

CHD_WMPBOX SDK

Windows Linux IOS Mac 版

文档版本 01

发布日期 2015-03-07

部件编码



开发参考

杭州科技有限公司为客户提供多样化的产品方案定制和全方位的技术支持，具体服务请联系公司总部。

杭州炽鸟科技有限公司

客户服务邮箱：chirdtcl@aliyun.com

版权所有 ©杭州炽鸟科技有限公司，保留一切权利。

非经公司书面许可，任何单位或个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



为杭州炽鸟科技有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人所有。

注意

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

前言	2
1. 概述	4
1.1 API函数参考域	4
1.2 基本数据类型定义	4
2. 基本操作	6
2.1 概述	6
2.2 API参考	6
3. 视频操作	13
3.1 概述	13
3.2 API参考	13
4. 串口操作	43
4.1 概述	43
4.2 API参考	43
5. 音频操作	64
5.1 概述	64
5.2 API参考	64
6. GPIO操作	78
6.1 概述	78
6.2 API参考	78
7. I2C 操作	82
7.1 概述	82
7.2 API参考	82
8. 高级设置	86
8.1 概述	86
8.2 API参考	86
9. AVI录像	94
9.1 概述	94
9.2 API参考	94
10. 简单示例	98
10.1 内容简介	98

10.2 程序流程图 98

10.3 测试程序 99

前言

概述

本开发手册专门为使用杭州炽鸟科技有限公司的 **CHD**系列产品，进行上位机开发的软件工程师而作。本手册详细描述了炽鸟科技**SDK**开发包中所提供的**API**接口的使用方法以及注意事项等。

在本手册最后，提供了使用**SDK**的简单测试程序，供开发人员开发进行参考。

产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
CHD-T5 150M无线视频模块	V1
CHD-T7 300M无线视频模块	V1

读者对象

本文档主要适用于以下的工程师：

- 技术支持工程师
- 软件开发工程师

约定

通用格式约定

格式	说明
宋体	正文采用宋体表示。
黑体	一级、二级、三级标题采用黑体。
楷体	警告、提示等内容一律用楷体表示。
Courier New格式	API 语法，数据类型定义采用Courier New格式表示。

数值单位约定

地址、数据的表达方式说明如下：

符号	举例	说明
0x	0xFFFFFFFF	用16进制表示的数据值、协议值

修订记录

修订日期	版本	修订说明
2015-03-01	01	创建
2015-03-05	01	校验
2015-03-07	01	审核发布

1. 概述

1.1 API函数参考域

本文档对API参考信息使用以下域来描述，具体信息如表 1-1 所示。

表 1-1 API函数参考域说明

参考域	作用
描述	描述 API 的功能。
语法	显示 API 的语法样式。
参数	列出 API 的参数、参数说明及其属性。
返回值	列出 API 的返回值及其返回值说明。
注意	使用 API 时应注意的事项。
参数相关	同本 API 相关的参数信息。

1.2 基本数据类型定义

```
#define IN // 输入
#define OUT // 输出
typedef unsigned long CHD_WMP_T; // 设备句柄

typedef unsigned char CHD_U8;
typedef char CHD_S8;
typedef unsigned short CHD_U16;
typedef short CHD_S16;
typedef unsigned int CHD_U32;
typedef int CHD_S32;
typedef unsigned long CHD_UL32;
typedef long CHD_SL32;
```

```

typedef void                                CHD_VOID;

#define CONST const

#define SUCCESSFUL                          (0x0000)
typedef enum{
    CHD_RET_SUCCESS                        = SUCCESSFUL,           // 成功
    CHD_RET_FAILED                         = (SUCCESSFUL - 1),    // 失败
    CHD_RET_TIMEOUT                       = (SUCCESSFUL - 2),      // 超时
    CHD_RET_CONNECT_ERROR                 = (SUCCESSFUL - 3),      // 连接错误
    CHD_RET_SEND_ERROR                    = (SUCCESSFUL - 4),      // 网络错误
    CHD_RET_RECV_ERROR                    = (SUCCESSFUL - 5),      // 网络错误
    CHD_RET_DISCONNECT                    = (SUCCESSFUL - 6),      // 设备已掉线
    CHD_RET_HANDLE_ERROR                  = (SUCCESSFUL - 7),      // 句柄错误
    CHD_RET_SESSIONID_ERROR               = (SUCCESSFUL - 8),      // 会话ID错误
    CHD_RET_PROTOCOL_ERROR                = (SUCCESSFUL - 9),      // 协议错误
}CHD_RET_E;

```


2. 基本操作

2.1 概述

操作设备及数据传输所必须的操作函数。

2.2 API参考

设备搜索API(已知设备IP则此步骤可省略):

- `CHD_WMP_ScanDevice_Init` : 初始化设备扫描模块
- `CHD_WMP_ScanDevice_UnInit` : 设备扫描反初始化
- `CHD_WMP_Scan_GetDeviceInfo` : 获取搜索到的设备信息

设备连接及数据监听API:

- `CHD_WMP_ConnectDevice` : 连接设备
- `CHD_WMP_Disconnect` : 断开设备连接
- `CHD_WMP_Poll` : 数据监听

编译相关:

- 头文件 : `CHD_WMP_Type.h`、`CHD_WMP_Return.h`、`CHD_WMP.h`
- Windows库 : `chd_base.dll`、`chd_efast.dll`、`chd_wmp.dll`
- Linux库 : `chd_base.so`、`chd_efast.so`、`chd_wmp.so`
- IOS库 : `chd_base.a`、`chd_efast.a`、`chd_wmp.a`
- Mac库 : `chd_base.dylib`、`chd_efast.dylib`、`chd_wmp.dylib`

CHD_WMP_ScanDevice_Init

【描述】

设备扫描初始化，自动创建扫描线程，并定时扫描同一网段下的在线设备信息。

【语法】

```
CHD_RET_ECHD_WMP_ScanDevice_Init(INCHD_U32 SearchInterTime);
```

【参数】

参数名称	描述	输入/输出
SearchInterTime	定时扫描间隔时间	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败

【注意】

1、程序退出需调用 CHD_Devface_SerachUninit() 函数销毁扫描线程

【参数相关】

无

CHD_WMP_ScanDevice_UnInit

【描述】

设备扫描反初始化，销毁设备扫描线程。

【语法】

```
CHD_RET_E CHD_WMP_ScanDevice_UnInit();
```

【参数】

无

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	扫描设备未初始化或参数错误

【注意】

无

【参数相关】

无

CHD_WMP_Scan_GetDeviceInfo

【描述】

获取扫描到的在线设备个数及设备当前信息。

【语法】

```
CHD_RET_E CHD_WMP_Scan_GetDeviceInfo(OUTCHD_DEVINFO_S*dev,  
                                       OUTCHD_U32 *Number);
```

【参数】

参数名称	描述	输入/输出
dev	设备信息结构体	输出
Number	扫描到的在线设备个数	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	扫描设备未初始化或参数错误

【注意】

- 1、必须先设备扫描初始化后才能获取到设备信息。
- 2、设备扫描时最多存储MAX_ONLINE_DEVICE (默认值为50)个不同在线设备。用户不可自行修改。

【参数相关】

```
#define MAX_ONLINE_DEVICE    50  
#define CHD_ALIAS_LEN      20  
#define CHD_IPADDR_LEN     20  
  
typedef struct{  
    CHD_U32  Id;  
    CHD_S8   Alias[CHD_ALIAS_LEN];  
    CHD_S8   IpAddr[CHD_IPADDR_LEN];  
}CHD_DEVINFO_S;
```

成员名称	描述
Id	设备ID号
Alias	设备别名
IpAddr	设备IP地址

CHD_WMP_ConnectDevice

【描述】

设备连接，初始化网络，等待数据传输。

【语法】

```
CHD_RET_E CHD_WMP_ConnectDevice(OUTCHD_WMP_T * phandle,
                                  INCHD_S8 * IPAddress);
```

【参数】

参数名称	描述	输入/输出
phandle	设备操作句柄	输出
IPAddress	设备IP地址	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	连接成功
CHD_RET_FAILED	连接失败
CHD_RET_HANDLE_ERROR	输入参数错误
CHD_RET_CONNECT_ERROR	无法连接设备，请确认设备IP地址是否正确
CHD_RET_RECV_ERROR	网络传输错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络传输错误，网络环境较差，传输不稳定

【注意】

- 1、请确保设备已正常工作。
- 2、请确保设备网络连通。

【参数相关】

无

CHD_WMP_Disconnect

【描述】

断开设备连接，销毁并回收资源。

【语法】

```
CHD_RET_E CHD_WMP_Disconnect(IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handler	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、程序退出或连接另一设备前，请调用此函数销毁线程并回收资源。

【参数相关】

无

CHD_WMP_Poll

【描述】

数据监听函数，同步模式，实时监听视频、图片、串口、音频数据的到来。

【语法】

