|  |
| --- |
| **合智安方** |
| 侦码接口定义 |
|  |
| 宁金龙  孙亭亭 |
| confidential |

# 目录

[目录 1](#_Toc417292643)

[文档修订历史记录 3](#_Toc417292644)

[1 设备接收消息 4](#_Toc417292645)

[1.1 消息格式 4](#_Toc417292646)

[1.2 消息类型 4](#_Toc417292647)

[1.3 消息结构 4](#_Toc417292648)

[1.3.1 系统参数RECV\_SYS\_PARA 4](#_Toc417292649)

[1.3.2 系统选项RECV\_SYS\_OPTION 7](#_Toc417292650)

[1.3.3 下行接收参数RECV\_DLRX\_PARA（TBD） 7](#_Toc417292651)

[1.3.4 邻区表参数RECV\_NC\_PARA 9](#_Toc417292652)

[1.3.5 IMSI库处理RECV\_IB\_OPER 9](#_Toc417292653)

[1.3.6 控制命令RECV\_CTRL\_CMD 10](#_Toc417292654)

[1.3.7 射频参数RECV\_RF\_PARA 11](#_Toc417292655)

[1.3.8 查询版本RECV\_QUERY\_VER 11](#_Toc417292656)

[1.3.9 计数清零RECV\_COUNT\_ZERO 11](#_Toc417292657)

[1.3.10 接收参数RECV\_RX\_PARA 11](#_Toc417292658)

[1.3.11 重新获取RECV\_NEW\_TAC 11](#_Toc417292659)

[1.3.12 获取当前参数RECV\_NOW\_PARA 12](#_Toc417292660)

[1.3.13 系统控制RECV\_SYS\_CONTROL 12](#_Toc417292661)

[1.3.14 配置系统时间RECV\_SYS\_TIME 13](#_Toc417292662)

[1.3.15 心跳包确认 RECV\_HEART\_BEAT\_CNF 13](#_Toc417292663)

[1.3.16 设备自检 RECV\_SELF\_CHECK（TBD） 13](#_Toc417292664)

[1.3.17 查询设备状态RECV\_DEV\_STATE 14](#_Toc417292665)

[1.3.18 风扇控制RECV\_FAN\_CTRL 14](#_Toc417292666)

[1.3.19 升级命令RECV\_UPDATE\_CMD 14](#_Toc417292667)

[2 设备发送消息 14](#_Toc417292668)

[2.1 消息格式 14](#_Toc417292669)

[2.2 消息类型 15](#_Toc417292670)

[2.3 消息结构 15](#_Toc417292671)

[2.3.1 打印信息SEND\_PRINT\_INFO 15](#_Toc417292672)

[2.3.2 心跳包SEND\_HEART\_BEAT 15](#_Toc417292673)

[2.3.3 用户设备信息SEND\_UE\_INFO 15](#_Toc417292674)

[2.3.4 请求确认SEND\_REQ\_CNF 17](#_Toc417292675)

[2.3.5 返回当前参数SEND\_NOW\_PARA 17](#_Toc417292676)

[2.3.6 寻呼UE发射功率SEND\_PAGING\_PWR（TBD） 18](#_Toc417292677)

[2.3.7 返回频点信息SEND\_FREQ（TBD） 18](#_Toc417292678)

[2.3.8 OM信息 SEND\_OM\_INFO 18](#_Toc417292679)

[2.3.9 版本信息SEND\_DEV\_VERSION 19](#_Toc417292680)

[2.3.10 自检结果 SEND\_CHECK\_RESULT（TBD） 20](#_Toc417292681)

[2.3.11 上报Lac号改变 SEND\_NEW\_LAC 21](#_Toc417292682)

[2.3.12 设备状态参数 SEND\_DEV\_STATE 21](#_Toc417292683)

# 文档修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 说明 | 作者 |
| 2015.04.10 | V0.1 | 初稿 | 合智安方 |
| 2015.04.17 | V0.2 |  | 闻远通信 |
|  | V0.3 |  | 闻远通信 |
|  | V0.4 |  | 闻远通信 |
| 2015.04.27 | V0.5 |  | 合智安方 |
| 2015.05.14 | V0.6 |  | 合智安方 |
| 2015.05.16 | V0.7 | 4字节对齐，加padding | 合智安方 |
|  | V0.9 |  | 闻远通信 |
|  | V1.0 | R9重定向参数，及其他细节 | 闻远通信 |
|  | V1.1 | 增加RECV\_REBOOT\_ENB，  修改RECV\_RF\_PARA，  修改SEND\_DEV\_VERSION | 闻远通信 |
|  |  |  |  |

