亿享云网关模组软件

设计说明书

V1.0

**版本信息**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 时间 | 修改内容 | 修改者 |
| V1.0 | 2019-02-15 | 初始版本 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1. 工程配置 4](#_Toc9974402)

[1．1 配置文件 4](#_Toc9974403)

[1．2 添加宏定义 4](#_Toc9974404)

[1．3 导入用户代码和路径 5](#_Toc9974405)

[2. 工程代码详解 5](#_Toc9974406)

[2．1 初始化 6](#_Toc9974407)

[2．2 rf-rx-uart-tx-task.c 7](#_Toc9974408)

[2．3 rf-tx-command.c 7](#_Toc9974409)

[2．4 serial-rx-queue.c 8](#_Toc9974410)

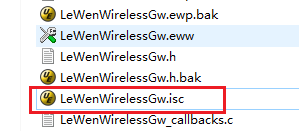
[2．5 serial-rx-task.c 8](#_Toc9974411)

[2．6 回调函数 9](#_Toc9974412)

# 工程配置

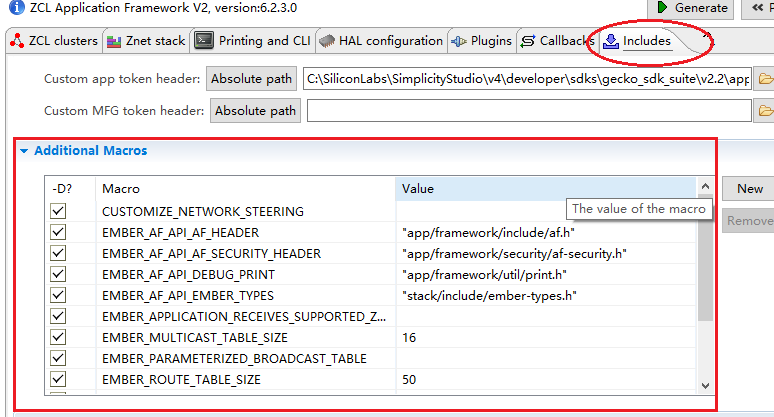
## 1．1 配置文件

配置文件LeWenWirelessGw.isc，在工程目录下。



## 1．2 添加宏定义

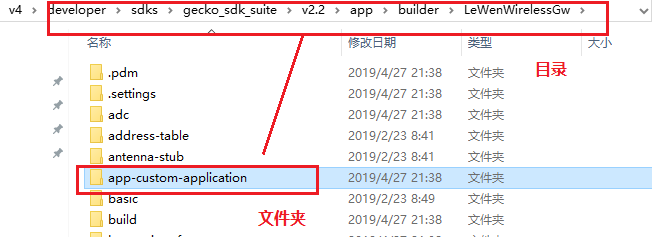
在isc的配置页面include选项卡中几个重要参数配置和宏定义：



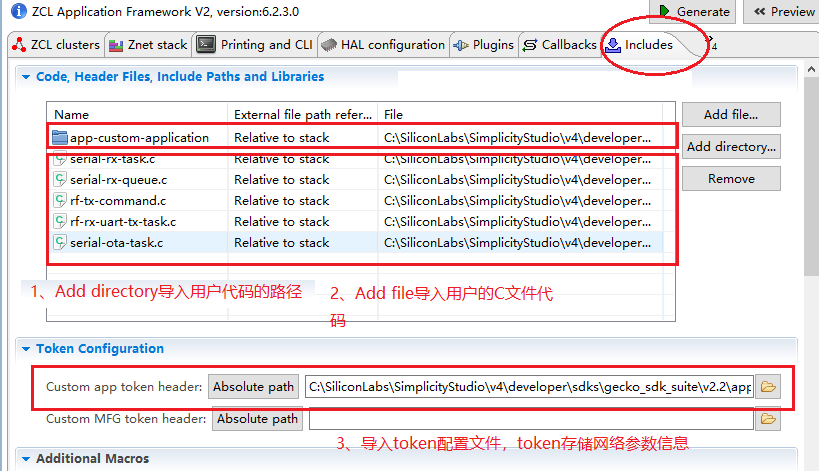
1. 添加宏：CUSTOMIZE\_NETWORK\_STEERING
2. 添加宏：EMBER\_APPLICATION\_RECEIVES\_SUPPORTED\_ZDO\_REQUESTS
3. 添加宏：EMBER\_MULTICAST\_TABLE\_SIZE
4. 添加宏：EMBER\_ROUTE\_TABLE\_SIZE
5. 添加宏：NO\_LED
6. 添加宏：SOC\_OTA\_SERVER