```
CHD_RET_E CHD_WMP_Poll(IN CHD_WMP_T handle,  
                        OUT CHD_STREAM_TYPE_E * type,  
                        IN CHD_U32 timeout_sec,  
                        IN CHD_U32 timeout_msec);
```

【参数】

参数名称	描述	输入/输出
handler	设备操作句柄	输入
type	监听到来数据类型	输出
timeout_sec	超时时间（秒）	输入
timeout_msec	超时时间（毫秒）	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_TIMEOUT	超时
CHD_RET_HANDLE_ERROR	操作句柄错误
CHD_RET_PROTOCOL_ERROR	传输协议错误
CHD_RET_DISCONNECT	网络断开

【注意】

- 1、实时监听函数，当有数据到来或监听超时函数返回。
- 2、此为同步模式，用户必须调用此函数后才可接收设备数据。
- 3、建议用户需开一线程实时监听数据。

【参数相关】

```
typedef enum {  
    CHD_STREAM_VIDEO           = 0x00,  
    CHD_STREAM_PICTURE         = 0x01,  
    CHD_STREAM_AUDIO           = 0x02,  
    CHD_STREAM_SERIAL           = 0x03,  
    CHD_PARAM_CHANGE_VABILITY = 0x04,  
    CHD_PARAM_CHANGE_VPARAM    = 0x05,  
    CHD_PARAM_CHANGE_VCTRL     = 0x06,  
    CHD_PARAM_CHANGE_AUDIO     = 0x07,  
    CHD_PARAM_CHANGE_SERIAL    = 0x08,  
    CHD_PARAM_CHANGE_GPIO      = 0x09,  
} CHD_STREAM_TYPE_E;
```

成员名称	描述
CHD_STREAM_VIDEO	视频数据
CHD_STREAM_PICTURE	图片数据
CHD_STREAM_AUDIO	音频数据
CHD_STREAM_SERIAL	串口数据
CHD_PARAM_CHANGE_VABILITY	像头性能参数发生改变
CHD_PARAM_CHANGE_VPARAM	摄像头设置参数发生改变
CHD_PARAM_CHANGE_VCTRL	摄像头控制参数发生改变
CHD_PARAM_CHANGE_AUDEO	音频参数发生改变
CHD_PARAM_CHANGE_SERIAL	串口参数发生改变
CHD_PARAM_CHANGE_GPIO	GPIO状态发生改变

3. 视频操作

3.1 概述

用户只需调用函数接口即可实现摄像头参数控制、视频实时传输，拍照等功能。

- 视频格式

在视频采集中，由于摄像头的不同，采集的视频流数据格式也不同，常见的有MJPEG格式，YUYV格式，YUV422格式，还有高端的H264格式。这些格式的产生是由于使用了不同的视频数据编码格式或视频标准。

- 视频分辨率

在本模块中，视频分辨率是由摄像头来决定的，不同摄像头支持的分辨率不同，采集的图像的分辨率也不同。

- 视频控制参数

在本模块中，视频控制参数主要有：亮度，对比度，饱和度，色调，白平衡，伽玛值，增益，清晰度，光补偿，曝光值。这些控制参数都是结构体类型，在结构体中具体进行控制。如亮度控制结构体中，包含最大亮度值，最小亮度值，亮度递增量，自动调整亮度使能等控制变量。

3.2 API参考

视频数据相关API:

- | | |
|--|------------|
| ● CHD_WMP_Video_Begin | : 开启视频流 |
| ● CHD_WMP_Video_End | : 停止视频流 |
| ● CHD_WMP_Video_RequestVideoData | : 获取视频数据 |
| ● CHD_WMP_Video_ReleaseVideoData | : 释放视频数据缓存 |

拍照相关API:

- `CHD_WMP_Video_SnapShot` : 拍照
- `CHD_WMP_Video_SnapShotResolu` : 固定分辨率拍照
- `CHD_WMP_Video_RequestPicData` : 获取照片数据
- `CHD_WMP_Video_ReleasePicData` : 释放照片数据缓存

视频图片缓存相关API:

- `CHD_WMP_Video_GetCurVideoFrameNum` : 获取当前采集到的视频帧数
- `CHD_WMP_Video_GetCurPictureFrameNum` : 获取当前采集到的图片缓存数
- `CHD_WMP_Video_GetPeerMaxFrameNum` : 获取设备最大视频缓存帧数
- `CHD_WMP_Video_GetLocalMaxFrameNum` : 获取本地最大视频缓存帧数
- `CHD_WMP_Video_SetPeerMaxFrameNum` : 设置设备视频最大缓存帧数
- `CHD_WMP_Video_SetLocalMaxFrameNum` : 设置本地视频缓存帧数

摄像头控制相关API:

- `CHD_WMP_Video_GetAbility` : 获取摄像头设备性能
- `CHD_WMP_Video_GetVideoCtrl` : 获取摄像头控制参数
- `CHD_WMP_Video_GetParam` : 获取摄像头设置参数
- `CHD_WMP_Video_GetFormat` : 获取视频格式
- `CHD_WMP_Video_GetResolu` : 获取视频分辨率
- `CHD_WMP_Video_GetFPS` : 获取视频帧率
- `CHD_WMP_Video_SetVideoCtrl` : 设置摄像头控制参数
- `CHD_WMP_Video_ResetVCtrl` : 恢复摄像头控制参数默认值
- `CHD_WMP_Video_SetFormat` : 设置视频格式
- `CHD_WMP_Video_SetResolu` : 设置视频分辨率
- `CHD_WMP_Video_SetFPS` : 设置视频帧率

编译相关:

- 头文件 : `CHD_WMP_Type.h`、`CHD_WMP_Return.h`、`CHD_WMP_Video.h`
- Windows库 : `chd_base.dll`、`chd_efast.dll`、`chd_wmp.dll`
- Linux库 : `chd_base.so`、`chd_efast.so`、`chd_wmp.so`
- IOS库 : `chd_base.a`、`chd_efast.a`、`chd_wmp.a`
- Mac库 : `chd_base.dylib`、`chd_efast.dylib`、`chd_wmp.dylib`

CHD_WMP_Video_Begin

【描述】

开启视频流。

【语法】

```
CHD_RET_E CHD_WMP_Video_Begin(IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Video_End

【描述】

停止视频流。

【语法】

```
CHD_RET_E CHD_WMP_Video_End (IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Video_RequestVideoData

【描述】

获取一帧视频数据。

【语法】

```
CHD_RET_E CHD_WMP_Video_RequestVideoData(IN CHD_WMP_T handle,  
                                           OUT CHD_VIDEO_FRAME_S ** pvdata);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pvdata	视频数据结构体指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

- 1、必须配合CHD_WMP_Poll()函数使用，调用CHD_WMP_Poll()函数后网络才会接受设备传输过来的视频数据。
- 2、可用CHD_WMP_Video_GetCurVideoFrameNum() 获取本地视频缓存数
- 3、视频缓存为动态开辟，使用完成必须调用CHD_WMP_Video_ReleaseVideoData() 函数释放视频缓存，否则会造成内存泄露。

【参数相关】

```
typedef struct{
    CHD_U32    bexist;
    CHD_U32    format;
    CHD_U32    resolu;
    CHD_U32    fps;
    CHD_U32    BPS;
    CHD_U32    timestamp;
    CHD_U32    datalen;
    CHD_S8     pData[0];
}CHD_VIDEO_FRAME_S;
```

成员名称	描述
bexist	当前摄像头是否存在
format	视频格式
resolu	视频分辨率，由图像宽和高组合而成
fps	视频帧率
BPS	传输速率
timestamp	时间戳
datalen	当前帧视频数据长度
pData[0]	视频数据起始地址

分辨率、图像宽高转换函数：

```
void CHD_Video_AnalysisResolu(int resolu,
                               unsigned int * w, unsigned int * h);
```

CHD_WMP_Video_ReleaseVideoData

【描述】

释放视频帧缓存。

【语法】

```
CHD_RET_E CHD_WMP_Video_ReleaseVideoData(IN CHD_WMP_T handle,  
                                             IN CHD_VIDEO_FRAME_S * pvdata);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pvdata	指向视频帧数据的指针	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Video_SnapShot

【描述】

拍照，请求设备获取当前一帧图片数据。

【语法】

```
CHD_RET_E CHD_WMP_Video_SnapShot(IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、必须结合CHD_WMP_Poll()、CHD_WMP_Video_RequestPicData() 函数才能获取图片数据。

【参数相关】

无

CHD_WMP_Video_SnapShotResolu

【描述】

固定分辨率拍照，请求设备获取固定分辨率的图片数据。

【语法】

```
CHD_RET_E CHD_WMP_Video_SnapShotResolu(INCHD_WMP_T handle,  
                                           IN CHD_U32 Resolu);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Resolu	拍照的分辨率	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、图片数据使用图片流进行传输，必须结合 CHD_WMP_Poll()、CHD_WMP_Video_RequestPicData() 函数才能获取图片数据。

【参数相关】

宽高转换为分辨率函数：

```
int CHD_Video_GroupResolu(const unsigned int width,  
                           const unsigned int height);
```

CHD_WMP_Video_RequestPicData

【描述】

获取一帧图像数据。

【语法】

```
CHD_RET_E CHD_WMP_Video_RequestPicData(IN CHD_WMP_T  
                                         handle, OUT CHD_VIDEO_FRAME_S **  
                                         pvdata);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pvdata	视频数据结构体指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

- 1、必须配合CHD_WMP_Poll()函数使用，调用CHD_WMP_Poll()函数后网络才会接收设备传输过来的图片数据。
- 2、图片缓存为动态开辟，使用完成必须调用CHD_WMP_Video_ReleaseVideoData()函数释放图片缓存，否则会造成内存泄露。