# 1 设备接收消息

## 消息格式

struct msg\_struct

{

Uint16 head; //头部标识 0xffff

Uint16 pkt\_type; //消息类型

Uint32 data\_length; //消息长度

Uint8 data[4\*data\_length]; //DATA

};

## 1.2 消息类型

enum RecvPktType

{

RECV\_SYS\_PARA =1, //系统参数，MCC、MNC等

RECV\_SYS\_OPTION, //系统选项，如是否上报IMEI等

RECV\_DLRX\_PARA, //下行接收参数，用于获取公网信息

RECV\_NC\_PARA //邻区表参数

RECV\_IB\_OPER, =5, //IMSI库处理

RECV\_CTRL\_CMD, //寻呼控制命令

RECV\_RF\_PARA, //射频参数，如频点、功率等

RECV\_QUERY\_VER, //查询版本

RECV\_RX\_PARA, =10, //接收参数，如RACH接收功率门限等

RECV\_NEW\_TAC, //重新获取新的MCC、MNC、TAC和Cell ID

RECV\_NOW\_PARA, //获取当前设备内的参数

RECV\_HEART\_BEAT\_CNF, //心跳包确认

RECV\_DEV\_STATE, //返回设备状态

RECV\_REDIRECT\_R9

RECV\_REBOOT\_ENB

};

## 1.3 消息结构

### 1.3.1系统参数RECV\_SYS\_PARA

1）结构体定义

struct RecvSysPara //系统参数数据结构

{

Uint8 paraSysNo;

Uint8 paraMcc[3];

Uint8 paraMnc[2];

Uint16 paraPciNo;

Uint16 paraTac;

Uint8\_t padding1[2];

Uint32 paraCellId;

Uint32 paraPeri;

Uint8 paraIDReqNumOUT;

Uint8 paraIDReqNumIN;

uint8 UEcapaEnquiryNumOUT;

uint8 UEcapaEnquiryNumIN;

uint8 BoolMeasureOUT;

uint8 BoolMeasureIN;

Uint8 paraTAURejCauOUT;

Uint8 paraTAURejCauIN;

Uint8 paraATTRejCauOUT;

Uint8 paraATTRejCauIN;

Uint8 redirectedRATOUT;

Uint8 padding2;

Uint16 redirectedCarrierOUT;

Uint8 PriorityOUT;

Uint8 redirectedRATIN;

Uint16 redirectedCarrierIN;

Uint8 PriorityIN;

Uint8 redirectedCellIN;

