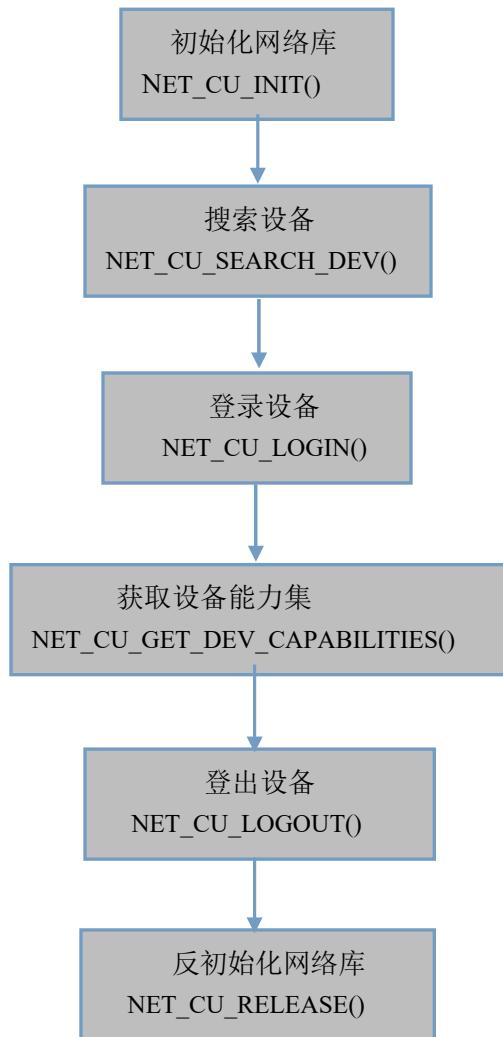


目录

一、函数调用流程.....	2
1.添加登录设备流程.....	2
2.打开对讲流程.....	3
二、接口定义.....	4
1. 初始化网络库.....	4
2. 反初始化网络库.....	4
3. 搜索设备.....	4
4. 组播设置设备网络参数.....	4
5. 登录设备.....	4
6. 登出设备.....	5
7. 异步获取设备能力集.....	5
8. 订阅或取消订阅告警消息.....	5
9. 开始实时视频预览.....	6
10. 停止视频预览.....	6
11. 获取预览码流.....	6
12. 云台控制.....	6
13. 查询设备录像.....	7
14. 设备端抓图.....	7
15. 录像下载.....	7
16. 停止下载录像.....	8
17. 录像回放.....	8
18. 停止录像回放.....	8
19. 录像回放选择播放.....	8
20. 开始对讲.....	9
21. 停止对讲.....	9
22. 发送对讲数据.....	9
23. 获取远程参数.....	9
24. 设置远程参数.....	10
25. 远程升级.....	10
26. 停止升级.....	11
27. 格式化磁盘.....	11
28. 查询格式化进度.....	11
29. 停止格式化.....	11
30. 设备控制.....	11
31. 获取设备状态.....	12

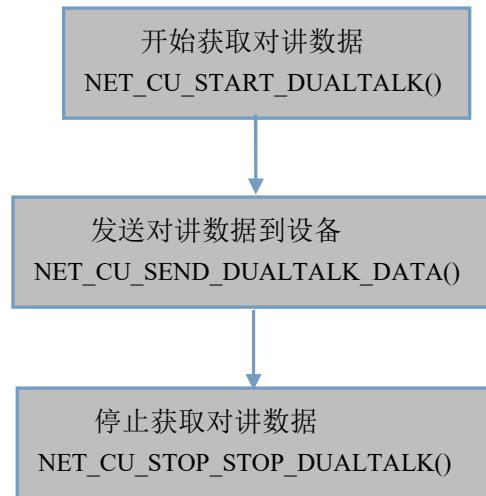
一、函数调用流程

1. 添加并登录设备流程



2. 打开对讲流程

前置条件：设备已登录



备注：图中的调用流程展示了获取和设置设备参数，预览和停止预览，对讲和停止对讲的函数调用流程，是最为典型和常用的业务功能，其他的业务功能可以参见 demo 中的源码。

