

**双向RF通信协议**

**版本号V003R001**

**概要设计文档**

**深圳市银河风云网络系统股份有限公司**

**Shenzhen GALAXYWIND Network Systems Co., Ltd.**

地址：深圳市高新技术产业园区北区新西路5号银河风云大厦

Address: GALAXYWIND building, No.5 Xinxi road, Shenzhen High-Tech Industry Park, NanShan district, China

邮编 P.C.: 518055

电话 Tel: +86-755-83400088

传真 FAX: +86-755-86139063

网站 Web: http://[www.galaxywind.com](http://www.galaxywind.com)

客服 Hotline: 4000009879

企业QQ: 4000009879

微信公众号 WeChat:



**版权声明**

版权所有©深圳市银河风云网络系统股份有限公司2015。深圳市银河风云网络系统股份有限公司（以下简称"银河风云"）对本资料进行版权声明，未经银河风云书面许可，任何单位及个人不得以任何方式或理由对本资料的任何内容进行复制、修改、抄录、传播。

**商标声明**

"银河风云"，"GALAXYWIND"等是深圳市风云实业有限公司的注册商标，本文档提及的其他商标、服务标志、注册商标及注册服务标志均为其所有者拥有。

**免责声明**

本资料会进行不定期更新，本资料中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

[1 Ssv7241介绍 4](#_Toc416443206)

[1.1 总体介绍 4](#_Toc416443207)

[1.2 特性 5](#_Toc416443208)

[1.3 控制状态 6](#_Toc416443209)

[1.4 帧格式 6](#_Toc416443210)

[1.5 报文处理 7](#_Toc416443211)

[1.6 RSSI记录 8](#_Toc416443212)

[1.7 测试数据 8](#_Toc416443213)

[1.7.1 功耗测试 8](#_Toc416443214)

[1.7.2 丢包率测试 9](#_Toc416443215)

[1.7.3 醒来收包时间测试 10](#_Toc416443216)

[2 组网需求 11](#_Toc416443217)

[3 几种方案比较 11](#_Toc416443218)

[3.1 方案1（持续发送） 11](#_Toc416443219)

[3.2 方案2（poll） 12](#_Toc416443220)

[3.3 方案3（划分时隙） 12](#_Toc416443221)

[3.4 对比 12](#_Toc416443222)

[4 划分时隙协议设计 12](#_Toc416443223)

[4.1 网络拓扑 12](#_Toc416443224)

[4.2 稳定状态 13](#_Toc416443225)

[4.2.1 周期广播beacon帧和收发同步 13](#_Toc416443226)

[4.2.2 时间划分 14](#_Toc416443227)

[4.2.3 升级模式 14](#_Toc416443228)

[4.2.4 保活 15](#_Toc416443229)

[4.2.5 无线待机功耗计算 15](#_Toc416443230)

[4.2.5.1 200ms周期 15](#_Toc416443231)

[4.2.5.2 400ms周期 16](#_Toc416443232)

[4.2.5.3 800ms周期 16](#_Toc416443233)

[4.3 协商阶段 16](#_Toc416443234)

[4.3.1 网络地址约定 16](#_Toc416443235)

[4.3.2 网关初始化 17](#_Toc416443236)

[4.3.3 一键配对 17](#_Toc416443237)

[4.3.4 连接到网关 17](#_Toc416443238)

[4.3.5 漫游和安全性 17](#_Toc416443239)

[4.3.6 重新配置 18](#_Toc416443240)

前言

**概述**

基于2.4G实现的通信系统的协议设计。

**产品版本**

与本文档相对应的产品版本如下所示：

|  |  |
| --- | --- |
| **产品名称** | **产品版本** |
| 双向RF通信协议概要设计 | 0.0.3 |

**阅读对象**

本文档主要适用于以下用户：

结构设计工程师【√】

硬件开发工程师【√】

软件开发工程师【√】

系统规划工程师【√】

系统维护工程师【√】

**约定**

**符号约定**

文档中可能出现的符号定义如下所示：

|  |  |
| --- | --- |
| **符号** | **说明** |
| 警告警告 | 表示有中度或低度潜在危险，如果不能避免，可能导致较严重结果。 |
| 注意注意 | 表示有潜在风险，如果忽视这些文本，可能导致不可预知的结果。 |
| 备注备注 | 表示对正文的强调和补充。 |