【参数相关】

<pre>typedef struct{ CHD_U32 bexist; CHD_U32 format; CHD_U32 resolu; CHD_U32 fps; CHD_U32 BPS; CHD_U32 timestamp; CHD_U32 datalen; CHD_S8 pData[0]; }CHD_VIDEO_FRAME_S;</pre>	
成员名称	描述
bexist	当前摄像头是否存在
format	图像格式
resolu	图像分辨率，由图像宽和高组合而成
fps	视频帧率，总是为零，可忽略
BPS	传输速率，总是为零，可忽略
timestamp	时间戳
datalen	当前帧图像数据长度
pData	图像数据起始地址

分辨率、图像宽高转换函数：

```
void CHD_Video_AnalysisResolu(int resolu, unsigned int * w,  
                                unsigned int * h);
```

CHD_WMP_Video_ReleasePicData

【描述】

释放图片帧缓存。

【语法】

```
CHD_RET_ECHD_WMP_Video_ReleaseVideoData(IN          CHD_WMP_T  
                                           handle, IN  CHD_VIDEO_FRAME_S  *  
                                           pvdata);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pvdata	指向视频帧数据的指针	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Video_GetCurVideoFrameNum

【描述】

获取当前接收到的视频缓存帧数。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetCurVideoFrameNum(  
    IN CHD_WMP_T handle, OUT CHD_U32 * pNum);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pNum	整形指针，用于存放获取到缓存帧数	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Video_GetCurPictureFrameNum

【描述】

获取当前接收到的图片缓存帧数。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetCurPictureFrameNum(  
    IN CHD_WMP_T handle, OUT CHD_U32 * pNum);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pNum	整形指针，用于存放获取到缓存帧数	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Video_GetPeerMaxFrameNum

【描述】

获取设备采集视频可缓存的最大帧数。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetPeerMaxFrameNum (  
    IN CHD_WMP_T handle, OUT CHD_U32 * pNum);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pNum	整形指针，用于存放获取到缓存帧数	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Video_GetLocalMaxFrameNum

【描述】

获取本地接收视频最大的缓存帧数，超过则覆盖。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetCurPictureFrameNum(  
    IN CHD_WMP_T handle, OUT CHD_U32 * pNum);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pNum	整形指针，用于存放获取到缓存帧数	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Video_SetPeerMaxFrameNum

【描述】

设置设备采集视频可缓存的最大帧数。

【语法】

```
CHD_RET_E CHD_WMP_Video_SetPeerMaxFrameNum(  
    IN CHD_WMP_T handle, IN CHD_U32 Num);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Num	最大缓存帧数	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Video_SetLocalMaxFrameNum

【描述】

设置本地接收视频最大缓存帧数。

【语法】

```
CHD_RET_E CHD_WMP_Video_SetLocalMaxFrameNum(  
    IN CHD_WMP_T handle, IN CHD_U32 Num);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Num	最大缓存帧数	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Video_GetAbility

【描述】

获取摄像头设备性能（支持哪些采集格式、分辨率及帧率等信息）。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetAbility(IN CHD_WMP_T handle,  
                                     OUT CHD_VIDEO_ABILITY_S * pstAbility);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pstAbility	指向摄像头设备性能的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

- 1、设备支持多种采集格式，每种采集格式下支持多种分辨率，每种分辨率下支持多种采集帧率。
- 2、设置格式、分辨率、帧率时请确保所设置的参数设备是支持的。

【参数相关】

```
typedef enum{
    CHD_FMT_YUYV = 0x1,
    CHD_FMT_MJPEG,
    CHD_FMT_H264
}CHD_VIDEO_FMT_E;

#define FORMAT_MAX_NUM 4
#define RESOLU_MAX_NUM 12
typedef struct video_support_ability
{
    struct {
        CHD_U32 format;
        struct {
            CHD_U32 resolu;
            CHD_U32 maxfps;
        }resolu[RESOLU_MAX_NUM];
    }format[FORMAT_MAX_NUM];
}CHD_VIDEO_ABILITY_S;
```

成员名称	描述		
format	结构体数组，保存了设备所支持的全部采集格式。		
	成员名称	描述	
	format	视频格式： CHD_FMT_YUYV CHD_FMT_MJPEG CHD_FMT_H264	
	resolu	结构体数组，当前视频格式下支持的所有采集分辨率	
		成员名称	描述
		resolu	当前格式下的分辨率
fps	当前分辨率下的采集帧率		

CHD_WMP_Video_GetVideoCtrl

【描述】

获取摄像头单个控制参数（如亮度、白平衡、曝光值等）。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetVideoCtrl(IN CHD_WMP_T handle,
    IN CHD_VCTRL_E type, OUT CHD_VIDEO_CTRL_S * pstCtrl);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
type	摄像头控制参数类型	输入
pstCtrl	指向存放摄像头控制参数的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、摄像头控制参数由摄像头而决定，与设备无关。

【参数相关】

```
typedef enum{  
    CHD_VCTRL_BRIGHTNESS = 0x0,  
    CHD_VCTRL_CONTRAST,  
    CHD_VCTRL_SATURATION,  
    CHD_VCTRL_HUE,  
    CHD_VCTRL_WHITE_BALANCE,  
    CHD_VCTRL_GAMMA,  
    CHD_VCTRL_GAIN,  
    CHD_VCTRL_SHARPNESS,  
    CHD_VCTRL_BACKLIGHT,  
    CHD_VCTRL_EXPOSURE,  
    CHD_VCTRL_MAX,  
}CHD_VCTRL_E;
```

成员名称	描述
CHD_VCTRL_BRIGHTNESS	亮度
CHD_VCTRL_CONTRAST	对比度
CHD_VCTRL_SATURATION	饱和度
CHD_VCTRL_HUE	色调
CHD_VCTRL_WHITE_BALANCE	白平衡
CHD_VCTRL_GAMMA	伽马
CHD_VCTRL_GAIN	增益
CHD_VCTRL_SHARPNESS	清晰度
CHD_VCTRL_BACKLIGHT	背光补偿
CHD_VCTRL_EXPOSURE	曝光值

```
typedef struct
{
    CHD_S32  val_valid; // 值为1时，后面五个参数才有效
    CHD_S32  minval;
    CHD_S32  curval;
    CHD_S32  maxval;
    CHD_S32  stepval;
    CHD_S32  defval;

    CHD_S32  auto_valid;
    CHD_S32  autoval;
}CHD_VIDEO_CTRL_S;
```

成员名称	描述
val_valid	设备是否支持type指定的控制参数类型 如：设备是否支持曝光值调节 为1则支持，否则不支持
minval	控制参数最小值
curval	控制参数当前值
maxval	控制参数最大值
stepval	调节步长
defval	控制参数默认值
auto_valid	设备是否支持自动调节 为1则支持，否则不支持
autoval	若支持自动调节，是否开启自动调节 为1则开启，否则不开启

CHD_WMP_Video_GetParam

【描述】

获取视频设置参数（格式、分辨率、帧率）。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetParam(IN CHD_WMP_T handle,
                                   OUT CONST CHD_VIDEO_PARAM_S * pstParam);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pstParam	指向视频设置参数结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、视频设置参数由摄像头决定，与设备无关。

【参数相关】

<pre>typedef struct{ CHD_U32 format; CHD_U32 resolu; CHD_U32 fps; CHD_U32 maxfps; }CHD_VIDEO_PARAM_S;</pre>	
成员名称	描述
format	视频格式
resolu	视频分辨率
fps	视频帧率
maxfps	当前格式和分辨率下支持设置的最大帧率

CHD_WMP_Video_GetFormat

【描述】

获取视频采集格式。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetFormat(IN CHD_WMP_T handle,
                                   OUT CHD_U32 * pFmt);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pFmt	指向采集格式的整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、视频格式由摄像头来决定。

【参数相关】

<pre>typedef enum{ CHD_FMT_YUYV = 0x1, CHD_FMT_MJPEG, CHD_FMT_H264 }CHD_VIDEO_FMT_E;</pre>	
成员名称	描述
CHD_FMT_YUYV	YUYV 数据
CHD_FMT_MJPEG	MJPEG 数据
CHD_FMT_H264	H264 数据

CHD_WMP_Video_GetResolu

【描述】

获取视频采集分辨率。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetResolu (IN CHD_WMP_T handle,  
                                     OUT CHD_U32* pResolu);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pResolu	指向视频采集分辨率的整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、视频采集分辨率由摄像头来决定。

【参数相关】

分辨率、图像宽高转换函数：

```
void CHD_Video_AnalysisResolu(int resolu,  
                              unsigned int * w, unsigned int * h);
```

CHD_WMP_Video_GetFPS

【描述】

获取视频采集帧率。

【语法】

```
CHD_RET_E CHD_WMP_Video_GetFPS(IN CHD_WMP_T handle,
                                OUT CHD_U32 * pFPS);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pFPS	指向视频采集帧率的整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、视频采集分辨率由摄像头来决定。

【参数相关】

无

CHD_WMP_Video_SetVideoCtrl

【描述】

设置摄像头单个控制参数(如亮度、白平衡、曝光值等)。

【语法】

```
CHD_RET_E CHD_WMP_Video_SetVideoCtrl(IN CHD_WMP_T handle,
                                       IN CHD_VCTRL_E type, IN CONST CHD_VIDEO_CTRL_S * pCtrl);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
type	摄像头控制参数类型	输入
pstCtrl	指向存放摄像头控制参数的指针	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

- 1、摄像头控制参数由摄像头而决定，于设备无关。

【参数相关】