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| paraSysNo | 系统号 | 1 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| paraMcc | 移动国家码 | 000～999 | 460 | 中国为460 |
| paraMnc | 移动网号 | 00～99 | 00 | 常用的MNC列表如下  00 移动  01 联通  02 移动  03 电信  05 电信  06联通  07 移动  11 电信 |
| paraPciNo | 物理层小区号 | 0~503 | 0 |  |
| paraTac | 跟踪区号 | 1～65535 | 1 |  |
| paraCellId[7] | 小区ECGI |  | 10 | EnodeB ID(20bit)  +cellID(8bit) |
| paraPeri | 周期性  更新TAC周期 | 0~500 | 0 | 基本单位  10ms  0the timer is deactivated. |
| paraIdReqNumOUT | 库外用户索取ID的次数 | 1~64 | 2 | 库外用户Identity request的次数 |
| paraIdReqNumIN | 库内用户索取ID的次数 | 1~64 | 4 | 库内用户Identity request的次数 |
| paraTAURejCauOUT | 库外用户位置更新  拒绝原因 | 2、3、5、6、  11、12、13、  15、17、22 | 15 | 一般不用更改  2IMSI\_UNKNOWN\_IN\_HSS  3ILLEGAL\_UE  5IMEI\_NOT\_ACCEPTED  6EPS\_SERVICES\_NOT\_ALLOWED  11PLMN\_NOT\_ALLOWED  12TRACKING\_AREA\_NOT\_ALLOWED  13ROAMING\_NOT\_ALLOWED  \_IN\_THIS\_TRACKING\_AREA  15NO\_SUITABLE\_CELLS  \_IN\_TRACKING\_AREA  17NETWORK\_FAILURE  22CONGESTION |
| ParaTAURejCauIN | 库内用户位置更新  拒绝原因 | 2、3、5、6、  11、12、13、  15、17、22 | 15 | 一般不用更改  2IMSI\_UNKNOWN\_IN\_HSS  3ILLEGAL\_UE  5IMEI\_NOT\_ACCEPTED  6EPS\_SERVICES\_NOT\_ALLOWED  11PLMN\_NOT\_ALLOWED  12TRACKING\_AREA\_NOT\_ALLOWED  13ROAMING\_NOT\_ALLOWED  \_IN\_THIS\_TRACKING\_AREA  15NO\_SUITABLE\_CELLS  \_IN\_TRACKING\_AREA  17NETWORK\_FAILURE  22CONGESTION |
| ParaATTRejCauOUT | 库外用户  ATTACH  拒绝原因 | 2、3、5、6、  11、12、13、  15、17、22 | 15 | 一般不用更改  2IMSI\_UNKNOWN\_IN\_HSS  3ILLEGAL\_UE  5IMEI\_NOT\_ACCEPTED  6EPS\_SERVICES\_NOT\_ALLOWED  11PLMN\_NOT\_ALLOWED  12TRACKING\_AREA\_NOT\_ALLOWED  13ROAMING\_NOT\_ALLOWED  \_IN\_THIS\_TRACKING\_AREA  15NO\_SUITABLE\_CELLS  \_IN\_TRACKING\_AREA  17NETWORK\_FAILURE  22CONGESTION |
| ParaATTRejCauIN | 库内用户  ATTACH  拒绝原因 | 2、3、5、6、  11、12、13、  15、17、22 | 15 | 一般不用更改  2IMSI\_UNKNOWN\_IN\_HSS  3ILLEGAL\_UE  5IMEI\_NOT\_ACCEPTED  6EPS\_SERVICES\_NOT\_ALLOWED  11PLMN\_NOT\_ALLOWED  12TRACKING\_AREA\_NOT\_ALLOWED  13ROAMING\_NOT\_ALLOWED  \_IN\_THIS\_TRACKING\_AREA  15NO\_SUITABLE\_CELLS  \_IN\_TRACKING\_AREA  17NETWORK\_FAILURE  22CONGESTION |
| redirectedRAT | 库外用户重定向RAT信息 | 1、2、3、4 | 1 | 0、重定向到E-UTRAN优先  1  2  3、重定向到UTRAN TDD优先 |
| redirectedCarrier | 库外用户重定向载波信息（EARFCN） | 0~65535 |  | 备注：库外用户通常重定向到E-UTRAN |
| Priority | 库外用户idle下重选优先级 | 0-7 |  | 对载波设置 |
| redirectedRAT | 库内用户重定向RAT信息 | 1、2、3、4 | 3 | 1、重定向到E-UTRAN优先  2、重定向到UTRAN TDD优先  3、重定向到GERAN优先  4、重定向到UTRAN FDD优先 |
| RedirectedCarrier | 库内用户重定向载波信息 | 0~65535 |  | Geran ARFCN |
| Priority | 库内用户idle下重选优先级 | 0-7 | 7 | 将Geran设为最高优先级，V9e0 |
| redirectedCell | 库内用户重定向小区信息 | 0-63 |  | BSIC包括NCC和BC |

8注：①ECGI发下去，打算怎么用？

>>先保留，配置下去，在系统消息中进行广播

② TAU更新周期打算怎么生效？照正常流程是要有带完保的TAU/ATTACH ACCEPT带下去这个值。现在不带正确的完保MAC，UE不会接受该消息设定的值。

>>该更新周期是基站设备自动更新TAI的周期

自动更新是否需要一个TAI列表？自动循环更新？

>>为方便用户操作，在界面人工手动配置时只要配置一个TAI即可。但为了实时刷新，我们在后台会配置一个TAI文件，这个会下载到板卡上，周期的变更并发送广播。这个配置可以放在你们的配置文件或者重启一个配置文件。

③Identity request的次数是指同一段时间内请求多次么？

>>是同一接入过程如TAU过程中Identity request的次数

，比如说对IMSI的请求次数，对IMEI的请求次数，这里是求和么？每次请求的间隔？？收到回复再请求？假设没收到回复，加个定时器？定时器要考虑和基站的release定时器不要冲突。

>>只是对IMSI的请求次数。收到回复就请求。

>>索要IMSI有UE不回复的情况吗？

>>需要考虑基站release UE的时间

我们是根据IMSI区分UE是属于库内还是库外的吗？

>>对的，对库内用户要多次索取。库外用户只要1次就够了

既然已经知道UE属于库内还是库外，就说明知道了UE的IMSI，为什么还要请求多次。

如果不知道UE的IMSI，就不知道UE是属于库内还是库外，怎么确定请求次数。

>>请求次数界面配置。在release UE之前完成这些次数的ID request

### 1.3.2 系统选项RECV\_SYS\_OPTION

1）结构体定义

struct RecvSysOption //系统选项数据结构

{

Uint16 opSysNo;

Uint8 opLuImei;

Uint8 opLuSTMSI;

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| opSysNo | 系统号 | 1 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| opLuImei | 登录时获取IMEI | 0/1 | 0 | 1表示在该手机登录时获取IMEI号，  0表示不获取 |
| opLuSTMSI | 登录时获取STMSI | 0/1 | 0 | 1表示在该手机登录时获取STMSI号，  0表示不获取 |
|  |  |  |  |  |