**通用格式约定**

|  |  |
| --- | --- |
| **格式** | **说明** |
| 黑体 | 中文：1 -6级标题，图表/表格标题采用黑体 |
| Arial | 英文：1 -6级标题，图表/表格标题采用Arial |
| 宋体 | 中文：正文采用宋体 |
| Times New Roman | 英文：正文采用Times New Roman |

**修改记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **修订时间** | **修订人** | **审核人** | **修订版本** |
| 2015-3-26 | 谭洪国 |  | 0.0.2 |
| 2015-4-10 | 谭洪国 |  | 0.0.3 |

# Ssv7241介绍

## 总体介绍

SSV7241是一个单片GFSK收发器工作在ISM频段2400~2483.5MHz。外部低端MCU通过SPI接口对其进行配置。组包过程，自动重发，自动ACK都整合到无线引擎中，减少MCU负担。



## 特性

无线

2.4GHz ISM 频段，126频道，间隔1MHz

GFSK调制

250kbps, 1Mbps, and 2Mbps air data rate

发送

支持连续无间隔发送

接收

-92 dBm sensitivity at 250kbps

-90 dBm sensitivity at 1Mbps

-88 dBm sensitivity at 2Mbps

Advanced RSSI for channel monitoring

电源管理

Integrated voltage regulators for digital and analog/RF blocks

1.9V to 3.6V supply range

9uA power-down current, 47uA standby current

包管理

收发两个FIFO。发送FIFO保存3个发送包，接收FIFO保存3个接收包。每个包最大32字节。

支持变长payload

自动装配，反装配包。

自动ACK，重传

支持6个发送者和1个接收者组成的星形网络。

## 控制状态

一个SSV7241同一时间要么配置为发送者（PTX），要么配置为接收者（PRX）。



上电后，SSV7241处于power down状态，MCU通过SPI设置寄存器PWR\_ON位段进入standby状态。如果RX\_ON位段设置为0，则是进入TX状态（PTX），否则进入RX状态（PRX）。

引脚CE由MCU控制，使能发送或接收，模块开始发送FIFO的数据 或 接收数据送往FIFO。

当设置了自动ACK，则PTX发送一个报文后进入RX模式接收ACK；PRX接收一个报文后进入TX模式发送ACK。

## 帧格式



Adress：目的地址5bytes。

PID：判断报文是不是重传的。

NO\_ACK：是否需要ACK。

报文中没有包含源地址，所以收到报文后，不知道是谁发送的。

接收方可以设置6个接收地址，一个地址对应一个PIPE。其中只有P0和P1的地址是5bytes都可以任意设置，P2~P5的地址只能设置最后一个byte，前面四个bytes和P1是一样的。地址中包含全0，全1，或和前导码相似的字节01010101,10101010，会使得错误率上升。

接收方能得到payload内容和是从哪个PIPE收到的数据，以及信号强度。

## 报文处理

在发送状态，PHY引擎从TX FIFO中取得payload，装配到报文中，发送。发送完成，如果报文设置了NO\_ACK，就设置TX\_DS为1，并使能到MCU的IRQ引脚。如果报文需要ACK，则等ACK收到后才设置TX\_DS和IRQ。

接收者一直侦听无线信号，如果收到报文，就检验CRC和目的地址，并根据报文是否需要ACK决定什么时候设置RX\_DR和IRQ。

发送者根据以下寄存器（位段）决定重发行为。

ARD：ACK最大允许延迟。

ARC：最大重试次数。

如果重试次数用完，则设置MAX\_RT和IRQ。

ARC\_CNT和PLOS\_CNT分别记录本次传输重试次数和信道丢包数，可作为信道质量评估。

ACK可带数据，PRX的发送FIFO用来存放发往特定发送者的ACK携带的数据，而PTX的接收FIFO用来接收ACK携带的数据。此机制可用来实现PRX向PTX发送数据，但前提是PTX要定期向PRX发送数据。如果PTX的接收FIFO满，而收到的ACK带数据，则PTX会重发。PRX的发送FIFO中的ACK payload是指定了PTX的，如果此PTX一直未发数据，该发送FIFO单元就被占用，必要时需要用FLUSH\_TX清除。