```
typedef enum{
    CHD_VCTRL_BRIGHTNESS = 0x0,
    CHD_VCTRL_CONTRAST,
    CHD_VCTRL_SATURATION,
    CHD_VCTRL_HUE,
    CHD_VCTRL_WHITE_BALANCE,
    CHD_VCTRL_GAMMA,
    CHD_VCTRL_GAIN,
    CHD_VCTRL_SHARPNESS,
    CHD_VCTRL_BACKLIGHT,
    CHD_VCTRL_EXPOSURE,
    CHD_VCTRL_MAX,
}CHD_VCTRL_E;
```

成员名称	描述
CHD_VCTRL_BRIGHTNESS	亮度
CHD_VCTRL_CONTRAST	对比度
CHD_VCTRL_SATURATION	饱和度
CHD_VCTRL_HUE	色调
CHD_VCTRL_WHITE_BALANCE	白平衡
CHD_VCTRL_GAMMA	伽马
CHD_VCTRL_GAIN	增益
CHD_VCTRL_SHARPNESS	清晰度
CHD_VCTRL_BACKLIGHT	背光补偿
CHD_VCTRL_EXPOSURE	曝光值

```
typedef struct
{
    CHD_S32 val_valid; // 值为1时，后面五个参数才有效
    CHD_S32 minval;
    CHD_S32 curval;
    CHD_S32 maxval;
    CHD_S32 stepval;
    CHD_S32 defval;

    CHD_S32 auto_valid;
    CHD_S32 autoval;
}CHD_VIDEO_CTRL_S;
```

成员名称	描述
val_valid	设备是否支持type指定的控制参数类型 如：设备是否支持曝光值调节 为1则支持，否则不支持
minval	控制参数最小值
curval	控制参数当前值
maxval	控制参数最大值
stepval	调节步长
defval	控制参数默认值

auto_valid	设备是否支持自动调节 为 1 则支持，否则不支持
autoval	若支持自动调节，是否开启自动调节 为 1 则开启，否则不开启

CHD_WMP_Video_ResetVCtrl

【描述】

所有摄像头控制参数恢复默认参数。

【语法】

```
CHD_RET_E CHD_WMP_Video_ResetVCtrl(IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Video_SetFormat

【描述】

设置视频格式。

【语法】

```
CHD_RET_E CHD_WMP_Video_SetFormat(IN CHD_WMP_T handle,  
                                     IN CHD_VIDEO_FMT_E Fmt);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Fmt	视频格式	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_FMT_YUV = 0x1, CHD_FMT_MJPEG, CHD_FMT_H264 }CHD_VIDEO_FMT_E;</pre>	
成员名称	描述
CHD_FMT_YUV	YUV 数据
CHD_FMT_MJPEG	MJPEG 数据
CHD_FMT_H264	H264 数据

CHD_WMP_Video_SetResolu

【描述】

设置视频采集分辨率。

【语法】

```
CHD_RET_E CHD_WMP_Video_SetResolu (IN CHD_WMP_T handle,
                                     IN CHD_U32 Resolu);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Resolu	视频采集分辨率	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

图像宽高—>分辨率转换函数：

```
int CHD_Video_GroupResolu(const unsigned int width,
                           const unsigned int height);
```

CHD_WMP_Video_SetFPS

【描述】

设置视频采集帧率。

【语法】

```
CHD_RET_E CHD_WMP_Video_SetFPS(IN CHD_WMP_T handle,  
                                IN CHD_U32 FPS);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
FPS	视频采集帧率	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

4. 串口操作

4.1 概述

网络串口：将串口网络化，串口的配置、数据通信都与本地操作无异。

4.2 API参考

串口数据相关API：

- [CHD_WMP_Serial_Begin](#) : 开启串口数据流传输
- [CHD_WMP_Serial_End](#) : 停止串口数据流传输
- [CHD_WMP_Serial_SendData](#) : 发送串口数据
- [CHD_WMP_Serial_RequestData](#) : 获取接收到的串口数据
- [CHD_WMP_Serial_ReleaseData](#) : 释放串口数据缓存

- [CHD_WMP_Serial_GetCurRxCacheSize](#) : 获取当前接收流中数据个数
- [CHD_WMP_Serial_GetRxTotalNum](#) : 获取串口接收数据总数
- [CHD_WMP_Serial_GetTxTotalNum](#) : 获取串口发送数据总数

串口配置相关API:

- [CHD_WMP_Serial_GetParam](#) : 获取串口设备参数信息
- [CHD_WMP_Serial_GetSpeed](#) : 获取串口波特率
- [CHD_WMP_Serial_GetDataBit](#) : 获取串口数据位
- [CHD_WMP_Serial_GetStopBit](#) : 获取串口停止位
- [CHD_WMP_Serial_GetParity](#) : 获取串口校验位
- [CHD_WMP_Serial_GetTimeout](#) : 获取串口自动发送超时时间
- [CHD_WMP_Serial_SetSpeed](#) : 设置串口波特率
- [CHD_WMP_Serial_SetDataBit](#) : 设置串口数据位
- [CHD_WMP_Serial_SetStopBit](#) : 设置串口停止位
- [CHD_WMP_Serial_SetParity](#) : 设置串口校验位
- [CHD_WMP_Serial_SetTimeout](#) : 设置串口自动发送超时时间

编译相关:

头文件 : CHD_WMP_Type.h、CHD_WMP_Return.h、CHD_WMP_Serial.h
Windows库 : chd_base.dll、chd_efast.dll、chd_wmp.dll
Linux库 : chd_base.so、chd_efast.so、chd_wmp.so
IOS库 : chd_base.a、chd_efast.a、chd_wmp.a
Mac库 : chd_base.dylib、chd_efast.dylib、chd_wmp.dylib

CHD_WMP_Serial_Begin

【描述】

开启串口数据流传输。

【语法】

```
CHD_RET_E CHD_WMP_Serial_Begin(IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Serial_End

【描述】

关闭数据流传输。

【语法】

```
CHD_RET_E CHD_WMP_Serial_End(IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Serial_SendData

【描述】

发送串口数据。

【语法】

```
CHD_RET_E CHD_WMP_Serial_SendData(IN CHD_WMP_T handle,  
                                     OUT CONST CHD_VOID * pdata, OUT CHD_U32 datalen);
```


【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pdata	指向发送数据的指针	输入
datalen	要发送数据的长度	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Serial_RequestData

【描述】

获取设备发送过来的串口数据。

【语法】

```
CHD_RET_E CHD_WMP_Serial_RequestData(IN CHD_WMP_T handle,  
                                       OUT CHD_VOID ** pdata, OUT CHD_U32 * datalen);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pdata	指向串口数据的指针	输出
datalen	指向串口数据长度的整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

- 1、必须配合CHD_WMP_Poll()函数使用，调用CHD_WMP_Poll()函数后网络才会接受设备传输过来的串口数据。
- 2、接收缓存为动态开辟，使用完成必须调用CHD_WMP_Serial_ReleaseData()函数释放图片缓存，否则会造成内存泄露。

【参数相关】

无

CHD_WMP_Serial_ReleaseData

【描述】

释放串口接收缓存。

【语法】

```
CHD_RET_E CHD_WMP_Serial_ReleaseData(IN CHD_WMP_T handle,  
                                       IN CONST CHD_VOID * pdata);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pdata	指向串口接收缓存数据的指针	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Serial_GetCurRxCacheSize

【描述】

获取当前串口接收缓存字节数。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetCurRxCacheSize(  
    IN CHD_WMP_T handle, OUT CHD_U32 * pcnt);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pcnt	指向保存缓存字节数的整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Serial_GetRxTotalNum

【描述】

获取当前串口接收数据总字节数。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetRxTotalNum(  
    IN CHD_WMP_T handle, OUT CHD_U32 * pcnt);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pcnt	指向接收缓存总字节数的整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Serial_GetTxTotalNum

【描述】

获取当前串口接收数据总字节数。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetTxTotalNum(  
    IN CHD_WMP_T handle, OUT CHD_U32 * pcnt);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pcnt	指向发送缓存总字节数的整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Serial_GetParam

【描述】

获取串口配置参数信息。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetParam (IN CHD_WMP_T handle,  
                                     OUT CHD_SERIAL_PARAM_S * pparam);
```

【参数】

参数名称	描述	输入/输出
handler	串口模块句柄	输入
pparam	串口配置参数结构体指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef struct{ CHD_SERIAL_SPEED_E speed; CHD_SERIAL_DATABIT_E databit; CHD_SERIAL_STOPBIT_E stopbit; CHD_SERIAL_PARITY_E parity; CHD_U32 timeout; }CHD_SERIAL_PARAM_S;</pre>	
成员名称	描述
speed	波特率
databit	数据位
stopbit	停止位
parity	奇偶校验位
timeout	串口自动发送超时时间

