## **0．配置文件：**

[g\_connect]

[g\_connect]

send\_ip=172.17.18.11

send\_Port=5557

ip\_recvNum=1

ip\_addr1=172.17.18.80

recv\_Port1=5558

send\_ip：本地ip

send\_Port：本地端口

ip\_recvNum：板卡数量

ip\_addr\*：板卡ip

recv\_Port\*：板卡端口

修改黑白名单，需要reboot板卡

其他情况下，查询频点，只要有频点，修改其他就立即生效，无需开关频点

**动态库文件：g**\_control\_dll.dll

## **1．初始化运行**

DllExport void g\_initRun()；

## **2．小区设置**

DllExport void g\_setCellConfig(int ipIndex,U8 carrierIndicat, U8\* mcc,U8\* mnc, U8\* lac,U8\* ci, U8\* tac,U8\* carrierFrePoint, U8\* downAttenuation);

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| U8\*(char\*) | mcc | 国家码 |  |
| U8\* | mnc | 地区码 |  |
| U8\* | lac | 位置区号 |  |
| U8\* | ci | 小区号 |  |
| U8\* | tac | 周期 |  |
| U8\* | carrierFrePoint | 载波频点 |  |
| U8\* | downAttenuation | 下行衰减 |  |

## **3．运行模式**

DllExport void g\_setRunningMode(int ipIndex,U8 carrierIndicat, U8 controlMode, U8 workMode, U8 msgSendMode )；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| U8 | controlMode | 管控模式 | 整数。1，管控模式黑名单 2管控模式白名单3管控模式全部释放4管控模式全管控 |
| U8 | workMode | 工作模式 | 整数。1为侦码模式，2为管控模式，3定位模式，4救援模式5频点轮询模式 |
| U8 | msgSendMode | 短信发送模式 | 整数 1 为短信关闭，2黑名单发送3白名单发送4全部发送 |

## **4．打开射频**

DllExport void g\_turnOnRF(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **5．关闭射频**

DllExport void g\_turnOffRF(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **6．重启系统**

DllExport void g\_restartSystem(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

设备回复重启系统应答消息。重启系统消息用于更新小区的广播参数、自动修改LAC和清空小区用户缓存。对于GSM两小区板卡，根据载波指示字段进行区分。

## **7．重启板卡**

DllExport void g\_restartCard(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

设备回复重启板卡应答消息，并且在3秒后重启板卡。

## **8．添加黑名单（单个）**

DllExport void g\_addBlackImsi(int ipIndex,U8 carrierIndicat,char\* imsi)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| char\* | imsi | Imsi |  |

## **9．添加白名单（单个）**

DllExport void g\_addWhiteImsi(int ipIndex,U8 carrierIndicat,char\* imsi)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| char\* | imsi | Imsi |  |

## **10．查询黑名单**

DllExport void g\_checkBlackImsi(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **11．查询白名单**

DllExport void g\_checkWhiteImsi(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **12．删除黑名单（单个）**

DllExport void g\_deleteBlackImsi(int ipIndex,U8 carrierIndicat,char\* imsi)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| char\* | imsi | Imsi |  |

## **13．删除白名单（单个）**

DllExport void g\_deleteWhiteImsi(int ipIndex,U8 carrierIndicat,char\* imsi)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| char\* | imsi | Imsi |  |

## **14．清空黑名单（所有）**

DllExport void g\_clearBlackImsiList(int ipIndex,U8 carrierIndicat)

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **15．清空白名单（所有）**

DllExport void g\_clearWhiteImsiList(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **16．查询频点**

DllExport void g\_carrFrePointCheck(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号，从1开始 |  |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **17．回调函数：心跳返回**

DllExport int g\_getHeartBeatCallBack(int (\*pFun) (int ipIndex, int carrIndex,int state))；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Char | carrIndex | 载波编号 | 0x00：载波一，0x01：载波二 |
| Int | state | 状态 | 固定值为1，连接正常 |

## **18. 回调函数：信息返回**

DllExport int g\_getMsgReportCallBack(int (\*pFun)(MsgInfo\* p\_msgInfo));

typedef struct msgInfo

{

int ipIndex; //ip标号

U8 carrierIndex; //0:载波一;1:载波二

char msgType; //消息类型

} MsgInfo;