## RSSI记录



SSV7241每隔128us检测一次信号强度，得到如下的RSSI：

RSSI1（1bit），RSSI2（1bit），当前RSSI指示，根据两个门限比较所得。

RSSIREC1，RSSIREC2，历史8bit记录。

## 测试数据

### 功耗测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **测试名称：SSV7241功耗测试 软件配置：输出功率：6dBm(4mw)，发送速率：250Kbps，启用AA模式，77信道（2477Mhz），分别设置寄存器0x00的PWR\_ON，RX\_ON字段进行功率测量。 测试方法：将功率测量电路串接在被测组模的电源输入端（VCC=3.3V） 发包速率：间隔2ms发包，每包数据32字节(无线帧总长329bit)，发包速率：128.52Kbps 测试地点：E11栋3楼办公室 测试时间：2015.3.23下午** | | | | |
|
|
|
|
|
|
|  |  | **平均功率(mw)，2分钟/次** | |  |
| **序号** | **工作模式** | **测试1** | **测试2** | **备注** |
| 1 | Standby Mode | 0.514 | 0.541 | 电流档测试：51.9uA |
| 2 | Rx Mode(不收包，接收准备好) | 93.067 | 93.340 | 电流档测试：29.3mA |
| 3 | Rx Mode（收包，128Kbps） | 87.641 | 87.564 | 电流档测试：27.6mA |
| 4 | Tx Mode(不发包，发包准备好) | 1.380 | 1.345 | 电流档测试：51.9uA |
| 5 | Tx Mode(发包，128Kbps) | 74.598 | 74.718 | 电流档测试：23.5mA |

### 丢包率测试

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **测试名称：SSL7241无线收发包室外无障碍物测试1（W1模组天线） 软件配置：输出功率：6dBm(4mw)，发送速率：250Kbps，启用AA模式，77信道（2477Mhz） 测试方法：间隔1秒发100个包，共60秒，每包数据32字节(无线帧总长329bit) 测试地点：E11到E13那条马路上 测试日期：2015.3.19下午** | | | | | | |
|
|
|
|
|
|  |  | **收包数** | | |  |  |
| **序号** | **距离(米)** | **测试1** | **测试2** | **测试3** | **测试4** | **平均丢包率** |
| 1 | 10 | 100 | 100 | 100 | N/A | 0 |
| 2 | 20 | 93 | 85 | 83 | N/A | 0.13 |
| 3 | 30 | 59 | 21 | 35 | 71 | 0.535 |
| 4 | 50 | 20 | 18 | 23 | 77 | 0.655 |
| 5 | 80 | 1 | 7 | 42 | 29 | 0.8025 |
| 6 | 100 | 9 | 12 | 42 | 53 | 0.71 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **测试名称：SSL7241无线收发包室内穿墙测试（BOX全向天线） 软件配置：输出功率：6dBm(4mw)，发送速率：250Kbps，启用AA模式，77信道（2477Mhz） 测试方法：间隔2ms发10000个包，共25秒，每包数据32字节(无线帧总长329bit)，  发包速率：128.52Kbps 测试地点：家里（80㎡套二） 测试日期：2015.3.2晚上** | | | | | | | |
|
|
|
|
|
|
|  |  | **收包数** | | | | |  |
| **序号** | **距离** | **测试1** | **测试2** | **测试3** | **测试4** | **平均丢包率(Kbps)** | **平均接收速率(Kbps)** |
| 1 | 一堵墙 | 9213 | 9992 | 9995 | 9990 | 0.0202 | 125.91 |
| 2 | 二堵墙 | 8954 | 8921 | 8733 | 8966 | 0.1106 | 114.29 |
| 3 | 二堵墙 | 8415 | 8490 | 8768 | 8345 | 0.1495 | 109.29 |