注:tmsi获取开关，需要m\_tmsi还是s\_tmsi？

>>s-tmsi

### 1.3.3 下行接收参数RECV\_DL\_SNIFF

1）结构体定义

struct RecvDlRxPara //下行接收参数数据结构

{

Uint8 dlSysNo;

Uint16 earfcn;

Uint32 dlFn; //帧数

Uint8 dlRxMod;

Uint8 dlEnable;

Uint8 padding

Uint8 padding

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| dlSysNo | 系统号 | 1/2 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| eafrcn | 信道号（EARFCN） | TDD  Band38:37750-38249  Band39: 38250-38649  Band 40:38650-38849  38850-39349  39350-39549  Band 41:40240-40439  40900-41039  41040-41239  FDD |  | 信道号 |
| dlFn | 总帧数 | 1~65535 | 100 | 测试使用，进行接收机灵敏度测试时计算BLER的总帧数 |
| dlRxMod | 下行接收模式 | 1/2/3 | 1 | 1表示正常接收下行广播，  2表示测试上行频段灵敏度，  3表示测试下行频段灵敏度 |
| dlEnable | 下行接收使能 | 0/1 | 0 | 1表示开始下行接收，  0表示结束下行接收 |

注：请详细描述该条的功能。可暂不用实现？！

>>

1表示正常接收下行广播：表示正常的sniffer

2表示测试上行频段灵敏度：用于上行测试模式，用于测试上行的接收性能

3表示测试下行频段灵敏度：用于下行测试模式，用于测试下行的接收性能

### 1.3.4邻区表参数RECV\_NC\_PARA

1）结构体定义

struct RecvNcPara //邻区参数数据结构

{

Uint8 ncSysNo;

Uint8 ncCmdType; //设置为0x01 清除为0x02

Uint8padding;

struct intraFreqCell; //同频邻区

struct interFreqCellList

};

struct intraFreqCell //同频邻区

{

Uint8 NumIntraFreqCell;

Uint8[3]padding;

Uint16 PCI[6];

};

struct interFreqCellList //异频邻区列表

{

Uint8 NumInterFreq;

Uint[3]padding;

struct interFreqList[6];

};

struct interFreqList //异频邻区列表

{

Uint8 NumInterFreqCell;

Uint8[3]padding;

Uint16 PCI[6];

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| ncSysNo | 系统号 | 1/2 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| ncCmdType | 命令类型 | 1/2 | 1 | 1表示设置邻区表，  2表示清除邻区表 |
| intraFreqCell | 同频邻区列表PCI | 0-503 | 1 | 邻区0~5的值必须按从小到大的顺序排列。 |
| NumIntraFreqCell同频邻区的个数 | 0~6 | 1 |  |
| interFreqCellList | interFreqList  异频邻区ARFCN列表 | 0-6 | 1 | 邻区0~5的值必须按从小到大的顺序排列。 |
| PCI  异频邻区PCI列表 | 0-503 | 1 | 邻区0~5的值必须按从小到大的顺序排列。 |
| NumInterFreq | 0~6 | 1 | 异频的个数 |
| NumInterFreqCell | 0~6 | 1 | 每个频率下的邻区个数 |

注：区分同频、异频邻区列表？

>>这个需要区分，需要细化，参照LTE系统信息的邻区配置；按照E-UTRANSIB 3/4/5/6/7进行各字段的设置，待添加。

### 1.3.5IMSI库处理RECV\_IB\_OPER

1）结构体定义

struct RecvNcPara //IMSI库参数数据结构

{

Uint8 ibSysNo;

Uint8 ibCmdType; //添加为1，删除为2，清空为3

Uint8 ibImsi[8];

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| ibSysNo | 系统号 | 1/2 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| ibCmdType | 命令类型 | 1/2/3 | 1 | 1表示在IMSI库中添加IMSI，  2表示在IMSI库中删除IMSI，  3表示清空IMSI。 |
| ibImsi[8] | IMSI号 |  |  | 数据格式为16进制，如460079440025157表示为ibImsi [0]~ ibImsi [7]分别为0x04, 0x60, 0x07, 0x94, 0x40, 0x02, 0x51, 0x57, ibImsi [0]的高4位无效 |

注：这里对IMSI库的处理，是为了做黑名单用？

>>可以作为黑/白名单用

### 1.3.6控制命令RECV\_CTRL\_CMD

1）结构体定义

struct RecvCtrlCmd //控制命令参数数据结构

{

Uint8 ctrlSysNo;

Uint8 ctrlCmdType; //连续寻呼为0x01

Uint8 ctrlCmdPara; //对连续寻呼而言，该参数为寻呼间隔

Uint8 ctrlPagingIDType； //寻呼ID类型

Uint8 ctrlImsi[8];