CHD_WMP_Serial_GetSpeed

【描述】

获取串口波特率。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetSpeed (IN CHD_WMP_T handle,
                                     OUT CHD_SERIAL_SPEED_E * pspeed);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pspeed	指向波特率枚举的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_SERIAL_BS300 = 300, CHD_SERIAL_BS1200 = 1200, CHD_SERIAL_BS2400 = 2400, CHD_SERIAL_BS4800 = 4800, CHD_SERIAL_BS9600 = 9600, CHD_SERIAL_BS19200 = 19200, CHD_SERIAL_BS38400 = 38400, CHD_SERIAL_BS57600 = 57600, CHD_SERIAL_BS115200 = 115200, CHD_SERIAL_BS230400 = 230400, }CHD_SERIAL_SPEED_E;</pre>	
成员名称	描述
CHD_SERIAL_BS300	每秒传输 300bit数据
CHD_SERIAL_BS1200	每秒传输 1200bit数据
CHD_SERIAL_BS 2400	每秒传输 2400bit数据
CHD_SERIAL_BS4800	每秒传输 4800bit数据
CHD_SERIAL_BS9600	每秒传输 9600bit数据
CHD_SERIAL_BS19200	每秒传输 19200bit数据

CHD_SERIAL_BS38400	每秒传输 38400bit数据
CHD_SERIAL_BS57600	每秒传输 57600bit数据
CHD_SERIAL_BS115200	每秒传输 115200bit数据
CHD_SERIAL_BS230400	每秒传输 230400bit数据

CHD_WMP_Serial_GetDataBit

【描述】

获取串口数据位。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetDataBit(IN CHD_WMP_T handle,  
                                     OUT CHD_SERIAL_DATABIT_E * pdatabit);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pdatabit	指向数据位枚举的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_SERIAL_DATABIT_7 = 7, CHD_SERIAL_DATABIT_8 = 8, }CHD_SERIAL_DATABIT_E;</pre>	
成员名称	描述
CHD_SERIAL_DATABIT_7	通信中实际数据位 为 7 bit
CHD_SERIAL_DATABIT_8	通信中实际数据位 为 8 bit

CHD_WMP_Serial_GetStopBit

【描述】

获取串口停止位。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetStopBit(IN CHD_WMP_T handle,  
OUT CHD_SERIAL_STOPBIT_E * pstopbit);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pstopbit	指向停止位枚举的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_SERIAL_STOPBIT_1 = 1, CHD_SERIAL_STOPBIT_0 = 0, }CHD_SERIAL_STOPBIT_E;</pre>	
成员名称	描述
CHD_SERIAL_STOPBIT_1	停止位为 1 bit
CHD_SERIAL_STOPBIT_0	停止位为 0 bit

CHD_WMP_Serial_GetParity

【描述】

获取串口奇偶校验位。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetParity(IN CHD_WMP_T handle,
                                     OUT CHD_SERIAL_PARITY_E * pparity);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pparity	指向奇偶校验位枚举的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

```
typedef enum{
    CHD_SERIAL_PARITY_EVEN    = 69,
    CHD_SERIAL_PARITY_NONE    = 78,
    CHD_SERIAL_PARITY_ODD     = 79,
    CHD_SERIAL_PARITY_SPACE   = 83,
}CHD_SERIAL_PARITY_E;
```

成员名称	描述
CHD_SERIAL_PARITY_NONE	无校验
CHD_SERIAL_PARITY_ODD	奇校验
CHD_SERIAL_PARITY_EVEN	偶校验
CHD_SERIAL_PARITY_SPACE	SPACE校验

CHD_WMP_Serial_GetTimeout

【描述】

获取串口自动发送超时时间。

【语法】

```
CHD_RET_E CHD_WMP_Serial_GetTimeout(IN CHD_WMP_T handle,
                                     OUT CHD_U32 * ptimeout);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
ptimeout	整形指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Serial_SetSpeed

【描述】

设置串口波特率。

【语法】

```
CHD_RET_E CHD_WMP_Serial_SetSpeed (IN CHD_WMP_T handle,  
                                     IN CHD_SERIAL_SPEED_E speed);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
speed	串口波特率值	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

typedef enum{	
CHD_SERIAL_BS300	= 300,
CHD_SERIAL_BS1200	= 1200,
CHD_SERIAL_BS2400	= 2400,
CHD_SERIAL_BS4800	= 4800,
CHD_SERIAL_BS9600	= 9600,
CHD_SERIAL_BS19200	= 19200,

<pre> CHD_SERIAL_BS38400 = 38400, CHD_SERIAL_BS57600 = 57600, CHD_SERIAL_BS115200 = 115200, CHD_SERIAL_BS230400 = 230400, }CHD_SERIAL_SPEED_E; </pre>	
成员名称	描述
CHD_SERIAL_BS300	每秒传输 300bit数据
CHD_SERIAL_BS1200	每秒传输 1200bit数据
CHD_SERIAL_BS 2400	每秒传输 2400bit数据
CHD_SERIAL_BS4800	每秒传输 4800bit数据
CHD_SERIAL_BS9600	每秒传输 9600bit数据
CHD_SERIAL_BS19200	每秒传输 19200bit数据
CHD_SERIAL_BS38400	每秒传输 38400bit数据
CHD_SERIAL_BS57600	每秒传输 57600bit数据
CHD_SERIAL_BS115200	每秒传输 115200bit数据
CHD_SERIAL_BS230400	每秒传输 230400bit数据

CHD_WMP_Serial_SetDataBit

【描述】

设置串口数据位。

【语法】

```

CHD_RET_E CHD_WMP_Serial_SetDataBit(IN CHD_WMP_T handle,
                                     IN CHD_SERIAL_DATABIT_E databit);

```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
databit	数据位	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_SERIAL_DATABIT_7 = 7, CHD_SERIAL_DATABIT_8 = 8, }CHD_SERIAL_DATABIT_E;</pre>	
成员名称	描述
CHD_SERIAL_DATABIT_7	通信中实际数据位 为 7 bit
CHD_SERIAL_DATABIT_8	通信中实际数据位 为 8 bit

CHD_WMP_Serial_SetStopBit

【描述】

设置串口停止位。

【语法】

```
CHD_RET_E CHD_WMP_Serial_SetStopBit (  
    IN CHD_WMP_T handle, IN CHD_SERIAL_STOPBIT_E stopbit);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
stopbit	停止位	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_SERIAL_STOPBIT_1 = 1, CHD_SERIAL_STOPBIT_0 = 0, }CHD_SERIAL_STOPBIT_E;</pre>	
成员名称	描述
CHD_SERIAL_STOPBIT_1	停止位为 1 bit
CHD_SERIAL_STOPBIT_0	停止位为 0 bit

CHD_WMP_Serial_SetParity

【描述】

设置串口奇偶校验位。

【语法】

```
CHD_RET_E CHD_WMP_Serial_SetParity(IN CHD_WMP_T handle,  
IN CHD_SERIAL_PARITY_E parity);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
parity	奇偶校验位	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

```
typedef enum{
    CHD_SERIAL_PARITY_EVEN    = 69,
    CHD_SERIAL_PARITY_NONE    = 78,
    CHD_SERIAL_PARITY_ODD     = 79,
    CHD_SERIAL_PARITY_SPACE   = 83,
}CHD_SERIAL_PARITY_E;
```

成员名称	描述
CHD_SERIAL_PARITY_NONE	无校验
CHD_SERIAL_PARITY_ODD	奇校验
CHD_SERIAL_PARITY_EVEN	偶校验
CHD_SERIAL_PARITY_SPACE	SPACE校验

CHD_WMP_Serial_SetTimeout

【描述】

设置串口自动发送超时时间。

【语法】

```
CHD_RET_E CHD_WMP_Serial_SetTimeout(IN CHD_WMP_T handle,
IN CHD_U32  timeout);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
timeout	超时时间	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

5. 音频操作

5.1 概述

用户只需调用函数接口即可获取到音频参数及实时音频原始数据功能。

5.2 API参考

音频数据相关API:

- [CHD_WMP_Audio_Begin](#) : 开启视频流
- [CHD_WMP_Audio_End](#) : 停止视频流
- [CHD_WMP_Audio_RequestData](#) : 获取音频数据
- [CHD_WMP_Audio_ReleaseData](#) : 释放音频数据缓存

音频参数相关API:

- [CHD_WMP_Audio_GetSampleFreq](#) : 获取采样频率
- [CHD_WMP_Audio_GetADBits](#) : 获取AD量化精度
- [CHD_WMP_Audio_GetChanner](#) : 获取音频通道
- [CHD_WMP_Audio_GetParam](#) : 获取参数

- [CHD_WMP_Audio_SetSampleFreq](#) : 设置采样频率
- [CHD_WMP_Audio_SetADBits](#) : 设置AD量化精度
- [CHD_WMP_Audio_SetChannel](#) : 设置音频通道
- [CHD_WMP_Audio_SetParam](#) : 设置参数

编译相关:

- 头文件 : [CHD_WMP_Type.h](#)、[CHD_WMP_Return.h](#)、[CHD_WMP_Audio.h](#)
- Windows库 : [chd_base.dll](#)、[chd_efast.dll](#)、[chd_wmp.dll](#)
- Linux库 : [chd_base.so](#)、[chd_efast.so](#)、[chd_wmp.so](#)
- IOS库 : [chd_base.a](#)、[chd_efast.a](#)、[chd_wmp.a](#)
- Mac库 : [chd_base.dylib](#)、[chd_efast.dylib](#)、[chd_wmp.dylib](#)

CHD_WMP_Audio_Begin

【描述】

开启音频流。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_Begin
    (IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_Audio_End

【描述】

关闭音频流。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_End (IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

无

CHD_WMP_Audio_RequestData

【描述】

获取音频数据。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_RequestData  
(IN CHD_WMP_T handle, OUT CHD_AUDIO_FRAME_S ** pstData)
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
PstData	音频数据指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