## 1．3 导入用户代码和路径

用户代码目录以及文件夹如下：

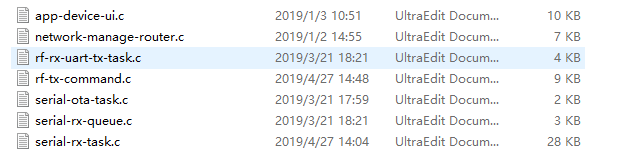


在isc的配置页面include选项卡导入用户代码，路径以及token配置文件，如下图：



# 工程代码详解

用户代码：



以上的几个C文件，再加上：LeWenWirelessGw\_callbacks.c

代码中都有注释说明，帮助快速理解代码。

代码模块功能说明如下：

|  |  |
| --- | --- |
| 文件 | 功能说明 |
| app-device-ui.c | LED的指示，目前暂时用不上，可以不管。 |
| network-manage-router.c | 设备入网的功能代码，目前通过串口配网，可以不看（预留用做按键入网）。 |
| rf-rx-uart-tx-task.c | 网关接收来自设备发送的RF数据，然后通过串口输出到网关端的主控 |
| rf-tx-command.c | 接收串口命令通过RF发送到设备 |
| serial-ota-task.c | 用做网关主控通过传输设备端的升级镜像文件，然后写入到网关zigbee模组，然后给设备升级（预留功能） |
| serial-rx-queue.c | 网关zigbee模组串口环形队列接收。 |
| serial-rx-task.c | 网关zigbee模组串口接收任务处理。 |

功能模块划分：

1. 串口接收

串口接收分两部分，配网和应用数据传输两部分

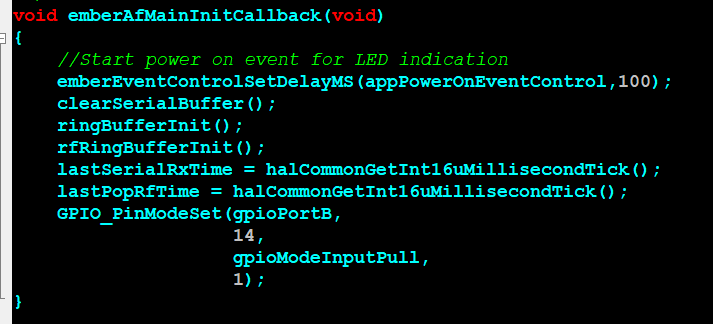
1. 配网命令直接处理不进入队列中。
2. 应用数据传输通过环形队列方式接收入队列，然后在主循环里面从队列里面取数据，然后转化成无线数据包。
3. RF接收

RF接收也存在另外的一个环形队列中，然后从队列中取出通过串口输出到主控。

## 2．1 初始化

在LeWenWirelessGw\_callbacks.c中：

void emberAfMainInitCallback(void)



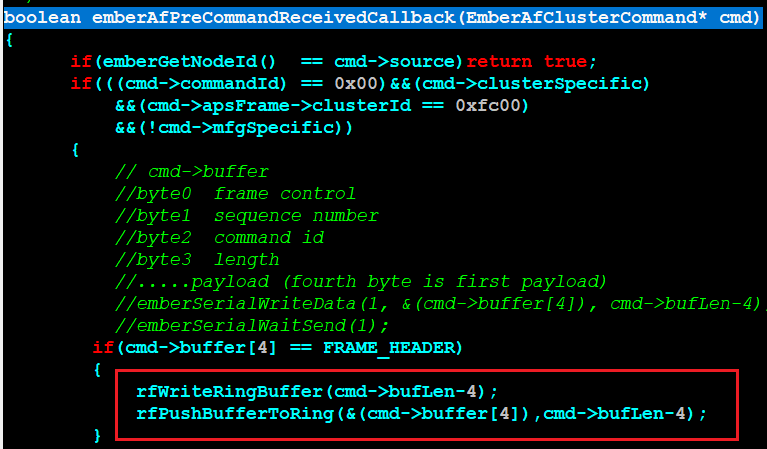
## 2．2 rf-rx-uart-tx-task.c

1、rf接收入口

在LeWenWirelessGw\_callbacks.c中：

boolean emberAfPreCommandReceivedCallback(EmberAfClusterCommand\* cmd)