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| ctrlSysNo | 系统号 | 1/2 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| ctrlCmdType | 命令类型 |  | 1 | 1表示连续寻呼 |
| ctrlCmdPara | 命令参数 | 1~255 | 1 | 寻呼间隔，单位为秒 |
| CtrlPagingIDType | 寻呼类型参数 | 1/2 |  | 1：S-TMSI寻呼  2：IMSI寻呼 |
| ctrlImsi[8] | IMSI号 |  |  | 数据格式为16进制，如460079440025157表示为ibImsi [0]~ ibImsi [7]分别为0x04, 0x60, 0x07, 0x94, 0x40, 0x02, 0x51, 0x57, ibImsi [0]的高4位无效  当CtrlPagingIDType为2时，CtrlImsi的低5字节表示40bit S-TMSI值，高3字节无效。 |

注：这里，寻呼流程的作用是什么？正常的是寻呼发下去后，上来SERVICE REQUEST，后续的流程怎么处理？

>>参考给你们的参考流程。在AKA过程无法通过的后续流程不用处理，通过RRC release进行释放

>>请参考一下流程。用户被寻呼后索要他的ID和Capability，然后用户被重定向到其他小区或或者其他RAT



### 1.3.7射频参数RECV\_RF\_PARA

1）结构体定义

struct RecvRfPara //rf参数数据结构

{

Uint16 rfSysNo;

Uint8 rfEnable;

fastConfigEarfcn

Uint16 eutra\_band;

Uint16 DlEarfcn;

Uint16 UlEarfcn;

Uint8 FrameStrucureType;

Uint8 SubframeAssinment;

Uint8 specialSubframePatterns;

Uint8 UlBandWidth;

Uint8 DlBandWidth;

Uint8 padding;

Uint32 rfPwr;

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| rfSysNo | 系统号 | 1 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| rfEnable | 射频使能 | 0/1 | 0 | 0表示射频关闭，  1表示射频打开 |
| fastConfigEarfcn | 快速配置频点 | 0/1 | 1 | 0 开启快速配置频点  1 关闭快速配置频点 |
| eutra\_band | 频带号 | 38/39/40/41 |  |  |
| earfcn | 信道号EARFCN | TDD  Band38:37750-38249  Band39: 38250-38649  Band 40:38650-38849  38850-39349  39350-39549  Band 41:40240-40439  40900-41039  41040-41239  FDD |  |  |
| rfPwr | 发射功率 | -33~43 | -33 | 单位dBm |

注：信道号指EARFCN？BAND要配置么？

>>rfFreq和（BAND，EARFCN）的对应关系在基站设备内对应，或者在接口上直接配置（BAND，EARFCN）

### 1.3.8查询版本RECV\_QUERY\_VER

无内容。

### 1.3.10接收参数RECV\_RX\_PARA

待定义。

### 1.3.11重新获取RECV\_NEW\_TAC

1）结构体定义

struct RecvNewTac //重新设置位置区TAI相关信息

{

Uint8 newSysNo;

Uint16 newTimer;

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| newSysNo | 系统号 | 1/2 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| newTimer | 定时器 | 0~500 | 0 | 间隔多长时间重新设置一次，单位基本  10ms，  0表示定时器无效。 |

### 1.3.12获取当前参数RECV\_NOW\_PARA

1）结构体定义

struct RecvNowPara //获取当前参数

{

Uint8 nowSysNo;

Uint16 nowType; //返回的请求类型，同enum RecvPktType

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 符号 | 名称 | 取值范围 | 默认值 | 说明 |
| nowSysNo | 系统号 | 1/2 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| nowType | 返回类型 |  |  | 同enumRecvPktType，  如要返回当前的RecvSysPara参数，cnfType=1 |

注：该条的功能需要进一步讨论。

>>这条消息的功能是返回最近RecvPktTyp的消息内容



### 1.3.13心跳包确认RECV\_HEART\_BEAT\_CNF

无内容

### 1.3.14 RECV\_REDIRECT\_R9

重定向R9的配置参数

typedef struct \_RedirectedR9 //rf参数数据结构

{

uint8\_t BoolR9;

uint8\_t MsgType;

uint16\_t CellID;

uint16\_t LAC;

uint8\_t MSCR;

uint8\_t ba\_ag\_blks\_res;

uint8\_t ccch\_conf;

uint8\_t bs\_pa\_mfrms;

uint8\_t T3412\_value;

uint8\_t pwrc;

uint8\_t DTX;

uint8\_t radio\_link\_timeout;

uint8\_t MsTxPwrMaxCch;

uint8\_t Neci;

uint8\_t RxlevAccessMin;

uint8\_t RachMax\_retrans;

uint8\_t RachTx\_int;

uint8\_t CelllReselOffset;