### 醒来收包时间测试

发送端：非AA模式，250Kbps，以4.5ms间隔，连续发送3个32字节的包：

接收端：睡眠1秒，收包9ms，每次能收到3-5个包,4个包居多

接收端：睡眠1秒，收包5ms，每次能收到2-3个包,2个包居多

发送端：非AA模式，250Kbps，以1.5ms间隔，连续发送3个1字节的包

接收端：睡眠1秒，收包3ms，每次能收到4-6个包

接收端：睡眠1秒，收包2ms，每次能收到3-5个包

接收端：睡眠1秒，收包1.5ms，每次能收到2-4个包，2个居多

接收端：睡眠1秒，收包0.5ms，偶尔能收到一个包。

# 组网需求

单网关下带0~128个设备由低功耗2.4G无线通信组成星形网络，设备需要支持电池供电低功耗。网关和AP通信。

多网关组成网关集合，网关之间通过802.11协议组成mesh网络，一个或多个网关和AP通信。设备可以在多网关漫游，选择最优网关。

本文只讨论单网关情形，只讨论低功耗2.4G无线通信协议部分。偶尔提到多网关支持。

协议设计目的：

* 尽可能省电，需减少设备收发包时间。
* 最多支持128个设备。
* 网内的包尽可能不碰撞。
* 让设备迅速响应。
* 容错性，错误和误差恢复。
* 传输可靠性，失败检测。

# 几种方案比较

为了使得设备端能够实现电池供电，设备端无线一直处于接收状态是不现实的（见1.7.1功耗测试）。设备端只能周期性的醒来和网关交互很小一段时间，接收网关发送给设备的消息。设备端要能反应及时，周期不能太大。每个周期交互的时间，直接决定了设备耗电量。

当网络中设备量很大，或者有较多设备和网关通信的时候，需要避免碰撞的机制。

## 方案1（持续发送）

方案描述：设备每个周期醒来2ms接收网关beacon帧，网关平时不发数据，有消息需要发送给设备的时候，连续发送大于一个周期的beacon帧给设备端以使得设备接收命令报文。

优点：平时网络很安静，网关和设备不需同步。

缺点：连续发送占用信道，设备醒来的时间长，耗电多，设备收包失败可能性加大，不能持续恢复恢复错误。不能支持大规模网络。

## 方案2（poll）

方案描述：设备每个周期向网关询问是否有自己的消息，然后接收消息。

优点：不用长时间占用信道，不需同步。

缺点：当设备多的时候，网络拥塞严重，碰撞率增大。不适合大规模组网。

## 方案3（划分时隙）

方案描述：网关每个周期向子网广播beacon帧，报文中带时间戳和有报文需要接收的设备列表。设备根据报文时间戳决定下次醒来接收beacon帧时间，根据报文设备列表决定是否接收网关命令报文。

优点：设备端醒来时间最短，最省电。由于网关和子网中所有设备端周期时间是同步的，可以划分设备和网关收发时隙，避免碰撞，这在多设备和网络繁忙的时候，能体现出来。

缺点：需要网关和设备端精确对时，每次发送和接收时间被安排，设计和实现较复杂。

## 对比

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 网络规模 | 信道拥塞情况 | 信道利用率 | 每周期电量消耗 |
| 持续发送 | 小 | 严重 | 低 | 2ms RX |
| Poll | 中 | 一般 | 中 | 0.4msTX，1ms RX |
| 划分时隙 | 大 | 不拥塞 | 高 | 0.5ms RX |

# 划分时隙协议设计

## 网络拓扑

udp

服务器

手机

udp udp

我家网关2

我家网关1

2.4G 2.4G 2.4G

设备3

设备2

设备1

2.4G 2.4G 2.4G

暂定只有网关才是向服务器注册的实体，设备看做网关的内部功能，减少设备和网关的交互报文类型，便于讨论。

## 稳定状态

稳定状态下网关和网关相连的0~128个设备组成一个星型网络，使用同一个无线信道通信。

网关周期性地广播Beacon报文：

时间标示：用于全网时间同步

待处理消息列表：告知设备时候有需要处理的消息通知

设备周期性醒来接收Beacon，校准时间，并决定是否继续接收命令消息

ACK由软件控制，不使用硬件提供的ack机制。

### 周期广播beacon帧和收发同步

网关每200ms向自己的子网广播3个连续的beacon帧，报文包含：

时间偏移量：1字节（只用7bit），200ms周期内的时间偏移量，单位为16us。

Beacon序号：1字节，0~127循环。

唤醒设备列表：可能有0~10字节，列举有消息需要发送的设备ID。一次最多发送10个。

* 设备以200ms的整数倍时间醒来一次接收beacon帧。只要收到一个beacon帧，马上计算出下次醒来的时间，然后决定是否睡眠。

为了使设备醒来的时间短，设备需要在接近网关每次发送beacon帧前进入RX模式，然后收到一个包就进入睡眠模式。这需要设备端和网关精确对时，初步设计目标在多数时间误差只有几十微秒。为了达到这个要求，网关使用中断触发报文发送，使用硬件定时器计时，这需要两个硬件定时器。过程描述如下：

