino IDE, 选择file->Examples->Ebyte LoRa E220 library->01\_getConfiguration.

②主程序简要分析

```

// ----- Arduino pins -----
/*指定Arduinio 的软件串口引脚、AUX、M0、M1的引脚号*/
LoRa_E220 e220ttl(4, 5, 3, 7, 6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1

```

```

// ----- esp32 pins -----
//LoRa_E220 e220ttl(&Serial2, 18, 21, 19); // RX AUX M0 M1

// ----- esp8266 pins -----
//LoRa_E220 e220ttl(RX, TX, AUX, M0, M1); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX
//LoRa_E220 e220ttl(D3, D4, D5, D7, D6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1

// ----- Raspberry PI Pico pins -----
// LoRa_E220 e220ttl(&Serial2, 2, 10, 11); // RX AUX M0 M1
// -----

// ----- STM32 -----
//HardwareSerial Serial2(USART2); // PA3 (RX) PA2 (TX)
//LoRa_E220 e220ttl(&Serial2, PA0, PB0, PB10); // RX AUX M0 M1
// -----

//使用Arduino/esp32/esp8266/Raspberry PI pico/STM32的时候各自的引脚初始化不同
void setup() {
    Serial.begin(9600);
    while(!Serial){};
    delay(500);

    Serial.println();

    // Startup all pins and UART
    e220ttl.begin();

    ResponseStructContainer c;
    c = e220ttl.getConfiguration(); //获取E220模块的配置参数，该函数内部实现了工作模式切换
    // It's important get configuration pointer before all other operation
    Configuration configuration = *(Configuration*) c.data;
    Serial.println(c.status.getResponseDescription());
    Serial.println(c.status.code);

    printParameters(configuration); //打印模块的配置参数

    /*获取打印模块产品信息*/
    ResponseStructContainer cMi;
    cMi = e220ttl.getModuleInformation();
    // It's important get information pointer before all other operation
    ModuleInformation mi = *(ModuleInformation*)cMi.data;

    Serial.println(cMi.status.getResponseDescription());
    Serial.println(cMi.status.code);

    printModuleInformation(mi);
}
    
```

③在Arduino IDE 中的 Tools→Board根据自己的开发板选择对应型号，然后下载验证程序

3、E22/E32/E220 的驱动程序几乎一致，剩余示例按照同样的方式使用。

## 第四步 E32例程使用

### 1、连接方式

E32	ESP32	ESP8266	Arduino UNO
M0	19	D7	7
M1	21	D6	6
RX	TX2	D3	4

TX	RX2	D4	5
AUX	18	D5	3
VCC	3.3/5V	3.3/5V	3.3/5V
GND	GND	GND	GND

2、以配置模块参数为例，简单使用，其余例程按照同样方法使用

①打开Arduino IDE，选择file->Examples->Ebyte LoRa E220 library->01\_setConfiguration。

②主程序简要分析

```
//该例程中没有连接m0,m1直接将其接到vcc处于配置模式
//LoRa_E32 e32ttl100(2, 3); // e32 TX e32 RX
//实际使用需要接上m0,m1
//LoRa_E32 e32ttl100((4, 5, 3, 7, 6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1
)
// ----- esp32 pins -----
//LoRa_E32 e32ttl100(&Serial2, 18, 21, 19); // RX AUX M0 M1

// ----- esp8266 pins -----
//LoRa_E32 e32ttl100(RX, TX, AUX, M0, M1); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX
LoRa_E32 e32ttl100(D3, D4, D5, D7, D6); // Arduino RX <-- e22 TX, Arduino TX --> e22 RX AUX M0 M1

void printParameters(struct Configuration configuration);
void printModuleInformation(struct ModuleInformation moduleInformation);

void setup() {
    Serial.begin(9600);
    delay(500);

    // Startup all pins and UART
    e32ttl100.begin();

    ResponseStructContainer c;
    c = e32ttl100.getConfiguration();//获取配置参数
    // It's important get configuration pointer before all other operation
    Configuration configuration = *(Configuration*) c.data;
    Serial.println(c.status.getResponseDescription());
    Serial.println(c.status.code);

    printParameters(configuration);//打印配置参数
    //模块地址
    configuration.ADDL = 0x0;
    configuration.ADDH = 0x1;
    //信道
    configuration.CHAN = 0x19;

    configuration.OPTION.fec = FEC_0_OFF; //是否开启前向纠错
    configuration.OPTION.fixedTransmission = FT_TRANSPARENT_TRANSMISSION;//配置是否定点
    configuration.OPTION.ioDriveMode = IO_D_MODE_PUSH_PULLS_PULL_UPS;//IO驱动方式
    configuration.OPTION.transmissionPower = POWER_17;//功率，空速
    configuration.OPTION.wirelessWakeupTime = WAKE_UP_1250;//WOR周期

    configuration.SPED.airDataRate = AIR_DATA_RATE_011_48; //空速
    configuration.SPED.uartBaudRate = UART_BPS_115200; //波特率
    configuration.SPED.uartParity = MODE_00_8N1;

    // Set configuration changed and set to not hold the configuration
    ResponseStatus rs = e32ttl100.setConfiguration(configuration, WRITE_CFG_PWR_DWN_LOSE);
    Serial.println(rs.getResponseDescription());
    Serial.println(rs.code);
    printParameters(configuration);//把配置后的参数打印出来
    c.close();
}
```

}

③在Arduino IDE 中的 Tools--> Board根据自己的开发板选择对应型号，然后下载验证程序

3、其余例程也按照该方式使用。

## 修订历史

版本	修订日期	修订说明	维护人
1.0	2023-2-10	初始版本	Luo

## 关于我们



销售热线：4000-330-990      公司电话：028-61399028  
 技术支持：[support@cdebyte.com](mailto:support@cdebyte.com)      官方网站：[www.ebyte.com](http://www.ebyte.com)  
 公司地址：四川省成都市高新西区西区大道 199 号 B5 栋