}RedirectedR9;

### 1.3.15 RECV\_REBOOT\_ENB







# 2 设备发送消息

## 2.1 消息格式

struct msg\_struct

{

Uint16head; //头部标识 0xff

Uint16 pkt\_type; //消息类型

Uint32 data\_length; //密文消息长度

Uint8 data[4\*data\_length]; //DATA

};

## 2.2 消息类型

enum SendPktType

{

SEND\_HEART\_BEAT, //心跳包，每10s发送一次

SEND\_UE\_INFO, //用户设备信息，包括IMSI、IMEI等

SEND\_REQ\_CNF, //确认接收到的请求

SEND\_NOW\_PARA =0x0005, //返回设备的当前参数

SEND\_PAGING\_PWR, //上报寻呼UE发射功率

SEND\_FREQ, //返回频点信息

SEND\_DEV\_VERSION, //打印版本信息

SEND\_NEW\_TAC, //报告新的PLMN\TAC号生成

SEND\_DEV\_STATE, //上报设备状态数据

};

## 2.3 消息结构

### 2.3.2 心跳包SEND\_HEART\_BEAT

无内容，每10s发送一次。

### 2.3.3 用户设备信息SEND\_UE\_INFO

1）结构体定义

struct SendUeInfo //UE信息数据结构

{

Uint16 ueSysNo;

Uint16 ueSendNo; //发送序号

Uint8 CurrentMcc[3];

Uint8 CurrentMnc[2];

Uint8 padding1;

Uint16 CurrentTac;

char ueImsi[16];

char ueImei[16];

char ueSTmsi[12];

Uint8 ueMsisdn[4]; //手机号码号段

Uint8 ueType； //手机类型（如LTE单模数据卡,多模LTE终端）

Uint8 ueTaType; //位置更新类型

Unit8 ueTaTime[19] //设备端取号的时间戳

Uint8 uePwrNo; //有效功率值的数量

int8 uePwr[100]; //交互过程中接收的手机信号功率值

};

2）说明

|  |  |  |  |
| --- | --- | --- | --- |
| 符号 | 名称 | 格式 | 说明 |
| ueSysNo | 系统号 |  | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| ueSendNo | 发送序号 |  | 从0开始计算 |
| CurrentMcc | 当前Mcc |  |  |
| CurrentMnc | 当前Mnc |  |  |
| CurrentTac | 当前Tac |  |  |
| ueType | UE终端类型 |  | Bit  8 7 6 5 4 3 2 1  Reserved 0 WCDMA TD-S CDMA GSM  0 LTE  1 LTE/GSM  2 LTE/CDMA2000  3 LTE/CDMA2000/GSM  4 LTE/TD-S  5 LTE/TD-S/GSM  6 LTE/TD-S/CDMA2000  7 LTE/TD-S/CDMA2000/GSM  8 LTE/WCDMA  9 LTE/WCDMA/GSM  10 LTE/WCDMA/CDMA2000  11 LTE/WCDMA/CDMA2000/GSM  12 LTE/WCDMA/TD-S  13 LTE/WCDMA/TD-S/GSM  14 LTE/WCDMA/TD-S/CDMA2000  15 LTE/WCDMA/TD-S/CDMA2000/GSM |
| ueImsi[16] | IMSI号 |  | 从ueImsi[0]开始填 |
| ueImei[16] | IMEI号 |  | 同IMSI |
| ueSTmsi | STMSI |  | 同IMSI |
| ueMsisdn [4] | 手机号码号段 | 16进制 | 如1398626表示为ueMsisdn [0]~ueMsisdn [3]分别为0x13, 0x98, 0x62, 0x60, ueMsisdn [3]的低4位无效。用于查询归属地。 |
| ueTaType | 位置更新类型 | 简写：  1 EpsAtt  2 EPSImsiAtt  6 EpsEmerAtt  16 TAu  17 TaLau  18 TaLauImsi  19 PeriU  24 TaUBr  25 TaLaUBr  26 TaLaImsiBr  27 PeriUBr | 全称：  1 EPS attach  2 EPS attach  6 EPS emergency attach  16 TA updating，No bear requested  17 combined TA/LA updating，No bear requested  18 combined TA/LA updating with IMSI attach，No bear requested  19 periodic updating，No bear requested  24 TA updating， bear requested  25 combined TA/LA updating， bear requested  26 combined TA/LA updating with IMSI attach， bear requested  27 periodic updating， bear requested  解析：  Bits  8 7 6 5  0 0 0 0 ATTACH  0 0 0 1 TAU  ATTACH情况下  Bits  4 3 2 1  0 0 0 1 EPS attach  0 0 1 0 combined EPS/IMSI attach  0 1 1 0 EPS emergency attach  0 1 1 1 reserved  TAU情况下  Bits  3 2 1  0 0 0 TA updating  0 0 1 combined TA/LA updating  0 1 0 combined TA/LA updating with IMSI attach  0 1 1 periodic updating  1 0 0 unused  1 0 1 unused  Bits  4  0No bearer establishment requested  1Bearer establishment requested |
| ueTaTime[19] | 设备端取号的时间戳 |  | 表示设备抓取号的时间格式为：  2013-10-19 14:21:20 |
| uePwrNo | 上报功率数 |  | 获取该IMSI号手机功率的次数 |
| uePwr[100] | 上报功率值 |  | 获取该IMSI号手机的功率值，PUSCH power@1PRB |