```
typedef struct{

    CHD_AUDIO_SAMPLE_E    esample;
    CHD_AUDIO_CHN_E        echn;
    CHD_AUDIO_ADBITS_E    eadbits;

    CHD_TIMESTAMP_T        timestamp;
    CHD_U32                datalen;
    CHD_S8                 pData[0];
}CHD_AUDIO_FRAME_S
```

成员名称	描述
esample	采样频率
echn	通道号
eadbits	量化精度
timestamp	时间戳
datalen	音频数据长度
pData[0]	音频数据指针

CHD_WMP_Audio_ReleaseData

【描述】

释放音频数据缓存。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_ReleaseData
    (IN CHD_WMP_T handle, IN CHD_AUDIO_FRAME_S * pstData);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
PstData	指向音频数据的指针	输出

【返回值】

返回值	描述
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SUCCESS	成功

【注意】

无

【参数相关】

无

CHD_WMP_Audio_GetSampleFreq

【描述】

获取采样频率。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_GetSampleFreq  
(IN CHD_WMP_T handle, OUT CHD_AUDIO_SAMPLE_E * peSampleFreq);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
PeSampleFreq	指向采样频率结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功

CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_AUDIO_SAMPLE_8000 = 8000, CHD_AUDIO_SAMPLE_44100 = 44100, CHD_AUDIO_SAMPLE_48000 = 48000 }CHD_AUDIO_SAMPLE_E;</pre>	
成员名称	描述
CHD_AUDIO_SAMPLE_8000	采样频率为8000Hz
CHD_AUDIO_SAMPLE_44100	采样频率为44100Hz
CHD_AUDIO_SAMPLE_48000	采样频率为48000Hz

CHD_WMP_Audio_GetADBits

【描述】

获取 AD 量化精度。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_SetADBits
(
    IN CHD_WMP_T    handle,
    OUT CHD_AUDIO_ADBITS_E peADBits);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
peADBits	指向AD量化精度的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_AUDIO_ADBITS_8 = 8, CHD_AUDIO_ADBITS_16 = 16, }CHD_AUDIO_ADBITS_E;</pre>	
成员名称	描述
CHD_AUDIO_ADBITS_8	AD量化精度为8位
CHD_AUDIO_ADBITS_16	AD量化精度为16位

CHD_WMP_Audio_GetChannl

【描述】

获取音频通道。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_GetChannel  
    (IN CHD_WMP_T    handle, OUT CHD_AUDIO_CHN_E * peChn);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
peChn	指向音频通道的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

<pre>typedef enum{ CHD_AUDIO_CHN_SIGNAL = 1, CHD_AUDIO_CHN_STEREO = 2, }CHD_AUDIO_CHN_E;</pre>	
成员名称	描述
CHD_AUDIO_CHN_SIGNAL	音频通道为单通道
CHD_AUDIO_CHN_STEREO	音频通道为立体声

CHD_WMP_Audio_GetParam

【描述】

获取相关参数。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_GetParam
(
    IN CHD_WMP_T handle,
    OUT CONST
    CHD_AUDIO_PARAM_S * pstParam);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
ps tParam	指向参数的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功

CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

无

【参数相关】

<pre>typedef struct { CHD_AUDIO_SAMPLE_E esample; CHD_AUDIO_CHN_E echn; CHD_AUDIO_ADBITS_E eadbits; }CHD_AUDIO_PARAM_S;</pre>	
成员名称	描述
esample	采样的频率
echn	通道数
eadbits	AD量化精度

CHD_WMP_Audio_SetSampleFreq

【描述】

设置采样频率。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_SetSampleFreq  
    (IN CHD_WMP_T handle, IN CHD_AUDIO_SAMPLE_E peSampleFreq);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
PeSampleFreq	指向采样频率结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

目前暂不支持修改参数。

【参数相关】

```
typedef enum{  
    CHD_AUDIO_SAMPLE_8000    = 8000,  
    CHD_AUDIO_SAMPLE_44100   = 44100,  
    CHD_AUDIO_SAMPLE_48000   = 48000  
}CHD_AUDIO_SAMPLE_E;
```

成员名称	描述
CHD_AUDIO_SAMPLE_8000	设置采样频率为8000Hz
CHD_AUDIO_SAMPLE_44100	设置采样频率为44100Hz
CHD_AUDIO_SAMPLE_48000	设置采样频率为48000Hz

CHD_WMP_Audio_SetADBits

【描述】

设置 AD 量化精度。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_SetADBits
(
    IN CHD_WMP_T    handle,
    IN CHD_AUDIO_ADBITS_E peADBits);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
peADBits	指向量化精度结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

目前暂不支持修改参数。

【参数相关】

<pre>typedef enum{ CHD_AUDIO_ADBITS_8 = 8, CHD_AUDIO_ADBITS_16 = 16, }CHD_AUDIO_ADBITS_E;</pre>	
成员名称	描述
CHD_AUDIO_ADBITS_8	设置AD量化精度为8位
CHD_AUDIO_ADBITS_16	设置AD量化精度为16位

CHD_WMP_Audio_SetChannl

【描述】

设置音频通道。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_SetChannel  
(IN CHD_WMP_T handle, IN CHD_AUDIO_CHN_E * peChn);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
peChn	指向音频通道的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

目前暂不支持修改参数。

【参数相关】

<pre>typedef enum{ CHD_AUDIO_CHN_SIGNAL = 1, CHD_AUDIO_CHN_STEREO = 2, }CHD_AUDIO_CHN_E;</pre>	
成员名称	描述
CHD_AUDIO_CHN_SIGNAL	设置音频通道为单通道
CHD_AUDIO_CHN_STEREO	设置音频通道为立体声

CHD_WMP_Audio_SetParam

【描述】

设置相关参数。

【语法】

```
CHD_WMP_API CHD_RET_E CHD_WMP_Audio_SetParam  
(IN CHD_WMP_T handle, IN CONST CHD_AUDIO_PARAM_S * pstParam);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
ps tParam	指向参数的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

目前暂不支持修改参数。

【参数相关】

```
typedef struct {  
    CHD_AUDIO_SAMPLE_E esample;  
    CHD_AUDIO_CHN_E     echn;  
    CHD_AUDIO_ADBITS_E eadbits;  
}CHD_AUDIO_PARAM_S;
```

成员名称	描述
esample	采样的频率
echn	通道数
eadbits	AD量化精度

6. GPIO操作

6.1 概述

通过网络获取和设置设备GPIO的状态。

6.2 API参考

串口数据相关API：

- `CHD_WMP_Gpio_GetStatus` : 获取GPIO状态
- `CHD_WMP_Gpio_SetStatus` : 设置GPIO状态

编译相关：

头文件 : `CHD_WMP_Type.h`、`CHD_WMP_Return.h`、`CHD_WMP_Gpio.h`
Windows库 : `chd_base.dll`、`chd_efast.dll`、`chd_wmp.dll`
Linux库 : `chd_base.so`、`chd_efast.so`、`chd_wmp.so`
IOS库 : `chd_base.a`、`chd_efast.a`、`chd_wmp.a`
Mac库 : `chd_base.dylib`、`chd_efast.dylib`、`chd_wmp.dylib`

CHD_WMP_Gpio_GetStatus

【描述】

获取设备相应GPIO口的状态。

【语法】

```
CHD_RET_E CHD_WMP_Gpio_GetStatus(IN CHD_WMP_T handle,  
                                   IN CHD_GPIO_E gpio, OUT CHD_GPIO_STATE_E * estate);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
gpio	GPIO口编号	输入
estate	指向GPIO口状态枚举的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef enum{ GPIO1 = 0x0, GPIO2, GPIO3, GPIO4, GPIO5, GPIO_MAX, }CHD_GPIO_E;</pre>	
成员名称	描述
GPIO1	编号为1的GPIO口
GPIO2	编号为2的GPIO口
GPIO3	编号为3的GPIO口
GPIO4	编号为4的GPIO口
GPIO5	编号为5的GPIO口
<pre>typedef enum{ CHD_GPIO_STATE_LOW, CHD_GPIO_STATE_HIGH = 1, }CHD_GPIO_STATE_E;</pre>	
成员名称	描述
CHD_GPIO_STATE_LOW	低电平
CHD_GPIO_STATE_HIGH	高电平

CHD_WMP_Gpio_SetStatus

【描述】

设置设备相应GPIO口的状态。

【语法】

```
CHD_RET_E CHD_WMP_Gpio_SetStatus(IN CHD_WMP_T handle,
                                   IN CHD_GPIO_E gpio, IN CHD_GPIO_STATE_E estate);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
gpio	GPIO口编号	输入
estate	GPIO口状态	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

```
typedef enum{
    GPIO1 = 0x0,
    GPIO2,
    GPIO3,
    GPIO4,
    GPIO5,
    GPIO_MAX,
}CHD_GPIO_E;
```

