# 加入过程

## 信道扫描

从第一个信道开始扫描beacon，一直扫描到最后一个。记录下每一个信道的RSSI，LQI，空闲负载数量。设置一个ScanTimeOut，超时后跳到下一个信道。

## 簇头选择

根据每个采集到的信道的RSSI，LQI，负载数量判断。RSSI>LQI>负载数量(选择负载数量最小的SINK连接)。记录下选择到的最好的信道，发送入网申请。若没有扫描到可用的信道，输出错误后进行周期性休眠，休眠时间为ScanSleepPeriod。

## 入网申请

节点向目标SINK发出JoinRequest，等待接受ACK。若在ACKTimeOut时间内没有收到ACK，则重发JoinRequest，最多重发JoinRepeat(3)次。

SINK节点收到加入请求后需要遍历自己的网络参数表，找出空闲的LA，将LA填入ACK中。ACK中的accept置1。若拒绝加入，accept置0.

## 离开请求

1. 正常模式：
2. 主动离开：由于电量等原因需要主动离开网络，向sink发送LeaveRequest。其中包含离开原因。
3. 被动离开：sink节点如果连续PackageLostCount(3)次没有收到节点发来的消息，则认为该节点已经脱离连接，释放相关资源。
4. 低功耗模式(节点只有在检测到车辆时发送数据，其他时间休眠。(此模式可能会造成延迟或者测量不准确))

Sink节点在KeepAliveOut时间内没有收到节点发来的KeepAlive信息，则认为该节点已经离开，释放相关资源。

# 时隙同步

还没想好

# 重发机制

在每个超帧后增加ReSendSlotNum（节点数量/3）个时隙

# 网络记忆

节点需要记录自己的PA，LA

Sink需要记录自己的PA，LA，网络参数表

把相关信息写入到EEPROM中。

# 休眠

还没想好

# 调试

## 串口调试信息分级

根据DebugLevel把输出的串口调试信息分级，数字越高级别越高。预编译时剔除不用的串口信息，以免对时序造成影响。

## \*串口上位机

用python编写一个串口调试工具，可以读出网络当前所处的流程状态。尝试用串口替代另一节点的无线发包动作。