### 2.3.4 请求确认SEND\_REQ\_CNF

1）结构体定义

struct SendReqCnf //请求确认数据结构

{

Uint16 cnfSysNo;

Uint16 cnfType; //确认的请求类型，同enum RecvPktType

Uint16 cnfInd; //0表示正确接收，1表示参数错误...

Uint16 padding;

};

2）说明

|  |  |  |  |
| --- | --- | --- | --- |
| 符号 | 名称 | 格式 | 说明 |
| cnfSysNo | 系统号 |  | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| cnfType | 确认的请求类型 |  | 同enumRecvPktType，  如对RecvSysPara消息的确认，cnfType=1 |
| cnfInd | 确认指示 |  | 0表示正确接收，  1表示参数错误... |

### 2.3.5 返回当前参数SEND\_NOW\_PARA

1）结构体定义

struct SendNowPara //获取当前参数

{

Uint8 nowSysNo;

Uint16 nowType; //返回的请求类型，同enum RecvPktType

Uint8 nowPara[100]; //返回的参数

};

2）说明

|  |  |  |  |
| --- | --- | --- | --- |
| 符号 | 名称 | 格式 | 说明 |
| nowSysNo | 系统号 |  | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| nowType | 返回类型 |  | 同enumRecvPktType，  如要返回当前的RecvSysPara参数，nowType=1 |
| nowPara[100] | 返回参数 |  | 同发送的结构体，  如要返回系统参数RECV\_SYS\_PARA，则nowPara的定义同structRecvSysPara |

### 2.3.6 寻呼UE发射功率SEND\_PAGING\_PWR

1）结构体定义

struct SendPagingPwr

{

Uint8 pagingSysNo;

Uint8 pagingImsi[8];

Uint8 uePwrNo;

Uint8 padding;

int8uePwr[100]; //交互过程中接收的手机信号功率值

};

2）说明

|  |  |  |  |
| --- | --- | --- | --- |
| 符号 | 名称 | 格式 | 说明 |
| pagingSysNo | 系统号 |  | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| pagingImsi [8] | IMSI号 | 16进制 | 数据格式为16进制，如460079440025157表示为ibImsi [0]~ ibImsi [7]分别为0x04, 0x60, 0x07, 0x94, 0x40, 0x02, 0x51, 0x57, ibImsi [0]的高4位无效 |
|  |  |  |  |
| uePwrNo | 上报功率数 |  | 获取该IMSI号手机功率的次数 |
| uePwr[100] | 上报功率值 |  | 获取该IMSI号手机的功率值，PUSCHpower@1PRB |

### 2.3.7 返回频点信息SEND\_FREQ

1）结构体定义

struct SendFreq //返回频点值

{

Uint8 sysNo;

Uint16earfcn;

};

2）说明

|  |  |  |  |
| --- | --- | --- | --- |
| 符号 | 名称 | 格式 | 说明 |
| sysNo | 系统号 |  | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| earfcn | 频点 |  | 下行接收返回的设备应该设置的频点值 |



### 2.3.9版本信息SEND\_DEV\_VERSION

1）结构体定义

struct SEND\_DEV\_VERSION //上报操作管理信息

{

Uint8 type;

Uint8 fpgaVersion[2];

Uint8 BBUVersion[2];

char SoftWareVersion[64];

};

2）说明

|  |  |  |  |
| --- | --- | --- | --- |
| 符号 | 名称 | 格式 | 说明 |
| type | 上报的类型 |  | 0代表设备主动上报，  1代表服务端请求后上报 |
| cpuVersion | CPU(gsm\_code)版本 |  | 十六进制[0x20][0x13][0x11][0x28][0x01][0x02]表示：2013/11 /28 0102 |
| dspVersion | Dsp版本 |  | 同上 |
| fpgaVersion | FPGA版本 |  | 同上 4个十六进制数 |

### 2.3.10返回下行接收参数SEND\_DL\_SNIFF

1）结构体定义

struct SEND\_DL\_SNIFF

{

Uint8 numberCell;

struct cellInfomation[numberCell];