定时器1初始化为周期200ms循环，以后不会重新设置，只会读取。

定时器0用来设定在定时器1归零前中断，中断处理函数中循环检测到定时器1处在归零后2ms内，发送beacon帧。然后定时器0继续设定在定时器1归零前中断。

网关发送beacon帧的时间偶尔可能被禁止中断延后，应该不会提前。延后报文中携带的时间仍然是精确的。

实验测得网关每200ms发送定长beacon报文，设备端和网关对时准确后，每次在网关发送前1ms进入RX模式，结果每次从进入RX到收到报文的时间都在1ms左右，并且误差最大2us。计算得进入RX模式的最小时间为： 120us+350us=470us，其中120us为standby到RX的转换时间，350us为最小报文传输时间。既然重复精度这么高，并且网关是绝对不会提前发送的，所以没必要提前那么多时间进入RX模式，可逐步延后进入RX的时间，直到进入RX的实际时间大于理论时间50us左右。

设备端和网关对时准确后，第一次在网关发送前1ms进入RX模式，最大接收2ms。设备收到包后，根据包的长度计算收包理论需要时间，和从进入RX到收到包的时间差。如果差值大于50us，则适当调整下次进入RX的时间，直到时间差小于50us。如果差值小于50us，则适当调整下次进入RX的时间，直到时间差大于50us。最后时间差在50us上下徘徊。然后检查beacon报文。如果发现有报文中活动设备列表包含自己的ID，就不进入睡眠模式，继续接收命令消息；如果不包含，就进入睡眠模式，按自己设定的周期醒来。

如果允许设备最大醒来时间为1秒，只要晶振的精度在万分之一（1秒钟误差100us），对我们的设计无影响。

### 时间划分

网关发送beacon帧的周期是200ms，由于设备和网关已经精确对时，所以200ms内可以划分时隙，规定什么时候谁发谁接收，尽量避免碰撞。

200ms时间划分为4部分，分别用A，B，C，D表示：

A B C D

5ms 5ms 5ms … … 5ms 5ms 5ms … … 5ms

0 200ms

A阶段：网关发送beacon帧，时间长度5ms。

B阶段：网关发送设备命令消息，发送完为止。可能有0到16个设备的消息，每个消息占5ms。时间长度0~50ms。在5ms内可以发送3个32字节的报文。

C阶段：设备发送ACK，和B阶段的消息一一对应，每个消息占5ms。时间长度0~50ms。

D阶段：设备主动向网关发送消息，所有设备随机发送。大多数情况下，网关不会立即应答而在下个周期应答，只有特殊情况，例如设备发送probe发现网关，需要网关立即应答。时间长度95ms~195ms。