接收到数据后存在rf中的环形队列中，

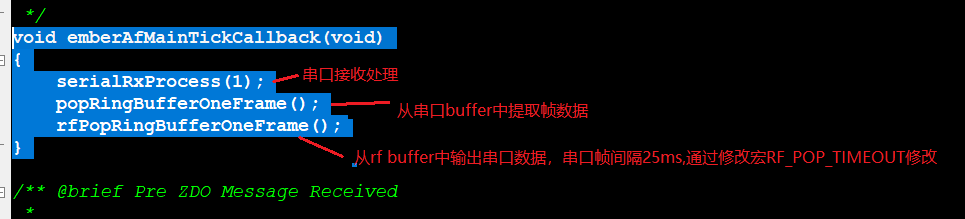


2、从rf的buffer中输出到串口

void rfPopRingBufferOneFrame(void)

在LeWenWirelessGw\_callbacks.c中：

void emberAfMainTickCallback(void)这个回调函数中调用，此回调函数会在while循环体中被调用。



## 2．3 rf-tx-command.c

1、无线发送打包函数

uint8\_t rfCommandSetup(uint16\_t dstAddr,uint8\_t sendMode,

uint8\_t \*buffer,uint8\_t bufferlen)

2、各种命令封包

封包的数据从串口中解码出来，重新封装成RF包。

void rfSendCommandQueryControlPointValue(void) //查询控制点

void rfSendCommandCtrlControlPoint(void) //操控控制点

void rfSendCommandUpdatePannel(void) //同步面板

void rfSendCommandDirectTx(void) //数据透传命令

## 2．4 serial-rx-queue.c

串口接收队列处理：

入队列：uint8\_t pushBufferToRing(uint8\_t \*buffer,uint8\_t length)

出队列：uint8\_t popRingBufferData(uint8\_t \*buffer,uint8\_t length)

## 2．5 serial-rx-task.c

在入队列前会做合法性判断，串口接收以20ms超时作为结束判断，20ms内没有收到数据认为一帧结束。

对应的宏定义

**#define SERIAL\_RX\_TIMEOUT 20**

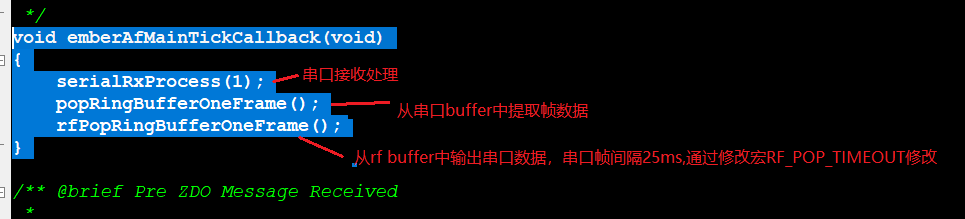
串口接收有两种协议

1. 配置网络协议
2. 数据传输协议

1、串口接收入口：void serialRxProcess(uint8\_t Port)

在LeWenWirelessGw\_callbacks.c中：

void emberAfMainTickCallback(void)这个回调函数中调用，此回调函数会在while循环体中被调用。



1. 串口命令解码

接口void serialCommandParse(void)

里面区分两种协议通过帧头A5 AA 5A区分

1. 数据传输协议的数据会通过串口环形队列接收缓存

void customDefineProtocalProcess(void)

串口队列入队列：pushBufferToRing

出队列：popRingBufferOneFrame();

队列中取出一帧后，通过RF发送出去到设备端。

在LeWenWirelessGw\_callbacks.c中：

void emberAfMainTickCallback(void)这个回调函数中调用，此回调函数会在while循环体中被调用。

取队列的间隔时间

**#define RF\_TX\_TIMEOUT 100**

## 2．6 回调函数

回调函数集中在LeWenWirelessGw\_callbacks.c中

几个重要回调函数：

1. 初始化的回调函数：void emberAfMainInitCallback(void)
2. 循环回调函数：void emberAfMainTickCallback(void)
3. RF接收的回调函数：

boolean emberAfPreCommandReceivedCallback(EmberAfClusterCommand\* cmd)