};

struct cellInfomation

{

Uint8 SysNo;

Uint8 Mcc;

Uint8 Mnc;

Uint16 Tac;

Uint16 PciNo;

Uint8 CellId[7];

Uint16 Earfcn

Uint8 CP\_Type;

Uint16 Crs\_RP;

Uint16 Crs\_RQ;

};

2）说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SysNo | 系统号 | 1 | 1 | 对双系统设备取值为1或2，  对单系统设备只能为1 |
| Mcc | 移动国家码 | 000～999 | 460 | 中国为460 |
| Mnc | 移动网号 | 00～99 | 00 | 常用的MNC列表如下  00 移动  01 联通  02 移动  03 电信  05 电信  06联通  07 移动  11 电信 |
| PciNo | 物理层小区号 | 0~503 | 0 |  |
| Tac | 跟踪区号 | 1～65535 | 1 |  |
| paraCellId[7] | 小区ECGI |  | 10 | EnodeB ID(20bit)  +cellID(8bit) |
| EARFCN | 绝对频点号 | 取值范围：0~65535 |  | 小区频点。采用E-UTRA绝对射频信道编号（EARFCN，E-UTRA Absolute Radio Frequency Channel Number）表示。 |
| CP\_Type | 循环前缀类型 | 取值范围：0或1 |  | 循环前缀类型：  0：CP\_TYPE\_NORMAL常规CP  1：CP\_TYPE\_EXTEND扩展 CP |
| CRS\_RP |  |  |  | Cell specific RS RP  数值范围:-1200~ 0, 单位0.125dBm。  对应实际信号范围为：-150dBm~0dBm |
| CRS\_RQ |  |  |  | Cell specific RS RQ  数值范围:-800~ 800, 单位：0.0625dB  对应实际信号范围为：-30dB~-3dB |

# 3.序列图

### 3.1心跳检测

流程描述：PC设置10s定时器，侦听设备上报的心跳包SEND\_HEART\_BEAT，10s内收到心跳则清空定时器，10s后仍没收到心跳，则界面上系统状态显示红色，表示已经断开连接。



### 3.2下行sniffer

流程描述：PC配置下行接收参数RECV\_DL\_SNIFF，点击测试，设备返回请求确认SEND\_REQ\_CNF，设备上报返回小区信息SEND\_DL\_SNIFF。



### 3.3邻区表设置

流程描述：PC打开邻区表配置界面，配置好参数，点设置，给设备发送邻区表参数RECV\_NC\_PARA，点清除修改ncCmdType发送RECV\_NC\_PARA，设备收到后返回请求确认SEND\_REQ\_CNF。



### 3.4配置和获取系统选项、系统参数、射频参数

流程描述：PC配置系统参数和系统选项点确定，PC发送RECV\_SYS\_PARA和RECV\_SYS\_OPTION给设备，设备收到回复请求确认SEND\_REQ\_CNF，PC点获取系统参数和系统选项，PC发送获取当前参数RECV\_NOW\_PARA给设备，设备收到回复请求确认SEND\_REQ\_CNF，并上报返回当前参数SEND\_NOW\_PARA，PC收到后显示。之后PC配置射频参数，勾选发射选项，发送射频参数RECV\_RF\_PARA给设备，设备收到后返回请求确认SEND\_REQ\_CNF，并开始工作，上报用户设备信息SEND\_UE\_INFO给PC，PC收到后显示在侦码接收界面。PC配置射频参数，取消勾选发射选项，发送RECV\_RF\_PARA给设备，设备收到返回请求确认SEND\_REQ\_CNF，并停止扫描上报工作。



### 3.5控制命令

PC根据设备上报的UE信息，配置并发送控制命令RECV\_CTRL\_CMD，设备收到后返回请求确认SEND\_REQ\_CNF，并返回寻呼UE发射功率SEND\_PAGING\_PWR消息，PC收到后显示在功率列表中。



### 3.6更新TAC

PC配置并下发RECV\_NEW\_TAC，设备收到收到后返回请求确认SEND\_REQ\_CNF，并周期性更新TAC。



### 3.7IMSI库处理（添加、删除、清空属于本地操作，同步属于PC与设备的操作）

PC输入IMSI，点击添加、删除、清空、同步、查询任意一个按钮，PC下发IMSI库处理消息RECV\_IB\_OPER，设备收到后返回请求确认SEND\_REQ\_CNF，并对列表进行处理后返回当前参数SEND\_NOW\_PARA。

### 3.8查询版本

PC发送查询版本RECV\_QUERY\_VER，设备收到后返回请求确认SEND\_REQ\_CNF，并返回版本信息SEND\_DEV\_VERSION。

# 4、消息处理流程