成员名称	描述
GPIO1	编号为1的GPIO口
GPIO2	编号为2的GPIO口
GPIO3	编号为3的GPIO口
GPIO4	编号为4的GPIO口
GPIO5	编号为5的GPIO口

```
typedef enum{
    CHD_GPIO_STATE_LOW,
    CHD_GPIO_STATE_HIGH = 1,
}CHD_GPIO_STATE_E;
```

成员名称	描述
CHD_GPIO_STATE_LOW	低电平
CHD_GPIO_STATE_HIGH	高电平

7. I2C 操作

7.1 概述

将本地的I2C操作设置为网络I2C，用户只需调用相应的接口函数即可操作I2C设备。

7.2 API参考

- `CHD_WMP_I2C_GetValue` : 读取I2C
- `CHD_WMP_I2C_SetValue` : 写入I2C

编译相关:

头文件 : `CHD_WMP_Type.h`、`CHD_WMP_Return.h`、`CHD_WMP_I2C.h`
Windows库 : `chd_base.dll`、`chd_efast.dll`、`chd_wmp.dll`
Linux库 : `chd_base.so`、`chd_efast.so`、`chd_wmp.so`
IOS库 : `chd_base.a`、`chd_efast.a`、`chd_wmp.a`
Mac库 : `chd_base.dylib`、`chd_efast.dylib`、`chd_wmp.dylib`

CHD_WMP_I2C_GetValue

【描述】

读取I2C。

【语法】

```
CHD_RET_E CHD_WMP_I2C_GetValue(IN CHD_WMP_T handle,  
                                IN CHD_IIC_DATA_S * pdata);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pdata	指向I2C结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef struct { unsigned char address; unsigned char subaddress; unsigned char readmode; unsigned char datasize; unsigned char data[128]; }CHD_IIC_DATA_S;</pre>	
成员名称	描述
address	i2c设备地址
subaddress	i2c设备子地址
readmode	i2c读数据时，选择有无停止位: 0(无停止位) 1(有停止位，默认)，在读数据时，需要先写入设备地址、子地址、读标志，该成员控制是否在写入这三个数据之后有停止位发送
datasize	读写数据大小
data[128]	写入/读取数据的缓冲区

CHD_WMP_I2C_SetValue

【描述】

写入I2C。

【语法】

```
CHD_RET_E CHD_WMP_I2C_SetValue(IN CHD_WMP_T handle,  
                                IN CHD_IIC_DATA_S * pdata);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
pdata	指向I2C结构体的指针	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef struct { unsigned char address; unsigned char subaddress; unsigned char readmode; unsigned char datasize; unsigned char data[128]; }CHD_IIC_DATA_S;</pre>	
成员名称	描述
address	i2c设备地址
subaddress	i2c设备子地址

readmode	i2c读数据时，选择有无停止位: 0(无停止位) 1(有停止位，默认)，在读数据时，需要先写入设备地址、子地址、读标志，该成员控制是否在写入这三个数据之后有停止位发送
datasize	读写数据大小
data[128]	写入/读取数据的缓冲区

8. 高级设置

8.1 概述

设备高级设置包括获取加密信息、设备信息、设置视频串口数据传输模式等。

8.2 API参考

串口数据相关API：

- [CHD_WMP_GetEncrypt](#) : 获取加密信息
- [CHD_WMP_GetDeviceInfo](#) : 获取设备信息
- [CHD_WMP_GetMac](#) : 获取设备MAC地址
- [CHD_WMP_GetSystemTime](#) : 获取设备当前系统时间
- [CHD_WMP_SetTransMode](#) : 设置数据传输模式
- [CHD_WMP_SetSystemTime](#) : 设置设备系统时间

编译相关：

头文件	: CHD_WMP_Type.h 、 CHD_WMP_Return.h 、 CHD_WMP.h
Windows库	: chd_base.dll 、 chd_efast.dll 、 chd_wmp.dll
Linux库	: chd_base.so 、 chd_efast.so 、 chd_wmp.so
IOS库	: chd_base.a 、 chd_efast.a 、 chd_wmp.a
Mac库	: chd_base.dylib 、 chd_efast.dylib 、 chd_wmp.dylib

CHD_WMP_GetEncrypt

【描述】

获取加密信息，本加密信息可由用户提供，直接烧录到固件中，防止竞争对手恶意抄袭，山寨，增加设备和软件的安全性。

【语法】

```
CHD_RET_E CHD_WMP_GetEncrypt(IN CHD_WMP_T handle,  
                              OUT CHD_S8 * Encrypt);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Encrypt	指向存储加密信息空间的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

无

CHD_WMP_GetDeviceInfo

【描述】

获取设备信息（ID号、设备别名、设备IP地址）。

【语法】

```
CHD_RET_E CHD_WMP_GetDeviceInfo(IN CHD_WMP_T handle,  
                                OUT CHD_DEVINFO_S * dev);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
dev	指向设备信息结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>#define CHD_ALIAS_LEN 20 #define CHD_IPADDR_LEN 20 typedef struct{ CHD_U32 Id; CHD_S8 Alias[CHD_ALIAS_LEN]; CHD_S8 IpAddr[CHD_IPADDR_LEN]; }CHD_DEVINFO_S;</pre>	
成员名称	描述

Id	设备ID号
Alias	设备别名
IpAddr	设备IP地址

CHD_WMP_GetMac

【描述】

获取设备MAC地址。

【语法】

```
CHD_RET_E CHD_WMP_GetMac(IN CHD_WMP_T handle,
                          OUT CHD_MAC_S * Mac);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Mac	指向MAC地址结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

```
typedef struct{
    char Mac[6];
}CHD_MAC_S;
```

成员名称	描述
Mac	Mac地址

CHD_WMP_GetSystemTime

【描述】

获取设备当前系统时间。

【语法】

```
CHD_RET_E CHD_WMP_GetSystemTime(IN CHD_WMP_T handle,
                                OUT CHD_TIME_S * stime);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
stime	指向系统时间信息结构体的指针	输出

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误
CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

```
typedef struct{
    CHD_U16  year;
    CHD_U8   month;
```

<pre> CHD_U8 mday; CHD_U8 wday; CHD_U8 hour; CHD_U8 min; CHD_U8 sec; }CHD_TIME_S; </pre>	
成员名称	描述
year	年
month	月
mday	日
wday	星期
hour	时
min	分
sec	秒

CHD_WMP_SetTransMode

【描述】

设置视频、音频、串口数据传输模式。

【语法】

```

CHD_RET_E CHD_WMP_SetTransMode(IN CHD_WMP_T handle,
                                IN CHD_TRANSMODE_E emode);

```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
emode	传输模式	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

1、EFAST传输模式暂时未对外开放。

【参数相关】

<pre>typedef enum{ CHD_TRANSMODE_TCP, CHD_TRANSMODE_EFAST, }CHD_TRANSMODE_E;</pre>	
成员名称	描述
CHD_TRANSMODE_TCP	TCP传输
CHD_TRANSMODE_EFAST	稳定可靠性UDP传输

CHD_WMP_GetSystemTime

【描述】

获取设备当前系统时间。

【语法】

```
CHD_RET_E CHD_WMP_SetSystemTime(IN CHD_WMP_T handle,
                                IN CONST CHD_TIME_S * stime);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
stime	指向系统时间信息结构体的指针	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

CHD_RET_SESSIONID_ERROR	会话错误
CHD_RET_CONNECT_ERROR	网络连接错误，无法连接设备
CHD_RET_RECV_ERROR	网络错误，网络环境较差，传输不稳定
CHD_RET_SEND_ERROR	网络错误，网络环境较差，传输不稳定

【注意】

无

【参数相关】

<pre>typedef struct{ CHD_U16 year; CHD_U8 month; CHD_U8 mday; CHD_U8 wday; CHD_U8 hour; CHD_U8 min; CHD_U8 sec; }CHD_TIME_S;</pre>	
成员名称	描述
year	年
month	月
mday	日
wday	星期
hour	时
min	分
sec	秒

9. AVI录像

9.1 概述

将连续的JPEG图片流录制AVI格式的视频文件。

9.2 API参考

- [CHD_WMP_Mjpeg2Avi_Begin](#) : 开始视频录制
- [CHD_WMP_Mjpeg2Avi_PutData](#) : 将图片打包到视频文件中
- [CHD_WMP_Mjpeg2Avi_SetParam](#) : 设置视频文件参数
- [CHD_WMP_Mjpeg2Avi_End](#) : 完成录制

编译相关:

头文件 : [CHD_WMP_Type.h](#)、[CHD_WMP_Return.h](#)、[CHD_WMP_Code.h](#)
Windows库 : [chd_base.dll](#)、[chd_efast.dll](#)、[chd_wmp.dll](#)
Linux库 : [chd_base.so](#)、[chd_efast.so](#)、[chd_wmp.so](#)
IOS库 : [chd_base.a](#)、[chd_efast.a](#)、[chd_wmp.a](#)
Mac库 : [chd_base.dylib](#)、[chd_efast.dylib](#)、[chd_wmp.dylib](#)

CHD_WMP_Mjpeg2Avi_Begin

【描述】

开始视频录制AVI视频文件。

【语法】

```
CHD_RET_E CHD_WMP_Mjpeg2Avi_Begin(IN CHD_WMP_T handle,  
                                     INCHD_S8 * filename);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
filename	视频文件路径+文件名	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