二、接口定义

1. 初始化网络库

函数原型:

```
BOOL NET_CU_INIT()
```

参数: 无

返回值: 成功返回 TRUE, 失败返回FALSE

备注: 无

2. 反初始化网络库

函数原型:

```
void NET_CU_RELEASE()
```

参数: 无

备注: 无

3. 搜索设备

函数原型:

```
void NET_CU_SEARCH_DEV(const NET_DEV_SearchDevParam* Param)
```

参数: NET_DEV_SearchDevParam成员:

ProtocolType: 协议类型, 一般使用私有协议, NET_DEV_PROTOCOL_ME

UserParam: 用户自定义参数

Callback: 用户回调, 用于通知用户搜索结果, 每搜到一个设备, 会回调一次

返回值: 无

备注: 用户回调函数传入参数如果不为空, 则指向搜索到的设备数据; 如果为空则表示搜索结束。

4. 组播设置设备网络参数

函数原型:

```
BOOL NET_CU_SET_DEV_INFO(const NET_DEV MODIFY_NET* Param)
```

参数: NET_DEV_MODIFY_NET的定义请参见结构体头文件NET_DEV_MODIFY_NET

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该返回值不能作为是否设置成功的依据。

备注: 无

5. 登录设备

函数原型:

```
BOOL NET_CU_LOGIN(const NET_DEV_ConnParam* Param, NET_DEV_Connection *Conn)
```

参数: NET_DEV_ConnParam成员:

IpAddr: 设备IP

Port: 端口号
UserName: 用户名
PassWord: 密码
AutoReconnect: 是否自动重连
EventDesired: 是否订阅报警事件
TimeOut: 超时时间
bIsP2PDevice: 是否是P2P设备
P2PDeviceUUID: P2P设备的唯一ID
P2PLocalMsgPort: P2P设备映射到本地的命令端口
UserParam: 用户自定义参数
EventCallback: 报警事件通知回调
hWnd: 进入用户回调后要发送通知的窗口句柄
Message: 进入用户回调后要发送通知的窗口消息
StatusCallback: 连接状态回调函数

Conn: 用于保存和设备的连接句柄

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该返回值不能作为是否登录成功的标识, 是否登录成功需要在连接状态的回调函数里判断。

备注: 无

6. 登出设备

函数原型:

```
void NET_CU_LOGOUT(NET_DEV_Connection Conn)
```

参数: Conn: 用户保存的设备连接句柄

返回值: 无

备注: 登出设备后, 用户应该将保存的连接句柄置空

7. 异步获取设备能力集

函数原型:

```
BOOL NET_CU_GET_DEV_CAPABILITIES(NET_DEV_Connection Conn, const  
NET_DEV_GetDeviceCapabilities* param)
```

参数: Conn: 用户保存的设备连接句柄

NET_DEV_GetDeviceCapabilities 成员:

UserParam: 用户自定义参数

pCapabilities: 接收设备能力集的内存指针

DeviceCapabilitiesCallback: 用户回调函数

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该返回值不能作为是否获取成功的标识。

备注: 无

8. 订阅或取消订阅告警消息

函数原型:

```
void NET_CU_ALARM_SUBSCRIBE(NET_DEV_Connection Conn, BOOL Desired)
```

参数: Conn: 用户保存的设备连接句柄

Desired: TRUE订阅, FALSE取消订阅

返回值: 无

备注: 无

9. 开始实时视频预览

函数原型:

```
BOOL NET_CU_START_PREVIEW(NET_DEV_Connection Conn  
                           , const NET_DEV_PreviewParam* Param  
                           , NET_DEV_Preview *Preview);
```

参数: Conn: 用户保存的设备连接句柄

NET_DEV_PreviewParam成员:

Channel: 通道号

StreamType: 码流类型, 赋值请参见头文件中NET_DEV_STREAM_TYPE

Content: 流混合类型, 赋值请参见头文件中NET_DEV_STREAM_MIX_TYPE

Protol: 传输协议, 赋值请参见头文件中NET_DEV_TRANSPROTOL

AutoReconnect: 是否自动重连

wRTSPPort: P2P设备时, 使用的端口号

UserParam: 用户自定义参数

DataCallback: 接收视频流帧数据的回调

hWnd: 进入用户回调后要发送通知的窗口句柄

Message: 进入用户回调后要发送通知的窗口消息

StatusCallback: 预览状态通知回调

Preview: 保存视频预览的句柄, 停止预览时需要使用该参数

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该返回值不能作为是否获取成功的标识。

备注: 该函数的使用需要配合解码库的接口, 详细的使用方法请参见demo中的源码。

10. 停止视频预览

函数原型:

```
void NET_CU_STOP_PREVIEW(NET_DEV_Preview Preview)
```

参数: Preview开始视频预览时传回的预览句柄。

返回值: 无

备注: 停止预览后, 用户应该把保存的预览句柄置空。

11. 获取预览码流

函数原型:

```
ULONGLONG NET_CU_GET_PREVIEW_BITRATE(NET_DEV_Preview Preview)
```

参数: Preview开始视频预览时传回的预览句柄。

返回值: 码流值

备注: 只有在开启预览视频时才有效。

12. 云台控制

函数原型:

```
BOOL NET_CU_PTZ_CONTROL(NET_DEV_Connection Conn, UINT Channel, const  
                           NET_DEV_PTZ_CONTROL* Req);
```

参数: Conn: 用户保存的设备连接句柄

Channel: 通道号

NET_DEV_PTZ_CONTROL的定义请参见结构体头文件

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该函数不能作为操作是否成功的标识。

备注: 无

13. 查询设备录像

函数原型:

```
BOOL NET_CU_QUERY_RECORD(NET_DEV_Connection Conn, const  
NET_DEV_QueryRecordParam* Param);
```

参数: Conn: 用户保存的设备连接句柄

NET_DEV_QueryRecordParam成员:

Channel: 通道号

Cond: 查询录像条件, 包括录像类型和起止时间, 具体定义请参见接头文件

NET_DEV_RECORD_COND的定义

UserParam: 用户自定义参数

DetailCallback: 按天查询时的回调函数

OutlineCallback: 按月查询时的回调函数

返回值: 同上

备注: 无

14. 设备端抓图

函数原型:

```
BOOL NET_CU_CAPTURE(NET_DEV_Connection Conn, const NET_DEV_DevCaptureParam*  
Param)
```

参数: Conn: 用户保存的设备连接句柄

NET_DEV_DevCaptureParam成员:

Channel: 通道号

Req: 抓图输入参数, NET_DEV_MANUL_SNAPPIC_REQ的定义请参见结构体头文件

FileName: 抓图文件保存名称

UserParam: 用户自定义参数

返回值: 同上

备注: 无

15. 录像下载

函数原型:

```
BOOL NET_CU_START_DOWNLOAD_RECORD(NET_DEV_Connection Conn,  
const NET_DEV_DownloadRecordParam* Param,  
NET_DEV_DownloadRecord *DownloadRecord)
```

参数: Conn: 用户保存的设备连接句柄

Param: NET_DEV_DownloadRecordParam的成员:

Channel: 通道号

RecordItem: 设备上一条录像记录的信息, 具体定义参见结构体头文件

LocalParam: 包含保存文件类型和文件名的结构体

UserParam: 用户自定义参数
Callback: 下载进度通知回调
DownloadRecord: 录像下载连接句柄
返回值: 同上
备注: 无

16. 停止下载录像

函数原型:

```
void NET_CU_STOP_DOWNLOAD_RECORD(NET_DEV_DownloadRecord DownloadRecord)
```