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| char | msgType | 数据类型 |  |

|  |  |
| --- | --- |
| **msgType** | |
| **取值** | 说明 |
| 0x02 | 小区设置 |
| 0x03 | 运行模式设置 |
| 0x04 | 开启射频 |
| 0x05 | 关闭射频 |
| 0x06 | 重启系统 |
| 0x07 | 重启板卡 |
| 0x50 | 添加黑名单 |
| 0x51 | 添加白名单 |
| 0x52 | 删除黑名单 |
| 0x53 | 删除白名单 |
| 0x54 | 清空黑名单 |
| 0x55 | 清空白名单 |

## **19．回调函数：imsi查询**

DllExport int g\_getImsiCheckCallBack(int (\*pFun)(ImsiCheckInfo\* p\_imsiCheckInfo));

typedef struct imsiCheckInfo

{

int ipIndex; //ip标号

U8 carrierIndex; //0:载波一;1:载波二

char msgType;

char imsi[16];

} ImsiCheckInfo;

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| U8 | carrierIndicat | 载波 | 0：载波一，1：载波二 |
| char | msgType | 数据类型 |  |
| char[16] | imsi | imsi |  |

|  |  |
| --- | --- |
| **msgType** | |
| **取值** | **说明** |
| 0x56 | 查询黑名单 |
| 0x57 | 查询白名单 |

## **20．回调函数：imsi/imei 上报**

DllExport int g\_getImsiReportCallBack (int (\*pFun)( ImsiReportInfo\* p\_ imsiReportInfo));

typedef struct imsiReportInfo

{

int ipIndex; //ip标号

Byte carrierIndex; //0:载波一;1:载波二

char imsi[16];

char imei[16];

}ImsiReportInfo;

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0：载波一，1：载波二 |
| Char[16] | imsi | imsi |  |
| char[16] | imei | imei |  |

## **21．回调函数：频点查询**

/\*

DllExport int g\_carrFrePointCheckCallBack(int (\*pFun)(CarrFrePointCheckInfo\* p\_carrCheckInfo))；

typedef struct carrFrePointCheckInfo{

int ipIndex;

Byte carrierIndex;

int carrFrePoint;

}CarrFrePointCheckInfo;

\*/

更改为：

DllExport int g\_carrFrePointCheckCallBack(int (\*pFun)(int IPIndex， U8 carrierIndicate，int carrFrePoint))；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0：载波一，1：载波二 |
| Int | carrFrePoint | 频点 |  |

## **22．信息查询**

DllExport void g\_checkMsg(int ipIndex,U8 carrierIndicat)；

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |

## **23．回调函数：信息查询**

DllExport int g\_checkMsgInfoCallBack(int (\*pFun)(CheckMsgInfo\* msgInfo))；

typedef struct checkMsgInfo{

int ipIndex;

Byte carrierIndex;

char mcc[8]; //

char mnc[8];

char lac[8];

char ci[8];

char carrierFrePoint[8];

char downAttenuation[8];

Byte controlMode;

Byte workMode;

Byte msgSendMode;

Byte radioFreq;

}CheckMsgInfo;

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 参数 | 名称 | 说明 |
| Int | ipIndex | ip编号 | 从1开始 |
| Byte | carrierIndicat | 载波 | 0x00：载波一，0x01：载波二 |
| Char[8] | mcc | 国家码 |  |
| Char[8] | mnc | 地区码 |  |
| Char[8] | lac | 位置区号 |  |
| Char[8] | ci | 小区号 |  |
| Char[8] | carrierFrePoint | 载波频点 |  |
| Char[8] | downAttenuation | 下行衰减 |  |
| Byte | controlMode | 管控模式 | 整数。0x01，管控模式黑名单； 0x02管控模式白名；0x03管控模式全部释放；0x04管控模式全管控 |
| Byte | workMode | 工作模式 | 整数。1为侦码模式，2为管控模式，3定位模式，4救援模式5频点轮询模式 |
| Byte | msgSendMode | 短信发送模式 | 整数 1 为短信关闭，2黑名单发送3白名单发送4全部发送 |
| Byte | radioFreq |  |  |