需要可靠传输的报文都会在本周期连续传送3次，如果未收到对方回应，就在下个周期继续相同的交互，直到发现对方离线。

### 保活

网关和设备都能探测到对方不在线。离线的设备，网关会回收ID和其他资源。

### 无线待机功耗计算

从深睡到standby再到RX，最后恢复深睡，耗电曲线如下图：

A B C D E F

A，F：深睡，电流9uA，时间根据周期长短决定。

B：深睡到standby转换时间。电流3mA，时间4ms。

C：standby，电流47uA，时间不定，可以很短。

D：standby转换为RX转换时间，电流7mA，时间0.12ms。

E：RX状态，电流27mA，只需0.35ms能收到通知包。加上提前量，计0.5ms。

#### 200ms周期

200ms周期电流量（深睡）：

27mA\*0.5ms + 7mA\*0.12ms + 3mA\*4ms + 0.009mA\*（200-4）ms = 28mA\*ms

200ms周期电流量（不深睡）：

27mA\*0.5ms + 7mA\*0.12ms + 0.047mA\*200ms = 24mA\*ms

可以看出，200ms的设备，不需要进入深睡模式耗电更低。

总结：不深睡平均电流24mA\*ms /200ms = 0.12mA，一年1051mAh。

#### 400ms周期

400ms周期电流量（深睡）：

27mA\*0.5ms + 7mA\*0.12ms + 3mA\*4ms + 0.009mA\*（400-4）ms = 30mA\*ms

400ms周期电流量（不深睡）：

27mA\*0.5ms + 7mA\*0.12ms + 0.047mA\*400ms = 33mA\*ms

可以看出，400ms的设备，进入深睡模式耗电更低。

总结：深睡平均电流30mA\*ms /400ms = 0.075mA，一年657mAh。

#### 800ms周期

800ms周期电流量（深睡）：

27mA\*0.5ms + 7mA\*0.12ms + 3mA\*4ms + 0.009mA\*（800-4）ms = 32mA\*ms

总结：深睡平均电流32mA\*ms /800ms = 0.04mA，一年350mAh。

## 协商阶段

### 网络地址约定

设备ID：1字节，网关分配的设备的ID，网关的ID为0xfd。

网络ID：4字节，无线通信地址的前4字节使用网络ID，由网关指定。

单播地址：网络ID+设备ID。存储于P1。设备ID范围0~0xef。

网内广播：网络ID+0xfb。存储于P2。

网内组播：网络ID+组播ID，存储于P3~P5。组播ID范围规定0xfe。

全网广播：0xE7E7E7E7E6。存储于P0。

前面“稳定状态”章节讲到的广播地址，都是网内广播地址。新加入的设备不知道网络ID，所以新加入全网广播地址。

设备在加入网络后，除能全网广播地址，减少干绕，并保留ACK用。

### 网关初始化

网关使用推荐的8个信道中选择一个信道，避让自己wifi的频带。然后网关就工作在这一个信道。

网关根据序列号算出自己4字节的网络ID。

为了支持多网关，网关通过802.11 mesh网络互相通告网络ID，以及关联的设备信息。网关通过mesh网络了解到同一mesh中有使用和自己一样的信道，就更换信道。

### 一键配对

网关通过手机控制进入新设备发现模式。

设备在8个信道中切换发送probe全网广播帧，发送完后进入接收状态。当网关收到probe帧后，立刻应答加入报文,报文中携带如下内容：

设备ID：1字节。

时间戳：2字节。

网络ID：4字节。

序列号：8字节。

设备收到网关的回应后，就不再切换信道，保持和一键配对网关的通信。手机上显示出新发现的设备供用户确认，一旦确认后，网关分配设备的共享密钥发送到设备端。设备保存网络ID和共享密钥到flash，并用该密钥和网关认证，通信加密。

### 连接到网关

一键配对过的设备，启动时，自动连上配置过的网关。

某种原因设备5秒都没有收到网关的beacon帧，也需要重新连上配置过的网关。

重新连接时，发送probe和一键配对类似，只不过使用的地址是网内广播地址。网关的应答报文和一键配对的应答报文一致。收到网关回应后，设备使用网关分配的ID作为单播地址，开始认证。认证后，通信都是加密的。

为了支持多网关，设备和网关连上后，网关会将网关组的信息推送到设备端。设备端支持漫游，若发现信号变弱，将尝试连接网关组其他网关。

考虑到网关重启时大量设备重新连接网关会使得网络瘫痪，或者网关不在时，设备耗电快，重连重试的周期不能太短，失败后重连的时间需要随机退让。

### 漫游和安全性

从网关到手机端仍用原来的方式加密。从网关到设备端，使用一键配对时随机分配的秘钥加密和认证。手机登陆网关（组）时，需要输入网关密码，操作网关（组）下的设备的时候，不需要输入密码。

手机对网关组进行配置，分配一个共享密钥供网关组的网关之间通信，网关保存网关组的共享密钥和其他网关的网络号。

网关组的网关之间通过局域网二层协议发现，并使用手机分配的共享密钥认证和加密。一旦开始通信，就开始同步一键配对信息，即（序列号-设备共享密钥）对应关系，并保存。

设备一旦连上网关，网关推送网关组中所有的网络号到设备端，设备端保存到flash。

设备用一键配对获得的共享密钥，和网关组中的任意一个网络号，都能认证成功，并使用这个密钥加密。

### 重新配置

除了加reset按钮外，暂未想到比较安全的办法。