- 1、第二个参数必须是视频文件存放路径+文件名。
- 2、视频存放路径必须具有可写权限。

【参数相关】

无

CHD_WMP_Mjpeg2Avi_PutData

【描述】

将图片打包到视频文件中。

【语法】

```
CHD_RET_E CHD_WMP_Mjpeg2Avi_PutData(IN CHD_WMP_T handle,  
                                       IN CHD_S8* Dest, IN CHD_U32 datalen);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
Dest	指向图片缓存的指针	输入
datalen	图片数据的长度	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

1、传入的数据必须为MJPEG格式的图片数据。

【参数相关】

无

CHD_WMP_Mjpeg2Avi_SetParam

【描述】

设置视频文件参数。

【语法】

```
CHD_RET_E CHD_WMP_Mjpeg2Avi_SetParam(IN CHD_WMP_T handle,  
                                         IN CHD_U32 width, IN CHD_U32 height, IN CHD_U32 fps);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入
width	视频图像宽度	输入
height	视频图像高度	输入
fps	视频图像帧率	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

1、接收视频录制前必须调用此函数设置视频格式，否则会造成视频无法播放。

【参数相关】

无

CHD_WMP_Mjpeg2Avi_End

【描述】

设置视频文件参数。

【语法】

```
CHD_RET_E CHD_WMP_Mjpeg2Avi_End(IN CHD_WMP_T handle);
```

【参数】

参数名称	描述	输入/输出
handle	设备操作句柄	输入

【返回值】

返回值	描述
CHD_RET_SUCCESS	成功
CHD_RET_FAILED	失败
CHD_RET_HANDLE_ERROR	句柄错误

【注意】

- 1、结束录制前必须调用CHD_WMP_Mjpeg2Avi_SetParam()函数设置视频格式，否则会造成视频无法播放。

【参数相关】

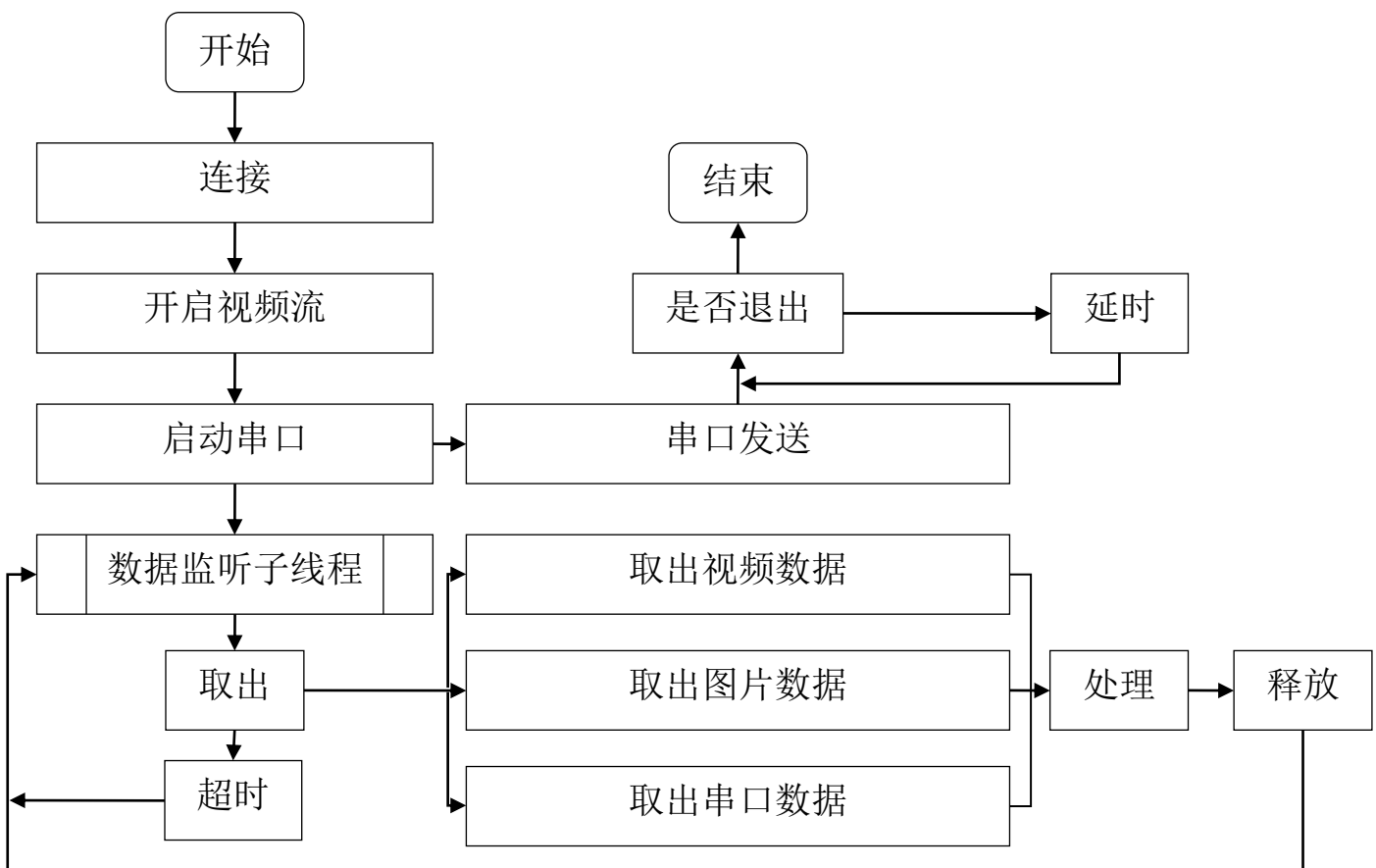
无

10. 简单示例

10.1 内容简介

- 1、SDK视频、串口通信最简流程，以下程序为**Windows**测试程序，如想在Linux、IOS、Mac上使用，只需要把测试程序中的线程创建函数改为其他平台的线程创建函数即可。
- 2、直接连接设备AP，则设备的IP地址为192.168.100.254。

10.2 程序流程图



10.3 测试程序

```
/*-----*
 * Copyright (C), Hangzhou Chird Technology Co., Ltd
 *-----*/

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <windows.h>
#include <process.h>
#include "CHD_WMP.h"
#include "CHD_WMP_Type.h"
#include "CHD_WMP_Video.h"
#include "CHD_WMP_Serial.h"
#include "CHD_WMP_Return.h"

int thread_run = 0;
CHD_WMP_T handle = 0;
unsigned __stdcall chird_thread( void *arg)
{
    CHD_RET_E ret;
    CHD_U32 datalen;
    CHD_STREAM_TYPE_E type;
    CHD_VIDEO_FRAME_S *pvdata;
    CHD_VOID *psdata;

    while(thread_run){
/* 数据监听 */
        ret = CHD_WMP_Poll(handle, &type, 2, 0);
/* 设备已断开连接 */
        if (ret == CHD_RET_DISCONNECT) break;
        if (ret != CHD_RET_SUCCESS) continue;
```

```

switch(type) {
    case CHD_STREAM_VIDEO : /* 视频 */
        ret=CHD_WMP_Video_RequestVideoData(handle, &pvdata);
        if (ret != CHD_RET_SUCCESS) break;
// 视频数据处理(显示、保存、图像识别等)
        printf("----->video<-----\n");
        CHD_WMP_Video_ReleaseVideoData(handle, pvdata);
    case CHD_STREAM_PICTURE : /* 图片 */
        ret = CHD_WMP_Video_RequestPicData(handle, &pvdata);
        if (ret != CHD_RET_SUCCESS) break;
// 图片数据处理 (显示、保存、图像识别等)
        printf("----->picture<-----\n");
        CHD_WMP_Video_ReleasePicData(handle, pvdata);
    case CHD_STREAM_SERIAL : /* 串口 */
        ret = CHD_WMP_Serial_RequestData(handle, &psdata, \
&datalen);
        if (ret != CHD_RET_SUCCESS) break;
// 串口数据处理
        printf("----->serial<-----\n");
        CHD_WMP_Serial_ReleaseData(handle, psdata);
    default: break;
}

return 0;
}

int main(int argc, char* argv[])
{
    CHD_RET_E ret;

    ret = CHD_WMP_ConnectDevice(&handle, "192.168.100.254");
    if (ret != CHD_RET_SUCCESS) {

```

```

    printf("connect fail, error code:%d\n",ret);
    return -1;
}

ret = CHD_WMP_Video_Begin(handle);
if (ret != CHD_RET_SUCCESS){
    printf("open video stream fail, error code:%d\n",ret);
    return -1;
}

ret = CHD_WMP_Serial_Begin(handle);
if (ret != CHD_RET_SUCCESS){
    printf("open serial fail, error code:%d\n",ret);
    return -1;
}

/* 创建视频、图片、串口数据处理线程 */
thread_run = 1;
HANDLE thread = (HANDLE)_beginthreadex(NULL, 0, chird_thread, \
NULL, 0, NULL);

CHD_WMP_Serial_SendData(handle, "12345678", 8);
CHD_WMP_Serial_SendData(handle, "12345678", 8);

while (getchar() != 'q') Sleep(1000);

thread_run = 0;
Sleep(5000);
/* 断开设备连接 */
CHD_WMP_Disconnect(handle);

return 0;
}

```