参数: DownloadRecord: 开始下载录像时传回的下载连接句柄

返回值: 无

备注: 下载录像的进度回调如果发现进度值达到100，则需要主动调用该接口来关闭连接。

17. 录像回放

函数原型:

```
BOOL NET_CU_START_RECORD_PLAY(NET_DEV_Connection Conn,  
                               const NET_DEV_RecordPlayParam* Param,  
                               NET_DEV_RecordPlay *RecordPlay)
```

参数: Conn: 用户保存的设备连接句柄

Param: NET_DEV_RecordPlayParam的成员:

Channel: 通道号

RecordType: 录像类型

BeginTime: 开始时间

EndTime: 结束时间

UserParam: 用户自定义参数

DataCallback: 接收帧数据的回调

hWnd: 消息通知的窗口句柄

Message: 消息ID

StatusCallback: 录像回放状态回调

RecordPlay: 录像回放的连接句柄

返回值: 调用网络库成功返回TRUE，失败返回FALSE，该函数不能作为操作是否成功的标识。

备注: 无

18. 停止录像回放

函数原型:

```
void NET_CU_STOP_RECORD_PLAY(NET_DEV_RecordPlay RecordPlay)
```

参数: RecordPlay: 开始录像回放时传回的句柄

返回值: 无

备注: 无

19. 录像回放选择播放

函数原型:

```
void NET_CU_REPOS_RECORD_DATA(NET_DEV_RecordPlay RecordPlay, const  
NET_DEV_ReposRecordReq* Req)
```

参数: RecordPlay: 开始录像回放是传回的句柄

Req: NET_DEV_ReposRecordReq的成员:

BeginTime: 选择播放的时刻

EndTime: 无效

返回值: 无

备注: 无

20. 开始对讲

函数原型:

```
BOOL NET_CU_START_DUALTALK(NET_DEV_Connection Conn,  
                           const NET_DEV_DualtalkParam* Param,  
                           NET_DEV_Dualtalk* DualTalk);
```

参数: Conn: 用户保存的设备连接句柄

Param: NET_DEV_DualtalkParam的成员:

Channel: 通道号

UserParam: 用户自定义参数

DataCallback: 接收设备发来的音频帧数据的回调

hWnd: 消息通知窗口句柄

Message: 消息通知

StatusCallback: 对讲状态回调

DualTalk: 对讲连接句柄

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该函数不能作为操作是否成功的标识。

备注: 无

21. 停止对讲

函数原型:

```
void NET_CU_CALL NET_CU_STOP_DUALTALK(NET_DEV_Dualtalk DualTalk)
```

参数: DualTalk: 开始对讲时传回的对讲连接句柄

返回值: 无

备注: 无

22. 发送对讲数据

函数原型:

```
void NET_CU_SEND_DUALTALK_DATA(NET_DEV_Dualtalk DualTalk, void* Data, UINT  
Size);
```

参数: DualTalk: 开始对讲时传回的对讲连接句柄

Data: 音频帧数据

Size: 音频帧大小

返回值: 无

备注: 无

23. 获取远程参数

函数原型:

```
BOOL NET CU QUERY PARAM(NET DEV Connection Conn, const  
NET DEV QueryParamParam* Param)
```

参数: Conn: 用户保存的设备连接句柄

Param: NET DEV QueryParamParam的成员:

Channel: 通道号

ParamID: 要获取的参数的ID, 具体的值请参见结构体头文件中

NET DEV PARAM ID的定义

InParam: 适用于需要输入信息的远程参数的情况, 具体的值请参见

NET DEV PARAM QUERY的定义

OutBuffer: 接收获取到的远程参数的内存地址

OutBufferSize: 接收获取到的远程参数的内存大小

ReturnedSize: 实际获取到的远程参数的大小

TimeOut: 超时时间, 以秒为单位

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该函数不能作为操作是否成功的标识。

备注: 无

24. 设置远程参数

函数原型:

```
BOOL NET CU SET PARAM(NET DEV Connection Conn, const NET DEV SetParamParam*  
Param)
```

参数: Conn: 用户保存的设备连接句柄

Param: NET DEV SetParamParam的成员:

Channel: 通道号

ParamID: 要获取的参数的ID, 具体的值请参见结构体头文件中

NET DEV PARAM ID 的 定 义

InBuffer: 要设置的参数源地址

InBufferSize: 要设置的参数的大小

TimeOut: 超时时间, 以秒为单位

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该函数不能作为操作是否成功的标识。

备注: 无

25. 远程升级

函数原型:

```
BOOL NET CU START REMOTE UPGRADE(NET DEV Connection Conn,  
const NET DEV RemoteUpgradeParam* param,  
NET DEV RemoteUpgrade* RemoteUpgrade);
```

参数: Conn: 用户保存的设备连接句柄

Param: NET DEV RemoteUpgradeParam的成员:

FileType: 升 级 文 件 的 类 型

FileName: 升级文件名

UserParam: 用户自定义参数

Callback: 升级进度回调函数

RemoteUpgrade: 远程升级连接句柄

返回值：调用网络库成功返回TRUE，失败返回FALSE，该函数不能作为操作是否成功的标识。

备注：无

26. 停止升级

函数原型：

```
void NET_CU_STOP_REMOTE_UPGRADE(NET_DEV_RemoteUpgrade RemoteUpgrade)
```

参数：RemoteUpgrade：开始远程升级时传回的升级连接句柄

返回值：无

备注：格式化进度达到100时需要主动调用该接口关闭连接。

27. 格式化磁盘

函数原型：

```
BOOL NET_CU_FORMAT_DISK(NET_DEV_Connection Conn, const  
NET_DEV_FormatDiskParam *Param, NET_DEV_FormatDisk* FormatDisk)
```

参数：Conn：用户保存的设备连接句柄

Param：NET_DEV_FormatDiskParam的成员：

DiskNo：磁盘号

UserParam：用户参数

Callback：无效，格式化进度需要主动去查询

FormatDisk：格式化连接句柄

返回值：调用网络库成功返回TRUE，失败返回FALSE，该函数不能作为操作是否成功的标识。

备注：应用程序关闭，设备磁盘格式化不会停止。

28. 查询格式化进度

函数原型：

```
void NET_CU_QUERY_FORMAT_PROGRESS(NET_DEV_Connection Conn, const  
NET_DEV_FormatDiskParam* Param)
```

参数：Conn：用户保存的设备连接句柄

Param：NET_DEV_FormatDiskParam的成员：

DiskNo：磁盘号

UserParam：用户参数

Callback：回调通知查询到的格式化进度

返回值：无

备注：无

29. 停止格式化

函数原型：

```
void NET_CU_STOP_FORMAT_DISK(NET_DEV_FormatDisk FormatDisk)
```

参数：FormatDisk：开始格式化时，传回的格式化句柄

返回值：无

备注：格式化完成后要调用该接口关闭连接

30. 设备控制

函数原型：

```
BOOL NET_CU_DEVICE_CONTROL(NET_DEV_Connection Conn, const  
NET_DEV_CONTROL *Param)
```

参数: Conn: 用户保存的设备连接句柄

Param: 控制命令参数, 具体值请参见结构体头文件中NET_DEV_CONTROL的定义

返回值: 调用网络库成功返回TRUE, 失败返回FALSE, 该函数不能作为操作是否成功的标识。

备注: 无

31. 获取设备状态

函数原型:

```
BOOL NET_CU_DEVICE_STATES(NET_DEV_Connection Conn, const  
NET_DEV_QueryDeviceStateParam* Param)
```

参数: Conn: 用户保存的设备连接句柄

Param: NET_DEV_QueryDeviceStateParam的成员:

UserParam: 用户自定义参数

Req: 指明要获取设备的什么状态

Rsp: 接收获取状态值的内存指针

OutBufferSize: 接收获取状态值的内存大小

TimeOut: 超时时间, 以秒为单位

返回值: 同上

备注: 无